# DFD: Distilling the Feature Disparity Differently for Detectors

**Kang Liu** [1 2] **Yingyi Zhang** [3] **Jingyun Zhang** [2] **Jinmin Li** [2] **Jun Wang** [2] **Shaoming Wang** [2] **Chun Yuan** [1 †]
**Rizen Guo** [2 †]

## Abstract

Knowledge distillation is a widely adopted model compression technique that has been successfully applied to object detection. In feature distillation, it is common practice for the student model to imitate the feature responses of the teacher model, with the underlying objective of improving its own abilities by reducing the disparity with the teacher. However, it is crucial to recognize that the disparities between the student and teacher are inconsistent, highlighting their varying abilities. In this paper, we explore the inconsistency in the disparity between teacher and student feature maps and analyze their impact on the efficiency of the distillation. We find that regions with varying degrees of difference should be treated separately, with different distillation constraints applied accordingly. We introduce our distillation method called Disparity Feature Distillation(DFD). The core idea behind DFD is to apply different treatments to regions with varying learning difficulties, simultaneously incorporating leniency and strictness. It enables the student to better assimilate the teacher's knowledge. Through extensive experiments, we demonstrate the effectiveness of our proposed DFD in achieving significant improvements. For instance, when applied to detectors based on ResNet50 such as RetinaNet, FasterRCNN, and RepPoints, our method enhances their mAP from $37.4\%, 38.4\%, 38.6\%$ to $41.7\%, 42.4\%, 42.7\%$, respectively. Our approach also demonstrates substantial improvements on YOLO and ViT-based models. The code is available at https://github.com/luckin99/DFD.

## 1. Introduction

The ongoing advancements in deep learning have prompted increasingly deep and wide models to tackle progressively complex tasks. Correspondingly, these larger models come with an increase in the number of parameters and computational complexity. This results in more time consumption and memory usage, posing great challenges for real-time applications in various industries. Object detection is a computationally intensive task that requires processing large amounts of data in real-time, thus the inference speed and size of the parameters are crucial. Knowledge distillation is an important technique to address these challenges. By introducing additional supervision provided by a complex teacher model, the performance of the compact student model will be improved with knowledge distillation. Many approaches(Romero et al., 2014; Guo et al., 2021; Yang et al., 2022b; Cao et al., 2022) have achieved significant improvements in detectors by supervising the student using intermediate features from the teacher. These methods generally enhance the performance of the student by enforcing it to mimic the teacher's feature. Hence, the difference in feature responses is crucial for the effectiveness of feature distillation. To investigate this issue, we visualize the spatial

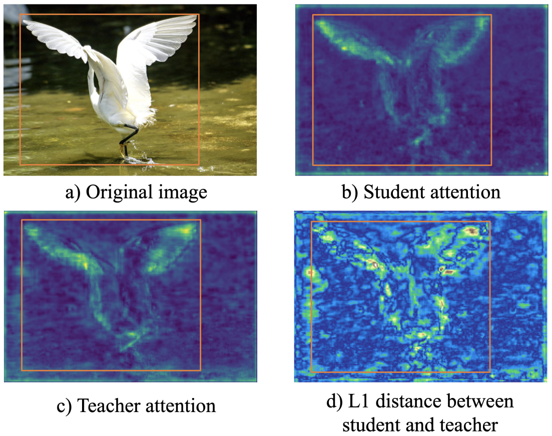


*Figure 1.* Visualization of spatial attention. We use RetinaNet with ResNeXt101 as the teacher model and RetinaNet with ResNet50 as the student model. We use a different color bar to visualize the L1 distance between teacher's and student's spatial attention.

[1] Tsinghua University [2] Tencent WeChat Pay Lab33 [3] Tencent Youtu Lab. Work was done when Kang Liu was an intern at Tencent WeChat Pay Lab33. † Correspondence to: Chun Yuan <yuanc@sz.tsinghua.edu.cn>, Rizen Guo <rizenguo@tencent.com>.

*Proceedings of the $41^{st}$ International Conference on Machine Learning*, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

attention of the teacher and student model in Fig.1.

Firstly, we observe that both the student and teacher models generate strong responses at specific regions in Fig.1(b)(c), with the majority of attention concentrated in small areas within the foreground bounding boxes. Additionally, certain regions in the background also exhibit a noticeable level of attention. The spatial distribution information is crucial for accurate detection, as the specific regions of response aid the detectors in accurately localizing the target objects. These visualization results differ from those mentioned in FGD(Yang et al., 2022b) and Defeat(Guo et al., 2021), as they distinguish between foreground and background based on the bounding boxes and perform distillation separately for each. The response of the feature maps not only lacks a one-to-one correspondence with the foreground-background distinction but also lacks correlation with the refined target mask region in the image. The distinction between foreground and background is based on the perspective of object detection rather than knowledge transfer. Secondly, by calculating the L1 distance between the attention maps of the student and teacher models in Fig.1(d), we observe that the disparities in their responses are more intricate. There are notable differences in the intensity of disparity at various locations, which are present in both foreground and background regions. This disparity in response distribution reflects, to some extent, the varying capabilities between the student and teacher models. The disparities in capabilities make it challenging for students to learn under overly strict constraints, prompting the use of weaker constraints in some methods(Cao et al., 2022; Yang et al., 2020) to enhance distillation performance. However, there is a question regarding whether weaker constraints can lead to the development of lazy students.

*Table 1.* Comparison of distillation on different regions. Teacher: RetinaNet-ResNeXt101. Student: RetinaNet-ResNet50. **HD**: High disparity region. **LD**: Low disparity region. **Split**: Split these regions, and use different weights for the distillation losses of different parts. **DC**: Using different constraints for different regions.

| Model | HD | LD | Split | DC | mAP |
|---|---|---|---|---|---|
| RetinaNet<br>RX101<br>-R50 | - | - | - | - | 36.4 |
| | ✓ | - | - | - | 39.8 |
| | - | ✓ | - | - | 38.2 |
| | ✓ | ✓ | - | - | 39.6 |
| | ✓ | ✓ | ✓ | - | 40.0 |
| | ✓ | ✓ | ✓ | ✓ | **40.4 (Ours)** |

To investigate whether these observation impact the distillation performance, we conducted a simple experiment as shown in Table.1. We divided the feature maps into two regions, namely high disparity regions(HD) and low disparity regions(LD), based on an L1 distance threshold of spatial attention. We applied the MSE loss to directly guide the student in mimicking the teacher's features in different regions. Surprisingly, distilling only the high disparity regions outperformed distilling the entire feature map without differentiation, which indicates that these two regions hold different significance. The high disparity regions highlight the differences in capabilities between the teacher and the student, we conducted separate distillation on these two regions with a higher weight assigned to the high disparity regions. We can find that the performance is improved from $39.6$ to $40.0$ mAP($+0.4$). Taking into consideration that the high disparity regions pose greater learning challenges and to some extent reflect the structural differences within the models themselves, we applied weaker constraints specifically to these regions while maintaining strict constraints(MSE) on the low disparity regions.By leveraging this different constraints, we further improve the performance to $40.4$ mAP($+0.8$). These results convincingly demonstrate that the varying difficulty levels across these regions significantly impact the efficacy of distillation. Based on the aforementioned observations, we proposed our Disparity Feature Distillation(DFD), a distillation method that combines both leniency and strictness simultaneously. DFD have devised distinct distillation strategies for different regions, taking into account the disparities in feature maps between the teacher and student models. For high disparity regions, we diminish the requirement for students to learn exclusively from the teachers' features by employing weaker supervision through a learnable transformation module. These disparities indicate differences in ability between the student and teacher models. This approach aims to prevent students from being misguided by overly strict constraints. For regions with low disparities, where students are capable of fully acquiring knowledge to some extend, we employ strict constraints to ensure that students learn as comprehensively as possible. We conducted experiments using our method on various detectors and achieved state-of-the-art (SOTA) performance. In summary, the contributions of this paper can be outlined as follows:

- We investigated the characteristics of response distributions between the student and teacher models and explored the impact of spatial disparities in responses on the efficiency of knowledge distillation.

- We proposed DFD, a distillation method that combines both leniency and strictness simultaneously. This approach helps the student better learn the teacher's knowledge by utilizing different learning strategies based on varying difficulty levels.

- We validated the performance of DFD on multiple detectors and achieved SOTA on the COCO dataset. Additionally, we extended the application of DFD to segmentation and pose estimation, demonstrating the scalability of DFD.

## 2. Related Work

### 2.1. Object Detection

Object detection is a challenging and essential computer vision task. It requires detecting foreground targets in the images and assigning correct category labels to these targets. Most detectors can be divided into two types, one-stage detectors and two-stage detectors. For one-stage detectors such as RetinaNet(Lin et al., 2017), Reppoints(Yang et al., 2019) and YOLO(Redmon et al., 2016), the detection head will directly perform bounding box regression and classification on the feature maps. In contrast, two-stage detectors like Faster-RCNN(Ren et al., 2015) and Mask-RCNN(He et al., 2017) generate candidate regions through Region Proposal Network(RPN) before the detector head. As a general distillation method, our method can be applied to both types of detectors.

### 2.2. Knowledge Distillation

By extracting additional information from the teacher network as supervision signals, knowledge distillation can help student models achieve better performance. Teacher models usually have stronger performance but with heavy parameters and slower running speeds. Student models are usually lighter and faster than teacher models but with weaker performance. Knowledge distillation be categorized into logits-based(Hinton et al., 2015), feature-based(Romero et al., 2014; Yang et al., 2023b; Liu et al., 2023a), and relation-based distillation(Park et al., 2019; Yang et al., 2022a). Fitnet(Romero et al., 2014) first used feature maps for distillation. Many knowledge distillation algorithms are designed for image classification(Yang et al., 2023a; Zhao et al., 2022). Chen et al.(Chen et al., 2017) first introduced knowledge distillation into object detection, including the distillation of feature maps and detector heads. GID(Dai et al., 2021) distilled inconsistent object instances detected by students and teachers. Sun et al. (Sun et al., 2020) proposed a distillation method that simultaneously distills features, classification heads, and bounding box regression heads. Some distillation methods noticed the inconsistency of foreground and background in object detection. Defeat(Guo et al., 2021) tried to distill the foreground and background with different weights. FGD(Yang et al., 2022b) used annotation region masks to divide foreground and background regions for distillation with different weights and introduced global distillation. However, these methods only considered this issue from the perspective of object detection but did not consider it from the perspective of distillation. Some methods have been proposed to impose weak constraints to prevent excessive constraints from misleading students' training. PKD (Cao et al., 2022) helps students learn better from stronger teachers by normalizing the feature maps of student and teacher. MGD(Yang et al., 2022c) restores randomly masked features under the guidance of the teacher, allowing students to have more freedom to learn. While these methods impose more relaxed constraints on students, it may lead to the risk of training a lazy student.

## 3. Method

### 3.1. Preliminaries

The feature-based distillation method has been widely utilized in object detection(Romero et al., 2014). Typically, these methods extract feature maps generated by the intermediate layers of the teacher networks, which are used as additional supervisory information. Generally, The feature-based distillation methods can be formulated as:

$$L_{fea} = \sum_{k}^{C} \sum_{i}^{H} \sum_{j}^{W} \left(F_{i,j,k}^{T} - F_{i,j,k}^{S}\right)^2 \tag{1}$$

where $L_{fea}$ represents the distillation loss of feature mimicking, $F^T$ and $F^S$ are the feature maps from teacher and student network. $C, H, W$ denote the channel number, height, and width of the feature map.

### 3.2. Disparity Feature Distillation

The overall architecture of disparity feature distillation(DFD) is illustrated in Figure 2. Specifically, we extract the feature maps $F^T$ and $F^S$ and then proceed to calculate the spatial attention of both the teacher and student models:

$$A(F) = H \cdot W \cdot \text{softmax}\left(\frac{1}{C} \cdot \sum_{c=1}^{C} |F|\right) \tag{2}$$

By compressing channel information, we obtained spatial attention maps $A(F^T)$ and $A(F^S)$ of size $N \times H \times W$. Here, $A(F)$ is a computational formula where the input feature is denoted as $F$. Then we calculate the L1 distance to obtain the spatial attention disparity map:

$$D = \left|A(F^T) - A(F^S)\right| \tag{3}$$

$D$ indicates the difference in spatial attention distribution between the student and teacher with shape $N \times H \times W$. We use the mean of the disparity map $D$ as the threshold $P$ to partition the regions, which allows the binary map to change during the training process dynamically.

$$Mask_{i,j} = \begin{cases} 0, \text{if } D_{i,j} < P \\ 1, \text{ Otherwise} \end{cases} \tag{4}$$

Based on this adaptive binary map, features are divided into two parts, namely high disparity regions $R_{HD}$ and low disparity regions $R_{LD}$. In order to better convey information from the teacher, a learning strategy that combines leniency

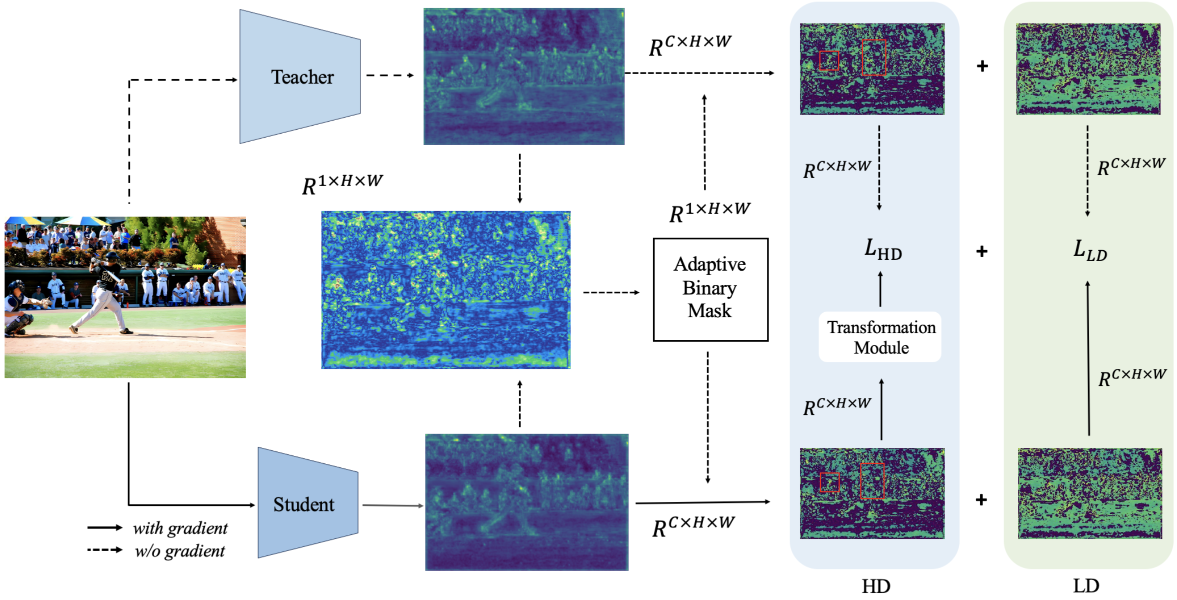


*Figure 2.* An illustration of over all architecture of our DFD. We calculate the L1 distance of spatial attention between the teacher and student model, and use its mean as the threshold to divide it into high disparity regions(HD) and low disparity regions(LD). Then we use different constraint methods to calculate distillation losses for these two regions separately. The specific structure of the transformation module is shown in Figure.3.

and strictness is used. For these two parts, $R_{HD}$ and $R_{LD}$, strict supervision and weak supervision were selected to be provided. The distillation losses can be described as:

$$L_{HD} = \sum_{k}^{C}\sum_{i}^{H}\sum_{j}^{W}\left(F^{T}_{i,j,k} - f_{trans}(F^{S}_{i,j,k})\right)^{2}, i,j\epsilon R_{HD} \tag{5}$$

$$L_{LD} = \sum_{k}^{C}\sum_{i}^{H}\sum_{j}^{W}\left(F^{T}_{i,j,k} - F^{S}_{i,j,k}\right)^{2} i,j\epsilon R_{LD} \tag{6}$$

where $L_{HD}$ and $L_{LD}$ refer to the loss functions computed separately for the high disparity and low disparity regions. As illustrated in Figure.2, we employ the Mean Squared Error (MSE) loss as a strict constraint in a point-by-point manner for low disparity regions(LD). Simultaneously, we pass the student's features through a learnable transformation module $f_{trans}$ then employ MSE loss to align them with the teacher's features, establishing our weak constraint for high disparity regions(HD). This module can be trained along with the student network to form appropriate constraints during the training process adaptively. We will further discuss these in ablation studies.

### 3.3. Overall Loss

By applying our DFD, the overall loss can be formulated as:

$$L_{total} = L_{task} + \alpha \cdot L_{HD} + \beta \cdot L_{LD} \tag{7}$$

where $L_{task}$ is the training loss of the student in the specific task, $\alpha$ and $\beta$ are two weight coefficients used to control the relative weight of different distillation losses. Our method is simple and general and contains only two hyper parameters.

## 4. Experiments

### 4.1. Experiment Setup

To evaluate the effectiveness of our method, we conduct comprehensive experiments on the COCO2017 dataset (Lin et al., 2014) using 8 Tesla V100 GPUs. This dataset comprises 80 object categories, and we use the default split of 120k images for training and 5k images for testing. We report the mean Average Precision (mAP) as the evaluation metric. We train all detectors for 24 epochs (2x schedule) or 12 epochs (1x schedule) with the stochastic gradient descent (SGD) optimizer. The optimizer is configured with a momentum of 0.9 and a weight decay of 0.0001. We select all the feature maps obtained after the neck of each model for distillation. Our implementation is based on MMDetection (Chen et al., 2019) with Pytorch (Paszke et al., 2019) framework, and we follow the default train-

*Table 2.* The main results of different kinds of detectors on COCO dataset. RetinaNet: anchor-based one-stage detector. Faster-RCNN: two-stage detector. RepPoints: anchor-free one-stage detector.

| Teacher | Method | schedule | mAP | AP s | AP m | AP L |
|---|---|---|---|---|---|---|
| RetinaNet ResNext101 | *RetinaNet ResNet50* | *2x* | *37.4* | *20.0* | *40.7* | *49.7* |
| | FGD(Yang et al., 2022b) | 2x | 40.9 | 23.1 | 45.1 | 54.9 |
| | MGD(Yang et al., 2022c) | 2x | 41.2 | 23.6 | 45.3 | 54.6 |
| | PKD(Cao et al., 2022) | 2x | 41.2 | 23.0 | 45.4 | 55.6 |
| | **Ours** | 2x | **41.7(+4.0)** | **24.4** | **46.0** | **55.7** |
| CascadeMaskRCNN ResNeXt 101 | *FaterRCNN ResNet 50* | *2x* | *38.4* | *21.5* | *42.1* | *50.3* |
| | FGD(Yang et al., 2022b) | 2x | 42.0 | 23.8 | 46.4 | 55.5 |
| | MGD(Yang et al., 2022c) | 2x | 42.1 | 23.7 | 46.4 | 56.1 |
| | PKD(Cao et al., 2022) | 2x | 41.4 | 22.7 | 45.1 | 56.0 |
| | **Ours** | 2x | **42.4(+4.0)** | **23.9** | **46.8** | **56.3** |
| Reppoints ResNeXt101 | *Reppoints ResNet 50* | *2x* | *38.6* | *22.5* | *42.2* | *50.4* |
| | FGD(Yang et al., 2022b) | 2x | 42.0 | 24.0 | 45.7 | 55.6 |
| | MGD(Yang et al., 2022c) | 2x | 42.3 | 24.4 | 46.2 | 55.9 |
| | PKD(Cao et al., 2022) | 2x | 42.4 | 24.3 | 46.7 | **56.4** |
| | **Ours** | 2x | **42.7(+4.1)** | **24.8** | **46.7** | 56.2 |

ing settings of MMDetection. For YOLO experiments, we use MMYOLO(Contributors, 2022) framework. The inheriting strategy (Kang et al., 2021) is an effective method that can improve the student network's convergence performance and speed without introducing additional computing costs. In the main experiments, we use this strategy to initialize the student which has the same head structure as the teacher. For all one-stage detectors, we use $\alpha = 0.000028$ and $\beta = 0.00001$. For two-stage detectors, we use $\alpha = 0.00000035$ and $\beta = 0.0000001$.

### 4.2. Main Results

We conduct experiments on three different types of detectors to verify the performance of our method. We select several methods (Yang et al., 2022b;c; Cao et al., 2022) with outstanding performances for comparison. For anchor-based one-stage detector RetinaNet(Lin et al., 2017), we choose RetinaNet with ResNext101 as the teacher and RetinaNet with ResNet50 as the student. Our method improve the performance of the student network from 37.4 mAP to 41.7 mAP. We achieve a gain of + 4.0 mAP. For anchor-free detector RepPoints(Yang et al., 2019), our method improves the performance of the student model from 38.6 mAP to 42.7 mAP, achieving a gain of + 4.1 mAP. Our method is also applicable to two-stage detectors. We improve the performance of FasterRCNN-Res50 from 38.4 mAP to 42.4 mAP, achieving a gain of + 4.0 mAP. These results demonstrate that our DFD can better transfer the teacher's knowledge to the students and can be applied to various types of detectors.

### 4.3. Experiments of Other Detectors

To further demonstrate the performance of our method, we selected some other networks for experiment as shown in Table.3. We first test our method on the different backbones, we choose Swin Transformer (Liu et al., 2021) for experiment. When using MaskRCNN with Swin-Small, our method improved MaskRCNN with Swin-Tiny from 42.7 mAP to 44.4 mAP. These results demonstrate that our DFD is not only effective for CNN-based models and applicable to transformer-based models. YOLOv6(Li et al., 2022) is an advanced detector that exhibits remarkable performance and fast inference speed. We select YOLOv6-Small as the teacher and YOLOv6-Tiny as the student. Our method improve the student's performance from 40.6 mAP to 41.7 mAP. These results demonstrate the robust performance and versatility of our approach.

*Table 3.* Experiments of other detectors on COCO dataset. We performed MaskRCNN based experiments baesd on MMDetection(Chen et al., 2019) and YOLOv6 experiments based on MMYOLO(Contributors, 2022).

| Method | Schedule | mAP |
|---|---|---|
| MaskRCNN-SwinS(Teacher) | 1x | 48.2 |
| MaskRCNN-SwinT(Student) | 1x | 42.7 |
| PKD | 1x | 43.9 |
| **Ours** | 1x | **44.4** |
| Method | Epoch | mAP |
| YOLOv6-Small(Teacher) | 400 | 44.0 |
| YOLOv6-Tiny(Student) | 300 | 40.6 |
| PKD | 300 | 41.3 |
| **Ours** | 300 | **41.7** |

*Table 4.* Experiments with progressively stronger teacher. Faster-RCNN with ResNet50 is uniformly used as the student model. We train all models with 1x schedule on COCO dataset.

| Teacher | Method | Schedule | mAP |
|---|---|---|---|
| FasterRCNN R101<br>39.8 | *student* | *1x* | *37.4* |
| | PKD | 1x | 39.6 |
| | **Ours** | 1x | **39.7** |
| FasterRCNN Rx101<br>41.2 | PKD | 1x | 40.0 |
| | **Ours** | 1x | **40.5** |
| MaskRCNN Rx101<br>42.2 | PKD | 1x | 40.5 |
| | **Ours** | 1x | **41.1** |

*Table 5.* Test results of different transformation modules. We train all models with 1x schedule on COCO dataset and we apply these modules to the whole feature maps.

| Teacher | Method | Schedule | mAP |
|---|---|---|---|
| RetinaNet<br>ResNext101 | *RetinaNet R50* | *1x* | *36.5* |
| | Conv1-1 | 1x | 40.1 |
| | Conv3-3 | 1x | 40.1 |
| | Conv1-3-1(0.5C) | 1x | 39.8 |
| | Conv1-3-1(C) | 1x | 40.1 |
| | **Conv1-3-1(2C)** | 1x | **40.2** |

### 4.4. Experiments with Progressively Stronger Teachers

To test the applicability of our method, we conducted experiments using different teacher-student pairs. The results are shown in Table 4. We used FasterRCNN- ResNet50 as the student model for all experiments and tested our method with various teacher models. We first used Faster-RCNN as the teacher. When using the backbone network ResNet101 and ResNext101, our method improved the student by +2.3 mAP and +3.1 mAP, respectively. When using MaskRCNN-ResNext101, our method improved the student's performance by +3.7 mAP. Our results surpassed those of PKD for all teacher-student pairs.

### 4.5. Design of Transformation Module

To establish a suitable weak constraint, we choose to design a transformation module to implement it. This module can be trained along with the student network to form appropriate constraints during the training process adaptively. We test several alignment modules, and their performances are shown in Table 5. To avoid the transformation module from requiring excessive training and interfering with the student network's training, we only experiment on compact convolution-based transformation modules. We use the most fundamental $1 \times 1$ convolutions and $3 \times 3$ convolutions to compose the transformation module. ReLU are applied between different convolutional layers. For example, as shown in Table 5, *Conv1-1* represents the sequential use of two $1 \times 1$ convolutions, with *ReLU* added between the convolutional layers. As shown in Figure 3 (a), *Conv1-3-1(2C)* represents the sequential use of $1 \times 1$ convolution, $3 \times 3$ convolution, and $1 \times 1$ convolution. Simultaneously, the channel size is expanded to *2C* during the first $1 \times 1$convolution and then transformed back to *C* during the last $1 \times 1$ convolution. According to the experimental results in Table.5, we ultimately choose to use *Conv1-3-1(2C)* as our transformation module. We also measured the L2 distance between student and teacher features after using this transformation module and MSE loss, which are shown in Figure 3 (b). We can find that through this transformation module, students can adaptively establish constraints with teachers and retain a certain level of disparity.

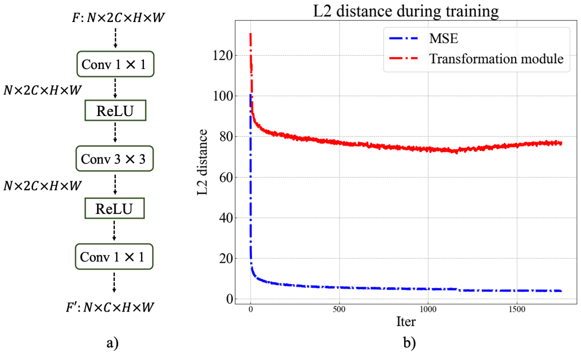


*Figure 3.* Experiment of transformation module. (a). The structure of the transformation module we used. (b). L2 distance during the training.

### 4.6. Comparison with Latest Methods

In order to compare the performance of DFD, we selected several latest relevant methods for comparison. For fair comparison, we report their original results and replicate ours under their settings. As shown in Table.6, DFD outperforms DiffKD(Huang et al., 2023) and CTFD(Liu et al., 2023b) in both RetinaNet and Reppoints.

*Table 6.* Comparison with latest methods.

| Model | Method | mAP |
|---|---|---|
| RetinaNet<br>ResNeXt101<br>-ResNet50 | DiffKD(Huang et al., 2023) | 40.7 |
| | CTFD(Liu et al., 2023b) | 41.0 |
| | **Ours** | **41.2** |
| RepPoints<br>ResNeXt101<br>-ResNet50 | DiffKD(Huang et al., 2023) | 41.7 |
| | CTFD(Liu et al., 2023b) | 42.0 |
| | **Ours** | **42.5** |

*Table 7.* Results of pose estimation on COCO-Body and segmentation on Cityscapes.

| Pose estimation | Method | Input Size | mAP |
|---|---|---|---|
| Heatmap Res50 | Teacher | 256 × 192 | 71.8 |
| Heatmap MobileNetV2 | student | 256 × 192 | 62.0 |
| | CWD | 256 × 192 | 62.2 |
| | **Ours** | **256 × 192** | **62.6** |
| Segmentation | Method | Input Size | mAP |
| PspNet Res101 | Teacher | 512 × 512 | 78.34 |
| PspNet Res18 | student | 512 × 512 | 69.85 |
| | CWD | 512 × 512 | 73.53 |
| | **Ours** | **512 × 512** | **73.74** |

*Table 8.* The performance of using different constraints in different regions. Teacher: RetinaNet ResNeXt101. Student: RetinaNet ResNet101. **TF**: Using transformation module. **HD**: high disparity regions. **LD**: low disparity regions.

| HD | LD | Schedule | mAP |
|---|---|---|---|
| MSE | MSE | 1x | 39.7 |
| TF | TF | 1x | 40.2 |
| TF | - | 1x | 39.9 |
| - | TF | 1x | 39.4 |
| MSE | TF | 1x | 39.9 |
| TF | TF + MSE | 1x | 40.2 |
| **TF** | **MSE** | 1x | **40.4** |

### 4.7. Task Extension and FGD Comparison

DFD partitions regions by computing the differences in features, allowing it to be applied to various tasks. FGD(Yang et al., 2022b) relies on annotations to compute foreground regions, making it challenging to extend its applicability. Pose estimation and segmentation are tasks that also require attention to spatial distribution information. Our method can be effectively extended to these tasks, as shown in Table.7. DFD demonstrates excellent performance on heatmap-based models for pose estimation tasks on the COCO-Body dataset. Additionally, it shows significant improvements on segmentation models when evaluated on the Cityscape dataset(Cordts et al., 2016). It is worth noting that due to the need to compute the annotation bounding boxes regions, FGD requires a longer computation time. Under the same framework and environment, for Reppoints-Rx101 distills Reppoints-R50, FGD takes nearly 29 hours, while our method only requires nearly 18 hours.

*Table 9.* Performance of different strict constraint methods. Student: RetinaNet ResNet50. $F^T$and $F^S$ represent the feature maps of the teacher and the student, respectively.

| Teacher | Method | Schedule | mAP |
|---|---|---|---|
| RetinaNet ResNext101 | $\sqrt{\lvert F^T - F^S \rvert}$ | 1x | 40.0 |
| | $\lvert F^T - F^S \rvert$ | 1x | 40.2 |
| | $\mathbf{\lvert F^T - F^S \rvert^2}$ | **1x** | **40.4** |
| | $\lvert F^T - F^S \rvert^4$ | 1x | 40.1 |

*Table 10.* Experiments of combining our method with other methods on COCO dataset. Teacher: RetinaNet ResNet101. Student: RetinaNet Res50.

| Teacher | Method | Schedule | mAP |
|---|---|---|---|
| RetinaNet ResNext101 | MGD | 1x | 40.0 |
| | MGD + Ours | 1x | 40.3 |
| | PKD | 1x | 39.9 |
| | PKD + Ours | 1x | 40.3 |
| | PKD + MGD | 1x | 39.9 |

### 4.8. Combining Different Approaches

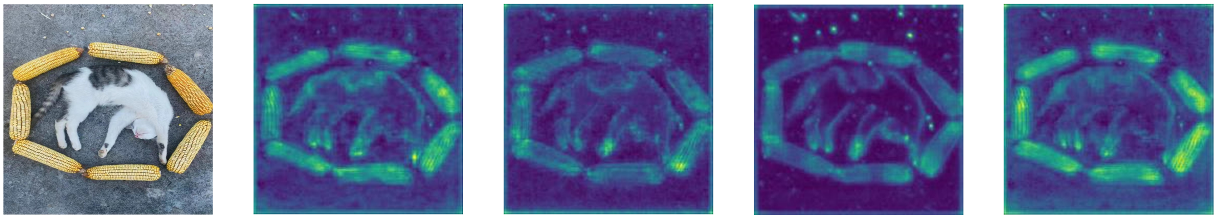
In this section, we will elaborate on how we combine different constraint methods to achieve better performance. As shown in Table.8. The performance of using MSE and transformation module separately is 39.7 mAP and 40.2 mAP. The results of using the transformation module in significance and low disparity regions were 39.9 mAP and 39.4 mAP, respectively. We replace the high disparity area with MSE, resulting in a decrease in performance. However, using the transformation module in all regions while using MSE for additional constraints in low disparity regions resulted in a performance of 40.2 mAP. The best result is achieved by using the transformation module and MSE in high and low disparity regions, respectively. Therefore, we find that combining weak and strict constraint methods can help students better learn teachers' knowledge. We also test the performance of different strict constraint methods, as shown in Table.9. We consider not performing any operations and directly constraining the feature maps of teachers and students point by point. According to Table.9, using L2 distance, i.e., MSE Loss, produce the best results.

*Table 11.* Performance of different distance function methods.

| Teacher | Method | Schedule | mAP |
|---|---|---|---|
| RetinaNet ResNext101 | *RetinaNet R50* | *1x* | *36.5* |
| | **L1 Distance** | **1x** | **40.4** |
| | L2 Distance | 1x | 40.2 |

### 4.9. Combining DFD with Other Methods

Several distillation methods have been proposed to prevent excessive constraints from misguiding student training. For example, PKD (Cao et al., 2022) normalizes feature maps to help students learn better from a stronger teacher. MGD (Yang et al., 2022c) restores randomly masked features under the teacher's guidance. Here we combine PKD and MGD with our methods by utilizing them as weak constraint methods. As shown in Table.10, when replacing the



Original image Teacher Student With MGD With our method

*Figure 4.* Visualization of spatial attention. Student: RetinaNet Res50. We visual the spatial attention of the teacher and student models. As a comparison, we also visualized the spatial attention of student after distillation with MGD and our method, respectively.

*Table 12.* Using different threshold functions. Here we use L1 distance maps as the difference map, and use different threshold functions as thresholds for region partitioning.

| Teacher | Method | Schedule | mAP |
|---|---|---|---|
| RetinaNet<br>ResNext101 | *RetinaNet R50* | *1x* | *36.5* |
| | Medium | 1x | 40.1 |
| | **Mean** | **1x** | **40.4** |

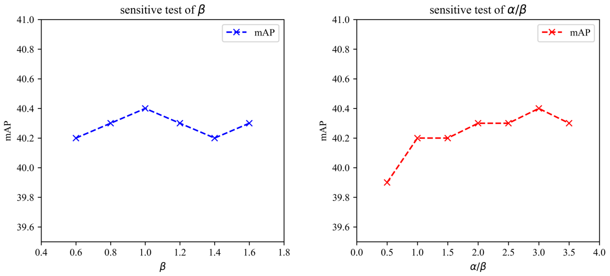


*Figure 5.* The results of sensitive test. Teacher: RetinaNet ResNeXt101. Student: RetinaNet ResNet101. We train the student model with 1x schedule on COCO dataset.

weak constraint method with MGD, we increased MGD by 0.3 mAP. When replacing the weak constraint method with PKD, we increased PKD by 0.4 mAP. At the same time, we also test using MGD as a strong constraint and PKD as a weak constraint and found that it can't bring any improvement, which further validates our idea.

### 4.10. Dsicussion of Area Selection Method

In this section, we discuss how to select different regions. We first consider using different distance functions to measure the differences between the feature map of the student and teacher, the results are shown in Table.11. By comparing the results, it can be found that the best performance is achieved when using L1 distance as the metric. Then we discuss what threshold function can achieving better performance, as shown in Table.12. In order to avoid introducing additional hyper parameters and to ensure that this threshold can adaptively change during training, we use different statistics as the discriminative threshold. We tested the mean and median for comparison. The results showed that using the mean as the threshold achieved the best performance.

### 4.11. Visualization of Attention after Distillation

Our method combines strong and weak constraints to help the student comprehensively learn the teacher's knowledge. To explore how our method affects the student's characteristics, we visualized the spatial attention of both the student and teacher, as shown in Figure.4. The initial student displays a noteworthy difference from the teacher, exhibiting incomplete and low-intensity responses to objects. After training with MGD, the student's features become more sparse and have a stronger response intensity at crucial points. However, the response to critical areas is still weak. After training with our method, we obtained a response in which the high attention areas matched the teacher's characteristics well. Moreover, our method exhibits a more concentrated and intense response on the main target objects compared to the teacher.

### 4.12. Sensitive Study of Hyper Parameter

Our method only has two hyper parameters, $\alpha$ and $\beta$. They control two impacts: the relative scale of distillation losses and task losses and the relative size of loss between high and low disparity regions. Here we use RetinaNet for experiments, the results are shown in Figure.5. We first conducted sensitivity testing on $\beta$. We fixed $\alpha/\beta$ to 3, and the results showed that our method is not sensitive to the scale of $\beta$. Then we set the $\beta$ to 0.00001 and adjust the relative size of $\alpha/\beta$. We find that the best results were achieved when $\alpha$ is larger than $\beta$, which means the high disparity regions are more important.

## 5. Conclusion

In this paper, we explore the imbalance of feature distillation in object detection, which refers to the disparity in feature map response between the student and the teacher. We investigated the impact of this difference on distillation efficiency and proposed our DFD based on this insight. We partition the regions based on the magnitude of the difference and applied distinct constraints for each of them. Our method has shown good distill performance on multiple detectors and achieved SOAT performance.

## Acknowledgements

This work was supported by the National Key R&D Program of China (2022YFB4701400/4701402), SSTIC Grant (KJZD20230923115106011), Shenzhen Key Laboratory (ZDSYS20210623092001004), and Beijing Key Lab of Networked Multimedia, and Tencent.

## Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

## References

Cao, W., Zhang, Y., Gao, J., Cheng, A., Cheng, K., and Cheng, J. Pkd: General distillation framework for object detectors via pearson correlation coefficient. *arXiv preprint arXiv:2207.02039*, 2022.

Chen, G., Choi, W., Yu, X., Han, T., and Chandraker, M. Learning efficient object detection models with knowledge distillation. *Advances in neural information processing systems*, 30, 2017.

Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y., Li, X., Sun, S., Feng, W., Liu, Z., Xu, J., et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

Contributors, M. Mmyolo: Openmmlab yolo series toolbox and benchmark, 2022.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3213–3223, 2016.

Dai, X., Jiang, Z., Wu, Z., Bao, Y., Wang, Z., Liu, S., and Zhou, E. General instance distillation for object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7842–7851, 2021.

Guo, J., Han, K., Wang, Y., Wu, H., Chen, X., Xu, C., and Xu, C. Distilling object detectors via decoupled features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2154–2164, 2021.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Huang, T., Zhang, Y., Zheng, M., You, S., Wang, F., Qian, C., and Xu, C. Knowledge diffusion for distillation. *arXiv preprint arXiv:2305.15712*, 2023.

Kang, Z., Zhang, P., Zhang, X., Sun, J., and Zheng, N. Instance-conditional knowledge distillation for object detection. *Advances in Neural Information Processing Systems*, 34:16468–16480, 2021.

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*, 2022.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

Liu, Z., Wang, Y., and Chu, X. A simple and generic framework for feature distillation via channel-wise transformation. *arXiv preprint arXiv:2303.13212*, 2023a.

Liu, Z., Wang, Y., Chu, X., Dong, N., Qi, S., and Ling, H. A simple and generic framework for feature distillation via channel-wise transformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1129–1138, 2023b.

Park, W., Kim, D., Lu, Y., and Cho, M. Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3967–3976, 2019.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

Sun, R., Tang, F., Zhang, X., Xiong, H., and Tian, Q. Distilling object detectors with task adaptive regularization. *arXiv preprint arXiv:2006.13108*, 2020.

Yang, C., Zhou, H., An, Z., Jiang, X., Xu, Y., and Zhang, Q. Cross-image relational knowledge distillation for semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12319–12328, 2022a.

Yang, J., Martinez, B., Bulat, A., and Tzimiropoulos, G. Knowledge distillation via softmax regression representation learning. In *International Conference on Learning Representations*, 2020.

Yang, Z., Liu, S., Hu, H., Wang, L., and Lin, S. Reppoints: Point set representation for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9657–9666, 2019.

Yang, Z., Li, Z., Jiang, X., Gong, Y., Yuan, Z., Zhao, D., and Yuan, C. Focal and global knowledge distillation for detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4643–4652, 2022b.

Yang, Z., Li, Z., Shao, M., Shi, D., Yuan, Z., and Yuan, C. Masked generative distillation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XI*, pp. 53–69. Springer, 2022c.

Yang, Z., Zeng, A., Li, Z., Zhang, T., Yuan, C., and Li, Y. From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. *arXiv preprint arXiv:2303.13005*, 2023a.

Yang, Z., Zeng, A., Yuan, C., and Li, Y. Effective whole-body pose estimation with two-stages distillation. *arXiv preprint arXiv:2307.15880*, 2023b.

Zhao, B., Cui, Q., Song, R., Qiu, Y., and Liang, J. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pp. 11953–11962, 2022.