arXiv:2108.12176v2 [cs.CV] 8 Sep 2021

Rethinking the Aligned and Misaligned Features in One-stage Object Detection

Yang Yang[1,2], **Min Li**[1,2], **Bo Meng**[3], **Junxing Ren**[1,2], **Degang Sun**[1,2], **Zihao Huang**[1,2],

[1]Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3]Beijing Institute of Technology, Beijing, China
{yangyang1995, limin, renjunxing, sundegang}@iie.ac.cn

Abstract

One-stage object detectors rely on a point feature to predict the detection results. However, the point feature often lacks the information of the whole object, thereby leading to a misalignment between the object and the point feature. Meanwhile, the classification and regression tasks are sensitive to different object regions, but their features are spatially aligned. Both of these two problems hinder the detection performance. In order to solve these two problems, we propose a simple and plug-in operator that can generate aligned and disentangled features for each task, respectively, without breaking the fully convolutional manner. By predicting two task-aware point sets that are located in each sensitive region, the proposed operator can align the point feature with the object and disentangle the two tasks from the spatial dimension. We also reveal an interesting finding of the opposite effect of the long-range skip connection for classification and regression. On the basis of the **O**bject-**A**ligned and **T**ask-disentangled operator (OAT), we propose OAT-Net, which explicitly exploits point-set features for accurate detection results. Extensive experiments on the MS-COCO dataset show that OAT can consistently boost different state-of-the-art one-stage detectors by $\sim$2 AP. Notably, OAT-Net with Res2Net-101-DCN backbone achieves 53.7 AP on the COCO *test-dev*.

Object detection is one of the fundamental tasks in the computer vision field, which can be divided into one-stage (Redmon et al. 2016; Law and Deng 2018; Tian et al. 2019) and two-stage ones (Ren et al. 2015; Lin et al. 2017a; Cai and Vasconcelos 2018; Zhang et al. 2020a). The two-stage detector predicts several regions of interest (RoIs) in the first stage. Then aligned RoI features are fed into the region convolutional network for classification and bounding box (i.e., bbox) regression. This operation can alleviate the feature misalignment problem and improve the detection accuracy at the cost of lower inference speed and more memory footprint. To obtain faster speed, the one-stage detector removes the RoI pooling process and directly gets the detection results. Therefore, the features it utilizes are misaligned with the matched object.

The main goal of object detection contains two tasks, one is to give the accurate location of the object on an image, and the other is to predict the category of the object. A recent academic study (Song, Liu, and Wang 2020) has shown that these two tasks require features from different locations.

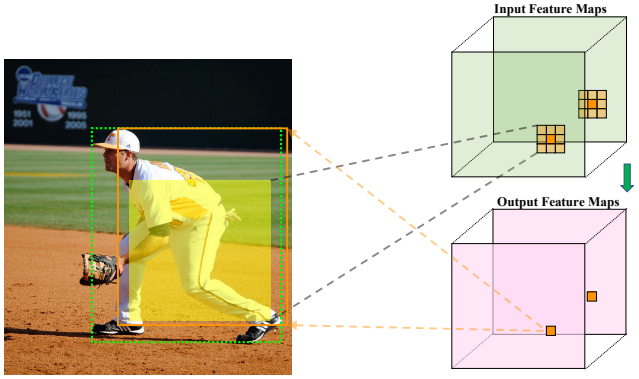


Figure 1: **Illustration of the misalignment problem.** The green box is the ground truth bbox, and the orange box is the prediction bbox. The yellow region is the coverage area where the nine points of the input feature maps are mapped to the image plane.

However, most detectors do not have solutions to disentangle both tasks spatially. These two tasks commonly share a coupled head or the same receptive field. Consequently, the input features for them are spatially aligned, thereby hindering the detection accuracy.

Misalignment. One-stage detectors, such as RetinaNet (Lin et al. 2017b) and FCOS (Tian et al. 2019) detect the object with a point-based feature. As shown in Figure 1, the predicted bbox is obtained by the orange point feature of the output feature maps, and this point feature is the convolution result of the nine-point features from the input feature maps (i.e., 3×3 convolution). If these points are mapped to the input image, then the yellow region is obtained. One can see that the mapping region does not cover the whole object. Thus, the point feature is misaligned with the object.

Guided Anchoring (Wang et al. 2019) is one of the earliest explorations to solve this problem. Instead of using preset anchors, it utilizes an anchor generation module to predict anchor shapes and then feeds them to a feature adaption module to extract aligned features. However, adjusting the preset anchors is a compromise because the misalignment with the object is the receptive field of the feature point.

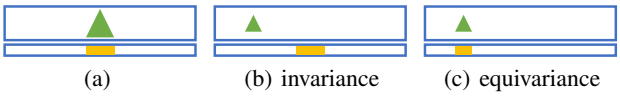


Figure 2: (a) is the control group. The input images of (b) and (c) have the location and size transformations.

Alignment. As illustrated in Figure 2, the first and second rows are the input images and their output features, respectively. Classification has translation and scale invariance, that is, the location and size transformations of the object do not affect the classification result (i.e., Figure 2 (b)). Nevertheless, regression has translation and scale equivariance, that is, the location and size transformations of the object have the same effect on the regression result (i.e., Figure 2 (c)). Their nature is so different that the features they utilize should be quite distinct.

Most detectors produce many redundant results. Therefore, non-maximum suppression (NMS) is usually applied to filter out poor detection results with the classification score as the ranking keyword. However, as illustrated by the second row of Figure 3, the prediction scores of these two tasks from the same location can be quite different. Therefore, if taking classification score as the only criterion in ranking, the best detection result (i.e., green box) will be filtered out by the NMS. The philosophy behind this problem is that the two tasks share an identical receptive field, or we can say they are spatially aligned (please refer to the appendix for details). IoU-net (Jiang et al. 2018) utilizes an extra head to predict the IoU between the regression result and the matched ground truth. Then, the IoU score is taken as the NMS ranking keyword. This approach does alleviate the alignment problem, but the features are still entangled because the regression and classification still share the coupled head and aligned receptive field.

In this paper, we revisit the aligned and misaligned features in object detection and propose a plug-in operator, namely, the object-aligned and task-disentangled operator (OTA). First, OTA can align the input features with the object following the propose-and-align mechanism. Instead of the bbox, we use the point set as the proposal to represent the object. OTA can sample features from the point set locations through bilinear interpolation to align them with the object. Second, to disentangle the regression and classification features, OTA adaptively learns to generate two sets of task-aware points. The features for the classification and regression heads are spatially disentangled by extending their convolution receptive fields to the locations of the matched task-aware points, respectively.

In summary, the key contributions of our work can be summarized as follows:

1. We propose an operator that can generate two task-aware semantic point sets, which represent the object to be localized and classified, respectively. Then aligned point features are extracted for each task. The proposed operator contains three modules: regression feature alignment (RFA), classification feature alignment (CFA), and spatial disentanglement (SD).
2. The proposed operator can be easily plugged into most one-stage object detectors and bring a considerable improvement of ∼2 AP.
3. Without bells and whistles, our best single-scale model (Res2Net-101-DCN) yields 51.4 AP on the COCO *test-dev* set and 53.7 AP with test-time augmentations, which are very competitive results among dense object detectors.

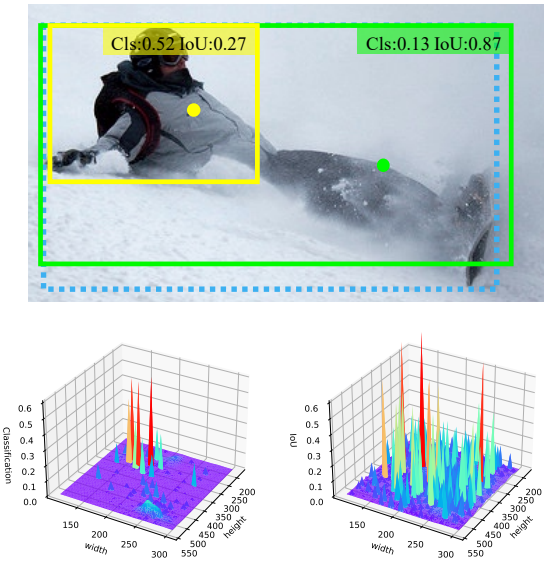


Figure 3: **Illustration of the alignment problem.** The blue box denotes the ground truth, and the other boxes are the detection results of ATSS (Zhang et al. 2020b). The two points are the locations where the detection results are predicted. "Cls" denotes the classification score, and "IoU" denotes the intersection over union between the predicted box and the ground truth. The second row is the distribution of classification and IoU scores.

Related Work

Misalignment. Guided Anchoring (Wang et al. 2019), RefineDet (Zhang et al. 2018), and SRN (Chi et al. 2019) learn an offset field for the preset anchor and then utilize a feature adaption module to extract features from the refined anchors. Following a similar paradigm, AlignDet proposes an RoI convolution to extract aligned features. All these works can achieve feature alignment but in a compromised manner because the misalignment is between the object and the corresponding feature point, not with the anchor. Instead of predicting a better anchor, RepPoints (Yang et al. 2019) and VFNet (Zhang et al. 2021) directly predict a coarse regression result in the first stage, and then the initial regression results are fed into the deformable convolution (Dai et al. 2017) to extract accurate point features. However, all the aforementioned methods share the same offset for regression and classification. Thus, the features for these tasks are spatially aligned, which leads to inferior performance.

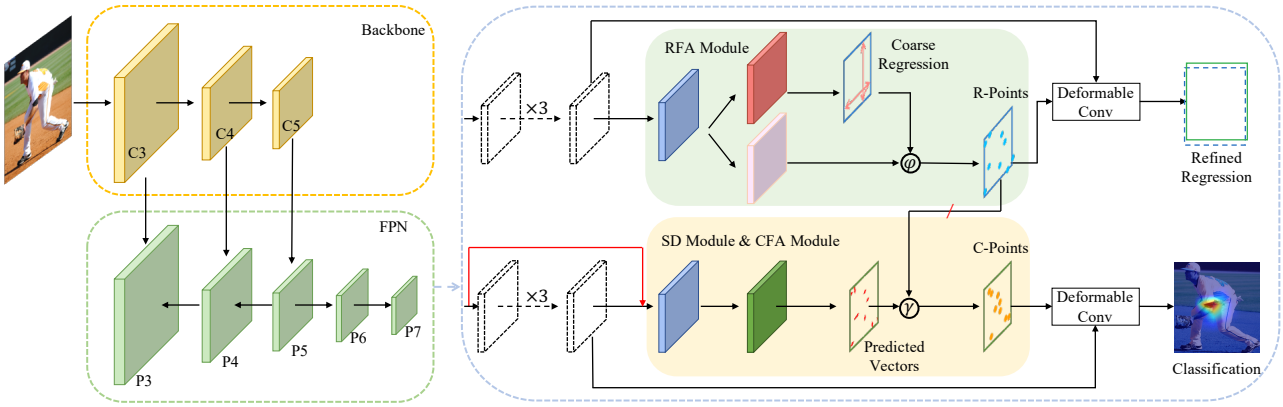


Figure 4: **Architecture of OAT-Net.** Our proposed architecture consists of a backbone, an FPN (P3-P7), and two heads for classification and regression, respectively. The first three layers of these two heads are both standard convolutional layers. The regression head consists of an RFA module for predicting coarse bbox, regression-aware points, and extracting aligned regression features. "φ" and "γ" denote Equations (2) and (6), respectively. "/" denotes the gradient flow detachment. The classification head consists of an SD module and a CFA module. One for predicting vectors that can map the regression-aware points to different locations, the other for generating classification-aware points and extracting aligned classification features.

Alignment. The decoupled parallel head framework is introduced into various one-stage detectors (Lin et al. 2017b; Tian et al. 2019) to disentangle the two tasks. However, the parallel heads share the same receptive field, which makes the two subnetworks still spatially aligned. In addition to IoU-Net, one-stage detectors, such as IoU-aware (Wu, Li, and Wang 2020) and PAA (Kim and Lee 2020) also apply an extra branch to predict the localization confidence and combine it with the classification confidence as the detection score. Different from previous methods, GFL (Li et al. 2020) and VFNet (Zhang et al. 2021) propose a joint representation format by merging the localization confidence and classification result to eliminate the inconsistency between training and inference. These approaches disentangle these two tasks by scoring both tasks. However, the features for these two tasks are still spatially aligned. To disentangle these two tasks, TSD (Song, Liu, and Wang 2020) generates two proposals for them, respectively. This approach can improve two-stage detectors by a large margin. However, it is incompatible with one-stage detectors because they do not have proposals and the RoI pooling process.

Proposed Approach

In this section, we first detail the proposed three modules. Then, we compare our approach with other feature extraction strategies and illustrate our advantages. Finally, we introduce the network (Figure 4) and the loss function of OAT-Net.

RFA Module

Motivated by the propose-and-align mechanism of the two-stage detection paradigm, we propose the RFA module. As shown in Figure 5, the subnetwork predicts a coarse bbox (i.e., the distance between the location and the four bounds of the object (l, t, r, b)). This vector can be transformed into the top-left corner point and the size of the coarse bbox

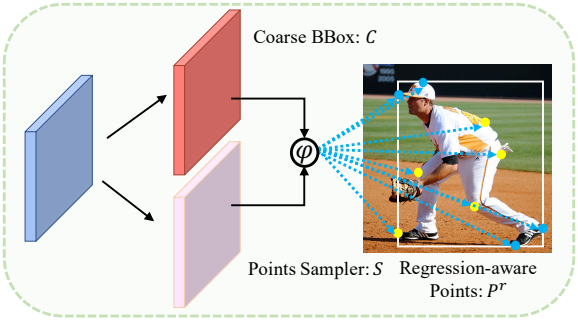


Figure 5: **Illustration of the RFA module.** "φ" and the white box denotes Equation (2) and the coarse bbox prediction, respectively.

$\mathcal{C}$ $(xmin, ymin, w, h)$. Then nine semantic points located within the coarse bbox are sampled by predicting normalized distances between the points and the top-left corner.

Given the feature map F^r from the last layer of the regression towers (i.e., the 3× convolutions shown in Figure 4), points sampler $\mathcal{S}$ and bbox $\mathcal{C}$ are both obtained by two convolution layers, i.e.:

$$\begin{cases} \mathcal{C} = \delta(conv_c\delta(conv_0(F^r))) \\ \mathcal{S} = \sigma(conv_s\delta(conv_0(F^r))) \end{cases} \tag{1}$$

where σ and δ are Sigmoid and ReLU, respectively. $\mathcal{C} \in \mathbb{R}^{H\times W\times 4}$ and $\mathcal{S} \in \mathbb{R}^{H\times W\times 14}$. With $\mathcal{C}$ and $\mathcal{S}$, the i-th regression-aware point Δp_i can be obtained with Equation (2):

$$\begin{cases} x_i = x_{\min} + w * S_{ix} \\ y_i = y_{\min} + h * S_{iy} \end{cases} \tag{2}$$

Therefore, the object can be represented by a points set $\mathcal{P}^r$, i.e.:

$$\mathcal{P}^r = \{\Delta p_i^r\}_{i=1}^n, n = 9 \tag{3}$$

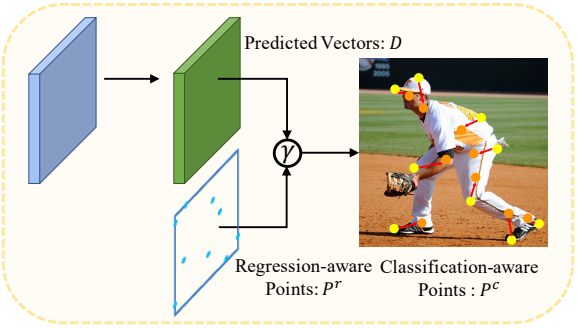


Figure 6: **Illustration of the SD and CFA.** "γ" denotes Equation (6).

Note that the channels of $\mathcal{S}$ and $\mathcal{P}^r$ are 14 and 18, respectively. The reason for this inconstancy is that we want to ensure that the sampled semantic points contain the four extreme points, which encode the location of the object. On this account, four points are sampled on the four bounds of the coarse bbox, respectively, as represented by the blue points in Figure 5. Therefore, four axial coordinates are preset and do not need to be learned.

For a standard convolution with kernel size 3×3, its sampling offset $\mathcal{R}$ is $(-1,-1),(-1,0),...,(1,1)$. By feeding the learned semantic points into the deformable convolution, the aligned regression features at location p on the feature map can be obtained with Equation (4).

$$y^r(\boldsymbol{p}) = \sum_{\boldsymbol{p}_i^r \in \mathcal{R}, \Delta \boldsymbol{p}_i^r \in \mathcal{P}^r} w\left(\boldsymbol{p}_i^r\right) \cdot x\left(\boldsymbol{p} + \boldsymbol{p}_i^r + \Delta \boldsymbol{p}_i^r\right) \quad (4)$$

where x and y^r are the input and output of the deformable convolution layer, respectively. The aligned regression features are then fed into a subnetwork for regression refinement, and the final output is supervised by the ground truth bboxes as in GFLv2 (Li et al. 2021). The gradient flow of the points sampler is originally generated by the regression loss. Therefore, the learned semantic points are regression-aware and should be located on the essential areas for the regression task.

The proposed RFA module can adaptively learn the locations of the semantic points that represent the location and size of the object, and extract aligned regression features to obtain fine-grained localization results.

SD and CFA Modules

Regression and classification are sensitive to different areas of the object. For this reason, extracting features from the regression-interested-locations, as in (Yang et al. 2019), hinders the detection performance. Thus, we propose the SD module to disentangle features of the regression and the classification tasks spatially. Similar to the points sampler, this module also consists of two convolution layers.

In the SD module, the regression-aware points act as the shape hypothesis of the object to be classified. To disentangle the gradient flows of the two tasks, the input boundary points are detached from the current graph, which means the supervision of the classification task will not affect the learning of regression-aware points. As shown in Figure 6, given the feature map F^c from the last layer of the classification towers (i.e., the $3\times$ convolutions shown in Figure 4), the disentanglement vectors $\mathcal{D}$ are obtained by two convolutional layers:

$$\mathcal{D} = \delta(conv_D \delta(conv_1(F^c))) \quad (5)$$

With the regression-aware points $\mathcal{P}^r$ and $\mathcal{D}$, the CFA module outputs the classification-aware points using Equation (6). Following a similar paradigm of the RFA module, the classification-aware points are also fed into a deformable convolution layer to extract aligned features for the classification task.

$$\mathcal{P}^c = e^{\mathcal{D}} \cdot \mathcal{P}^r \quad (6)$$

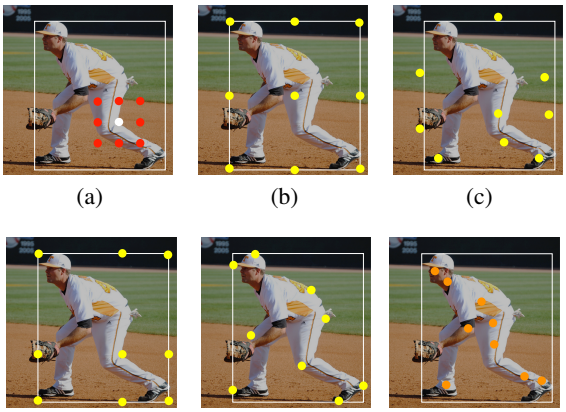


Figure 7: **Different feature extraction strategies.** (a): The standard convolution. The white point denotes the location that predicts the detection result. (b): RoI pooling/align (Ren et al. 2015). (c): Anchor alignment (Wang et al. 2019; Chen et al. 2019b; Zhang et al. 2018; Chi et al. 2019) and Deformable Convolutional Networks (i.e., DCN) (Dai et al. 2017). (d): Star Dconv (Zhang et al. 2021). (e): Our disentangled feature extraction approach.

Difference from Other Approaches

Figure 7 (a) shows the standard convolution, which can cause the misalignment between the point feature and the object, and the entanglement between classification and regression. Figure 7 (b) illustrates the feature alignment approach of two-stage detectors (i.e., RoI pooling/align), which requires the NMS process, thus increasing the computation. As shown in Figure 7 (c), some works (Wang et al. 2019; Chen et al. 2019b; Zhang et al. 2018) take anchor as the alignment target, and the way they generate offsets is as implicit as DCN. Without explicit constraints, some points are scattered in the background area and outside the bbox. As shown in Figure 7 (d), VFNet utilizes nine fixed sampling points to represent the object, which is explicit but lacks adaptability. As shown in Figure 7 (e), our features extraction module represents the object by two task-aware

point sets. Unlike previous approaches, we take the receptive field of the convolutional kernel as the alignment target instead of an anchor. Not only can our strategy adaptively sample spatially disentangle locations for the two tasks but also can give the learning process clear interpretability.

Network Architecture

Figure 4 presents the network of our proposed OAT-Net. We take state-of-the-art one-stage detector GFLv2 as our baseline. The backbone and feature pyramid network (FPN) (Lin et al. 2017a) in our model are the same as in GFLv2.

The input features from FPN are then fed into two parallel heads for the regression and classification tasks. We also apply a skip connection on the classification towers (i.e., the red line shown in Figure 4) because we find that it can improve the accuracy, and details will be explained in Section "Skip Connection". First, the RAF module predicts the coarse bbox and the regression-aware points with Equations (1) and (2). Then, the aligned regression features are extracted from the regression-sensitive regions with the deformable convolution. With the aligned regression features, the regression head outputs the refined localization result.

The classification features are fed into the SD module to generate a vector set. With Equation (6), the CAF module outputs the classification-aware points. After extracting the aligned classification features, the classification head outputs the final detection quality score as in (Li et al. 2020). Note that the gradient flow of the input regression-aware points for the SD module is detached to ensure that the two tasks are completely disentangled.

Loss Function

The proposed OAT-Net is optimized in an end-to-end fashion, and both the first and the second detection stages utilize ATSS (Zhang et al. 2020b) as the positive and negative targets assignment strategy. The training loss of OAT-Net is defined as follows:

$$\begin{aligned}\mathcal{L} = &\frac{1}{N_{\text{pos}}}\sum_{z}\lambda_0\mathcal{L}_{\mathcal{Q}} \\ &+\frac{1}{N_{\text{pos}}}\sum_{z}\mathbb{1}_{\{c_z^*>0\}}\left(\lambda_1\mathcal{L}_{\mathcal{C}}+\lambda_2\mathcal{L}_{\mathcal{R}}+\lambda_3\mathcal{L}_{\mathcal{D}}\right)\end{aligned} \tag{7}$$

where $\mathcal{L}_{\mathcal{Q}}$ is the Quality Focal loss (Li et al. 2020) for the classification task. $\mathcal{L}_{\mathcal{C}}$ and $\mathcal{L}_{\mathcal{R}}$ are both $GIoU$ loss (Rezatofighi et al. 2019), one for the coarse bbox prediction and the other for the refined regression result. $\mathcal{L}_{\mathcal{D}}$ is the Distribution Focal Loss (Li et al. 2020) for optimizing the general distribution representation of the bbox. $\lambda_0 \sim \lambda_3$ are the hyperparameters used to balance different losses, and they are set as $1, 1, 2$, and 0.5, respectively. N_{pos} denotes the number of selected positive samples, and z denotes all the locations on the pyramid feature maps. $\mathbb{1}_{\{c_z^*>0\}}$ is the indicator function, being 1 if $c_z^* > 0$ and 0 otherwise.

Experiments

Our OAT-Net is evaluated on the challenging MS-COCO benchmark (Lin et al. 2014). Following the common practice, we use the COCO train2017 split (115K images) as the training set and the COCO val2017 split (5K images) for the ablation study. To compare with state-of-the-art detectors, we report the COCO AP on the *test-dev* split (20K images) by uploading the detection result to the MS-COCO server.

Implementation

OAT-Net is implemented with MMDetection (Chen et al. 2019a) and Pytorch 1.7. If not specified, we take ResNet-50 (He et al. 2016) with the FPN as the basic network. The initial weights of different backbones are from the pre-trained models on ImageNet (Deng et al. 2009). All models are trained with stochastic gradient descent over 8 GPUs with the minibatch set as 16. The learning rate is initialized as 0.01, and the momentum and the weight decay are set as 0.9 and 0.0001, respectively. For the ablation study, the total training epoch number is 12, and the learning rate is decreased by 10 at epochs 8 and 11. Input images are resized such that the shorter edge being 800 and the longer edge being less than 1333.

Ablation Study

To validate the effectiveness of different counterparts of our proposed approach, we gradually add the proposed modules to the baseline (i.e., GFLV2). As shown in Table 1, the first row is the baseline performance, and 40.9 AP is acquired.

As presented in the second row, the first experiment is to investigate the effect of feature alignment without task disentanglement. Therefore, we only let the RFA module generates the regression-aware points, and both the RFA and CFA modules extract aligned features from the regression-aware points. The AP is improved to 41.3, which indicates that the aligned features do improve the detection accuracy.

As shown in the third row, to test the effect of spatial disentanglement, we utilize a set of adaptively learned points as the offset in the CFA module. Note that these points are learned without the regression-aware points acting as the shape hypothesis, yet the AP is still boosted to 41.6. These class-aware points are located in different positions from the regression-aware points, and higher accuracy is obtained (41.6 *vs.* 41.3). Therefore, sharing the same locations on both tasks as in RepPoints and VFNet will hinder the detection performance.

The fourth row shows the performance when applying the proposed SD module on the classification task. It can be observed that a significant performance gain is achieved (i.e., 1.2 AP absolute improvement). That thereby proves the effectiveness of utilizing the regression-aware points as the shape hypothesis and the importance of task disentanglement. Figure 8 is the visualization of task-aware points and their sensitive regions. This figure indicates that classification and regression are sensitive to different locations of the object, which also gives the interpretability of the SD module.

Finally, as shown in the last row, the long-range skip connection of the regression towers can also bring a considerable performance boost and lead to a gain of 0.4 AP. Note that the overall performance has been improved by 1.6 AP and 2.9 AP_L compared with the strong baseline.

Method	RFA	CFA	SD	skip	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
baseline					40.9	58.3	44.4	23.9	44.7	53.5
ours	✓	r			41.3	58.7	44.9	23.3	45.0	54.2
ours	✓	d	d		41.6	59.1	45.6	23.7	45.4	54.7
ours	✓	✓	✓		42.1	59.6	45.6	24.8	45.4	55.5
ours	✓	✓	✓	✓	**42.5**	**60.1**	**46.2**	**25.1**	**45.9**	**56.4**

Table 1: **Individual contributions.** Performance of different counterparts of OAT-Net on the COCO 2017val split. "skip" denotes the skip connection of the regression towers. "r" indicates that the CFA module utilizes the regression-aware points to generate aligned features. "d" indicates that the classification-aware points are directly predicted without the regression-aware points acting as the shape hypothesis.

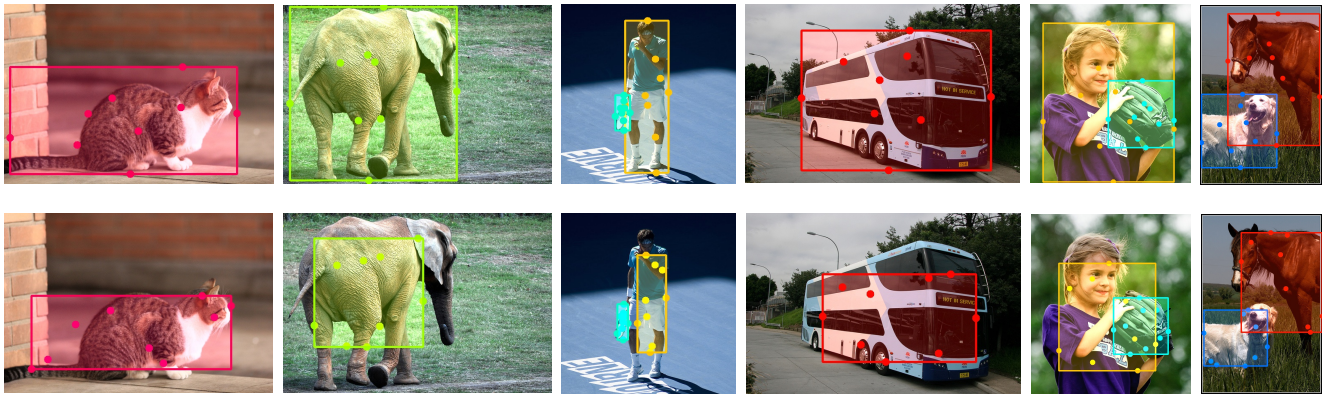


Figure 8: **Qualitative results on the val2017 split**. Visualization of the regression-aware (upper row) and classification-aware points (lower row). Different task-aware points are located on the different areas of the object, and their sensitive regions are distinct.

Method	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
w/ +	42.3	60.1	46.1	24.7	46.0	56.1
w/ exp	**42.5**	**60.1**	**46.2**	**25.1**	**45.9**	**56.4**

Table 2: **Spatial disentanglement strategies.** "+" indicates that the classification-aware points are obtained by adding the output disentanglement vectors and the regression-aware points. "exp" denotes utilizing Equation (6) to generate the classification-aware points.

C	R	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
		42.1	59.6	45.6	24.8	45.4	55.5
	✓	41.8	59.4	45.6	23.6	45.6	55.6
✓		**42.5**	**60.1**	**46.2**	**25.1**	**45.9**	**56.4**

Table 3: **Different skip connection strategies.** "C" and "R" denote classification and regression, respectively. We have tried three kinds of strategies: applying the long-range skip connection on the regression and the classification towers, respectively, and without the skip connection in both heads.

Spatial Disentanglement Strategies

In Equation (6), the output vectors $\mathcal{D}$ appear as the exponent to generate the classification-aware points. We also investigate the influence of another disentanglement strategy that $\mathcal{D}$ is directly aggregated with the regression-aware points ($\mathcal{P}^c = \mathcal{D} + \mathcal{P}^r$). As illustrated in Table 2, the "exp" strategy performs better than that of "+." The reason is that $\mathcal{D}$ in the proposed strategy has a smaller variance and codomain than the other one, making it easier to be learned.

Skip Connection

As previously discussed, classification has translation and scale invariance, whereas regression is quite the opposite. The "stride" and "zero-padding" of the convolution operation can also affect the invariance property. However, the cumulative number of stride and padding times varied in feature maps with different depths of the CNNs. Therefore, we want to explore how the feature fusion would affect the two tasks, respectively. The long-range skip connection (i.e., LS) is used in our investigation to fuse different feature maps. As shown in Table 3, the implementation of the LS on the classification towers can improve the AP by 0.4. Nevertheless, applying LS on the regression towers decreases the AP by 0.3. That indicates that the effect of the LS on these two tasks is quite the opposite.

Generality

Our proposed OAT can act as a plug-in operator for one-stage detectors. Therefore, we plug OAT into popular de-

Method	Backbone	Epoch	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
multi-stage								
GuidedAnchor(Wang et al. 2019)	R-50	12	39.8	59.2	43.5	21.8	42.6	50.7
DCNV2 (Zhu et al. 2019)	X-101-32x8d-DCN	24	44.5	65.8	48.4	27.0	48.5	58.9
RepPointsV2(Chen et al. 2020)	X-101-64x4d-DCN	24	49.4	68.9	53.4	30.3	52.1	62.3
TSD†(Song, Liu, and Wang 2020)	SE154-DCN	24	51.2	71.9	56.0	33.8	54.8	64.2
VFNet(Zhang et al. 2021)	X-101-32x8d-DCN	24	50.0	68.5	54.4	30.4	53.2	62.9
LSNet(Duan et al. 2021)	R2-101-DCN^p	24	51.1	70.3	55.2	31.2	54.3	65.1
BorderDet(Duan et al. 2021)	X-101-64x4d-DCN	24	48.0	67.1	52.1	29.4	50.7	60.5
one-stage								
CornerNet (Law and Deng 2018)	HG-104	200	40.5	59.1	42.3	21.8	42.7	50.2
SAPD(Zhu et al. 2020)	X-101-32x8d-DCN	24	46.6	66.6	50.0	27.3	49.7	60.7
ATSS(Zhang et al. 2020b)	X-101-32x8d-DCN	24	47.7	66.5	51.9	29.7	50.8	59.4
GFL(Li et al. 2020)	X-101-32x8d-DCN	24	48.2	67.4	52.6	29.2	51.7	60.2
FCOS-imprv (Tian et al. 2020)	X-101-32x8d-DCN	24	44.1	63.7	47.9	27.4	46.8	53.7
PAA (Kim and Lee 2020)	X-101-64x4d-DCN	24	49.0	67.8	53.3	30.2	52.8	62.2
baseline								
GFLV2 (Li et al. 2021)	R-50	24	44.3	62.3	48.5	26.8	47.7	54.1
GFLV2 (Li et al. 2021)	X-101-32x8d-DCN^2	24	49.0	67.6	53.5	29.7	52.4	61.4
GFLV2 (Li et al. 2021)	R2-101-DCN^2	24	50.6	69.0	55.3	31.3	54.3	63.5
OAT-Net	R-50	24	46.0	64.0	50.3	28.0	49.3	56.9
OAT-Net	X-101-32x8d-DCN^2	24	49.8	68.5	54.2	30.6	53.2	62.6
OAT-Net	X-101-32x8d-DCN	24	50.2	68.8	54.9	31.2	53.4	63.1
OAT-Net	R2-101-DCN^2	24	51.1	69.7	55.7	32.3	54.5	64.0
OAT-Net	R2-101-DCN	24	51.4	70.0	56.2	32.1	55.1	64.6
OAT-Net†	R2-101-DCN	24	**53.7**	**71.1**	**59.9**	**36.3**	**56.9**	**65.0**

Table 4: **OAT-Net *vs.* State-of-the-art Detectors.** All test results are reported on the COCO *test-dev* set. "**R**"-ResNet (He et al. 2016). "**SE**"-SENet (Hu, Shen, and Sun 2018). "**X**"-ResNeXt (Xie et al. 2017). "**HG**"-Hourglass (Newell, Yang, and Deng 2016). "**R2**"-Res2Net (Gao et al. 2019). "**DCN^2, DCN**"-applying Deformable Convolutional Network (Zhu et al. 2019) on the last two and three stages of the backbone, respectively. "**DCN^p**"-applying DCN on both the backbone and the FPN. "†" indicates test-time augmentations, including horizontal flip and multi-scale testing.

Method	AP	AP_{50}	AP_{75}
FCOS	38.6	57.2	41.7
OAT-FCOS	40.9 (+2.3)	59.3	44.2
RepPoints w/ GridF	37.4	58.9	39.7
RepPoints w/ RepF	38.1 (+0.7)	58.7	40.8
OAT-RepPoints	39.0 (+1.6)	60.5	41.6

Table 5: Performance of implementing our proposed approach in popular one-stage detectors. "GridF" and "RepF" denote the grid sampling and representative-points feature extraction strategies (Yang et al. 2019), respectively.

tectors FCOS and RepPoints, to validate its generality. As shown in Table 5, the performance gain is 2.3 AP on FCOS, which is a considerable improvement. For RepPoints, the baseline utilizes the GridF strategy to extract aligned features. Compared with RepF, our OAT-RepPoints performs better than it and gains 1.6 AP. One can see that OAT can significantly improve the accuracy of different detectors, which demonstrates the generality of proposed modules.

Comparisons with State-of-the-arts

The multi-scale training strategy (i.e., input images are resized from [400, 1333] to [960, 1333]) and the 2 $\times$ schedule (Chen et al. 2019a) are adopted as they are commonly used strategies in state-of-the-art methods. Our baseline model GFLV2 only applies DCN on the last two stages of the backbone, whereas the common practices (Tian et al. 2020; Zhang et al. 2021) usually apply it on the last three stages. Therefore, for a fair comparison, the results of the proposed method with both settings are reported. As Table 4 shows, our model achieves a 46.0 AP with Resnet-50, which outperforms other state-of-the-art methods with heavier backbones (e.g., FCOS with X-101-32x8d-DCN). With test-time augmentations and R2-101-DCN as the backbone, our best model achieves a 53.7 AP, which is a very competitive result among dense object detectors.

Conclusion

In this work, we presented OAT, a simple yet effective plug-in operator which consists of three modules: regression feature alignment (RFA), classification feature alignment (CFA), and spatial disentanglement (SD). With OAT, we proposed a new-fashioned framework for object detection that can adaptively learn to extract aligned and disentangled features for each task, without breaking the fully convolutional manner. Extensive experiments showed that OAT can considerably raise the performance of various one-stage detectors, and OAT-Net showed promising results among the state-of-the-art detectors.

References

Cai, Z.; and Vasconcelos, N. 2018. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6154–6162.

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

Chen, Y.; Han, C.; Wang, N.; and Zhang, Z. 2019b. Revisiting feature alignment for one-stage object detection. *arXiv preprint arXiv:1908.01570*.

Chen, Y.; Zhang, Z.; Cao, Y.; Wang, L.; Lin, S.; and Hu, H. 2020. Reppoints v2: Verification meets regression for object detection. *Advances in Neural Information Processing Systems*, 33.

Chi, C.; Zhang, S.; Xing, J.; Lei, Z.; Li, S. Z.; and Zou, X. 2019. Selective refinement network for high performance face detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 8231–8238.

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 764–773.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Duan, K.; Xie, L.; Qi, H.; Bai, S.; Huang, Q.; and Tian, Q. 2021. Location-Sensitive Visual Recognition with Cross-IOU Loss. *arXiv preprint arXiv:2104.04899*.

Gao, S.; Cheng, M.-M.; Zhao, K.; Zhang, X.-Y.; Yang, M.-H.; and Torr, P. H. 2019. Res2net: A new multi-scale backbone architecture. *IEEE transactions on pattern analysis and machine intelligence*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hu, J.; Shen, L.; and Sun, G. 2018. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7132–7141.

Jiang, B.; Luo, R.; Mao, J.; Xiao, T.; and Jiang, Y. 2018. Acquisition of localization confidence for accurate object detection. In *Proceedings of the European conference on computer vision (ECCV)*, 784–799.

Kim, K.; and Lee, H. S. 2020. Probabilistic anchor assignment with iou prediction for object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, 355–371. Springer.

Law, H.; and Deng, J. 2018. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European conference on computer vision (ECCV)*, 734–750.

Li, X.; Wang, W.; Hu, X.; Li, J.; Tang, J.; and Yang, J. 2021. Generalized focal loss v2: Learning reliable localization quality estimation for dense object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11632–11641.

Li, X.; Wang, W.; Wu, L.; Chen, S.; Hu, X.; Li, J.; Tang, J.; and Yang, J. 2020. Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection. In *NeurIPS*.

Lin, T.-Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; and Belongie, S. 2017a. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2117–2125.

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

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, 740–755. Springer.

Newell, A.; Yang, K.; and Deng, J. 2016. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, 483–499. Springer.

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

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28: 91–99.

Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; and Savarese, S. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 658–666.

Song, G.; Liu, Y.; and Wang, X. 2020. Revisiting the sibling head in object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11563–11572.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2019. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, 9627–9636.

Tian, Z.; Shen, C.; Chen, H.; and He, T. 2020. Fcos: A simple and strong anchor-free object detector. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Wang, J.; Chen, K.; Yang, S.; Loy, C. C.; and Lin, D. 2019. Region proposal by guided anchoring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2965–2974.

Wu, S.; Li, X.; and Wang, X. 2020. IoU-aware single-stage object detector for accurate localization. *Image and Vision Computing*, 97: 103911.

Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; and He, K. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1492–1500.

Yang, Z.; Liu, S.; Hu, H.; Wang, L.; and Lin, S. 2019. Reppoints: Point set representation for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9657–9666.

Zhang, H.; Chang, H.; Ma, B.; Wang, N.; and Chen, X. 2020a. Dynamic R-CNN: Towards high quality object detection via dynamic training. In *European Conference on Computer Vision*, 260–275. Springer.

Zhang, H.; Wang, Y.; Dayoub, F.; and Sunderhauf, N. 2021. Varifocalnet: An iou-aware dense object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8514–8523.

Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; and Li, S. Z. 2020b. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9759–9768.

Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; and Li, S. Z. 2018. Single-shot refinement neural network for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4203–4212.

Zhu, C.; Chen, F.; Shen, Z.; and Savvides, M. 2020. Soft anchor-point object detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, 91–107. Springer.

Zhu, X.; Hu, H.; Lin, S.; and Dai, J. 2019. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9308–9316.