







# Feature Enhancement and Alignment for Oriented Object Detection

Xu Xie , Zhi-Hui You , Si-Bao Chen , *Member, IEEE*, Li-Li Huang , Jin Tang , and Bin Luo 

**Abstract**—Over the last few years, developments in fields such as aviation and remote sensing have drawn increasing attention to the detection of rotated objects. Unlike general object detection, rotated object detection requires overcoming certain challenges, such as detecting objects with different directions and high aspect ratios. Recently proposed rotated object detectors have achieved good results, but most of them rely on hand-designed anchors, which require manual adjustment of the anchors settings in different scenarios. On the contrary, this article presents a method called FEADet, an anchor-free detector that utilizes feature enhancement and alignment to achieve competitive performance, without the use of anchors. Specifically, in order to better fuse features across different layers, we design an attention feature fusion (AAF) module to reduce the aliasing effect produced by the fusion of different layers. To deal with feature misalignment in detecting objects with orientation, we propose an adaptive alignconv (AAC) module, which is implemented by the constrained deformable convolution and align convolution. The AAC module can efficiently extract object features according to the decoded boxes and predicted constrained offsets. On the two benchmark datasets, dataset for object detection in aerial images (DOTA) and high resolution ship collection 2016 (HRSC2016), a comprehensive evaluation of our method has been conducted to demonstrate the effectiveness of these method in comparison with state-of-the-art methods.

**Index Terms**—Deep learning, feature alignment, feature enhancement, oriented object detection, remote sensing images.

## I. INTRODUCTION

OBJECT detection, which is one of the most fundamental tasks in the field of computer vision, requires identifying the location and class of the region of interest. Recently, due to the advancement of remote sensing technology, researchers have become more focused on the detection of aerial objects, which is important in both military and civilian applications, including military reconnaissance, environmental monitoring, disaster relief, and others.

With the advancement of deep learning, various general detectors [3], [5], [12] have been introduced to achieve competitive

performance, thereby promoting the development of detection technology for rotating objects. Based on whether or not anchors are used, these detectors can be grouped into two types. Anchor-based detectors [9], [13], [24] use a lot of anchors placed on the feature map and need to decode their offsets to get the final predicted boxes. For different scenarios, we need to artificially design the size of anchors. On the contrary, anchor-free detectors [3], [14], [23] can obtain the prediction boxes without using anchors. While anchor-free methods may not perform as well as advanced anchor-based methods in aerial detection, they offer greater flexibility and do not require manual design of anchor parameters. With this article, our goal is to improve the anchor-free network's performance while still maintaining its simplicity.

In contrast to object detection in natural scenes, detecting objects in aerial environments requires overcoming certain challenges. As a result of the top-down perspective, aerial environments contain numerous objects with differing scales, aspect ratios, and angles. In order to overcome these challenges, our primary research focus is on improving the extraction of semantic features from objects. In this article, we suggest the feature enhancement and alignment detector (FEADet), a one-stage anchor-free detector.

First, detectors are able to predict objects of different sizes by using different layers of feature maps. FPN [4] improves low-level semantic information by performing a straightforward summing of high and low-level information. We notice that simple summation may cause confusion about the low-level feature map information. To make feature fusion more effective, we propose an attention feature fusion (AFF) module, which uses attention mechanism to guide high-level feature maps to learn features needed by low-level feature maps, so as to enhance the semantic information of low-level feature maps and reduce the aliasing effect caused by the high-level feature maps.

Second, we observe that the standard convolution does not effectively align the rotated objects. In an attempt to overcome this issue, we introduce an adaptive alignment convolution (AAC) module that can adaptively learn the sampling point positions using refinement box information, resulting in more accurate alignment of object features.

We assess our approach on the widely used high resolution ship collection 2016 (HRSC2016) [25] and dataset for object detection in aerial images (DOTA) [28] benchmark datasets and compare it with state-of-the-art methods. The results show that our approach achieves competitive results.

In summary, our study is characterized by two main contributions, which are summarized in the following two ways.

Manuscript received 25 May 2023; revised 23 August 2023 and 15 October 2023; accepted 14 November 2023. Date of publication 17 November 2023; date of current version 4 December 2023. This work was supported in part by NSFC Key Project of International (Regional) Cooperation and Exchanges under Grant 61860206004, in part by NSFC Key Project of Joint Fund for Enterprise Innovation and Development under Grant U20B2068, and in part by the National Natural Science Foundation of China under Grant 61976004. (*Corresponding authors: Si-Bao Chen; Li-Li Huang.*)

The authors are with the IMIS Lab of Anhui Province, Anhui Provincial Key Lab of Multimodal Cognitive Computation, MOE Key Lab of ICSP, School of Computer Science and Technology, Anhui University, Hefei 230601, China (e-mail: e21201005@stu.ahu.edu.cn; 1574583514@qq.com; sbchen@ahu.edu.cn; hill\_ahu@ahu.edu.cn; tj@ahu.edu.cn; luobin@ahu.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2023.3333957

© 2023 The Authors. This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License. For more information, see <https://creativecommons.org/licenses/by-nc-nd/4.0/>

- 1) To improve the feature extraction of objects, we introduce the AFF module, which directs the high-level feature maps to learn the features necessary for the low-level feature maps to increase the semantic information of the latter.
- 2) For the feature misalignment problem, we design an AAC module that is able to adaptively learn the sampling points with the supervision of the refinement boxes, thus extracting the required features for objects.

## II. RELATED WORK

### A. Object Detection in Aerial Images

With the development of deep learning, numerous advanced detectors have emerged and achieved remarkable performance in aerial images. Two major categories can be distinguished between these detectors: anchor-based and anchor-free. A detector based on anchors starts by placing many anchors on the feature map, followed by assigning samples to help positive anchors to learn the ground truth boxes. RRPN [11] builds upon RPN [12] by introducing anchors with angles and uses rotated ROI pooling to extract features of objects with orientation. RoI transformer [13] reduces computational overhead by learning rotated anchors through a fully connected layer during the RPN phase. S<sup>2</sup>ANet [9] employs ARN to produce more accurate anchors and aligns the features to predict objects with high accuracy. Anchor-free detectors, in contrast to anchor-based ones, do not rely on predefined anchors but rather predict objects directly. FCOSR [14] adds the angle parameter to predict rotated objects based on FCOS [3]. BBAVectors [16] extends the CenterNet [26] with a set of bounding vectors to predict rotated objects. Oriented RepPoints [43] uses a set of points to predict rotated objects.

### B. Feature Fusion and Attention Mechanism

To predict objects of different scales, it is common practice to use features from different layers. Faster RCNN [12] uses a single feature map to predict objects. SSD [17] predicts objects using feature maps of different layers. FPN [4] utilizes a top-down pathway with lateral connections for feature fusion, enhancing the information in the lower feature map. PANET [4] enhances the FPN structure by introducing a bottom-up fusion path, resulting in faster information transfer. LMO-YOLO [45] proposes a multiple linear rescaling scheme and introduces dilated convolutions to maintain a large receptive field in the network, followed by multiscale feature fusion, enabling the network to learn richer features. In recent years, attention mechanisms have made rapid progress in the field of deep learning. SENet [20] introduces a channel attention module that captures the interdependence among feature channels. CBAM [21] proposes a combined spatial and channel attention that learns direct correlations of features from global dimensions. SOT-Net [46] proposes a cross-attention module to enhance the representation of semantic information. In object detection, incorporating attention mechanisms into convolutional neural networks frequently results in the extraction of more comprehensive features. UEFPN [22] proposes a channel attention based CAF module for multiscale feature fusion. IAANet [47] proposes a coarse-to-fine

interior attention-aware network, modeling attention between pixels in the coarse object region to obtain rich perceptual information. In this article, we introduce an AFF module that uses spatial attention on shallow feature maps and channel attention on deep feature maps to learn the features to be fused, reducing the confusion effect caused by fusion.

### C. Feature Alignment in Oriented Object Detection

Feature alignment pertains to the alignment of convolutional features with RoIs or anchor boxes, and various approaches have been suggested to alleviate the issue of feature misalignment. In the oriented object detection field, two-stage detectors often use rotated RoIPooling [11] or rotated RoIAlign [11] for feature alignment. For single-stage detectors, R<sup>3</sup>Det [24] obtains information of the refined anchors by feature interpolation, which achieves feature alignment. OSSDet [15] has proposed the AFAM module to reassign the positions and weights of sampling points during the refinement stage for better accuracy. S<sup>2</sup>ANet [9] designs a novel feature alignment method called alignconv. It fixedly extracts features of nine points of refined anchors for better feature alignment. FSDet [23] employs a multiheaded attention strategy to extract object-aligned features. Oriented RepPoints [43] proposes an adaptive sampling method based on quality evaluation and sample allocation strategies to select sampling points and positions with high quality. TS-Conv [44] proposes taskwise sampling convolutions that can adaptively sample task-aware features from the respective sensitive regions. Different from the methods mentioned above, our proposed AAC module can dynamically learn the positions of sampling points with the assistance of a refined box information to achieve better feature alignment.

## III. PROPOSED METHOD

Here, we provide a comprehensive overview of our proposed FEADet. As described Fig. 1, the architecture of FEADet is founded on the one-stage anchor-free detector, FCOS [3].

### A. Baseline Network

The baseline model we adopt is a one-stage anchor-free detector, which is an extension of the FCOS [3] detector in the MMRotate [2] toolbox. The model consists of the FPN [4] as the backbone for multiscale feature extraction and two task-specific subnetworks. Both subnetworks consist of many convolutional layers stacked on each other and are fully convolutional. The regression subnetwork forecasts the separation between each location to the bounding box, while the classification subnetwork predicts the object class.

It is worth noting that FCOS is designed for generic object detection. To enable the detection of arbitrarily oriented objects in aerial images, we introduce a new parameter  $\theta$ , which predicts the angle. Fig. 2 shows the relationship between the regression results  $\{(l, t, r, b, \theta)\}$  and the rotated object boxes. Our method adopts the long-edge definition approach, where  $l$ ,  $t$ ,  $r$ , and  $b$ , respectively, denote the distances from the anchor to the left, top, right, and bottom edges of the ground truth box, and parameter

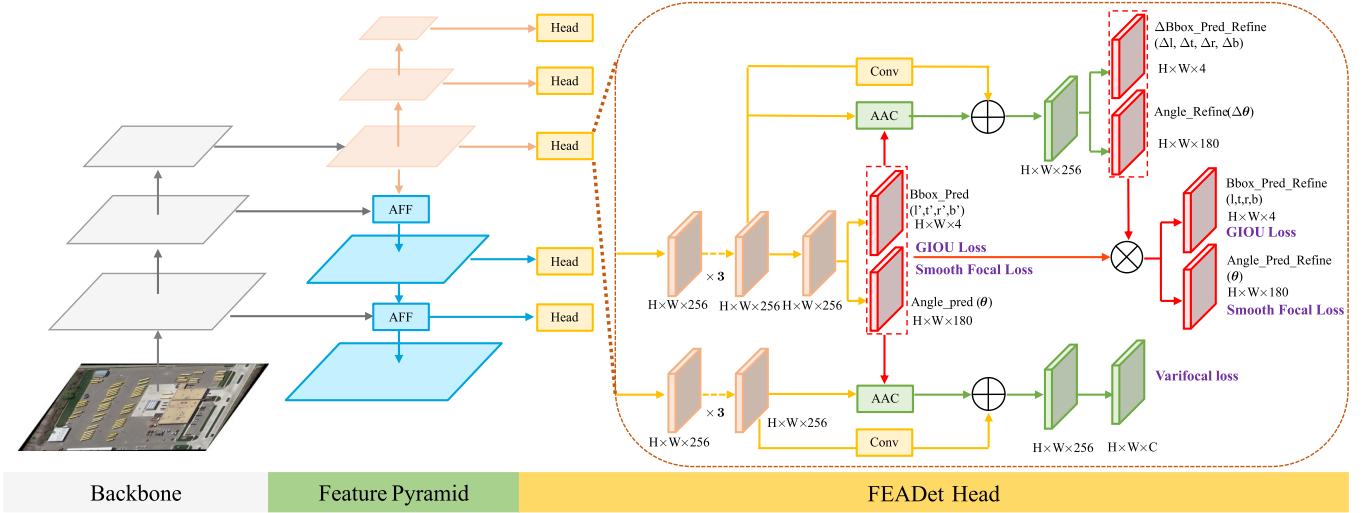


Fig. 1. Structure of the suggested FEADet. It is constructed based on the FPN, and incorporates the AFF module for feature enhancement as well as the AAC module for obtaining aligned features. The Conv represents a  $1 \times 1$  convolutional layer.

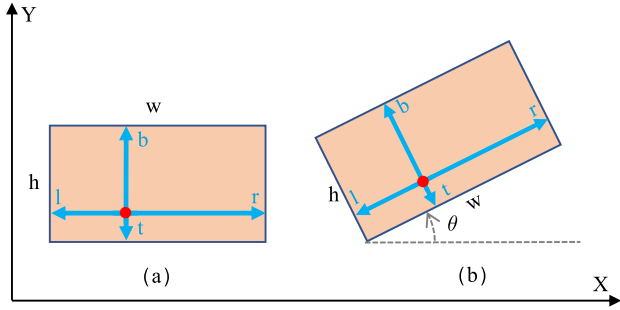


Fig. 2. Two categories of bounding boxes. (a) Horizontal bounding box, given by  $\{(l, t, r, b)\}$ , specifies the distance from the location to the left, top, right, and bottom sides of the ground truth box. (b) Oriented bounding box, given by  $\{(l, t, r, b, \theta)\}$ , specifies the distance from the location to the left, top, right, and bottom sides of the ground truth box, along with the orientation angle  $\theta$ . The angle  $\theta$  is the angle between the  $X$ -axis direction and the oriented direction, where  $\theta$  is within the range of  $\theta \in (-\frac{\pi}{2}, \frac{\pi}{2}]$ .

$\theta$  represents the angle formed between the  $X$ -axis and the width of the bounding box ( $w$ ).

We denote the location  $(x, y)$  on the input image that corresponds to each location  $(a, b)$  on the feature map  $O_i$  as follows:

$$\begin{aligned} x &= \left\lfloor \frac{s}{2} \right\rfloor + a \cdot s \\ y &= \left\lfloor \frac{s}{2} \right\rfloor + b \cdot s \end{aligned} \quad (1)$$

where  $s$  represents the stride of feature level  $i$ . We leverage five levels of feature maps, specifically  $\{O_3, O_4, O_5, O_6, O_7\}$ . Since we extract multiscale feature maps from both ResNet [6] and FPN [4], feature levels  $O_3, O_4, O_5, O_6$ , and  $O_7$  have strides of 8, 16, 32, 64, and 128, respectively. In aerial image object detection, the ground truth is commonly represented as  $\{(x, y, w, h, \theta)\}$ , where  $(x, y)$  is the center of the object, while  $w, h$ , and  $\theta$  signify the width, height, and angle of the oriented bounding box (OBB). Therefore, when computing the loss in the next step, we decode

the predicted regression vector values  $\{(l, t, r, b, \theta)\}$  as follows:

$$\begin{aligned} x^* &= x + \frac{b-t}{2} \sin \theta + \frac{r-l}{2} \cos \theta \\ y^* &= y + \frac{b-t}{2} \cos \theta + \frac{r-l}{2} \sin \theta \\ w^* &= l + r, \quad h^* = t + b, \quad \theta^* = \theta. \end{aligned} \quad (2)$$

In addition, different from the original FCOS [3], we replace the classification and quality branches with varifocal loss [7] and used ATSS [5] sample assignment as our baseline. With these definitions described above, the multitask loss function for training the entire baseline network is expressed as

$$L = \frac{1}{N_{\text{pos}}} \left( \sum_i L_c(\hat{c}_i, c_i^*) + \sum_i [l_i^* \geq 1] L_r(\hat{d}_i, d_i^*) \right) \quad (3)$$

where  $i$  denotes the index of locations across all feature map layers. The classification-IoU combination output and target values for location  $i$  are represented by  $\hat{c}_i$  and  $c_i^*$ , respectively.  $L_c$  adopts varifocal [7] classification loss. The decoded localization output for location  $\hat{d}_i$  is represented by  $m$ , which is obtained using (2), while  $d_i^*$  denotes the regression label for that same location. When location  $i$  is assigned positive sample, the inversion bracket indicator function  $[l_i^* \geq 1]$  outputs 1, otherwise 0. The value of  $L_r$  corresponds to the regression loss of positive samples, as defined by

$$\begin{aligned} L_r(\hat{d}, d^*) &= L_{\text{hbb}}((\hat{x}, \hat{y}, \hat{w}, \hat{h}), (x^*, y^*, w^*, h^*)) \\ &\quad + \lambda_1 L_{\theta}(\hat{\theta}, \theta^*). \end{aligned} \quad (4)$$

$L_{\text{hbb}}$  is the loss function for the horizontal bounding box, computed by comparing the predicted OBB with the ground-truth box using the GIOU [8] loss. Due to the periodicity of angles can lead to sudden changes in loss, we adopt the classification method CSL [1] to predict angles. We set 180 categories, representing 180 angles.  $L_{\theta}$  adopts smooth focal loss. Besides, in our experiments, we assign a value of 1 to the hyper-parameter  $\lambda_1$ .

TABLE I  
COMPARISON OF OUR FEADet WITH STATE-OF-THE-ART METHODS ON DOTA DATASET

Method	Backbone	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP(%)	FPS
<b>Two-stage Methods</b>																		
ICN [29]	ResNet101	81.36	74.30	47.70	70.32	64.89	67.82	69.98	90.76	79.06	78.20	53.64	62.90	67.02	64.17	50.23	68.16	-
CADNet [30]	ResNet101	87.80	82.40	49.40	73.50	71.10	63.50	76.60	<b>90.90</b>	79.20	73.30	48.40	60.90	62.00	67.00	62.20	69.90	-
RoI Transformer [13]	ResNet101	88.64	78.52	43.44	75.92	68.81	73.68	83.59	90.74	77.27	81.46	58.39	53.54	62.83	58.93	47.67	69.56	7.6
SCRDet [31]	ResNet101	<b>89.98</b>	80.65	52.09	68.36	68.36	60.32	72.41	90.85	87.94	<b>86.86</b>	65.02	66.68	66.25	68.24	65.21	72.61	-
Mask OBB [32]	ResNet50	89.61	85.09	51.85	72.90	75.28	73.23	85.57	90.37	82.08	85.05	55.73	68.39	71.61	69.87	66.33	74.86	-
Gliding Vertex [33]	ResNet101	89.64	85.00	52.26	77.34	73.01	73.14	86.82	90.74	79.02	86.81	59.55	<b>70.91</b>	72.94	70.86	57.32	75.02	9.2
ReDet [34]	ReResNet50	88.79	82.64	53.97	74.00	78.13	<b>84.06</b>	88.04	90.89	87.78	85.75	61.76	60.39	75.96	68.07	63.59	76.25	8.3
Oriented RCNN [35]	ResNet101	88.86	83.48	55.27	76.92	74.27	82.10	87.52	<b>90.90</b>	85.56	85.33	65.51	66.82	74.36	70.15	57.28	76.28	9.2
Oriented RepPoints [43]	ResNet101	89.53	84.07	<b>59.86</b>	71.76	79.95	80.03	87.33	90.84	87.54	85.23	59.15	66.37	75.23	73.75	57.23	76.52	10.7
<b>One-stage Methods</b>																		
PloU [36]	ResNet101	80.90	69.70	24.10	60.20	38.30	64.40	64.80	90.90	77.20	70.40	46.50	37.10	57.10	61.90	64.00	60.50	-
DRN [37]	ResNet101	88.91	80.22	43.52	63.35	73.48	70.69	84.94	90.14	83.85	84.11	50.02	58.41	67.62	68.60	52.50	70.70	-
BBAVectors [16]	ResNet101	88.35	79.96	50.69	62.18	78.43	78.98	87.94	90.85	83.58	84.35	54.13	60.24	65.22	64.28	55.70	72.32	-
CFC-NET [38]	ResNet101	89.08	80.41	52.41	70.02	70.02	78.11	87.21	90.08	84.47	85.64	60.51	61.52	67.82	68.02	50.09	73.50	-
R <sup>3</sup> Det [24]	ResNet101	88.76	83.09	50.91	67.27	76.23	80.39	86.72	90.78	84.68	83.24	61.98	61.35	66.91	70.63	53.94	73.79	8.8
S <sup>2</sup> A-Net [9]	ResNet50	89.11	82.84	48.37	71.11	78.11	78.39	87.25	90.83	84.90	85.64	60.36	62.60	65.26	69.13	57.94	74.12	13.3
OSSDet [15]	ResNet50	88.93	79.98	48.08	69.10	78.24	78.64	87.52	90.87	85.99	85.67	61.21	64.19	71.31	68.49	59.04	74.48	-
GGHL [39]	DarkNet53	89.74	<b>85.63</b>	44.50	<b>77.48</b>	76.72	80.45	86.16	90.83	<b>88.18</b>	86.25	67.07	69.40	73.38	68.45	<b>70.14</b>	76.95	-
FCOS [3]	ResNet50	89.17	71.98	47.96	61.64	79.30	73.51	85.76	90.90	81.12	84.29	59.57	62.67	62.08	69.93	49.31	71.28	16.7
FCOS <sup>^</sup> [3]	ResNet50	89.01	80.10	48.45	67.09	75.67	76.17	84.57	90.87	85.29	83.93	62.58	65.01	64.60	68.33	59.51	73.41	15.1
FEADet (ours)	ResNet50	88.60	78.80	50.11	72.85	80.54	80.67	87.40	90.80	84.73	83.90	65.11	62.79	66.55	69.86	52.84	74.37	11.5
FEADet* (ours)	ResNet50	89.20	82.07	54.24	72.67	<b>81.24</b>	83.72	<b>88.23</b>	90.84	86.13	84.34	<b>69.56</b>	63.39	<b>76.20</b>	75.43	58.48	77.05	11.5
FEADet* (ours)	ResNet101	89.11	83.99	55.33	72.25	77.74	83.80	86.30	90.84	86.84	85.82	64.35	66.80	74.80	<b>77.75</b>	60.71	<b>77.10</b>	9.1

\* Indicates using rotational data augmentation and a learning cycle three times as long as the original. ^ represents that the model is using our re-implementation of FCOS. Running the FPS test with an NVIDIA GeForce GTX 1080 Ti and an input image size of 1024x1024, the unit is frames per second. The bold results indicate the best for each column.

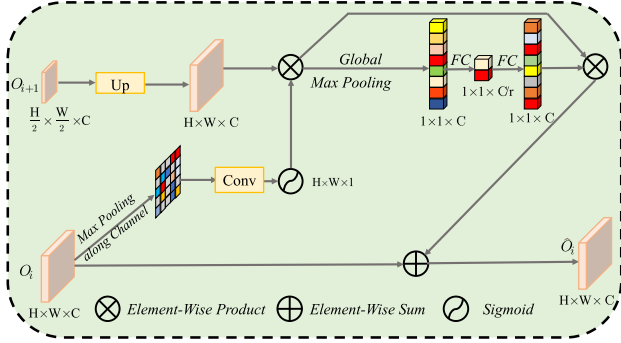


Fig. 3. Detailed attention feature fusion (AFF) module.

### B. Attention Feature Fusion Module

In aerial situations, detecting objects with arbitrary orientation typically requires more discriminative and sensitive features than natural images, as these features need to contain information about angles as well as scales. The amount of information that can be obtained from a single feature map is limited. Broadly speaking, shallow feature maps exhibit greater resolution but lesser semantic information, while deep feature maps have reduced resolution but more comprehensive semantic information. FPN [4] employs summation to fuse shallow and deep feature maps, resulting in highly semantic features. Since the low-level feature map detects small objects and the high-level feature map is sensitive to large objects, directly adding adjacent feature maps can cause interference in the lower-level feature maps and introduce unwanted information. To more effectively address the connection between deep features and shallow features and transfer deep features to shallow levels, we have developed a feature fusion module, the AFF module, as illustrated in Fig. 3.

Spatial and channel inconsistencies may arise between adjacent feature maps. To mitigate the spatial inconsistency, spatial attention is employed to guide the deep feature maps using the shallow ones. For channel inconsistency, we use channel attention to learn the corresponding weights. The calculation

equation can be described as follows:

$$\hat{O}_i = M_c (M_s (P_i) \otimes Up(O_{i+1})) \otimes Up(O_{i+1}) + O_i \quad (5)$$

where  $O_i$  and  $O_{i+1}$  represent deep and shallow feature map, respectively.  $\otimes$  denotes tensor product and  $Up$  indicates upsampling. First, we extract the spatial attention map  $M_s$  from the deep feature map. The objective of spatial attention is to produce a weight map of the foreground based on the deep feature map. Then, we use the channel attention map  $M_c$  to extract the channel relations of the feature map. The formula is given as follows:

$$M_s(x) = \sigma(c^{7 \times 7}(x_{mpac}))$$

$$M_c(x) = \sigma(K_1(K_0(x_{gap}))) \quad (6)$$

here,  $x_{gap}$  and  $x_{mpac}$  are obtained from the input feature  $x$  through global average and max pooling operations along the channel and spatial dimensions, respectively.  $K_0$  is the full connection layer for reducing dimensions.  $W_0 \in R^{C \times C/r}$  and  $K_1 \in R^{C/r \times C}$  are fully connected layers for dimension reduction and dimension increase, respectively. The hyper-parameter  $r$  defaults to 4.  $c^{7 \times 7}$  indicates convolution of  $7 \times 7$  filters.  $\sigma$  is the sigmoid function.

According to (5), the AAF module selects information from high-level feature maps through the attention mechanism to provide valuable information to shallow feature maps, thereby enhancing the semantic information of the shallow feature maps. So it is an efficient and flexible module.

### C. Adaptive AlignConv Module

AlignConv [9] cannot dynamically learn the position of feature alignment. When the object bounding box is determined, the position of feature alignment is also determined. Thus, in order to mitigate this phenomenon, we suggest an AAC module that can learn the feature extraction location under certain constraints, thereby better extracting the features of the object.

Fig. 4 displays the sampling points of various methods using a  $3 \times 3$  kernel at a certain place on a feature map. It can be observed that the stationary sampling position utilized in

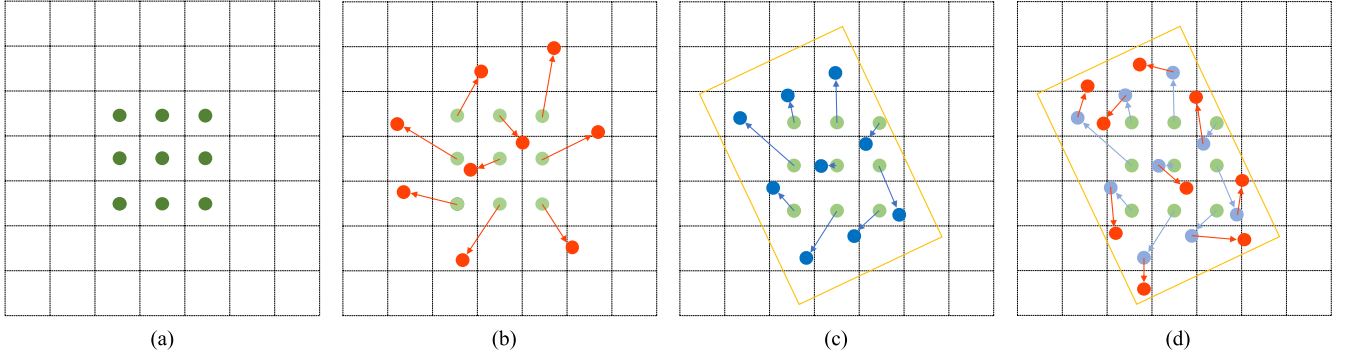


Fig. 4. Visual comparison of  $3 \times 3$  convolution kernel-based methods. (a) represents standard convolution with fixed sampling locations (indicated by green dots). (b) illustrates deformable convolution with learnable sampling locations (indicated by red dots). (c) is AlignConv with nine fixed sampling points of predicted box (in blue dots). (d) is ours adaptive AlignConv with nine learnable sampling points of predicted box (in red dots). (a) Standard convolution. (b) Deformable convolution. (c) AlignConv. (d) Adaptive AlignConv.

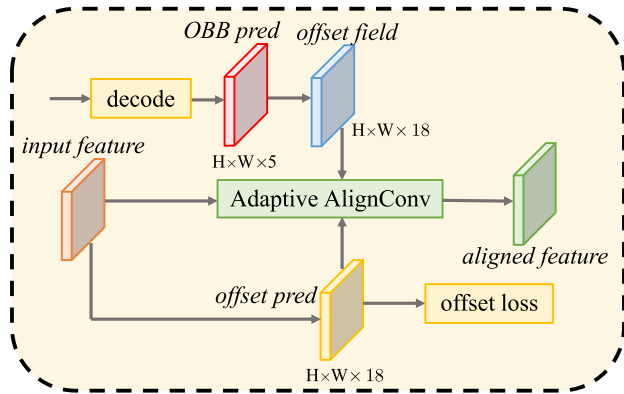


Fig. 5. Illustration of the adaptive AlignConv (AAC) module.

standard convolution is not effective in representing the object. This is because the standard convolution sample point position is fixed, not according to the shape of the object to change position. The deformable convolution [10] does not represent the object well due to the absence of a supervised signal and the learned position. AlignConv [9] uses rough prediction bounding boxes to extract the features at fixed locations, thus effectively extracting the key features of the object. Since the distribution of objects can differ within an image, it is necessary to dynamically adjust the extracted positions to adapt to different objects' distributions. To dynamically extract the features of the object, we design an AAC module, as shown in Fig. 5. The computation formula can be expressed as

$$Y(p) = \sum_{r \in R} W(r) \cdot X(p + r + o + d) \quad (7)$$

where  $p$  is the location on the feature map. Sampling is carried out using a regular grid  $O$  over the input feature map  $X$ , with  $Y$  denoting the weighted sum of sampled values by  $W$ . The grid  $O$  defines a  $3 \times 3$  kernel with dilation 1, and is composed of the points  $\{(-1, -1), (-1, 0), \dots, (0, 1), (1, 1)\}$ .  $o$  is the offset between the sampling point of the predicted rotated objects and the standard convolution at that location, as shown in Fig. 5.  $d$  is the bias value predicted for that position by a  $3 \times 3$  kernel with dilation 1. To make the predicted offset more accurate, we

use the predicted box to give it a certain supervisory signal. The offset loss function can be defined as

$$\text{mask}_d = \text{sgn} \left( \text{abs}(d_x, d_y) - \frac{1}{k}(w, h) \right)$$

$$L(d)_o = \left( \text{abs}(d_x, d_y) - \frac{1}{k}(w, h) \right) * \text{mask}_d \quad (8)$$

where  $h$  and  $w$  indicate the height and width of the OBB, respectively.  $\text{abs}$  is the absolute value function.  $\text{sgn}$  is a symbolic function. The hyper-parameter  $k$  defaults to 3.

As shown in Fig. 4, our AAC module is capable of more efficient object feature extraction by utilizing aligned convolution as a foundation. It retains both the supervised signal of aligned convolution and the flexibility of deformable convolution, making it an efficient and flexible module.

#### D. Detection Model

After adding the aforementioned two modules to the FCOS [3] baseline model, the FEADet framework depicted in Fig. 1 is obtained. Subsequently, the loss function for the FEADet model is formulated as follows:

$$L = \frac{1}{N_{\text{pos}}} \sum_i L_c(\hat{c}_i, c_i^*)$$

$$+ \frac{1}{N_{\text{pos}}} \sum_i [l_i^* \geq 1] (\lambda_2 L_r(\hat{d}_i, d_i^*) + \lambda_3 L_o(\hat{k}_i, k_i^*))$$

$$+ \frac{\lambda_4}{N_{\text{pos}}} \sum_i [l_i^* \geq 1] (L_r(d_i, d_i^*)) \quad (9)$$

where  $L_c$ ,  $L_r$ , and  $L_o$  are classification, regression, and offset loss, respectively. The hyper-parameters  $\lambda_2$ ,  $\lambda_3$ , and  $\lambda_4$  are set to 1. The rest of the settings are the same as in (5).

## IV. EXPERIMENTS

### A. Datasets

We verify the effectiveness of our method by testing it on two widely adapted benchmark datasets: DOTA [28] and HRSC2016 [25].

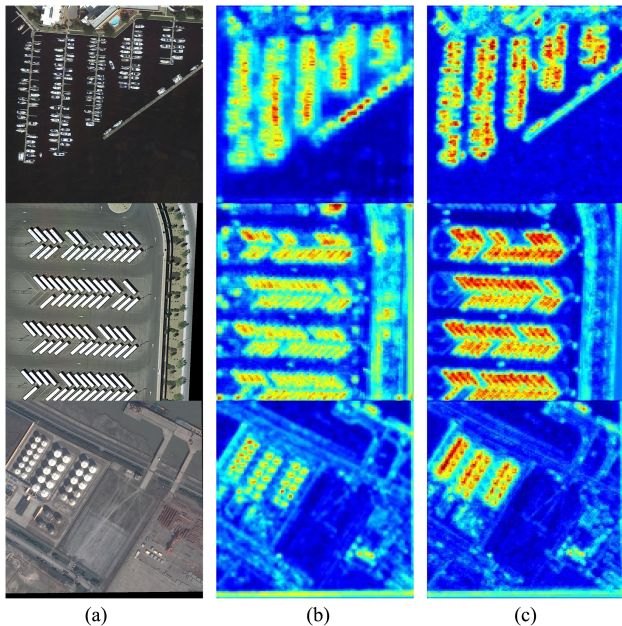


Fig. 6. Visual comparison of our method with FPN. (a) Original images to be detected. (b) Visual results of FPN on  $O_2$  level feature map. (c) Visual results of FPN+AFF on  $O_2$  level feature map. (a) Images. (b) FPN. (c) FPN+AFF.

DOTA is a challenging benchmark and one of the most widely used datasets in the field of horizontal and directional object detection in aerial images. Our primary goal is to detect objects by the use of oriented bounding boxes. In the dataset, 15 common object categories are represented, including swimming pools (SP), baseball diamonds (BD), basketball courts (BC), soccer fields (SBF), small vehicles (SV), roundabouts (RA), helicopters (HC), bridges (BR), tennis courts (TC), storage tanks (ST), ground field tracks (GTF), large vehicles (LV), planes (PL), ships (SH), and harbors (HA). The dataset comprises 2806 images with pixel dimensions ranging from  $800 \times 800$  to  $4000 \times 4000$ , consisting of a training set of 1,411 images, a validation set of 458 images, and a testing set of 937 images. All images are partitioned into subimages of  $1024 \times 1024$  pixels, with a 200-pixel overlap, for the purposes of training, validation, and testing.

HRSC2016 is a challenging remotely sensed data set to detect ships of any orientation and high aspect ratio ships. The dataset consists of a total of 1061 images, ranging from 300 by 300 pixels up to 1500 by 900 pixels. The training images are 436, the validation images are 181, and the testing images are 444. In particular, we use the training and validation datasets to train our models, and the test datasets to evaluate them. To ensure a fair comparison, we resize all the pictures sent into the network model to  $800 \times 800$  pixels.

### B. Implementation Details

We employ the MMRotate [2] toolbox for the development of our oriented object detection model (FEADet). The model is trained on an NVIDIA GeForce RTX 3080 GPU. Our batch size is set to 2 and initial learning rate is 0.0025, but is decayed by a factor of 10 at each time of decay of the learning rate. This model is trained for 12 epochs on DOTA [28] with a learning rate decay taking place at the 8th and 11th epochs. When incorporating rotate data augmentation, we train the model for 36 epochs and

TABLE II  
DETECTION RESULTS WITH DIFFERENT FUSION METHODS ON DOTA DATASET; SA AND CA REFER TO SPATIAL ATTENTION AND CHANNEL ATTENTION, RESPECTIVELY

Model	Backbone	Feature fusion method	mAP(%)
baseline	ResNet50	FPN	69.44
		FPN+CA	69.72
		FPN+SA	69.77
		FPN+CA+SA	<b>69.99</b>

The bold results indicate the best for each column.

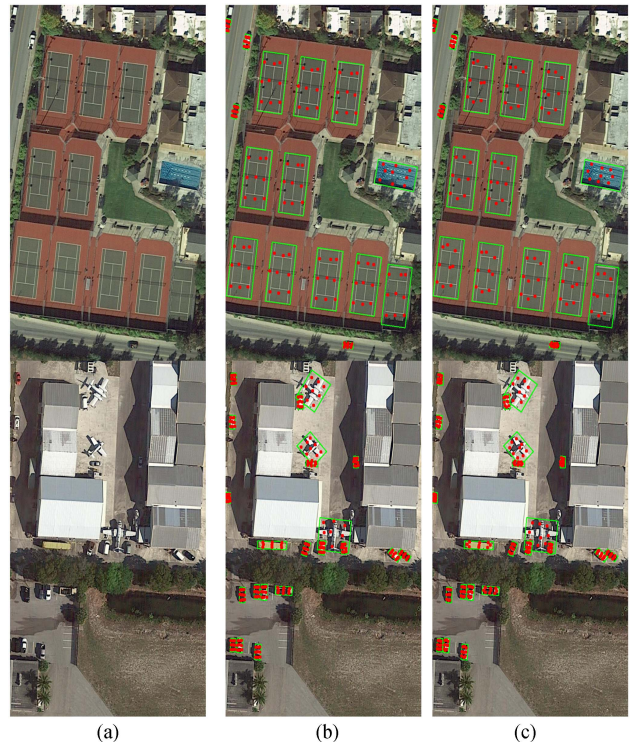


Fig. 7. Visualization of convolutional positions in our AAC module. (a) Original images to be detected. (b) ACC module visualization for the classification head. (c) ACC module visualization for the regression head. (a) Images. (b) Cls. (c) Reg.

decay the learning rate at the 24th and 33rd epochs. Specifically, during the ablation experiments, we train the model on the training set and evaluate it on the validation set. In contrast to other state-of-the-art methods, we train the model on both the training and validation sets and test it using the test set. When training on HRSC2016 [25], we conduct 36 epochs, and the learning rate decreases at the 24th and 33rd epochs. The remaining settings are the same as those used for DOTA [28]. If not specified, our settings are the same as MMRotate [2] settings.

### C. Ablation Studies

In this subsection, we carry out ablation experiments on the DOTA [28] and HRSC2016 [25] datasets to confirm the efficacy of our proposed method.

1) *Effectiveness of Attention Feature Fusion Module*: The outcomes of our experiments are shown in Table II. First, we use FPN [4] as a feature fusion method, achieving 69.44% accuracy. With the addition of the channel attention module, mAP improves to 69.72%; with the addition of the spatial attention module, mAP further improves to 69.77%. When both

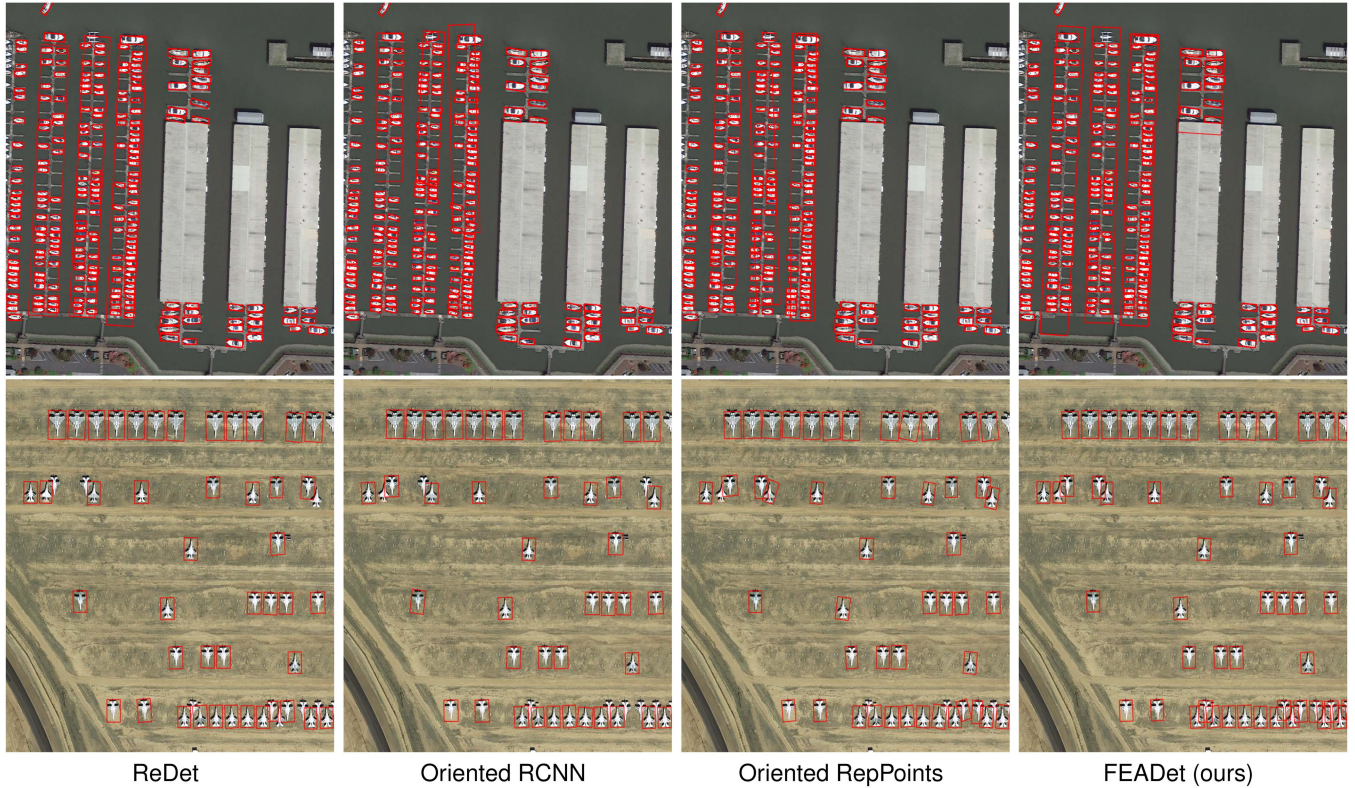


Fig. 8. Visual comparison of our FEADet with other advanced detectors on the DOTA dataset.

TABLE III  
COMPARISON OF OUR ADAPTIVE ALIGNCONV WITH OTHER CONVOLUTION METHODS ON DOTA DATASET

Model	Feature refinement	Offset loss	mAP(%)
baseline	Conv		69.55
	DeformConv		69.82
	AlignConv		70.28
	Adaptive AlignConv		70.59
	Adaptive AlignConv	✓	<b>70.74</b>

The bold results indicate the best for each column.

modules are added, mAP reaches 69.99%. This demonstrates that the attention feature fusion module significantly enhances the performance of the model. We visualized the extracted features, as shown in Fig. 6. Compared with FPN, our AAF module is able to better eliminate the aliasing effect caused by high-level feature maps and extract richer semantic information. This indicates that the AAF module can better capture the key features of the object, thus improving the accuracy of object detection.

2) *Effectiveness of Adaptive AlignConv Module*: To verify its effectiveness, we compare the adaptive AlignConv with other feature alignment approaches. The outcomes are shown in Table III. The AlignConv approach outperforms the deformable convolution and the standard convolution by achieving an mAP of 70.28%. This is because AlignConv uses the predicted rough boundary box as a prior knowledge to better extract the characteristics of the object. Our adaptive alignment convolution is superior to the alignment convolution, which can adaptively learn the sampling position of features and achieve 70.59% mAP. When adding offset loss to the learned offsets, mAP increases

TABLE IV  
ABLATION STUDIES WITH DIFFERENT COMPONENTS ON DOTA DATASET

Model	Feature enhance	Feature align	mAP(%)
baseline			69.44
	✓		69.99
		✓	70.74
	✓	✓	<b>71.05</b>

The bold results indicate the best for each column.

TABLE V  
ABLATION STUDIES WITH DIFFERENT COMPONENTS ON HRSC2016 DATASET

Model	Feature enhance	Feature align	mAP(%)
baseline			89.20
	✓		89.70
		✓	90.10
	✓	✓	<b>90.30</b>

The bold results indicate the best for each column.

to 70.74%, as it imposes constraints on the learned offsets and constrains points far from the object. The visualization results shown in Fig. 7 demonstrate that our method can combine explicit and implicit learning to adaptively learn object features. This means that our AAC module can better adapt to different feature representations of objects, thereby improving detection accuracy and robustness.

3) *Effectiveness of Different Components*: We perform componentwise experiments on DOTA [28] and HRSC2016 [25] datasets to confirm the contribution of the proposed methods. The results of the experiments are displayed in Tables IV and V. On the DOTA dataset, the mAP of the baseline model

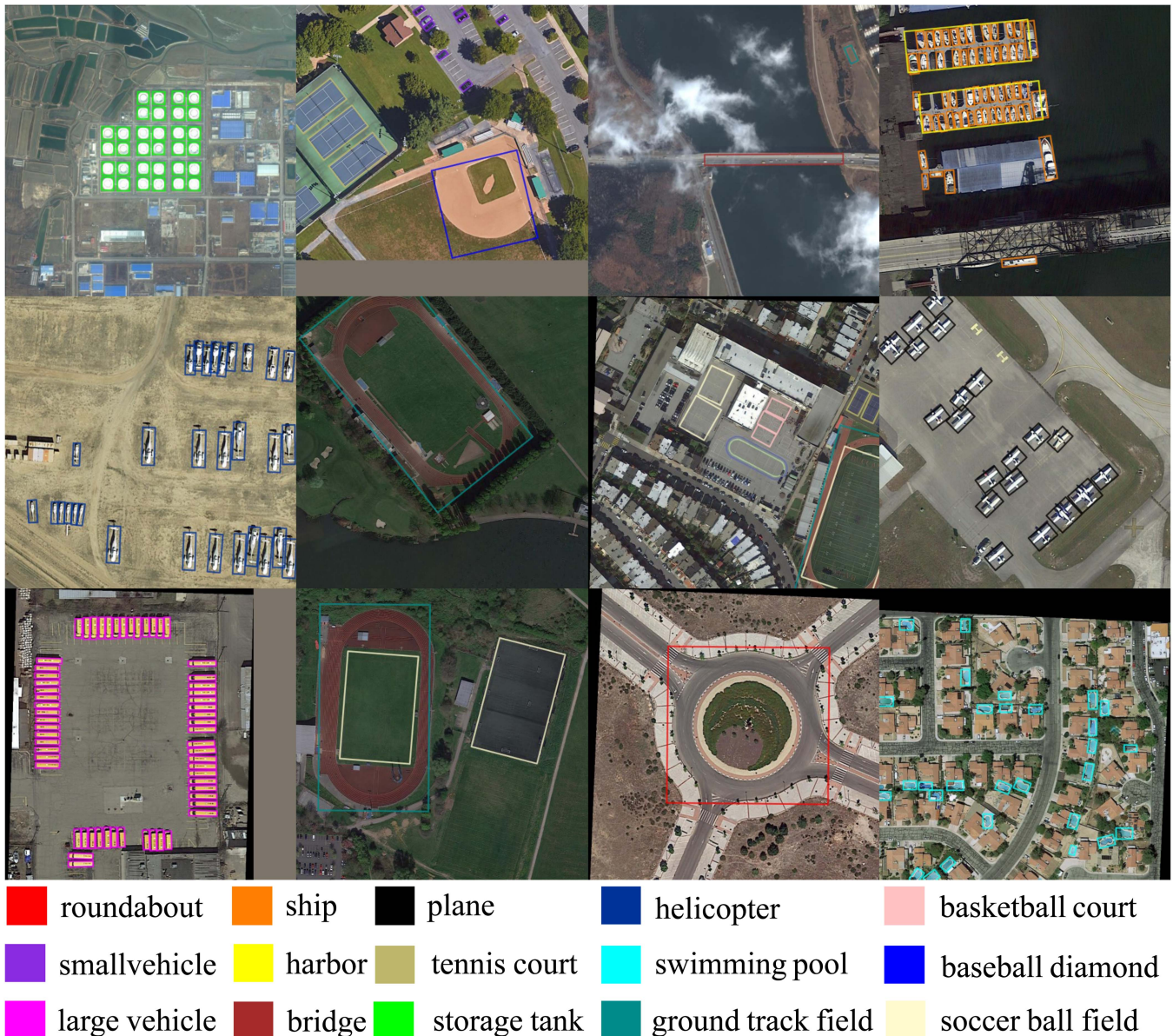


Fig. 9. Some visual results of the DOTA dataset using our FEADet.

was 69.44%, increased to 69.99% after adding the feature enhancement module, and further increased to 70.74% after adding the feature alignment module. Using both modules resulted in an mAP of 71.05%. On the HRSC2016 dataset, the mAP of the baseline model was 89.20%, increased to 89.70% after adding the feature enhancement module, and further increased to 90.10% after adding the feature alignment module. Using both modules resulted in an mAP of 90.30%. These experimental results indicate that the suggested AFF module and AAC module do not conflict with each other, but instead work in synergy to effectively enhance the performance of the model.

#### D. Comparisons With State-of-the-Art Detectors

1) *Results on DOTA*: As depicted in Table I, we evaluate our model against the state-of-the-art methods on the DOTA [28]

dataset. Compared to the FCOS [3] implemented by the MM-Rotate [2] toolbox, our reimplementation of FCOS has higher accuracy, but the FPS decreases from 16.7 to 15.1. The main reason is that CSL [1] is used for angle prediction, and the encoding and decoding process is time-consuming compared to direct prediction. The proposed FEADet achieved 74.37% mAP in 11.5 FPS with the ResNet50 [6]. When augmented with rotational data and trained with 36 epochs, the mAP of 77.05% is achieved. When using ResNet101 as the backbone, the accuracy further increased to 77.10% in 9.1 FPS. As illustrated in Fig. 8, our method has a higher detection accuracy compared to other advanced detectors. We visualize the some results of our detection in the DOTA dataset. The results, depicted in Fig. 9, reveal that our detector can achieve high-quality bounding boxes and is capable of handling diverse classes.



TABLE VI  
COMPARISON OF OUR FEADet WITH STATE-OF-THE-ART METHODS ON  
HRSC2016 DATASET

Method	Backbone	mAP(%)
BBAVectors [16]	ResNet101	88.60
R <sup>3</sup> Det [24]	ResNet101	89.30
CFC-Net [38]	ResNet101	89.50
DAL [40]	ResNet101	89.77
S <sup>2</sup> a-Net [9]	ResNet101	90.17
FEADet(ours)	ResNet50	90.30
FEADet(ours)	ResNet101	<b>90.40</b>

The bold results indicate the best for each column.

2) *Results on HRSC2016*: Our experiments on the HRSC2016 [25] dataset, which includes ship objects with large aspect ratios and arbitrary orientations, are presented in Table VI, along with comparisons with other methods. By using ResNet50 [6] as our backbone with an input image size of  $800 \times 800$ , we achieve an mAP of 90.30%. When using ResNet101 as the backbone, we achieved an mAP of 90.40%. The results show that in the detection of rotated objects, the proposed approach is better than other methods.

## V. CONCLUSION

The article proposes FEADet, a single-stage anchor-free detector for detecting oriented objects. Our method incorporates an AAF module and an AAC module. The AAF module facilitates the integration of higher level semantic information into lower level feature maps and mitigates the confounding effect of higher level features during fusion. The AAC module adapts to the position of object sampling points and improves feature alignment for object detection. We perform exhaustive experiments on the HRSC2016 and DOTA datasets, and the outcomes validate the effectiveness of our proposed approach.

## REFERENCES

- [1] X. Yang and J. Yan, "Arbitrary-oriented object detection with circular smooth label," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 677–694.
- [2] Y. Zhou et al., "Mmrotate: A rotated object detection benchmark using pytorch," in *Proc. ACM 30th ACM Int. Conf. Multimedia*, 2022, pp. 7331–7334.
- [3] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully convolutional one-stage object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9626–9635.
- [4] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 936–944.
- [5] S. Zhang, C. Chi, Y. Yao, Z. Lei, and S. Z. Li, "Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops. Comput. Vis. Found.*, 2020, pp. 9756–9765.
- [6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2016, pp. 770–778.
- [7] H. Zhang, Y. Wang, F. Dayoub, and N. Sünderhauf, "Varifocalnet: An iou-aware dense object detector," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2021, pp. 8514–8523.
- [8] H. Rezatofghi, N. Tsou, J. Gwak, A. Sadeghian, I. D. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 658–666.
- [9] J. Han, J. Ding, J. Li, and G. Xia, "Align deep features for oriented object detection," *IEEE Trans. Geosci. Remote. Sens.*, vol. 60, 2022, Art. no. 5602511.
- [10] J. Dai et al., "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 764–773.
- [11] J. Ma et al., "Arbitrary-oriented scene text detection via rotation proposals," *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3111–3122, Nov. 2018.
- [12] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.
- [13] J. Ding, N. Xue, Y. Long, G. Xia, and Q. Lu, "Learning ROI transformer for oriented object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 2849–2858.
- [14] Z. Li, B. Hou, Z. Wu, L. Jiao, B. Ren, and C. Yang, "FCOSR: A simple anchor-free rotated detector for aerial object detection," 2021, *arXiv:2111.10780*.
- [15] S. Chen, B. Dai, J. Tang, B. Luo, W. Wang, and K. Lv, "A refined single-stage detector with feature enhancement and alignment for oriented objects," *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, vol. 14, pp. 8898–8908, 2021.
- [16] J. Yi, P. Wu, B. Liu, Q. Huang, H. Qu, and D. N. Metaxas, "Oriented object detection in aerial images with box boundary-aware vectors," in *Proc. Winter Conf. Appl. Comput. Vis.*, 2021, pp. 2149–2158.
- [17] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.
- [18] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8759–8768.
- [19] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 10778–10787.
- [20] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.
- [21] S. Woo, J. Park, J. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.
- [22] Z. Qiao, D. Shi, X. Yi, Y. Shi, Y. Liu, and Y. Zhang, "Uefpn: Unified and enhanced feature pyramid networks for small object detection," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 19, no. 2s, pp. 1–21, Sep. 2022.
- [23] Y. Yu, X. Yang, J. Li, and X. Gao, "Object detection for aerial images with feature enhancement and soft label assignment," *IEEE Trans. Geosci. Remote. Sens.*, vol. 60, 2022, Art. no. 5624216.
- [24] X. Yang, J. Yan, Z. Feng, and T. He, "R3det: Refined single-stage detector with feature refinement for rotating object," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 3163–3171.
- [25] Z. Liu, L. Yuan, L. Weng, and Y. Yang, "A high resolution optical satellite image dataset for ship recognition and some new baselines," in *Proc. Int. Conf. Pattern Recognit. Appl. Methods*, 2017, pp. 324–331.
- [26] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," 2019, *arXiv:1904.07850*.
- [27] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, 2017, pp. 2980–2988.
- [28] G. Xia et al., "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Comput. Vis. Found.*, 2018, pp. 3974–3983.
- [29] S. M. Azimi, E. Vig, R. Bahmanyar, M. Körner, and P. Reinartz, "Towards multi-class object detection in unconstrained remote sensing imagery," in *Proc. Asian Conf. Comput. Vis.*, 2018, pp. 150–165.
- [30] Z. Zhang, W. Guo, S. Zhu, and W. Yu, "Toward arbitrary-oriented ship detection with rotated region proposal and discrimination networks," *IEEE Geosci. Remote. Sens. Lett.*, vol. 15, no. 11, pp. 1745–1749, Nov. 2018.
- [31] X. Yang et al., "Scrdet: Towards more robust detection for small, cluttered and rotated objects," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Seoul, Korea, 2019, pp. 8231–8240.
- [32] J. Wang, J. Ding, H. Guo, W. Cheng, T. Pan, and W. Yang, "Mask OBB: A semantic attention-based mask oriented bounding box representation for multi-category object detection in aerial images," *Remote Sens.*, vol. 11, no. 24, 2019, Art. no. 2930.
- [33] Y. Xu et al., "Gliding vertex on the horizontal bounding box for multi-oriented object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1452–1459, Apr. 2021.
- [34] J. Han, J. Ding, N. Xue, and G. Xia, "Redet: A rotation-equivariant detector for aerial object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Virtual*, 2021, pp. 2786–2795.
- [35] X. Xie, G. Cheng, J. Wang, X. Yao, and J. Han, "Oriented R-CNN for object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Montreal, QC, Canada, 2021, pp. 3500–3509.

- [36] Z. Chen et al., “Piou loss: Towards accurate oriented object detection in complex environments,” in *Proc. Comput. Vis. 16th Eur. Conf.*, Glasgow, U.K., 2020, pp. 195–211.
- [37] H. Wei, Y. Zhang, Z. Chang, H. Li, H. Wang, and X. Sun, “Oriented objects as pairs of middle lines,” *ISPRS J. Photogrammetry Remote Sens.*, vol. 169, pp. 268–279, 2020.
- [38] Q. Ming, L. Miao, Z. Zhou, and Y. Dong, “CFC-Net: A critical feature capturing network for arbitrary-oriented object detection in remote-sensing images,” *IEEE Trans. Geosci. Remote. Sens.*, vol. 60, 2022, Art. no. 5605814.
- [39] Z. Huang, W. Li, X. Xia, and R. Tao, “A general Gaussian heatmap label assignment for arbitrary-oriented object detection,” *IEEE Trans. Image Process.*, vol. 31, pp. 1895–1910, 2022.
- [40] Q. Ming, Z. Zhou, L. Miao, H. Zhang, and L. Li, “Dynamic anchor learning for arbitrary-oriented object detection,” in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 2355–2363.
- [41] Y. Zhang et al., “Orientation-first strategy with angle attention module for rotated object detection in remote sensing images,” *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, vol. 15, pp. 8492–8505, 2022.
- [42] Y. Li, Q. Huang, X. Pei, Y. Chen, L. Jiao, and R. Shang, “Cross-layer attention network for small object detection in remote sensing imagery,” *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, vol. 14, pp. 2148–2161, 2021.
- [43] W. Li, Y. Chen, K. Hu, and J. Zhu, “Oriented reppoints for aerial object detection,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, New Orleans, LA, USA, 2022, pp. 1819–1828.
- [44] Z. Huang, W. Li, X. Xia, H. Wang, and R. Tao, “Task-wise sampling convolutions for arbitrary-oriented object detection in aerial images,” 2022, *arXiv:2209.02200*.
- [45] Q. Xu, Y. Li, and Z. Shi, “LMO-YOLO: A ship detection model for low-resolution optical satellite imagery,” *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.*, vol. 15, pp. 4117–4131, 2022.
- [46] M. Zhang, W. Li, Y. Zhang, R. Tao, and Q. Du, “Hyperspectral and LiDAR data classification based on structural optimization transmission,” *IEEE Trans. Cybern.*, vol. 53, no. 5, pp. 3153–3164, May 2023.
- [47] K. Wang, S. Du, C. Liu, and Z. Cao, “Interior attention-aware network for infrared small target detection,” *IEEE Trans. Geosci. Remote. Sens.*, vol. 60, 2022, Art. no. 5002013.



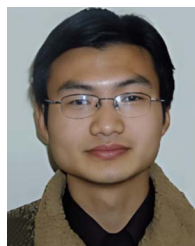
**Xu Xie** received the B.Mgt. degree in information management and information system in 2021 from Anhui University, Hefei, China, where he is currently working toward the master’s degree in computer science.

His current research interests include object detection and semantic segmentation.



**Zhi-Hui You** received the B.S. degree in computer science from Jiangxi Agricultural University, Nanchang, China, in 2020. He is currently working toward the Ph.D. degree in computer science with the Anhui University, Hefei, China.

His research interests include machine learning, pattern recognition, semantic segmentation, and change detection in remote sensing.



**Si-Bao Chen** (Member, IEEE) received the B.S. and M.S. degrees in probability and statistics and the Ph.D. degree in computer science from Anhui University, Hefei, China, in 2000, 2003, and 2006, respectively.

From 2006 to 2008, he was a Postdoctoral Researcher with the University of Science and Technology of China, Hefei. Since 2008, he has been a Teacher with Anhui University. From 2014 to 2015, he was a Visiting Scholar with the University of Texas in Arlington, Arlington, TX, USA. His current research interests include image processing, pattern recognition, machine learning, and computer vision.



**Li-Li Huang** received the B.E. and master’s degrees in computer science from Anhui University, Hefei, China, in 2010 and 2013, respectively, and the Ph.D. degree in computer science from Sun Yat-sen University, Guangzhou, China, in 2019.

She is currently a Teacher with the School of Computer Science and Technology, Anhui University. Her current research interests include artificial intelligence, computer vision, and cognitive science.



**Jin Tang** received the B.Eng. degree in automation and the Ph.D. degree in computer science from Anhui University, Hefei, China, in 1999 and 2007, respectively.

He is currently a Professor with the School of Computer Science and Technology, Anhui University. His current research interests include computer vision, pattern recognition, machine learning, and deep learning.



**Bin Luo** received the B.Eng. degree in electronics and the M.Eng. degree in computer science from Anhui University, Hefei, China, in 1984 and 1991, respectively, and the Ph.D. degree in computer science from the University of York, York, U.K., in 2002.

From 2000 to 2004, he was a Research Associate with the University of York. He is currently a Professor with Anhui University. His current research interests include graph spectral analysis, image and graph matching, statistical pattern recognition, digital watermarking, and information security.