remote sensing

MDPI

Article

ADT-Det: Adaptive Dynamic Refined Single-Stage Transformer Detector for Arbitrary-Oriented Object Detection in Satellite Optical Imagery

Yongbin Zheng *,†, Peng Sun †, Zongtan Zhou, Wanying Xu and Qiang Ren

College of Intelligence Science and Technology, National University of Defense Technology, Changsha 410073, China; sunpeng@nudt.edu.cn (P.S.); ztzhou@nudt.edu.cn (Z.Z.); wanyingxu@nudt.edu.cn (W.X.); renqiang@nudt.edu.cn (Q.R.)

* Correspondence: zybnudt@nudt.edu.cn

† Y. Zheng and P. Sun contributed equally to this work.

Abstract: The detection of arbitrary-oriented and multi-scale objects in satellite optical imagery is an important task in remote sensing and computer vision. Despite significant research efforts, such detection remains largely unsolved due to the diversity of patterns in orientation, scale, aspect ratio, and visual appearance; the dense distribution of objects; and extreme imbalances in categories. In this paper, we propose an adaptive dynamic refined single-stage transformer detector to address the aforementioned challenges, aiming to achieve high recall and speed. Our detector realizes rotated object detection with RetinaNet as the baseline. Firstly, we propose a feature pyramid transformer (FPT) to enhance feature extraction of the rotated object detection framework through a feature interaction mechanism. This is beneficial for the detection of objects with diverse patterns in terms of scale, aspect ratio, visual appearance, and dense distributions. Secondly, we design two special post-processing steps for rotated objects with arbitrary orientations, large aspect ratios and dense distributions. The output features of FPT are fed into post-processing steps. In the first step, it performs the preliminary regression of locations and angle anchors for the refinement step. In the refinement step, it performs adaptive feature refinement first and then gives the final object detection result precisely. The main architecture of the refinement step is dynamic feature refinement (DFR), which is proposed to adaptively adjust the feature map and reconstruct a new feature map for arbitrary-oriented object detection to alleviate the mismatches between rotated bounding boxes and axis-aligned receptive fields. Thirdly, the focus loss is adopted to deal with the category imbalance problem. Experiments on two challenging satellite optical imagery public datasets, DOTA and HRSC2016, demonstrate that the proposed ADT-Det detector achieves a state-of-the-art detection accuracy (79.95% mAP for DOTA and 93.47% mAP for HRSC2016) while running very fast (14.6 fps with a 600 × 600 input image size).

Keywords: arbitrary-oriented object detection in satellite optical imagery; adaptive dynamic refined single-stage transformer detector; feature pyramid transformer; dynamic feature refinement

Citation: Zheng, Y.; Sun, P.; Zhou, Z.; Xu, W.; Ren, Q. ADT-Det: Adaptive Dynamic Refined Single-Stage Transformer Detector for Arbitrary-Oriented Object Detection in Satellite Optical Imagery. *Remote Sens.* **2021**, *13*, 2623. https://doi.org/10.3390/rs13132623

Academic Editors: Anwaar Ulhaq, Douglas Pinto Sampaio Gomes and Danfeng Hong

Received: 19 May 2021
Accepted: 30 June 2021
Published: 4 July 2021

1. Introduction

In the past few decades, Earth observation satellites have been monitoring changes in the Earth's surface and the amount and resolution of satellite optical images have been greatly improved. The task of object detection in satellite optical images is to localize interest objects (such as vehicles, ships, aircraft, buildings, airports, ports) and identify their categories. This has numerous practical applications in satellite remote sensing and computer vision, warning of natural disasters, Earth surveying and mapping, and surveillance and traffic planning. Much progress in general-purpose horizontal detectors has been achieved by advances in deep convolutional neural networks (DCNNs) and the emergence of large datasets [1]. However, unlike natural images that are usually taken from horizontal

perspectives, satellite optical images are taken with a bird's eye view, which often leads to the arbitrary orientation of objects in satellite images [2], as shown in Figure 1. Moreover, as mentioned in [2–4], the following significant challenges further increase the difficulty of object detection in satellite optical images:

- Large-scale difference. Objects in satellite images vary in size hugely [5]. There are small objects such as cars, ships, aircraft, and small houses in satellite images, as well as large objects such as ports, airports, ground track fields, bridges, and large buildings. In addition, the size of objects within the same category (such as large aircraft and small aircraft) in the same image also varies greatly.
- Dense distribution. There are many densely distributed objects in satellite optical images, such as cars and ships [5].
- Large aspect ratio. There are lots of objects with large aspect ratios, such as large vehicles, ships, harbors, and bridges in satellite optical images. The mismatch between the ground truth bounding box and the predicted bounding box of these objects is very sensitive to the rotation angle of objects [4].
- Category imbalance. Satellite optical imagery datasets are long-tailed, and the number of instances in each category varies greatly. For example, the amount of small vehicles is about 105 times larger than that of soccer ball fields in satellite optical imagery.

Figure 1. Examples of objects with various orientations in satellite optical imagery.

Recent research [6–9] has focused on the design of rotation detectors, which apply rotated regions of interest (RRoI) instead of horizontal regions of interest (HRoI). To meet the above challenges, a framework for rotated object detection consisting of a rotation learning stage and a feature refinement stage is proposed to improve the detection accuracy. Despite the fact that some newly developed rotated object detection methods [10–14] have made some progress in this area, their performance still falls considerably below that required for real-world applications. A main reason for their low detection performance is improper feature extraction for instances with arbitrary orientations, large aspect ratios, and dense distributions. As shown in Figure 2a, the general receptive field of deep neural network-based detectors is axis-aligned and square, representing a mismatch with the actual shape of the instances, and this usually produces false detections. Thus, our goal is to design a special feature pyramid transformer and feature refinement module which can be adjusted adaptively according to the angle and scale of the instance, as shown in Figure 2b. Then, we introduce the above methods into the rotated object detection framework to help extract more accurate features.

(**a**) (**b**)

Figure 2. Comparison of receptive fields between (**a**) an axis-aligned neuron and (**b**) an adaptive neuron. The green rectangle represents the boundary of the instance, and the gray rectangle represents the boundary of the receptive field.

In this paper, we propose an adaptive dynamic refined single-stage transformer detector to address the aforementioned challenges, aiming to achieve a high recall and speed. Our detector realizes rotated object detection with RetinaNet as the baseline. Firstly, the feature pyramid transformer (FPT) is introduced into the traditional feature pyramid network (FPN) to enhance feature extraction through a feature interaction mechanism. This is beneficial for the detection of multi-scale objects and densely distributed objects. Secondly, the output features of FPT are fed into two post-processing steps. In the first step, the preliminary regression of locations and angle anchors for the refinement step is performed. In the refinement step, adaptive feature refinement is performed first and then the final object detection result is given precisely. The main architecture of the refinement step is the dynamic feature refinement (DFR), which is proposed to adaptively adjust the feature map and reconstruct a new feature map for arbitrary-oriented object detection to alleviate the mismatches between rotated bounding boxes and axis-aligned receptive fields. Experiments are carried out on two challenging satellite optical imagery public datasets, DOTA and HRSC2016, to demonstrate that our method outperforms previous state-of-the-art methods while running very fast.

The contributions of this work are three-fold:

(1) We propose a feature pyramid transformer for the feature extraction of the rotated object detection framework. This is beneficial for detecting objects with diverse patterns in terms of scale, aspect ratio, and visual appearance, and helps with the handling of challenging scenes with densely distributed instances through a feature interaction mechanism.

(2) We propose a dynamic feature refinement method for rotated objects with arbitrary orientations, large aspect ratios, and dense distributions. This can help to alleviate the bounding box mismatch problem.

(3) The proposed ADT-Det detector outperforms previous state-of-the-art detectors in terms of accuracy while running very fast.

2. Related Studies

Along with the wide application of satellite remote sensing and unmanned aerial vehicles, the amount of satellite optical imagery is increasing tremendously and object detection in satellite optical imagery has received increasing attention in the computer vision and remote sensing communities. Researchers have introduced DCNN-based detectors for object detection in satellite optical imagery, and oriented bounding boxes have been used instead of horizontal bounding boxes to reduce the mismatch between the predicted bounding box and corresponding objects. DCNN-based detectors are now reported as state-of-the-art.

In this section, we briefly review some previous well-known object detection methods in satellite or aerial optical images. In Section 2.1, we review the current mainstream

detectors used for satellite optical image detection. In Section 2.2, we summarize some classical designs of DCNN-based detectors that can improve the detection performance.

2.1. The Mainstream Detectors for Object Detection in Satellite Optical Imagery

The current mainstream detectors for satellite optical image detection are rotation detectors. Existing rotation detectors are mostly employed as alternatives to horizontal bounding boxes. Generally, these detectors can be organized into two main categories: multi-stage detectors and single-stage detectors.

The framework of multi-stage detectors includes a pre-processing stage for region proposal and one or more post-processing stages to regress the bounding box of an object and identify its category. In the pre-processing stage, classification-independent region proposals are generated from an input image. Then, CNNs with a special architecture are used to subsequently extract features from these regions, and regression and classification are performed over the next several stages [3,4]. In the last stage, the final detection results are generated by non-maximum suppression (NMS) or other methods. To the best of our knowledge, RoI-Transformer [2] and SCRDet [15] are state-of-the-art multi-stage rotated objects detectors. The RoI-Transformer is a two-stage rotated object detector. Its first stage is a RRoI Learner that generates a transformation from a horizontal bounding box to an oriented bounding box by learning from the annotated data. One important task in the second stage is RoI alignment, which extracts rotation-invariant features from the oriented RoI for subsequent object regression and classification. SCRDet introduced SF-Net [16] and MDA-Net into Faster-RCNN [17] to detect small and densely distributed objects. By introducing the Intersection over Union (IoU) factor into the traditional smooth L_1 loss function, the IoU-Smooth L_1 Loss enables the angle regression to be more concise. Generally, the numerous redundant region proposals make multi-stage detectors more accurate than anchor-free detectors. However, they rely on a more complicated structure, which greatly reduces their speed.

Single-stage object detectors drop the complex and redundant region proposal network, directly regress the bounding box, and identify the category of objects. YOLO [18–20] treats object detection as a regression task. Image pixels are regressed to spatially separate bounding boxes and associate them with class probabilities using the GoogLeNet network. Its improved versions are YOLOv2 and YOLO9000, in which GoogLeNet is replaced by a simpler Dark-Net19 and some special strategies (e.g., batch normalization) are introduced. Liu et al. [21] proposed SSD to preserve the real-time speed while keeping the detection accuracy as high as possible. Just like YOLO, a fixed number of bounding boxes and scores are predicted for the presence of object category in these boxes, followed by a NMS [22] step to generate the final detection result. As observed in [5], the detection performance of general single-stage methods is considerably lower than that of multistage methods. Recently, R^3Det [4] and R^4Det [3] demonstrated high performance in detecting rotated objects in satellite optical images. R^3Det adopts RetinaNet [23] for the baseline and adds refinement to the network. The focal loss alleviates any imbalance between positive and negative samples. R^4Det proposed a single-stage object detection framework by introducing the recursive feature pyramid (RFP) into RetinaNet to integrate feature maps of different levels.

2.2. General Designs for DCNN-Based Object Detection in Satellite Optical Imagery

2.2.1. Feature Pyramid Networks (FPN)

In many DCNN-based object detection frameworks, FPN is a basic component used to extract multi-level features for detecting objects at different scales. Low-level features represent less semantic information but the resolution is higher; on the contrary, high-level features represent more semantic information but the resolution is lower. In order to make full use of low-level features and high-level features at the same time, Lin et al. [24] proposed a generic FPN approach to fuse a multi-scale feature pyramid with a top-down pathway and lateral connections. This has become the benchmark and performs well in

feature extraction. Using a feature pyramid transformer [25] is an effective way to perform feature interaction between different scales and spaces. The transformed feature pyramid has a richer context than the original pyramid while maintaining the same size. In this paper, we introduce an FPT to enhance feature interaction in the feature fusion step.

2.2.2. Spatial Transformer Network

Atrous convolution [26] is an initial spatial transformer network. It increases the reception field by injecting holes into the standard convolution. Many improvements in dilated convolution have been proposed in recent years. Atrous spatial pyramid pooling (ASPP) [27] and denseASPP [28] obtained better results by cascading convolutions with different dilated rates in various forms. The Deformable Convolutional Network (DCN) [29] provides new ideas for spatial transformer networks. DCN can adjust the convolution kernels to make the receptive field more suitable for the feature map. General convolution is mostly horizontal and square. DCN can dynamically adjust according to the feature shape. We expect that it can improve the detection performance by introducing DCN into the feature extraction for rotated object detection.

2.2.3. Refined Object Detectiors

The research in [30] indicates that a low IoU threshold usually produces noisy detections. However, due to the mismatch between the optimal IoU of the detector and the IoU of the input hypothesis, detection performance tends to degrade as the IoU thresholds increase. To address these problems, Cascade RCNN [30] uses multiple stages with sequentially increasing IoU thresholds to train detectors. The main idea of RefineDet [31] is to coarsely adjust the locations and sizes of anchors using an anchor refinement module first. This is then followed by a regression branch to obtain more precise box information. Unlike two-stage detectors, the currently single-stage detector with a refinement stage is not well resolved in this respect. Feature misalignment is still one of the main reasons for the poor performance of refined single-stage detectors.

In this paper, we propose an adaptive dynamic refined single-stage transformer detector to address the aforementioned challenges, aiming to achieve a high recall and speed. Our detector realizes rotated object detection with RetinaNet as the baseline to achieve the detection of multi-scale objects and densely distributed objects. Firstly, the feature pyramid transformer (FPT) is introduced into the traditional feature pyramid network (FPN) to enhance feature extraction through a feature interaction mechanism. Secondly, the output features of FPT are fed into two post-processing steps considering the mismatch between the rotated bounding box and the general axis-aligned receptive fields of CNN. Dynamic Feature Refinement (DFR) is introduced to the refinement step. The key idea of DFR is to adaptively adjust the feature map and reconstruct a new feature map for arbitrary-oriented object detection to alleviate the mismatches between the rotated bounding box and the axis-aligned receptive fields. Extensive experiments and ablation studies show that our method can achieve state-of-the-art results in the task of object detection.

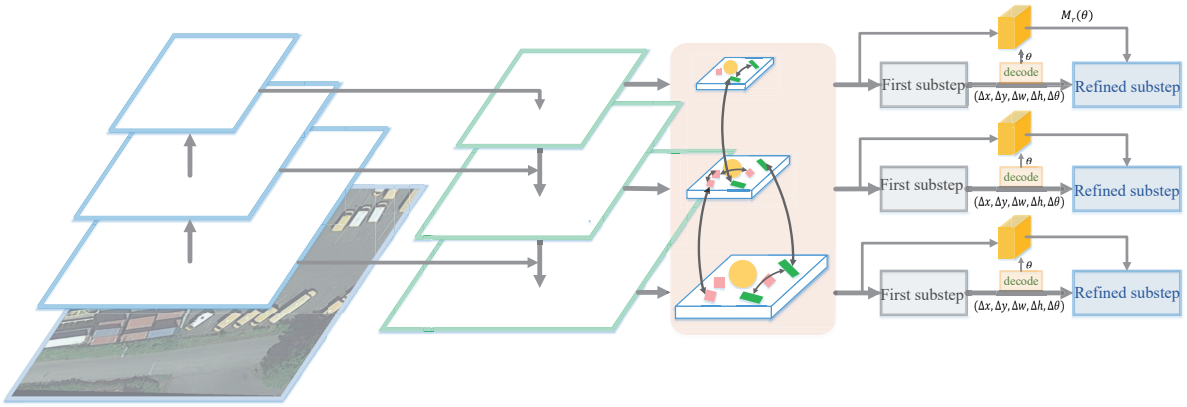
3. Methodology

In this section, we first describe our network architecture for arbitrary rotated object detection in Section 3.1. We then propose the feature pyramid transformer and dynamic feature refinement, which are our main contributions, in Sections 3.2 and 3.3, respectively. Finally, we show the details of our RetinaNet-based rotation detection method and the loss function in Section 3.4.

3.1. Network Architecture

The overall architecture of the proposed ADT-Det detector is sketched in Figure 3. Our pipeline improves upon RetinaNet and consists of a backbone network and two post-processing steps. The FPN network is utilized as the backbone and a feature pyramid transformer is proposed to enhance feature extraction for densely distributed instances.

Then, the backbone is attached in the post-processing steps. These consist of two sub-steps: first, a sub-step and a refinement sub-step, which will be described in detail in Sections 3.3 and 3.4. In the first sub-step, the preliminary regression of locations and angle anchors for the refinement sub-step is performed. In the refinement sub-step, adaptive feature refinement is performed first and then the final object detection result is given precisely. The main architecture of the refinement sub-step is the dynamic feature refinement (DFR), which is proposed to adaptively adjust the feature map and reconstruct a new feature map for rotated object detection (the detailed architecture of DFR is shown in Section 3.3). In the refinement sub-step, the feature fusion module (FFM) is considered as an important step to dynamically counteract the mismatch between the rotating object and the axis-aligned receptive fields of neurons. The overall framework is end-to-end trainable with a high efficiency.



(a) Feature Pyramid Network (FPN) (b) Feature Pyramid Transformer (c) Post-processing steps

Figure 3. The framework of the proposed ADT-Det detector. Our pipeline consists of a backbone network and two post-processing steps. An FPN network is used as backbone network and a feature pyramid transformer is proposed to enhance the feature extraction. Then, the backbone is attached in the post-processing steps, which consist of two sub-steps: first, a sub-step and a refinement step. In the first sub-step, the preliminary regression of locations and angles for the refinement sub-step is performed. In the refinement sub-step, adaptive feature refinement is performed first and then the final object detection result is given precisely.

3.2. Feature Pyramid Transformer

We introduce a feature pyramid transformer (FPT) and add it between the backbone FPN network and the post-processing network to produce features with stronger semantic information. Its architecture is shown in Figure 4. Firstly, the features from FPN are transformed and re-arranged. Then, the output features are concatenated with the original feature map to obtain the concatenated features. Finally, the Conv3×3 operation is carried out to reduce the channel and obtain the transformed feature pyramid.

The FPT is a light network that enhances features through feature interaction with multiple scales and layers. It allows features of different levels to interact across space and scale. The FPT consists of three transformer steps: a self-transformer, a grounding transformer, and a rendering transformer. The self-transformer is introduced to capture objects that appear simultaneously on the same feature map. The grounding transformer is a up-bottom non-local interaction transformer that is used to enhance shallow features with different levels of features. As shown in Figure 5a,b, the inputs of the self-transformer and the grounding transformer are q_i, k_j, and v_j, where $q_i = f_q(X_i)$ represents the i-th query; $k_j = f_k(X_j)$ represents the j-th key; $v_j = f_v(X_j)$ represents the j-th value; and $f_q(.)$, $f_k(.)$, and $f_v(.)$ are used to perform queries, keys, and values operations on the feature map, respectively. The self-transformer adopts dot products as similarity function F_{sim} to capture co-occurring features in the same feature map. The output of F_{sim} is fed to the normalization function F_{norm} to generate weights $w_{(i,j)}$. Lastly, we multiply v_j and $w_{(i,j)}$ to

obtain the transformed feature X. Unlike the self-transformer, the grounding transformer is a top-down non-local interaction that is used to strengthen shallow features with deep features. It uses Euclidean distance to measure the similarity of deep features and shallow features. The rendering transformer works with a bottom-up transformer to interact with the entire feature map, presenting higher-level semantic features in lower-level features. The transformation process is shown in Figure 5c. First, we calculate the weight w of Q through global average pooling from the shallow feature K. Then, the weights of Q (Q_{att}) and V are refined by Conv3×3 to reduce the size of the feature map. Finally, the refined Q_{att} and down-sampled V (V_{down}) are summed and processed by another Conv3×3 for rendering.

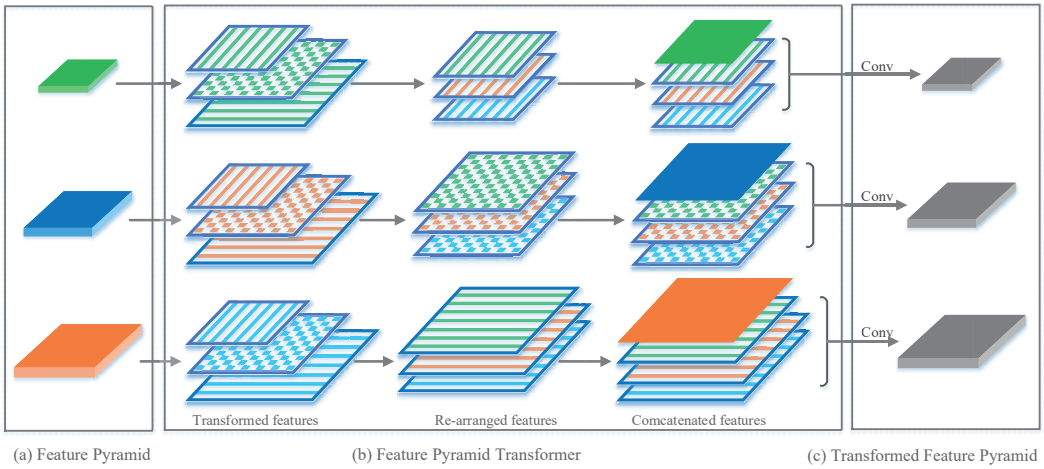


Figure 4. Three transformer steps: (**a**) self-transformer, (**b**) grounding transformer, (**c**) rendering transformer. $q_i = f_q(X_i)$ represents the i-th query, $k_j = f_k(X_j)$ represents the j-th key, and $v_j = f_v(X_j)$ represents the j-th value, where $f_q(.)$, $f_k(.)$, and $f_v(.)$ are used to perform queries, keys, and values operations on the feature map, respectively.

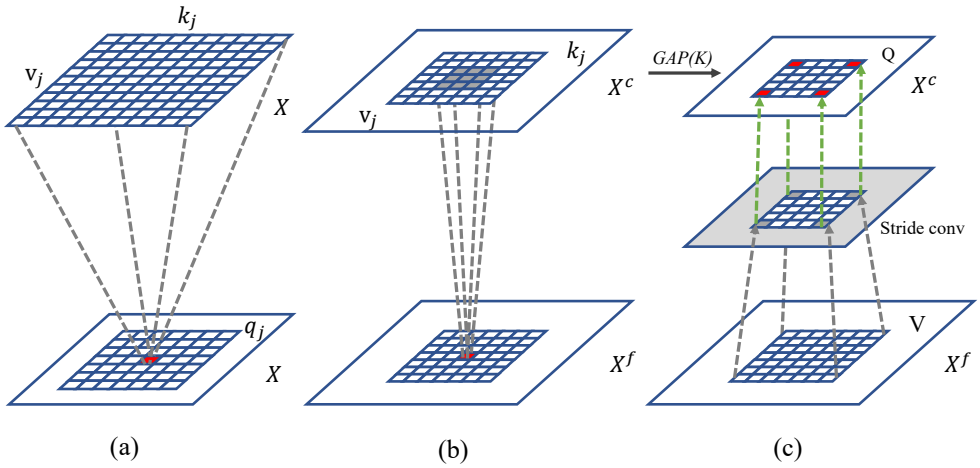


Figure 5. Architecture of the proposed feature pyramid transformer: (**a**) self-transformer, (**b**) grounding transformer, (**c**) rendering transformer. Firstly, the features from FPN are transformed and re-arranged. Then, the output features are concatenated with the original feature map to obtain the concatenated features. Finally, the Conv3×3 operation is carried out to reduce the channel and obtain the transformed feature pyramid.

3.3. Dynamic Feature Refinement

When detecting instances with arbitrary orientations, large aspect ratios, and dense distributions, the main reason for low detection performance is the feature misalignment problem, which is caused by differences in the scale and rotation between the orientated

bounding box and the axis-aligned receptive fields. To alleviate the feature misalignment problem, we introduce dynamic feature refinement (DFR) to obtain the refined accurate bounding box. The architecture of DFR is shown in the bottom of Figure 6.

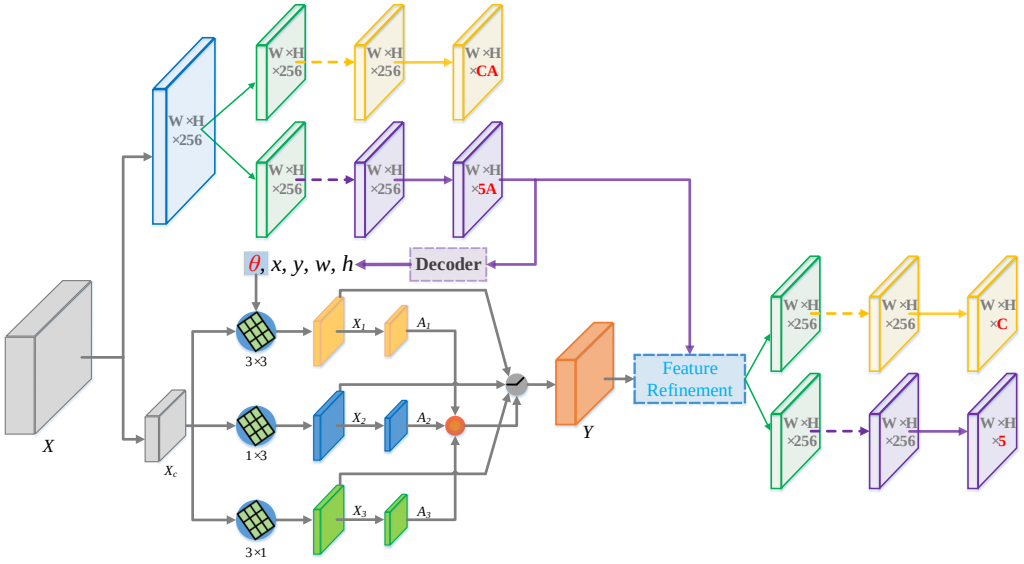


Figure 6. Architecture of the post-processing step. This consists of two sub-steps: the first sub-step and the refinement sub-step. Top: the first sub-step, which performs the preliminary regression of angle anchors for the refinement sub-step. Bottom: the refinement sub-step, which performs feature fusion and adaptive feature refinement and then gives the final object detection result precisely. On the left of the refinement sub-step is the feature fusion module, followed by the feature refinement module. On the right are two subnetworks, which perform object classification and regression.

We adopt a feature fusion module (FFM) to counteract the mismatches between arbitrary-orientation objects and axis-aligned receptive fields. This can dynamically and adaptively aggregate the features extracted by various kernel sizes, shapes (aspect ratios), and angles. The FFM takes the i-th stage feature map $X \in \mathbb{R}^{H\times W\times C}$ as an input and consists of two branches. In one branch, $X \in \mathbb{R}^{H\times W\times C}$ is connected to the classification and regression subnetworks to decode the location feature information. This is a normal network introduced from RetinaNet. The task of this branch is to generate initial location information and decode the angle feature information. In the other branch, we compress $X \in \mathbb{R}^{H\times W\times C}$ with a Conv1×1 layer and aggregate the improved information using batch normalization and ReLU. In order to further deal with the mismatches between rotated objects and axis-aligned receptive fields, we introduce the adaptive convolution (AdaptConv) into our DFR.

The AdaptConv is inspired by [32], and the implementation details are illustrated in Figure 7. Similar to DCN in [29], $\Re$ denotes the regular grid receptive field and dilation. For a 3×3 kernel, we have:

$$\Re = \{(-1,-1),(-1,0),...,(0,1),(1,1)\} \tag{1}$$

The output of AdaptConv is:

$$X_i(p_0) = \sum_{p_n\in\Re} w(p_n)\cdot X_c(p_0+p_n+\delta p_n) \tag{2}$$

where p_n represents the locations in $\Re$, w denotes the kernel weights, and δp_n is the offset field for each location p_n. In our method, we redefine the offset field δp_n so that DCN can

be transformed into a regular convolution with angle information. The offset of AdaptConv is defined as follows:

$$\delta p_i = M_r(\theta) \cdot p_i - p_i \tag{3}$$

where $M_r(\theta) \in \mathbb{R}^{H \times W \times 1}$ is the angle feature information that is split and resized from the location feature information.

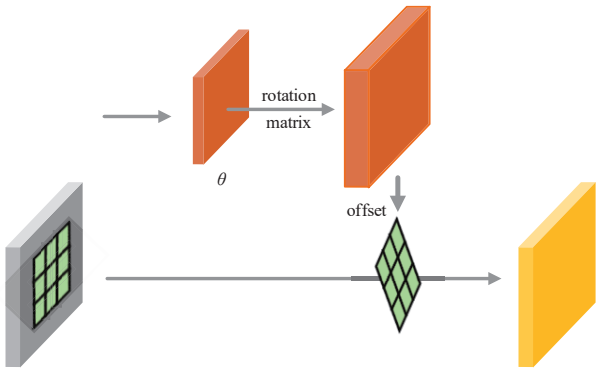


Figure 7. The overall process of AdaptConv. Decoded angle feature map θ is used to generate the offset. The special offset causes the DCN to have a receptive field with regular shape and angle information.

As shown in the bottom of Figure 6, in order to cope with objects with large aspect ratios, we use a three-split AdaptConv with 3×3, 1×3, and 3×1 kernels, which are denoted as $X_i \in \mathbb{R}^{H \times W \times C'}$ $(i \in \{1, 2, 3\})$, to extract multiple features from $X_c \in \mathbb{R}^{H \times W \times C'}$. In order to cause the receptive fields of neurons to adjust features dynamically, we adopt an attention mechanism to integrate features from the above three-split process. Let the attention map be $A_i \in \mathbb{R}^{H \times W \times 1} (i \in 1, 2, 3)$ and the computation be as follows:

Firstly, X_i is fed into the attention block, which is composed of Conv1×1 and the batch normalization operation. Secondly, $A_i (i = 1, 2, 3)$ is sent to SoftMax to obtain the normalized selection weight $A_i^{'}$:

$$A_i^{'} = SoftMax([A_1, A_2, A_3]) \tag{4}$$

Here, the SoftMax can be described as follows. Suppose v is a vector and v_i represents the i-th element in v. In this case, the SoftMax value of this element is formulated by:

$$p = \frac{e^{v_i}}{\sum_{j=1} e^{v_j}} \tag{5}$$

where the calculation result is between 0 and 1 and the sum of the SoftMax values of all elements is 1.

Thirdly, the feature map Y is obtained by implementing a ReLU operation on:

$$Y = \sum_i A_i^{'} \cdot X_i, \tag{6}$$

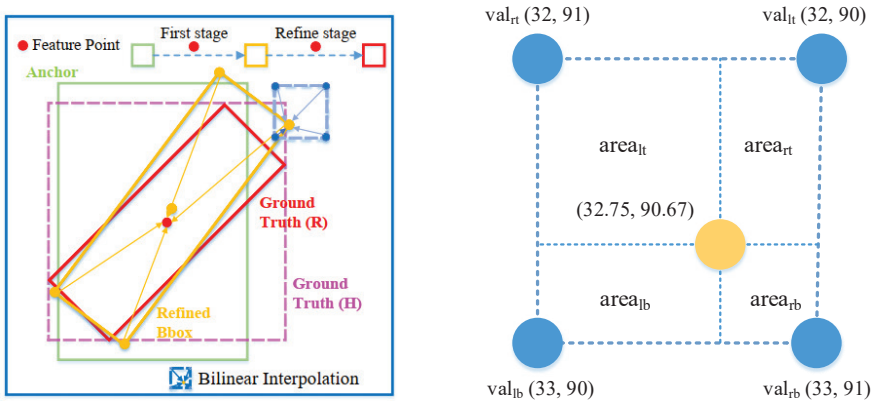
where $Y \in \mathbb{R}^{H \times W \times C}$ is the output feature.

The adjusted feature map Y is then sent to the feature refinement module (as shown in the middle of Figure 6) to reconstruct the features and achieve feature alignment. The feature alignment details are illustrated in Figure 8. For each feature map, the aligned feature vectors are obtained through interpolation, according to the five coordinates (orange points) of the refined bounding box. Following the method described in [4], we use feature bilinear

interpolation to generate more accurate feature vectors and replace the original feature vectors, as illustrated in Figure 8b. The bilinear interpolation is formulated as follows:

$$\begin{aligned} val = val_{lt} \times area_{rb} + val_{rt} \times area_{lb} \\ + val_{rb} \times area_{lt} + val_{lb} \times area_{rt}, \end{aligned} \tag{7}$$

where val denotes the result of bilinear interpolation. val_{lt}, val_{rt}, val_{rb}, and val_{lb} denote the values of the top-left, top-right, bottom-right, and bottom-left pixel, respectively. $area_{lt}$, $area_{rt}$, $area_{rb}$, and $area_{lb}$ denote the area of the top-left, top-right, bottom-right, and bottom-left rectangles, respectively.



(**a**) (**b**)

Figure 8. Feature refinement. (**a**) Refine the bounding box with aligned features. (**b**) Feature bilinear interpolation.

3.4. RetinaNet-Based Rotation Detection and Loss Function

We achieve rotated bounding box detection by using the oriented rectangle representation method proposed in [4]. For the completeness of the content, let us introduce the method briefly. We use a vector with five parameters (x, y, w, h, θ) to represent an arbitrarily oriented bounding box, where (x, y) denotes the coordinates of the bounding box center, w and h denote the width and height of the bounding box, and θ denotes the rotation angle of the bounding box relative to the horizontal direction. Compared to the horizontal bounding box, an additional angular offset must be predicted in the regression subnet, for which the rotation bounding box is described as follows:

$$\begin{aligned} t_x = (x - x_a)/\omega_a, t_y = (y - y_a)/h_a \\ t_\omega = \log(\omega/\omega_a), t_h = \log(h/h_a), t_\theta = (\theta - \theta_a) \end{aligned} \tag{8}$$

$$\begin{aligned} t'_x = (x - x_a)/\omega_a, t'_y = (y' - y_a)/h_a \\ t'_\omega = \log(\omega'/\omega_a), t'_h = \log(h'/h_a), t'_\theta = (\theta' - \theta_a) \end{aligned} \tag{9}$$

where (x, x_a, x') correspond to the ground-truth box, the anchor box, and the predicted box, respectively (likewise for y, w, h, θ).

The definition of the multi-task loss function is as follows:

$$\begin{aligned} L = \frac{\lambda_1}{N} \sum_{n=1}^{N} t'_n \sum_{j \in \{x,y,w,h,\theta\}} \frac{L_{reg}\left(v'_{nj}, v_{nj}\right)}{\left|L_{reg}\left(v'_{nj}, v_{nj}\right)\right|} |-\log(IoU)| \\ + \frac{\lambda_2}{h \times w} \sum_{i}^{h} \sum_{j}^{w} L_{att}\left(u'_{nj}, u_{nj}\right) + \frac{\lambda_3}{N} \sum_{n=1}^{N} L_{cls}(p_n, t_n) \end{aligned} \tag{10}$$

where N denotes the anchor number and t'_n denotes a binary value ($t'_n = 1$ for the foreground and $t'_n = 0$ for the background). v'_{nj} denotes the predicted offset vectors, and v_{nj} denotes the vector of the ground truth, t_n denotes the instance label, and p_n denotes the probability of the categories calculated by the sigmoid function. The hyperparameters λ_1, λ_2, and λ_3 control the trade-off and are set to 1 by default. The classification loss L_{cls} is implemented using focal loss. In [23], the authors noticed that the imbalance of instances categories results in a low accuracy for a single-stage detector compared with that of a two-stage detector. They proposed focal loss to address this problem. Thus, we use focal loss to optimize our classification loss, whereby our detector maintains single-stage speed while improving the detection accuracy.

Equation (11) shows the cross-entropy loss function that produces focal loss:

$$CE(p_t, y) = -\log(p_t), p_t = \begin{cases} p & if \quad y = 1 \\ 1 - p & otherwise \end{cases} \tag{11}$$

where y $\in \{\pm 1\}$ specifies the ground-truth class and $p_t \in [0, 1]$ is the model's estimated probability for the class with the label y = 1.

Furthermore, a weighting factor $\alpha_t \in [0, 1]$ and a modulating factor $(1 - p_t)^\gamma$ ($\gamma \geq 0$) are introduced (as shown in Equation (12)) to control the weights of positive and negative instances, meaning that the training is relatively more focused on positive samples.

$$FL(p_t) = -\alpha_t (1 - p_t)^\gamma \log(p_t) \tag{12}$$

In the rotated object detection task, the loss is very large due to the periodicity of the angle. Therefore, the model has to be regressed in other complex forms, increasing the difficulty of regression. Yang [15] proposed a loss function by introducing the IoU constant factor in the traditional smooth L_1 loss. The smooth L_1 loss is expressed by:

$$Smooth_{L_1}(x) = \begin{cases} 0.5x^2 & |x| < 1 \\ |x| - 0.5 & x < 1 \quad or \quad x > 1 \end{cases} \tag{13}$$

The new regression loss can be divided into two parts, as shown in Equation (10), where $\frac{L_{reg}\left(v'_{nj}, v_{nj}\right)}{\left|L_{reg}\left(v'_{nj}, v_{nj}\right)\right|}$ determines the direction of gradient propagation and $|-\log(IoU)|$ determines the magnitude of the gradient.

4. Experiments and Analysis

4.1. Benchmark Datasets

Extensive experiments and ablation studies were conducted. We compared our detector with 8 other well-known detectors through experiments on two challenging satellite optical image benchmarks: DOTA [5] and HRSC2016 [33].

DOTA is the largest and most challenging dataset with both horizontal and oriented bounding box annotations for object detection in satellite or aerial optical images. It contains 2806 satellite images, whose sizes range from 800 × 800 to 4000 × 4000. DOTA contains objects with a wide variety of scales, orientations, and appearances. These images have been annotated by experts using 15 common object categories. The object categories include plane (PL), ship (SP), large vehicle (LV), small vehicle (SV), helicopter (HC), tennis court (TC), bridge (BR), ground track field (GTF), basketball court (BC), baseball diamond (BD), soccer field (SBF), storage tank (ST), roundabout (RA), harbor (HA), and swimming pool (SP). Among them, there are huge numbers of densely distributed objects, such as small vehicles, large vehicles, ships, and planes. There are many object categories with large aspect ratios, such as large vehicles, ships, harbors, and bridges. Two detection tasks with horizontal bounding boxes and orientated bounding boxes can be performed on DOTA. In our experiment, we chose the task of detecting objects with an orientated bounding box. An official website (https://captain-whu.github.io/DOTA/dataset.html (accessed on 1

January 2018) is provided for the submission of the results. DOTA contains 1403 training images, 468 verification images, and 935 testing images, which are randomly selected from the original images.

HRSC2016 [33] is a challenging satellite optical imagery dataset for ship detection. It contains 1061 images collected from Google Earth and over 20 categories of ship instances with different shapes, orientations, sizes, and backgrounds. The images with the scenario of ships close to the shore in HRSC2016 were collected from six famous harbors, while the other images show the scenario of ships on the sea. The image size ranges between 300×300 and 1500×900. HRSC2016 contains 436 training images, 181 validation images, and 444 testing images. During the training and testing, we resized the images to 800×800. In our experiment, we chose the task of detecting ships with an orientated bounding box.

4.2. Implementation Details

We adopted ResNet101 FPN as the backbone of the experiment. The hyperparameters of the multi-task loss function were set to $\lambda_1 = 4$, $\lambda_2 = 1$, and $\lambda_3 = 2$. The hyperparameters of the focal loss were set to $\alpha = 0.25$ and $\gamma = 2.0$. SGD [34] was adopted as an optimizer. The initial learning rate was set at 0.04 and the learning rate was divided by 10 at each decay step. The momentum and weight decay were set to 0.9 and 0.0001. The learning rate warmup was set to 500 iterations. We adopted mmdetections [35] as training schedules and trained all the models in 12 epochs for DOTA and 36 epochs for HRSC2016. We used a sever with 4 NVIDIA TITAN Xp GPUs and 4 GPUs with a total batch size of 8 for training and a single GPU for inference.

4.3. Ablation Study

In order to evaluate the impact of DFR, FPT, and data augmentation on our detector, we conducted some ablation studies on the DOTA and HRSC2016. ResNet-50 pretrained on ImageNet was used as a backbone in the experiments. The weight decay and momentum were set to 0.0001 and 0.9, respectively. Detectors were trained using 4 GPUs with a total of 8 images per mini batch (two images per GPU).

4.3.1. Ablation Study for DFR

In this subsection, we present the ablation study results for the original feature refinement module (FRM) and the proposed DFR. As shown in Table 1, RetinaNet has a 62.22% accuracy. By introducing FRM, R^3Det (RetinaNet with refinement) obtained a 71.69% accuracy under ResNet101-FPN as a backbone with no multi-scale. FRM improved the accuracy by 9.47%. In this study, we introduced DFR to achieve feature misalignment instead of FRM. The accuracy with DFR was 73.10%, which is 1.41% higher then the accuracy with FRM. As shown in Table 2, the accuracy for some hard instance categories, such as BR, SV, LV, SH, and RA, increased by 2.06%, 7.71%, 2.8%, 9.42%, and 2.84%, respectively. We can see that the proposed DFR has a significant effect on improving the performance.

Table 1. Ablation study of DFR, FPT, and data augmentation.

Methods	mAP	FRM	DFR	FPT	Data Aug.
RetinaNet [23]	62.22	×	-	-	-
R^3Det [4]	71.69	√	-	-	-
ADT-Det (ours)	73.10	-	√	×	×
	73.77	-	√	√	×
	76.89	-	√	√	√

Table 2. Ablation study of FRM and the proposed DFR, where FRM is the original feature refinement module proposed by R^3Det.

Methods	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
FRM	89.54	81.99	48.46	62.52	70.48	74.29	77.54	90.80	81.39	83.54	61.79	59.82	65.44	67.46	60.05	71.69
DFR	88.99	79.42	50.52	68.62	78.19	77.09	86.96	90.85	79.82	85.45	58.99	62.66	66.01	67.56	55.45	73.10

4.3.2. Ablation Study on FPT

As shown in Table 1, the accuracy was 73.10% without FPT and 73.77% with FPT. It can be seen that the proposed FPT has a slight effect on improving the performance.

4.3.3. Ablation Study for Data Augmentation

A previous study showed that data augmentation is a very effective way to improve detection performance by enriching training datasets. In this subsection, we study the impact of data augmentation on the detection accuracy of our detector. The data augmentation methods used in the experiment includes horizontal and vertical flipping, random graying, multi-scales, and random rotation. As shown in Table 1, the detection accuracy was improved from 73.77% to 76.89% by data augmentation.

4.4. Comparison to State of the Art

4.4.1. Results on DOTA

We compared our proposed detector with some state-of-the-art detectors using the DOTA dataset. The results reported here were obtained by submitting our detection results to the official DOTA evaluation server. All the detectors involved in this experiment can be divided into three groups: multi-stage, anchor-free, and single-stage detectors. As shown in Table 3, the latest multi-stage detectors, such as SCRDet [15], Gliding Vertex [10], and APE [36], achieved values of 69.56%, 72.61%, 75.02%, and 75.75% mAP, respectively. The anchor-free method DRN [32] achieved a 73.23% mAP. The single-stage detectors R^3Det and R^4Det with ResNet-152 had 73.73% and 75.84% accuracies. Our ADT-Det with ResNet-152 achieved the highest accuracy of 77.43%, which is 1.59% higher than the previous best result.

The research of R^4Det [3] showed that feature recursion is a good method to improve the detection accuracy. We also adopted feature recursion in our pipeline, and it outperformed state-of-art methods and achieved a 79.95% accuracy.

The visualization of some of the detection results of our detector is shown in Figure 9. The results demonstrate that our detector can accurately detect most objects with arbitrary orientations, large aspect ratios, huge scale differences, and dense distributions.

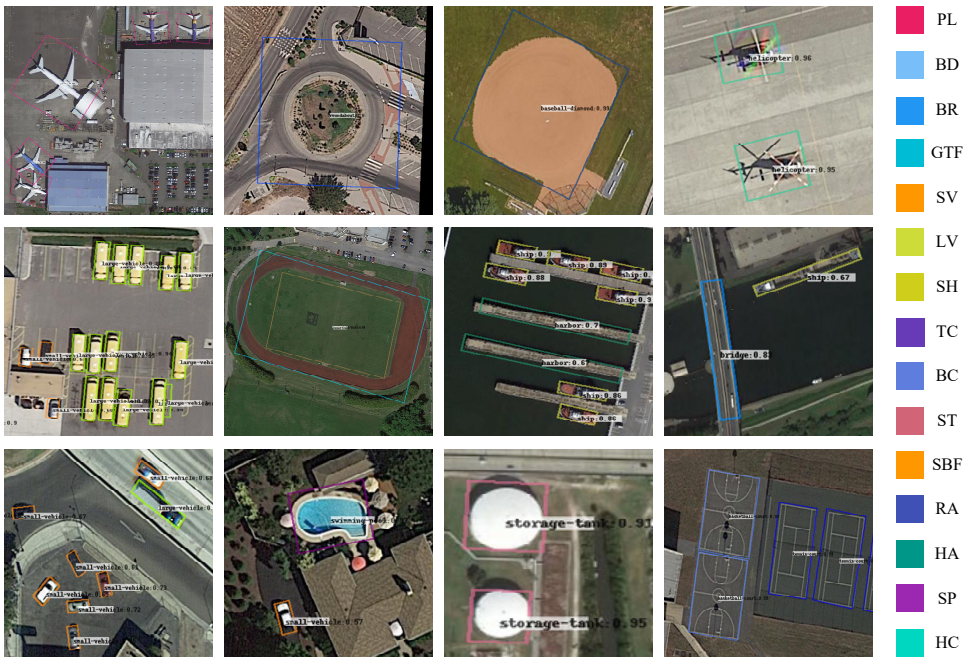


Figure 9. Visualization of some detection results on DOTA. Different colored bounding boxes represent instances of different categories (best viewed in color).

Table 3. Detection accuracy on different objects (AP) and overall performance (mAP) evaluation on DOTA.

Methods	PL	BD	BR	GTF	SV	LV	SH	TC	BC	ST	SBF	RA	HA	SP	HC	mAP
Two-stage methods																
R-FCN [12]	37.80	38.21	3.64	37.26	6.74	2.60	5.59	22.85	46.93	66.04	33.37	47.15	10.60	25.19	17.96	26.79
FR-H [5]	47.16	61.00	9.80	51.74	14.87	12.80	6.88	56.26	59.97	57.32	47.83	48.70	8.23	37.25	23.05	32.29
FR-O [5]	79.09	69.12	17.17	63.49	34.20	37.16	36.20	89.19	69.60	58.96	49.40	52.52	46.69	44.80	46.30	52.93
IE-Net [37]	80.20	64.54	39.82	32.07	49.71	65.01	52.58	81.45	44.66	78.51	46.54	56.73	64.40	64.24	36.75	57.14
R^2CNN [11]	80.94	65.67	35.34	67.44	59.92	50.91	55.81	90.67	66.92	72.39	55.06	52.23	55.14	53.35	48.22	60.67
RoI-Transformer [2]	88.64	78.54	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
SCRDet [15]	89.98	80.65	52.09	68.36	68.83	60.36	72.41	90.85	**87.94**	86.86	65.02	66.68	66.25	68.24	65.21	72.61
RSDet [4]	**90.10**	82.00	53.80	68.5	70.20	78.7	73.6	91.2	87.1	84.7	64.31	68.2	66.1	69.3	63.7	74.1
Gliding Vertex [10]	89.64	**85.00**	52.26	77.34	73.01	73.14	86.82	90.74	79.02	86.81	59.55	70.91	72.94	70.86	57.32	75.02
FFA [38]	90.10	82.70	54.20	75.20	71.00	79.90	83.50	90.70	83.90	84.60	61.20	68.0	70.70	76.00	63.70	75.00
APE [36]	89.96	83.64	53.42	76.03	74.01	77.16	79.45	90.83	87.15	84.51	67.72	60.33	74.61	71.84	65.55	75.75
Anchor-free methods																
DRN [32]	89.71	82.34	47.22	64.10	76.22	74.43	85.84	90.57	86.18	84.89	57.65	61.93	69.30	69.63	58.48	73.23
Single-stage methods																
SSD [21]	39.57	9.09	0.64	13.18	0.26	0.39	1.11	16.24	27.57	9.23	27.16	9.09	3.03	1.05	1.01	10.59
YOLO v2 [19]	39.49	20.29	36.58	23.42	8.85	2.09	4.82	44.34	38.35	34.65	16.02	37.62	47.23	25.5	7.45	21.39
R^3Det [4]-ResNet152	89.49	81.17	5.53	66.10	70.92	78.66	78.21	90.81	85.26	84.23	61.81	63.77	68.16	68.83	67.17	73.73
R^4Det [3]-ResNet152	88.96	85.42	52.91	73.84	74.86	81.52	80.29	90.79	86.95	85.25	64.05	60.93	69.00	70.55	**67.76**	75.84
ADT-Det (no Multi-Scale Training)	88.99	79.42	50.52	68.62	78.19	77.09	86.96	90.85	79.82	85.45	58.99	62.66	66.01	67.56	55.45	73.10
ADT-Det-ResNet50	89.28	83.97	51.44	79.12	78.31	82.18	87.79	90.82	84.84	87.46	65.47	64.23	71.87	71.40	65.08	76.89
ADT-Det-ResNet101	89.62	84.70	51.88	77.43	77.88	80.54	88.22	90.85	84.18	86.68	66.30	69.17	76.34	70.91	63.01	77.18
ADT-Det-ResNet152	89.61	84.59	53.18	**81.05**	78.31	80.86	88.22	90.82	84.80	**86.89**	69.97	66.78	76.18	72.10	60.03	77.43
ADT-Det (with Feature Recursion)	89.71	84.71	**59.63**	80.94	**80.30**	**83.53**	**88.94**	**90.86**	87.06	87.81	**70.72**	**70.92**	**78.66**	**79.40**	65.99	**79.95**

4.4.2. Result on HRSC2016

HRSC2016 contains many ship instances with large aspect ratios and arbitrary orientations. RRPN was originally developed for orientation scene text detection. RoI-Transformer and R^3Det are advanced satellite optical imagery detection methods. We performed comparative experiments with these methods, and the results are shown in Table 4. We can see that the scene text detection methods have competitive results for satellite optical imagery datasets; RRPN [13] achieved a 79.08% mAP. Under the PASCAL VOC2007 metrics, the famous multi-stage rotated object detector RoI-Transformer [2] could achieve an 86.20% accuracy. The state-of-art single-stage methods, R^3Det [4] and R^4Det [3], could achieve 89.26% and 89.56% accuracies, respectively. Meanwhile, the proposed ADT-Det detector achieved the best detection performance, with an accuracy of 89.75%. This accuracy is close to the accuracy for ship detection in the DOTA experiment (88.94%), which further proves the advantage of using DFR to reduce the mismatch between arbitrarily oriented objects and axis-aligned receptive fields. Evaluated under the PASCAL VOC2012 metrics, the anchor-free method DRN achieved a 92.7% accuracy, while the proposed ADT-Det detector (with ResNet-152) achieved the best detection result, with an accuracy of 93.47%.

4.4.3. Speed Comparison

Comparison experiments for detection speed and accuracy were carried out on HRSC2016. In the experiment, our ADT-Det detector was compared with eight other well-known methods. The detailed results are illustrated in Table 4 and the overall comparison results are also visualized in Figure 10. It can be seen that the multi-stage detector RoI-Transformer could achieve an 86.2% accuracy and a 6 fps speed when using ResNet101 as the backbone and when the input image size was 512 × 800. The single-stage R^3Det detector could achieve a 89.26% accuracy and a 10 fps speed. The existing state-of-art single-stage R^4Det could achieve an 89.5% accuracy, but the detection speed was slower than that of R^3Det. Our ADT-Det detector could achieve an 89.75% accuracy when evaluated under the PASCAL VOC2007 metrics and a 12 fps speed when the input image size was 800 × 800. Furthermore, we could achieve a 14.6 fps speed when the input image size was 600 × 600. The results demonstrate that our ADT-Det detector can achieve the highest accuracy of all the investigated detectors while running very fast.

Table 4. Evaluation results with the accuracy and speed of some well-known detectors on HRSC2016. All models were evaluated under ResNet-152. * indicates that the result was evaluated under the PASCAL VOC2012 metrics.

Methods	**RC1&RC2** [39]	**RRPN** [13]	**RRD** [40]	**RoI-Trans.** [2]	**DRN** [32]	**CenterMap-Net** [41]	**R^3Det** [4]	**R^4Det** [3]	**ADT-Det**	
Input size	300 × 300	800 × 800	384 × 384	512 × 800	768 × 768	768 × 768	800 × 800	800 × 800	600 × 600	80 0× 800
AP	75.7	79.08	84.3	86.20	92.7 *	92.8 *	89.26	89.56	88.96	89.75/93.47 *
Speed	Slow(<1 fps)	3.5fps	Slow(<1 fps)	6 fps	-	-	10 fps	6.5 fps	14.6 fps	12 fps

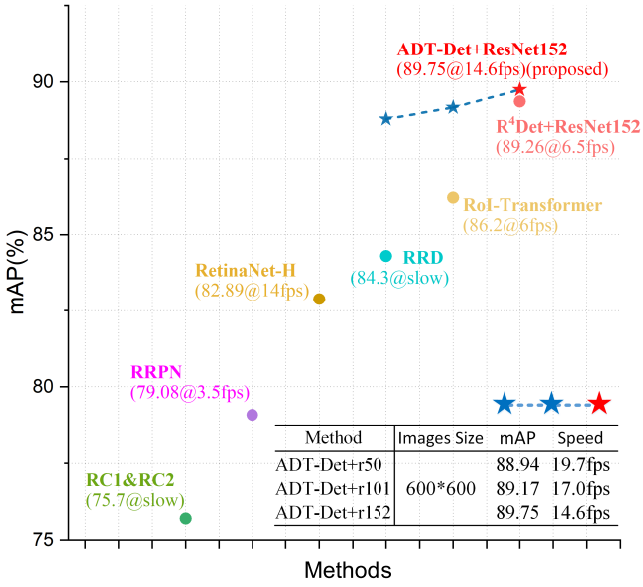


Figure 10. Detection performance (mAP) and speed comparison of our ADT-Det detector and 5 other famous detectors on HRSC2016. Our ADT-Det detector achieved the highest accuracy of all the investigated detectors while running very fast. Detailed results are listed in Table 4.

5. Conclusions

In this work, we identify inappropriate feature extraction as the primary obstacle preventing the high-performance detection of instances with arbitrary directions, large aspect ratios, and dense distributions. To address this, we proposed the use of an adaptive dynamic refined single-stage transformer detector to address the aforementioned challenges, aiming to achieve a high recall and speed. Our detector realizes rotated object detection with RetinaNet as the baseline to achieve the detection of multi-scale objects and densely distributed objects. Firstly, the feature pyramid transformer (FPT) was introduced into the traditional feature pyramid network (FPN) to enhance feature extraction through a feature interaction mechanism. Secondly, the output features of FPT were fed into two post-processing steps, considering the mismatch between the rotated bounding box and the general axis-aligned receptive fields of CNN. Dynamic Feature Refinement (DFR) was introduced in the refinement step. The key idea of DFR was to adaptively adjust the feature map and reconstruct a new feature map for arbitrary-oriented object detection to alleviate the mismatches between the rotated bounding box and the axis-aligned receptive fields. Extensive experiments and ablation studies were carried out to test the proposed detector based on two challenging satellite optical imagery public datasets, DOTA and HRSC2016. The proposed detector could achieve a 79.95% mAP accuracy for DOTA and 93.47% mAP for HRSC2016, and the running speed was 14.6 fps with an 600 $\times$ 600 input image size. The results show that our method achieved state-of-the-art results in the task of object detection in these optical imagery datasets.

Author Contributions: The first two authors have equally contributed to the work. Conceptualization, Y.Z.; methodology, Y.Z., P.S. and Z.Z.; software, P.S.; validation, W.X., Q.Z.; formal analysis, Y.Z., P.S. and Z.Z.; investigation, Y.Z., P.S. and W.X.; resources, Y.Z. and Z.Z.; writing—original draft preparation, Y.Z. and P.S.; writing—review and editing, Z.Z., W.X. and Q.R.; visualization, P.S. and Q.R.; supervision, Y.Z. and Z.Z.; project administration, Y.Z. and Z.Z.; funding acquisition, Z.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 61403412.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to their large size.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Deep learning for generic object detection: A survey. *Int. J. Comput. Vis.* **2020**, *128*, 261–318. [CrossRef]
2. Ding, J.; Xue, N.; Long, Y.; Xia, G.S.; Lu, Q. Learning roi transformer for oriented object detection in aerial images. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 2849–2858.
3. Sun, P.; Zheng, Y.; Zhou, Z.; Xu, W.; Ren, Q. R4 Det: Refined single-stage detector with feature recursion and refinement for rotating object detection in aerial images. *Image Vis. Comput.* **2020**, *103*, 104036. [CrossRef]
4. Yang, X.; Liu, Q.; Yan, J.; Li, A.; Zhang, Z.; Yu, G. R3det: Refined single-stage detector with feature refinement for rotating object. *arXiv* **2019**, arXiv:1908.05612.
5. Xia, G.S.; Bai, X.; Ding, J.; Zhu, Z.; Belongie, S.; Luo, J.; Datcu, M.; Pelillo, M.; Zhang, L. DOTA: A large-scale dataset for object detection in aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 3974–3983.
6. Cheng, G.; Zhou, P.; Han, J. Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 7405–7415. [CrossRef]
7. Liu, Z.; Wang, H.; Weng, L.; Yang, Y. Ship rotated bounding box space for ship extraction from high-resolution optical satellite images with complex backgrounds. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 1074–1078. [CrossRef]
8. Zhang, G.; Lu, S.; Zhang, W. CAD-Net: A context-aware detection network for objects in remote sensing imagery. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 10015–10024. [CrossRef]
9. Hou, J.B.; Zhu, X.; Yin, X.C. Self-Adaptive Aspect Ratio Anchor for Oriented Object Detection in Remote Sensing Images. *Remote Sens.* **2021**, *13*, 1318. [CrossRef]
10. Xu, Y.; Fu, M.; Wang, Q.; Wang, Y.; Chen, K.; Xia, G.S.; Bai, X. Gliding vertex on the horizontal bounding box for multi-oriented object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 1452–1459. [CrossRef] [PubMed]
11. Jiang, Y.; Zhu, X.; Wang, X.; Yang, S.; Li, W.; Wang, H.; Fu, P.; Luo, Z. R2CNN: Rotational Region CNN for Orientation Robust Scene Text Detection. *arXiv* **2017**, arXiv:1706.09579.
12. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. *arXiv* **2016**, arXiv:1605.06409.
13. Ma, J.; Shao, W.; Ye, H.; Wang, L.; Wang, H.; Zheng, Y.; Xue, X. Arbitrary-oriented scene text detection via rotation proposals. *IEEE Trans. Multimed.* **2018**, *20*, 3111–3122. [CrossRef]
14. Li, Y.; Huang, Q.; Pei, X.; Jiao, L.; Shang, R. RADet: Refine Feature Pyramid Network and Multi-Layer Attention Network for Arbitrary-Oriented Object Detection of Remote Sensing Images. *Remote Sens.* **2020**, *12*, 389. [CrossRef]
15. Yang, X.; Yang, J.; Yan, J.; Zhang, Y.; Zhang, T.; Guo, Z.; Sun, X.; Fu, K. Scrdet: Towards more robust detection for small, cluttered and rotated objects. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Long Beach, CA, USA, 16–20 June 2019; pp. 8232–8241.
16. Lee, J.; Kim, D.; Ponce, J.; Ham, B. Sfnet: Learning object-aware semantic correspondence. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 2278–2287.
17. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef] [PubMed]
18. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26–30 June 2016; pp. 779–788.
19. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,Venice, Italy, 22–29 October 2017; pp. 7263–7271.
20. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
21. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 21–37.
22. Neubeck, A.; Van Gool, L. Efficient non-maximum suppression. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 20–24 August 2006; Volume 3, pp. 850–855.
23. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
24. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Venice, Italy, 22–29 October 2017; pp. 2117–2125.
25. Zhang, D.; Zhang, H.; Tang, J.; Wang, M.; Hua, X.; Sun, Q. Feature pyramid transformer. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 323–339.
26. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
27. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September, 2018; pp. 801–818.

28. Yang, M.; Yu, K.; Zhang, C.; Li, Z.; Yang, K. Denseaspp for semantic segmentation in street scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 3684–3692.
29. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 764–773.
30. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 6154–6162.
31. Zhang, S.; Wen, L.; Bian, X.; Lei, Z.; Li, S.Z. Single-shot refinement neural network for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 4203–4212.
32. Pan, X.; Ren, Y.; Sheng, K.; Dong, W.; Yuan, H.; Guo, X.; Ma, C.; Xu, C. Dynamic refinement network for oriented and densely packed object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual Conference, 16–18 June 2020; pp. 11207–11216.
33. Liu, Z.; Yuan, L.; Weng, L.; Yang, Y. A high resolution optical satellite image dataset for ship recognition and some new baselines. In Proceedings of the International Conference on Pattern Recognition Applications and Methods, SCITEPRESS, Porto, Portugal, 24–26 February 2017; Volume 2, pp. 324–331.
34. Bottou, L. Stochastic gradient descent tricks. In *Neural Networks: Tricks of the Trade*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 421–436.
35. Chen, K.; Wang, J.; Pang, J.; Cao, Y.; Xiong, Y.; Li, X.; Sun, S.; Feng, W.; Liu, Z.; Xu, J.; et al. MMDetection: Open MMLab Detection Toolbox and Benchmark. *arXiv* **2019**, arXiv:1906.07155.
36. Zhu, Y.; Du, J.; Wu, X. Adaptive period embedding for representing oriented objects in aerial images. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 7247–7257. [CrossRef]
37. Lin, Y.; Feng, P.; Guan, J. IENet: Interacting embranchment one stage anchor free detector for orientation aerial object detection. *arXiv* **2019**, arXiv:1912.00969.
38. Qin, X.; Wang, Z.; Bai, Y.; Xie, X.; Jia, H. FFA-Net: Feature fusion attention network for single image dehazing. In Proceedings of the AAAI Conference on Artificial Intelligence, Hilton, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11908–11915.
39. Liu, L.; Pan, Z.; Lei, B. Learning a rotation invariant detector with rotatable bounding box. *arXiv* **2017**, arXiv:1711.09405.
40. Liao, M.; Zhu, Z.; Shi, B.; Xia, G.s.; Bai, X. Rotation-sensitive regression for oriented scene text detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 5909–5918.
41. Wang, J.; Yang, W.; Li, H.C.; Zhang, H.; Xia, G.S. Learning Center Probability Map for Detecting Objects in Aerial Images. *IEEE Trans. Geosci. Remote Sens.* **2020**, *5*, 4307–4323. [CrossRef]