symmetry

MDPI

Article

Novel Joint Object Detection Algorithm Using Cascading Parallel Detectors

Zihan Zhou †, Qinghan Lai †, Shuai Ding and Song Liu *

School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, China; 1043118416@stu.qlu.edu.cn (Z.Z.); 1043119228@stu.qlu.edu.cn (Q.L.); 1043119701@stu.qlu.edu.cn (S.D.)

* Correspondence: liusong@qlu.edu.cn

† These authors contributed equally to this work.

Abstract: Object detection is an essential computer vision task that aims to detect target objects from an image. The traditional models are insufficient to generate a high-quality anchor box. To solve the problem, we propose a novel joint model called guided anchoring Region proposal networks and Cascading Grid Region Convolutional Neural Networks (RCGrid R-CNN), enhancing the ability of object detection. Our proposed model design is a joint object detection algorithm containing an anchor-based and an anchor-free branch in parallel and symmetry. In the anchor-based, we use nine-point spatial information fusion to obtain better anchor box location and introduce the shape prediction method of Guided Anchoring Region Proposal Networks (GA-RPN) to enhance the accuracy of the predicted anchor box. In the anchor-free branch, we introduce the Feature Selective Anchor-Free module (FSAF) to reduce the overlapping anchor boxes to obtain a more accurate anchor box. Furthermore, inspired by cascading theory, we cascade the new-designed detectors to improve the ability of object detection by setting a gradually increasing Intersection over Union (IoU) threshold. Compared with typical baseline models, we comprehensively evaluated our model by conducting experiments on two open datasets: Pascal VOC2007 and COCO2017. The experimental results demonstrate the effectiveness of RCGrid R-CNN in producing a high-quality anchor box.

Keywords: object detection; anchor box; grid R-CNN; shape prediction; cascading detectors

Citation: Zhou, Z.; Lai, Q.; Ding, S.; Liu, S. Novel Joint Object Detection Algorithm Using Cascading Parallel Detectors. *Symmetry* **2021**, *13*, 137. https://doi.org/10.3390/sym13010137

Received: 18 December 2020
Accepted: 14 January 2021
Published: 15 January 2021

1. Introduction

Object detection [1] is an essential mission in the field of artificial intelligence computer vision. Its main tasks are object location and classification. In recent years, with the development and application of deep neural networks [2], object detection has been further developed in many fields, such as image recognition [3], automatic driving [4], and target tracking [5].

Object detection methods can be split into two types: anchor-based object detection [6] and anchor-free object detection [7]. The anchor-based method presets numerous anchor points and further refine these anchor points for prediction in the image. The accuracy of the anchor-based method is improved through the extraction of the region proposal [8]. However, those numerous anchor boxes generated by an anchor-based method can easily cause the overlapping problem, which influences the accuracy of the predicted anchor box. The anchor-free method uses the center point or the center region of the object to determine the location without anchor boxes. Therefore, the anchor-free method has a higher speed and a lower accuracy than the anchor-based method. However, the anchor-free method can be useful in assisting the anchor-based method to reduce the useless anchor boxes.

Faster Region Convolutional Neural Networks (Faster R-CNN) [9] is a classic object detection method with the deep neural network. Faster R-CNN creatively comes up with using the Region Proposal Network (RPN) [10] to replace the region proposal generated by the original selective search [11] method, which reduces the amount and improves

the quality of the anchor boxes. Thus, the end-to-end training [12] is realized, and object detection performance is much improved. However, Faster R-CNN and its extended work have limitations for detecting relatively small objects, which results in its anchor box not being accurate enough.

Later, Xin Lu proposed the Grid Region-Convolutional Neural Networks (Grid R-CNN) [13] method, which can guide the location of anchor boxes and generate appropriate anchor boxes according to the spatial information of objects. Grid R-CNN uses a heatmap produced by the convolutional layer to determine the initial grid points and fuses these grid point features [14] to generate the shape and location of the anchor box. Thus, Grid R-CNN leads to high quality object localization, but it lacks preciseness in anchor shape because its predicted grid points are often smaller than that of the ground truth and can not cover the ground truth exactly.

Moreover, most object detection models use a single IoU threshold [15] detector, which makes the detector easily produce noisy bounding boxes and decrease the accuracy of the predicted anchor boxes.

In the paper, to predict a more accurate anchor box, we bring forward a more efficient and accurate object detection algorithm called RCGrid R-CNN. To solve the problem of determining the anchor box shape, we use the shape prediction method in GA-RPN [8] to improve the anchor prediction of Grid R-CNN to obtain a more proper anchor box shape, which is the anchor-based branch of our algorithm. Next, to solve the problem of overlapping anchor selection, we adopt the FSAF [16] anchor-free branch to perform non-maximum suppression [17] parallel to the anchor-based branch to select a more appropriate anchor box. Afterward, referring to the idea of the cascading detectors, we cascade multiple detector modules and obtain a more accurate detection effect using the continuously increasing IoU threshold.

Especially, we combine the anchor-based and anchor-free branches with symmetric structure. Compared with a single branch, the symmetry is applied to integrate information extracted from two branches. Furthermore, the parallel anchor-based branch and anchor-free branch run in symmetry to select the best feature and anchor box.

Among the current object detection algorithms, the Grid RCNN performs well in predicting anchor box location, but its shape prediction needs to be improved. The GA-RPN is useful in predicting anchor box shape but insufficient in anchor box position prediction. Moreover, they can easily generate overlapping anchor boxes, which can be reduced by the anchor-free algorithm FSAF. Therefore, we design this joint integration algorithm, which uses the Grid R-CNN location prediction module and the GA-RPN shape prediction module to locate the anchor box and predict the anchor box shape. Finally, we solve the selection problem of overlapping anchor boxes by the parallel anchor-free FSAF branch. Furthermore, we consider introducing the cascading detector modules designed in Section 3.1 and continuously improving the IoU threshold of training to enhance the ability of object detection.

During the experiments, we verified our algorithm on two public datasets (COCO2017 [18] and Pascal VOC2007 [19]) in contrast to typical baseline models, for instance, Faster R-CNN, Grid R-CNN and Cascade Region Convolutional Neural Networks (Cascade R-CNN). The detection results of our proposed algorithm were evaluated with the standard COCO and VOC metrics, which are Average Precision (AP) and different AP values with various IoU thresholds or region size.

The main contributions of the work are as follows:

- We propose a new object detection algorithm called RCGrid R-CNN, which comprises cascading detector modules with parallel anchor-based branch and anchor-free branch. Thus, it improves the prediction of the anchor box and object classification efficiently.
- We design a new model with the anchor-based and anchor-free branches in parallel, which improves the accuracy of the anchor box and objects classification of Grid R-CNN comprehensively. In the anchor-based branch, GA-RPN is employed to obtain a more accurate anchor shape. In the anchor-free branch, the FSAF branch in parallel

with the anchor-based branch is added to obtain a more precise anchor box prediction. Finally, we combine the object features extracted from two branches to improve the ability of object classification.

- Using the above new model with two branches as the detector module, we cascade multiple detector modules in regression and classification analysis and improve the anchor box and classification effect using a gradually increasing IoU threshold.

2. Related Work

Recently, new approaches of object detection based on deep neural networks have outperformed the classical methods in many areas. Object detection can be split into the anchor-based method and anchor-free method. The anchor-free method, taking CenterNet [20] as an example, predicts the location of objects directly, which enhances the speed of object detection and reduces the ineffective anchor boxes, but decreases accuracy of detection. The anchor-based methods can be further divided into the one-stage method [21] and the two-stage method [22]. Single Shot MultiBox Detector (SSD) [23] is a classic one-stage method, which integrates the anchor idea of Faster R-CNN into its network to meet the multi-scale object detection task. Although the one-stage method runs faster, it is not as precise as the two-stage method. Therefore, the two-stage method is used more widely at present. Two-stage methods include Fast R-CNN [24], Faster R-CNN, Grid R-CNN, GA-RPN, and so on. The two-stage anchor method can achieve higher performance, so this paper employs the two-stage method.

Influencing factors related to the accuracy of anchor box prediction include object localization, anchor box shape prediction and the selection of overlapping anchor boxes, and so on.

Grid R-CNN has an advanced region proposal extraction network and high quality object localization, it outperforms many traditional two-stage methods in the accuracy of anchor box prediction. However, it can not solve its problems of insufficient anchor box shape prediction and anchor boxes overlapping. Therefore, we introduce the shape prediction module of GA-RPN and FSAF to improve the Grid R-CNN in the region proposal extraction.

GA-RPN is a guided anchor region recommendation network comprising location prediction, shape prediction, and feature adaptation. It can predict the location and shape of the anchor box from the feature map simultaneously. First, the GA-RPN ascertains the possible location of the object according to the probability map generated by the location prediction. Then, it determines the shape by the IoU threshold. Furthermore, it determines the most probable shape at the possible locations in combination with the location prediction. Finally, it uses the feature adaptation module to capture different shapes of the anchor boxes. The GA-RPN adopts a learnable anchoring regional proposal network. Unlike the traditional anchor determination method, the anchor box size of GA-RPN can be changed and not fixed, which enables the GA-RPN network to handle some extensive objects in an image.

Simultaneously, to solve the selection of overlapping anchor boxes, we employ an FSAF module to work jointly with the anchor-based branch by outputting better prediction of anchor boxes in parallel.

Most two-stage methods use a single IoU threshold detector in the detector module, but the IoU is different in various proposals. A single IoU threshold detector can easily lead to noisy (low-quality) detection. Therefore, this paper uses the cascading detection method of Cascade R-CNN [25] for reference and obtains better detection results by continuously increasing the IoU threshold.

In brief, we propose a novel joint object detection algorithm called RCGrid R-CNN, which composes of cascading detector modules with a parallel anchor-based branch and anchor-free branch. In the detector module, combing the high quality object localization of Grid R-CNN, we introduce the GA-RPN to predict anchor shape more accurately in the anchor-based branch, and employ FSAF to solve the selection problem of overlapping an-

chor boxes in the anchor-free branch. Next, we use the cascading detector modules with the continuously increasing IoU threshold, which further improves object detection accuracy.

3. Model

The architecture of RCGrid R-CNN is shown in Figure 1. The RCGrid R-CNN algorithm proposed in this paper composes of cascading detector modules with a parallel anchor-based branch and anchor-free branch:

- In the anchor-based branch, combining the location prediction of Grid R-CNN, the shape prediction of GA-RPN is introduced to improve the prediction of the anchor box shape. Thus we can get more accurate anchor boxes;
- Simultaneously, the FSAF branch (anchor-free branch) parallel with the anchor-based branch is employed to select more appropriate anchor boxes and object features;
- The detector modules are cascaded to address anchor boxes and image features, achieving a more accurate detection effect by gradually increasing the IoU threshold in training.

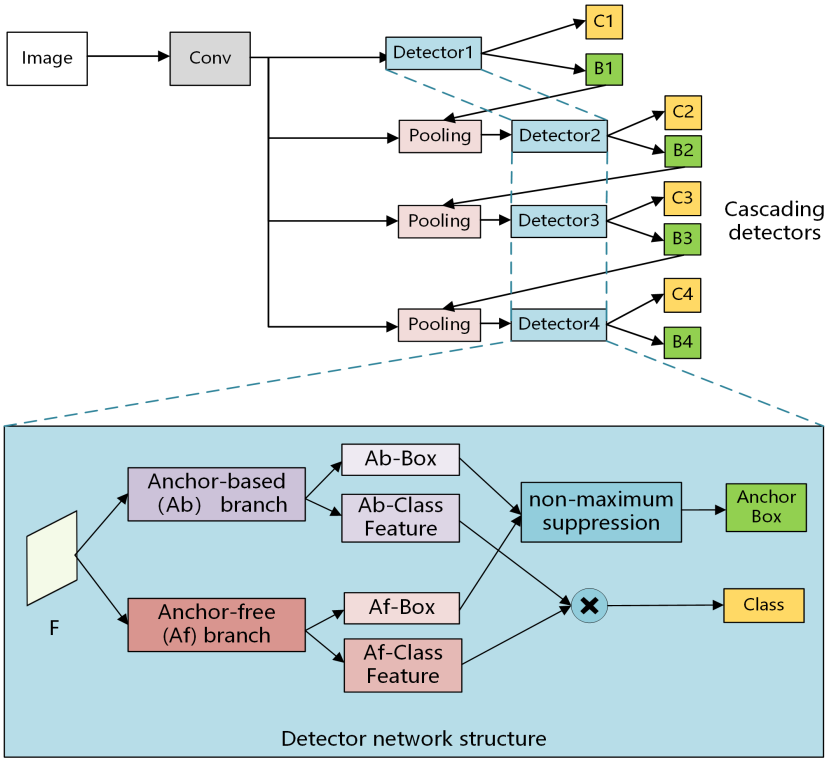

Figure 1. Region proposal networks and Cascading Grid Region Convolutional Neural Networks (RCGrid R-CNN) architecture.

3.1. Anchor Box Prediction And Selection

As mentioned above, to construct the detector module, we design a new anchor-based branch using GA-RPN to achieve better shape prediction. Meanwhile, we design an anchor-free branch using FSAF to obtain appropriate selection from overlapping anchor boxes. The two branches extract image features and predict anchor boxes in parallel. To obtain a better anchor box, we employ the non-maximum suppression to select the anchor box from two branches together.

The region proposal extracts all possible location regions of potential target objects from the input image. Then among the extracted region proposals, the anchor box is marked as accurately as possible, which is the bounding box determined by the pixel coordinates combined with width and height with different aspect ratios and sizes around each pixel.

As displayed in Figure 2, first, we use the grid guided location method in Grid R-CNN to determine the anchor box location. Next, we employ the shape prediction method of GA-RPN to replace the anchor box prediction method of Grid R-CNN to obtain a more accurate anchor shape.

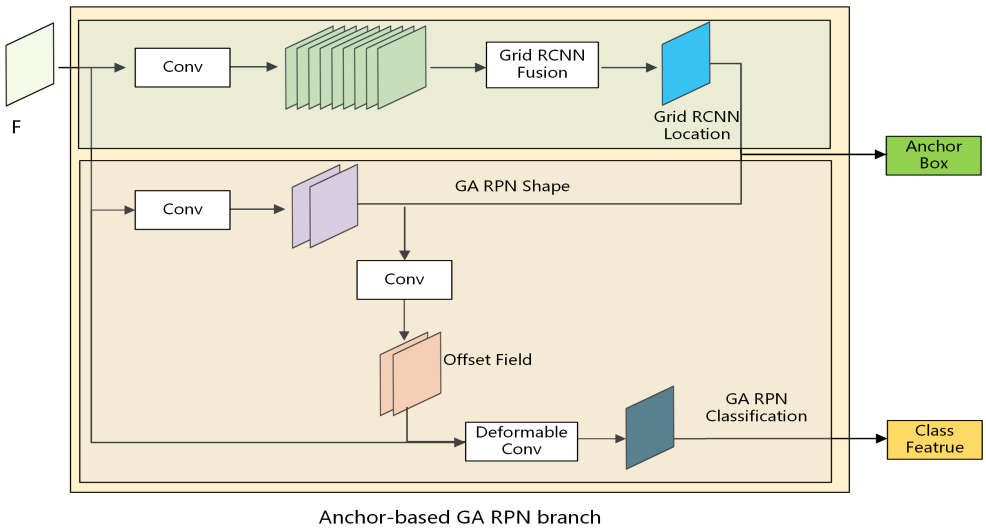

Figure 2. Anchor location and shape prediction.

Specifically, we use the grid guided location module of Grid R-CNN to determine the location of the anchor box in our model. Grid R-CNN uses the full convolutional network to predict the locations of the predefined grid points and selects the point with the highest probability value among the generated grid points as the predicted anchor's location. Here, the feature ROIAlign is extracted from each image, and the extracted feature pixels are convoluted and deconvoluted. Then, the image heatmaps are output. Next, the probability of each heatmap is obtained through the *sigmoid* function. Afterward, the binary cross-entropy loss [26] is used for optimization, and the heatmap with the highest pixel credibility is selected. Finally, the corresponding coordinates on the original image are calculated. The coordinate calculation formula is as follows:

$$\left(I_x = P_x + \frac{H_x}{w_0} w_p, I_y = P_y + \frac{H_y}{h_0} h_p\right) \tag{1}$$

where (H_x, H_y) are the pixel coordinates on the heatmap, (I_x, I_y) are the coordinates on the original image, (P_x, P_y) are the coordinates of the upper left corner of the input image, w_p is the width of the anchor box, h_p is the height, and w_0 is the width, h_0 is the height of the output in the heatmap. Thus the coordinates of the anchor are determined.

Then, we use the method of location prediction proposed by Grid R-CNN to obtain anchor box location. The feature fusion is carried out for the nine points obtained, and the features of the nine points are fused into a location point through 2-hop feature fusion, which is employed to determine the anchor box location. The structure of feature fusion is illustrated in Figure 3 and calculated as follows:

$$F'_{loc} = F_{loc} + \sum_{z \in Z_{loc}} Conv_{z \to loc}(F_z) \tag{2}$$

where loc are location points (Point 5 in Figure 3), F_{loc} are the features of location points, z are loc neighbor points, $Conv_{z \to loc}$ are the fusion convolution of extraction features from neighbor points to location points.

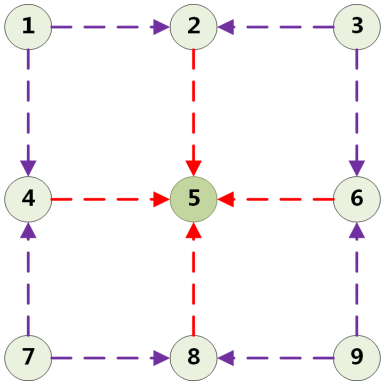

Figure 3. The structure of feature fusion.

Because the anchor determination method of Grid R-CNN is insufficient to locate the boundary of the target object accurately, we propose to use shape prediction to determine the shape of the anchor boxes. Shape prediction is not a predefined fixed-size anchor box, but a dynamic variable one, that is, a dynamic anchor box exists at a particular location. The anchor boxes with dynamic size can better predict extraordinarily high objects. After determining the possible location of the target object through the grid guided location module of Grid R-CNN, we use the shape prediction method of GA-RPN to determine the shape of each location. The shape prediction method of GA-RPN does not change the location of the anchor boxes and can ensure the accuracy of the anchor box location to the greatest extent, which is different from the traditional boundary box regression. Shape prediction is to predict the width w and height h of the anchor boxes. Because the value range of w and h is relatively broad, it is difficult to directly predict the two values. Thus, the conversion of w and h is carried out, which is calculated as follows:

$$w = \sigma \cdot s \cdot e^{dw} \tag{3}$$

$$h = \sigma \cdot s \cdot e^{dh} \tag{4}$$

where s is the step size, σ is the empirical scaling factor, and dw and dh are the mappings of (w, h). Through Formulas (3) and (4), the value $[0, 1000]$ can be converted to $[-1, 1]$, which reduces the value range and makes the learning goal easier.

The shape prediction of GA-RPN of the anchor box uses a sub-network N_s , which is a double channel 1×1 convolutional layer containing dw and dh values, and an element conversion layer using Formulas (3) and (4). The width and height of the anchor boxes are predicted indirectly by predicting dw and dh, and the shape prediction of the anchor box is determined by combining the previously predicted location of the anchor box. Moreover, we use a deformable convolutional network to extract object features and finally output a class feature vector corresponding to each object.

After shape prediction, numerous overlapping anchor boxes are selected. To solve the selection problem of overlapping anchor boxes, we use FSAF anchor-free branch parallel to the above anchor-based branch (Grid R-CNN and GA-RPN) to perform non-maximum suppression to choose better anchor boxes from the overlapping anchor boxes. The two branches work together to select the best feature and anchor box automatically.

The FSAF branch is parallel with the anchor-based branch. It adds two extra anchor-free convolution layers, which are responsible for branch classification and regression prediction, respectively. As illustrated in Figure 4, in this way, the anchor-based branch and anchor-free branch can work together.

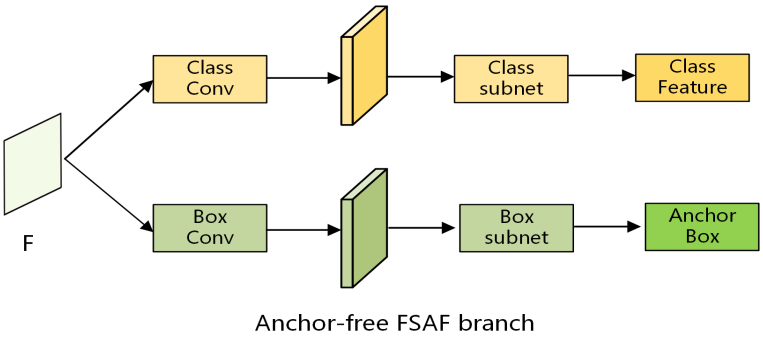

Figure 4. The structure of Feature Selective Anchor-Free (FSAF) anchor-free branch.

3.2. Cascading Detectors

Grid R-CNN uses a single IoU threshold detector in the detector module, and the ToU threshold value is not optimal in object detection. Therefore, we consider introducing the cascading detector modules designed in Section 3.1 and continuously improving the IoU threshold of training to enhance the ability of object detection.

First, we obtain image features from an image using CNN. Next, we input these image features into cascading detectors to predict an anchor box with increasing IoU threshold. Especially, our proposed model follows the method in paper [25] to cascade four stages detector to achieve the best performance. As shown at the bottom of Figure 1, the four stages of the model contain the same symmetric structure with parallel anchor-based and anchor-free branches.

To obtain the predicted anchor box from each stage, we use non-maximum suppression to select the best prediction anchor box from the two branches. For object classification, we use the dot product of the class feature vector extracted by two branches to get the final class feature vector. Using a $softmax$ classifier, we can obtain a predicted object category from each stage. Furthermore, the features in the detection boxes generated in the previous stage are conducted by the pooling layer to obtain input features of the detector module in the next stage. Each cascading detector module is trained to be as close to the IoU threshold of the anchor box as possible to achieve a better detection effect.

As displayed in Figure 5, the $Detector1$ in the first stage is used to generate the preliminary prediction. In the second stage, the prediction result of the $Bbox1$ produced by the $Detector1$ is inputted into a pooling layer to further extract the features. Then, the $Detector2$ processes the pooling features of the image and the anchor box of the previous stage to generate the new anchor box and object classification result. In the third and the fourth stages, similar to the processing of the second stage, the $Detector3$ and $Detector4$ are used, respectively. The primary purpose of cascading detectors is to enhance the effect of the positive IoU threshold training for the next stage by adjusting the bounding boxes. With the IoU threshold increase, the detector can gradually train the prediction anchor box to approach the real anchor box. Thus, the performance of the cascading detectors is improved. Finally, we select the category and anchor box with the minimization loss of classification and anchor box prediction among the four stages as the category and anchor box of object detection. The loss formula of the detector model in each stage is as follows:

$$L(x^t, g) = L_{cls}(h_t(x^t), y^t) + \lambda[y^t \geq 1]L_{loc}(f_t(x^t, b^t), g) \tag{5}$$

where $b^t = f_{t-1}(x^{t-1}, b^{t-1})$ is the anchor box with t stage. x^t is the final classification vector obtained by the dot product of the anchor-based branch and anchor-free branch classification feature vector. g is a basic truth object of x^t, λ is the weighing coefficient, $[\cdot]$ is the indicator function, and y^t is the label of x^t.

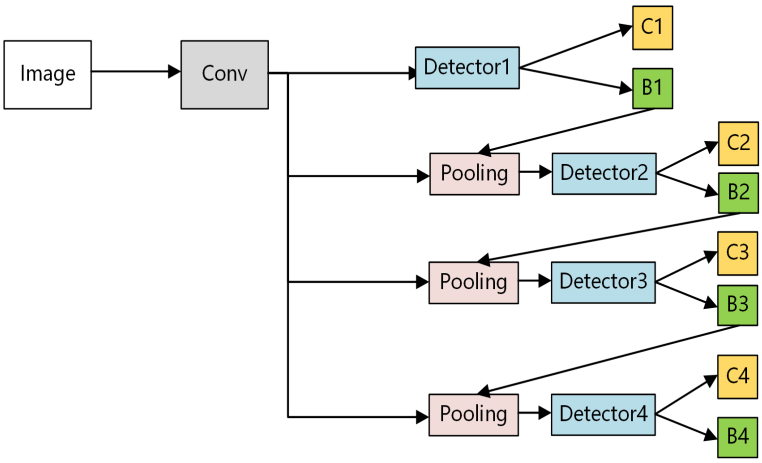

Figure 5. The architecture of the cascading detector modules.

3.3. Implementation Details

We build the proposed model using deep learning framework PyTorch and object detection framework MMDetection [27]. Our model is trained on 8 Nvidia 2080Ti GPUs with CUDA 10.2 and CUDNN 7.6, using one image per GPU. As a general approach, we use the fixed image and resize images to the scale of 1333 × 800. Furthermore, the ResNet-50 [28] and ResNet-101 [28] are selected as backbone network of our model.

As for the detector module, in the anchor-based branch, we use the grid guided location module of Grid R-CNN to determine the location of the anchor box. Here, we fuse nine grid points into an anchor box location point through 2-hop feature fusion by eight 5 × 5 convolutional layers, with 64 feature channels corresponding to each grid point. Next, we employ the shape prediction method of GA-RPN to predict the shape of the anchor box and set the input and output channels of RPN as 256 and 256, respectively. To adopt a deformable convolutional network, we employ a 1 × 1 convolutional layer to yield a two-channel map. In the parallel anchor-free branch, the input and output channels of the FSAF head are both 256. Then, we use non-maximum suppression to select the best prediction anchor box from the two branches and adopt the dot product of the class feature vector extracted by two branches to get the final class feature vector with the dimension of 1024. At last, the above same detector modules containing the parallel anchor-based and anchor-free branches are cascaded into four stages, with the gradually IoU threshold of 0.5, 0.6, 0.7, and 0.75.

To optimize our model, we use the smooth L1 loss and the cross entropy loss function of Pytorch to calculate box prediction regression loss and object classification loss, respectively. Furthermore, the SGD is employed to optimize the training loss with 0.9 momentum and 0.0001 weight decay.

4. Experiments

In the experiments, we applied the RCGrid R-CNN algorithm to two object detection datasets: Pascal VOC2007 and COCO2017. The size of COCO2017 is significantly larger than that of VOC2007, so we chose the two datasets of different size for the experiment.

The Pascal VOC2007 dataset contains the label annotations required for detection. The Pascal VOC2007 dataset comprises 20 categories, divided into two parts: *trainval* (training and validation set) and *test* (testing set). The two parts each account for 50% of the total data. *Trainval* is further split into a training set and validation set, each accounting for 50% of *trainval*. An image in the Pascal VOC2007 dataset may contain more than one target object.

The COCO2017 dataset is a classic object detection dataset, which contains 80 object categories. We trained our model on the union of 80 k train images and 35 k subset of validation images and tested it on a 5 k *val* subset (*minival*) and 20k *test-dev*.

In the first two comparisons of RCGrid R-CNN with Grid R-CNN, Faster R-CNN, and Cascade R-CNN on Pascal VOC2007 and COCO2017 datasets, we used the train split for training and obtained the performance of the AP value on the validation set. In the comparison of RCGrid R-CNN with some other advanced baseline models on the COCO2017, object detection results are reported on the *test-dev* split.

4.1. Experimental Procedure

We performed experiments with the RCGrid R-CNN model on two public datasets (COCO2017 and Pascal VOC2007) and contrasted experiment results with typical baseline models. To comprehensively evaluate our model, we adopted the AP value as the metrics to evaluate the results, where AP_S was used as the evaluation index of small region objects, AP_M was used as the evaluation index of medium region objects, and AP_L was used as the evaluation index of large region objects.

Ablation Experiment

Three ablation tests were performed to demonstrate the effect of each part of the overall model improvement in RCGrid R-CNN on COCO2017 *minival*. For the fairness of the investigation, the backbone network used the unified ResNet-50 [28] as the primary network structure.

Grid R-CNN, as the baseline network configuration, started with a learning rate of 0.02. At the 17th and the 23rd epochs, the learning rate was decayed twice, and the decay rate was 0.1 until 25 epoch iterations stopped. The default settings were used unless otherwise specified below.

Comparison of Grid R-CNN and SGrid R-CNN: After the grid points of Grid R-CNN were generated, the shape prediction method of GA-RPN was used to determine the shape of anchor boxes. We called the algorithm Shape Grid Region Convolutional Neural Networks (SGrid R-CNN) and compared it with Grid R-CNN. Moreover, in SGrid R-CNN, we adopted the deformable convolutional layer of GA-RPN to extract the shape information of the object. As listed in Table 1, the results reveal that the AP value increased by 0.6%. The experimental results indicate that the shape prediction makes the shape of the anchor box more accurate. The algorithm SGrid R-CNN made certain progress in ordinary images and can adapt to some exceptional cases.

Table 1. Ablation experiment results.

Method	Backbone	AP
Grid R-CNN [13]	ResNet-50	35.9
SGrid R-CNN	ResNet-50	36.5
FGrid R-CNN	ResNet-50	36.3
CGrid R-CNN	ResNet-50	36.8

Comparison of Grid R-CNN and FGrid R-CNN: In the original grid branch of Grid R-CNN, an adaptive FSAF branch is added parallel with the original anchor-based grid guided location to generate the best anchor box. We called this algorithm FSAF Grid Region Convolutional Neural Networks (FGrid R-CNN) and compared it with Grid R-CNN. Combining the anchor-free branch and the anchor-based branch in symmetry, our detector can predict the anchor boxes and classify the objects more effectively by significantly reducing the overlapping anchor boxes. The results indicate that the AP performance improved by 0.4 %. The experimental results reveal that the addition of the FSAF branch can locate anchor boxes closer to the ground truth. Additionally, a certain number of anchor boxes were reduced.

Comparison of Grid R-CNN and CGrid R-CNN: The detectors of Grid R-CNN were cascaded in four stages, and the increasing IoU threshold was used to improve the detection quality. We called this algorithm Cascade Grid Region Convolutional Neural Networks

(CGrid R-CNN) and compared it with Grid R-CNN. As listed in Table 1, the results indicate that the AP performance improved by 0.9 %. The results reveal that the cascading detectors have an outstanding effect on object detection and improve the detection accuracy due to the benefits of four stages of detector modules with the increasing IoU threshold from 0.5 to 0.75.

4.2. Results of Contrast Experiment with Baseline Models and Analysis

In contrast experiments, RCGrid R-CNN was compared with several widely used object detection algorithms as baseline models, with two backbone networks ResNet-50 and ResNet-101. To maintain the consistency of experimental conditions, we used the same Feature Pyramid Networks (FPN) [29] when comparing RCGrid R-CNN with several baseline models. Furthermore, we comprehensively evaluated our model by conducting experiments on the Pascal VOC2007 dataset and COCO2017 dataset.

4.2.1. Pascal VOC2007 *test* Experimental Results

We employed RCGrid R-CNN on the Pascal VOC2007 dataset and compared it with the baseline models. All algorithms started with a learning rate of 0.02. At the 20th and 25th epochs, the learning rate was decayed twice, and the early stop epoch is 28.

Table 2 lists the experimental results of the RCGrid R-CNN w FPN model and the baseline models for the Pascal VOC2007 dataset. To maintain the consistency of experimental conditions, we selected ResNet-50 as the backbone network. Compared with Grid R-CNN w FPN, RCGrid R-CNN w FPN is enhanced by 0.2 % in AP value, which suggests that RCGrid R-CNN w FPN has achieved better performance mainly by its improvements on Grid R-CNN w FPN. Moreover, compared with Cascade R-CNN w FPN, the performance of RCGrid R-CNN w FPN improved by 0.6% benefitting from the detection ability of our newly designed detector module. Overall, the experiments reveal that our algorithm has a particular improvement in object detection accuracy and consistency in the dataset of small size.

Table 2. Comparison of RCGrid R-CNN with the baseline models on the Pascal VOC2007 dataset.

Method	Backbone	AP
Grid R-CNN w FPN [13]	ResNet-50	55.3
Cascade R-CNN w FPN [25]	ResNet-50	54.9
RCGrid R-CNN w FPN	ResNet-50	55.5

4.2.2. COCO2017 Dataset Experimental Results

We further used a more challenging COCO2017 dataset for experiments. In this experiment, the learning rate of RCGrid R-CNN was 0.02. At the 17th and the 23rd epochs the learning rate was decayed twice, and the decay rate was 0.1 until the 25 epoch iterations stopped.

We further used the more challenging COCO2017 dataset for experiments. In this experiment, the learning rate of RCGrid R-CNN was 0.02. The 17th and the 23rd epochs were decayed twice, and the decay rate was one-tenth until the 25 epoch iterations stopped.

Table 3 lists the experimental results of the RCGrid R-CNN w FPN model and the baseline models for the COCO2017 *minival*. Similar to the baseline models, we selected ResNet-101 as the backbone. Compared with the Faster R-CNN w FPN, the performance of RCGrid R-CNN has a significant improvement, which suggests that RCGrid R-CNN is effective in detecting an object. Additionally, the AP value of RCGrid R-CNN w FPN outperforms Grid R-CNN w FPN by 1.9%. This indicates that we can enhance the ability of object detection by improving the shape prediction module and introducing cascade theory. Furthermore, compared with Cascade R-CNN w FPN, the RCGrid R-CNN can achieve a competitive performance.

Table 3. Comparison of RCGrid R-CNN with baseline models on COCO2017 *minival*.

Method	Backbone	AP	$AP_{.5}$	$AP_{.75}$	AP_S	AP_M	AP_L
Faster R-CNN w FPN [9]	ResNet-101	39.5	61.2	43.1	22.7	43.7	50.8
Grid R-CNN w FPN [13]	ResNet-101	41.2	60.3	44.4	23.4	45.8	54.1
Cascade R-CNN W FPN [25]	ResNet-101	42.7	61.6	46.6	23.8	46.2	57.4
RCGrid R-CNN w FPN	ResNet-101	43.1	61.4	46.9	23.9	46.6	58.0

To make a complete comparison, we compared RCGrid R-CNN w FPN with some other advanced detectors on the COCO2017 *test-dev*, with ResNet-101 as the backbone network. The results were shown in Table 4. Compared with the traditional models (SSD-513, DSSD-513, RefineDet-512, Faster R-CNN++, and Faster R-CNN w FPN), RCGrid R-CNN w FPN has improved significantly in overall AP values. Considering the results of the Grid R-CNN w FPN, we can find that the AP_L is enhanced by 2.8%. This suggests that we improve the shape prediction by introducing the GA-RPN and reduce the overlapping anchor boxes by using FSAF anchor-free branch. Moreover, it also shows the effectiveness of the deformable convolutional network in extracting object shape information, especially in large region objects. Furthermore, RCGrid R-CNN w FPN outperforms Cascade R-CNN w FPN in several different AP values (AP, $AP_{.75}$, AP_S, AP_M, and AP_L) using the same cascade structure. This shows that our new-designed detector module that contains an anchor-free branch and an anchor-based branch performs better than the traditional detector module. Furthermore, compared with single stage detectors, our model improved significantly on $AP_{.75}$ and reached the highest performance 46.8% by setting a gradually increasing IoU threshold from 0.5 to 0.75. In short, the experimental results indicate that RCGrid R-CNN w FPN achieves a more competitive performance than other models and gets the best overall performance at an AP of 43.3%. The result examples of RCGrid R-CNN are shown in Figure 6.

Table 4. Comparison of RCGrid R-CNN with some other advanced detectors on the COCO2017 *test-dev*.

Method	Backbone	AP	$AP_{.5}$	$AP_{.75}$	AP_S	AP_M	AP_L
SSD-513 [23]	ResNet-101	31.2	50.4	33.3	10.2	34.5	49.8
DSSD-513 [30]	ResNet-101	33.2	53.3	35.2	13.0	35.4	51.1
RefineDet-512 [31]	ResNet-101	36.4	57.5	39.5	16.6	39.9	51.4
Faster R-CNN++ [9]	ResNet-101	34.9	55.7	37.4	15.6	38.7	50.9
Faster R-CNN w FPN [9]	ResNet-101	36.2	59.1	39.0	18.2	39.0	48.2
Grid R-CNN w FPN [13]	ResNet-101	41.5	60.9	44.5	23.3	44.9	53.1
GA-RPN w FPN [8]	ResNet-101	39.8	59.2	43.5	21.8	40.1	48
Cascade R-CNN w FPN [25]	ResNet-101	42.8	62.1	46.3	23.7	45.5	55.2
RCGrid R-CNN w FPN	ResNet-101	43.3	62.0	46.8	23.9	46.0	55.9

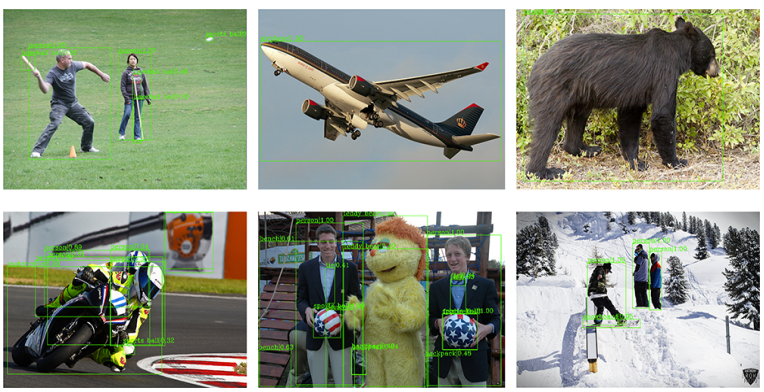

Figure 6. The example results of RCGrid R-CNN on COCO2017 *test-dev*.

5. Conclusions

In this paper, we proposed a novel joint object detection algorithm called RCGrid R-CNN. Combining an anchor-based and an anchor-free branch in parallel, we constructed a symmetric parallel detector module to improve the accuracy of anchor box prediction. Using the improved shape prediction method GA-RPN in the anchor-based branch, we enhanced the quality of anchor box shape prediction. Moreover, the overlapping anchor boxes were reduced by introducing FSAF into the anchor-free branch. Finally, due to introducing the cascading detector modules, we improved the ability of anchor box prediction regression and object classification. The experimental results show that the RCGrid R-CNN has achieved competitive performance on the Pascal VOC2007 and COCO2017 datasets compared with recent state-of-the-art baseline models.

The main contribution of this paper is to improve the correctness of anchor box positioning and shape prediction, and there is no obvious improvement in object classification. In the future we will improve the algorithm in the aspect of object classification. Furthermore, we would like to introduce more backbone networks such as the capsule network [32] to extract the image features more precisely and improve the ability to handle the affine and projective transformation of the images.

Author Contributions: Conceptualization, S.L.; investigation, Z.Z.; methodology, Z.Z.; resources, Q.L.; software, Z.Z. and Q.L.; supervision, S.L.; writing—original draft, Z.Z. and S.D.; writing—review and editing, S.L and Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Science and Technology Development Plan for Higher Education in the Shandong Province under Grant J18KA360 and the Key Research and Development Program in Shandong Province under Grant 2019GGX105010.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are openly available in [COCO2017, VOC2007] at [https://doi.org/10.1007/978-3-319-10602-1_48, https://doi.org/10.1007/s11263-014-0733-5], reference number [18,19].

Acknowledgments: The authors are very thankful to the editor and referees for their valuable comments and suggestions for improving the paper.

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

References

1. Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object detection in 20 years: A survey. *arXiv* **2019**, arXiv:1905.05055.
2. Szegedy, C.; Toshev, A.; Erhan, D. Deep neural networks for object detection. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 2553–2561.
3. Chen, Z.M.; Wei, X.S.; Wang, P.; Guo, Y. Multi-label image recognition with graph convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 5177–5186.
4. Sathyanarayana, A.; Sadjadi, S.O.; Hansen, J.H. Leveraging sensor information from portable devices towards automatic driving maneuver recognition. In Proceedings of the 2012 15th International IEEE Conference on Intelligent Transportation Systems, Anchorage, AK, USA, 16–19 September 2012; pp. 660–665.
5. Jin, L.; Li, S.; La, H.M.; Zhang, X.; Hu, B. Dynamic task allocation in multi-robot coordination for moving target tracking: A distributed approach. *Automatica* **2019**, *100*, 75–81. [CrossRef]
6. Zhang, S.; Chi, C.; Yao, Y.; Lei, Z.; Li, S.Z. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–18 June 2020; pp. 9759–9768.
7. Kong, T.; Sun, F.; Liu, H.; Jiang, Y.; Li, L.; Shi, J. FoveaBox: Beyound Anchor-Based Object Detection. *IEEE Trans. Image Process.* **2020**, *29*, 7389–7398. [CrossRef]
8. Wang, J.; Chen, K.; Yang, S.; Loy, C.C.; Lin, D. Region proposal by guided anchoring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 2965–2974.
9. 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]
10. Li, B.; Yan, J.; Wu, W.; Zhu, Z.; Hu, X. High performance visual tracking with siamese region proposal network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8971–8980.

11. Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
12. Henderson, P.; Ferrari, V. End-to-end training of object class detectors for mean average precision. In *Asian Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 198–213.
13. Lu, X.; Li, B.; Yue, Y.; Li, Q.; Yan, J. Grid r-cnn. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 7363–7372.
14. Sun, Q.S.; Zeng, S.G.; Liu, Y.; Heng, P.A.; Xia, D.S. A new method of feature fusion and its application in image recognition. *Pattern Recognit.* **2005**, *38*, 2437–2448. [CrossRef]
15. Bochinski, E.; Senst, T.; Sikora, T. Extending IOU based multi-object tracking by visual information. In Proceedings of the 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Auckland, New Zealand, 27–30 November 2018; pp. 1–6.
16. Zhu, C.; He, Y.; Savvides, M. Feature selective anchor-free module for single-shot object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 840–849.
17. Hosang, J.; Benenson, R.; Schiele, B. Learning non-maximum suppression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4507–4515.
18. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In European conference on computer vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
19. Everingham, M.; Eslami, S.A.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes challenge: A retrospective. *Int. J. Comput. Vis.* **2015**, *111*, 98–136. [CrossRef]
20. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, South Korea, 27 October–3 November 2019; pp. 6569–6578.
21. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, South Korea, 27 October–3 November 2019; pp. 9627–9636.
22. Li, Z.; Peng, C.; Yu, G.; Zhang, X.; Deng, Y.; Sun, J. Light-Head R-CNN: In Defense of Two-Stage Object Detector. *arXiv* **2017**, arXiv:1711.07264.
23. 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.
24. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
25. Cai, Z.; Vasconcelos, N. Cascade R-CNN: High Quality Object Detection and Instance Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**. [CrossRef] [PubMed]
26. Zhang, Z.; Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. *Adv. Neural Inf. Process. Syst.* 2018, 31, 8778–8788.
27. 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.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
29. 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, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
30. Fu, C.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. DSSD : Deconvolutional Single Shot Detector. *arXiv* **2017**, arXiv:1701.06659.
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. Kosiorek, A.; Sabour, S.; Teh, Y.W.; Hinton, G.E. Stacked capsule autoencoders. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; pp. 15512–15522.