Anchor-free Proposal Generation Network for Efficient Object Detection

Hoanh Nguyen
Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

***Abstract*—Deep learning object detection methods are usually based on anchor-free or anchor-based scheme for extracting object proposals and one-stage or two-stage structure for producing final predictions. As each scheme or structure has its own strength and weakness, combining their strength in a unified framework is an interesting research topic. However, this topic has not attracted much attention in recent years. This paper presents a two-stage object detection method that utilizes an anchor-free scheme for generating object proposals in the initial stage. For proposal generation, this paper employs an efficient anchor-free network for predicting object corners and assigns object proposals based on detected corners. For object prediction, an efficient detection network is designed to enhance both detection accuracy and speed. The detection network includes a lightweight binary classification subnetwork for removing most false positive object candidates and a light-head detection subnetwork for generating final predictions. Experimental results on the MS-COCO dataset demonstrate that the proposed method outperforms both anchor-free and two-stage object detection baselines in terms of detection performance.**

Keywords—Object detection; deep learning; convolutional neural network; proposal generation network

I. Introduction

Object detection has seen significant advancements in recent years thanks to deep learning, particularly convolutional neural networks (CNN). According to the way of generating object proposals from input images, Current object detection techniques can be divided into two categories: anchor-based and anchor-free object detection methods. Anchor-based approaches consider each object as a rectangular bounding box on feature map. Features inside the bounding box are extracted and inputted into either a proposal generation network to generate proposals or a detection network to generate final outputs. To address the issue of scale variation, anchor-based methods define multiple bounding boxes with varying sizes and aspect ratios, enabling the network to detect objects of diverse sizes and proportions. The sizes and aspect ratios defined in anchor-based object detection approaches vary depending on the specific object and structure. Anchor-based schemes are dominant in early deep learning object detection methods since they are easy to implement and facilitate the learning process. However, object detection methods based on anchor-based scheme face another problem as they cannot detect objects with rare sizes/aspect ratios due to the limitation of anchor box sizes and ratios.

On the other hand, anchor-free object detection approaches examine points (i.e., anchor points or keypoints) on feature map to predict objects. These approaches can be categorized into two categories: anchor points object detection approaches and keypoints object detection approaches. While object detection methods based on anchor points classify each point on feature map into object/background classes and predict the distances from the positive points to object borders, keypoint-based object detection methods predict keypoints such as corner points or center points on feature map and group valid points to form objects. Compared to anchor points object detection methods, keypoints object detection methods usually have a more complicated structure and achieve better detection performance. However, they need an optimal grouping algorithm so that the network can efficiently group valid keypoints to form objects.

Alternatively, according to the learning process, Current object detection techniques can be divided into two categories: one-stage and two-stage object detection methods. One-stage object detection techniques directly use the detection network on input feature maps to generate final outputs, whereas two-stage object detection methods generate object proposals in the first stage, followed by the use of the detection network in the second stage to produce the final predictions. Since one-stage methods eliminate proposal generation process, they obtain fast processing speed. However, detection accuracy is typically improved through the use of two-stage methods [1], [2].

In recent years, many object detection structures followed the above schemes or structures have obtained great achievements [3], [4], [5]. In general, each of the above schemes or structures has its own strengths and weaknesses. Thus, combining the strength of these schemes or structures in a unified framework is an interesting topic. However, this topic has received limited attention from the academic community in recent years. This paper presents a novel object detection framework that combines the benefits of both an anchor-free approach and a two-stage structure. The proposed method uses an anchor-free object detection scheme to generate object proposals in the initial stage. To efficiently generate final predictions in the subsequent stage, an efficient detection network is designed. The efficient detection network includes a lightweight binary classification subnet and a light-head detection subnet. Experimental results on the MS-COCO dataset prove the effectiveness of the proposed method.

The structure of the article is presented as follows. Section II introduces recent related works. Section III provides details on the design of the proposed method. Section IV presents the experiments and results achieved by the method. Section V provides conclusions.

II. Literature Review

A. Anchor-based Object Detection Methods

Anchor-based object detection methods depict each object as an anchor bounding box on feature maps. To address the scale variation issue, these object detectors establish multiple anchor boxes at each location on the feature map. Each anchor box is linked to a scale and aspect ratio. In Faster R-CNN [3], three aspect ratios (128^2, 256^2, 512^2) and three scales (1:1, 1:2, 2:1) are employed in the definition of anchor boxes, yielding nine anchor boxes at each position on feature map. In FPN [4], since region proposal network is applied on the feature pyramid, anchor boxes at each spatial location of a feature level are defined using one scale and three aspect ratios. Anchor-based scheme has been employed in many deep object detection frameworks [5], [6], [7], [31]. However, anchor box scales and aspect ratios must be meticulously designed for the specific domain to ensure the detection network attains optimal detection performance. To eliminate problems caused by anchor box settings, various methods propose to replace the manual design of the anchor boxes by a deep network so that the shape of the anchors is automatically learned during training. For this purpose, MetaAnchor [8] developed a generator for anchor functions that maps a given prior bounding box to its corresponding anchor function. The anchor function generator is formed by a simple network and computed from customized prior bounding boxes, and thus it can be inserted into any deep learning object detection methods for joint optimization. The results show that MetaAnchor is more robust than manual design of anchor settings as it can detect objects with rare shapes. However, MetaAnchor obtains minor improvements on two-stage object detectors. Moreover, it requires customized prior bounding boxes to be chosen by handcraft and more computation for extra network. In [9], a novel anchor box optimization method is proposed. The training process employs the optimization technique based on localization loss to automatically learn the anchor shapes. In addition, soft assignment and online clustering scheme are introduced to warm up the anchor shapes. Recently, Sparse R-CNN [10] represented object candidates by a limited set of bounding boxes that can be learned. These learnable bounding boxes represent the statistics of potential object locations within the training set. The back propagation algorithm will update the parameters of these adjustable bounding boxes during training. By eliminating the hand-designed anchor boxes, Sparse R-CNN strikes a favorable balance between accuracy, runtime, and training convergence performance.

B. Anchor-free Object Detection Methods

Anchor-free object detection methods employ points (i.e., anchor points or keypoints) for predicting objects. For this purpose, CornerNet [11] suggests detecting objects based on their top-left and bottom-right corners. For detecting corners, CornerNet employs a corner prediction network that includes a corner pooling layer for producing corner proposals, a heatmap generation layer for generating corner heatmaps, an offset generation layer for predicting corner offsets, and a network for calculating embeddings which are used to group valid corner points to form objects. Based on CornerNet, CenterNet [12] introduced an extra keypoint (i.e., center point) for predicting objects. A center pooling layer is also designed to enrich center and corner information, which improves the detection performance of CenterNet. Zhou et al. [35] utilized a single point at the center of the bounding box to represent objects, eliminating the need for the grouping stage in keypoint detectors like CornerNet and CenterNet. Peaks in the heatmaps generated by a keypoint estimation network are used to predict object center. In [13], representation of objects in input images is achieved through the use of a set of sample points that adaptively position themselves over the object. The sample points are learned through both object localization and recognition loss. Based on predicted sample points, converting functions are designed to form object bounding boxes. In [14], an object prediction mechanism utilizing a star-shaped bounding box is designed. The star-shaped bounding box employs features from nine fixed sampling points with deformable convolution [15] to represent a bounding box. This new bounding box design can incorporate both the geometry of the bounding box and its surrounding context, crucial for encoding any misalignment between the predicted and actual bounding box.

An alternative approach is to use each point on the feature map for object prediction. For this purpose, FCOS [16] employs a fully convolutional network for classifying each location on feature map. For each positive point, FCOS predicts the distances from the location to the four sides of the bounding box. FCOS incorporates a center-ness branch to down-weight the scores of low-quality predicted bounding boxes generated by locations far from the center of an object, in order to remove them. Similar to FCOS, FoveaBox [17] presents an anchor-free framework that predicts category-sensitive semantic maps for the presence of objects and generates category-agnostic bounding boxes for each potential object location. FoveaBox defines positive and negative training samples based on the fovea area, which is the center of the visual field with the highest resolution. Different from FCOS and FoveaBox, Zhu et al. [34] introduced a new feature selective anchor-free module (FSAF), which takes pixels on feature pyramid as inputs and directly feeds these pixels into two convolutional networks: a classification network for predicting class scores for each pixel and a regression network for producing offsets encoding the distances from the current pixel location to the top, left, bottom, and right boundaries of the target bounding box. Recently, SAPD [18] introduced an optimal training approach using two softening optimization techniques, soft-weighted anchor points and soft-selected pyramid levels. The soft-weighted anchor points technique adjusts the contribution of anchor points on the same pyramid level to the network loss based on their geometry relative to the instance box, while the soft-selected pyramid levels technique learns the participation level of each pyramid. The results show that SAPD balances speed and accuracy effectively.

The above methods have their own strengths and weaknesses. Specifically, due to the limitation of anchor box sizes and ratios as well as the variability of object sizes, anchor-based object detection methods have limitations in detecting objects with various shapes. On the other hand, anchor-free object detection approaches have limitations in determining geometric relations between an object and nearby contextual information. This paper focuses on exploiting and combining the strengths of the above methods. Based on that, a model is designed that can achieve better accuracy and execution speed.

III. Methodology

The proposed structure is described in this section. Initially, an evaluation of anchor-based and anchor-free methods for generating object proposals is conducted. Then, the specifics of each module in the proposed structure are depicted in subsequent subsections.

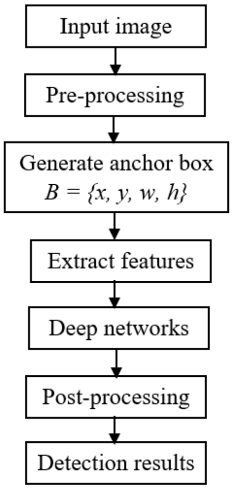

Fig. 1. The flowchart of anchor-based object detection methods.

C. Object Proposal Generation Methods

As shown in Fig. 1, anchor-based object detection methods first depict each object as an anchor box $B = \{x, y, w, h\}$ on feature maps, where the center point is represented by the coordinates (x, y), and the width and height of the object bounding box are (w, h). Then, the features within anchor box B are extracted and inputted into a deep network to generate object proposals (for two-stage approaches) or direct final predictions (for one-stage approaches). Anchor-based object detection techniques define a collection of anchor boxes at each position in a feature map in the input image to accommodate objects of varying size, position, and scale, enabling the model to detect all of them. In Faster R-CNN [3], the center of each anchor box corresponds to the center of the sliding window, and each is paired with a specific scale and aspect ratio. Faster R-CNN uses three aspect ratios (i.e., 1:1, 1:2, 2:1) and three scales (i.e., 128^2, 256^2, 512^2), which results in 9 anchor boxes at each position on feature map. Recently, the FPN [4] has established a single scale and three aspect ratios for each anchor location, utilizing the feature pyramid. When using anchor-based object detection methods, it is important to carefully design the number and shape of anchor boxes. Too few anchor boxes or inappropriate anchor shapes may be insufficient to cover a large range of objects in various sizes and ratios, which reduces the performance of proposal generation network or detection network. The top portion of Fig. 2 shows some examples where Faster R-CNN with anchor-based scheme faces difficulty in predicting objects with rare shapes since there are no defined ratios or scales that fit these objects.

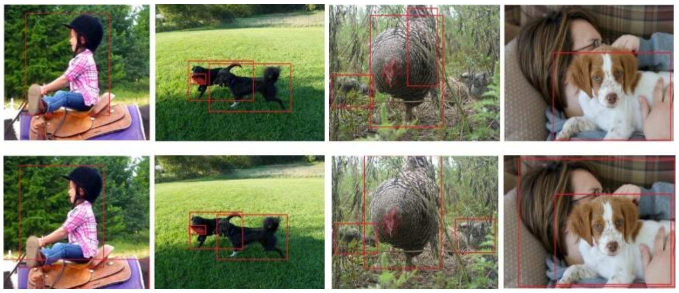

Fig. 2. Results of detection on the validation set of MS-COCO dataset. Top: Faster R-CNN with anchor-based scheme. Bottom: CornerNet with anchor-free scheme.

Alternatively, anchor-free methods use keypoints or anchor-points to depict an object. Methods based on keypoints first predict the locations of keypoints like corner or center points, and then use these keypoints to form an object bounding box by a grouping algorithm. On the contrary, anchor-point based methods first categorize each point on the feature map and then estimate the distances from the potential point to the four edges of the actual bounding box to produce object proposals. Since anchor-free methods eliminate all problems related to anchor box settings, they have a better ability of detecting objects, especially objects with rare shapes, which improves the recall rate. The bottom portion of Fig. 2 shows some detection results of the CornerNet framework [11]. As shown, CornerNet with anchor-free scheme obtains better detection results compared with Faster R-CNN with anchor-based scheme. However, anchor-free methods face another problem of forming an object candidate based on keypoints. Take CornerNet as an example, CornerNet determines an embedding vector for every identified corner, then groups the corners to create the object bounding box based on the distances between the embeddings. Due to the significant number of false positives produced by the corner detection network, determining an embedding vector for each detected corner may result in many false positive outcomes. As seen in Fig. 3, CornerNet generates some incorrect corner pairs because of similar appearance leading to similar embeddings.

Based on the above analysis, this paper introduces an efficient object detection structure that inherits the merits of anchor-free scheme for producing object proposals and two-stage structure for generating predictions. Based on anchor-free scheme, this paper designs an efficient two-stage object detection approach that eliminates the grouping stage, which hinders the detection performance of anchor-free object detection pipelines. The details of the proposed method are outlined in subsequent sections.

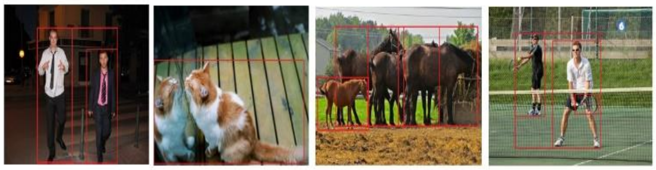

Fig. 3. Results of CornerNet on the validation set of MS-COCO dataset showed some false corner pairs generated due to similarities in embeddings.

D. Overview of the Proposed Approach

The structure of the proposed method is shown in Fig. 4. It integrates an anchor-free approach and a two-stage structure into a single object detection framework. The first stage generates object proposals, and the second stage produces predictions. The proposal generation network in the first stage is based on CornerNet [11]. Specifically, CornerNet employs input feature maps to predict top-left and bottom-right corner keypoints of the bounding box for objects. Based on the corner keypoints, object proposals are formed according to the corner locations and the corresponding classifying scores. Since there are many false positive object proposals generated by the first stage, an efficient detection network is designed in the second stage. Specifically, a lightweight classification subnet is first designed to remove most false positive object candidates. A detection subnet with light-head structure is then adopted to produce prediction results based on remaining object candidates. With the anchor-free scheme for proposal generation in the initial stage and an efficient detection structure in the following stage, the proposed approach integrates the merits of anchor-free scheme into a two-stage structure. The details of each module are depicted in the subsequent sections.

E. CornerNet as Object Proposal Generation

To obtain high recall rate for generating object proposals from input images, especially for objects with various shapes, this paper adopts CornerNet [11] as object proposal generation network. CornerNet identifies an object through two crucial keypoints - the top-left corner keypoint and the bottom-right corner keypoint. The structure of CornerNet, as depicted in Fig. 5, involves the utilization of the Hourglass model [19] to extract feature maps from input images. The Hourglass network initially processes input features through convolution and max pooling layers to reduce the resolution and then uses up-sampling, convolution layers, and skip layers to increase the resolution back to its original state. The Hourglass architecture combines both global and local features into a single structure. As in [11], this paper employs the Hourglass architecture with two Hourglass modules for extracting input features. The final layer of the Hourglass network is utilized for further prediction by using two prediction branches with identical structures. These prediction branches, based on the last feature map produced by Hourglass, detect the top-left and bottom-right corners. Each branch generates *C* channel heatmaps, where *C* represents the number of object categories. Each channel is a binary map that shows the locations of corners for each class. To refine the corner locations, each branch predicts offset values. A corner pooling module, consisting of two 3×3 convolution layers followed by a corner pooling layer, is utilized in each branch to pool features from the Hourglass network. These features are then fed into a 3×3 convolution layer for projection. Finally, the output features are used to produce heatmaps and offsets through a series of 3×3 and 1×1 convolution layers. It should be noted that since this paper adopts CornerNet for predicting corners, the embedding prediction branch in the original CornerNet is removed, thus reducing the computation of the proposed object proposal generation network.

After getting proposal corners through the proposed CornerNet, this paper extracts K top-left and K bottom-right corners from the heatmaps generated by CornerNet (K = 50 in this paper). Then, each pair of top-left and bottom-right corners belonging to the same class, where the coordinates of top-left corner are smaller than that of bottom-right corner is used to define an object proposal. By using corner points to define object candidates, the proposed object proposal generation method can detect more objects, especially objects with arbitrary size, which are usually missed by anchor-based proposal generation method. As a result, the recall rate is significantly improved. However, defining object proposals based on this scheme leads to many false positive proposals as the corner keypoints of two different objects of the same class may define an object proposal (as shown in Fig. 3). To eliminate most false proposal candidates, this paper designs an efficient detection network which is elaborated in the next subsection.

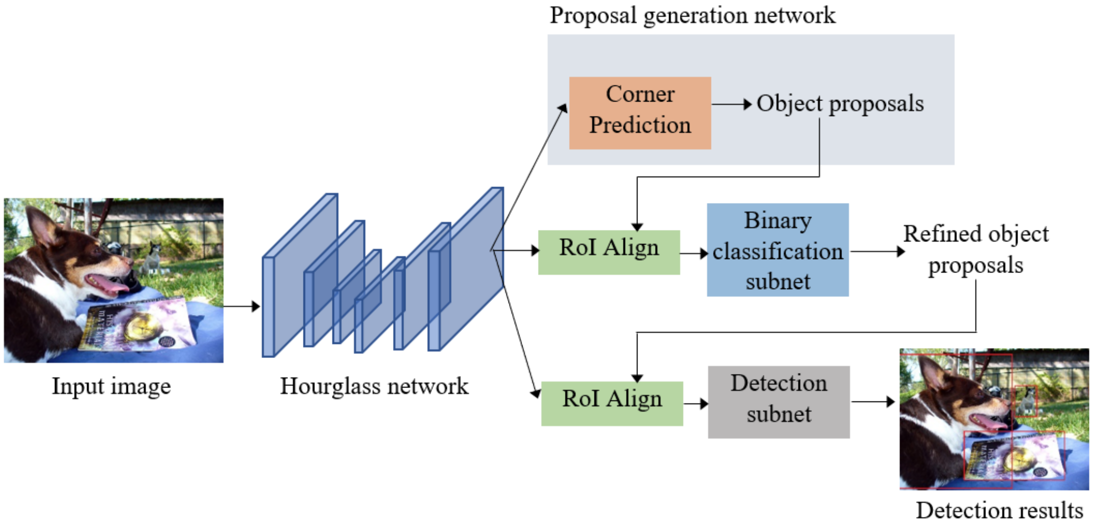

Fig. 4. The overall structure of the proposed model.

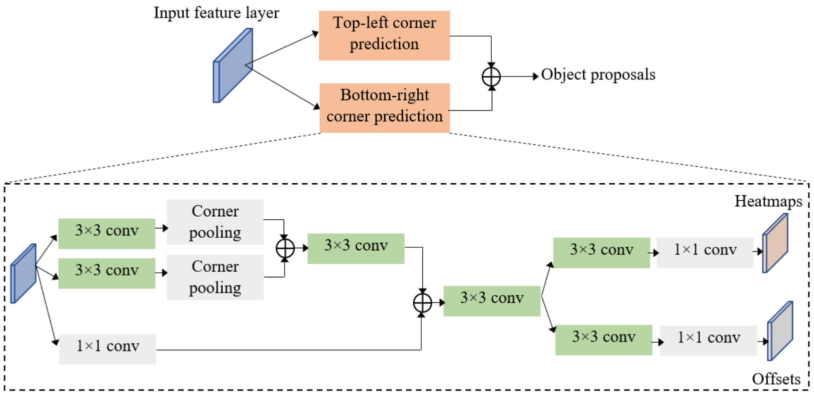

Fig. 5. The architecture of CornerNet used in this paper.

F. Detection Network

Since this paper employs CornerNet for generating object proposals, many false positive proposal candidates are produced. As a result, applying a heavy detection network for predicting objects based on a large number of object candidates is not efficient since it requires a huge amount of computational budget. In the paper, a high-performance detection network is proposed. The structure of the proposed network is depicted in Fig. 6. First, this paper employs a lightweight binary classification subnet to eliminate most of false positive proposal candidates. The lightweight binary classification subnet starts by applying a convolution layer to the final feature layer of the backbone to create a thin feature map with 32 channels. The RoIAlign layer [20] then creates the proposal feature map using the thin feature map and proposals from the proposed CornerNet. At the end of the lightweight binary classification subnet, a convolution layer followed by an average pooling layer computes the classification score for each proposal. To address the issue of imbalanced training samples, this paper implements a variation of focal loss [6] in the training process, which is illustrated as follows:

$$L_{class1} = -\frac{1}{N}\sum_i L_i \qquad (1)$$

$$L_i = \begin{cases} (1-p_i)^{\beta}\log(p_i)\,, & if\ IoU_i \geq T \\ {p_i}^{\beta}\log(1-p_i)\,, & otherwise \end{cases} \qquad (2)$$

where N represents the number of positive samples, p_i is the objectness score of i-th proposal, IoU_i is the maximum IoU value between i-th proposal and all ground truth boxes. T and β are set at 0.7 and 2, respectively, in this paper.

Next, since the lightweight binary classification subnet effectively eliminates most false positive proposal candidates, a light-head detection subnet is utilized to generate final predictions from the remaining proposals. This paper designs a light-head structure, similar to Light-Head R-CNN [21], in the final detection subnet for both classification and bounding box regression. The light-head detection subnet can improve computational speed without compromising the detection performance. As illustrated in Fig. 6, the first step is to apply a large separable convolution layer to the final feature layer of the backbone to improve these features while simultaneously decreasing the number of channels. Compared with 1×1 convolution, large separable convolution is more efficient as it produces thin output features with more semantic information. Then, PSRoI Align [22] is adopted to produce fixed size features (i.e., 7×7×10) for remaining proposal candidates based on feature map generated by the large separable convolution layer. Here, the PSRoI Align is utilized as it reduces the number of channels in the output features. Finally, this paper uses a single 1024-dimensional fully connected layer followed by two parallel fully connected layers to produce the final classification and regression results.

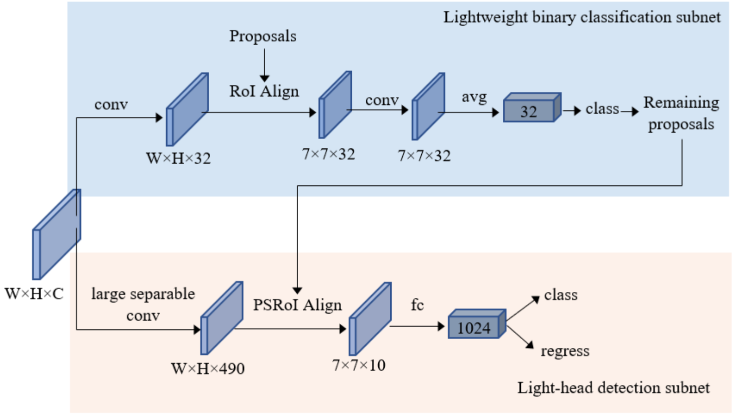

Fig. 6. The structure of the proposed detection network.

By designing an efficient detection network with two prediction steps, most of false positive object candidates are removed by the classifier, and remaining object candidates are efficiently predicted by the detection subnet. The detection subnet with light-head structure can enhance the detection speed without compromising the detection performance. The experimental results demonstrate that this design is more efficient than using a heavy detection network directly as the detection network.

IV. Results

A. Implementation Details

This paper uses the MS-COCO dataset [23] to evaluate the proposed model, which contains 80 object categories. The images in the dataset are divided into three sets, with 80K images for training, 40K images for validation, and 20K images for testing. In accordance with the standard protocol [4], [6], this paper trains the proposed model using all images in the training set and 35K images from the validation set. To evaluate the detection performance, the paper reports the results on the test-dev set, which are submitted to an external evaluation server.

For evaluation metrics, this paper follows metrics defined in the MS-COCO dataset for evaluating object detection tasks. To be more specific, this paper uses the average precision (AP), AP_{50}, AP_{75}, AP_S, AP_M, and AP_L as evaluation metrics. AP is the average precision over 80 categories under multiple IoU values (i.e., 0.5:0.05:0.95). AP is considered the key metric when assessing object detection techniques on the MS-COCO dataset. AP_{50} and AP_{75} are AP computed at a specific IoU threshold value. AP_S, AP_M, and AP_L are AP computed based on object sizes (AP_S for objects with area $< 32^2$, AP_M for objects with $32^2 <$ area $< 96^2$, and AP_L for objects with area $> 96^2$).

The proposed network is designed based on Pytorch [24] and open-source object detection toolbox mmdetection [25]. The object proposal generation network is adopted from CornerNet [11], where the stacked Hourglass networks with 104 layers and the corner detection network are trained on the MS-COCO dataset. As in [11], the input size of the network is set to 511×511 during the training process. The proposed network is trained end-to-end with the full training loss as follow:

$$L = L_{corner} + L_{offset1} + L_{class1} + L_{class2} + L_{offset2} \quad (3)$$

where L_{corner} and $L_{offset1}$ are the variant of focal loss and the smooth L_1 loss, respectively; L_{class1} is the variant of focal loss for training the binary classification subnet; L_{class2} and $L_{offset2}$ are the cross-entropy loss and the smooth L_1 loss, respectively, for training the light-head detection network. For computationally efficient reasons, this paper uses Adam optimizer [26] to optimize the training loss. The Adam optimizer is a widely used optimization algorithm in the deep learning domain and is straightforward to implement. The proposed network is trained for 100K iterations on Nvidia RTX 3070 GPU.

During the inference process, this paper adopts a confidence threshold of 0.3 to remove false positive proposal candidates by the binary classification subnet. In addition, Soft-NMS [27] is employed after the light-head detection network to eliminate redundant boxes, and the top 150 scoring boxes are selected for evaluation.

B. Detection Results on the MS COCO Dataset

The detection accuracy of the proposed method is shown in Table I alongside recent state-of-the-art object detection methods, both anchor-based and anchor-free pipelines, on the MS-COCO test-dev set. The results in the table demonstrate that the proposed model significantly outperforms the CornerNet baseline model [11]. To be more specific, the proposed model improves AP, AP_{50}, AP_{75}, AP_S, AP_M, and AP_L by 6.8, 11.0, 10.7, 7.6, 8.7, and 8.1 points, respectively, compared with CornerNet on the same backbone network and input size. The results show that the combining of the binary classifier and the light-head detection network are very efficient for removing false positive object candidates and predicting remaining object candidates. Compared with Faster R-CNN using ResNet-101 backbone, which is a popular two-stage anchor-based object detection framework, the proposed model also achieves significantly higher detection accuracy. The proposed network improves AP, AP_{50}, AP_{75}, AP_S, AP_M, and AP_L by 11.2, 8.3, 14.9, 8.5, 12.5, and 14.2 points, respectively, compared with Faster R-CNN using ResNet-101 backbone. The results demonstrate that using an anchor-free approach to generate object proposals and designing a suitable detection network can enhance object detection performance. As seen in Table I, the proposed model also outperforms both anchor-based and anchor-free models while using lower input resolution. Moreover, as AP_S, AP_M, and AP_L denote AP for predicting objects at different sizes, the results in Table I reveal that the proposed model has improved performance in detecting medium and large objects compared to small ones. To be more specific, compared with CornerNet, the proposed model improves AP_S, AP_M, and AP_L by 7.6, 8.7, and 8.1 points, respectively. This result shows that the proposed network has difficulty in detecting small objects since predicting small objects requires richer semantic information features, which can be achieved by employing feature pyramids.

Table II shows the inference speed of the proposed model and several efficient methods on the MS-COCO dataset. All methods are implemented on Nvidia RTX 3070 GPU. As demonstrated in Table II, the proposed model obtains 2.8 fps on the MS-COCO dataset with input resolution 511×511, which is comparable to the speed of the baseline CornerNet. The results shown in Table II also indicate that Faster R-CNN and FCOS have better speed, however, the proposed method achieves a better detection accuracy as shown in Table I. This demonstrates that the proposed method strikes a good balance between inference speed and detection accuracy.

TABLE I. EVALUATION OF THE PROPOSED METHOD AGAINST RECENT OBJECT DETECTION TECHNIQUES ON THE MS-COCO TEST-DEV SET IN TERMS OF DETECTION ACCURACY

Method	**Backbone network**	**Input resolution**	**AP**	**AP$_{50}$**	**AP$_{75}$**	**AP$_{S}$**	**AP$_{M}$**	**AP$_{L}$**
Anchor-based methods								
Faster R-CNN [3]	ResNet-101	600×1000	36.2	59.1	39.0	18.2	39.0	48.2
Light-Head R-CNN [21]	ShuffleNetV2	800×1200	23.7					
ThunderNet [28]	SNet535	320×320	28.1	46.2	29.6	-	-	-
RetinaNet [6]	ResNet-101	800×1200	39.1	59.1	42.3	21.8	42.7	50.2
Mask R-CNN [20]	ResNext-101	800×1200	39.8	62.3	43.4	22.1	43.2	51.2
TridentDet [29]	ResNet-101	800×1200	42.7	63.6	46.5	23.9	46.6	56.6
YOLOv4 [30]	Darknet-53	608×608	43.5	65.7	47.3	26.7	46.7	53.3
Cascade R-CNN [32]	ResNet-101	800×1200	42.8	62.1	46.3	23.7	45.5	55.2
FFAD [33]	ResNet-101	800×1333	44.1	62.2	47.9	27.4	47.6	56.7
Anchor-free methods								
FCOS [16]	ResNext-101	800×1024	44.7	64.1	48.4	**27.6**	47.5	55.6
CornerNet [11]	Hourglass-104	511×511	40.6	56.4	43.2	19.1	42.8	54.3
CenterNet [12]	Hourglass-104	511×511	44.9	62.4	48.1	25.6	47.4	57.4
FoveaBox [17]	ResNext-101	800×1024	42.1	61.9	45.2	24.9	46.8	55.6
SAPD [18]	ResNext-101	800×1024	45.4	65.6	48.9	27.3	48.7	56.8
Proposed method	Hourglass-104	511×511	**47.4**	**67.4**	**53.9**	26.7	**51.5**	**62.4**

TABLE II. THE INFERENCE SPEED OF THE PROPOSED MODEL AND SEVERAL EFFICIENT METHODS ON THE MS-COCO DATASET

Method	**Backbone network**	**Input resolution**	FPS
Faster R-CNN [3]	ResNext-101	600×1000	4.2
CornerNet [11]	Hourglass-104	511×511	3.0
CenterNet [12]	Hourglass-104	511×511	2.6
FCOS [16]	ResNext-101	800×1024	4.1
Proposed method	Hourglass-104	511×511	2.8

C. Ablation Study on Detection Network

To assess the efficacy of the proposed detection network for anchor-free proposal generation scheme, this paper examines the detection performance of several structures on the MS-COCO validation set. First, the original R-CNN detection head [3] is applied on the last backbone feature layer to produce final predictions based on object proposals produced by the proposed CornerNet. The proposed CornerNet extracts 50 top-left corner points and 50 bottom-right corner points based on the heatmaps to form object proposals. The R-CNN architecture consists of two fully connected layers with 4096 neurons each, featuring ReLU activations, followed by two additional fully connected layers for performing classification and regression tasks. Second, the proposed lightweight binary classification subnet is applied to remove false positive proposals. This binary classification takes 7×7×32 feature maps generated by a RoIAlign layer as inputs. Remaining proposals are fed into the original R-CNN detection head to output final predictions. Finally, the proposed light-head detection subnet is applied on the last backbone feature layer to produce predictions without the lightweight binary classification subnet. It should be noted that a large separable convolution layer is employed to reduce the feature map channels to 490 before feeding to the light-head detection subnet. For all experiments, this paper employs the same input resolution at 511×511 for fair comparison. Table III illustrates the detection performance of the proposed structures. As demonstrated in Table III, directly applying the R-CNN subnet on object proposals generated by the proposed CornerNet does not improve AP as many false positive proposals hinder the classification ability of R-CNN. When employing the binary classification subnet before the R-CNN subnet, the detection performance is improved. However, the detection speed is reduced as this structure uses a heavy detection head for prediction. On the other hand, using the light-head detection subnet after the binary classification subnet improves both the detection performance and speed. The result shows that the proposed detection network with a binary classification subnet and a light-head detection subnet obtains the best trade-off between detection accuracy and speed.

TABLE III. EVALUATION OF VARIOUS DESIGNS ON THE MS-COCO VALIDATION SET IN TERMS OF THEIR DETECTION ABILITY

Method	**Input resolution**	**AP**	**FPS**
CornerNet [11]	511×511	41.0	3.0
CornerNet + R-CNN	511×511	40.8	2.1
CornerNet + binary classifier + R-CNN	511×511	46.4	1.6
CornerNet + light-head detection subnet	511×511	41.1	3.8
CornerNet + binary classifier + light-head detection subnet	511×511	46.8	2.8

V. CONCLUSIONS

This paper presents a new object detection framework that combines the benefits of anchor-free and two-stage approaches. In the first stage, an anchor-free scheme is designed to generate object proposals. In the second stage, an efficient detection network comprised of a lightweight binary classification subnetwork for reducing false positive object proposals and a light-head detection subnetwork for final predictions is utilized. The proposed model was tested on the MS-COCO dataset and achieved the best balance between speed and accuracy compared to state-of-the-art anchor-based and anchor-free object detection methods. Specifically, the proposed model obtains 47.4 of AP on the MS-COCO test-dev set, which surpasses both anchor-free and one-stage model baselines. The focus of this study was on efficiency, and thus techniques for improving accuracy, such as combining different network layers or using multi-layer predictions, were not explored. As a result, the model struggles with detecting small objects. Future work will focus on improving the detection of small objects by replacing the backbone network with a feature pyramid network.

REFERENCES

[1] Huang, Jonathan, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer et al. "Speed/accuracy trade-offs for modern convolutional object detectors." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310-7311. 2017.

[2] Lu, Xin, Quanquan Li, Buyu Li, and Junjie Yan. "Mimicdet: Bridging the gap between one-stage and two-stage object detection." In *European Conference on Computer Vision*, pp. 541-557. Springer, Cham, 2020.

[3] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems* 28 (2015).

[4] Lin, Tsung-Yi, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. "Feature pyramid networks for object detection." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117-2125. 2017.

[5] Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. "Ssd: Single shot multibox detector." In *European conference on computer vision*, pp. 21-37. Springer, Cham, 2016.

[6] Lin, Tsung-Yi, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. "Focal loss for dense object detection." In *Proceedings of the IEEE international conference on computer vision*, pp. 2980-2988. 2017.

[7] Fu, Cheng-Yang, Wei Liu, Ananth Ranga, Ambrish Tyagi, and Alexander C. Berg. "Dssd: Deconvolutional single shot detector." *arXiv preprint arXiv:1701.06659* (2017).

[8] Yang, Tong, Xiangyu Zhang, Zeming Li, Wenqiang Zhang, and Jian Sun. "Metaanchor: Learning to detect objects with customized anchors." *Advances in neural information processing systems* 31 (2018).

[9] Zhong, Yuanyi, Jianfeng Wang, Jian Peng, and Lei Zhang. "Anchor box optimization for object detection." In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1286-1294. 2020.

[10] Sun, Peize, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka et al. "Sparse r-cnn: End-to-end object detection with learnable proposals." In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 14454-14463. 2021.

[11] Law, Hei, and Jia Deng. "Cornernet: Detecting objects as paired keypoints." In *Proceedings of the European conference on computer vision (ECCV)*, pp. 734-750. 2018.

[12] Duan, Kaiwen, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. "Centernet: Keypoint triplets for object detection." In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6569-6578. 2019.

[13] Yang, Ze, Shaohui Liu, Han Hu, Liwei Wang, and Stephen Lin. "Reppoints: Point set representation for object detection." In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9657-9666. 2019.

[14] Zhang, Haoyang, Ying Wang, Feras Dayoub, and Niko Sunderhauf. "Varifocalnet: An iou-aware dense object detector." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8514-8523. 2021.

[15] Dai, Jifeng, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. "Deformable convolutional networks." In *Proceedings of the IEEE international conference on computer vision*, pp. 764-773. 2017.

[16] Tian, Zhi, Chunhua Shen, Hao Chen, and Tong He. "Fcos: Fully convolutional one-stage object detection." In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9627-9636. 2019.

[17] Kong, Tao, Fuchun Sun, Huaping Liu, Yuning Jiang, Lei Li, and Jianbo Shi. "Foveabox: Beyound anchor-based object detection." *IEEE Transactions on Image Processing* 29 (2020): 7389-7398.

[18] Zhu, Chenchen, Fangyi Chen, Zhiqiang Shen, and Marios Savvides. "Soft anchor-point object detection." In *European conference on computer vision*, pp. 91-107. Springer, Cham, 2020.

[19] Newell, Alejandro, Kaiyu Yang, and Jia Deng. "Stacked hourglass networks for human pose estimation." In *European conference on computer vision*, pp. 483-499. Springer, Cham, 2016.

[20] He, Kaiming, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. "Mask r-cnn." In *Proceedings of the IEEE international conference on computer vision*, pp. 2961-2969. 2017.

[21] Li, Zeming, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. "Light-head r-cnn: In defense of two-stage object detector." *arXiv preprint arXiv:1711.07264* (2017).

[22] Dai, Jifeng, Yi Li, Kaiming He, and Jian Sun. "R-fcn: Object detection via region-based fully convolutional networks." *Advances in neural information processing systems* 29 (2016).

[23] Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. "Microsoft coco: Common objects in context." In *European conference on computer vision*, pp. 740-755. Springer, Cham, 2014.

[24] Paszke, Adam, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. "Automatic differentiation in pytorch." (2017).

[25] Chen, Kai, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun et al. "MMDetection: Open mmlab detection toolbox and benchmark." *arXiv preprint arXiv:1906.07155* (2019).

[26] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).

[27] Bodla, Navaneeth, Bharat Singh, Rama Chellappa, and Larry S. Davis. "Soft-NMS--improving object detection with one line of code." In *Proceedings of the IEEE international conference on computer vision*, pp. 5561-5569. 2017.

[28] Qin, Zheng, Zeming Li, Zhaoning Zhang, Yiping Bao, Gang Yu, Yuxing Peng, and Jian Sun. "ThunderNet: Towards real-time generic object detection on mobile devices." In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6718-6727. 2019.

[29] Li, Yanghao, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. "Scale-aware trident networks for object detection." In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6054-6063. 2019.

[30] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection." *arXiv preprint arXiv:2004.10934* (2020).

[31] BOURJA, Omar, Hatim DERROUZ, Hamd AIT ABDELALI, Abdelilah MAACH, Rachid OULAD HAJ THAMI, and François BOURZEIX. "Real time vehicle detection, tracking, and inter-vehicle distance

estimation based on stereovision and deep learning using yolov3." *International Journal of Advanced Computer Science and Applications* 12, no. 8 (2021).

[32] Cai, Zhaowei, and Nuno Vasconcelos. "Cascade r-cnn: Delving into high quality object detection." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6154-6162. 2018.

[33] Yu, Guoyi, You Wu, Jing Xiao, and Yang Cao. "A novel pyramid network with feature fusion and disentanglement for object detection." *Computational Intelligence and Neuroscience* 2021 (2021).

[34] Zhu, Chenchen, Yihui He, and Marios Savvides. "Feature selective anchor-free module for single-shot object detection." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019.

[35] Zhou, Xingyi, Dequan Wang, and Philipp Krähenbühl. "Objects as points." *arXiv preprint arXiv:1904.07850* (2019).