第 10 卷 第 5 期
2021 年 9 月
集 成 技 术
JOURNAL OF INTEGRATION TECHNOLOGY
Vol. 10 No. 5
Sep. 2021

引文格式：
王宏任, 陈世峰. 基于关键点检测二阶段目标检测方法研究 [J]. 集成技术, 2021, 10(5): 34-42.
Wang HR, Chen SF. Research on two-stage object detection method based on key point detection [J]. Journal of Integration Technology, 2021, 10(5): 34-42.

基于关键点检测二阶段目标检测方法研究

王宏任[1,2] 陈世峰[1*]

[1]（中国科学院深圳先进技术研究院 深圳 518055）
[2]（中国科学院大学深圳先进技术学院 深圳 518055）

摘 要 卷积神经网络被广泛应用于目标检测领域。该文提出一种新的无锚框二阶段目标检测算法：以 CornerNet 方法为基础，借助角点提取候选区域，并增加中心池化层来增强物体中心区域特征，通过判断中心关键点是否落在中心区域，可以过滤掉大量的误检候选框。随后，将保留的候选框送到多元分类器进行预测和回归，获取最终的检测结果。实验结果表明，该方法在 MS-COCO 数据集上能够取得 46.7% 的检测精度，与其他同类算法相比具有较强的竞争力。与原始的 CornerNet 算法相比，该方法在精度上有 6.2% 的提升，尤其对于形状特殊的物体，精度提升更加明显。

关键词 无锚框；二阶段；中心关键点；中心区域；多元分类器
中图分类号 TP 399 **文献标志码** A doi: 10.12146/j.issn.2095-3135.20210315001

Research on Two-stage Object Detection Method Based on Key Point Detection

WANG Hongren[1,2] CHEN Shifeng[1*]

[1](*Shenzhen Institute of Advanced Technology*, *Chinese Academy of Sciences*, *Shenzhen* 518055, *China*)
[2](*Shenzhen College of Advanced Technology*, *University of Chinese Academy of Sciences*, *Shenzhen* 518055, *China*)
[*]*Corresponding Author*: *shifeng.chen@siat.ac.cn*

Abstract Convolutional neural network is widely used in the field of object detection. In this paper, a novel anchor-free two-stage object detection algorithm is investigated. Region proposals are produced via corner points extracted based on CornerNet. In order to improve the inception ability to the internal information of the object, central pooling is introduced in the algorithm to enhance the features of interal regions for internal feature point detection. A large number of false-positive proposals can be filtered out by checking whether the internal key points exist in the internal area. The remaining proposals are fed into a multivariate classifier to obtain the final result. The proposed algorithm has been tested on the data set of

收稿日期：2021-03-15 **修回日期**：2021-04-06
基金项目：深圳市科技创新委员会基础研究重点项目（JCYJ20200109114835623）；国家自然科学基金委员会重点项目（U1713203）；广东省重点领域研发计划项目（2019B010155003）
作者简介：王宏任，硕士研究生，研究方向为目标检测；陈世峰（通讯作者），博士，副研究员，博士研究生导师，研究方向为计算机视觉，E-mail：shifeng.chen@siat.ac.cn。

MS-COCO with an accuracy of 46.7%, which is a competitive result compared to that of the state-of-the-art object detection methods. The proposed algorithm outperforms CornerNet by 6.2% in accuracy. For the objects with special (huge, tiny, or large aspect-ratio) shapes, higher accuracy increments can be obtained which demonstrates the effectiveness of the proposed algorithm.

Keywords　anchor-free; two-stage; internal key point; internal area; multivariate classifier

Funding　This work is supported by Shenzhen Science and Technology Innovation Commission (JCYJ20201009114835623), National Natural Science Foundation of China (U1713203), and Key-Area Research and Development Program of Guangdong Province (2019B010155003)

1　引　言

目标检测是计算机视觉中很常见的任务。根据有无提取候选区域(Region Proposal)，目标检测领域的检测方法通常分为一阶段(One-stage)检测网络和二阶段(Two-stage)检测网络。其中，一阶段检测方法直接回归物体的类别概率和位置坐标值。常见的一阶段算法包括：YOLOv1[1]、YOLOv2[2]、YOLOv3[3]、SSD[4]、DSSD[5]和Retina-Net[6]。二阶段检测方法的任务包括第一阶段提取候选区域以及第二阶段将候选区域送到分类器进行分类与检测。常见的二阶段算法包括：R-CNN[7]、SPP-Net[8]、Fast R-CNN[9]、Faster R-CNN[10]、Mask R-CNN[11]和 Cascade R-CNN[12]。与一阶段检测网络相比，二阶段检测网络的检测精度更高，但速度慢于一阶段检测网络。

另外，根据是否利用锚框(Anchor)提取候选目标框，目标检测框架也可分为基于锚框的方法(Anchor-based)、基于无锚框的方法(Anchor-free)以及两者融合类。其中，基于锚框类算法有 Fast R-CNN、SSD、YOLOv2 和 YOLOv3；基于无锚框类算法有 CornerNet[13]、ExtremeNet[14]、CenterNet[15]和 FCOS[16]；融合基于锚框和基于无锚框分支的方法有 FSAF[17]、GA-RPN[18]和 SFace[19]。

目前，所有的主流探测器，如 Faster R-CNN、SSD、YOLOv2 和 YOLOv3 都依赖一组预先定义的锚框。其中，人们认为锚框的使用是检测器成功的关键。尽管这些主流探测器取得了巨大的成功，但基于锚框方法仍存在一些缺点：(1)即使经过仔细的设计，但由于锚框的尺度和长宽比是预先设定的，检测器在处理形状变化较大的候选物体时也会遇到困难，尤其是对于小物体，这无疑阻碍了检测器的泛化能力；(2)为了达到较高的召回率，需要在输入图像上密集放置锚框(如对于短边为 800 的图像，在特征金字塔网络(FPN)中放置超过 180k 的锚框)，但大多数锚框在训练中被标记为负样本，而过多的负样本会加剧训练中正负样本之间的不平衡；(3)锚框涉及复杂的计算，如计算与真实边框(Ground-truth)的重叠度(Intersection over Union，IoU)。

为了克服基于锚框方法的缺点，CornerNet 采用基于关键点检测角点提取候选区域的方法：利用单个卷积神经网络来检测一个以左上角和右下角为一对关键点的目标包围框，通过将目标作为成对的关键点进行检测，消除了以往检测器通常需要人为设计锚框的需要。然而，CornerNet 也存在一些问题：(1)CornerNet 对物体内部信息的感知能力相对较弱，制约了 CornerNet 的性能。(2)在进行关键点配对时，CornerNet 认为属

于同一类别的关键角点间应尽可能靠近，属于不同类别的关键角点间应尽可能远离。但在实验过程中发现，通过计算左上角点的嵌入向量及右下角点的嵌入向量间的距离来决定是否将两个点进行组合，经常会发生配对错误的情况。(3)采用关键点配对的方式确定一个目标的候选区域，会产生大量误检目标的候选区域，这样不仅会使检测精度降低而且会花费较长时间。本文提出一种新的无锚框二阶段目标检测算法对以上 3 个问题进行优化。

2 基于关键点目标检测方法

本文将 CornerNet 作为基准，提出一种基于无锚框 3 个关键点检测的二阶段目标检测网络方法。如图 1 所示：第一阶段采用基于无锚框关键点检测的方法分别检测角点以及中心关键点，同时判断中心点是否落在中心区域以进行误检候选区域的剔除，即提取候选区域；第二阶段将第一阶段过滤后保留下来的候选区域送到多元分类器中进行分类与检测。

2.1 基于无锚框 3 个关键点检测

为了检测角点，本文先采用基于 CornerNet 关键点检测的方法来定位左上以及右下角点；然后，通过角点池化[13]生成左上角以及右下角两个热图来代表不同类别关键点的位置；最后，进行角点关键点的偏移修正。

另外，为了加强网络对物体内部信息的感知能力，本文增加了中心关键点的检测分支，并采用中心池化操作加强中心点的特征。同时定义了物体中心度的概念——设定中心度大于 0.7 时，可认为中心关键点落在中心区域，很好地解决了不同尺寸物体中心区域的判定。最终，只有当物体的中心点落在预测框的中心区域才进行保留，否则去除。需要说明的是，当中心关键点同时落在多个不同的预测框中时，取中心度最大的那个预测框予以保留，并剔除多余的预测框，以减少误检框出现的概率。具体如图 2 所示。

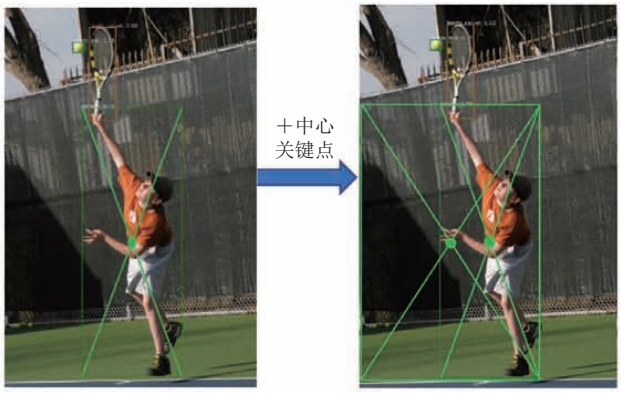

图 2 利用中心关键点过滤误检候选区域

Fig. 2 Filtering false detection candidate regions using the internal key point

2.1.1 角点关键点检测

关于角点关键点的检测，本文借鉴 CornerNet 来定位被检对象的两个角点关键点——分别位于其左上角和右下角。计算 3 个热图(即左上的热图和右下的热图以及中心点的

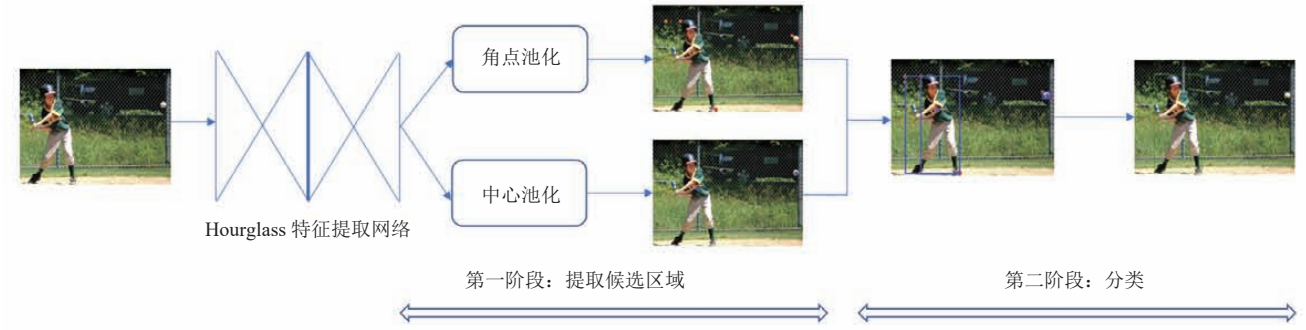

图 1 基于关键点检测的二阶段目标检测方法网络框架

Fig. 1 The network architecture of two-stage object detection method based on key point detection

热图，热图上的每个值表示一个角的关键点出现在相应位置的概率），其分辨率变成原始图像分辨率的 1/4。其中，热图有两个损失，$L_{\text{det}}^{\text{corner}}$ 用来定位热图上的左上角关键点，$L_{\text{offsets}}^{\text{corner}}$ 用来定位热图上的右下角关键点和偏移损失，具体如公式（1）～（3）。在计算热图之后，从所有热图中提取固定数量的关键点（左上角 k 个，右下角 k 个），每个角点的关键点都配有一个类标签。

$$L_{\text{det}}^{\text{corner}}=\begin{cases}\dfrac{-1}{N}\displaystyle\sum_{c=1}^{C}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(1-p_{cij}\right)^{\alpha}\log\left(p_{cij}\right) & \text{如果 } y_{cij}=1\\ \dfrac{-1}{N}\displaystyle\sum_{c=1}^{C}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(1-y_{cij}\right)^{\beta}\left(p_{cij}\right)^{\alpha}\log\left(1-p_{cij}\right) & \text{其他}\end{cases} \tag{1}$$

其中，C 为目标的类别；H、W 分别为热图的高和宽；p_{cij} 为预测热图中 c 类在位置(i, j)的得分；y_{cij} 为加了非归一化高斯热图；N 为图像中物体的数量；α 和 β 为控制每个点贡献的超参数。

$$\boldsymbol{o}_k=\left(\frac{x_k}{n}-\left\lfloor\frac{x_k}{n}\right\rfloor,\frac{y_k}{n}-\left\lfloor\frac{y_k}{n}\right\rfloor\right) \tag{2}$$

$$L_{\text{offsets}}^{\text{corner}}=\frac{1}{N}\sum_{k=1}^{N}Smooth\ L_1\ Loss(\boldsymbol{o}_k,\boldsymbol{o}_k^*) \tag{3}$$

其中，$\boldsymbol{o}_k$ 是偏移量；$\boldsymbol{o}_k^*$ 表示在取整计算时丢失的精度信息；x_k 和 y_k 为角 k 的 x 和 y 坐标；(x_k, y_k) 在映射到热图中为$\left(\frac{x_k}{n},\frac{y_k}{n}\right)$，$n$ 为下采样值，在本文中为 4；$\lfloor\cdot\rfloor$ 表示向下取整。特别地，预测一组由所有类别的左上角共享的偏移量，以及另一组由右下角共享的偏量，在训练时采用 Smooth L_1 Loss[20]。

在进行关键点配对时，CornerNet 认为属于同一类别的关键角点间应尽可能靠近，属于不同类别的关键角点间应尽可能远离[21]。但在实验的过程中，配对关键点时可能会出现错误，同时为了充分利用物体的内部信息，本文将这一机制舍弃，留给二阶段中的多元分类器来完成关键点的配对问题。

2.1.2 中心度——中心区域的定义

为了有效剔除大量误检候选区域，本文通过判断中心关键点是否落在目标框的中心区域的方法来解决此问题。由于每个边界框的大小不同，所以中心区域不能设置为一个固定的数值。本文提出尺度可调节的中心区域定义法如公式（4）所示，引入新的定量指标中心度（Centrality）概念。

$$Centrality=\sqrt{\frac{\min(l,r)}{\max(l,r)}\times\frac{\min(t,b)}{\max(t,b)}} \tag{4}$$

其中，l 为计算中心点到预测框左边的距离；r 为中心点到右侧的距离；t 为中心点到上边框的距离；b 为中心点到下边框的距离，具体如图 3 所示。

图 3　中心度计算

Fig. 3　Centrality calculation

2.1.3 中心池化

中心池化操作参考 CornerNet 的两个角点池化模块——左上角点池化和右下角点池化，分别预测左上角关键点和右下角关键点。每个角点模块有 2 个输入特征图，相应图的宽、高分别用 W 和 H 表示。假设要对特征图上(i, j)点做左上角的角点池化，即计算(i, j)到(i, H)的最大值（最大池化），同时计算(i, j)到(W, j)的最大值（最大池化），随后将这两个最大值相加得到(i, j)点的值。右下角的角点池化操作类似，只不过计算最大值变成从$(0, j)$到(i, j)和从$(i, 0)$到(i, j)。

物体的几何中心不一定具有很明显的视觉特征，如人类头部包含强烈的视觉特征，但中心关键点往往在人体的中间。为了解决这个问题，本文采用中心池化来捕捉更丰富和可识别的视觉特征。图

4 为中心池化的原理：特征提取网络输出一幅特征图(宽、高分别用 W 和 H 表示)，中心池化可通过不同方向上的角点池化的组合实现。其中，水平方向上取最大值的操作可通过左边池化(Left Pooling)和右边池化(Right Pooling)串联实现。同理，垂直方向上取最大值的操作可通过上部池化(Top Pooling)和下部池化(Bottom Pooling)串联实现。

为了判断特征图中的某个像素是否为中心关键点，需要通过中心池化找到其在水平方向和垂直方向的最大值，且将二者相加，这样有助于更好地检测中心关键点。具体操作为特征图的两个分支分别经过一个 3×3 卷积层、BN(Batch Normalization)层以及一个 ReLU 激活函数，做水平方向和垂直方向的角点池化，最后再相加。假设对图上(i, j)点在水平方向做右边池化，即计算(i, j)到(W, j)的最大值(最大池化)；同理，计算左边池化，再将二者串联相加获得(i, j)点水平方向的值。同理，找到垂直方向，最后将水平与垂直方向的值进行相加获得(i, j)点的值。

2.2 分类

采用关键点检测的方式提取候选区域，虽然能够解决需人为设定锚框大小以及长宽比等超参数的问题，大大提高检测的灵活度，但也因此带来了两个问题：大量的误检候选区域以及过滤掉这些误检区域而带来的高计算成本。基于此，本文采取的解决方案主要包括两个步骤：

(1)先判断角点与中心点是否属于同一类别，再通过计算中心点的中心度是否大于 0.7 来过滤掉大量错误的候选区域。

(2)将第一步筛选后存留的候选区域送到之后的多元分类器，对仍存在多个类别的目标分数进行排序。其中，采用 RoIAlign[26]提取每个候选区域上的特征，并通过 256×7×7 卷积层，得到一个表示类别的向量，为每一个存活的候选区域建立单独的分类器。损失函数 L_{class} 为 *Focal Loss*[6]：

$$L_{\text{class}}=\begin{cases}-\dfrac{1}{N}\sum_{n=1}^{M}\sum_{c=1}^{C}(1-p_{nc})^{\partial}\log(p_{nc}) & \text{如果 } IoU_{nc}\geqslant\tau\\ -\dfrac{1}{N}\sum_{n=1}^{M}\sum_{c=1}^{C}p_{nc}^{\partial}\log(1-p_{nc}) & \text{其他}\end{cases}\tag{5}$$

其中，M 和 N 分别为保留的候选区域数量和其中的正样本数量；C 为数据集中与之交叉的类别数；IoU_{nc} 为第 n 个候选区域与第 c 个类别中所有真实框之间的最大 IoU 值；τ 为 IoU 的阈值(设为 0.7)；p_{nc}^{∂} 为第 n 个目标中第 c 个类别的分类分数；∂ 为平滑损失函数的超参数(设为 2)。

3 实 验

3.1 数据集与评估指标

MS-COCO[22]是目前最流行的目标检测基准数据集之一，总共包含 12 万张图片，超过 150 万个边界框，覆盖 80 个对象类别，是一个非常具有挑战性的数据集。本文使用 trainval35k 来训练基于关键点检测二阶段目标检测网络模型，并在 MS-COCO 数据集上进行评估。其中，trainval35k 是由 80k 张训练图片和 35k 张验证

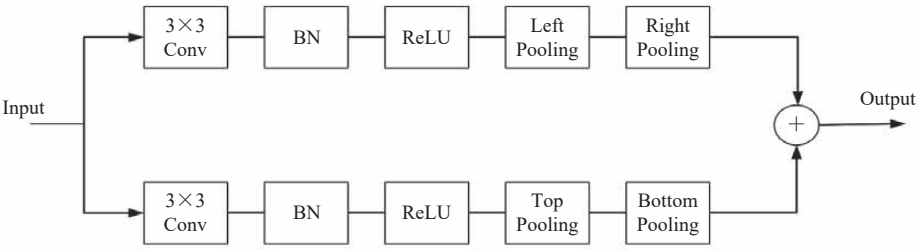

图 4 中心池化结构示意图

Fig. 4 Schematic diagram of central pooling structure

图像的子集组成的联合集。

本文使用 MS-COCO 中定义的平均精度(Average Precision，AP)作为度量来表征网络模型的性能以及其他竞争对手的性能。单个 *IoU* 阈值从 0.5 到 0.95 每隔 0.05 记录一次精度 AP，最后取平均值(即0.5:0.05:0.95)。实验中也记录了一些其他重要指标，如 AP_{50} 和 AP_{75} 为在单个 *IoU* 阈值 0.50 和 0.75 下计算精度，AP_s、AP_m 和 AP_l 为在不同的目标尺度下计算精度(小尺寸物体面积小于 32×32，中尺寸物体面积大于 32×32 小于 96×96，大尺寸物体面积大于 96×96)。所有的度量都是在每个测试图像上允许最多保留 100 个候选区域计算的。

3.2　网络的训练和测试

本文以 CornerNet 作为基线，部分参考了 CornerNet、FCOS 的代码，特征提取网络仍然延用 CornerNet 中采用的 52/104 层的 Hourglass[24] 网络，并借助 Pytorch[23]实现算法。

网络从零开始训练，输入图像的分辨率为 511×511，输出热图的分辨率为 128×128。利用 Adam[25]来优化训练损失，整个网络的损失函数 L 为：

$$L=L_{\text{det}}^{\text{corner}}+L_{\text{offsets}}^{\text{corner}}+L_{\text{det}}^{\text{center}}+L_{\text{offsets}}^{\text{center}}+L_{\text{class}} \tag{6}$$

其中，$L_{\text{det}}^{\text{corner}}$ 和 $L_{\text{det}}^{\text{center}}$ 采用的是 *Focal Loss*，分别用于训练网络检测角点和中心关键点；$L_{\text{offsets}}^{\text{corner}}$ 和 $L_{\text{offsets}}^{\text{center}}$ 采用 Smooth L_1 Loss 分别训练网络预测角点和中心关键点的偏移量。在 8 张 NVIDIA 2080-Ti 上进行模型训练，batch size 大小设为 48(每张卡分配 6 个样本)，前 250k 次迭代学习率设为 2.5×10^{-4}，接下来的 50k 次迭代减小学习率到 2.5×10^{-5}。训练 Hourglass-104、Hourglass-52 的时间分别是 9 d 和 5 d。

4　结果与讨论

本文在通用检测数据集 COCO test-2017 上对近年来比较常见的基于锚框与基于无锚框的检测框架进行精度测试，结果如表 1 所示。从表 1 可知，本文基于无锚框关键点检测的二阶段方法比基于锚框的二阶段方法 YOLOv4 精度提升 3.2%；比基于无锚框的一阶段方法如 FCOS、CenterNet 精度分别提升 5.2% 和 1.8%，比 CornerNet 精度提升 6.2%。其中，在检测尺寸以及长宽比特殊的物体时，检测精度提升更明显。

表 1　本文方法和最先进的检测框架在 COCO test-2017 上的精度对比

Table 1　Inference accuracy of ours and state-of-the-art detectors on the COCO test-2017 set

方法	有无锚框	特征提取网络	输入尺寸	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
Faster R-CNN w/FPN[10]	有	ResNet-101	600	36.2	59.1	39.0	18.2	39.0	48.2
RetinaNet	有	ResNet-101	800	39.1	59.1	42.3	21.8	42.7	50.2
Cascade R-CNN	有	ResNet-101	800	42.8	62.1	46.3	23.7	45.5	55.2
YOLOv4[29]	有	CSPDarknet-53	608	43.5	65.7	47.3	26.7	46.7	53.3
FCOS	无	ResNet-101-FPN	800	41.5	60.7	45.0	24.4	44.8	51.6
CornerNet	无	Hourglass-104	511	40.5	56.5	43.1	19.4	42.7	53.9
FoveaBox[27]	无	ResNet-101	800	42.1	61.9	45.2	24.9	46.8	55.6
CenterNet	无	Hourglass-104	511	44.9	62.4	48.1	25.6	47.4	57.4
CentripetalNet[28]	无	Hourglass-104	511	46.1	63.1	49.7	25.3	48.7	59.2
Ours	无	Hourglass-104	511	46.7	65.2	51.0	26.5	50.2	60.7

注：AP_{50} 和 AP_{75} 为在单个 *IoU* 阈值 0.50 和 0.75 时的精度；AP_s、AP_m、AP_l 分别为小目标、中目标和大目标的检测精度。下同表 2、表 4

这表明，基于无锚框方法进行提取候选区域更具优势。

在单尺度测试时，将原始分辨率的图像和水平翻转的图像输入网络中，而在多尺度测试时，将原始图像的分辨率分别设置为 0.6、1、1.2、1.5 和 1.8 倍。此外，在单尺度评价和多尺度评价中都增加了翻转变量。在多尺度评价时，将所有尺度的预测结果(包括翻转变量)融合到最终结果中，然后使用 soft-NMS 来抑制冗余的限定框，并保留 100 个得分最高的限定框作为最终评价，结果如表 2 所示。

将 3 种不同检测框架与本研究检测方法在 COCO 数据集上进行召回率评估，即记录不同长宽比和不同大小目标的平均召回率(Average Recall，AR)，结果如表 3 所示。

通常来说，在物体非常大时，如尺寸大于 (400×400, ∞)，更容易被检测到。与其他基于无锚框的方法相比，基于锚框的方法 Faster R-CNN 并没有达到期望的较高召回率。但当物体长宽比比较特殊(如 5∶1 和 8∶1)时，基于无锚框的检测方法比基于锚框的方法表现更加优异。这是因为基于无锚框的检测方法摆脱了人为设置锚框长宽比的束缚。本文方法继承了 FCOS 和 CornerNet 的优点，使目标定位更灵活，特别是长宽比例特殊的物体。

本文在 CornerNet 算法基础上加上中心关键点检测分支与原始算法进行对比来进行消融实验，其中特征提取网络采用 Hourglass-52，结果如表 4 所示。分析数据可以看到，当引入中心关键点检测分支后精度提升 3%，小目标检测精度提升 5.8%，大目标检测精度提升 3.6%。表明引入中心关键点检测分支后，小目标误检候选区域去除得更多。这是因为从概率上讲，小目标由于面积小更容易确定其中心点，因此那些误检候选区域不在中心点附近的概率更大。

图 5 为基于锚框方法 Faster R-CNN 与基于无锚框关键点检测的方法进行检测任务的可视化对

表 4 添加中心关键点分支的消融实验

Table 4 The ablation experiment with the addition of the branch of the central key point

方法	特征提取网络	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_s (%)	AP_m (%)	AP_l (%)
CornerNet	Hourglass-52	38.5	54.1	41.1	17.7	41.1	52.5
CornerNet+中心关键点检测	Hourglass-52	41.5	59.2	44.2	23.5	44.0	56.1

表 2 多尺度测试

Table 2 Multi-scale evaluation

方法	特征提取网络	输入尺寸	AP(%)	AP_{50}(%)	AP_{75}(%)	AP_s(%)	AP_m(%)	AP_l(%)
Ours	Hourglass-52	≤1.8×原始尺寸	45.8	63.9	49.7	27.2	48.1	59.5
Ours	Hourglass-104	≤1.8×原始尺寸	49.6	67.4	53.7	31.2	51 8	62.3

表 3 基于锚框和无锚框检测方法的平均召回率(AR)比较

Table 3 Comparison among the average recall (AR) of anchor-based and anchor-free detection methods

方法	特征提取网络	AR (%)	AR_{1+} (%)	AR_{2+} (%)	AR_{3+} (%)	AR_{4+} (%)	$AR_{5:1}$ (%)	$AR_{6:1}$ (%)	$AR_{7:1}$ (%)	$AR_{8:1}$ (%)
Faster R-CNN	X-101-64×4d	57.6	73.8	77.5	79.2	86.2	43.8	43.0	34.3	23.2
FCOS	X-101-64×4d	64.9	82.3	87.9	89.8	95.0	45.5	40.8	34.1	23.4
CornerNet	Hourglass-104	66.8	85.8	92.6	95.5	98.5	50.1	48.3	40.4	36.5
Ours	Hourglass-104	69.2	88.3	93.6	96.0	99.1	54.4	50.6	46.2	35.4

注：X 为 ResNeXt[29]；AR_{1+}、AR_{2+}、AR_{3+}、AR_{4+} 分别表示边界框面积在 $(96^2, 200^2]$、$(200^2, 300^2]$、$(300^2, 400^2]$、$(400^2, \infty)$ 时的召回率；$AR_{5:1}$、$AR_{6:1}$、$AR_{7:1}$、$AR_{8:1}$ 分别表示物体长宽比为 5∶1、6∶1、7∶1、8∶1 时的召回率

图 5　目标检测可视化对比图

Fig. 5　Visual contrast diagram of object detection

比结果。可以看到，本文研究方法无需人为设置锚框大小及长宽比，对于检测小目标以及形状特殊的物体具有更好的检测效果。

5　结　论

本文提出了基于无锚框二阶段目标检测框架，即分别提取角点关键点以及物体中心关键点，并将它们组合成候选区域。通过判断物体中心点是否落在中心区域来过滤掉大量误检候选区域，同时舍弃了 CornerNet 中采取的角点关键点结合的方式，采用二阶段的方式，将保留下来的候选区域送入多元分类器进行分类与回归。

通过以上两个阶段，本文网络模型检测的查全率和准确率均有显著提高，其结果也优于大多数现有目标检测方法，在召回率与检测精度上都取得了良好的表现。最重要的是，基于无锚框的方法在提取候选区域时更加灵活，克服了基于锚框方法需人为设置锚框超参数的缺点。

参 考 文 献

[1] Huang R, Pedoeem J, Chen CX. YOLO-LITE: a real-time object detection algorithm optimized for non-GPU computers [C] // 2018 IEEE International Conference on Big Data, 2018: 2503-2510.

[2] Zhang J, Huang M, Jin X, et al. A real-time Chinese traffic sign detection algorithm based on modified YOLOv2 [J]. Algorithms, 2017, 10(4): 127.

[3] Redmon J, Farhadi A. YOLOv3: an incremental improvement [Z/OL]. arXiv Preprint, arXiv: 1804.02767, 2018.

[4] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector [C] // European Conference on Computer Vision, 2016: 21-37.

[5] Fu CY, Liu W, Ranga A, et al. DSSD: deconvolutional single shot detector [Z/OL]. arXiv Preprint, arXiv: 1701.06659, 2017.

[6] Lin TY, Goyal P, Girshick R, et al. *Focal Loss* for dense object detection [C] // Proceedings of the IEEE International Conference on Computer Vision, 2017: 2980-2988.

[7] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014: 580-587.

[8] Purkait P, Zhao C, Zach C. SPP-Net: deep absolute pose regression with synthetic views [Z/OL]. arXiv Preprint, arXiv: 1712.03452, 2017.

[9] Girshick R. Fast R-CNN [C] // Proceedings of the IEEE International Conference on Computer Vision, 2015: 1440-1448.

[10] Ren S, He K, Girshick R, et al. Faster R-CNN:

towards real-time object detection with region proposal networks [Z/OL]. arXiv Preprint, arXiv: 1506.01497, 2015.

[11] He K, Gkioxari G, Dollár P, et al. Mask R-CNN [C] // Proceedings of the IEEE International Conference on Computer Vision, 2017: 2961-2969.

[12] Cai ZW, Vasconcelos N. Cascade R-CNN: delving into high quality object detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 6154-6162.

[13] Law H, Deng J. CornerNet: detecting objects as paired keypoints [C] // Proceedings of the European Conference on Computer Vision, 2018: 734-750.

[14] Höltinger S, Baumgartner J, Schmidt J, et al. Extreme net load events in fully renewable power systems: a 30 year case study for Sweden [C] // EGU General Assembly Conference Abstracts, 2018: 16883.

[15] Duan KW, Bai S, Xie LX, et al. Centernet: keypoint triplets for object detection [C] // Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019: 6569-6578.

[16] Tian Z, Shen CH, Chen H, et al. FCOS: fully convolutional one-stage object detection [C] // Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019: 9627-9636.

[17] Zhu CC, He YH, Savvides M. Feature selective anchor-free module for single-shot object detection [C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 840-849.

[18] Wang JQ, Chen K, Yang S, et al. Region proposal by guided anchoring [C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 2965-2974.

[19] Wang JF, Yuan Y, Li BX, et al. SFace: an efficient network for face detection in large scale variations [Z/OL]. arXiv Preprint, arXiv: 1804.06559, 2018.

[20] Liu YL, Jin LW. Deep matching prior network: toward tighter multi-oriented text detection [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 1962-1969.

[21] Newell A, Huang ZA, Deng J. Associative embedding: end-to-end learning for joint detection and grouping [Z/OL]. arXiv Preprint, arXiv: 1611.05424, 2016.

[22] Lin TY, Maire M, Belongie S, et al. Microsoft COCO: common objects in context [C] // European Conference on Computer Vision, 2014: 740-755.

[23] Paszke A, Gross S, Chintala S, et al. Automatic differentiation in PyTorch [C] // The 31st Conference on Neural Information Processing Systems, 2017: 1-4.

[24] Newell A, Yang KY, Deng J. Stacked hourglass networks for human pose estimation [C] // European Conference on Computer Vision, 2016: 483-499.

[25] Kingma DP, Ba J. Adam: a method for stochastic optimization [Z/OL]. arXiv Preprint, arXiv: 1412.6980, 2014.

[26] He KM, Gkioxari G, Dollár P, et al. Mask R-CNN [C] // Proceedings of the IEEE International Conference on Computer Vision, 2017: 2961-2969.

[27] Kong T, Sun FC, Liu HP, et al. FoveaBox: beyound anchor-based object detection [J]. IEEE Transactions on Image Processing, 2020, 29: 7389-7398.

[28] Dong ZW, Li GX, Liao Y, et al. CentripetalNet: pursuing high-quality keypoint pairs for object detection [C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 10519-10528.

[29] Xie SN, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017: 1492-1500.