软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2023,34(6):2776−2788 [doi: 10.13328/j.cnki.jos.006509]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于空间语义信息特征融合的目标检测与分割*

郭琪周[1], 袁 春[2,3]

[1](清华大学 计算机科学与技术系, 北京 100084)

[2](清华大学 深圳国际研究生院, 广东 深圳 518055)

[3](鹏城实验室, 广东 深圳 518055)

通信作者: 袁春, E-mail: yuanc@sz.tsinghua.edu.cn

摘 要: 高质量的特征表示可以提高目标检测和其他计算机视觉任务的性能. 现代目标检测器诉诸于通用的特征金字塔结构以丰富表示能力, 但是他们忽略了对于不同方向的路径应当使用不同的融合操作, 以满足其对信息流的不同需求. 提出了分离式空间语义融合 (separated spatial semantic fusion, SSSF) , 它在自上而下的路径中使用通道注意模块 (channel attention block, CAB) 来传递语义信息, 在自下而上的路径中使用具有瓶颈结构的空间注意模块 (spatial attention block, SAB) 来通过较少的参数和较少的计算量 (相比于直接利用不降维的空间注意模块) 将精确的位置信号传递到顶层. SSSF 十分有效, 并且具有很强大的泛化能力: 对于目标检测, 它可以将 AP 提高 1.3% 以上, 对于自上而下的路径进行语义分割的融合操作, 它可以将普通加和版本的 AP 提高约 0.8%, 对于实例分割, 所提方法能够在所有指标上提高实例分割的包围框 AP 和掩膜 AP.

关键词: 目标检测; 特征融合; 注意力机制; 深度学习; 图像分割

中图法分类号: TP18

中文引用格式: 郭琪周, 袁春. 基于空间语义信息特征融合的目标检测与分割. 软件学报, 2023, 34(6): 2776–2788. http://www.jos.org.cn/1000-9825/6509.htm

英文引用格式: Guo QZ, Yuan C. Leveraging Spatial-semantic Information in Object Detection and Segmentation. Ruan Jian Xue Bao/Journal of Software, 2023, 34(6): 2776–2788 (in Chinese). http://www.jos.org.cn/1000-9825/6509.htm

Leveraging Spatial-semantic Information in Object Detection and Segmentation

GUO Qi-Zhou[1], YUAN Chun[2,3]

[1](Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

[2](Shenzhen International Graduate School, Tsinghua University, Shenzhen 518055, China)

[3](Pengcheng Laboratory, Shenzhen 518055, China)

Abstract: High quality feature representation can boost performance for object detection and other computer vision tasks. Modern object detectors resort to versatile feature pyramids to enrich the representation power but neglect that different fusing operations should be used for pathways of different directions to meet their different needs of information flow. This study proposes separated spatial semantic fusion (SSSF) that uses a channel attention block (CAB) in top-down pathway to pass semantic information and a spatial attention block (SAB) with a bottleneck structure in the bottom-up pathway to pass precise location signals to the top level with fewer parameters and less computation (compared with plain spatial attention without dimension reduction). SSSF is effective and has a great generality ability: It improves AP over 1.3% for object detection, about 0.8% over plain addition for fusing operation of the top-down path for semantic segmentation, and boost the instance segmentation performance in all metrics for both bounding box AP and mask AP.

Key words: object detection; feature fusion; attention mechanism; deep learning; image segmentation

---

* 基金项目: 国家自然科学基金 (U1833101); 深圳市基础研究项目 (JCYJ20190809172201639)
收稿时间: 2020-11-30; 修改时间: 2020-12-28, 2021-05-27; 采用时间: 2021-10-04; jos 在线出版时间: 2022-12-08
CNKI 网络首发时间: 2022-12-10

目标检测是计算机视觉领域中基础而又重要的问题, 随着深度学习的发展, 近年来目标检测的精度不断提升、进展飞速[1−5]. 目标检测也是许多任务的基础, 如语义分割、实例分割等[6−14]. 在工业界, 目标检测有着巨大实用价值, 如自动驾驶、安防等. 在深度学习时代, 目标检测的研究主要着眼于多尺度结构的设计、高效计算过程的设计、更精确的检测头的设计等方面. 在多尺度特征方面, 高质量的特征表示对于目标检测器至关重要, 近年来, 有大量的工作都着眼于探索高质量的多级特征融合策略[15−19].

比如, 特征金字塔网络 (feature pyramid network, FPN)[19] 设计了自顶向下的多级特征金字塔结构, 成为了目标检测领域中广泛使用的基础构件之一. 路径聚合网络 (path aggregation network, PANet)[20] 在特征金字塔网络后添加了一个自底向上的路径, 使得低层的特征到顶层特征之间的路径得以缩短. EfficientDet[17]提出了双向特征金字塔网络 (bi-directional feature pyramid network, BiFPN), 在相同层上添加了一个旁路分支连接. 这些方法在各个连结处采用的特征融合操作始终是简单的加和, 他们没有考虑路径具体是如何进行延伸的. 路径延伸的方向不同, 其作用也是不同的. 比如, 自顶向下支路的主要作用是将高层通道中蕴含的语义信息向下传递到低层, 而自底向上支路的主要作用应当是将低层精确的定位信息传递到高层. 在不同的信息通路上, 其特征融合的操作也应该是不同的.

基于此, 我们提出了分离式空间语义融合 (separated spatial semantic fusion, SSSF), 具体的, 我们采用了注意力机制, 在自顶向下的信息通路中, 我们设计了通道注意力模块来更好地从上向下传递语义信息, 在自底向上的通路中, 我们设计了空间注意力模块来更好地从下向上传递定位信息. 其中, 通道注意力模块可以在相邻特征的通道中捕捉相邻层之间的关系, 空间注意力模块可以在相邻特征的空间上捕捉相邻层之间的关系.

我们的贡献总结如下.

(1) 我们提出了对自上而下的路径和自下而上的路径应使用不同的融合操作, 以更好地满足他们的不同需求, 即语义信息在自上而下的路径中传递, 精确的位置信号在自下而上的路径中传递.

(2) 我们提出了自上而下的通道注意力模块 (CAB), 以捕捉两个相邻级别之间的通道相关性, 从而有效地将语义信息从高级特征传递到低级特征.

(3) 我们为自下而上的路径提出了一个具有瓶颈结构的空间注意力模块 (SAB), 来表示当前特征层与较低一级特征层位置之间的相关性, 从而将精确的空间信号传递给顶部, 同时使用更少的内存和计算量.

(4) 我们提出的方法具有很强大的通用能力, 可以使各种计算机视觉任务受益, 包括基于锚点框和无锚的目标检测, 语义分割和实例分割.

1 研究背景

1.1 目标检测的研究方向

现代的目标检测器根据结构大概可以分为两个类别: 两阶段[1,21,22]以及单阶段检测器[2,4,5,23,24]. 两阶段检测器在第 1 个阶段首先会生成一系列候选框的集合, 在第 2 个阶段会对这些候选框进行包围框的再次回归以及分类. 而单阶段检测器没有生成候选框这一步骤, 而是直接一步对检测框的位置以及类别进行预测. 在最新的研究中, 研究人员尝试将检测器中预先定义的锚点框 (anchor box) 去掉, 形成无锚点框的检测器 (anchor-free), 我们提出的方法使用了注意力机制来形成高质量的多尺度特征层, 该方法对于不论是两阶段还是单阶段, 不论是有锚点框还是无锚点框都有效果.

1.2 特征的多尺度融合

特征的多尺度融合是一个广泛研究的话题, 全卷积网络 (fully convolutional network, FCN)[25]以及 Hypercolumns[26] 将多层特征得到的结果进行聚合来计算最后的分割结果. 在目标检测的多尺度融合中, SSD[5] 和 MS-CNN[27] 使用了特征化的图像金字塔, 其中每个特征层都单独地用来生成结果, 在这个过程中不存在特征融合的操作. FPN[3] 进一步添加了自顶向下的信息通路, 并添加了旁路分支进行特征融合, 获得了很大的精度提升. RON[28] 进一步研究了特征融合机制, 提出了反转连接来更好地传递高层特征. MSDNet[29]设计了一个二维的多尺度网络架构, 同时维护了粗糙的特征和细腻的特征. HRNet[30]在前向传播的整个过程中维护了高分辨率的特征表示, 极大

地保留了丰富的高精度信息. PANet[20]在 PFN 后面添加了一个自底向上的分支, 缩短了低层定位信息到顶层的距离. EfficientDet[17]进一步扩展了 PANet, 设计了带权的双向特征金字塔网络, 使得多级特征融合简单又快速. DetectoRS[31]设计了一个递归的特征金字塔结构以及可切换的空洞卷积来更好地利用多尺度特征. DLA[32]提出了深度层级聚合的结构来更好地进行跨层信息的融合. M2Det[15]使用了一系列类似 U-Net 的结构来达到更好的多层特征融合效果.

但是这些方法在自顶向下以及自底向上的信息通路中使用的都是简单的加和操作, 他们忽视了不同方向的信息通路所扮演的不同角色. 我们提出的方法可以满足不同信息通路的不同作用: 在自顶向下的通路上, 高层特征的语义信息被传播到低层; 在自底向上的通路上, 低层特征的空间信息被传播到高层.

1.3 注意力机制在视觉模型中的应用

注意力机制最初起源于机器翻译[33], 之后便应用到了许多自然语言处理以及计算机视觉的任务上去. 自注意力机制[34,35]通过计算所有输入的加权和来得到一个位置上的输出, 表现出了强大的长距离关系捕捉的能力. 非局部神经网络[36]可以认为是自注意力机制的泛化版本.

DANet[9]以及 DFN[37]用注意力机制来加强特征, 从而提高分割的效果. DANet[9]同时使用了空间注意力以及通道注意力来根据他们全局依赖而合成局部的特征. 但是在这个过程中, 每个特征层都是单独进行处理的, 每个位置只从当前层的全局中获得信息. 我们的方法移除了这一限制. DFN[37]设计了两个网络来分别处理类内的不一致以及类间的不可区分, 其中的平滑网络 (smooth network) 将相邻的高底层特征拼接在一起来生成一个通道特征图, 从而将丰富的语义信息传递到低层. 但是特征拼接这一操作并不足以捕捉相邻两层通道之间的关系, 同时, 文章中设计的模块具有很多参数和很大的计算量, 我们的方法可以更好地捕捉相邻特征层通道之间的关系.

我们的方法主要受到 PANet, DFN 以及 DANnet 的启发, 语义信息以及空间信息的处理对比见图 1. 在图 1 中, 虚线框住的地方表示语义信息的处理, 点画线框住的地方表示空间信息的处理. Cur_{in} 和 Cur_{out} 分别表示当前特征层的输入和输出. Low 和 high 分别表示低一层和高一层的特征. 在目标检测的部分, PANet[20]在 FPN[3]后面加了一个额外的自底向上的信息通路, 其所有的信息通路进行的都是简单的加和操作. 在语义分割中, DFN[37]的平滑网络 (smooth network) 将语义信息通过拼接以及聚合来从高层传递到低层. DANet[9]使用了位置注意力模块 (position attention module, PAM) 以及通道注意力模块 (channel attention module, CAM) 对每个特征层进行处理. 我们的方法使用了通道注意力模块 (channel attention block, CAB) 来自顶向下传递语义信息, 还使用了空间注意力模块 (spatial attention block, SAB) 来自底向上传递空间信息.

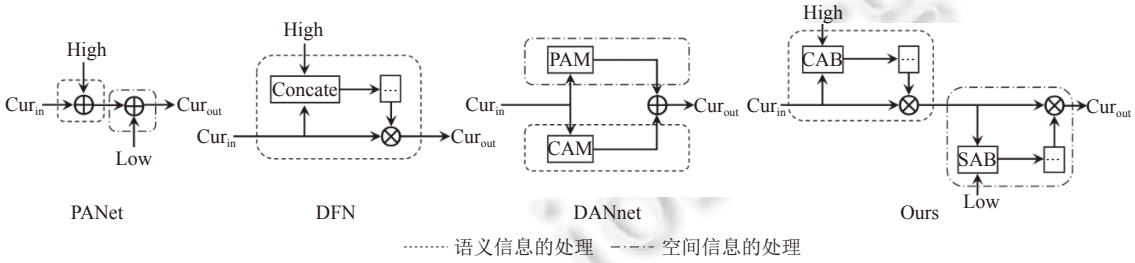


图 1 设计结构演化

2 提出方法

2.1 基于空间语义信息特征融合的总体框架

令 $\mathcal{X}$ 为特征层的集合, 令 $x \in \mathcal{X}$ 为一个张量. 令 x_j 为第 j 层特征, x_{j+1} 为 x_j 的上一层特征. 自顶向下信息通路的融合操作可以写为:

$$y_j = f(x_j, y_{j+1}) \tag{1}$$

类似地, 自底向上的信息通路的融合操作可以写为:

$$y_j = g(x_j, y_{j-1}) \tag{2}$$

其中, $y_j \in \mathcal{Y}$ 是第 j 层的输出, $f(\cdot)$ 和 $g(\cdot)$ 分别是自顶向下通路以及自底向上通路的融合操作符.

之前的工作对于融合操作符 $f(\cdot)$ 和 $g(\cdot)$ 使用的都是简单的加和, 这样忽视了不同信息通路的不同作用. 为了改进这个缺点, 我们为不同的特征通路设计了不同的融合操作符. 具体而言, 我们设计了通道注意力模块 (channel attention block, CAB) 作为自顶向下信息通路的融合操作符, 作用是捕捉当前层和上一层通道之间的相关性, 从而使得语义信息能够高效地从高层传递到低层, 还设计了空间注意力模块 (spatial attention block, SAB) 作为自底向上信息通路的融合操作符, 作用是捕捉当前层和下一层位置之间的相关性, 从而使得精确的空间信息能够从低层传递到高层. 目标检测整体的框架见图 2. 在图 2 中, 类似 RetinaNet[2] 中的操作, 我们首先提取 ResNet 骨干网络的 P_3, P_4, P_5 特征层. 另外两个特征层 P_6, P_7 通过在 P_5 上加两层卷积得到. 另外一个大小为 1×1 的特征层 P_8 通过全局平均池化 (global average pooling) 得到. 从最高层开始, P_7 的输出通过将 P_8 和 P_7 喂给通道注意力模块 (channel attention block, CAB) 得到, 这一输出随后又作为 P_6 输出的输入. 这一过程从上到下依次执行, 直到得到所有层的输出. 红色箭头表示发生了上采样操作, 目的是使通道注意力模块的两个输入的空间大小一致, 但是他们的通道数可以是任意的. 在自底向上的信息通路中, 空间注意力模块 (spatial attention block, SAB) 被用来捕捉每相邻两层之间的关系. 蓝色的箭头表示发生了下采样操作, 目的也是使空间注意力模块的两个输入的空间大小一致. 出于简洁性, 检测头 (用于分类和定位的子网络) 没有画出.

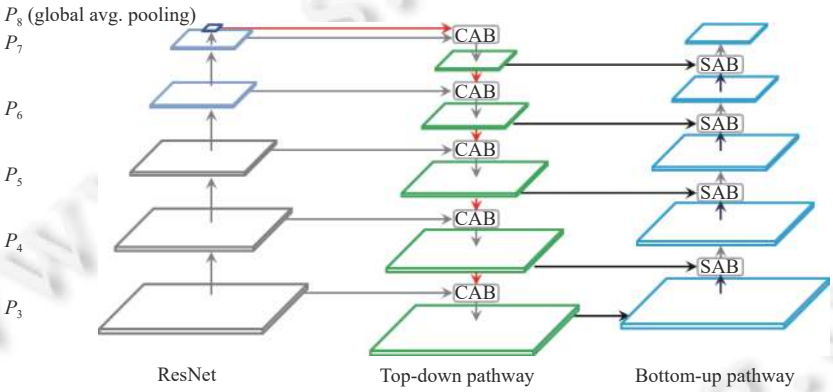


图 2 整体框架

2.2 增强语义信息传递的自顶向下通路

将 P_i 的特征层表示为 $A_i \in \mathbb{R}^{C_i \times H_i \times W_i}$, 其中 C_i, H_i, 和 W_i 分别是第 i 层通道的数量、特征层的高度和宽度. 将 P_{i+1} 的特征层表示为 $A_{i+1} \in \mathbb{R}^{C_{i+1} \times H_{i+1} \times W_{i+1}}$. 通道注意力模块的操作流程见图 3. 每个通道注意力模块接收 A_i 和 A_{i+1} 作为输入. 将 A_{i+1} 的上采样版本表示为 $A^{\star}_{i+1} \in \mathbb{R}^{C_{i+1} \times H_i \times W_i}$. 第 1 步是使用全局平均池化 (global average pooling, *GAP*) 来得到每一个特征层的通道向量 $X_i \in \mathbb{R}^{C_i \times 1}$ 和 $X_{i+1} \in \mathbb{R}^{C_{i+1} \times 1}$:

$$X_i = GAP(A_i) \tag{3}$$

$$A^{\star}_{i+1} = UpSample(A_{i+1}) \tag{4}$$

$$X_{i+1} = GAP(A^{\star}_{i+1}) \tag{5}$$

下一步是使用外积以及 Softmax 函数得到通道注意力图 M:

$$M' = X_i \otimes X_{i+1} \tag{6}$$

$$M_{jk} = \frac{\exp(M'_{j,k})}{\sum_{l=1}^{C_i+1} \exp(M'_{j,l})} \tag{7}$$

其中, $M', M \in \mathbb{R}^{C_i \times C_{i+1}}$. 如公式 (7) 所示, 通道注意力图 M 在最后一个维度进行了归一化. M 中的每一行 $M_{j,\cdot}$ 表示第 i+1 层上相对于第 i 层第 j 个通道的重要性分布. 然后将这个注意力图作用到第 i+1 层经过上采样之后得到的特征图, 便可以得到第 i+1 层重新加权之后的版本:

$$Y_{i+1} = M \otimes A^{\star}_{i+1} \tag{8}$$

其中, $Y_{i+1} \in \mathbb{R}^{C_i \times H_i \times W_i}$. 为了使通道数量一致, 我们将 $A^{\star}_{i+1}$ 传一个 1×1 卷积层来得到 $A'_{i+1} \in \mathbb{R}^{C_i \times H_i \times W_i}$. 最后, 添加残差连接后可以得到第 i 层的输出:

$$O_i = \gamma_i Y_{i+1} + A_i + A'_{i+1} \tag{9}$$

其中, γ_i 是一个缩放参数.

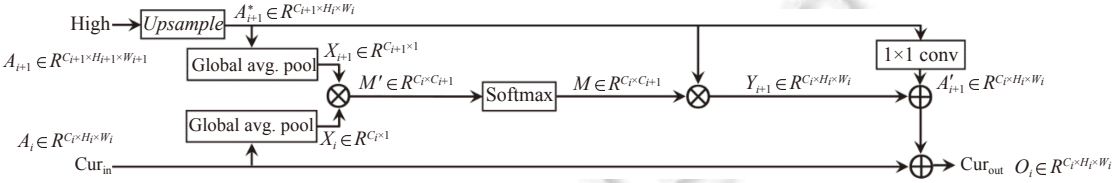


图 3 通道注意力模块的细节

2.3 增强语义信息传递的自顶向下通路

在描述自底向上通路中的融合操作符 $g(\cdot)$ 之前, 我们注意到当:

$$f(x_i, y_{i+1}) = x_i + UpSample(y_{i+1}) \tag{10}$$

自顶向下的信息通路便退化为简单的加和, 等价于 FPN[3] 的原始设计. 相对应的, 自底向上的信息通路也有简单加和的版本:

$$g(x_i, y_{i-1}) = x_i + DownSample(y_{i-1}) \tag{11}$$

作为通道注意力模块相对应的部分, 空间注意力模块的作用为自底向上通路中的融合操作符 $g(\cdot)$. 一个很直接的构造是将通道注意力模块的通道维度和空间维度进行互换, 但是这样会消耗大量的显存, 因为需要存储形状为 $H_iW_i \times H_{i+1}W_{i+1}$ 的矩阵. 我们发现通常是最低的两个特征层消耗了大部分的显存, 我们因此设计了一个更加精巧的空间注意力模块, 如图 4 所示, 其中我们引入了一个瓶颈 (bottleneck) 结构来减少显存的消耗, 同时能够达到更好的效果.

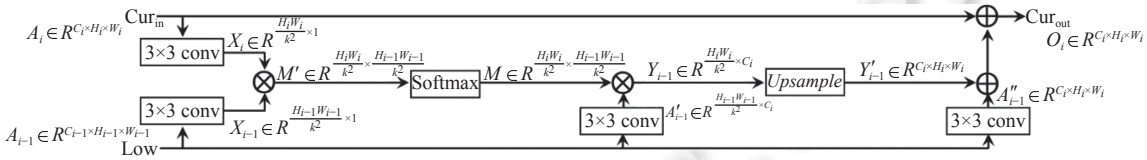


图 4 空间注意力模块的细节

对于自顶向下的信息通路, $A_i \in \mathbb{R}^{C_i \times H_i \times W_i}$ 是 P_i 的输入特征层, 低一级的特征层是 $A_{i-1} \in \mathbb{R}^{C_{i-1} \times H_{i-1} \times W_{i-1}}$. 我们使用步长为 k (默认 k=2), 输出通道数为 1 的 3×3 卷积来减小特征层的大小以及通道的数量, 得到: $X_i \in \mathbb{R}^{\frac{H_iW_i}{k^2} \times 1}$ 和 $X_{i-1} \in \mathbb{R}^{\frac{H_{i-1}W_{i-1}}{k^2} \times 1}$.

空间注意力图 M 可以通过公式 (12), 公式 (13) 得到:

$$M' = X_i \otimes X_{i-1} \tag{12}$$

$$M_{jk} = \frac{\exp(M'_{j,k})}{\sum_{l=1}^{\frac{H_{i-1}W_{i-1}}{k^2}} \exp(M'_{j,l})} \tag{13}$$

其中, M 表示空间特征图, M_{jk} 表示 M 在位置 (j, k) 上的值. 使用步长为 l 的 3×3 卷积层来得到 A'_{i-1}, 在其上应用注

意力图, 得:

$$Y_{i-1} = M \otimes A'_{i+1} \tag{14}$$

其中, Y_{i-1}表示的是空间重新加权后的第 i–1 特征层. 将 Y_{i-1} 的上采样本记作 Y'_{i-1}, 将 A_{i-1} 的下采样版本记作 A''_{i-1}, 则第 i 层的输出为:

$$O_i = \gamma_i Y'_{i-1} + A_i + A''_{i-1} \tag{15}$$

其中, γ_i 是一个缩放参数.

3 实 验

数据集: 我们在目标检测数据集 MS-COCO 2017[38] 上进行了实验, 其中包含 80 个类别. 按照常规做法 [2−3], 我们使用 trainval35 来训练模型 (包含 115k 张图像) , 并使用 minival (包含 5k 张图像) 来进行验证和消融实验. 与主流方法的比较[39−43] 通过评估服务器报告在 test-dev 上.

实现细节: 我们的实现基于 mmdetection[44] : 我们修改了 RetinaNet[2] 和 FreeAnchor[43] 以添加我们的组件, 同时将其他设置保留为 mmdetection 中的默认设置. 我们在 4 卡 NVIDIA GeForce RTX 2080 Ti 上进行了所有的实验, 每张 GPU 卡中包含 2 张图像, 因此批大小 (batch size) 为 8. 我们在开始时使用线性热身策略 (linear warmup) 进行 500 次迭代训练, 然后根据线性缩放规则 (linear scaling rule) [45] 将学习率初始化为 0.005, 动量为 0.9, 权重衰减为 0.000 1. 我们以 12 epoch 为单位训练每个模型, 学习率分别在 8 和 11 epoch 时减少了 10^{-1} . 输入图像的大小缩放为 1333×800.

3.1 在 COCO 测试集上的主要结果

表 1 显示了使用主干网络 ResNet-50 (用于 RetinaNet) 和 ResNet-101 (用于 FreeAnchor) 在 COCO test-dev 数据上评估的结果. 为了比较的公平起见, 对比的模型及数据从 mmdetection 框架中取出, 我们的实验设置和对比方法完全相同. 对于基于锚点的检测器 RetinaNet-ResNet-50, 如果将通道注意力模块仅用于自顶向下的通路 (RetinaNet-CAB) , 则平均精度 (AP) 可以提高 0.8%, 如果仅将注意力模块用于自底向上的通路 (RetinaNet-SAB) , 则可以提高 1.1%; 如果两个都使用 (标记为 SSSF) , 则可以提高 1.4%. 对于无锚检测器 FreeAnchor-ResNet-101, 我们的方法可实现最佳检测性能: AP 达到 42.1%, 比原始 FreeAnchor 提高了 1.3%. 这些结果表明, 我们的方法利用不同路径中的语义和空间信息, 可以有效地提高基于锚和无锚对象检测器的检测性能.

表 1 在 COCO test-dev 数据集上与主流检测器的性能比较 (%)

方法	主干网络	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
RetinaNet[2]	ResNet-50	36.0	56.1	38.4	20.0	39.1	45.2
FCOS[41]	ResNet-101	39.3	59.1	42.1	22.2	42.3	49.4
FoveaBox[42]	ResNet-101	38.9	58.7	41.5	21.7	42.4	48.1
FSAF[40]	ResNet-101	39.7	59.5	42.5	21.9	42.4	50.1
SABL[39]	ResNet-101	40.0	58.8	42.8	22.0	43.4	50.7
FreeAnchor[43]	ResNet-101	40.8	59.8	43.9	22.3	43.9	51.6
RetinaNet-CAB	ResNet-50	36.8	57.5	39.2	20.8	40.0	46.3
RetinaNet-SAB	ResNet-50	37.1	57.3	39.7	20.4	40.6	46.9
RetinaNet-SSSF	ResNet-50	37.4	57.8	40.0	21.0	40.8	46.8
FreeAnchor-SSSF	ResNet-101	42.1	61.3	45.1	23.7	45.5	53.2

3.2 自顶向下通路的消融实验

我们基于 RetinaNet, 以 ResNet-50 为主干, 在单尺度训练/测试下使用标准的 COCO 评估指标在 minival 数据划分上进行了消融实验. 表 2 显示了自顶向下通路的消融结果, 其中基线是 RetinaNet-ResNet-50-FPN. CAB 指通道注意力模块. ADR 代表降维之后 (after dimension reduction) , 这意味着将通道注意力模块放置在降维之后. DFN-CAB 是我们对 DFN[37] 中描述的特征融合方法的重新实现.

表 2 自顶向下的消融实验 (%)

ADR	DFN-CAB	CAB	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
—	—	—	35.6	55.4	38.1	19.3	39.6	46.5
√	—	√	36.0	56.5	38.2	**22.1**	40.2	46.6
—	√	—	36.1	56.5	38.6	20.7	40.1	47.2
—	—	√	**36.6**	**56.9**	**38.9**	20.5	**40.6**	**48.0**

通道注意力模块 (CAB) 的位置: 在描述我们的方法以及对主要结果进行实验评估时, 我们的通道注意力模块位于 1×1 旁路卷积层之前, 即在通道数量减少之前. 这种安排的动机是, 大量的通道可以在高级特征中传达更精确的语义信息, 这对于通道注意力模块增强低级特征是有利的. 表 2 中显示, 在通道数量减小后面放置通道注意力模块会使平均精度从 (36.6%) 减少到 (36.0%) , 这印证了我们的假设.

复现 DFN-CAB: 我们还重新实现了 DFN[37] 中描述的特征融合方法, 该方法将两个相邻特征层拼接起来以生成注意力图. 表 2 显示, 对于自顶向下的路径, 我们的通道注意力模块的性能优于 DFN-CAB: 36.6% vs. 36.1%. 此外, 我们的通道注意力模块比 DFN-CAB 更加轻量.

3.3 自底向上通路的消融实验

表 3 显示了自底向上通路的消融结果, 该结果是通过在自顶向下的通路中保留普通融合操作 (简单加和) 并仅对自底向上的通路进行更改而获得的.

表 3 自底向上通路的消融实验 (%)

方法	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
基线	35.6	55.4	38.1	19.3	39.6	46.5
PA	35.9	55.7	38.0	20.1	40.0	47.4
MIC	36.2	56.3	38.9	20.6	40.5	47.2
SIG	36.3	56.3	38.9	19.8	40.5	48.0
PVB	36.3	56.3	**39.0**	20.5	40.5	48.1
SAB	**36.7**	**56.9**	**39.0**	**21.3**	**41.0**	**48.5**

复现 PANet: 我们重新实现了 PANet[20] 中描述的自底向上的通路. 表 3 显示, 仅有自底向上的通路对平均精度的增益只有 0.3%. 相比之下, 添加了我们的空间注意力模块后的增益要多得多: 1.1%.

通道注意力模块的镜像实现: 如描述通道注意力的细节时所示, 可以通过交换通道维和空间维度从通道注意力模块直接构造空间注意力模块, 这在表 3 中被称为通道注意力的镜像实现 (mirror implementation of CAB, MIC). 它的平均精度为 36.2%, 比不上我们设计的空间注意力模块 (36.7%) , 并且, 这种方法消耗更多的显存.

Sigmoid 版本: 表 3 中的 SIG (Sigmoid version of SAB) 将 Sigmoid 函数应用到低级特征上来生成注意力图. 它不涉及空间相关性的任何处理. 从表 3 中显示的实验结果中, 我们可以看到我们的空间注意力模块比 SIG 表现更好, 表明空间相关性在自底向上的通路中起着至关重要的作用.

简单加和版本: 我们将先前工作使用的相邻特征简单加和操作来实现自底向上通路的方法称为 PVB (plain version of bottom-up pathway) . 它不包含任何注意力机制. 实验表明这一简单方法的性能与上述方法相似. 也就是说, 表 3 表明, 我们的空间注意力模块比所有其他自下而上的通路都具有更好的性能, 这表明空间信息融合在自底向上的通路中起着重要的作用.

瓶颈参数 k: 在自底向上的过程中, 瓶颈参数 (bottleneck parameter) k 起到减少参数和计算复杂度的作用. 它还有去除冗余信息的作用. 表 4 显示了 k 不同值时的检测性能. k=1 时的显存消耗以及参数量已经超过了我们的承受范围, 因此没有列出. 从表 4 中我们可以看到, 当 k 增加时, AP, AP_{50}, AP_{75}, AP_S 和 AP_M 持续减小. 这表明空间大小的减小通常会抹去关键信息. 但是, 大物体的平均精度在 k=4 时最高. 这意味着一定程度的减少可以帮助网络识别大物体, 但会降低中小物体的性能.

表 4 瓶颈参数 k 的消融实验

k	AP (%)	AP_{50} (%)	AP_{75} (%)	AP_S (%)	AP_M (%)	AP_L (%)
2	**37.0**	**57.4**	**39.8**	**21.4**	**41.5**	48.3
4	36.9	57.2	39.5	20.8	41.2	**48.6**
8	36.7	57.0	39.0	20.7	41.0	48.4
16	36.4	56.5	39.0	20.4	40.9	48.0

3.4 整体结构的消融实验

表 5 显示了同时使用通道注意力模块 (CAB) 和空间注意力模块 (SAB) 的整个结构的消融结果, 其中, LReLU 和 ReLU 是自上而下和自下而上这两条通路之间 (见图 2 中 top-down pathway 和 bottom-up pathway 之间各层上的连接, 用实黑线来表示) 使用的分离方法. 从表中我们可以看到, 通道注意力模块和空间注意力模块的组合可将 AP 提高 1.7%, 而没有添加任何的花哨技巧, 这证实了我们的假设, 即在自上而下的路径中传播语义信息和在自下而上的路径中传播空间信息有助于改善检测性能.

表 5 使用通道注意力模块和空间注意力模块的整个结构的消融实验 (%)

CAB	SAB	LReLU	ReLU	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
—	—	—	—	35.6	55.4	38.1	19.3	39.6	46.5
√	—	—	—	36.6	56.9	38.9	20.5	40.6	48.0
—	√	—	—	36.7	56.9	39.0	21.3	41.0	48.5
√	√	—	—	37.0	57.4	**39.8**	**21.4**	41.5	48.3
√	√	√	—	37.0	57.3	39.6	20.7	41.1	48.8
√	√	—	√	**37.3**	**57.5**	39.7	21.0	**41.6**	**49.2**

分离: 表 5 显示, 当我们对不同的路径应用不同的融合操作时, 自上而下的 CAB 和自下而上的 SAB , 每种方法对 AP 的增益都可提高约 1.0%. 但是, 当将它们简单地组合在一起时, 例如, 使用简单的顺序堆叠, 在仅应用一条路径的情况下, 性能会有约 0.3% 的微小增益. 我们可以在两个路径之间添加分离, 即: 用非线性结构来分离两个线性模块, 从而提升模型的表达能力, 减弱两部分的耦合程度, 提升模型的整体效果. 如果使用 Leaky ReLU (LReLU) [46] 进行分离, 则性能与将它们简单堆叠在一起大致相同. 当使用 ReLU[47] 时, 它有着最高的 AP: 37.3%. 在我们的方法中, ReLU 是默认的分离操作符.

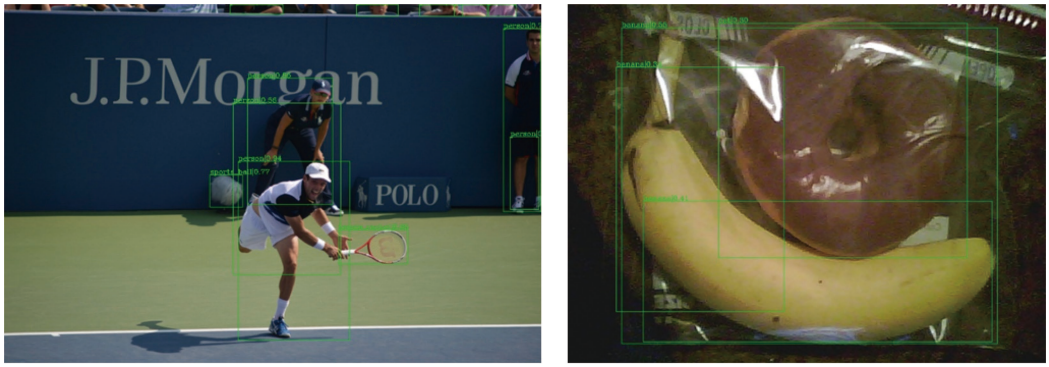
平行 (parallel) vs. 顺序 (sequential) : 两种通路有几种可能的相对位置, 可以看作是另一种分离的形式. 表 6 显示了将它们与基准线相比的性能. 平行放置 (RetinaNet-SSSF-Para) 表示以平行方式放置自上而下的路径和自底向上的路径. 主干的输出同时馈入两个路径. 顺序放置 (RetinaNet-SSSF) 为我们的方法所使用, 如图 2 所示. 从表 6 和表 5 中可以看出, 并行分隔并没有显示出比简单顺序堆叠更好的性能, AP 分别为 36.9% 和 37.0%, 并且我们的方法 (即 RetinaNet-SSSF) 实现了最高的平均精度.

表 6 平行与顺序分支的消融实验 (%)

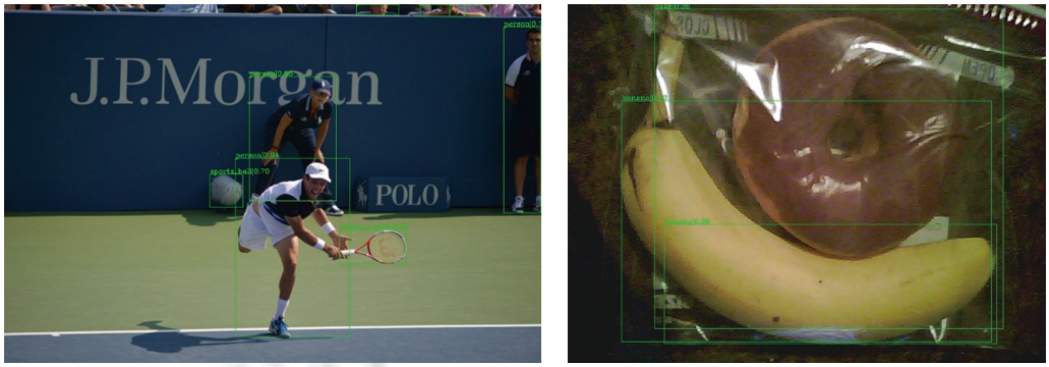
方法	AP	AP_{50}	AP_{75}
RetinaNet-FPN (基线)	35.6	55.4	38.1
RetinaNet-SSSF-Para	36.9	57.2	39.6
RetinaNet-SSSF	**37.3**	**57.5**	**39.7**

3.5 效果展示

图 5 展示了我们的方法和基线方法进行比较的检测效果图, 可以看到, 我们的方法可以降低误检率, 提高检测效果.



(a) 基线方法



(b) 所提方法

图 5　基线方法与所提方法检测效果对比展示

3.6　拓展到其他检测器

两阶段检测器: 我们的方法可以轻松运用到两阶段目标检测器上. 表 7 显示了将我们的方法集成到 Faster R-CNN 中的结果. 平均精度提高了 0.7%, 表明我们的方法在不同检测器结构上的通用性.

表 7　两阶段检测器的实验效果 (%)

方法	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
Faster R-CNN	36.3	58.1	39.2	21.2	40.2	45.8
Faster-SSSF	**37.0**	**58.9**	**39.7**	**21.4**	**40.8**	**47.3**

其他单阶段检测器: 表 8 显示了将我们的方法集成到其他单阶段目标检测器中的实验结果. 我们可以看到平均精度大概有 0.7% 到 1.6% 的提升, 这表明我们的方法对于单阶段检测器的通用性很强.

表 8　在 ATSS[48]和 FreeAnchor[43]上的消融实验 (%)

方法	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
ATSS	39.4	57.3	42.6	23.6	43.1	51.2
ATSS-SSSF	**40.1**	**58.3**	**43.2**	**23.7**	**43.9**	**51.9**
FreeAnchor	38.4	56.8	41.1	20.4	41.9	51.7
FreeAnchor-SSSF	**40.0**	**58.8**	**42.5**	**22.3**	**43.8**	**52.9**

3.7　拓展到语义分割以及实例分割任务

拓展到语义分割: 我们的融合运算符通道注意力模块可以集成到语义分割的流程中. 我们在本节中报告我们

的实验评估. 由于语义分割不需要额外的自底向上的通路, 因此仅使用通道注意力模块 (CAB) . 数据集: 我们在 3 个数据集上进行了语义分割实验: Cityscapes[49] , PASCAL VOC 2012[50] , 以及 ADE20k[51] . 实现细节: 我们的实现基于 mmsegmentation_v0.5[52] . 对于 Cityscapes, 图像的裁剪大小为 512×1024, 训练时进行 40k 次迭代, 对于 PASCAL VOC 2012 (加上增强数据后) , 采用 20k 次训练迭代, 对于 ADE20k, 图像裁剪大小则为 512×512, 训练迭代次数为 80k. 基线是具有空洞网络的基于 ResNet-50 的 FCN[25] . 其他所有内容均保留为 mmsegmentation 的原始实现, 以进行公平比较. 结果: 这 3 个数据集的实验结果显示在表 9 中. FCN-CAB 将通道注意力模块 (CAB) 集成到 FCN 中, 而 FCN-CAB-plain 用简单的加和替换了我们的通道注意力模块. 从表 9 中我们可以看到, 相对于 FCN-CAB-plain 平均 IoU, 我们的方法可以稳定增长 0.8%.

表 9　语义分割的实验结果

方法	数据集	Mean IoU (%)
FCN (基线)	Cityscapes	72.25
FCN-CAB-plain		75.70 (+3.45)
FCN-CAB		**76.49** (+4.24)
FCN (基线)	PASCAL VOC	67.08
FCN-CAB-plain		73.49 (+6.41)
FCN-CAB		**74.26** (+7.18)
FCN (基线)	ADE20k	35.94
FCN-CAB-plain		39.81 (+3.87)
FCN-CAB		**40.50** (+4.56)

拓展到实例分割: 我们的方法还可以轻松用在实例分割流程中. 为了评估其在 COCO 实例分割数据集[38] 上的性能, 我们基于 mmdetection[44] 对其进行了实现. 基线是基于 ResNet-50 的 Mask R-CNN[11] . 我们使用 0.01 作为初始学习率. 其他所有内容均保持默认状态. 表 10 和表 11 分别报告了包围框和掩膜的平均精度. 可以看到, 我们的方法提高了包围框和掩膜精度所有指标的性能.

表 10　实例分割的包围框 AP (%)

方法	AP^{bbox}	AP^{bbox}_{50}	AP^{bbox}_{75}	AP^{bbox}_{S}	AP^{bbox}_{M}	AP^{bbox}_{L}
Mask R-CNN	37.3	58.9	40.7	21.7	41.0	48.0
Mask SSSF	**38.0**	**60.0**	**41.1**	**22.0**	**42.0**	**49.4**

表 11　实例分割的掩膜 AP (%)

方法	AP^{mask}	AP^{mask}_{50}	AP^{mask}_{75}	AP^{mask}_{S}	AP^{mask}_{M}	AP^{mask}_{L}
Mask R-CNN	34.2	55.6	36.4	18.1	37.4	46.6
Mask SSSF	**34.8**	**56.7**	**37.1**	**18.2**	**38.5**	**47.8**

4 结　论

在本文中, 我们提出了一种将不同方向的路径的融合操作分开的方法: 自上而下的路径中使用了通道注意力模块 (CAB) 以有效地将语义信息传递给较低的特征层, 而空间注意力模块 (SAB) 可以自下而上地将精确的空间信息传递到顶部特征层. 该方法可以应用于基于锚和无锚的目标检测, 语义分割和实例分割. 实验结果表明, 我们的方法将 AP 的目标检测性能提高了 1.3% 以上, 对于自上而下的路径进行语义分割的融合操作, 其 AP 的平均性能提高了约 0.8%, 并且在所有指标上都提高了实例分割的包围框 AP 和掩膜 AP.

References:

[1] Ren SQ, He KM, Girshick RB, Sun J. Faster R-CNN: Towards real-time object detection with region proposal networks. arXiv:1506.01497, 2015.

[2] Lin TY, Goyal P, Girshick R, He KM, Dollár P. Focal loss for dense object detection. In: Proc. of the 2017 IEEE Int'l Conf. on Computer Vision. Venice: IEEE, 2017. 2980–2988. [doi: 10.1109/ICCV.2017.324]

[3] Lin TY, Dollár P, Girshick R, He KM, Hariharan B, Belongie S. Feature pyramid networks for object detection. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 936–944. [doi: 10.1109/CVPR.2017.106]

[4] Redmon J, Farhadi A. YOLO9000: Better, faster, stronger. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 6517–6525. [doi: 10.1109/CVPR.2017.690]

[5] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, Berg AC. SSD: Single shot multibox detector. In: Proc. of the 14th European Conf. on Computer Vision. Amsterdam: Springer, 2016. 21–37. [doi: 10.1007/978-3-319-46448-0_2]

[6] Zhao HS, Shi JP, Qi XJ, Wang XG, Jia JY. Pyramid scene parsing network. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 6230–6239. [doi: 10.1109/CVPR.2017.660]

[7] Chen LC, Papandreou G, Schroff F, Adam H. Rethinking atrous convolution for semantic image segmentation. arXiv:1706.05587, 2017.

[8] Yuan YH, Huang L, Guo JY, Zhang C, Chen XL, Wang JD. OCNet: Object context network for scene parsing. arXiv:1809.00916, 2021.

[9] Fu J, Liu J, Tian HJ, Li Y, Bao YJ, Fang ZW, Lu HQ. Dual attention network for scene segmentation. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 3141–3149. [doi: 10.1109/CVPR.2019.00326]

[10] Huang ZL, Wang XG, Huang LC, Huang C, Wei YC, Liu WY. CCNet: Criss-cross attention for semantic segmentation. In: Proc. of the 2019 IEEE/CVF Int'l Conf. on Computer Vision. Seoul: IEEE, 2019. 603–612. [doi: 10.1109/ICCV.2019.00069]

[11] He KM, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proc. of the 2017 IEEE Int'l Conf. on Computer Vision. Venice: IEEE, 2017. 2980–2988. [doi: 10.1109/ICCV.2017.322]

[12] Wang ZY, Yuan C, Li JC. Instance segmentation with separable convolutions and multi-level features. Ruan Jian Xue Bao/Journal of Software, 2019, 30(4): 954–961 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5667.htm [doi: 10.13328/j.cnki.jos.005667]

[13] Wang XL, Kong T, Shen CH, Jiang YN, Li L. SOLO: Segmenting objects by locations. In: Proc. of the 16th European Conf. on Computer Vision. Glasgow: Springer, 2020. 649–665. [doi: 10.1007/978-3-030-58523-5_38]

[14] Lee Y, Park J. CenterMask: Real-time anchor-free instance segmentation. In: Proc. of the 2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 13903–13912. [doi: 10.1109/CVPR42600.2020.01392]

[15] Zhao QJ, Sheng T, Wang YT, Tang Z, Chen Y, Cai L, Ling HB. M2Det: A single-shot object detector based on multi-level feature pyramid network. In: Proc. of the 2019 AAAI Conf. on Artificial Intelligence, 2019. 9259–9266. [doi: 10.1609/aaai.v33i01.33019259]

[16] Li ZX, Zhou FQ. FSSD: Feature fusion single shot multibox detector. arXiv:1712.00960, 2018.

[17] Tan MX, Pang RM, Le QV. EfficientDet: Scalable and efficient object detection. In: Proc. of the 2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 10778–10787. [doi: 10.1109/CVPR42600.2020.01079]

[18] Chen K, Cao YH, Loy CC, Lin DH, Feichtenhofer C. Feature pyramid grids. arXiv:2004.03580, 2020.

[19] Ghiasi G, Lin TY, Le QV. NAS-FPN: Learning scalable feature pyramid architecture for object detection. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 7029–7038. [doi: 10.1109/CVPR.2019.00720]

[20] Liu S, Qi L, Qin HF, Shi JP, Jia JY. Path aggregation network for instance segmentation. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 8759–8768. [doi: 10.1109/CVPR.2018.00913]

[21] Dai JF, Li Y, He KM, Sun J. R-FCN: Object detection via region-based fully convolutional networks. In: Proc. of the 30th Int'l Conf. on Neural Information Processing Systems. Barcelona: Curran Associates Inc., 2016. 379–387.

[22] Li ZM, Peng C, Yu G, Zhang XY, Deng YD, Sun J. Light-head R-CNN: In defense of two-stage object detector. arXiv:1711.07264, 2017.

[23] Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 779–788. [doi: 10.1109/CVPR.2016.91]

[24] Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv:1804.02767, 2018.

[25] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. In: Proc. of the 2015 IEEE Conf. on Computer Vision and Pattern Recognition. Boston: IEEE, 2015. 3431–3440. [doi: 10.1109/CVPR.2015.7298965]

[26] Hariharan B, Arbeláez P, Girshick R, Malik J. Hypercolumns for object segmentation and fine-grained localization. In: Proc. of the 2015

IEEE Conf. on Computer Vision and Pattern Recognition. Boston: IEEE, 2015. 447–456. [doi: 10.1109/CVPR.2015.7298642]

[27] Cai ZW, Fan QF, Feris RS, Vasconcelos N. A unified multi-scale deep convolutional neural network for fast object detection. In: Proc. of the 14th European Conf. on Computer Vision. Amsterdam: Springer, 2016. 354–370. [doi: 10.1007/978-3-319-46493-0_22]

[28] Kong T, Sun FC, Yao AB, Liu HP, Lu M, Chen YR. RON: Reverse connection with objectness prior networks for object detection. In: Proc. of the the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 5244–5252. [doi: 10.1109/CVPR.2017.557]

[29] Huang G, Chen DL, Li TH, Wu F, van der Maaten L, Weinberger KQ. Multi-scale dense networks for resource efficient image classification. arXiv:1703.09844, 2017.

[30] Sun K, Xiao B, Liu D, Wang JD. Deep high-resolution representation learning for human pose estimation. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 5686–5696. [doi: 10.1109/CVPR.2019.00584]

[31] Qiao SY, Chen LC, Yuille A. DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution. In: Proc. of the 2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021. 10208–10219. [doi: 10.1109/CVPR46437.2021.01008]

[32] Yu F, Wang DQ, Shelhamer E, Darrell T. Deep layer aggregation. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 2403–2412. [doi: 10.1109/CVPR.2018.00255]

[33] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv:1409.0473, 2014.

[34] Lin ZH, Feng MW, dos Santos CN, Yu M, Xiang B, Zhou BW, Bengio Y. A structured self-attentive sentence embedding. arXiv:1703.03130, 2017.

[35] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I. Attention is all you need. arXiv:1706.03762. 2017.

[36] Wang XL, Girshick R, Gupta A, He KM. Non-local neural networks. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 7794–7803. [doi: 10.1109/CVPR.2018.00813]

[37] Yu CQ, Wang JB, Peng C, Gao CX, Yu G, Sang N. Learning a discriminative feature network for semantic segmentation. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 1857–1866. [doi: 10.1109/CVPR.2018.00199]

[38] Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. Microsoft COCO: Common objects in context. In: Proc. of the 13th European Conf. on Computer Vision. Zurich: Springer, 2014. 740–755. [doi: 10.1007/978-3-319-10602-1_48]

[39] Wang JQ, Zhang WW, Cao YH, Chen K, Pang JM, Gong T, Shi JP, Loy CC, Lin DH. Side-aware boundary localization for more precise object detection. In: Proc. of the 16th European Conf. on Computer Vision. Glasgow: Springer, 2020. 403–419. [doi: 10.1007/978-3-030-58548-8_24]

[40] Zhu CC, He YH, Savvides M. Feature selective anchor-free module for single-shot object detection. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019. 840–849. [doi: 10.1109/CVPR.2019.00093]

[41] Tian Z, Shen CH, Chen H, He T. FCOS: Fully convolutional one-stage object detection. In: Proc. of the 2019 IEEE/CVF Int'l Conf. on Computer Vision. Seoul: IEEE, 2019. 9626–9635. [doi: 10.1109/ICCV.2019.00972]

[42] Kong T, Sun FC, Liu HP, Jiang YN, Li L, Shi JB. FoveaBox: Beyound anchor-based object detection. IEEE Trans. on Image Processing, 2020, 29: 7389–7398. [doi: 10.1109/TIP.2020.3002345]

[43] Zhang XS, Wan F, Liu C, Ji XY, Ye QX. FreeAnchor: learning to match anchors for visual object detection. arXiv:1909.02466, 2019.

[44] Chen K, Wang JQ, Pang JM, *et al*. MMDetection: Open MMLab detection toolbox and benchmark. arXiv:1906.07155, 2019.

[45] Goyal P, Dollár P, Girshick R, Noordhuis P, Wesolowski L, Kyrola A, Tulloch A, Jia YQ, He KM. Accurate, large minibatch SGD: Training ImageNet in 1 hour. arXiv:1706.02677, 2018.

[46] Xu B, Wang NY, Chen TQ, Li M. Empirical evaluation of rectified activations in convolutional network. arXiv:1505.00853, 2015.

[47] Nair V, Hinton GE. Rectified linear units improve restricted boltzmann machines. In: Proc. of the 27th Int'l Conf. on Machine Learning. Haifa: Omnipress, 2010. 807–814.

[48] Zhang SF, Chi C, Yao YQ, Lei Z, Li SZ. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: Proc. of the 2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020. 9756–9765. [doi: 10.1109/CVPR42600.2020.00978]

[49] Cordts M, Omran M, Ramos S, Rehfeld T, Enzweiler M, Benenson R, Franke U, Roth S, Schiele B. The cityscapes dataset for semantic

urban scene understanding. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 3213–3223. [doi: 10.1109/CVPR.2016.350]

[50] Everingham M, van Gool L, Williams CKI, Winn J, Zisserman A. The PASCAL visual object classes (VOC) challenge. Int'l Journal of Computer Vision, 2010, 88(2): 303–338. [doi: 10.1007/s11263-009-0275-4]

[51] Zhou BL, Zhao H, Puig X, Fidler S, Barriuso A, Torralba A. Scene parsing through ADE20K dataset. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 5122–5130. [doi: 10.1109/CVPR.2017.544]

[52] Xu J, Chen K. MMSegmentation. 2020. https://github.com/open-mmlab/mmsegmentation

附中文参考文献:

[12] 王子愉, 袁春, 黎健成. 利用可分离卷积和多级特征的实例分割. 软件学报, 2019, 30(4): 954–961. http://www.jos.org.cn/1000-9825/5667.htm [doi: 10.13328/j.cnki.jos.005667]

郭琪周(1997－), 男, 硕士生, 主要研究领域为计算机视觉.

袁春(1969－), 男, 博士, 副研究员, 博士生导师, CCF 杰出会员, 主要研究领域为机器学习, 计算机视觉.