

文章编号: 1002-1582(2022)01-0080-06

# 改进 YOLO 轻量化网络的行人检测算法

常青<sup>1</sup>, 韩文<sup>2</sup>, 王清华<sup>1\*</sup>, 李振华<sup>1</sup>

(1. 南京理工大学 理学院, 江苏 南京 210094)

(2. 南京林业大学 信息科学技术学院, 江苏 南京 210037)

**摘 要:** 针对当前行人检测方法计算量大、检测精度低的问题, 基于 YOLOv4-tiny 提出一种改进的行人检测算法。引入通道注意力和空间注意力模块(CBAM)至 CSPDarknet53-tiny 网络中, 通过学习图像的位置信息和通道信息得到更加丰富的特征; 在骨干网络 CSPDarknet53-tiny 之后引入空间金字塔池化模块, 能够极大地增加感受野, 分离出最显著的上下文特征; 使用 CIoU 损失函数对算法的多任务联合损失进行了优化。实验分别使用 INRIA 和 WiderPerson 中的训练集作为训练模型, INRIA 和 WiderPerson 中的测试集分别作为测试集来验证模型。实验表明, 对比 YOLOv4-tiny 目标检测模型, 改进后的网络在 INRIA 测试集中精确度、召回率和平均精度值分别提升了 6.23%、3.15% 和 6.12%; 改进后的网络在 WiderPerson 测试集中精确度、召回率和平均精度值分别提升了 3.65%、3.28% 和 4.41%。改进后的模型在几乎不影响检测实时性的前提下, 更易于行人特征提取, 提高了检测精度。

**关 键 词:** 深度学习; 行人检测; YOLOv4-tiny; 注意力机制

中图分类号: TP391.41 文献标识码: A

DOI:10.13741/j.cnki.11-1879/o4.2022.01.012

## Pedestrian detection algorithm based on improved YOLO lightweight network

CHANG Qing<sup>1</sup>, HAN Wen<sup>2</sup>, WANG Qinghua<sup>1\*</sup>, LI Zhenhua<sup>1</sup>

(1. School of Science, Nanjing University of Science and Technology, Nanjing 210094, China)

(2. School of Information Science and Technology, Nanjing Forestry University, Nanjing 210037, China)

**Abstract:** As the current pedestrian detection method has the problems of large computation and low detection accuracy, an improved pedestrian detection method based on YOLOv4-Tiny was proposed. This method introduces Convolutional Block Attention Module into CSPDarknet53-tiny network to get richer features by learning the position information and channel information of the image, adds the spatial pyramid pooling module following CSPDarknet53-tiny, which can greatly increase the receptive field and isolate the most significant context features, and uses CIoU loss function to optimize the combined loss of multiple tasks. In the experiment, the training set in INRIA and WiderPerson are used as the training model, and the test set in INRIA and WiderPerson are used as the test set to verify the model. Compared with YOLOv4-Tiny, the precision, recall and average accuracy of the improved YOLOv4-Tiny network in INRIA test set are increased by 6.23%, 3.15% and 6.12%, respectively, and the improved network increased the precision, recall, and average accuracy in the WiderPerson test set by 3.65%, 3.28%, and 4.41%, respectively. It is found that this improved model can extract pedestrian features more easily and improve the detection accuracy on the premise that the real-time detection is hardly affected.

**Key words:** deep learning; pedestrian detection; YOLOv4-tiny; attention model

收稿日期: 2021-08-24; 收到修改稿日期: 2021-10-15

作者简介: 常青(1995—), 女, 硕士研究生, 从事图像处理研究。cqhc1105@163.com

王清华(1976—), 男, 副教授, 博士, 从事光学检测与图像处理研究。

\* 通讯作者: qhwang@njut.edu.cn

## 0 引言

行人检测是指计算机设备从图像中自动定位出待检测行人目标,是计算视觉研究的核心内容之一,也是自动驾驶、行为分析和人机交互等研究的基础和前提,具有重要的研究价值和广泛的应用场景。

传统的行人检测方法一般是使用手工提取特征,然后采用分类器来实现行人检测。主要采用边缘特征、形状特征、统计特征或者变换特征等图像的静态特征来描述行人,其中代表性的特征包括 Haar-like 特征<sup>[1]</sup>、HOG 特征<sup>[2]</sup>、Edgelet 特征<sup>[3]</sup>、Shapelet<sup>[4]</sup> 特征和轮廓模板特征<sup>[5,6]</sup>等,分类器一般采用 SVM 或增强学习等。由于手工设计的特征描述子的特征单一,具有检测速度快等优点,但是只可从梯度或纹理等某一方面来描述,判断力较差。

基于深度学习的目标检测算法主要分为两类,一类是以 RCNN<sup>[7]</sup>、Fast RCNN<sup>[8]</sup> 和 Faster RCNN<sup>[9]</sup> 为代表的基于候选框<sup>[10]</sup>的方法;另一类是以 SSD<sup>[11]</sup> 和 YOLO<sup>[12]</sup> 为代表的基于回归<sup>[13]</sup>的方法。

行人检测是目标检测的子问题,基于深度学习的行人检测模型不需要手动设计特征,可以通过卷积神经网络<sup>[14]</sup>结构从大量的样本中选择出判别力较强的特征,从而生成更加可靠的检测结果。

本文提出了一种基于 YOLOv4-tiny 轻量化网络改进的行人检测算法,具体来说,本文的主要贡献如下:

(1) 为了进一步丰富深度特征,引入了空间金字塔池化模块<sup>[15]</sup>,能够在同一卷积层中收集并且拼接不同尺度的局部特征;

(2) 基于 CBAM 注意力机制<sup>[16]</sup>,构建了更加高效准确的目标检测器;

(3) 使用 CIoU 损失函数,对算法的多任务联合损失进行了优化,加速网络的收敛过程。

## 1 YOLOv4-tiny 网络简介

YOLOv4-tiny 是 YOLOv4 的简化版,降低了对硬件的要求,提高了检测速度,但是检测精度有所下降。YOLOv4-tiny 网络结构如图 1 所示。

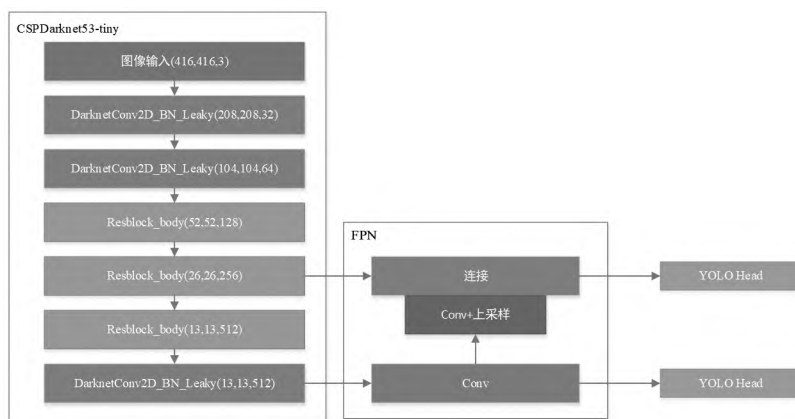


图 1 YOLOv4-tiny 网络结构

YOLOv4-tiny 采用 CSPDarknet53-tiny 作为主干提取网络,主要包括卷积、残差块堆叠和下采样三个过程。卷积过程是由卷积层(Convolution)、批量标准化(BatchNormalization)和激活函数(LeakyReLU)三部分构成;下采样通过最大池化(MaxPooling)操作完成;残差块堆叠采用了 CSP<sup>[17]</sup> (Cross Stage Partial Network)结构,利用该部分可获得两个有效特征层,传入到 FPN<sup>[18]</sup> (Feature Pyramid Networks)中进行特征融合;最后 YOLO Head 利用获得到的特征进行预测。

## 2 改进的 YOLOv4-tiny 算法

### 2.1 空间金字塔池化结构的引入

为了更全面地学习多尺度目标特征,本文在骨

干网络 CSPDarknet53-tiny 的第二个输出特征层,在送入 FPN 与第一个特征输出层相互融合之前,引入了空间金字塔结构池化(Spatial Pyramid Pooling, SPP)处理。其中, SPP 结构并行使用了三个大小分别为  $3 \times 3$ 、 $7 \times 7$ 、 $11 \times 11$  的池化层进行最大池化操作,分别经过卷积核大小为  $1 \times 1$ 、 $3 \times 3$ 、 $1 \times 1$  的卷积处理操作,对 SPP 模块的输出通道维度进行降维至 256。相比较 CSPDarknet53-tiny 的第二个输出特征层直接送入 FPN 中进行上采样的策略, SPP 模块的引入大幅增强网络的感受野,对输入特征层的信息利用更加充分。本文使用的 SPP 结构如图 2 所示。

### 2.2 注意力机制 CBAM 的引入

CBAM 可以分为两部分,一部分是通道注意力

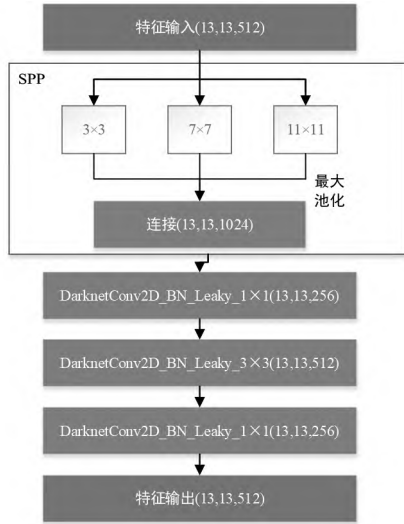


图 2 本文使用的 SPP 模块结构图

块(Channel Attention Module, CAM),和 SE 注意力机制<sup>[19]</sup>类似,另一部分为空间注意力块(Spatial Attention Module, SAM)。

在实际的行人检测识别过程中,由于目标中包含了大量混淆背景信息,现有的网络往往无法剔除该无效信息,会导致训练后得到的权重中包含大量背景信息,影响到检测性能。因此,在 3.1 节已经引入空间金字塔结构基础上,本文再次引入了 CBAM 注意力机制来关注重点特征并且抑制非必要特征,增加表现力。CBAM 模块结构如图 3 所示。

本文在 CSP 残差结构输出通道后添加 CBAM 注意力机制,其结构如图 4 所示。

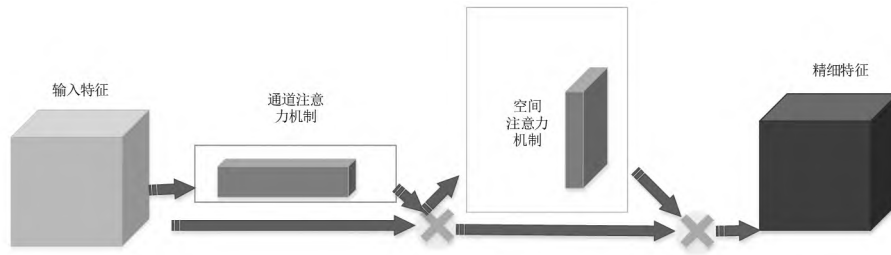


图 3 CBAM 注意力模块结构

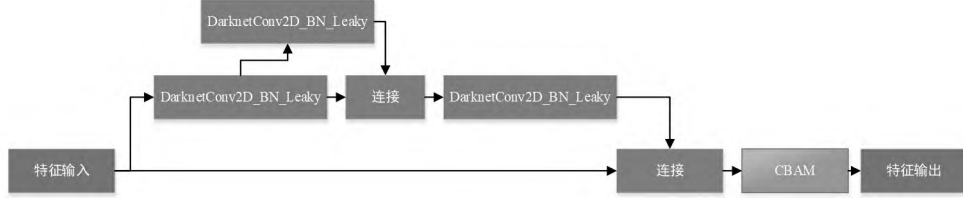


图 4 引入 CBAM 注意力机制的 CSP 残差结构图

### 2.3 损失函数的改进

目标检测算法的损失函数是由边界框定位损失  $L_{loc}$ 、预测类别损失  $L_{cls}$  和预测置信度损失  $L_{conf}$  三部分组成的。本文算法的损失函数表示为  $L = L_{loc} + L_{cls} + L_{conf}$ 。

其中,边界框定位损失  $L_{loc}$  表示为

$$L_{loc} = 1 - \text{IoU} = 1 - \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \quad (1)$$

式中,  $B = (x, y, w, h)$  表示预测框的位置,  $B^{gt} = (x^{gt}, y^{gt}, w^{gt}, h^{gt})$  表示真实框的位置。

相比较 IoU 损失函数,CIoU 损失函数将目标与 anchor 之间的距离、重叠率、尺度以及惩罚项都考虑进去,使得目标框回归变得更加稳定,不会像在 IoU 训练过程中出现发散等问题,同时 CIoU 损失函数对尺度变换敏感,收敛速度比 IoU 损失函数快。此时对于边界框定位损失  $L_{loc}$  部分,将 IoU 损

失函数替换为 CIoU 损失函数,可表示为

$$\text{CIoU} = \text{IoU} - \frac{\rho^2(B, B^{gt})}{c^2} - \beta \quad (2)$$

式中,  $\rho^2(B, B^{gt})$  表示预测框和真实框中心点之间的欧氏距离;  $\beta$  是权重参数;  $\nu$  是衡量长宽比一致性的参数;  $c$  代表同时包含预测框和真实框的最小矩阵区域的对角线距离,如图 5 所示,图中,  $d = \rho^2(B, B^{gt})$ 。 $\rho$  和  $\nu$  的计算公式可表示为

$$\begin{cases} \beta = \frac{\nu}{1 - \text{IoU} + \nu} \\ \nu = \frac{4}{\pi^2} \left( \arctan \frac{w^{gt}}{h^{gt}} - \arctan \frac{w}{h} \right)^2 \end{cases} \quad (3)$$

式中,  $w$  和  $w^{gt}$ 、 $h$  和  $h^{gt}$  分别代表预测框和真实框的宽和高。对应的 CIoU 的损失函数  $L_{\text{CIoU}}$  可表为

$$\begin{aligned} L_{\text{CIoU}} &= 1 - \text{CIoU} \\ &= 1 - \text{IoU} + \frac{\rho^2(B, B^{gt})}{c^2} + \beta \end{aligned} \quad (4)$$

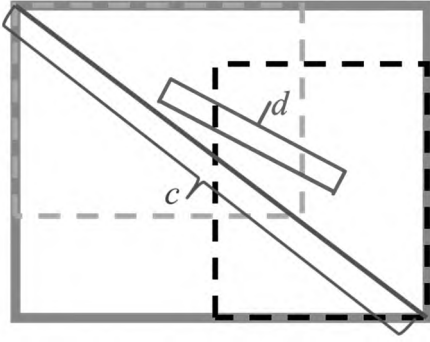


图 5 CIoU 计算示意图

预测类别损失  $L_{cls}$  用于衡量预测框与真实框之间的类别误差,采用交叉熵损失函数进行衡量,可表示为

$$L_{cls} = - \sum_{i=0}^{K \times K} I_{ij}^{obj} \sum_{c \in \text{classes}} [\hat{p}(z) \log(p_i(z)) + (1 - \hat{p}(z)) \log(1 - p_i(z))] \quad (5)$$

式中,  $K \times K$  ( $13 \times 13$  或  $26 \times 26$ ) 表示特征图上的网格数量;  $I_{ij}^{obj}$  表示此处的 box 有目标则为 1, 否则为 0;  $z$  表示某一个目标类别;  $p_i(z)$  和  $\hat{p}(z)$  分别表示先验框中预测的目标对象的属于  $z$  类别的预测概率和真实概率。预测置信度损失  $L_{conf}$  同预测类别损失  $L_{cls}$  一样均采用交叉熵损失函数进行衡量,可表示为

$$L_{conf} = \sum_{i=0}^{K \times K} \sum_{j=0}^M I_{ij}^{obj} [\hat{C}_i \log(C_i) + (1 - \hat{C}_i) \log(1 - C_i)] - \sum_{i=0}^{K \times K} \sum_{j=0}^M I_{ij}^{noobj} [\hat{C}_i \log(C_i) + (1 - \hat{C}_i) \log(1 - C_i)] \quad (6)$$

式中,  $M$  表示先验框的个数;  $\hat{C}_i$  和  $C_i$  代表置信度的真实值和预测值;  $I_{ij}^{noobj}$  表示此处的 box 没有目标则为 1, 否则为 0, 其余的参数与式(5)含义一致。

以下基于 YOLOv4-tiny 的实验中损失函数均以上述更新后的损失函数替代进行。

### 3 算法的实验结果分析

为了验证改进的 YOLOv4-tiny 网络在行人检测中的有效性和合理性,本实验使用 YOLOv4-tiny 和改进 YOLOv4-tiny 算法以及其他主流算法在相同的数据集进行训练和测试比较各算法的检测精度;同时在同一设备上对 YOLOv4-tiny 以及改进的 YOLOv4-tiny 算法进行检测速度测试,比较两者的检测速度性能。本实验是在 Windows 10 操作系统;内存为 16G;GPU 为 NVIDIA GeForce 1650;学习框架为 tensorflow-gpu 1.13.1, keras 2.1.5, cuda10.0 和 cudnn7.4.1.5。

#### 3.1 数据集

本实验使用的是 INRIA 和 WiderPerson 数据集。在 INRIA 数据集中,训练集中有 614 张图片,包含 1237 个标注行人数目,测试集中有 288 张图片,共 599 个标注行人数目,其中行人的类名定义为 person。WiderPerson 数据集是一种多样化的密集行人检测数据集,其图片具有丰富的前景和背景,同时具有大量丰富的行人高度遮挡的人群场景。由于 WiderPerson 原数据集中的测试数据及其真实框标签未公布,实验中将原 8000 张图片训练集图像根据 9:1 的比例划分为训练集和验证集,将原验证集作为测试集,其中行人的类名定义为 pedestrians。两种实验数据集中包含了不同姿态、行人之间的相互遮挡以及不同光照背景下的行人,有效的提升了行人检测的鲁棒性。

#### 3.2 实验结果

##### 3.2.1 评价指标

为了准确评估模型检测效果,本文使用精准率  $P$  (Precision) 和召回率  $R$  (Recall) 和平均精度均值 mAP (mean Average Precision) 作为评估检测精度指标,计算公式如下式所示,同时使用每秒帧数 FPS (Frames Per Second) 作为检测速度评估的指标。

$$P = \frac{TP}{TP + FP} \quad (7)$$

$$R = \frac{TP}{TP + FN} \quad (8)$$

$$mAP = \frac{\sum_{i=1}^k AP_i}{k} \quad (9)$$

式(7)~(8)中, TP (True Positive) 表示正类判定为正类; FP (False Positive) 表示负类判定为正类; FN (False Negative) 表示正类判定为负类;式(9)中,  $k$  为类别,  $AP$  (Average Precision) 的计算公式可表示为

$$AP = \int_0^1 P(R) dR \quad (10)$$

##### 3.2.2 检测精度和检测速度性能对比实验

对比 YOLOv4-tiny, 基于不同优化程度 YOLOv4-tiny 网络对 INRIA 数据集和预处理好的 WiderPerson 数据集分别进行训练和测试,实验结果如表 1 和表 2 所示。其中表 1 中的  $P$ 、 $R$  分别指 INRIA 数据集中 person 类检测的准确率和召回率,表 2 中的  $P$ 、 $R$  分别指 WiderPerson 数据集中 pedestrians 类检测的准确率和召回率。

表 1 INRIA 数据集实验结果

训练数据集	测试数据集	检测算法	评价指标			
			$P/\%$	$R/\%$	mAP/ $\%$	FPS/(帧/秒)
INRIA 训练集	INRIA 测试集	YOLOv4-tiny	90.93	90.64	90.72	83.99
		YOLOv4-tiny+SSP	94.24	88.19	96.58	77.13
		YOLOv4-tiny+CBAM	93.23	91.40	95.22	86.51
		本文算法	97.16	93.79	96.84	81.04

在 INRIA 数据集的实验中,与 YOLOv4-tiny 算法相比较,不同程度改进的 YOLOv4-tiny 算法在准确率、召回率和平均精度值都略有提升,本文算法

在准确率、召回率和平均精度均值分别提高了 6.23%、3.15% 和 6.12%,在此同时,行人检测的实时性也几乎没有很大的影响。

表 2 WiderPerson 数据集实验结果

训练数据集	测试数据集	检测算法	评价指标			
			$P/\%$	$R/\%$	mAP/ $\%$	FPS/(帧/秒)
WiderPerson 训练集	WiderPerson 测试集	YOLOv4-tiny	84.34	57.87	51.32	83.91
		YOLOv4-tiny+SSP	86.12	59.12	53.68	72.47
		YOLOv4-tiny+CBAM	86.79	59.39	53.95	85.22
		本文算法	87.99	61.15	55.73	81.46

在 WiderPerson 数据集的实验中,与 YOLOv4-tiny 算法相比较,不同程度改进的 YOLOv4-tiny 算法对 pedestrians 类检测的准确率、召回率和平均精度均值都略有提升,本文算法在 pedestrians 类检测的准确率、召回率和平均精度均值分别提高了 3.65%、5.26% 和 4.41%,检测速度虽然略有下降,但模型整体的精确度得到了显著提升。

在以上的两组实验中,本文算法在准确率、召回率和平均精度三个指标上均优于 YOLOv4-tiny 算法。以上两组实验的部分检测结果对比如下图所示,其中检测场景中包含背景光照强弱不一情形,行人之间相互遮挡以及行人姿态多样化的情形,其中图 6 为 YOLOv4-tiny 和本文算法在 INRIA 测试集测试结果对比;图 7 为 YOLOv4-tiny 和本文算法在 WiderPerson 测试集测试结果。

YOLOv4-tiny 的检测效果不佳。图 6(a)中存在对遮挡行人的漏检以及对图片中低分辨率远视角行人的漏检,图 6(b)中上述存在于图 6(a)中的问题基本予以解决。本文算法的检测效果显著优于 YOLOv4-tiny 的检测效果。



(a) YOLOv4-tiny 在 WiderPerson 数据集检测结果



(b) 本文算法在 WiderPerson 数据集检测结果

图 7 WiderPerson 数据集检测结果

在图 7 中,检测图片的场景光照不良,此时在人物相互遮挡的情形下,加大了检测难度,图 7(a)中,当存在严重的遮挡问题和光照不佳时,YOLOv4-tiny 网络的检测中均出现了漏检问题,对比本文改进的算法,后者的检测效果明显优于前者,后者模型具有较强的鲁棒性。

### 3.2.3 与其他主流算法的检测精度和检测速度对比实验

本文还将 YOLOv3<sup>[20]</sup> 和 YOLOv4<sup>[21]</sup> 以及 YOLOv5<sup>[22]</sup> 在数据集中进行了比较实验,使用 INRIA 训练集分别对各网络进行训练,再使用 INRIA 测试集进行测试,实验结果如表 3 所示。



(a) YOLOv4-tiny 在 INRIA 数据集检测结果



(b) 本文算法在 INRIA 数据集检测结果

图 6 INRIA 数据集检测结果

图 6 可知,当存在行人之间相互遮挡,行人姿态多样化或存在人物在图像中的分辨率低下时,

表 3 不同算法检测精度对比

训练数据集	测试数据集	检测算法	mAP/%	FPS(帧/秒)
INRIA 训练集	INRIA 测试集	YOLOv3	92.17	24.30
		YOLOv4	94.09	33.39
		YOLOv5	96.26	71.43
		YOLOv4-tiny	90.72	83.99
		本文算法	96.84	81.04

由表中结果可以看出,本文改进的 YOLOv4-tiny 算法在行人检测中的平均精度均值均高于当前主流的 YOLOv3 和 YOLOv4 以及 YOLOv5 算法。

4 结 论

基于 YOLOv4-tiny 目标检测算法,本文改进的算法通过引入了空间金字塔池化模块、添加注意力机制以及优化损失函数,通过在 INRIA 数据集和 WiderPerson 数据集下的实验表明,在对检测实时性几乎没有影响前提下,改进后的算法大大的提高了行人检测精度。因此本文改进的算法适用于多场景下的行人检测,具有较强的鲁棒性。

参考文献:

[1] Lienhart R, Maydt J. An extended set of haar-like features for rapid object detection[C]// 2002 IEEE International Conference on Image Processing. New York, USA;IEEE,2002;900—903.

[2] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]// 2005 IEEE Conference on Computer Vision and Pattern Recognition. New York, USA;IEEE,2005;886—893.

[3] Wu B, Nevatia R. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors[C]// 2005 IEEE International Conference on Computer Vision. Beijing, China;IEEE,2005;90—97.

[4] Ye L, Keogh E. Time series shapelets: a new primitive for data mining[C]// 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris, France;IEEE,2009; 947—956.

[5] Gavrilu D M. A bayesian, exemplar-based approach to hierarchical shape matching[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2007,29(8):1408—1421.

[6] Gao W, Ai H, Lao S. Adaptive contour features in oriented granular space for human detection and segmentation[C]// 2009 IEEE Conference on Computer Vision and Pattern Recognition. New York, USA;IEEE,2009;1786—1793.

[7] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]// 2014 IEEE Conference on Computer Vision and Pattern Recognition. New York, USA;IEEE,2014;580—587.

[8] Girshick R. Fast R-CNN[C]// 2015 IEEE International Conference on Computer Vision. Santiago, Chile;IEEE,2015;1440—1448.

[9] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017,39(6):1137—1149.

[10] 赵永强, 饶元, 董世鹏, 等. 深度学习目标检测方法综述[J].

中国图象图形学报,2020,25(04):629—654.

Zhao Yongqinang, Yao Yuan, Dong Shipeng, et al. A review of deep learning object detection methods[J]. Journal of Image and Graphics,2020,25(04):629—654.

[11] Liu W, Anguelov D, Erhan D, et al. SSD: single shot multibox detector[C]// 14th European Conference on Computer Vision. Amsterdam, Netherlands;Springer-Verlag,2016;21—37.

[12] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition. Seattle, WA: IEEE,2016;779—788.

[13] Song Y, Fu Z. Uncertain multivariable regression model[J]. Soft Computing,2018,22(17):5861—5866.

[14] 周飞燕, 金林鹏, 董军. 卷积神经网络研究综述[J]. 计算机学报,2017,40(06):1229—1251.

Zhou Feuyan, Jin Linpeng, Dong Jun. A review of convolutional neural networks[J]. Chinese Journal of Computers, 2017,40(06):1229—1251.

[15] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2015,37 (9):1904—1916.

[16] Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module[C]// 15th European Conference on Computer Vision. Munich, German; Springer-Verlag,2018;3—19.

[17] Wang C Y, Liao H Y M, Wu Y H, et al. CSPNet: A new backbone that can enhance learning capability of CNN[C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Seattle, WA: IEEE,2020;1571—1580.

[18] Lin T Y, Dollár P, Girshick R, et al. Feature pyramid networks for object detection[C]// IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI; IEEE, 2017: 2117—2125.

[19] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2020,42(08):2011—2023.

[20] Redmon J, Farhadi A. Yolov3: An incremental improvement [EB/OL]. (2018-08-08) [2021-08-17]. <https://arxiv.org/abs/1804.02767>.

[21] Bochkovskiy A, Wang C Y, Liao H Y M. Yolov4: Optimal speed and accuracy of object detection[EB/OL]. (2020-08-23) [2021-08-17]. <https://arxiv.org/abs/2004.10934>.

[22] Wen H K, Dai F Z, Yuan Y S. A Study of YOLO algorithm for target detection[C]// 26th International Conference on Artificial Life and Robotics. ELECTR Network, Japan; OITA, 2021;622—625.