



文章编号:1001-9081(2019)01-0186-06

DOI:10.11772/j.issn.1001-9081.2018061351

## 基于级联网络的行人检测方法

陈光喜<sup>1\*</sup>, 王佳鑫<sup>1</sup>, 黄勇<sup>2</sup>, 詹益俊<sup>1</sup>, 詹宝莹<sup>1</sup>

(1. 广西图像图形智能处理重点实验室(桂林电子科技大学), 广西 桂林 541004;

2. 广东省数学教育软件工程技术研究中心(广州大学), 广州 510006)

(\* 通信作者电子邮箱 tensorflowandcaffe@gmail.com)

**摘要:**针对复杂环境下行人检测不能同时满足高召回率与高效率检测的问题,提出一种基于卷积神经网络(CNN)的行人检测方法。首先,采用CNN中的单步检测升级版网络YOLOv2初步检测行人;然后,设计一个网络与YOLOv2网络级联。设计的网络具有目标分类和边界框回归的功能,对YOLOv2初步检测出的行人位置进行再分类与回归,以此降低误检,提高召回率;最后,采用非极大值抑制(NMS)处理的方法去除冗余的边界框。实验结果显示,在数据集INRIA和Caltech上,所提方法与原始YOLOv2相比,召回率提高3.3个百分点,准确率提高5.1个百分点,同时速度上达到了11.6帧/s,实现了实时检测。与现有的流行的行人检测方法相比,所提方法具有更好的整体性能。

**关键词:**行人检测;卷积神经网络;级联网络;分类回归;实时检测

**中图分类号:** TP391.413; TP18    **文献标志码:**A

### Pedestrian detection method based on cascade networks

CHEN Guangxi<sup>1</sup>, WANG Jiaxin<sup>1\*</sup>, HUANG Yong<sup>2</sup>, ZHAN Yijun<sup>1</sup>, ZHAN Baoying<sup>1</sup>

(1. *Guangxi Key Laboratory of Intelligent Processing of Computer Images and Graphics (Guilin University of Electronic Technology), Guilin Guangxi 541004, China*;

2. *Guangdong Engineering Technology Research Center for Mathematical Educational Software (Guangzhou University), Guangzhou Guangdong 510006, China*)

**Abstract:** In complex environment, existing pedestrian detection methods can not be very good to achieve high recall rate and efficient detection. To solve this problem, a pedestrian detection method based on Convolutional Neural Network (CNN) was proposed. Firstly, pedestrian locations in input images were initially detected with single step detection upgrade network (YOLOv2) derived from CNN. Secondly, a network with target classification and bounding box regression was designed to cascade with YOLOv2 network, which made reclassification and regression of pedestrian location initially detected by YOLOv2, to reduce error detections and increase recall rate. Finally, a Non-Maximum Suppression (NMS) method was used to remove redundant bounding boxes. The experimental results show that, in INRIA and Caltech dataset, the proposed method increases recall rate by 3.3 percentage points, and the accuracy is increased by 5.1 percentage points compared with original YOLOv2. It also reached a speed of 11.6FPS (Frames Per Second) to realize real-time detection. Compared with the existing six popular pedestrian detection methods, the proposed method has better overall performance.

**Key words:** pedestrian detection; Convolutional Neural Network (CNN); cascade network; classification and regression; real-time detection

### 0 引言

随着智慧城市建设在国际国内如火如荼地展开,针对智能监控等相关计算机视觉的技术需求也日益增加。行人检测作为计算机视觉领域的基础任务,引起了国内外计算机视觉领域专家学者的特别关注<sup>[1]</sup>。传统的行人检测方法是利用人工设计的特征提取器,通过提取方向梯度直方图(Histogram of Oriented Gradients, HOG)、局部二值模式(Local Binary Pattern, LBP)等特征来训练分类器,实现对行人的检测,但人工设计的行人特征很难适应行人行为的大幅度变化。

本质上说,行人检测只是一种特殊的通用目标检测,因而

可以借鉴通用目标检测的方法来实现。目前主流的通用目标检测方法主要分为两种:一种是two-stage;另一种是一one-stage。two-stage方法主要使用更快、更富有特征层次的卷积神经网络(Faster Regions with Convolutional Neural Network feature, Faster R-CNN)<sup>[2]</sup>、快速且丰富的特征层次结构网络(Fast Regions with Convolutional Neural Network feature, Fast R-CNN)<sup>[3]</sup>,而one-stage的方法有统一的实时对象检测(You Only Look Once, YOLO)<sup>[4]</sup>、YOLO升级版(YOLOv2)<sup>[5]</sup>、单发多框检测器(Single Shot multibox Detector, SSD)<sup>[6]</sup>。与two-stage的方法相比,one-stage的方法检测速度更快,但在检测质量上稍低。实时检测是目前智能产品市场的主流需求,

收稿日期:2018-06-28;修回日期:2018-08-14;录用日期:2018-08-31。    基金项目:国家自然科学基金资助项目(61462018);广东省数学教育软件工程技术研究中心开放基金资助项目(LD16124X);桂林电子科技大学研究生教育创新项目(2016XWYJ09)。

**作者简介:**陈光喜(1971—),男,四川金堂人,教授,博士,CCF会员,主要研究方向:可信计算、图像处理;王佳鑫(1992—),男,江苏泰州人,硕士研究生,主要研究方向:图像处理;黄勇(1958—),男,四川达州人,教授,博士,主要研究方向:数学教育智能软件与应用;詹益俊(1990—),男,河南商城人,硕士研究生,主要研究方向:图像处理;詹宝莹(1994—),女,辽宁辽阳人,硕士研究生,主要研究方向:图像处理。



YOLOv2 在实时性检测方面表现突出,速度更快,但是直接使用 YOLOv2 检测行人时,因为在 INRIA<sup>[7]</sup> 和 Caltech<sup>[8]</sup> 中行人的像素比较低, YOLOv2 检测效果较差, 行人位置也不够准确。此外, 在高重叠度(Intersection Over Union, IOU) 阈值条件下, YOLOv2 的效果也不甚理想。

CNN (Convolutional Neural Network) 方法如超网络(HyperNet)<sup>[9]</sup> 大量使用了特征融合来提高小目标的检测质量, 而特征空间网络(Feature Pyramid Network, FPN)<sup>[10]</sup> 则利用多层特征预测来提高检测质量。行人检测方法也可以融合 CNN 特征与传统行人特征, 如 Mao 等<sup>[11]</sup> 提出边缘特征, 分割特征对行人检测很有效, 他们设计了超学习网络(Hyperlearn)进行特征融合, 提高了小目标行人的检测质量。特征融合的优点在于低层的特征语义信息比较少, 但是目标位置准确; 高层的特征语义信息比较丰富, 但是目标位置比较粗略, 同时融入了上下文语义特征, 利用了高层低层特征易于检测, 但是这些方法的主干网络非常深, 或者采用多特种融合的方法导致网络参数过于庞大, 导致了检测速度非常慢, 影响了在实际检测中的应用性。

分阶段检测方法的优点是单个网络阶段计算量不大, 计算效率高。如多任务级联卷积网络(Multi-Task Cascaded Convolutional Network, MTCNN)<sup>[12]</sup> 检测人脸采用了一种级联网络, 分 3 个阶段: 第一个阶段提取人脸候选框, 第二个阶段定位人脸, 第三个阶段精确人脸位置, 此方法检测速度快, 检测质量高。本文参考了 MTCNN 方法设计了一个行人级联网络, 根据深层特征与浅层特征融合的思想改进了 YOLOv2。首先使用 YOLOv2 对行人进行检测, 在检测时使其输出预测的坐标; 然后利用行人在水平方向上比较密集这一特点设计网络 Person 用于级联 YOLOv2, 这个网络主要具备两个作用: 1) 用于对 YOLOv2 检测出的行人再次判断是否为行人, 2) 对 YOLOv2 检测出的行人的位置进行回归。第二个网络是一个回归网络, 其网络层数浅, 且检测速度快。本文的创新之处在于级联网络方法, 结合了深层特征网络与浅层特征网络的优点, 能够完成实时检测任务。

## 1 本文方法

本文方法首先使用了 YOLOv2 的算法。YOLOv2 的计算流程公式为:  $y = f_n(\sigma_{n-1}) = f_n(f_{n-1}(\dots f_1(x)))$ ,  $x$  为原图。可以看出 YOLOv2 的每一层参数必须训练恰当, 才能得到合适的检测效果。YOLOv2 网络有 19 层, 训练的行人为小目标时, 网络难以训练。

直接采用 YOLOv2 检测时, 较深的网络得到抽象的特征, 浅层的网络得到图像的细节。上述公式采用的是高级抽象特征作检测, 只能检测出行人的大致位置, 或者会误检出一些行人, 而不能检测出行人的具体位置。本文的主要思路是, 采用级联方法, 根据浅层网络提出人的细节特征这一特点设计了一个浅层网络, 精确定位人的位置, 但由于网络较浅, 学习到的特征不足以用于定位行人, 即输入原图无法得到行人位置, 但是可用于候选框位置小范围的修正, 因此结合了浅层网络与深层网络的特点, 用浅层网络再次回归深层网络预测的结果。

检测网络提取特征的过程公式为  $D = F_4(F_3(\dots F_1(x)))$ ,  $x$  为原图, 经过 4 层网络提取特征训练, 级联网络检测公式为

$D_f = F_4(F_3(\dots F_1(y)))$ ,  $y$  为 YOLO 的检测结果, 因此结合了深层网络与浅层网络的特点能够得出高质量的检测效果, 且由于是分阶段检测, 没有集中在同一个网络, 网络的计算量没有增大, 检测速度快。与此相对的是, FPN 等网络是在一个网络中结合了深层、浅层特征, 计算量加大, 不利于网络训练。

本文改进的 YOLOv2 方法首先是在 YOLOv2 上作了大量的对比, 分析在行人检测场景中使用 YOLOv2 产生的效果, 通过改变其判断行人的 threshold(阈值) 来调整检测性能。此外, 对于如何在降低 threshold 的同时保持准确率不变, 如何在提高行人定位准确性的同时提高召回率, 如何保证检测速度尽可能地快, 这三个问题上提出了相应的解决方案。

1) 针对准确率不理想的问题, 设计新的网络架构, 此网络具有再次判断行人类别的功能, 作一个二分类预测, 即人和背景, 同时在网络中通过使用正负样本的方法来提高准确率。

2) 针对召回率较低的问题, 设计一个具有边界框回归功能的网络架构, 通过回归原始 YOLOv2 的预测框, 提高行人预测框定位的准确性, 检测到的行人正确定位的数量就会增加, 行人召回率随之提升。

3) 针对实时检测必须保证检测速度这一问题, 这就要求设计的网络层数比较浅, 网络层数越浅, 检测速度越快。

综合这三个问题的解决方案, 在以行人为目标的基础上, 设计一个网络 Person。此网络具有分类与回归的功能, 网络层数很浅, 只有 4 层。分析行人的真实框纵横比, 选取合适的卷积核, 再对上一步 YOLOv2 测试的三种阈值下的预测框分别传入 Person 网络中作分析对比, 得出 threshold 为 0.01 时的效果最优, 最后再与其他行人检测算法进行对比分析。

### 1.1 YOLOv2 初步检测

为了提高检测速度, 采用 YOLOv2 初步检测行人。YOLOv2 是目前最快的通用目标检测网络, 能够在检测速度和检测质量上进行权衡。

#### 1.1.1 置信度计算

YOLOv2 检测时是整图输入的, 首先将图片分成  $S \times S$  个网格, 按式(1) 计算每一个网格置信度。首先判断网格中是否包含目标, 见式(2), 如果包含目标  $\text{Pr}(\text{Object}) = 1$ , 再判断该目标是否为人的概率  $\text{Pr}(\text{Person} | \text{Object})$ , 最后乘以定位的准确性 IOU, 即 IOU 为预测框与真实框的交并比。预测框与真实框的交集见式(3):

$$\text{Conf}(\text{Person}) = \text{Pr}(\text{Object}) \times \text{Pr}(\text{Person} | \text{Object}) \times \text{IOU} \quad (1)$$

$$\text{Pr}(\text{Object}) = \begin{cases} 1, & \text{网格中有目标} \\ 0, & \text{网格中没有目标} \end{cases} \quad (2)$$

$$\text{IOU} = \text{AO}/\text{AU} \quad (3)$$

其中: AO(Area of Overlap) 为面积的重叠, 表示预测框的面积与真实框的面积的交集; AU(Area of Union) 为面积的并集, 表示预测框的面积与真实框的面积的并集。

#### 1.1.2 YOLOv2 网络结构

采用官方的 YOLOv2 检测网络结构, 输入任意尺寸图片, 重设大小到  $416 \times 416$ (宽  $\times$  高), 经各层卷积池化后, 图片大小改为  $13 \times 13$ , 并将输出类别修改为 2 类, 即行人与背景。在训练网络时, 需要预设锚点数量及位置。随着迭代次数不



不断增加,从网络学习到行人特征,预测框参数不断调整,最终接近真实框。YOLOv2结构如图1所示。

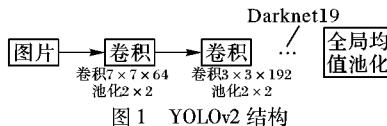


图1 YOLOv2结构

Fig. 1 YOLOv2 structure

## 1.2 Person 网络精确检测

原始YOLOv2并不是专为行人检测所设计的网络,检测行人的召回率和准确率均不高。为提高行人分类的准确率及行人预测的召回率,设计了Person网络。Person网络的主要功能是对YOLOv2的预测坐标及类别进行微调,使预测更加精准。

### 1.2.1 Person 网络结构

Person网络是一个具有目标分类和位置回归两种功能的网络,采用4个卷积层、3个Max-pooling层、1个全连接(Full Connection, FC)层如表1。全连接层包含2个分支:一个用于Softmax分类,另一个用于边界框回归。Person网络的输入是像素为 $30 \times 90$ 的行人样本图,行人的检测框都是高比宽长的矩形框且宽高比接近1:3,因此选取的卷积核宽高比也为1:3,采用两个 $2 \times 6$ 、两个 $1 \times 3$ 的卷积核。非线性激活层采用PReLU(Parametric Rectified Linear Unit)<sup>[13]</sup>,提高网络收敛速度。参考全卷积网络(Fully Convolutional Network, FCN)分割网络思想,采用了不同卷积层之间的特征融合,将Conv3、Conv4的特征通过双线性插值算法重设为相同大小并进行拼接,这种不同层之间的特征融合可以提高网络的检测性能。例如,原始行人像素为 $30 \times 90$ ,经过3次池化,行人像素值变为 $3 \times 6$ 。检测这种 $3 \times 6$ 的图像发现其效果不佳,而采用不同层的特征拼接且联合了不同层的语义特征,网络对于不同大小的行人具有很高的检测能力,因此最终网络结构添加了两个上采样层以及一个用于降维的卷积层。网络结构示意图如图2所示。YOLOv2预测出的行人框,输出结果作为级联网络的最终预测框。

表1 Person 网络

Tab. 1 Person network

Type	Filters	Size/Stride	Output
Conv1	32	$2 \times 6$	$29 \times 82$
Max-pooling1	—	$3 \times 3/2$	$14 \times 42$
Conv2	64	$2 \times 6$	$13 \times 37$
Max-pooling2	—	$3 \times 3/2$	$6 \times 18$
Conv3	64	$1 \times 3$	$6 \times 16$
Max-pooling3	—	$3 \times 3/2$	$3 \times 8$
Conv4	128	$1 \times 3$	$3 \times 6$
FC5	256	—	—

### 1.2.2 代价函数

Person网络代价函数包含两个部分:第一个部分是分类,第二个部分是边界框回归。

1) 分类损失采用交叉熵:

$$Loss_i^{\text{class}} = -(y_i^{\text{class}} \ln(p_i) + (1 - y_i^{\text{class}}) \times (1 - \ln(p_i))) \quad (4)$$

2) 边界框回归损失采用欧氏距离:

$$Loss_i^{\text{box}} = 0.5 \| y_i^{\text{box}} - y'_i^{\text{box}} \|_2^2 \quad (5)$$

其中: $y'_i$ 为预测框,  $y_i$ 为真实标签框,  $P_i$ 为网络预测出的目标为行人的概率。

总代价函数为:

$$Loss = Loss_i^{\text{class}} + Loss_i^{\text{box}} \quad (6)$$

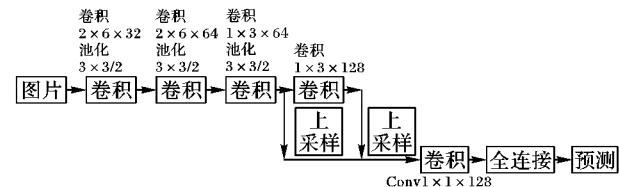


图2 Person 网络结构

Fig. 2 Person network structure

### 1.2.3 Person 算法设计

在训练阶段,简单样本、困难样本不平衡的比率对训练结果产生较大影响。简单样本是指样本行人比较清晰且容易被网络检测,而另一种样本行人较为模糊,网络学习这种模糊样本较为困难,称为困难样本。如果不作样本平衡,困难样本对网络学习过程中产生的权值影响小,随着网络训练,困难样本可能会被当成背景进而被忽视。Person网络设计阶段为了提升网络分类的准确率,采用平衡样本算法。在全连接层的loss阶段,对一个batch的样本进行快速排序。选择loss较高的70%的样本作为困难样本进行前传,剩下30%的样本作为简单样本回到原数据集。通过这种方式,在每个batch训练时会减少简单样本产生,以此来平衡样本,提高分类的准确性。

## 1.3 本文总体算法

本文第一步采用的是卷积神经网络<sup>[14-15]</sup>中YOLOv2对行人进行初步检测。为了尽可能多地检测出复杂环境下的行人,降低置信度阈值,在INRIA和Caltech上,总检测框的数量从523个提高到653个,但是误检率也随之提升。这一问题利用Person网络的分类功能得到解决:第一步再次判断目标是否为行人,去除YOLOv2降低阈值后的背景框;第二步采用Person网络的回归功能,对YOLOv2检测出的位置进行回归,提高定位的准确性。Person网络结构如图2所示。图3是本文方法的训练阶段网络,级联了YOLOv2算法和Person网络,训练时分阶段训练。传统CNN网络没有结合相邻通道特征,对小目标的检测效果不佳,Person网络引入了通道特征拼接的方法,提升了小目标的检测性能。从结构图可看出本文方法是一个多阶段训练的网络。图4是本文方法的检测阶段,首先图片经过深层网络YOLOv2预测出行人的位置,在经过Person网络回归行人的位置使最终的检测框比单个的CNN网络更加准确,传统网络没有边界框再次回归的功能。

算法步骤如下:

- 1) 原始图片1在YOLOv2和Person网络上各自训练;
- 2) 降低YOLOv2的阈值;
- 3) 原始图片进入YOLOv2进行初步行人检测得到初步检测位置 $L$ ;
- 4) Person网络再次分类预测框,并将预测框的位置进行回归得到精确回归位置 $R$ ;
- 5) 对预测框进行非极大值抑制(Non-Maximum



Suppression, NMS) 处理<sup>[16]</sup> 得到最终检测位置  $O$ 。

数据流程为  $I \rightarrow L \rightarrow R \rightarrow O$ 。

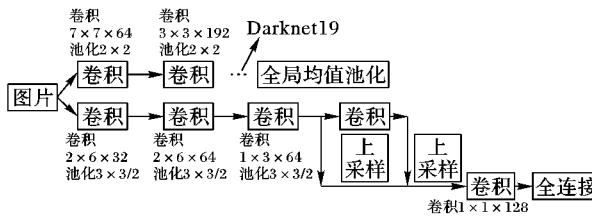


图3 训练阶段网络结构

Fig. 3 Network structure at training stage

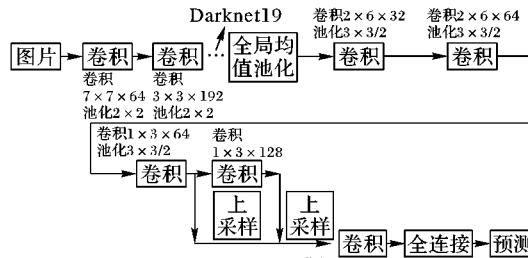


图4 测试阶段网络结构

Fig. 4 Network structure at test stage

#### 1.4 总体代价函数

本文总代价函数如式(7)所示。训练 YOLOv2 阶段时,  $(\alpha, \beta)$  取  $(1, 0)$ ,  $(\lambda_{coord}, \lambda_{noobj})$  取  $(5, 0.5)$ ; 训练 Person 网络时,  $(\alpha, \beta)$  取  $(0, 1)$ ,  $\omega_i^{obj}$ : 如果目标出现在第  $i$  个网格中时为 1,  $\omega_i^{obj}$ : 第  $i$  个网格中的第  $j$  个边界框代表这个网格物体的预测。 $S$  代表有  $S$  个网格,  $B$  表示边界框的数量。

$$\begin{aligned} L = & \alpha \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \omega_{ij}^{obj} [(x_i - x'_i)^2 + (y_i - y'_i)^2] + \\ & \alpha \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B \omega_{ij}^{obj} [(\sqrt{w_i} - \sqrt{w'_i})^2 + (\sqrt{h_i} - \sqrt{h'_i})^2] + \alpha \sum_{i=0}^{S^2} \sum_{j=0}^B \omega_{ij}^{obj} (C_i - C'_i)^2 + \\ & \alpha \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B \omega_{ij}^{noobj} (C_i - C'_i)^2 + \alpha \sum_{i=0}^{S^2} \omega_{ij}^{obj} (p_i(c) - p_i(c')^2 + \beta (- (y_i^{class} \log(p_i)) + (1 - y_i^{class}) \times (1 - \log(p_i)))) + 0.5 \beta \|y_i^{box} - y'^{box}_i\|_2^2 \quad (7) \end{aligned}$$

根据总体代价函数求解权值更新的过程如下:

$$E_k = \sum_{j=1}^l (x_i - x'_i)^2$$

$$f(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

$$\beta_i = \sum_{k=1}^q w_{ki} b_k$$

其中:  $E_k$  代表误差项,  $f(x)$  为激活函数,  $\beta_i$  为输出层的输入值,  $b_k$  为上一层 CNN 的输出值。假设所有层的阈值为 0, 权值更新公式为:

$$\begin{aligned} \Delta w_{hi} = & -a \frac{\partial E_k}{\partial w_{hi}} = -a \left( \frac{\partial E_k}{\partial x'_i} \cdot \frac{\partial x'_i}{\partial \beta_i} \cdot \frac{\partial \beta_i}{\partial w_{hi}} \right) = \\ & -ab_h \left( \frac{\partial E_k}{\partial x'_i} \cdot \frac{\partial x'_i}{\partial \beta_i} \right) = -ab_h (2 \times (x_i - x'_i) \cdot f'(\beta_i)) \end{aligned}$$

其中  $a$  为学习率, 所以

$$\Delta w_{hi} = \begin{cases} -2ab_h(x_i - x'_i), & \beta_i > 0 \\ 0, & \text{其他} \end{cases}$$

以此类推可以得到整体 Loss 的更新权值公式。

## 2 实验分析

### 2.1 实验环境及评估

本文算法运行环境为一台 64 位的 Ubuntu 14.04 LTS, 内存为 16 GB, CPU 为 8 核, GPU 为 GTX750TI。YOLOv2 算法和 Person 网络算法在公开数据集 INRIA 和 Caltech 中选取训练数据, 实验中采用的评价指标为准确率(Precision, P)、召回率(Recall, R)以及准确率与召回率的综合评价指标 F 值(F-score)。一般来说, 准确率和召回率是相互矛盾的, 而 F 值则综合了这两个指标的评价参数, 当 F 值越高时则实验的检测性能更好。

$$Recall = \frac{\text{正确的行人框数}}{\text{真实标签框数}} \quad (8)$$

$$Precision = \frac{\text{正确的行人框数}}{\text{总共预测的框数}} \quad (9)$$

$$F = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (10)$$

### 2.2 YOLOv2 初步检测实验

以开源框架 Darknet19 为基础, 使用 YOLOv2 网络模型开始训练。为提高训练速度、防止过拟合, 选用的冲量常数为 0.9, 学习率衰减系数为 0.0005, 初始学习率为 0.0001, 学习率下降策略为 step。当迭代 25 000 次, 模型趋于稳定。为减少训练时间, 以 Darknet19 网络模型训练得到的网络参数, 初始化卷积层网络。该预训练模型是在 ImageNet1000 类数据集训练 10 个循环(epoch)后得到的预训练参数。

验证集为 288 张图片, 选自 INRIA 和 Caltech, 验证时将图片重设大小到  $64 \times 128$ 。NMS 为 0.4, best\_iou 控制着正确预测行人的阈值, 当  $IOU > best\_iou$  时才会将预测出的行人记为正确的行人; 反之记为错误。threshold 在 YOLOv2 阶段为置信度(confidence)阈值, 在 Person 网络阶段控制着预测是否为人的阈值, 初始值为 0.24, 当置信度  $> threshold$  时才会预测该目标为行人。本实验在验证集上均采用  $NMS = 0.4$ 、 $best\_iou = 0.5$  作为判断检测出的行人的阈值, 调整行人的置信度阈值效果如表 2( $threshold: 0.24, 0.1, 0.01$  分别代表预测概率超过 0.24、0.1、0.01 为行人)。

由表 2 可以看出, 随着 threshold 的下降, 总检测出的框的数量增加, 但正确的检测数没有增加, 准确率明显下降, 所以不能通过简单地降低判断目标概率的阈值 threshold 来提升 recall。

表 2 采用 YOLOv2 的检测结果

Tab. 2 Detection results by using YOLOv2 with different thresholds

阈值	总行人数	正确检测数	总检测数	准确率	召回率	F 值	平均 IOU
0.24	589	477	523	0.912	0.809	0.857	0.675
0.10	589	477	545	0.875	0.809	0.843	0.675
0.01	589	477	653	0.730	0.809	0.768	0.675

### 2.3 Person 网络实验

以开源框架 Caffe 为基础, 将从 YOLOv2 网络得到的坐标



作为输入值,输入 Person 网络进行分类与边界框回归,输入固定尺寸  $width \times height$  为  $30 \times 90$ 。当冲量为 0.9,权值衰减为 0.004,初始学习率为 0.001,学习率下降策略为 step,且当迭代训练达到 1000000 次时,模型达到饱和。训练时样本是从原始真实行人框上随机截取的,按截取的图片与真实行人框的 IOU,将截取的图片分为正样本、部分样本和负样本。负样本是  $IOU < 0.5$  的样本,正样本是  $IOU > 0.8$  的样本,其他样本为部分样本,正样本、部分样本、负样本比例为 1:1:3。实验结果如表 3 所示,threshold:0.24、0.1、0.01 是 YOLOv2 检测行人的阈值。对比表 2 与表 3 可知,使用级联方法可以得到比 YOLOv2 更高的召回率、准确率,以及定位的准确性更为精准。表 4 展示卷积核不同宽高比下的 Person 网络效果,从中可以得出选取为 1:3 的卷积核时,F 值最高,且检测效果最为突出。

表 3 采用级联网络的结果

Tab. 3 Detection results by using cascading networks with different thresholds

阈值	总行人数	正确检测数	总检测数	准确率	召回率	F 值	平均 IOU
0.24	589	477	489	0.975	0.809	0.884	0.778
0.10	589	488	502	0.972	0.828	0.894	0.776
0.01	589	496	515	0.963	0.842	0.898	0.776

表 4 采用 Person 网络的检测结果

Tab. 4 Detection results by using Person networks with different width-to-height ratios

宽高比	总行人数	正确检测数	总检测数	准确率	召回率	F 值	平均 IOU
1:1	589	475	520	0.913	0.806	0.816	0.669
1:3	589	496	515	0.963	0.842	0.893	0.776
1:4	589	490	512	0.957	0.832	0.890	0.771

实验结果表明,本文算法比原始 YOLOv2 算法可以得到更高的召回率、准确率以及更准确的 IOU。其效果对比展示在同一张图片,如图 5。



图 5 不同算法的结果对比

Fig. 5 Results comparison of different algorithms

对比图 5(a),图 5(b)出现虚框(误检的行人框),通过 Person 网络,对 YOLOv2 检测出的标记框再次判断是否为行人并将虚框去除。对比图 5(a),图 5(c)检测出的行人数量有所增加,且检测出标记框的位置更为精准。本实验将 YOLOv2 阈值设为 0.01 时,将 YOLOv2 的坐标传进 Person 网络能够得到比较高的召回率、准确率以及高质量的定位,因此取 YOLOv2 threshold 为 0.01 情况下初步预测行人,Person 网络 threshold 取 0.6 再次预测行人的数据作为本实验的最终检测数据。

## 2.4 不同算法行人检测性能分析

目前最常用的传统行人检测方法是基于滑动窗口策略,具有代表性的工作是 Felzenszwalb 等<sup>[17]</sup>提出的形变部位模型(Deformable Part Model, DPM)。在一定程度上,这个方法能消除部分遮挡的影响。其次,提取候选框集的描述特征过程,Dollar 等<sup>[18]</sup>提出积分通道特征(Integral Channel Feature, ICF),利用积分图技术对图像的各个特征通道进行快速计算,在文献[19]中进一步提出了聚合通道特征(Aggregate Channel Feature, ACF)。这些方法对严格依赖于手动提取的特征检测的鲁棒性差。基于深度学习——卷积神经网络算法的目标检测,典型的代表性的工作是 R-CNN 系列的结合区域候选框(Region Proposal)和 CNN 分类的目标检测框架。文献[17]中针对行人检测对 Faster R-CNN 作出改进,提出了区域候选框网络(Region Proposal Network, RPN)与提升森林(Boosted Forest, BF)结合的思想(RPN + BF)用于行人检测,此方法能有效地降低行人检测的误检率。SSD 是一种 one-stage 算法,没有候选框预选的过程,并且适应不同尺度的特征图,可以用于行人检测。Hyperlearn 是关于行人检测的研究方法,提出了传统方法与 CNN 方法特征融合的思想,其优点是利于小目标的行人检测,缺点是对大目标的行人适应为较差。

表 5 为各种算法的参数指标:准确率、召回率、F 值、检测时间的对比数据。从实验数据可以看出,本文算法对于行人检测任务的准确率略低于 RPN + BF 方法,高于 ACF、DPM、Hyperlearn 算法;算法表现出来的检测速度远远高于 RPN + BF、DPM、Hyperlearn 算法,略低于 ACF,但是,本文采用的算法是 CNN 算法,可以端到端地执行,且它的召回率最为显著。由于本文算法在第一阶段利用 YOLOv2 检测行人,相比 two-stage 算法,YOLOv2 没有候选框提取这一步骤,直接采取锚点进行预测,候选框提取的时间较长,所以 YOLOv2 这类 one-stage 算法检测速度普遍快,但因为没有采用候选框提取,产生的预测结果一般没有 two-stage 算法准确。YOLOv2 是 one-stage 算法,其特征是图片越小,其检测速度越快,检测质量越低;图片越大,其检测速度越慢,检测质量越高。行人在图片中是小目标,利用级联的 Person 网络分类和回归过程,对行人进行筛选并回归行人的位置,从而使得召回率和准确率比原始 YOLOv2 高,本文方法速度比 RPN + BF、DPM、Hyperlearn 均快,能够达到 11.6 帧/s(Frames Per Second, FPS)的检测速度,速度达到实时检测的目标,因此,可以认为本文算法的综合性能最佳。

表 5 不同算法的性能对比

Tab. 5 Performance comparison of different algorithms

算法	总行人数	正确检测数	总检测数	准确率	召回率	F 值	时间/s
DPM	589	461	480	0.960	0.782	0.862	4.956
ACF	589	466	485	0.961	0.791	0.868	0.046
RPN + BF	589	443	457	0.969	0.752	0.852	0.492
SSD	589	472	518	0.911	0.801	0.852	0.063
YOLOv2	589	477	523	0.912	0.809	0.857	0.059
Hyperlearn	589	490	521	0.940	0.832	0.883	0.596
本文算法	589	496	515	0.963	0.842	0.898	0.086



### 3 结语

针对复杂环境下行人检测不能同时满足高召回率与高效率检测的问题,本文提出了一种改进的级联网络——原始YOLOv2+Person网络的行人检测方案。结合了深层CNN网络与浅层CNN网络的优点,进而得到更准确的行人分类和行人预测框。与原始YOLOv2相比,本文方法的行人预测的准确率、召回率均有所提升,与ACF、RPN+BF、DPM、SSD相比,召回率、F值提升较为显著。本文主要工作在于将原始YOLOv2检测器改进为二阶段检测器,在速度与精度上达到了均衡且可以在GPU上进行并行计算,减少计算开销。如何进一步提高检测质量将是下一步研究的方向。

#### 参考文献 (References)

- [1] 苏松志,李绍滋,陈淑媛,等.行人检测技术综述[J].电子学报,2012,40(4):814–820.(SU S Z, LI S Z, CHEN S Y, et al. A survey on pedestrian detection [J]. Acta Electronica Sinica, 2012, 40(4):814–820.)
- [2] 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.
- [3] GIRSHICK R. Fast R-CNN[C]// Proceedings of the 2015 IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE, 2015: 1440–1448.
- [4] REDMON J, DIVVALA S, GIRSHICK R, et al. You Only Look Once: unified, real-time object detection [C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2016: 779–783.
- [5] REDMON J, FARHADI A. YOLO9000: better, faster, stronger [C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2017: 6517–6525.
- [6] LIU W, ANGUELOV D, ERHAN D, et al. SSD: Single Shot multibox Detector[C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2016: 21–37.
- [7] DALAL N, TRIGGS B. Histograms of oriented gradients for human detection[C]// Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2005: 886–893.
- [8] DOLLAR P, WOJEK C, SCHIELE B, et al. Pedestrian detection: a benchmark [C]// Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2009: 304–311.
- [9] KONG T, YAO A, CHEN Y, et al. HyperNet: towards accurate region proposal generation and joint object detection [C]// Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2016: 845–853.
- [10] LIN T Y, DOLLAR P, GIRSHICK R, et al. Feature pyramid networks for object detection [C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2017: 936–944.
- [11] MAO J, XIAO T, JIANG Y, et al. What can help pedestrian detection? [C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2017: 6034–6043.
- [12] ZHANG K, ZHANG Z, LI Z, et al. Joint face detection and alignment using multitask cascaded convolutional networks [J]. IEEE Signal Processing Letters, 2016, 23(10):1499–1503.
- [13] HE K, ZHANG X, REN S, et al. Delving deep into rectifiers: surpassing human-level performance on ImageNet classification [C]// ICCV 2015: Proceedings of the 2015 IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE, 2015: 1026–1034.
- [14] SZEGEDY C, TOSHEV A, ERHAN D. Deep neural networks for object detection [J]. Advances in Neural Information Processing Systems, 2013, 26(1):2553–2561.
- [15] TOMÈ D, MONTI F, BAROFFIO L, et al. Deep convolutional neural networks for pedestrian detection [J]. Signal Processing: Image Communication, 2016, 47(1):482–489.
- [16] ROTHE R, CUILLAUMIN M, VAN COOL L. Non-maximum suppression for object detection by passing messages between windows [C]// Proceedings of the 2014 Asian Conference on Computer Vision. Berlin: Springer, 2014: 290–306.
- [17] FELZENSZWAIB P, MCALLESTER D, RAMANAN D. A discriminatively trained, multiscale, deformable part model [C]// Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC: IEEE Computer Society, 2008: 1–8.
- [18] DOLLAR P, APPEL R, BELONGIE S, et al. Fast feature pyramids for object detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 36(8):1532–1545.
- [19] ZHANG L, LIN L, LIANG X, et al. Is Faster R-CNN doing well for pedestrian detection? [C]// ECCV 2016: Proceedings of the 14th European Conference on Computer Vision. Berlin: Springer, 2016: 443–457.

This work is partially supported by the National Natural Science Foundation of China (61462018), the Open Fund of Guangdong Engineering Technology Research Center for Mathematical Educational Software (LD16124X), the Graduate Education Innovation Project of Guilin University of Electronic Science and Technology (2016XWYJ09).

**CHEN Guangxi**, born in 1971, Ph. D., professor. His research interests include trusted computing, image processing.

**WANG Jiaxin**, born in 1992, M. S. candidate. His research interests include image processing.

**HUANG Yong**, born in 1958, Ph. D., professor. His research interests include mathematical education intelligent software and application.

**ZHAN Yijun**, born in 1990, M. S. candidate. His research interests include image processing.

**ZHAN Baoying**, born in 1994, M. S. candidate. Her research interests include image processing.