计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2023,32(5):132−140 [doi: 10.15888/j.cnki.csa.009085]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

融合 VovNet 网络和可变形卷积的非机动车辆检测①

王　林, 翁友虎

(西安理工大学 自动化与信息工程学院, 西安 710048)
通信作者: 翁友虎, E-mail: wyh7251@163.com

摘　要: 针对道路监控下因监控探头高度角度不同, 目标非机动车辆存在不同形式的模糊形变问题且特征信息不足造成的漏检误检现象, 提出了一种融合 VovNet 网络和可变形卷积的非机动车辆检测模型. 使用一次聚类连接网络(VovNet) 结合原网络特点提出的 CSPVovNet 替换原有的 CSPDarknet 主干网络进行特征的提取, 增强了有效特征的复用, 缓解因深层卷积造成的小目标物体特征信息进一步丢失的问题. 将可变形卷积引入到不同的网络层替换传统卷积, 在公共数据集 Pascal VOC2007 和自建非机动车辆数据集上分别训练测试, 根据最终性能选择 YOLOv5-C 方案. 改进后的网络选取 EIoU_loss 作为定位损失, 通过消融实验验证得出最终改进对网络性能有所提升, 最终的网络优化结果较原 YOLOv5s 网络 *mAP* 提升了 4.14 个百分点, 对漏检误检现象很好的缓解.

关键词: 非机动车辆; 可变形卷积; YOLOv5s; 聚类网络; 目标检测; 卷积神经网络 (CNN)

引用格式: 王林,翁友虎.融合 VovNet 网络和可变形卷积的非机动车辆检测.计算机系统应用,2023,32(5):132–140. http://www.c-s-a.org.cn/1003-3254/9085.html

Non-motor Vehicle Detection Based on VovNet Network and Deformable Convolution

WANG Lin, WENG You-Hu

(School of Automation and Information Engineering, Xi'an University of Technology, Xi'an 710048, China)

Abstract: To solve missing and false detection caused by different fuzzy deformations and insufficient features of target non-motor vehicles due to different heights and angles of detectors under road monitoring, this study proposes a non-motor vehicle detection model based on one-shot aggregation (VovNet) network and deformable convolution. CSPVovNet proposed by the VovNet network combined with the characteristics of the original network is used to replace the original CSPDarknet backbone network for feature extraction. This enhances the reuse of effective features and alleviates the further loss of features of small target objects caused by deep convolution. Deformable convolution is introduced into different network layers to replace the traditional convolution. Training and testing are carried out on the public data set Pascal VOC2007 and the self-built non-motor vehicle data set, respectively. The YOLOv5-C scheme is selected according to the final performance. The improved network selects EIoU_loss as the location loss. The ablation experiment shows that the final improvement improves the network performance, with the final network optimization result being 4.14 percentage points higher than the original YOLOv5s network in terms of *mAP*, which thus effectively alleviates missing and false detection.

Key words: non-motor vehicle; deformable convolution; YOLOv5s; aggregation network; target detection; convolutional neural network (CNN)

① 基金项目: 陕西省科技计划重点项目 (2017ZDCXL-GY-05-03)
收稿时间: 2022-11-07; 修改时间: 2022-11-29; 采用时间: 2022-12-11; csa 在线出版时间: 2023-03-24
CNKI 网络首发时间: 2023-03-27

中国城市交通中非机动车扮演着重要角色, 我国依据国家道路交通发展的需求, 对非机动车有自己详细的分类. 将自行车, 电动三轮车, 电动自行车, 残疾人和老年人使用的助力车, 以及畜力车统一划分为非机动车. 随着科技的发展以及时代的进步, 部分车型以逐步淘汰, 但以共享单车为主的自行车得到了迅速普及, 除此之外适宜短途出行的电动自行车也仍是人们喜爱的交通工具. 非机动车辆数量的不断扩增, 以及其车辆驾驶员淡薄的道路交通安全意识, 随之带来许多因非机动车造成的严重交通事故. 非机动车驾驶员相较于汽车卡车等驾驶员处于交通弱势地位, 一旦发生交通事故, 往往会引起较为严重的伤亡. 为避免交通事故所带来的不必要人员伤亡, 近年来我国各地方纷纷制定了较为严格的非机动车安全条例, 加强非机动车辆的管理力度, 但相应的需要交通管理部门出动大量的人力物力进行交通监管. 现有交通路段均设有大量视频监控, 但与日俱增的交通监控视频信息未得到充分的利用. 人工智能技术的不断发展, 带来的智能化的监控方式很大程度上减少了不必要的人力资源, 基于路面监控的非机动车在线智能检测对城市交通道路安全有着重要的意义. 传统的目标检测算法往往通过人工手动提取图像特征信息, 其过程具有较强的可解释性, 但其识别度低, 计算量大, 运行速度慢. 常见的经典传统目标检测算法如 HOG+SVM[1]、DPM[2]、SIFT[3]. 传统的方法都是将所提出的目标特征放入标准分类器中进行分组识别, 但是因为在实际场景中非机动车体积较小, 而且容易被屏蔽等原因导致无法获得较好的目标特性, 使得传统方法难以适应现实场景需要. 随着机器学习领域分支的延伸, 对深度学习探索不断深入的同时将目标检测技术推入新的台阶. 目前深度学习的目标检测算法主要包括两大类:基于区域建议概念的 two-stage 检测算法和基于回归思想的 one-stage 检测算法[4] 两阶段的目标检测算法先生成候选区域 (region proposals), 再对生成的候选框进行分类检测, 典型代表算法: R-CNN[5], SPPNet[6], Fast R-CNN[7], Faster R-CNN[8]. 其识别精度高, 误检漏检率较低, 但检测速率缓慢, 无法满足实际场景中的实时检测任务. 单阶段检测算法无需提前生成候选框, 能直接对目标物体进行分类回归, 典型代表算法: SSD[9]、YOLO[10–13]. 文献 [14] 提出基于组件的传统检测算法 DPM, 并通过已知的几何上下文信息提升检测效率. 文献 [15] 利用 EdgeBoxes[16] 提取感兴趣区域并与 Fast R-CNN 相结合提升非机动车的检测效率. 文献 [17] 通过重新设计 YOLOv3 的特征融合方式降低非机动车的漏检率. 上述文献均都未能对视频图像中的非机动车小目标, 遮挡目标特征进行有效处理, 检测效率不高, 同时参数较大, 不利于移动部署. 本文根据易部署移植, 实时性等因素选用了 YOLO 家族中 YOLOv5 目标检测算法, 在其基础上进行改进优化. 该算法较之前提的一些其他算法, 其精度和复杂场景下的检测能力有待提升. 本文针对上述不足对 YOLOv5 算法进行改进优化, 使其检测精度和复杂场景的适应能力有很大提升, 同时仍具备良好的实时性.

1 本文算法

1.1 YOLOv5 模型

YOLOv5 算法共分为 4 部分, 分别为输入端、主干网络 (Backbone)、多尺度特征融合网络 (Neck) 以及头部分类预测网络 (Prediction).

网络输入端主要用来进行数据的预处理, 使用了 Mosaic 数据增强对待检测图像进行随机缩放、裁剪、排布的处理; 采用自适应锚框计算, 对不同数据集计算产生合适的锚框尺寸; 运用自适应图片缩放对不同尺寸原始图像进行黑边填充, 再缩放到符合输入网络的图像大小. 主干网络主要用来提取图像特征信息, 提出了一种切片操作 Focus, 对特征图进行类似邻近下采样操作, 将一张特征图分成 4 份扩充到通道空间中, 与普通卷积相比充分保留特征信息. 同时还设计出两种跨阶段部分网络 (cross stage partial network, CSPNet), 一种用于主干网络如图 1 中的 CSP1_X, 另一种用在 Neck 网络如图 1 中的 CSP2_X 所示. Neck 网络主要使用 CSP2_X 和 PANet 用来进行加强网络的特征融合能力. 头部网络即输出端主要采用 nms 非极大值抑制, 以及使用 CIoU_loss 作为 bounding box 的损失分类回归.

1.2 主干网络的改进

1.2.1 DenseNet 与 VovNet 网络

DenseNet 网络是由 Huang 等人[18] 提出了一种新型的网络结构. 分析参照了 ResNet 以及 Inception 网络的思想, 其网络从检测物体的特征入手, 通过合理的特征重用方式提升网络提取信息的能力同时减小了参数数量并很好地缓解卷积网络造成的梯度问题, 相比

ResNet 网络有一定的提升. 一般卷积网络中有N层卷积, 网络就会形成N条连接, 但在 DenseNet 网络中会产生$N(N+1)/2$条连接, 通过特征图的 concat 聚合提升网络宽度从而增加深层网络输入变化, 增强信息表达能力. 单个阶段的 DenseNet 网络主要由一个 denseblock 模块和 transition layers 组成. Denseblock 通过若干个 dense 层拼接而成. 其网络从检测物体的特征入手, 通过合理的特征重用方式提升网络提取信息的能力同时减小了参数数量并很好地缓解卷积网络造成的梯度问题, 相比 ResNet 网络有一定的提升. 但由于 DenseNet 网络由于多次的 concatenate 操作, 数据需要多次的复制存储, 显存容易增加过快, 需要一定的显存优化技术, 同时利用简单粗暴的方式对特征堆叠利用必然带来了一定程度的特征冗余, 为改善上述问题本文使用其改进网络 VovNet[19], 该网络 VovNet 可以使用一次性聚合 (OSA) 模块有效地提取不同的特征表示, 该模块一次连接后续层. 由于 OSA 模块可以捕获多尺度的接收场, 因此不同的特征图允许对象检测和分割, 以很好地处理多尺度对象和像素, 尤其适用于小对象. 图 2 为 DenseNet 与 VovNet 网络主要结构示意图.

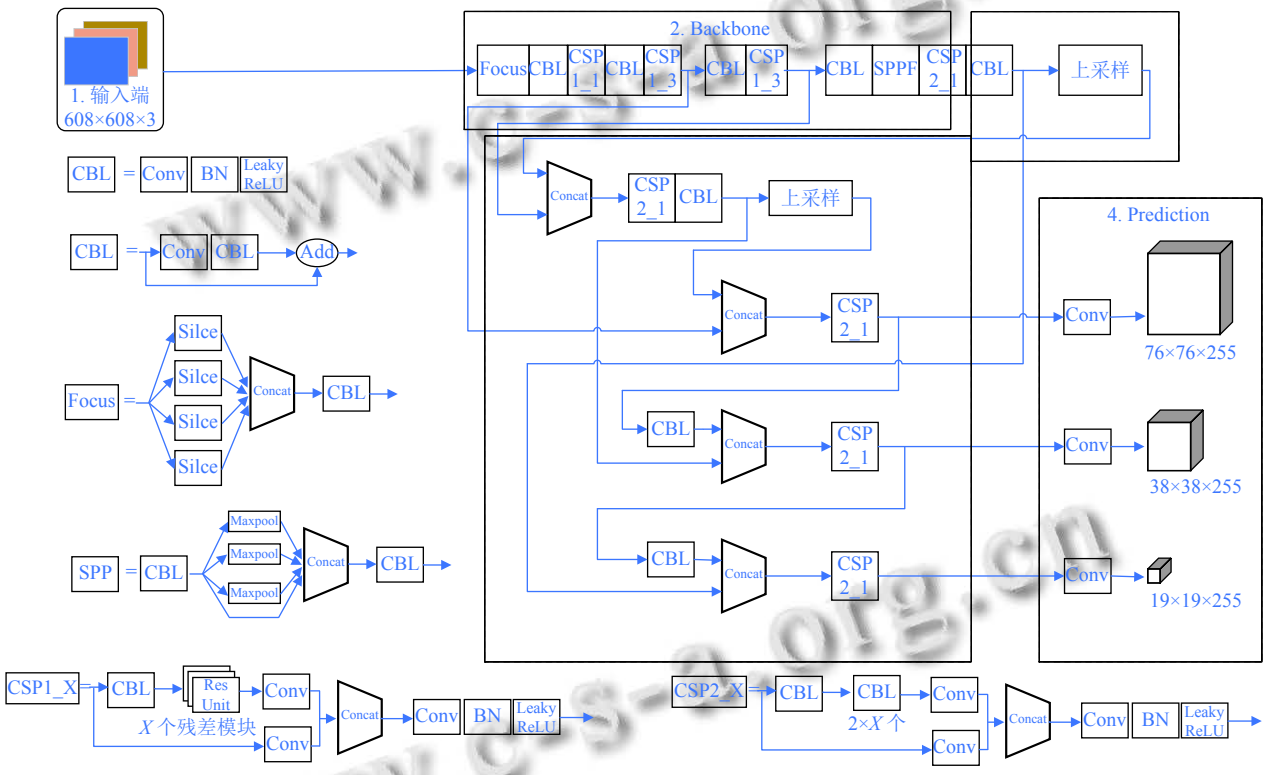


图 1 YOLOv5 网络结构

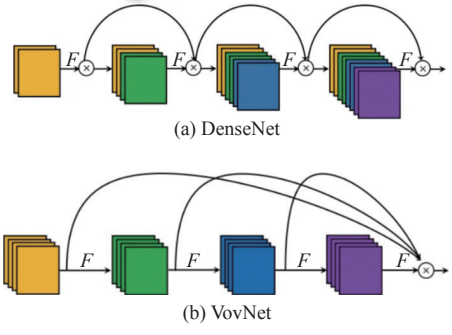


图 2 DenseNet 与 VovNet 网络主要结构图

1.2.2 跨区域局部 VovNet

非机动车辆与汽车相比, 在交通监控下的像素点相对较少, 除密集车流量的情况下车辆之间也存在相互遮掩现象外, 驾驶者自身会带来严重的遮挡. 含目标物体的图片输入深度网络经过多层卷积后其有效信息可能进一步减少. 上述情况均带来一定的特征缺失从而引起造成漏检, 错检. 为缓解此类问题, 本文使用更为高效的主干网络 VovNet, 并结合 CPSNet 特点对其进行改进. VovNet 网络继承了 DenseNet 网络特征复用的优点, 充分保留了小目标浅层特征信息同时避免

了网络各层新输出比重逐渐降低的问题, 每个卷积层仅为包含双向连接, 即一个连接下一层获取更大感受野和深层语义信息, 一个在最终输出中聚合映射. 图 3 为改进前的 VovNet 和改进后的 VovNet.

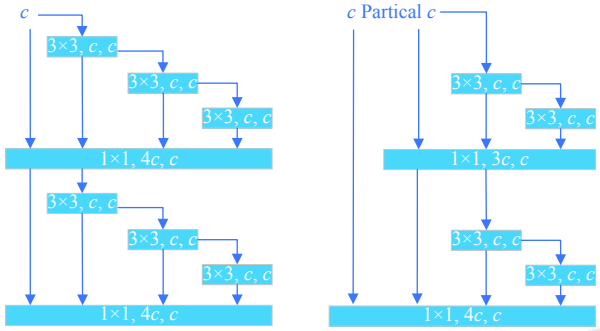


图 3 改进前后的主体结构

当输入图像进入改进后的模块后会分为 2 个部分, 第 1 部分将输入图像通过 1×1 的卷积对浅层特征进行简单的压缩提取, 第 2 部分通过不断 3×3 卷积操作提取更深层次的特征信息, 在保证输入与输出图像大小保持不变的情况下, 最终得到所有层的输出, 将所有输出进行通道拼接再进行 1×1 的卷积减少输出通道数得到最终的输出. 将改进后的 VovNet 网络运用到 YOLOv5 的主干网络后, 其网络结构为图 4 所示.

上述网络在原有 CSPDarknet 网络的基础上将具有 ResNet 模块的 CSP1_x 模块进行了替换, 该结构的设计对网络不同的深度采用了不同的配置, 本文根据 VovNet-57 对改进后 4 个阶段的 CSP-VovNet 进行配置, 其比例为 1:1:4:3. 根据 VovNet 网络的特点删除了原网络中的步长为 2 的 3×3 卷积改为步长为 1 的 3×3 最大池化进行下采样, 即图 4 中 P3, P4, P5. 通过对原有残差结构的改进让网络进一步加强了特征提取的能力, 保留了相对比例的浅层信息, 如纹理, 颜色, 边缘, 棱角等. 同时通过不同阶段增加输出通道来增加高层语义信息的比重, 避免了 DenseNet 网络只有少量新输出的特点. 本文 CSPVovNet 提升主干提取特征的能力, 同时降低很大程度的计算量, 提高了检测的精准度.

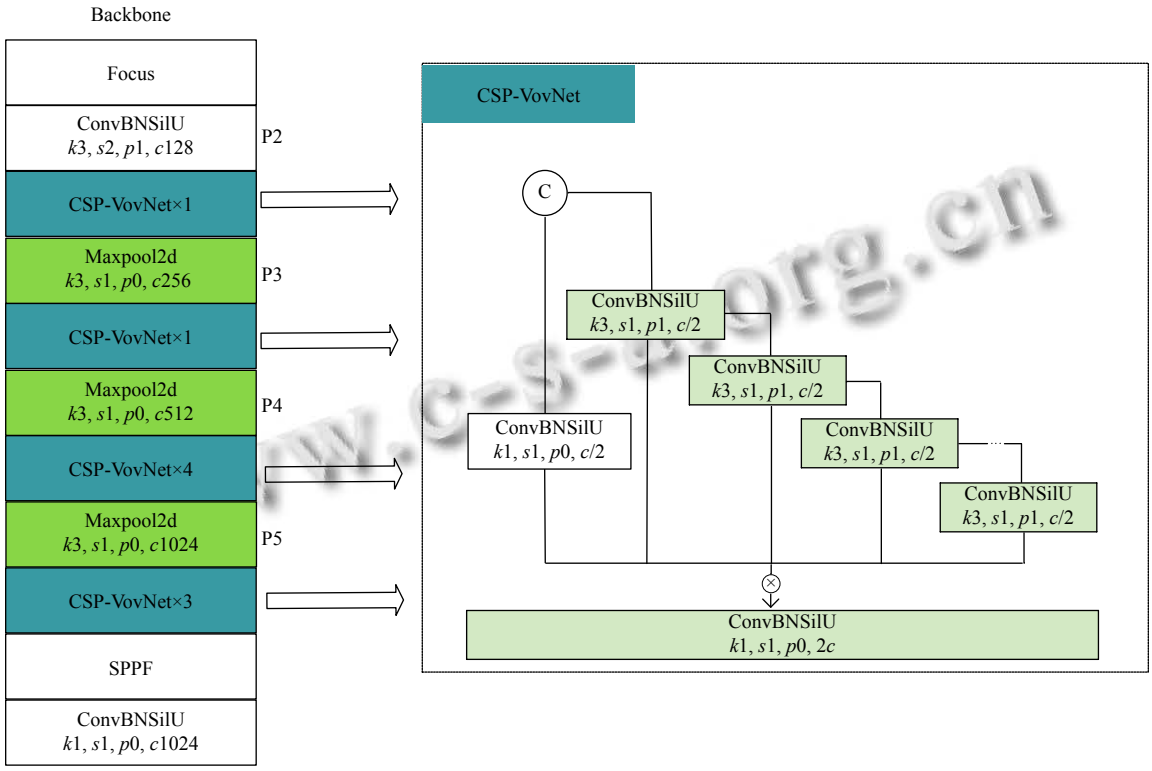


图 4 改进后的 YOLOv5 主干网络

1.3 可变形卷积的引入

交通道路数据中非机动车属于像素点较小的目标之一且交通摄像头拍摄的高度和角度大相径庭, 因此存在不同程度的模糊和形变, 为解决这类问题将可变

形卷积运用于提取目标特征的基础网络中.

1.3.1 可变形卷积结构 (DCN)

相对于普通卷积来说, 可变形卷积的卷积核的形状不是固定的几何形状, 可以根据图像中目标的内容进行自适应的改变. 相对于传统卷积, 其映射方式是十分规整统一的, 可理解为刚性映射, 大多数的目标检测任务中目标物体往往都具备非刚性结构, 但传统的卷积仅能用规整的方形结构来覆盖被测物体. 图 5 为可变形卷积不同表现形式.

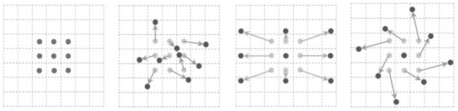


图 5 可变形卷积不同表现形式

而本文用到的可变形卷积, 能有效地映射目标区域, 其卷积核会根据目标物体内容进行不同形式的变化, 卷积区域可以尽可能地覆盖到被测非机动车辆的外观形状, 有利于获取更多有用的特征信息, 进一步实现更好的特征提取效果. 如图 6 所示.

(a) 普通卷积

(b)可变形卷积

图 6 普通卷积和可变形卷积效果

普通的 Conv2 通常分为两部分, 一是使用图 5 中普通卷积在输入特征映射X上进行采样处理, 二是对w加权的特征采样值进行求和运算. 卷积核的尺寸决定了感受野的大小. 假设R为一个 3×3 的卷积核, 则有:

$$R=\{(-1,-1),(-1,0),\cdots,(0,1),(1,1)\} \tag{1}$$

对输出映射特征图上的每个像素点P_0:

$$F(P_0)=\sum_{P_n\in R} w(P_n)*X(P_0+P_n) \tag{2}$$

在可变形卷积中, 卷积核R的形状可以根据偏转量ΔP_n来控制:

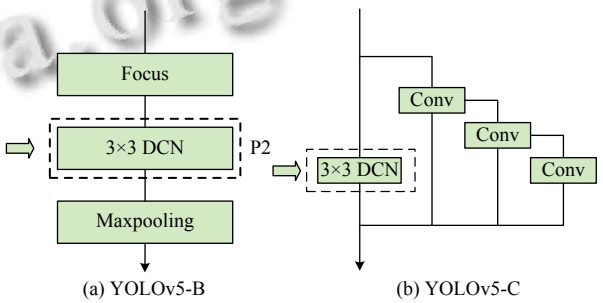
$$F(P_0)=\sum_{P_n\in R} w(P_n)*X(P_0+P_n+\Delta P_n) \tag{3}$$

其中, P_n为R中的每一个采样点.

本文将可变形卷积 (DCN) 运用到目标检测模型 YOLOv5 中去, 以此来提升对检测目标的几何变换的建模能力, 对原有的模型进行进一步的优化从而提高对此次非机动车检测的任务的检测效果.

1.3.2 3 种添加方式

本文根据主干网络结构设计出 3 种不同的卷积添加方式, 3 种方法对应 3 种模型分别记为: YOLOv5-A, YOLOv5-B, YOLOv5-C. YOLOv5-A: 在上述主干网络的 P2 层, 不同 CSPVovNet 结构中分别用 3×3 可变形卷积替换常规 3×3 的卷积及 CSP 结构中的 1×1 卷积. YOLOv5-B: 仅在网络的 P2 层加入可变形卷积, 如图 7(a). YOLOv5-C: 仅在网络的前两个 CSPVovNet 层加入可变形卷积, 如图 7(b). 两种添加模型如图 7.



(a) YOLOv5-B (b) YOLOv5-C

图 7 DCN 在网络中的添加方式

1.4 EIoU-YOLOv5s

YOLOv5 共采用 3 种损失函数分别为分类损失 (cls_loss), 定位损失 (box_loss) 和置信度损失 (obj_loss). 其中分类损失和置信度损失均使用了二元交叉熵损失函数 (BCEWithLogitsLoss) 进行该部分损失计算, 使用 CIoU_loss 作为 bounding box 回归定位损失. GIoU 的问题是使用闭包的面积减去并集的面积作为惩罚项, 这就导致了 GIoU 存在先扩大并集面积再优化 IoU 的走弯路的问题. CIoU 的问题是宽和高不能同时增大或者减小. 例如在图 8 中的第 2 行, anchor 是一个锚框, 它的宽和高均大于待检测物体, 但是在优化过程中它仍然会放大预测框的宽. 对比上面两个损失函数, EIoU 则拥有更快的收敛速度.

基于这个现象, EIoU 提出了直接对C_w和C_h的预测结果进行惩罚的损失函数, 它定义为

$$\begin{aligned}\mathcal{L}_{\text{EIoU}}&=\mathcal{L}_{IoU}+\mathcal{L}_{\text{dis}}+\mathcal{L}_{\text{asp}}\\&=1-IoU+\frac{\rho^2(b,b^{gt})}{c^2}+\frac{\rho^2(w,w^{gt})}{C_w^2}+\frac{\rho^2(h,h^{gt})}{C_h^2}\end{aligned} \tag{4}$$

其中, C_w和C_h分别是两个矩形的闭包的宽和高. 从式中可以看出, EIoU 将损失函数分成了 3 个部分, IoU 损失$\mathcal{L}_{IoU}$, 距离损失$\mathcal{L}_{\text{dis}}$, 边长损失$\mathcal{L}_{\text{asp}}$. 可以看出 EIoU 是

直接将边长作为惩罚项的, 这样也能一定程度上解决在文献 [20] 中分析的 DIoU_loss 可能的边长被错误放大的问题.

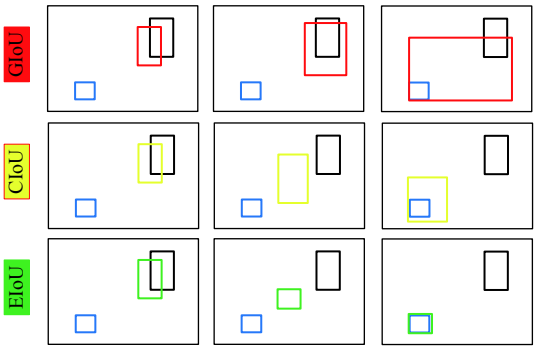


图 8 不同 loss 的多次迭代过程

2 实验对比

2.1 实验准备

为证明优化后模型的性能, 提取使用 Pascal VOC2007 开源数据集中的 motorbike 和 bicycle 这两类进行训练测试. 同时由于我国的交通环境的复杂, 现有的开源公共数据集不能满足本次实验全部需求, 故通过网络爬虫, 实地拍摄等方式获取大量有关非机动车的图片数据建立新的数据集, 该自制数据集中包含大量各类电动车, 自行车图片. 通过使用 labeling 工具人工方式进行标注, 统一标注为 nm-vehicle, 共使用 12 000 张图片作为本次检测识别任务的训练集和测试集, 其中用于训练图片 9 600 张, 用于验证测试图片 2 400 张, 比例设置为 8:2. 图 9 为自制非机动车数据集中部分图片.

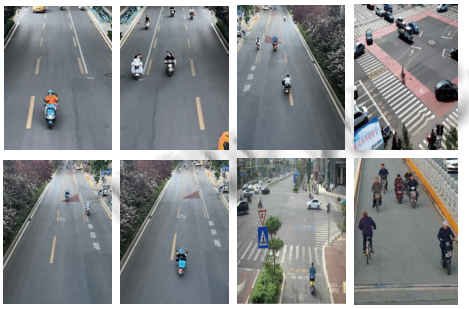


图 9 自制非机动车数据集

本次模型验证主要在 Ubuntu 系统上对 YOLOv5 基础源码上实验分析, 针对非机动车检测识别的问题对原有算法进行不同的优化和调试. 具体实验环境如表 1 所示.

本文实验使用到的评价指标如下: *mAP* (平均精度)、*precision* (精准率)、*recall* (召回率)、*mAP*@0.5 (*IoU*=0.5 时的平均精度)、*mAP*@0.5:0.95 (在不同 *IoU* 阈值 (*IoU* 在 0.5–0.95 之间, 步长为 0.5 的平均精度), *FPS* (图片每秒传输帧率). 各指标计算方式如下:

$$precision = \frac{TP}{TP+FP} \tag{5}$$

$$recall = \frac{TP}{TP+FN} \tag{6}$$

$$AP = \int_0^{recall} (precision) drecall \tag{7}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP \tag{8}$$

$$FPS = \frac{FrameNum}{ElapsedTime} \tag{9}$$

其中, *TP* 表示实际为正样本标签而且被预测为正样本的数量, *FP* 表示实际是正样本但被预测为负样本的数量, *FN* 表示实际为负样本标签但预测为正样本的数量. *N* 为检测类别数, *FrameNum* 表示处理图片的数量, *ElapsedTime* 表示处理图片所用的时间. 其中 3 种 *mAP* 的数值均越高越能说明该种模型性能一定程度上的优异性, *FPS* 数值的大小体现了该类模型处理图片帧的速度, 和其硬件配置有一定的关系, 同等配置情况下 *FPS* 值越大, 模型检测实时性更出色.

表 1 实验环境

名称	参数配置	名称	参数配置
编程环境	PyCharm	操作系统	Ubuntu
GPU	GTX1080TI	内存	32 GB
CUDA	10.2.0	cuDNN	7.6.6
CPU	i5-10600K	PyTorch-GPU	1.10.1

2.2 主干更换实验

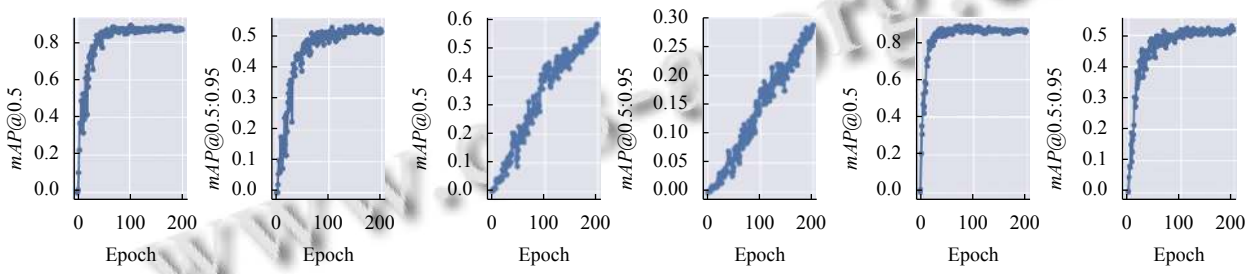
为了验证改进后模型的可行性和有效性, 本文在自制的非机动车数据集中分别进行 Faster R-CNN 算法, SSD 算法, YOLOv3 算法, YOLOv4 算法, YOLOv5s 算法和 YOLOv5s-MobileNetV3 算法验证, 将本文算法与上述流行的目标检测算法进行对照. 图片输入根据基础算法要求均调整为 640×640, 批次处理大小 batch_size 设置为 16, 初始学习率为 0.01, 衰减系数设置为 0.000 5, 优化器为 SGD, 训练次数设为 200 轮. 训练后其各项性能指标如表 2 所示 .

可见, 经过改进主干后的 YOLOv5s 网络在 *IoU*=0.5 的设定下达到几乎和 YOLOv4 相持平的 *mAP* 值, 仅落后 0.01, 远高于 SSD 等其他类型算法, 表 2 中 YOLOv5 系列算法无论从检测速度还是精度上看都超越其他非

YOLO 类算法. 本文算法在 *IoU*=(0.5:0.95)时其*mAP*有一个巨大的提升, 领先其他几种算法达到了 0.56. 同时, 我们改进的算法处理图片速度仅比原始算法略低, 仍然远高于检测性能几乎持平的 YOLOv4网络. 其改进后的参数量同样远小于 YOLOv5s 的参数量, 便于进行算法的移植部署. 对比上述实验中的不同主干网络, 本文提出并使用的改进 CSPVovNet 检测的效果和精度有一定的提升, 同时因其使用 Maxpooling 进行下采样和 concat 通道连接方式大大减小模型的推理时间, 同时便于梯度的反向传播. 图 10 为 YOLOv5s, YOLOv5s-M 及改进后 YOLOv5-CSPVovNet 算法训练 200 个 epoch 下的平均精度对比.

表 2　不同算法各性能指标

方法	Backbone	Parameters (M)	*mAP*@0.5	*mAP*@0.5:0.95	*FPS*
Faster R-CNN	ResNet50	60.2	0.82	—	11
SSD	VGG16	41.2	0.71	—	45
YOLOv3-SPP	Darknet53	63.0	0.79	0.44	29
YOLOv4	CSPDarknet53	244.3	0.90	0.54	22
YOLOv5s	CSPDarknet	7.3	0.86	0.51	48
YOLOv5s-M	MobileNetV3[21]	**3.7**	0.59	0.29	**65**
Ours	CSPVovNet	3.9	**0.89**	**0.56**	54



(a) YOLOv5s　(b) YOLOv5s-M　(c) YOLOv5-CSPVovNet

图 10　平均精度对比

在上述对比图中仅对 3 种不同主干网进行对比, 以此推断改进后的性能. 从图 10 不难看出使用 MoblieNet-V3 改进后的特征提取模块效果最差, 在 200 个 epoch 后精度仍然偏低, 其原 CSPDarknet 网络和改进后的 CSPVOvNet 网络在精度上不容易看出太大差别, 但其收敛速度改进后的网络明显优于原网络, 且在 *mAP*@0.5:0.95 这张图中原 YOLOv5s 网络精度随着 epoch 的增加后期开始出现下降趋势, 存在过拟合的现象.

2.3　DCN 嵌入对比实验

本部分的实验在公开数据集 Pascal VOC2007 和自制数据集上分别进行验证, 在 VOC2007 数据集中直接使用 motorbike 和 bicycle 两类进行训练测试, 在自制数据集仅划分 nm-vehicle 这一类进行训练测试. 实验均不使用预训练模型, 4 种测试算法训练过程均使用相同的参数配置, 输入图片大小为 640×640, epoch 设置为 200 轮, batch_size 设置为 16, 优化器为 SGD, 权重初始学习率设为 0.01, 衰减系数设置为 0.000 5, *IoU* 设定为 0.5, 随机划分训练和测试集比例为 8:2. 测试结果如表 3 所示. 通过表 3 可以看出 DCN 添加不同位置和不同数据集上的效果, YOLOv5-A 将 P2 层普通 3×3 卷积和前两个 CSP-VovNet 中的 1×1 卷积替换为 3×3 可变形卷积, 其性能与原 YOLOv5s 算法同比在 VOC2007 和自制非机动车量数据上均有明显下降, YOLOv5-C 在 VOC2007 上的精准率 *precision* 比原始算法提升了 4%, *mAP*提升了 2%, 在自制的数据集上同样有 0.88 的召回率 *recall*, 高于其他 3 种算法. 综上可以看出, 可变形卷积并非添加的越多越好, 除了会增大网络的复杂度的同时, 也一定程度上可能降低网络性能, 上述 YOLOv5-C 仅在 CSP-VovNet 层加入可变形卷积其性能表现最为优异, 提升幅度最大.

表 3　4 种不同 DCN 添加方式测试结果

算法模型	VOC2007			自制数据集		
	precision	*recall*	*mAP*	*precision*	*recall*	*mAP*
Original YOLOv5s	0.85	0.88	0.91	0.87	0.85	0.86
改进YOLOv5-A	0.82	0.84	0.83	0.78	0.82	0.81
改进YOLOv5-B	0.89	0.86	0.89	0.88	0.82	0.87
改进YOLOv5-C	0.89	0.87	0.93	0.83	0.88	0.86

2.4　消融实验

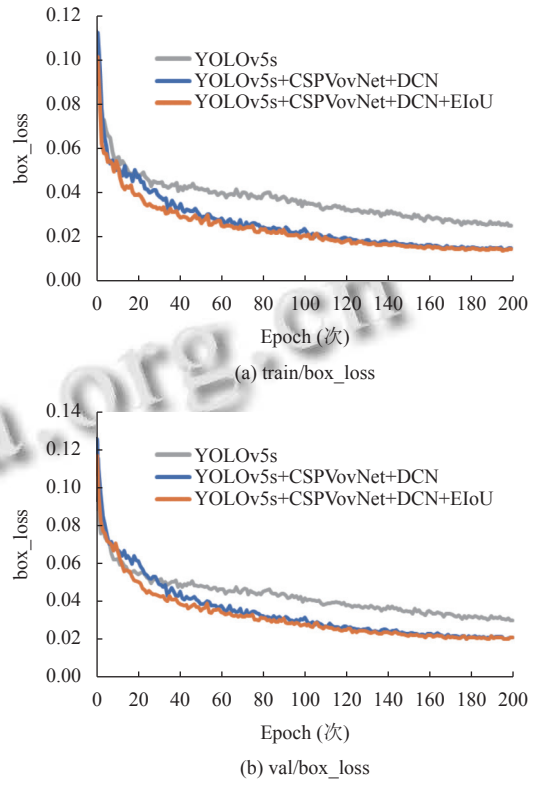
在原 YOLOv5s 算法的基础上在非机动车辆数据集进行消融实验, 验证各模块改进后的性能表现, 图片输入大小仍为 640×640, 批次处理大小 batch_size 设置为 16, 初始学习率为 0.01, 衰减系数设置为 0.000 5, 优化器为 SGD, 训练次数设为 200 次. 实验数据如表 4 所示. 通过表 4 中的实验 a (YOLOv5s), 实验 b (YOLOv5+CSPVovNet+DCN) 和实验 c (YOLOv5+CSPVovNet+

DCN+EIoU) 可以验证其损失函数更换为 EIoU_loss 效果, 图 11 为 3 种实验在训练集和验证集上回归损失的收敛结果.

表 4 消融实验

实验	CSPVovNet	DCN	EIoU	Parameters (M)	*mAP* (*IoU*=0.5) (%)	*FPS*
a	—	—	—	7.3	85.89	48
b	√	—	—	3.9	89.21	54
c	√	√	—	4.2	89.10	42
d	√	√	√	4.3	90.03	46

从图 11 中我们可以看到实验 b 和 c 对比原始的 YOLOv5s 算法其最终损失值更低, 效果更佳优越. 实验 c 的不同在于将实验 a 和 b 中的回归损失 CIoU 换成 EIoU, 可以清楚看出无论在训练集还是验证集上选取 EIOU 作为回归损失其收敛速度都更快, 性能更好. 由表 4 实验 a, b 对照看出 CSPVovNet 结构取代 CSPDarknet 作特征提取其平均精度*mAP*有显著的提升, 可以表明浅层特征在深层网络中很好的保存传递有利于提升模型的精度. 实验 b 和 c 对比虽然未能看出有较大的提升, 但图 12 可视化结果中可以看出其置信度一定的提升. 本文模型在自制的真实交通场景数据集训练验证, 最终改进后的网络模型其参数量仅有 4.3M, 对比原模型大幅度下降, 所需存储更小, 同时其每秒处理视频帧数达到 46 帧, 达到实时检测的要求, 其算法的时空复杂度能均够满足实际场景下的移动部署和检测实时性的要求, 达到在一些小型系统上部署的条件.



(a) train/box_loss

(b) val/box_loss

图 11 不同损失函数在 train 和 val 中训练结果

(a) (b) (c) (d)

图 12 可视化结果

3 结束语

本文提出了一种基于 YOLOv5 网络改进的非机动车检测识别算法, 将 YOLOv5 中特征提取网络 CSP-Darknet53 更换为 CSPVovNet 网络, 根据 VovNet 网络核心思想改进的 CSPVovNet 能够充分有效地加强特征传递, 改善因多层卷积造成的特征缺失, 提高输入目标特征图浅层特征复用的同时, 能够缓解因深层网络带来的梯度消失的问题. 针对交通监控场景下非机动车呈现的多角度多尺度问题, 引入可变形卷积, 通过实验对比选取最佳的添加位置, 提升了网络定位目标的能力. EIoU 用于目标的回归损失的计算, 直接将宽高差值作为惩罚项, 加快收敛速度的情况下, 同时解决了 CIoU 只能同时增减的缺点. 实验表明, 本文提出的改进算法相较于 YOLOv5s 原始的算法在 *mAP* 指标上提升了 4.14%, 能够一定程度上减少漏检, 误检的现象, 其算法的时空复杂度均能够满足实际交通场景下的移动部署和实时检测要求. 后续, 增加非机动车种类, 扩充数据集, 引入跟踪算法进一步解决遮挡问题, 改进用于交通场景下的非机动车各类违规行为识别判定.

参考文献

1 Tian DX, Zhang C, Duan XT, *et al*. The cooperative vehicle infrastructure system based on machine vision. Proceedings of the 6th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications. Miami: ACM, 2017. 85–89.

2 Drożdż M, Kryjak T. FPGA implementation of multi-scale face detection using HOG features and SVM classifier. Image Processing & Communications, 2017, 21(3): 27–44.

3 Dalal N, Triggs B. Histograms of oriented gradients for human detection. Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. San Diego: IEEE, 2005. 886–893.

4 Felzenszwalb PF, Girshick RB, McAllester D, *et al*. Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(9): 1627–1645. [doi: 10.1109/TPAMI.2009.167]

5 Girshick R, Donahue J, Darrell T, *et al*. Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. Columbus: IEEE, 2014. 580–587.

6 He KM, Zhang XY, Ren SQ, *et al*. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904–1916. [doi: 10.1109/TPAMI.2015.2389824]

7 Girshick R. Fast R-CNN. Proceeding of the 2015 IEEE International Conference on Computer Vision (ICCV). Santiago: IEEE, 2015. 1440–1448.

8 Ren SQ, He KM, Girshick R, *et al*. Faster R-CNN: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137–1149. [doi: 10.1109/TPAMI.2016.2577031]

9 Liu W, Anguelov D, Erhan D, *et al*. SSD: Single shot multibox detector. Proceedings of the 14th European Conference on Computer Vision. Amsterdam: Springer, 2016. 21–37.

10 Redmon J, Divvala S, Girshick R, *et al*. You only look once: Unified, real-time object detection. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas: IEEE, 2016. 779–788.

11 Redmon J, Farhadi A. YOLO9000: Better, faster, stronger. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu: IEEE, 2017. 779–788.

12 Redmon J, Farhadi A. YOLOv3: An incremental improvement. arXiv:1804.02767, 2018.

13 Bochkovskiy A, Wang CY, Liao HYM. YOLOv4: Optimal speed and accuracy of object detection. arXiv:2004.10934, 2020.

14 Dahiya K, Singh D, Mohan CK. Automatic detection of bike-riders without helmet using surveillance videos in real-time. Proceedings of the 2016 International Joint Conference on Neural Networks (IJCNN). Vancouver: IEEE, 2016. 3046–3051.

15 路雪, 刘坤, 程永翔. 一种深度学习的非机动车辆目标检测算法. 计算机工程与应用, 2019, 55(8): 182–188, 214. [doi: 10.3778/j.issn.1002-8331.1801-0199]

16 Zitnick CL, Dollár P. Edge boxes: Locating object proposals from edges. Proceedings of the 13th European Conference on Computer Vision. Zurich: Springer, 2014. 391–405.

17 叶佳林, 苏子毅, 马浩炎, 等. 改进 YOLOv3 的非机动车检测与识别方法. 计算机工程与应用, 2021, 57(1): 194–199. [doi: 10.3778/j.issn.1002-8331.2005-0343]

18 Huang G, Liu Z, van der Maaten L, *et al*. Densely connected convolutional networks. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 2261–2269.

19 Lee Y, Hwang JW, Lee S, *et al*. An energy and GPU-computation efficient backbone network for real-time object detection. Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Long Beach: IEEE, 2019. 756–760.

20 Zheng Z, Wang P, Liu W, *et al*. Distance-IoU loss: Faster and better learning for bounding box regression. Proceedings of the 2020 AAAI Conference on Artificial Intelligence. 2020, 34(7): 12993–13000.

21 Howard A, Sandler M, Chen B, *et al*. Searching for MobileNetV3. Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019. 1314–1324.

(校对责编: 孙君艳)