

路侧停车中目标检测与识别算法研究

唐金鹏¹ 李玲琳¹

TANG Jinpeng LI Linglin

摘要

针对现有路侧停车目标检测和识别方法在复杂场景中效率和精度的不足,提出了一种基于梯形卷积算法的优化方案,并将其集成至 YOLOv5 模型中。此算法通过选择性跳过部分不必要的卷积运算,减少了计算量,从而提升了模型推理效率。实验使用视频采集的图像数据集进行训练和验证,比较了传统卷积和梯形卷积在不同设备下的性能差异,改进后的模型在 CPU 环境下推理时间减少,且实现了较明显的速度提升。因此,算法能够在路侧停车场景中提高目标检测模型的实时性,为部署低成本高效率的智能交通系统提供了新方案。

关键词

路侧停车; 图像数据; 目标检测; YOLOv5

doi: 10.3969/j.issn.1672-9528.2024.12.008

0 引言

路边停车位在一定程度上能有效缓解停车位不足的状况。然而,当前路边停车收费方式却暴露出诸多弊端^[1]。当下较为常见的人工收费、地磁检测收费以及咪表收费等模式,均存在局限性,如成本高昂、部署工作繁杂以及大众接受程度低等^[2]。为解决路边停车收费的局限性,本文提出了一种基于深度学习和图像识别的方案,通过单目相机检测和跟踪车辆、识别车牌、识别车辆停车行为以及记录车辆停车时间段的方案,以解决路边停车收费问题。该研究具有部署方便、人工成本低、效率高和准确性高的优点。

路边停车的目标检测和识别依赖于物体检测和识别算法。早期算法主要依赖于传统的计算机视觉技术,如 Haar 级联分类器、HOG 特征和 SVM 分类器^[3-6]。这些方法在一定程度上能够实现车辆检测和识别,但在复杂场景和遮挡情况下效果有限^[7]。近年来,深度学习技术的崛起逐步改变了目标检测与识别的方法^[8]。

如卷积神经网络(CNN)在车辆检测中表现出色^[9]。许多基于 CNN 的目标检测模型,如 Faster R-CNN、YOLO (you only look once) 和 SSD (single shot multiBox detector), 已经被广泛应用于路侧停车检测中^[10]。在路侧停车场景中,设计目标检测和识别算法时需要考虑两方面。

一是多尺度检测:路侧停车场景中通常有多个不同尺度

的车辆,因此多尺度目标检测算法变得非常重要。部分方法通过特定的网络结构或金字塔特征来实现多尺度检测。

二是实时性和效率:实时性对于路侧停车管理非常关键。因此,一些轻量级的深度学习模型,如 Tiny YOLO, 被设计用于在嵌入式设备上实时运行^[11]。

本文提出一种梯形卷积算法,将其应用于 YOLOv5 中,使其能够达到与 YOLO 相同的准确率,但识别速度更快。

1 基本算法选择

在目标检测和分类领域有两类算法^[12]。一类是两阶段算法^[13],其算法实现分为两步:第一步提取感兴趣区域;第二步对感兴趣区域进行分类。较为常用的两阶段算法包括 R-CNN、Fast R-CNN、Faster R-CNN 等。另一类是单阶段算法^[14],只需推断一次,即可得到图像中所有感兴趣物体的位置和类别。YOLO 是一种应用较为广泛的一阶段算法。与两阶段算法相比,一阶段算法的优点比两阶段算法速度更快。缺点是其精度低于两阶段算法。此前通过将 YOLO 与快速 R-CNN 进行比较,后者在彼时效果较好^[15]。之后,使用 PASCAL-VOC 数据集进行比较。首先,比较计算速度。YOLO 算法的速度优于其他算法,速度可以达到 45 帧/s,其中快速 YOLO 甚至可以达到 155 帧/s,但是平均精度均值却从 63.4% 下降到 52.7%。然后将快速 R-CNN 模型与 YOLO 的最高地图进行比较。快速 R-CNN 的 mAP 达到 71.8%,而 YOLO 的 mAP 为 63.4%,两者差距不大。但 YOLO 比快速的 R-CNN 更具优势。YOLO 对背景的误判率(4.75%)远低于快速 R-CNN (13.6%)。

首先,YOLO 算法在速度上优势明显。除 YOLO 的实时

1. 湖南交通职业技术学院 湖南长沙 410132

[基金项目] 湖南省职业院校教育教学改革项目“基于混合式教学的高职电子信息类课程数字化转型实践研究”阶段性研究成果(ZJGB2023056)

检测效果, Faster R-CNN、SSD 等算法也可以满足实时检测的要求。然而, 这时不仅需要检测和分类目标, 还需要执行复杂的功能, 如跟踪和车牌识别。

其次, 虽然 YOLO 算法速度快, 却不适合检测重叠的对象。当然, 为了达到高 FPS, 不得不做出这样的计算。但对于检测路边停放的车辆影响不大, 因为根据摄像头拍摄观察, 车辆之间并没有大面积的重叠。所以, 选择 YOLO 算法, 其较低的准确率不会对车辆检测产生太大影响。此外, YOLO 是基于网格的目标检测, 每个网格检测 8 个目标, 所以 YOLO 在检测小目标方面优势少。

目前, YOLO、Faster R-CNN、SSD 等算法都能满足实时检测要求^[16-17]。然而, 在路边停车场场景中, 不仅需要对目标进行检测和分类, 还需要执行其他功能, 如对车辆跟踪及对车牌进行识别等。因此, 目标检测和识别算法的速度还需更快。

1.1 梯形卷积算法

因路边摄像器材其拍摄角度一般采用自上而下的进行拍摄, 摄像头的方向和道路的方向几乎一致, 所以距离摄像头远的车辆成像较小, 距离摄像头近的车辆成像较大。在此情况下, 同一个画面中的物体密度将呈非均匀分布。由于物体的密度不均匀, 如果在不同密度的区域进行相同的处理, 不可避免地会对性能造成损失。

针对这种情况, 本文提出了一种梯形卷积算法, 在路侧停车的场景下可达到与 YOLOv5 相近的精度, 并提高检测与识别速度。

1.2 梯形卷积算法的原理

传统的卷积方法是通过以固定的步长水平或纵向移动卷积核来执行卷积运算^[18-21]。如: 使用步长为 1 的 2×2 过滤器, 在一张高 4、宽 6 的图片中进行卷积。需要完成 15 次矩阵乘法。

梯形卷积算法则通过跳过一些指定的矩阵乘法进行卷积达到了减少卷积次数的目的。同样, 高 4、宽 6 的图片中进行卷积, 梯形卷积过程如图 1 所示。在该算法中, 执行一次完整的卷积层运算所需的卷积次数为 12 次, 减少了约 20% 的计算量。

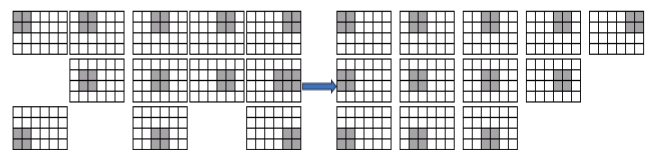


图 1 梯形卷积算法

1.3 梯形卷积的定位原理

基于 YOLO 的定位原理, 可以实现梯形卷积。YOLO 算法基于 CNN 进行, CNN 中多次卷积具备空间位置信息不变

的特点, 即在卷积过程中图片的尺寸减小了, 但是位置对应关系被保留。如图 2 所示, 第一张图中标有灰色的区域的值只会影响第一次卷积后第二张图中标有灰色的区域。第二张图片上用红色标记的区域只会影响第三张图片上用灰色标记的区域。梯形卷积就是基于此原理, 但在卷积过程中选择性跳过非重要的卷积运算, 既能保证识别精度, 又能提高识别速度。

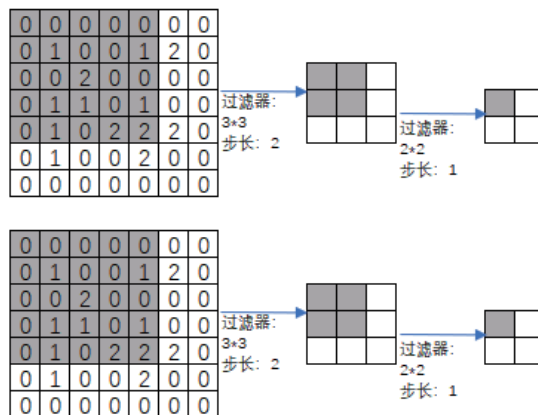


图 2 定位原理：空间位置信息不变性

1.4 梯形卷积算法的实现

YOLOv5 使用 PyTorch 作为深度学习框架^[22-23]。卷积层使用 Conv2d 类, Conv2d 类中应用 functional.py 中的 Conv2d 函数, 这个函数的实现依赖于 THNN 库。矩阵乘法的实现具体在 THNN 库的 generic 文件夹下的 SpatialConvolutionMM.c 文件中。

SpatialConvolutionMM.c 中的正向传播使用了 unfolded_copy 函数, 其具体实现在 SpatialConvolutionMM.c 的同级目录下的 unfolded.c 文件中, 要实现梯形卷积算法, 就要修改 unfolded_copy 函数。

unfolded_copy 函数的原理类似于 im2col 的原理, 其主要作用是将卷积转化为矩阵乘法。图 3 是乘法示例, 其中输入和输出 channel=batch=1, 卷积核大小 $kH=kW=3$, 输入尺寸 $inputHeight=inputWidth=1$, 步长大小 $dw=dh=2$, 填充尺寸 $padW=padH=1$ 。输出为: $outputHeight=outputWidth=3$

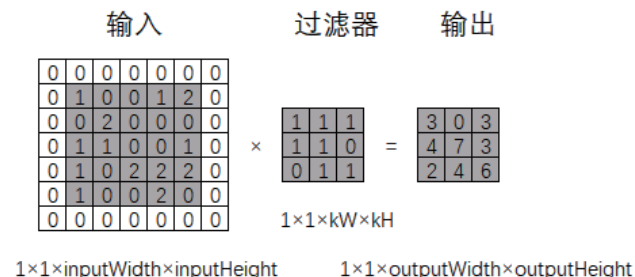


图 3 卷积示例

要将卷积转换为矩阵乘法运算, 需要将输入中的每个卷

积窗口转换为单独的列，这就是 `unfolded_copy` 所要实现的，如图 4 所示。

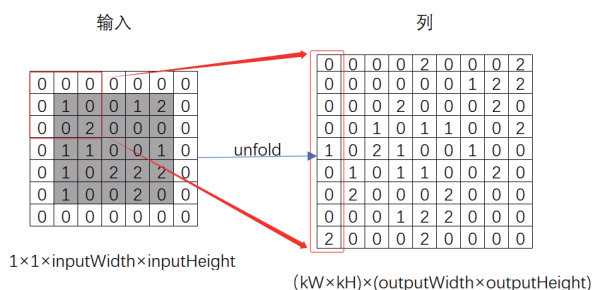


图 4 将卷积转化为矩阵乘法运算

图 5 中右边矩阵的每一列都是原始输入矩阵的卷积窗。转换后的矩阵大小为 $(n\text{Inputplane} \times kH \times kW) \times (\text{outputHeight} \times \text{outputWidth})$ 。调整大小为 $(n\text{OutputPlane}) \times (n\text{inputplane} \times kH \times kW)$ 可直接将矩阵乘以卷积核得到卷积结果，如图 5 所示。

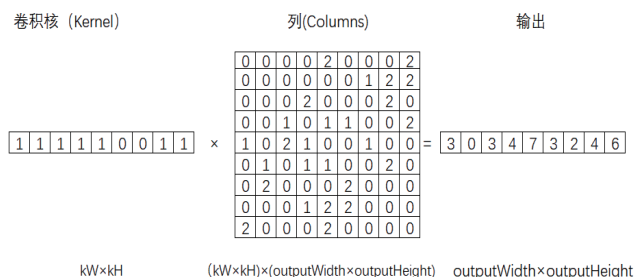


图 5 卷积核乘以矩阵

要实现基于 `unfolded_copy` 函数原理的梯形算法，需要在 `unfolded` 后对列进行修剪，在不影响检测算法精度的情况下删除不必要的列，然后与核进行矩阵相乘即可。例如，如果此处删除了第三列和第七列，卷积结果如图 6 所示。

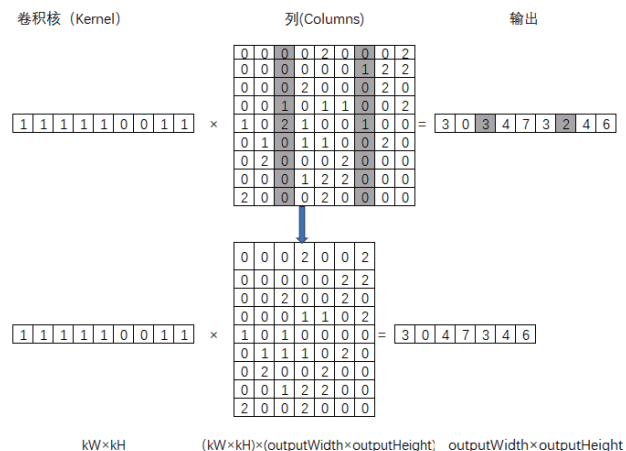


图 6 删除列示例

原本需要 9 次矩阵乘法运算，裁剪后只需要 7 次运算，减少了计算量。与此同时，需要注意输出维度的变化。此时输出后有 7 列，这样就无法还原 3×3 矩阵，影响后续操作。

为不影响后续计算，需要对原来丢失的地方进行填充，填充值使用其相邻位置的平均值，如图 7 所示。

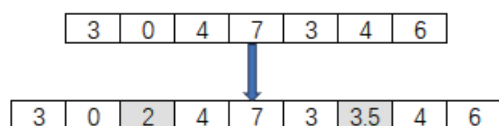


图 7 使用相邻位置的平均值填充缺失列

2 实验

此实验旨在比较不同卷积策略（包括梯形卷积）在目标检测任务中的性能差异。研究使用了特定的图像大小、卷积核参数和卷积策略，以进行性能比较和分析。

(1) 实验使用了一张大小为 $640 \text{ px} \times 640 \text{ px}$ 的 3 通道 RGB 图像，并通过 YOLOv5 的 Focus 算法将其分成了 12 个通道的 320×320 子图像。卷积核大小为 3×3 ，卷积核数量为 32。实验中的卷积操作被重复执行了 100 次，这是为接近 YOLOv5s 模型识别过程的算法复杂性。此外，为使结果更具可比性，创建 PyTorch 的 `Conv2d` 对象时使用了相同的权重。

(2) 输入图像大小：该研究中使用的输入图像大小为 $640 \text{ px} \times 640 \text{ px}$ 。

(3) 卷积策略：在研究中，卷积操作的策略是根据图像的不同部分采用不同的跳跃（skipped）次数。第一行卷积跳过一次，第二行卷积跳过两次，最后一行卷积跳过 N 次。这意味着在检测一张图像时，总共跳过了 $1+2+\dots+N$ 次卷积操作。

使用梯形卷积和使用传统卷积的速度对比如表 1 所示。

表 1 梯形卷积和传统卷积的比较

设备	方法	速度 /ms
CPU	传统卷积	983
	梯形卷积	843
GPU	传统卷积	8
	梯形卷积	8

从表 1 可以看出，虽然梯形卷积在 GPU 上的速度与传统卷积算法相同，但是在 CPU 上，梯形卷积相比传统卷积在速度上有明显提升。

3 模型训练

3.1 数据集

实验采用的数据集是多个路段上车辆停车行为的视频集合，从每个路段视频中随机截取 1000 张图片。其中，800 张图片作为训练集，200 张图片作为验证集。然后使用数据标注工具来标注图片。

每一帧的图片对应一个 `txt` 格式文件，这个文件中的每一行代表一辆汽车。一张图有几辆车，对应的 `txt` 文件就有几行。在每一行中，第一个数字代表类别，本文中类别只有

汽车，所以此值总为 0。后面 4 个数字分别表示边界框中心的横坐标与图像宽度 x 的比值、中心纵坐标与图像高度 y 的比值、边界框宽度与图像宽度 w 的比值以及边界框高度与图像高度 h 的比值。

为了验证梯形卷积的有效性，用传统卷积同样的方法训练了简化的 YOLOv5s 模型，然后对两个模型进行比较。实验过程，GPU 使用英伟达 RTX GeForce 2060，CPU 使用英特尔酷睿 i7 模型。实验结果见表 2。

表 2 梯形模型与传统模型比较

方法	GPU 速度 /s	CPU 速度 /s	mAP@0.5	mAP 0.5:0.95
梯形	0.018	0.046	0.727	0.486
传统	0.018	0.054	0.731	0.489

可以看出，使用梯形卷积可以在丢失少量准确性的前提下提高了推理速度。精度损失约为 0.55%，在 CPU 下推理时间减少约 14%，因此梯形卷积对 CPU 的速度提升效果明显。

4 结论

本文提出了旨在提高模型推理速度的梯形卷积算法。利用数据集对使用梯形卷积的模型的效果进行训练和分析，与不使用梯形卷积的模型进行比较。通过实验结果可以看到，在路侧停车检测场景中，准确性基本保持不变的前提下，应用梯形卷积算法模型的速度有所提高。

参考文献:

[1] 王殿海,李逸文,蔡正义.收费及奖励策略对通勤出行方式选择的影响分析[J].交通运输系统工程与信息,2024,24(2): 1-12.

[2] 林琳,张杰.动态停车收费标准对居民出行方式的影响:以宁波市为例[J].交通科技与管理,2024,5(3): 11-13+4.

[3] 周旭东,李毅,朱尧于,等.基于计算机视觉技术的桥梁管养应用综述[J].公路,2021,66(12): 389-394.

[4] 刘宽,郎磊.轻量化 SSD 目标检测方法研究[J].湖北民族大学学报(自然科学版),2021,39(4): 418-424.

[5] 马科伟,张锬石,康宇航,等.移动机器人中视觉里程计技术综述[J].计算机工程,2021,47(11): 1-10.

[6] 孙龙龙,王其宽,施凯,等.基于知识图谱的建筑安全领域计算机视觉研究综述[J].安全与环境工程,2021,28(2): 44-49.

[7] 周飞,郭杜杜,王洋,等.基于改进 YOLOv8 的交通监控车辆检测算法[J].计算机工程与应用,2024,60(6):110-120.

[8] 孔煜,彭佳强,张杰,等.面向低光照环境的车辆目标检测方法[J/OL].湖南大学学报(自然科学版),1-9[2024-06-19].
<http://kns.cnki.net/kcms/detail/43.1061.N.20240618.1345.002.html>.

[9] 崔梦举.交叉口场景下机动车检测算法研究[D].北京:北方工业大学,2024.

[10] 江岫,殷时蓉,罗天洪,等.基于改进 YOLOv4 的多目标车辆检测算法[J].计算机工程与设计,2024,45(4): 1181-1188

[11] 高山.基于目标检测的高速公路隧道预警系统[J].西部交通科技,2024(10): 201-202.

[12] 陈懿惘,李万益,翁汉锐,等.基于深度学习的两阶段目标检测算法综述[J].信息与电脑(理论版),2023,35(14): 112-114.

[13] 王琪,贾洪飞,芮一康.高密度场景下基于车载图像的多目标检测[C]//2024 世界交通运输大会(WTC2024)论文集(交通工程).南京:东南大学交通学院,2024: 8.

[14] 王量子,黄妙华,刘若璿,等.改进 PointPillars 和 Transformer 的路侧两阶段三维目标检测算法[J].激光与光电子学进展,2024,61(18): 413-422.

[15] JOSEPH R, SANTOSH D, ROSS G, et al. You only look once: unified, real-time object detection[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway:IEEE, 2016:779-788.

[16] 古佳欣,陈高华,张春美.YOLOv8-DEL: 基于改进 YOLOv8n 的实时车辆检测算法研究[J/OL].计算机工程与应用,1-13[2024-06-26].
<http://kns.cnki.net/kcms/detail/11.2127.tp.20240926.1133.010.html>.

[17] 张绪德,李康,唐厚炳.弱光照条件下道路交叉口实时车流量自动检测研究[J].激光杂志,2024,45(9): 238-243.

[18] 李创,刘宗林,刘胜,等.快速卷积算法的综述研究[J].计算机工程与科学,2021,43(10): 1711-1719.

[19] 方标,黄高明,高俊.多通道盲反卷积算法综述[J].信号处理,2013,29(6): 712-722.

[20] 杨沐津.多 GPU 环境下三点相关函数算法研究[D].天津:天津大学,2012.

[21] 张蕾,常丽媛,樊冬梅.图像恢复盲解卷积之文献综述[J].科技资讯,2012(21): 202.

[22] 毛少华,王文东.基于深度学习的 YOLO 系列物体检测算法研究综述[J].延安大学学报(自然科学版),2024,43(2): 88-95.

[23] 王鑫杰,王吉平.YOLO 目标检测算法综述[J].广西物理,2024,45(2): 50-53.

【作者简介】

唐金鹏(1975—),男,湖南冷水江人,硕士,教授,研究方向:计算机应用、智能交通。

(收稿日期:2024-08-02)