

城市轨道交通小型障碍物检测

张林¹, 沈拓^{1,2}, 张轩雄¹

(1. 上海理工大学 光电信息与计算机工程学院, 上海 200093;

2. 同济大学 铁路基础设施和系统安全上海重点实验室, 上海 201804)

摘要: 障碍物是影响列车行车安全的重要因素, 小型障碍物由于体积小, 在检测中容易被遗漏。针对上述问题, 提出了一种基于卷积神经网络的自动化检测算法。算法首先通过数据增强策略平衡样本数量, 然后使用卷积网络进行特征融合, 很好地结合了小型障碍物的位置信息和语义信息。实验结果表明, 算法对列车前方的小型障碍物有良好的检测效果, 能较好地实现小型障碍物的探测。

关键词: 深度学习; 卷积神经网络; 小型障碍物; 数据增强; 特征融合; 轨道交通

中图分类号: T 520 **文献标志码:** A

Detection of small obstacles in urban rail transit

ZHANG Lin¹, SHEN Tuo^{1,2}, ZHANG Xuanxiong¹

(1. School of Optical-Electrical and Computer Engineering, University of Shanghai of Science and Technology, Shanghai 200093, China; 2. Shanghai Key Laboratory of Rail Infrastructure Durability and System Safety, Tongji University, Shanghai 201804, China)

Abstract: Obstacles are an important factor affecting train operation safety. Due to little size, small obstacles can be ignored easily. To solve above problems, an automatic detection algorithm had been proposed based on convolutional neural network. Data augmentation was used in this algorithm to balance the sample size and then convolutional neural network is used to feature fusion, combining the location information and semantic information of small obstacles. The experimental results show that the algorithm mentioned has good detection effect on small obstacles.

Keywords: deep learning; convolutional neural network; small obstacles; data augmentation; feature fusion; rail transit

近年来, 随着国内各个城市轨道交通的快速发展, 自动化运营成为轨道交通的发展方向。自动检测列车前方的障碍物, 从而提前制动, 降低行车风险非常重要。因此, 国内外专家采取了不同的方法来研究列车前方的障碍物检测。

列车运行轨道上的障碍物检测一直都是研究热点, 早在 20 世纪 90 年代, 日本在其新干线上使用相机作为图像采集工具, 利用模板匹配的方法, 检测列车前方的障碍物^[1], 但是, 其需要依赖先验信息。之后, 众多学者提出通过多数据融合的

收稿日期: 2021-01-04

基金项目: 国家自然科学基金资助项目 (U1734211)

第一作者: 张林 (1996-), 男, 硕士研究生。研究方向: 目标检测。E-mail: slg_zhanglin@163.com

通信作者: 张轩雄 (1963-), 男, 研究员。研究方向: 微电子机械系统。E-mail: xuanxiongzhong@163.com

方案^[2], 提高障碍物检测的准确性。随着深度学习的发展, 使用神经网络进行障碍物检测已经成为目标检测方向不可或缺的一个分支, 在自动驾驶、行人检测等诸多场景都获得了大规模的应用^[3-4]。虽然这些方法对于大型障碍物有着良好的检测准确率, 但是, 对于小型障碍物的检测很不理想。

本文针对小型障碍物检测效果不理想的问题, 提出一种基于卷积神经网络 (convolutional neural network, CNN) 的检测算法。

1 数据采集及处理

1.1 数据采集

城市轨道交通的运行环境复杂多变, 列车既可能运行在地面, 也可能运行在隧道, 同时还面临着雨雪、雾霾等恶劣天气的影响, 所以, 直接用其他通用的数据来训练模型不太合适。本文通过摄像机、网络爬虫等方式, 建立特定的数据集来训练模型, 并通过实验来比对各种数据集的效果。

通过以上多种方式, 共采集图像 3 812 张。为了提升模型的泛化能力, 数据集包含不同环境下的图像, 例如: 隧道内的图像、地面运行时的图像、光污染时的图像及雨雪雾霾天气下的图像, 如图 1 所示。

在采集到的数据集中, 依据 COCO 数据集定义大小型障碍物的标准^[5], 对数据集中大小型障碍物的分布情况进行统计, 统计结果如表 1 所示。

从表 1 中的统计结果来看, 大小型障碍物在不同场景下的分布极不均匀, 小型障碍物的数量

表 1 障碍物的分布

Tab.1 Distribution of obstacles %

环境	大型障碍物	小型障碍物	不含障碍物
白天	58	14	28
黑夜	40	13	47

非常少, 小型障碍物对模型训练时的损失函数贡献小, 导致小型障碍物的检测率低下^[6]。所以, 本文针对小型障碍物数量少的问题, 利用数据增强策略, 提升小型障碍物的数量。

1.2 数据增强

小型障碍物检测性能低下, 主要由两个原因导致^[5]: a. 只有少数图片包含小型障碍物; b. 即使图中有小型障碍物, 但是, 小型障碍物在整幅图中的占比很小。针对这两大问题, 采用过采样的方法, 对图片中的小型障碍物进行过采样, 同时利用复制、粘贴的方式, 提升小型障碍物的数量。

针对小型障碍物样本数量非常少的问题, 通过简单的复制、粘贴策略, 改变粘贴规则的方式来模拟实际的场景, 提升小型障碍物的检测率。针对图中所有的小型障碍物, 在随机位置复制、粘贴多次, 同时保留原始的样本。在对小型障碍物进行粘贴之前, 需要对小型障碍物的状态作相应的规则限制。首先对待粘贴障碍物的大小进行随机的放大或者缩小 20%, 同时将待粘贴障碍物随机旋转 15°; 其次, 粘贴到图中的障碍物不能覆盖到图中其他障碍物; 最后, 粘贴到图中的障碍物必须距图像边缘 5 个像素以上。部分图样粘贴效果如图 2(a) 和 2(b) 所示。

在解决了小型障碍物数量少的问题之后, 大小型障碍物样本不均的问题依然没有得到有效的解决。通过采样法来解决样本不均的问题, 采样法不仅能从数据的角度改变模型的阈值, 而且能改变模型收敛优化等一系列过程。常见的采样方法有欠采样和过采样两种^[7]。欠采样通过在负样本中选取一些样本与正样本进行结合, 从而弱化数据集中负样本带来的影响, 但是, 欠采样容易造成数据偏差、模型过拟合等一系列问题。过采样与欠采样正好相反, 在正样本中选取一些样本与负样本进行结合, 强调正样本所带来的影响, 但是对于标记错误的正样本, 会成倍地放大噪声。

针对以上问题, 采用 SMOTE (synthetic minority over-sampling technique) 算法^[8], SMOTE 属于过采样方法, SMOTE 利用插值法为少数类人工合成新样本, 然后加入到数据集中, 从而解决样本分布

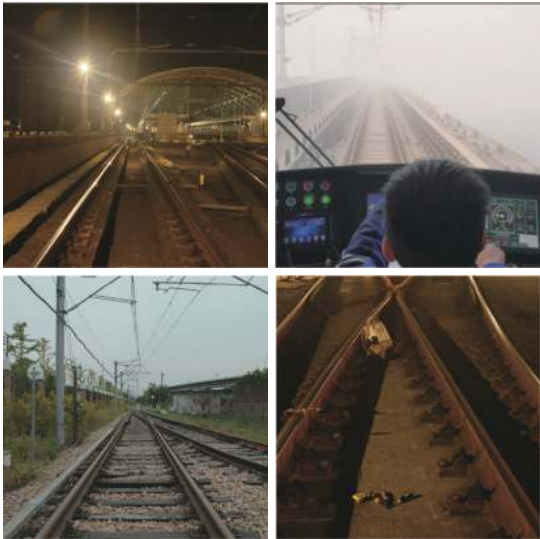


图 1 不同环境下的轨道图像

Fig. 1 Track images in different environments



图 2 粘贴后的效果图

Fig.2 Pasted figure

不均的问题。SMOTE 算法流程如下：

- a. 针对少数类的样本 x ，以欧式距离 (式 (1)) 为标准计算该样本到同类样本 x_i 的距离，获取其 k 个近邻样本；
- b. 根据样本不平衡的比例，确定采样的倍率 N ，对于每一个少数类样本 x ，从其 k 个近邻中选取若干个样本，假设选取的近邻样本数为 x_n ；
- c. 对于每个样本选出的 x_n ，分别与原样本按照式 (2) 构建新的样本 x_{new} 。

$$L_2(x_i, x_j) = \left(\sum_{l=1}^n |x_i^{(l)} - x_j^{(l)}|^2 \right)^{1/2} \tag{1}$$

$$x_{\text{new}} = x + \text{rand}(0, 1) |x - x_n| \tag{2}$$

式中： L 表示样本之间的欧式距离； l 表示样本所在的维度； rand 函数表示在数字 0~1 之间选择一个随机数。

经过 SMOTE 算法进行数据平衡之后，再次统计数据集中大小型障碍物的分布，结果如表 2 所示。

表 2 采样结束后的障碍物分布			
Tab.2	Obstacle distribution after sampling		%
环境	大型障碍物	小型障碍物	不含障碍物
白天	45	68	7
黑夜	37	72	18

从表 2 中可以清晰地看出，经过 SMOTE 和复制、粘贴策略之后，大小型障碍物样本的占比保持一致，同时不含障碍物的图片比例大幅下降，样本平衡良好。

2 特征融合

卷积神经网络是一种深度前馈式的神经网络，可以对图像特征信息进行有效提取，同时因其模型的复杂度低、权值参数少等特征，在图像信息提取上具有良好的效果。本文基于卷积神经网络提出自动化检测算法，并针对小型障碍物检测率低的问题作相应的算法改进。小型障碍物由于体积小，在特征提取的过程中信息容易丢失，导致模型对于小型障碍物的检测率不高。特征融合的主要目的就是在卷积神经网络提取小障碍物特征的过程中，使小型障碍物低层的位置信息和高层的语义信息相融合，从而提升小型障碍物的检测率。本文基于原始的 Faster RCNN (fast region based CNN) 骨干网络^[9]加入 FPN (feature pyramid network) 网络^[10]，使得整个网络模型中，低层的特征拥有高层的语义，高层的特征拥有低层的位置信息，大大提升了小型障碍物的检测率。

2.1 Faster RCNN

卷积神经网络应用在目标检测领域从 RCNN^[11] (region CNN) 到 Fast RCNN^[12]，再到 Faster RCNN。RCNN 算法预先找到图中可能出现目标的区域，即候选区域 (region proposal)，然后再进行目标检测，大大提升了目标检测率，但是，由于存在大量的重复计算，所以，RCNN 的检测速度很慢。Fast RCNN 针对 RCNN 每一个候选区域都要经过 CNN 进行特征提取，提出共享卷积层，直接输入一张图片，一次性提取所有的特征，在保证检测准确率的同时，大大提升了检测的速度。但是，Fast RCNN 在挑选候选网络时，使用的是选择性搜索，非常耗时，这时 Faster RCNN 提出使用单独的一个区域候选神经网络 (region proposal network, RPN) 来找到候选框，并将 RPN 直接放在卷积层的后面，在保证检测精度的同时，大大提升了目标

检测的速度, 因此, 整个 Faster RCNN 网络的性能取决于 RPN 网络的性能。

RPN 网络的作用就是将区域候选的过程融合到 Faster RCNN 网络框架中, RPN 网络和 Faster RCNN 网络共用卷积网络, 减少网络整体的计算量。RPN 网络提取候选区域的原理如图 3 所示。

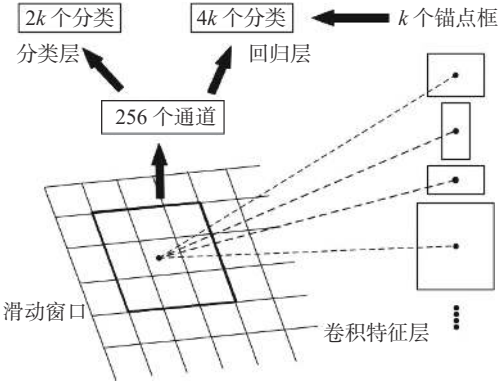


图 3 候选区域网络
Fig.3 Region proposal network

RPN 网络在卷积出来的特征层上建立滑动窗口, 以图中的每个点为中心, 提取出 9 个大小不同的候选框。利用滑动窗口进行特征提取之后, 得到一个 256 维的特征向量, 然后将此向量输入到全连接层, 输出每个候选框分类的得分及每个候选框的坐标。

2.2 FPN 网络

在卷积神经网络中, 不同层级的神经元拥有不同大小的感受野。低层特征拥有更小的感受野, 更加关注图片中的位置信息, 相反高层特征拥有更大的感受野, 更加关注图片的语义信息。小型障碍物在图片中占比较小, 通过卷积神经网络提取到的语义信息有限, 因此, 需要更加关注低层的位置信息。而在原始的 Faster RCNN 网络框架中, 对于目标的检测统一使用顶层特征显然不合适, 所以, 本文利用 FPN 网络融合小型障碍物的位置信息和语义信息, 提升小型障碍物的检测率。

FPN 网络主要分为两个部分, 一部分是自下而上的卷积, 另一部分是自上而下的采样和横向连接部分。自下而上的部分是主干网络 Resnet 卷积过程中的一部分。在 Resnet 中, 本文将输出同样特征映射的多个层归类成一个阶段。在每个阶段里, 最后一层拥有最丰富的特征信息, 所以, 选择最后一层作为每个阶段的特征输出。通过 Resnet 的框架结构, 将整个特征金字塔分为 5 个部分, 由于第一层卷积占用大量内存, 所以, 将它单独拿出来看, 剩下的 4 个部分根据特征的输出分成 4 个部分 {C2, C3, C4, C5}, 每一阶段相对于

输入图片的步长分别是 {4, 8, 16, 32}。

自上而下的部分使用上采样的方法将顶层的特征扩大到和自下而上部分同一阶段的大小, 横向连接利用高层的语义信息和低层的位置信息, 将多种特征信息融合起来, 使得低层特征拥有高层的语义信息, 高层特征拥有低层的位置信息。图 4 展示了特征金字塔网络的结构设计。

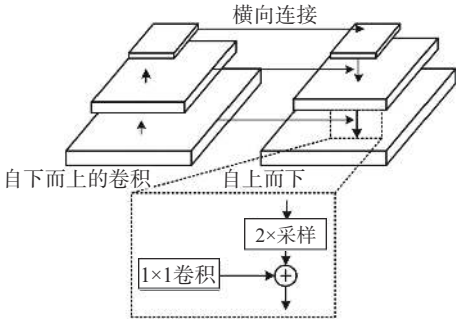


图 4 特征金字塔网络
Fig.4 Feature pyramid network

从图 4 中可以看出, 对于顶层的特征, 采用了一个上采样的方法来融合相关阶段的特征, 然而不同层之间的通道维数不一致, 所以, 采用一个 1x1 的全卷积网络来降低通道的数目, 使得不同层之间可以相互连接。针对上采样, 采用了最近邻插值算法, 算法公式为

$$X_{src} = X_{dst} \left(\frac{W_{src}}{W_{dst}} \right) \quad (3)$$

$$Y_{src} = Y_{dst} \left(\frac{H_{src}}{H_{dst}} \right) \quad (4)$$

式中: X_{src} 和 Y_{src} 分别是原始像素矩阵的索引; X_{dst} 和 Y_{dst} 是采样过后像素矩阵的索引; W 和 H 是像素矩阵的大小。

公式完全基于图像的像素矩阵, 为了消除上采样带来的重叠影响, 另外采用了一个 3x3 的卷积神经网络, 作用在特征融合的阶段, 而且每一个阶段都有各自的回归器和检测器, 从而达到多尺度的检测。

2.3 特征融合网络

在 Faster RCNN 的框架中, 区域候选网络作为一个独立的检测器, 使用一个 3x3 的滑动窗口在主干网络的特征层上抽取目标的特征, 然后将抽取到的特征送入 2 个 1x1 的卷积网络中进行分类和回归。分类器负责区分图像中的背景和障碍物, 回归器负责返回那些还有障碍物的区域。

针对 FPN 使用单一尺度的特征, 将 FPN 和 RPN 进行结合, 如图 5 所示, 在每层增加一个

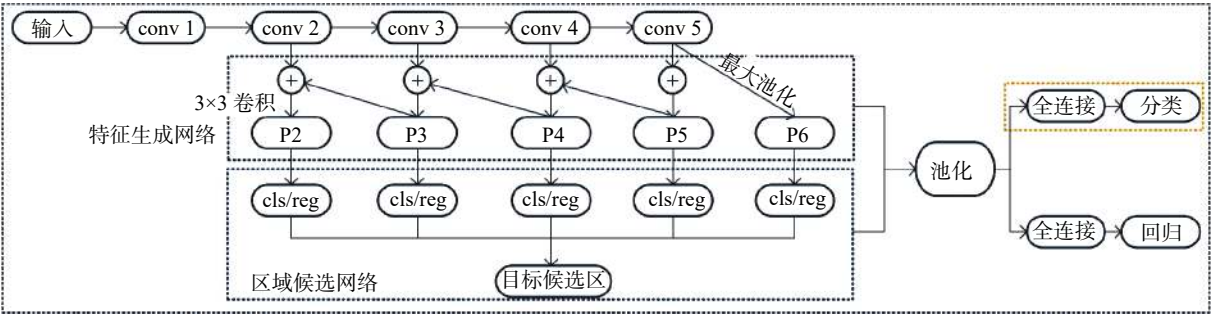


图5 特征融合网络结构图

Fig.5 Structure diagram of feature fusion network

3×3 的卷积和一个 1×1 的卷积，并且低层的特征由高层特征采样得到。对于每一层，均采用独立的检测器和回归器，所以，在每个具体的层上面，不需要多种尺度的锚框，每一阶段的锚框的大小被固定，{P2, P3, P4, P5}阶段的锚框大小分别是{32², 64², 128², 256²}个像素，每个阶段锚框的比例均是{1:2, 1:1, 2:1}。所以，每一窗口至少有 12 个锚框。经过以上的改进，特征金字塔的每一个特征既拥有障碍物低层的位置信息，也拥有高层特征的语义信息，模型的性能得到了很大的改善。同时，本文使用场景不需要确定障碍物的具体类型，所以，将原 Faster RCNN 网络的细分类分支进行裁剪(如图 5 黄色区域)，以此来达到加速检测的目的。

3 实验结果与分析

为了验证算法在不同环境下的有效性，在测试时选择不同环境下的多种图片，包括隧道环境、地表环境、轨道交叉口环境和雨雪雾霾等场景，通过对环境背景进行分类，分成白天和黑夜两大类。同时与当前主流的算法 YOLOv3^[13] 及 SSD^[14] 进行对比，验证实验的有效性。

3.1 实验环境

Faster RCNN 采用公共的 Resnet 作为网络主干，使用随机梯度下降的方法，在 ubuntu16.04 系统，Intel i7 4 核 8 线程的处理器、32 GB 内存和 Nvidia GTX1080TI 显卡，CUDA9.0 的 GPU 加速库环境下进行实验。整个训练过程是端到端的，每一批次只有一张图像，并且每一张图像只生成 2 000 个 ROIs。训练的学习率设置为 0.02，权重为 0.000 1，动量为 0.9，以此来加速收敛。

3.2 实验结果

通过不同环境下算法检测的准确性和算法检

测的速度来衡量实验结果的有效性。不同算法的实验结果如表 3 所示。

表 3 不同算法检测结果对比

Tab.3 Comparison of detection results with different algorithms

算法	大型障碍物 检测率/%	小型障碍物 检测率/%	平均 检测率/%	检测速度(FPS)
Faster-RCNN	63.5	32.6	48.05	12
SSD	67.7	34.3	51	49
YOLOv3	58.8	29.4	44.1	92
本文的方法	72.5	38.6	55.6	28

从表 3 中不难看出，本文的算法在检测率上明显高于其他算法，同时也验证了小型障碍物在检测时融合位置信息和语义信息的有效性。但是，本文的算法在检测速度上明显低于其他算法，在以检测率为主的基础上，本文的算法依然具有一定的优势。

固定抽取 4 种不同环境的测试图片各 20 张，使用本文的算法检测，计算各个复杂环境下的检测率，实验结果如表 4 所示。从表 4 中可以直观地看出，待检测的图片存在噪声干扰时，算法的检测性能下降得较快。尤其在存在光污染的情况下，算法的检测性能直接降到原有的一半。实际检测的效果图如图 6(a) 和 6(b) 所示。

表 4 算法在不同环境下的效果

Tab.4 Effect of the algorithm in different environments

环境	检测率/%	平均检测速度(FPS)
隧道	59.8	26
雨雪	47.9	26
光污染	35.6	26
地表	65.4	26



图 6 检测结果图

Fig.6 Results of detection

4 结束语

针对当前轨道列车前方障碍物检测，提出了使用卷积神经网络方法，有效地提升了检测列车前方障碍物的效率，为列车自动化运营提供了一种可行的途径。同时针对原有目标检测算法的小型障碍物检测率低的问题，提出特征融合策略，利用特征金字塔网络，将低层的位置信息和高层的语义信息进行结合，大幅提升了小型障碍物的检测率。但是，本文的方法也有不足，在检测速度上未能达到实时的要求，在后续的研究中将继续提升算法的性能。

参考文献:

[1] SASAKI A, TANAKA Y, ICHIKURA T. On-board self-regulating automated vehicle operation system[C]//1999 IEEE 49th Vehicular Technology Conference (Cat. No. 99CH36363). Houston: IEEE, 1999: 2298–2302.

[2] KARADUMAN M. Image processing based obstacle detection with laser measurement in railways[C]//2017 10th International Conference on Electrical and Electronics Engineering (ELECO). Bursa: IEEE, 2018: 899–903.

[3] LIU W, LIAO S C, REN W Q, et al. High-level semantic feature detection: a new perspective for pedestrian detection[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019: 5182–5191.

[4] YURTSEVER E, LAMBERT J, CARBALLO A, et al. A survey of autonomous driving: common practices and emerging technologies[J]. IEEE Access, 2020, 8: 58443–58469.

[5] KISANTAL M, WOJNA Z, MURAWSKI J, et al. Augmentation for small object detection[EB/OL]. (2019-02-19) [2021-01-04] <https://arxiv.org/abs/1902.07296v1>.

[6] CHEN Y K, ZHANG P Z, LI Z M, et al. Stitcher:feedback-

driven data provider for object detection[EB/OL]. (2021-03-14) [2021-01-04] <https://arxiv.org/abs/2004.12432v1#:~:text=Inspired%20by%20this%20finding%2C%20we%20present%20Stitcher%2C%20a,stitched%20into%20the%20same%20size%20to%20regular%20images>.

[7] MASTERS B R, GONZALEZ R C, WOODS R. Book review: digital image processing, third edition[J]. Journal of Biomedical Optics, 2009, 14(2): 029901.

[8] CHAWLA N V, BOWYER K W, HALL L O, et al. SMOTE: synthetic minority over-sampling technique[J]. Journal of Artificial Intelligence Research, 2002, 16: 321–357.

[9] REN S Q, HE K M, 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] LIN T Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 936–944.

[11] 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. Columbus: IEEE, 2014: 580–587.

[12] GIRSHICK R. Fast R-CNN[C]//2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015: 1440–1448.

[13] REDMON J, FARHADI A. YOLOv3: an incremental improvement[EB/OL]. (2018-04-08) [2021-01-04] <https://arxiv.org/abs/1804.02767>.

[14] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot multibox detector[M]//LEIBE B, MATAS J, SEBE N, et al. Lecture Notes in Computer Science. Cham: Springer, 2016, 9905: 21–37.