Computer Science and Application 计算机科学与应用, 2021, 11(1), 207-216
Published Online January 2021 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2021.111021

Hans汉斯

一种基于YOLOv3的道路多目标检测方法

傅景超[1,2]，苏庆华[1,2*]，张娣娣[1,2]，李俊韬[1,2]

[1]北京物资学院，北京
[2]北京市智能物流实验室，北京
Email: *qinghuasu@126.com

收稿日期：2020年12月27日；录用日期：2021年1月22日；发布日期：2021年1月29日

摘　要

道路场景中多目标检测对车辆的自动驾驶和辅助驾驶的智能化有着重要意义。现有道路目标检测算法存在检测精度低、实时性较差和目标漏检等问题。针对这些问题，本文构建一种基于YOLOv3的高精度、低时延、低漏检的道路多目标检测方法。通过对YOLOv3目标检测原理进行深入分析，基于迁移学习(Transfer Learning)的思想，在经过预训练的模型上仅使用Pascal VOC 2007中道路场景常见类别数据对模型进行训练，通过调整学习策略，利用较小的训练集和较少的训练轮次可以获得实时性强、精度较高的目标检测模型，单张图片检测时间只需0.04秒，在测试集上mAP (mean Average Precision)达到了91.5%，实验证明本文方法的有效性，该方法在精度、时延和漏检方面取得了较好的效果。

关键词

YOLOv3，道路场景，多目标检测，迁移学习，实时检测

A Road Multi-Object Detection Method Based on YOLOv3

Jingchao Fu[1,2], Qinghua Su[1,2*], Didi Zhang[1,2], Juntao Li[1,2]

[1]Beijing Wuzi University, Beijing
[2]Beijing Intelligent Logistics Laboratory, Beijing
Email: *qinghuasu@126.com

Received: Dec. 27th, 2020; accepted: Jan. 22nd, 2021; published: Jan. 29th, 2021

Abstract

Multi-object detection in road scenes is of great significance for automatic driving and intelligent

*通讯作者。

文章引用: 傅景超, 苏庆华, 张娣娣, 李俊韬. 一种基于 YOLOv3 的道路多目标检测方法[J]. 计算机科学与应用, 2021, 11(1): 207-216. DOI: 10.12677/csa.2021.111021

driving assistance of vehicles. The existing road object detection algorithms have some problems, such as low detection accuracy, poor real-time performance and missing target detection. To solve these problems, this paper constructs a road multi-object detection method based on YOLOv3 with high precision, low delay and low omission. Through in-depth analysis of the YOLOv3 object detection principle, based on the idea of Transfer Learning, the YOLOv3 model was trained on the pre-trained model using only the common category data of Pascal VOC 2007 (Pascal Visual Object Classes 2007) in the road scenes. By adjusting the learning strategy and using smaller training set and fewer training epochs, a target detection model with strong real-time performance and high accuracy can be obtained. The single image detection time is only 0.04 seconds, and the mean Average Precision on the test set reaches 91.5%. The experimental results show that the proposed method is effective and achieves good results in precision, delay and omission detection.

Keywords

YOLOv3, Road Scene, Multi-Object Detection, Transfer Learning, Real-Time Detection

1. 引言

在道路场景中，目标检测对于自动驾驶和辅助驾驶的发展有着重要意义。图像中含有丰富的语义信息，根据所采集的图像检测环境中目标的位置和类别，如实时人脸检测[1]、行人检测[2]、道路障碍物检测[3]和车牌识别[4]，可以增强车辆对周围环境的感知能力，辅助驾驶员、甚至取代驾驶员控制车辆行驶，提高车辆行驶的安全性，为汽车智能化提供保障。也有研究者将目标检测引入 SLAM 中，一方面可以利用目标检测的语义标签构建语义地图[5]，另一方面将语义标签引入回环检测中提高匹配正确率[6]，消除构建地图过程中的累计误差。

在深度学习尚未兴起之时，目标检测往往依赖于人工设计的特征，如 SHIFT 特征[7]和 HOG 特征[8]。但人工设计的特征表达能力有限，对光照、旋转等变化敏感的同时削弱了图像整体性，在复杂动态场景下鲁棒性较差。随着硬件的发展，深度学习在计算机视觉领域如图像分类、目标检测中都获得了长足的进步[9]。相比于传统方法，深度学习利用神经网络模型自主从数据中学习特征，可以获取更高级的语义信息，在数据量足够的情况下，深度学习模型可以有更好的泛化能力和鲁棒性以应对更加复杂场景中的工作，对于道路场景中的目标检测具有更好的效果。

现阶段基于深度学习的目标检测模型大致可以分为两种，基于区域提出的方法和基于回归/分类的方法[10]。其中基于区域提出的方法以 R-CNN 系列为代表，首先利用区域生成网络采集候选区域，再对所采集候选区域利用卷积神经网络进行分类，最终获取图像中目标框和类别；基于回归/分类的方法以 SSD、YOLO 系列为代表，将目标检测问题看作回归/分类问题，只需训练一个网络便可获取结果。基于区域提出的方法通常用于精度要求高的任务中，基于回归/分类的方法通常用于的实时性要求高的任务。在实际应用中目标检测更多考虑精度与实时性的折中，在保证一定精度的情况下尽量提高检测速度，或保证一定检测速度的情况下尽量提高精度。且现有的许多网络模型通常用于单一任务，在单一任务中效果优秀而复用性不足，利用迁移学习进行任务迁移可以提高现有模型的复用性，并显著降低模型设计训练的时间开销，减缓数据量不足的问题。

在实时场景中模型准确率并不能作为唯一标准，对于模型的实时性同样提出了很高的要求，YOLOv3模型在准确率和实时性上做到了很好的取舍，但从头开始训练一个有效的模型所需时间成本过高，尤其是对于算力较差的设备，模型训练的过大的时间开销既不利于实验阶段检验模型的有效性，也不利于后期优化模型所进行的迭代，无法及时对模型做出修正；且对于数据集较小的情况所训练模型容易产生过拟合，无法应用于实际场景。

本文使用 Pascal VOC 2007 中道路场景中较为常见的七个类别作为数据，在数据量较小、设备算力较差的情况下通过迁移学习思想，调整学习策略，在保证模型精度的情况下可实现对模型的调整和快速迭代，在实际场景中也能取得很好的效果，在测试集中 mAP 为 91.5%，单张图片检测速度约为 0.04 秒。

2. YOLOv3 的多目标检测算法

YOLO 系列目标检测算法由 Joseph Redmon 等人[11] [12]提出，在此之前基于滑动窗口的目标检测算法[12]需要重复遍历图像造成计算冗余耗费计算资源，基于区域提出的方法[13] [14] [15]将目标预测框生成器与目标分类器解耦合，分别对预测框生成网络和分类器网络进行训练，加大了优化难度，虽然具有很高的检测精度，由于其网络复杂特性，在实时性要求高的场景下效果较差。YOLO 系列利用单一神经网络，基于网格的思想将目标检测任务视作空间分离的边界框和相关的类概率的回归问题，在实时性要求高的场景下可以更好的做到实时目标检测，同时单一神经网络的模型优化也较为容易。

YOLOv3 模型由两部分组成：主干特征提取网络和利用主干特征提取网络所提取的特征进行预测的部分。

2.1. 主干特征提取网络

YOLOv3 以 Darknet-53 作为主干特征提取网络，Darknet-53 中主要使用了残差卷积[16] [17]。在 Darknet-53 的残差块中首先利用大小 3 × 3、步长为 2 的卷积核做卷积运算，保存其结果 X 后依次进行 1 × 1 和 3 × 3 的卷积操作，再加上之前的结果 X 作为该残差块最后的输出。残差卷积的使用可以缓解深度学习中梯度消失问题，用于训练更深层的网络。残差块示意图如图 1。

在 Darknet-53 中，每一次卷积操作后都使用了 BatchNormalizaiton 标准化与 LeakyReLU 激活函数缓解梯度消失问题，用于训练更深层的神经网络，其中 LeakyReLU 函数如公式(1)。

$$y_i = \begin{cases} x_i & if\ x_i \geq 0 \\ \dfrac{x_i}{a_i} & if\ x_i < 0 \end{cases} \tag{1}$$

如图 2，经由特征提取网络对输入数据进行特征提取，最后从 Darknet-53 中获取最后三个残差块的输出用于后续预测过程。

2.2. 从 Darknet-53 所得特征获取预测结果

在 YOLOv3 中，利用 Darknet-53 最后三个残差块所提取特征进行目标检测，三个特征层分别进行五次卷积操作，其中最后两层所获取特征既用于预测，同时通过上采样进行特征层扩张后与上一残差块所提取特征相加，共同用于上一层的预测，对特征提取网络所提取特征进行了有效利用。最后网络三层所输出网格大小从上到下依次为 52 × 52、26 × 26、13 × 13，利用特征金字塔由粗到精的思想提高模型检测的鲁棒性。

模型最后根据所获取的预测结果进行解码操作。在三个特征层获取的结果对应每张图分割为 52 × 52、26 × 26、13 × 13 的网格，每个网格负责一个区域的预测。根据每个区域预测所得结果，即目标中心坐标 x、y，预测框宽高 w、h，判断是否存在目标以及各个类别置信度，可得图像中目标预测框和类别置信度。

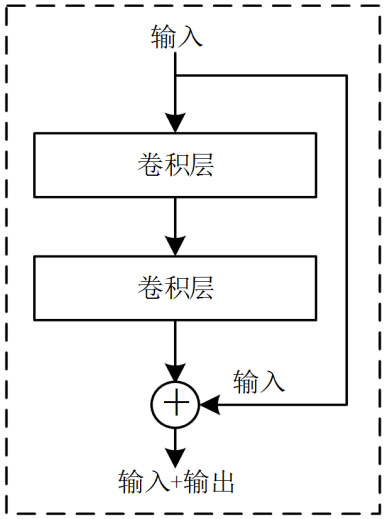

Figure 1. Schematic diagram of residual convolution
图 1. 残差卷积示意图

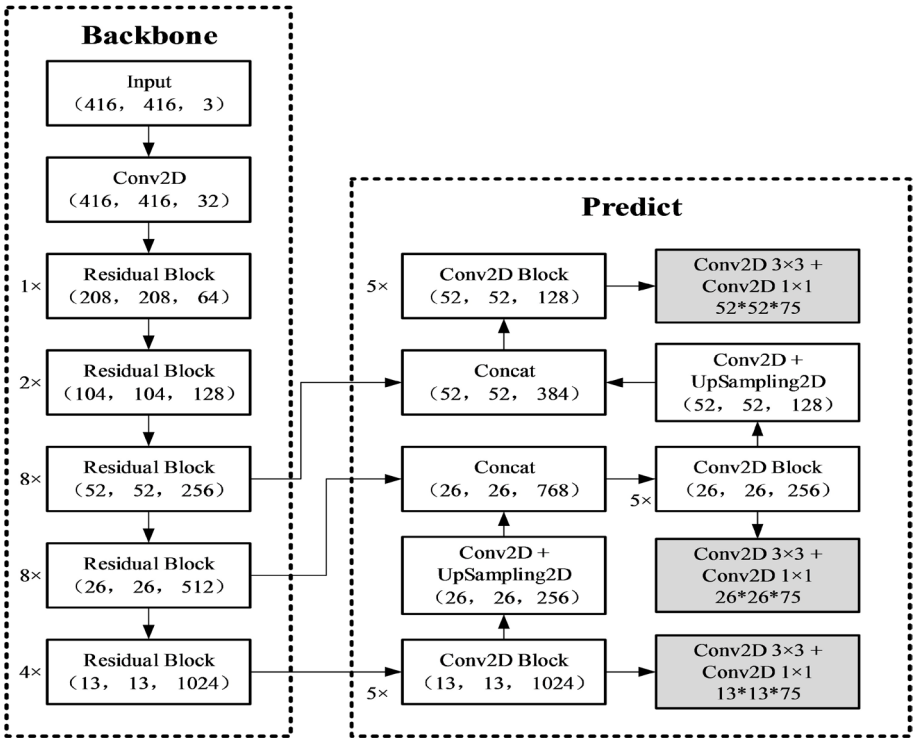

Figure 2. Schematic diagram of YOLOv3 model
图 2. YOLOv3 模型示意图

YOLO 中对同一目标会产生多个候选框，候选框之间可能存在相互重叠的情况，如图 3(左)。利用非极大值抑制(Non-Maximum Suppression, NMS)算法找到最佳目标边界框，消除冗余(取置信度最高的目标边界框)。步骤如下：

1) 根据置信度得分进行排序，选择置信度最高的边界框添加到最终输出列表中，将其从边界框列表删除；

2) 计算所有边界框的面积，计算置信度最高的边界框与其他边界框的交并比(Intersection-over-Union, IoU)；

3) 删除交并比大于阈值的边界框；

4) 重复上述过程，直到边界框列表为空。

在获取最终结果后在原图上绘制预测框和置信度，结果如图 3(右)。

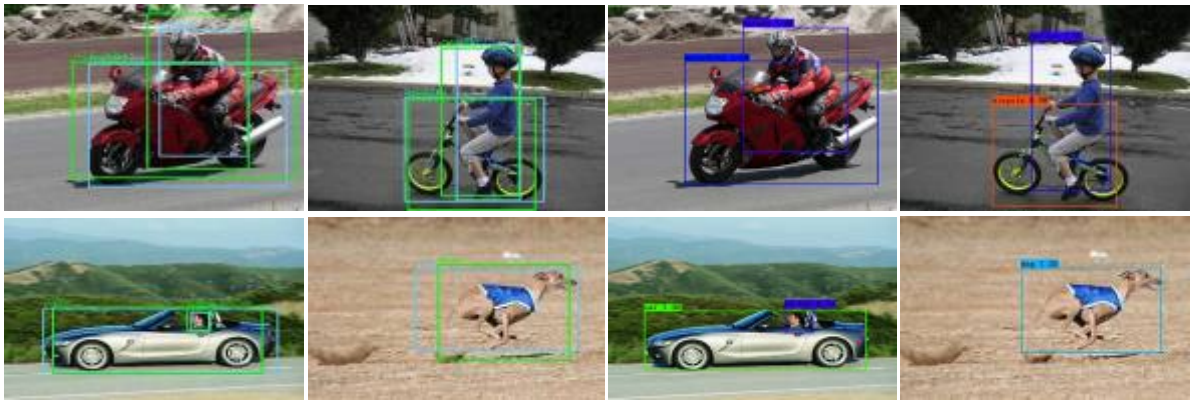

Figure 3. NMS pre- (left) and post- (right) candidate box results
图 3. NMS 前(左图)、后(右图)候选框结果

2.3. 损失函数

YOLO-v3 中，误差值由三个部分组成，分别为目标预测框误差 *lbox*，目标置信度误差 *lobj* 以及类别置信度误差 *lcls*。其中目标预测框误差使用 MSELoss(均方方差损失)，目标置信度误差与类别置信度误差使用 BCELoss (交叉熵损失)。

其中 MSE (Mean-Squared Error) Loss 的具体公式计算如公式(2)：

$$\text{MSE Loss}\left(y_i', y_i\right) = \frac{1}{n}\sum_{i=1}^{n}\left(y_i' - y_i\right)^2 \tag{2}$$

BCE (Binary-Cross Entropy) Loss 的具体公式计算如公式(3)：

$$\text{BCE Loss}\left(C_i', C_i\right) = -C_i'\log\left(C_i\right) - \left(1-C_i'\right)\log\left(1-C_i\right) \tag{3}$$

根据 MSE Loss 和 BCE Loss 可得目标预测框误差 *lbox*、目标置信度误差 *lobj* 和类别置信度误差 *lcls* 如公式(4)~(6)：

$$\begin{aligned} lbox = {} & \lambda_{coord}\sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{obj}\left(2-w_i\times h_i\right)\left[\left(x_i-x_i'\right)^2+\left(y_i-y_i'\right)^2\right] \\ & +\lambda_{coord}\sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{obj}\left(2-w_i\times h_i\right)\left[\left(w_i-w_i'\right)^2+\left(h_i-h_i'\right)^2\right] \end{aligned} \tag{4}$$

$$\begin{aligned} lobj = {} & \sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{obj}\left[C_i'\log\left(C_i\right)+\left(1-C_i'\right)\log\left(1-C_i\right)\right] \\ & -\lambda_{noobj}\sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{noobj}\left[C_i'\log\left(C_i\right)+\left(1-C_i'\right)\log\left(1-C_i\right)\right] \end{aligned} \tag{5}$$

$$lcls = \sum_{i=0}^{S\times S} I_{ij}^{obj}\sum_{C\in classes}\left[p_i'(c)\log\left(p_i(c)\right)+\left(1-p_i'\right)\log\left(1-p_i(c)\right)\right] \tag{6}$$

其中 S 为网格尺寸；B 表示预测框数目；I_{ij}^{obj} 为指示函数，表示如果在 i, j 处的预测框有目标，其值为 1，反之为 0；I_{ij}^{noobj} 为指示函数，表示如果在 i, j 处的预测框无目标，其值为 1，反之为 0。

综上可得目标损失函数 $loss(\text{object}) = lbox + lobj + lcls$，即公式(7)。

2.4. 基于预训练模型的迁移

在数据集大、设备算力充足、模型规模合适的情况下，通过合适的训练策略从零开始训练模型往往可以取得更好的效果，但在数据集规模较小且设备算力较低的情况下，即使训练结果收敛，模型往往也会由于过拟合而泛化性差导致不能使用。

迁移学习通过迁移包含在与目标域不同但相关的源域知识来提高模型在目标域上的表现，可以减少对模型所需数据的依赖[18]。机器学习的理想场景是有丰富的标签化训练实例，这些训练实例具有相同的测试数据分布，深度学习更甚，数据集的不足导致的模型欠拟合很难通过诸如图像增广、GAN 等技术做到大数据规模下的效果，也很难通过调整训练策略使模型收敛至可用程度。

深度学习模型属于层叠结构，在模型的不同层次提取不同的特征，从低级到高级的特征逐层提取。由于 Pascal VOC 2007 数据集中相关数据的匮乏，本文选择使用经过 MS COCO 数据集预训练后的 YOLOv3 模型，使用 Pascal VOC 2007 数据集中道路场景中的常见类别，冻结主干特征提取网络参数对 YOLOv3 模型进行训练，可以在很大程度上缓解目标领域数据集不足、模型训练耗时长的问题。

$$
\begin{aligned}
loss(\text{object}) = {} & \lambda_{coord}\sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{obj}\left(2-w_i\times h_i\right)\left[\left(x_i-x_i'\right)^2+\left(y_i-y_i'\right)^2\right] \\
& +\lambda_{coord}\sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{obj}\left(2-w_i\times h_i\right)\left[\left(w_i-w_i'\right)^2+\left(h_i-h_i'\right)^2\right] \\
& -\sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{obj}\left[C_i'\log\left(C_i\right)+\left(1-C_i'\right)\log\left(1-C_i\right)\right] \\
& -\lambda_{noobj}\sum_{i=0}^{S\times S}\sum_{j=0}^{B} I_{ij}^{noobj}\left[C_i'\log\left(C_i\right)+\left(1-C_i'\right)\log\left(1-C_i\right)\right] \\
& -\sum_{i=0}^{S\times S} I_{ij}^{obj}\sum_{C\in classes}\left[p_i'(c)\log\left(p_i(c)\right)+\left(1-p_i'\right)\log\left(1-p_i(c)\right)\right]
\end{aligned}
\tag{7}
$$

3. 实验与分析

3.1. 实验平台与实验数据

本文算法实验所使用配置如表 1 所示，使用深度学习中的 PyTorch 框架和计算机视觉中的 Open CV-Python 框架。

Table 1. Experimental platform configuration
表 1. 实验平台配置

Operating System	CPU	Memory	GPU	CUDA	CUDNN
Windows 10	Intel i5-9400F	16 GB	NVIDIA GEFORCE RTX 3070	CUDA 11.0	CUDNN 8.04

本实验所使用 Pascal VOC 2007 数据集的部分数据。Pascal VOC 2007 是一个非常流行的数据集，用于构建和评估图像分类、目标检测和分割的算法。Pascal VOC 2007 数据集共包含：训练集(5011 幅)，测试集(4952 幅)，共计 9963 幅图像，共包含 20 个种类。

针对道路场景的特殊性，选取道路场景中出现频率较大的七个类别用于模型训练，其中包含的类别及各个类别所包含的样本数如表 2 所示。

在实验中我们将 Pascal VOC 2007 数据集中的训练集和测试集视为一个主体后做切分，训练集与测试集相互独立，比例为 7:3，测试集用于最终检验模型效果；训练集在模型训练过程中又分为训练集和验证

集，比例为 9:1，验证集用以训练过程中检查网络训练效果。

3.2. 训练参数

神经网络的训练过程中，学习率的设置至关重要。过大的学习率虽然可以提高模型的收敛速度，但损失值容易出现震荡导致模型无法收敛；过小的学习率虽然最终可以使模型较好地收敛，但其收敛速度过慢会极大的增加模型训练的时间开销。在训练中同时采用了学习率衰减策略，在训练初期较大的学习率可以使得模型较快收敛，在训练后期较小的学习率可以帮助模型更好地收敛到极小值。设置 BatchSize 需考虑网络所需训练参数量以及 GPU 显存，过大的 BatchSize 会导致 GPU 显存溢出无法运行，过小的 BatchSize 会降低 GPU 的利用率造成算力浪费，且模型训练需要更大的时间开销，因此选择合适的 BatchSize 可以提高显存利用率，降低魔性的训练时间，提高模型收敛速度。

Table 2. Statistics of data sets used
表 2. 所使用数据集统计信息

Classes	Person	Bus	Car	Motorbike	Bicycle	Cat	Dog
Number	4015	360	1434	467	482	659	839

在训练中使用了 Adam 优化器[19]来优化模型参数。Adam 优化器结合了 Momentum 算法和 RMSProp 算法的优点，利用动量累积梯度，动量的使用既加快了模型的收敛速度，又减小了模型的波动幅度，并且进行了偏差修正，使得模型可以更快更好地收敛。

本文采用了经 ImageNet 数据集训练的 Darknet-53 作为主干网络进行训练，极大减小了模型从零开始训练的时间成本。模型优化共进行五十轮训练迭代，分为两部分。第一部分冻结特征提取网络 DarkNet-53 参数，仅对其余部分参数进行训练，训练轮次为 25 轮，初始学习率为 0.001，学习率衰减率为每轮 0.05，BatchSize 为 32。第二部分对特征提取网络参数进行解冻，整个网络参数都参与训练，训练轮次为 50 轮，初始学习率为 0.0001，学习率衰减率为 0.05，BatchSize 为 8。

3.3. 评价标准

目标检测中通常使用 mAP (mean Average Precision)作为评价标准，判断模型的检测精度。

实验中依据训练后模型对测试集的预测结果做出 *P*-*R* 曲线，求出各个目标的平均精准度 AP (*P*-*R* 曲线积分)，最后求出 mAP 表示不同种类的平均精准度均值，模型的 mAP 结合模型 TFP 系数求得 *M* 值，*M* 值越大表示模型效果越好。

其中 *P*-*R* 曲线由精确率 *P* (precision)与召回率 *R* (recall)构成：

$$P = \frac{TP}{TP + FP} \tag{8}$$

$$R = \frac{TP}{TP + FN} \tag{9}$$

其中 *TP* (True Positive)为真正例，*FP* (False Positive)为假正例，*FN* (False Negative)为假反例。

在现实道路场景中，目标的漏检同样会产生严重问题，因此以对数平均缺失率(log-average miss rate)作为评价标准之一。

3.4. 实验结果分析

在道路场景中，道路情况复杂，车流、人流密集的情况下识别目标众多，会增大道路目标检测的难度，且对于快速移动的车辆而言，需要快速检测道路场景中的目标，对模型的精度和速度都有一定要求，

且在保证前面两者的情况下，也应尽量避免漏检。

模型共进行 50 轮训练。其中前 25 轮冻结主干特征提取网络参数，仅对其余部分进行训练；后 25 轮对主干特征提取网络参数进行解冻，将整个模型进行训练。

训练过程中模型损失函数的损失值如图 4 所示。在 0 到 25 轮迭代中，仅优化主干特征提取网络之外的参数，模型很快趋于收敛时间；在 26 到 50 轮迭代中，模型训练集损失值先增长，后逐渐下降，但模型验证集损失函数逐渐上升。说明模型在冻结主干特征提取网络参数进行训练时模型可以很好的收敛，但主干特征提取网络参数解冻后由于训练数据量的不足导致模型无法收敛。表明在利用迁移学习训练模型时，冻结主干特征提取网络可在数据集较小的时候使模型很好地收敛。训练模型在测试集上获得的 mAP 达到 91.5%，对单张图片进行目标检测时间开销为 0.04 s，基本满足实时检测条件。

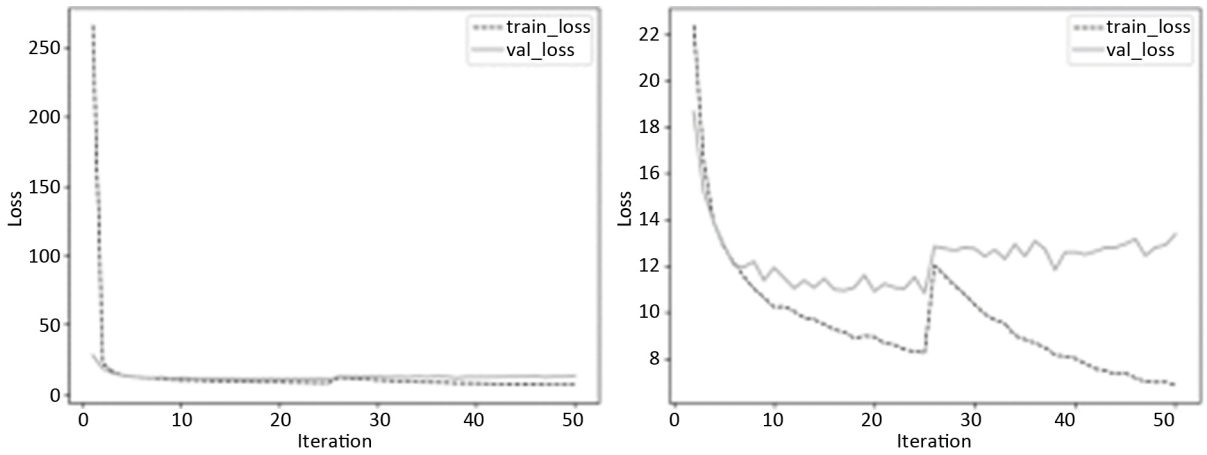

Figure 4. The change of loss in training process
图 4. 训练过程损失值变化情况

在道路场景中情况复杂，存在干扰信息较多，会对检测结果造成一定影响。因此选取部分图片测试在不同场景下，模型对道路场景中目标的检测效果，部分测试结果如图 5 所示。

从图 5 可以看出，在不同场景中，模型仍可有效地对单个、多个目标进行监测，且缺失率较小，实现了较好的检测效果，证明了道路场景中该模型的有效性。

根据测试集的划分，总共对 3989 张图像进行测试，各类别对数平均缺失率(Log-Average Miss Rate)、AP 和 mAP 结果如表 3。其中 person、car 类缺失率较高的原因是由于其聚集性相较于其他类更高，因此在检测中容易出现漏检情况。

Table 3. The experimental results
表 3. 实验结果

Classes	Log-Average Miss Rate	AP	mAP
person	30.04%	84.65%	
bus	10.04%	92.94%	
car	18.40%	91.09%	
motorbike	14.15%	91.56%	91.5%
bicycle	15.31%	93.61%	
cat	10.07%	93.73%	
dog	15.01%	90.93%	

Figure 5. Detection results of different scene
图 5. 不同场景下检测结果

从表 3 可以看出，本文模型在各个检测类别中都表现出了很好的监测效果，表明了在训练数据集较小的情况下，利用迁移学习思想对效果较好的其他任务中使用的模型迁移至目标域中也可取得很好地效果，一方面降低了模型训练的时间开销，减缓了数据集不足的问题，另一方面也有利于提高模型的泛用性，为其他场景的使用提供借鉴。

4. 结论

本文使用基于 Darknet-53 的 YOLOv3 算法应用于道路场景，构建了一个低缺失率、高精准的道路多目标检测算法。该算法基于 MS COCO 数据集预训练的模型，利用 VOC 2007 数据集中道路场景常出现的类进行模型训练，在较低缺失率的情况下取得了 91.5%的 mAP，基本实现了道路场景多目标检测。但由于网络规模较大，利用视频进行道路目标检测中 FPS 值约为 30。针对这一问题，还可对网络进行进一步优化以提升模型的实时性。

基金项目

国家自然科学基金，基金资助号(61803035)；北京市社科基金，基金资助号(20GLB026)。

参考文献

[1] 徐建亮, 周明安, 毛建辉, 方坤礼. 基于一种卷积神经式类网络的实时人脸识别方法研究[J]. 计算机科学与应用, 2020, 10(1): 11-20.

[2] 朱波, 黄茂飞, 谈东奎, 等. 基于神经网络与数据融合的行人检测方法[J]. 汽车工程, 2020(11): 37-44.

[3] 彭育辉, 郑玮鸿, 张剑锋. 基于深度学习的道路障碍物检测方法[J]. 计算机应用, 2020, 40(8): 2428-2433.

[4] 李冬伟, 孙卓, 常书林. 基于低分辨率车牌识别系统的应用研究[J]. 计算机科学与应用, 2020, 10(4): 721-731.

[5] Andreas, N. and Hertzberg, J. (2008) Towards Semantic Maps for Mobile Robots. *Robotics & Autonomous Systems*, **56**, 915-926. https://doi.org/10.1016/j.robot.2008.08.001

[6] Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., *et al*. (2016) Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, **32**, 1309-1332. https://doi.org/10.1109/TRO.2016.2624754

[7] Lowe, D.G. (2004) Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, **60**, 91-110. https://doi.org/10.1023/B:VISI.0000029664.99615.94

[8] Dalal, N. and Triggs, B. (2005) Histograms of Oriented Gradients for Human Detection. *IEEE Computer Society Conference on Computer Vision & Pattern Recognition*, San Diego, 20-26 June 2005, 886-893. https://doi.org/10.1109/CVPR.2005.177

[9] Lecun, Y., Bengio, Y. and Hinton, G. (2015) Deep Learning. *Nature*, **521**, 436. https://doi.org/10.1038/nature14539

[10] Zhao, Z.Q., Zheng, P., Xu, S.T. and Wu, X. (2018) Object Detection with Deep Learning: A Review. https://arxiv.org/abs/1807.05511

[11] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A. (2016) You Only Look Once: Unified, Real-Time Object Detection. *IEEE Computer Vision & Pattern Recognition*, Las Vegas, 26 June-1 July 2016, 779-788. https://doi.org/10.1109/CVPR.2016.91

[12] Felzenszwalb, P., *et al*. (2010) Object Detection with Discriminatively Trained Part-Based Models. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **32**, 1627-1645. https://doi.org/10.1109/TPAMI.2009.167

[13] Girshick, R., Donahue, J., Darrell, T. and Malik, J. (2014) Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *IEEE CVPR*, Columbus, 24-27 June 2014, 580-587. https://doi.org/10.1109/CVPR.2014.81

[14] Girshick, R. (2015) Fast R-CNN. *IEEE International Conference on Computer Vision* (*ICCV*), Santiago, 1440-1448. https://doi.org/10.1109/ICCV.2015.169

[15] Ren, S., He, K., Girshick, R. and Sun, J. (2017) Faster r-cnn: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **39**, 1137-1149. https://doi.org/10.1109/TPAMI.2016.2577031

[16] Redmonand, J. and Farhadi, A. (2018) Yolov3: An Incremental Improvement. https://arxiv.org/pdf/1804.02767

[17] He, K., Zhang, X., Ren, S. and Sun, J. (2016) Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision & Pattern Recognition*, Las Vegas, 27-30 June 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

[18] Pan, S.J. and Yang, Q. (2010) A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, **22**, 1345-1359. https://doi.org/10.1109/TKDE.2009.191

[19] Kingma, D. and Ba, J. (2014) Adam: A Method for Stochastic Optimization. https://arxiv.org/abs/1412.6980

.