

文章编号: 2095-2163(2023)08-0017-08

中图分类号: U491.1+13

文献标志码: A

车载相机和 GPS 信息融合的城市道路交通信息检测方法

严伟政, 王道斌, 严运兵

(武汉科技大学 汽车与交通工程学院, 武汉 430065)

摘要: 利用车载相机进行城市道路交通信息采集时通常会面临诸多难题,如城市道路环境复杂、驾驶视角下的车辆遮挡、无法独立获取自身位置信息等。针对上述问题,提出一种基于车载相机和 GPS 信息融合的城市道路车辆纵向间距与纵向速度估计方法,进而实现对路侧停靠车辆的识别以及城市道路通行速度、交通流密度以及交通流量的检测。实车实验结果表明本方法能够实现准确可靠、经济有效的移动式城市道路交通信息检测,能够为智能车辆和智能交通系统的相关研究提供数据支撑。

关键词: 智能交通; 交通信息检测; 目标检测; YOLOv5; Deep-SORT

Urban road traffic information detection method based on vehicle camera and GPS information fusion

YAN Weizheng, WANG Daobin, YAN Yunbing

(College of Automobile and Traffic Engineering, Wuhan University of Science and Technology, Wuhan 430065, China)

【Abstract】 During the process of using on-board camera to collect urban road traffic information, there usually exists many difficulties, such as complex urban road environment, blocking of vehicles from driving perspective, and unable to obtain its own position information independently. In order to solve the above problems, an estimation method of longitudinal distance and longitudinal speed of urban road vehicles based on on-board camera and GPS information fusion is proposed, which could realize the identification of roadside parked vehicles and the detection of urban road speed, traffic flow density and traffic flow. The results of real vehicle experiments show that this method can achieve accurate, reliable, economical and effective mobile urban road traffic information detection, and can provide data support for the related research of intelligent vehicles and intelligent transportation systems.

【Key words】 ITS; urban traffic information measurement; vehicle identification; YOLOv5; Deep-SORT

0 引言

交通拥堵已经成为众多城市发展中所面临的一个不可忽视的热点问题^[1]。准确有效的城市道路交通信息检测方法对于交通拥堵的成因分析具有决定性作用,对于开展智能车辆和智能交通相关领域的研究也有着重要意义^[2]。

现有的城市道路交通信息采集技术一般分为2类:固定式交通信息检测方式和移动式交通信息检测方法^[3]。其中,固定式交通信息检测方法通常是在某个固定的位置布设传感器,如环形线圈、微波雷达、监控视频等。这些设备成本高昂、维护困难、受环境影响较大,因此有关部门仅在关键路段或者交

叉口安装这些设备。为避免检测盲区往往需要提高传感器的布设密度,进而导致成本增加。移动式交通信息检测方法通常是在某个移动平台布设传感器,如浮动车、无人机(UAV)、车载相机等。与固定式交通信息检测方法相比,移动式交通检测方法经济高效,并且能有效弥补固定式交通信息检测方法中的部分盲区。

在固定式交通信息检测方法中,环形线圈因其检测结果准确、工作性能稳定、制作成本低等优点,广泛应用于交通信息检测领域。环形线圈通过感知电磁场的变化来判断是否有车辆通过,叶青等学者^[4]运用自适应共振神经网络提供的自聚类能力,将环形线圈采集的各种车辆波形聚类成不同类别模

基金项目: 国家自然科学基金(51975428)。

作者简介: 严伟政(1997-),男,硕士研究生,主要研究方向:图像识别、交通信息采集;王道斌(1987-),男,博士,副教授,主要研究方向:交通仿真、交通系统建模、干线协调控制;严运兵(1968-),男,博士,教授,主要研究方向:汽车动力学及其控制、新能源汽车驱动与控制、智能驾驶人机交互与决策。

通讯作者: 王道斌 Email: wangdaobin05@163.com

收稿日期: 2022-08-18

式,进而实现对城市道路车辆类型的识别分类。Cheol 等学者^[5]利用感应线圈收集到的信号估计跟驰情况下的安全停车距离,并提出一个量化指标衡量追尾的风险。不同于环形线圈,微波雷达通过计算发射波与反射波之间的时间间隔获取目标与传感器之间的距离。曹林等学者^[6]利用微波雷达回波信号构建数据集,并采用改进的 LeNet-5 卷积神经网络实现对城市道路大小车型的分类识别。监控相机在固定式交通信息检测领域中应用也比较广泛。李永上等学者提出一种改进 YOLOv5 结合 Deep-SORT 的车辆检测跟踪方法,实现了监控视角下的交通流量统计。Li 等学者^[7]通过提高 YOLOv3 对密集场景下小目标的检测能力,进而实现监控视角下的城市道路速度、密度和流量等参数的提取。但是固定式交通信息检测方法中传感器通常布设在固定位置,只能获取特定路段或者交叉口的交通信息,检测范围较窄、不够灵活,容易受到外界环境的干扰。

在移动式交通信息检测方法中,浮动车法应用较广。浮动车数据主要来源于车载 GPS 设备的导航系统和手机地图 APP。袁向红^[8]借助 GPS 浮动车数据,建立交通流量预测模型,并在 VISSIM 平台进行仿真验证,结果表明所提模型具有良好的预测效果。针对社会车辆轨迹难以获取的问题,Kong 等学者^[9]提出一种基于浮动车数据和车载社交网络的社会车辆轨迹生成方法,实验表明其轨迹与北京市交管局发布的实时交通状况基本吻合。但是浮动车法极大程度依赖浮动车的数量和通信质量,当道路浮动车占比不足或者处于信号较弱地段时,检测结果的准确性会大打折扣。深度学习的兴起,使得从视频中提取特征信息变得更为方便,因此利用相机进行移动式交通信息检测也得到了越来越多的关注。UAV 成本低且具有

较高的灵活性,Shan 等学者^[10]利用 YOLOv3 算法从 UAV 采集视频中检测车辆信息,提取相关交通流参数,但由于禁飞区的存在,使得 UAV 无法充分发挥其灵活性,同时高空较快的风速也容易造成 UAV 的漂移和抖动。车载相机检测范围广、机动性高,关闯等学者^[11]利用车道线消隐点的坐标计算出车载相机的外参,再利用相机的光学特性进而获得与前车之间的距离。另一些学者则侧重于视频信息的避障以及交通场景的理解,如 Godha^[12]利用车载相机检测道路障碍物以达到辅助驾驶的功能。车载相机不仅能够获取每一帧图像上的直接信息,还能够获取目标的相对速度和相对位置等信息。但是,如果要获取目标的绝对速度和绝对位置,还需要其它传感器提供一些额外信息。

因此本文结合车载相机检测范围广、灵活性高的特点,以及浮动车具有自身定位能力的优点提出一种移动式交通信息检测方法。该方法主要包括 2 个部分。第一部分为城市道路目标车辆的检测与跟踪,主要包括 YOLOv5 (you-only-look-once)^[13]算法的目标车辆检测,以及 Deep-SORT^[14]算法的目标车辆跟踪。第二部分为城市道路交通信息参数提取,主要包括目标车辆纵向间距、纵向速度的计算,以及城市道路通行速度、交通流密度和交通流量的获取。

1 城市道路目标车辆的检测

考虑到移动式交通信息检测对检测精度和速度的需求,本文采用最新一代 YOLOv5 算法进行城市道路目标车辆的检测。该方法在保证检测精度的同时还能极大提高检测效率,其主要结构包括 Input 层、Backbone 层、Neck 层和 Prediction 层,网络结构示意图如图 1 所示。

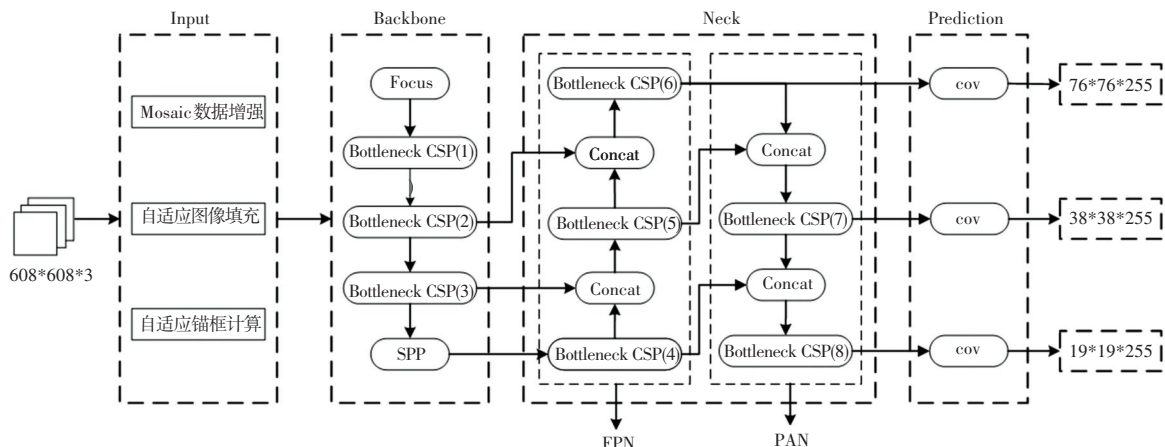


图 1 YOLOv5 模型结构图

Fig. 1 YOLOv5 model structure

由图 1 可看到, Input 输入层主要用于输入数据的预处理, 包含 Mosaic 数据增强、自适应图像填充和自适应锚框计算。Mosaic 数据增强用于丰富背景, 增加小目标数量, 提高 GPU 利用效率, 自适应填充降低信息冗余以提高计算速度, 自适应锚框能在更换数据集时, 自动计算初始锚框大小。Backbone 层主要通过卷积操作, 从不同层次提取图像中的特征参数。Backbone 层由 Focus、CSP (Cross Stage Partial Network) 以及 SPP (Spatial Pyramid Pooling) 三个部分构成。Focus 主要对输入图片进行切片操作以降低计算量。CSP 在加强骨干网络的学习能力的同时还能提高网络的推理速度。SPP 则用于扩大感知视野, 分离上下文特征。Neck 层包括 FPN (Feature Pyramid Networks) 和 PAN (Path Aggregation Network) 两个部分, FPN 向下传递强语义特征, PAN 向上传递强定位特征, 将 Backbone 层提取的特征从不同层面进行融合, 进一步提升检测能力。Prediction 层分别从大、中、小三个尺度检测车辆, 并通过 NMS (non maximum suppression) 非极大值抑制去除重复和多余的框, 最终输出目标车辆的类型、坐标和置信度等检测结果。

2 城市道路目标车辆的跟踪

为防止检测算法的误检与漏检, 提高检测算法的稳定性, 利用 Deep-SORT 算法对测得目标做进一步的预测跟踪, 流程如图 2 所示。

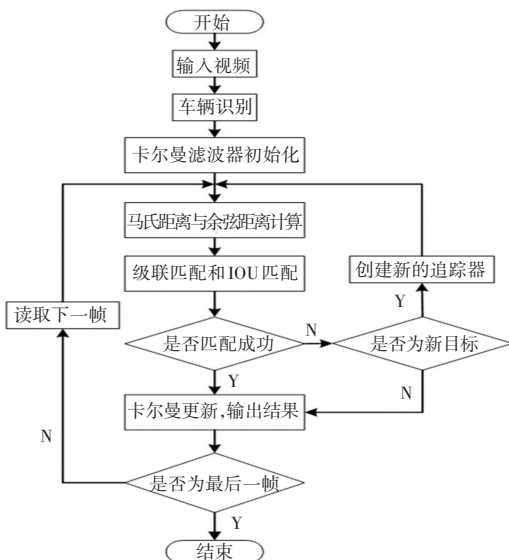


图 2 Deep-SORT 流程图

Fig. 2 Flow chart of Deep-SORT

2.1 城市道路目标车辆的跟踪

Deep-SORT 算法利用线性卡尔曼滤波器对目

标进行预测, 使用八维状态向量 $(x, y, r, h, vx, vy, vr, vh)$ 来表示目标的位置以及运动状态, 其中 x, y 为检测框中心点坐标, r 为检测框的宽高比, h 为检测框的高度, 余下 4 个变量分别表示对应变量的图像坐标系中的增量。已知 $t-1$ 时刻跟踪目标的具体状态, 则 t 时刻预测状态为:

$$X_{dt-1} = \mathbf{F}X_{dt-1} \quad (1)$$

其中, X_{dt-1} 为跟踪目标在 $t-1$ 时刻的最优估计; \mathbf{F} 为状态转移矩阵; X_{dt-1} 为 $t-1$ 时刻对 t 时刻的状态预测值, 状态转移矩阵见式(2):

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & 0 & 0 & dt & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & dt & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & dt & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & dt \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

协方差矩阵根据式(3)进行计算:

$$P_{dt-1} = \mathbf{F}P_{dt-1}\mathbf{F}^T + \mathbf{Q} \quad (3)$$

其中, P_{dt-1} 为跟踪目标在 $t-1$ 时刻最优估计值; P_{dt-1} 为 $t-1$ 对 t 时刻协方差矩阵的预测值; \mathbf{Q} 为系统噪声。

进而根据 $t-1$ 时刻的状态预测值来更新 t 时刻的最优估计值, 同时更新卡尔曼增益, 具体过程如式(4)和(5)所示:

$$X_{dt} = X_{dt-1} + \mathbf{K}_i(Z_i - \mathbf{H}X_{dt-1}) \quad (4)$$

$$\mathbf{K}_i = P_{dt-1}\mathbf{H}^T(\mathbf{H}P_{dt-1}\mathbf{H}^T + \mathbf{R})^{-1} \quad (5)$$

其中, X_{dt} 为 t 时刻的最优估计; Z_i 为 t 时刻的观测值, 由传感器获得; \mathbf{H} 为观测矩阵; \mathbf{R} 为系统的观测噪声。

最后, 根据式(6)更新 t 时刻的协方差矩阵:

$$P_{dt} = (1 - \mathbf{K}_i\mathbf{H})P_{dt-1} \quad (6)$$

2.2 匈牙利分配

Deep-SORT 利用匈牙利算法对已有的检测框和预测框进行分配。使用马氏距离来表征对象的运动信息, 具体公式如下:

$$d^{(1)}(i, j) = (\mathbf{d}_j - \mathbf{y}_i)^T \mathbf{S}_i^{-1} (\mathbf{d}_j - \mathbf{y}_i) \quad (7)$$

其中, \mathbf{d}_j 和 \mathbf{y}_i 分别表示第 j 个检测结果和第 i 个预测结果的状态向量, \mathbf{S}_i 表示检测结果和平均结果之间的协方差矩阵。

当物体被遮挡或者镜头抖动时, 马氏距离不再适合作为两者关联的标准, 因此引入最小余弦距离表示物体的外观特征信息, 公式如下:

$$d^{(2)}(i, j) = \min \{ 1 - \mathbf{r}_j^T \mathbf{r}_k^{(i)} \mid \mathbf{r}_k^{(i)} \in R_i \} \quad (8)$$

其中, \mathbf{r}_j 为检测框 d_j 所对应的外观特征描述信息, 且 $\|\mathbf{r}_j\| = 1$, R_i 为特征仓库, 用于保存每一个跟踪轨迹 k 最近 100 条成功关联的特征描述符。式(8)表示第 i 个跟踪框和第 j 个检测框的最小余弦距离。

最后, 将运动特征和外观特征进行加权融合, 得到最终的度量指标:

$$c_{i,j} = \lambda d^{(1)}(i, j) + (1 - \lambda) d^{(2)}(i, j) \quad (9)$$

2.3 特征提取网络重训练

Deep-SORT 的特征提取网络在设计时主要用于提取行人的外观特征, 为提高神经网络对车辆的外观提取能力, 在大规模的车辆 ReID 数据集上重新进行训练, 数据集包含 576 辆车、共 12 347 张从不同角度拍摄的图片, 将图片裁剪至 $126 * 135$ 大小后放入网络进行训练。训练效果如图 3 所示。

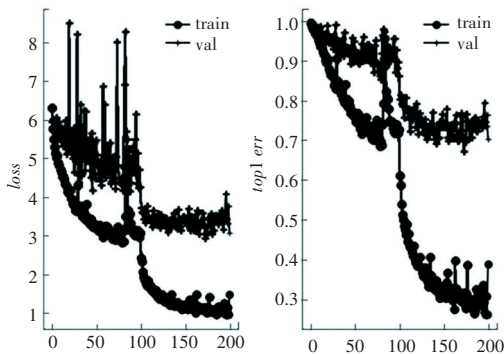


图3 特征提取网络训练效果

Fig. 3 Training effect of feature extraction network

当训练次数达到 200 轮后, 损失函数基本趋向于平衡, 训练完成后该网络即可用于提取车辆外观特征。

3 道路交通参数提取

3.1 纵向间距

利用单目相机光学特性进行前方车辆距离检测, 首先需要获取相机的内参矩阵, 内参矩阵如下所示:

$$\mathbf{matrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (10)$$

其中, (f_x, f_y) 表示相机的焦距长度, (u_0, v_0) 表示相机光轴投影到成像平面的坐标, 由于厂商工艺等因素, (u_0, v_0) 往往不在成像平面的中心点。

单目相机成像过程是将三维空间中的场景投影到二维平面上, 这一过程可以用小孔成像来描述, 如图 4 所示。

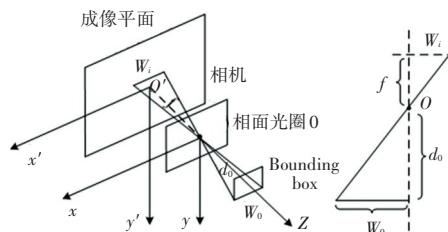


图4 小孔成像示意图

Fig. 4 Schematic diagram of small hole imaging

利用小孔成像原理, 根据式(11)估算试验车与前车的纵向间距:

$$d_0 = \frac{w_0 \times f_y}{w_i} \quad (11)$$

其中, d_0 表示相机到目标车辆的纵向间距; f_y 表示相机 y 轴方向的焦距, 由标定获得; w_0 表示目标车辆的真实宽度, 小汽车为 1.8 m, 货车为 2.6 m; w_i 为车辆成像后的像素宽度。

3.2 纵向速度

前车相对于试验车的速度可以根据式(12)进行计算:

$$v_{rela} = \frac{d_0(t) - d_0(t-1)}{\Delta t} \quad (12)$$

其中, $d_0(t)$ 和 $d_0(t-1)$ 分别为相邻两帧图像中目标车辆与试验车之间的纵向间距, Δt 为两帧图像之间的时间间隔。当 $v_{rela} < 0$ 时, 说明目标车辆的速度相对于试验车较慢; 当 $v_{rela} > 0$ 时, 说明目标车辆的速度相对于试验车较快; 当 $v_{rela} = 0$ 时, 说明目标车辆与试验车速度相同。

根据 GPS 信息获取试验车的实时通行速度, 并将试验车的通行速度视作基准车速 v_{bas} , 则目标车辆通行速度可以根据式(13)进行计算:

$$v_{real} = v_{rela} + v_{bas} \quad (13)$$

路侧停靠车辆如图 5 所示, 这类车辆通常具有以下 2 个特征:

- (1) 位于图像右侧边缘。
- (2) 绝对速度为 0 km/h。



图5 路侧停靠车辆

Fig. 5 Parked vehicles on the side of the road

因此, 根据这 2 个特征可以对路侧停靠车辆进

行识别。如果目标车辆位于图像右侧边缘,且其通行速度小于设定的阈值 α (α 大于但接近于 0 km/h), 则将该目标车辆视作路侧停靠车辆。综合考虑 GPS 和车载相机的数据精度, 本文将 α 的值定为 5。

3.3 交通参数提取

剔除路侧停靠车辆, 对视野内同向移动车辆保持检测和跟踪, 以获得每帧图像中的有效车辆。假定每辆有效车辆相对于试验车的纵向车间距检测值为 d_o , 进而根据式 (14) 计算每帧图像中单位长度内的车辆数目, 即交通流密度:

$$\rho = \frac{N}{\max_i d_o(i)} \quad (14)$$

其中, i 为有效车辆的编号; N 为有效车辆的个数。

假定第 j 帧图像获取的实时交通流密度为 $\rho(j)$, 则路段的交通流密度 ρ_{seg} 可根据式 (15) 进行估算:

$$\rho_{seg} = \frac{1}{N} \sum_{j=1}^{N_{seg}} \rho(j) \quad (15)$$

其中, N_{seg} 为路段拍摄视频的总帧数。

假定图像中每台有效车辆的绝对速度检测值为 v_{abs} , 进而根据式 (16) 计算道路的平均通行速度 v_{ave} :

$$v_{ave} = \frac{1}{N} \sum_{i=1}^N v_{abs}(i) \quad (16)$$

其中, i 为有效车辆的编号; N 为有效车辆的数目。

与交通流密度检测类似, 假定第 j 帧图像获取的实时通行速度为 $v_{ave}(j)$, 则待测路段的通行速度 v_{seg} 可根据式 (17) 进行估计:

$$v_{seg} = \frac{1}{N_{seg}} \sum_{j=1}^{N_{seg}} v_{ave}(j) \quad (17)$$

获得每帧图像的交通流密度 ρ 以及道路平均通行速度后, 进而根据式 (18) 估计当前位置单位时间内通过车辆的个数, 即交通流量:

$$vol = v_{ave} \cdot \rho \quad (18)$$

与交通流密度检测类似, 假定第 j 帧图像获取的实时交通流量为 $vol(j)$, 则待测路段的交通流量 vol_{seg} 可根据式 (19) 进行估计:

$$vol_{seg} = \frac{1}{N_{seg}} \sum_{j=1}^{N_{seg}} vol(j) \quad (19)$$

4 实验与分析

4.1 数据集

车载相机检测视野较宽, 往往会采集到对向车

道的车辆信息, 进而影响检测结果的准确性, 因此需排除对向车道车辆对检测结果的影响。同时常规的目标检测算法会将整辆车框检出, 不利于车辆纵向间距的测量。为区分目标车辆, 进一步提高距离检测精度, 本文搭建道路车辆分类数据集, 将车载视频采集到的 10 段视频分解成 8 244 张图片, 同时将图片中的车辆归为 car_front、car_side、car_back、truck_front、truck_side、truck_back 六个类别, 具体分类标准如图 6 所示。

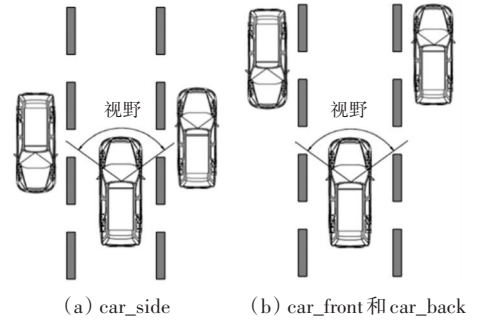


图 6 道路车辆分类

Fig. 6 Classification of road vehicles

由图 6(b) 可见, 在视野内若能观察到前方车辆头部或尾部, 则将该车辆归类为 car_front 或 car_back, 否则定义为 car_side, 参见图 6(a), 货车分类同样以此类推。特别地, 为了保证测距精度, car_back 这一类别仅标记车辆尾部区域。

4.2 纵向间距检测

准确的纵向车间距检测是道路交通参数提取的前提, 本次实验采用分辨率为 1 920×1 200 像素的 CMOS 相机, 相机内部参数的标定结果见表 1。

表 1 相机标定结果

Tab. 1 Camera calibration result

标定参数	标定结果
焦距 f_x /mm	1 569.5
焦距 f_y /mm	1 570.8
x 方向主点坐标/pixel	916.9
y 方向主点坐标/pixel	574.9

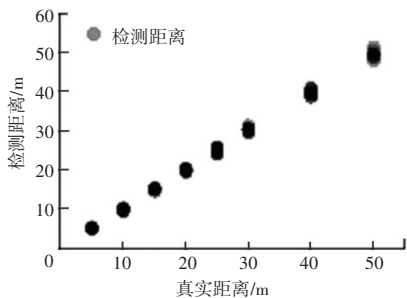
由于实验条件限制, 本文未进行跟车行驶条件下的动态测量, 仅在静止环境中完成了纵向间距的测量。共分为 5、10、15、20、25、30、40、50 m 八组实验, 每组实验分别在前车后方偏左 1 m、前车正后方、前车后方偏右 1 m 处拍摄 10 张图片, 取均值作为每组实验的检测值, 同真实值进行对比, 实验结果见表 2。

表2 纵向距离检测结果

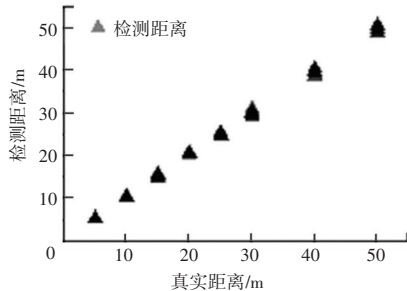
Tab. 2 Longitudinal distance detection results

组数	平均纵向间距/m			真实距离	误差/%		
	左后	正后	右后		左后	正后	右后
1	5.31	5.27	5.12	5	6.2	5.4	2.4
2	9.98	10.32	10.33	10	0.2	3.2	3.3
3	15.22	15.25	15.33	15	1.5	1.7	2.2
4	20.01	20.53	20.17	20	0.1	2.7	0.9
5	25.05	25.09	24.59	25	0.2	0.4	1.6
6	30.34	29.96	29.56	30	1.1	0.1	1.5
7	39.78	39.66	39.51	40	0.6	0.9	1.2
8	49.53	49.70	49.79	50	0.9	0.6	0.4

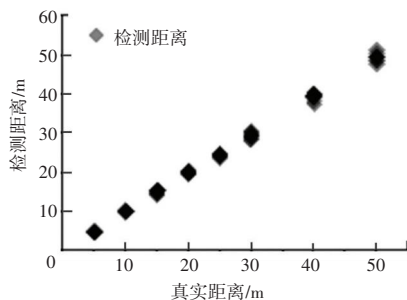
由表2中的数据可知,检测值与真实值之间最大误差为6.2%,最小误差为0.1%,具备了较高的检测精度,能够满足道路信息检测的需求。图7表示在不同横向距离检测结果图。图7中,横坐标表示真实距离,纵坐标为检测距离,从图7中可看出检测距离越远,测量结果越离散。



(a) 左后方检测结果



(b) 正后方检测结果



(c) 右后方检测结果

图7 测距散点图

Fig. 7 Scatter plot of ranging

4.3 道路交通信息检测

通过实车实验一共收集了2个路段的道路视频作为实验对象。路段A为三环线青菱立交起至北太子湖共11 km长度,于2022年3月29号16:06采集。路段B选取黄家湖大道叶家墩起南至何家湖小区2期共4.5 km长度,于2022年3月29号15:25采集。GPS数据由安装在试验车上的智能手机提供,用于同步收集试验车的自身速度、位置等信息。

实验结果如图8所示。图8中,检测框所标记的信息主要包括:类别、id、纵向间距以及纵向速度,同时显示当前时刻的平均通行速度、交通流密度和交通流量等参数。

图8(b)中,部分车辆被检测出来但并未参与计算,其原因在于前车距离过远时,检测框的抖动会使得检测结果出现较大偏差。为保证结果的稳定性和可靠性,经过多次实验,本文将检测的最远距离设定在80 m。



(a) 路段A



(b) 路段B

图8 实验图

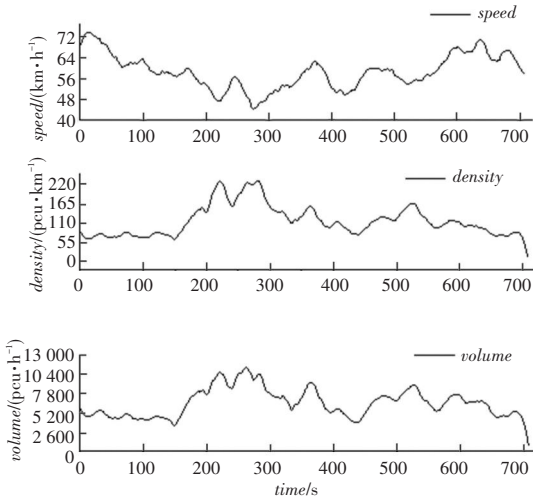
Fig. 8 Experimental figure

实验路段道路交通信息检测结果如图9所示。图9中,横坐标为时间,纵坐标分别表示根据当前时间对应帧计算的平均通行速度、交通流密度和交通流量,利用式(15)、(17)、(19)最终得到实验路段的通行速度、交通流密度和交通流量,计算结果见表3。表3中,数据按照2.5的换算系数将货车换算成标准车辆。

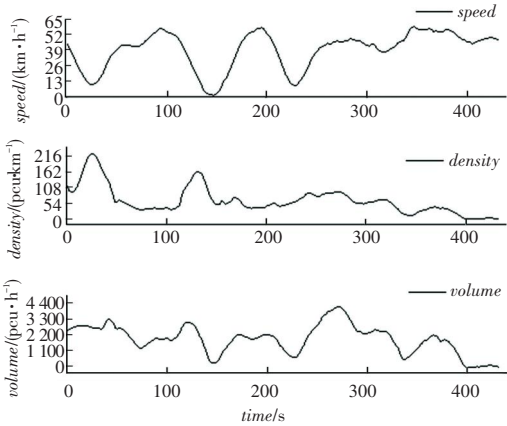
表 3 检测结果

Tab. 3 Detection results

实验路段	通行速度/ ($\text{km} \cdot \text{h}^{-1}$)	交通流密度/ ($\text{pcu} \cdot \text{km}^{-1}$)	交通流量/ ($\text{pcu} \cdot \text{h}^{-1}$)
A	52.55	102.98	5 838.65
B	30.04	54.24	1 736.12



(a) 路段 A 检测结果



(b) 路段 B 检测结果

图 9 道路检测结果

Fig. 9 Road inspection results

图 9(a) 为城市快速路路段检测结果, 在 220 s 和 280 s 处, 道路通行速度显著下降, 并伴随交通流密度和交通流量的上升, 这是由于该路段拥堵使得车辆行驶缓慢, 进而造成交通流密度和交通流量的增大, 在结尾处密度和流量的跌落则是由于试验车驶离快速路段造成。图 9(b) 为城市主干路段检测结果, 该路段平均通行速度在实验过程中出现了明显的波动, 是因为试验车在 24 s、150 s、220 s 时在交叉口处遇到红灯所导致。当试验车跟随车流于交叉口制动以及启动时, 由于视线的遮挡, 会导致检测的车辆纵向间距较低, 进而造成该时刻路段交通流密度的上

升, 因此在 24 s 和 150 s 处均出现了密度上升的现象。但在 220 s 处却并未出现上述状况, 是因为试验车驶至该交叉口时, 检测视野比较开阔, 对车辆的纵向间距的检测影响较小, 因此此刻交通流密度不会出现明显上升趋势。后期交通流密度的波动则是由于部分目标车辆出现或者离开检测范围造成的。

实验结果表明, 实验路段 A 的平均通行速度为 52.55 km/h , 略低于城市快速路段 60 km/h 的设计通行速度, 较快的通行速度和较高的密度使得该路段具有较大的流量。试验车在收集路段 B 的道路交通信息时, 车流连续遭遇 3 次红灯, 导致路段通行速度始终维持在较低水平, 仅有 30.04 km/h 的通行速度。

4.4 仿真实验验证

为验证所提城市道路交通流量检测框架, 利用 PreScan 仿真软件搭建测试场景, 建立一条长 1 km、平均通行速度 72 km/h 、三车道城市快速道路, 随机生成道路车辆, 计算获取路段交通流密度和交通流量, 实验路段和实验过程图如图 10、图 11 所示。

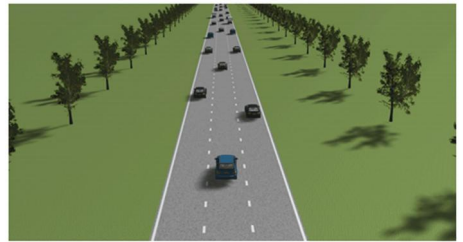


图 10 实验路段

Fig. 10 The experimental road

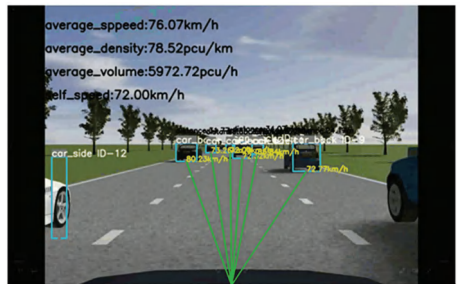


图 11 实验过程图

Fig. 11 Experimental process diagram

检测结果见表 4。

表 4 检测结果对比

Tab. 4 Comparison of detection results

	通行速度/ ($\text{km} \cdot \text{h}^{-1}$)	交通流密度/ ($\text{pcu} \cdot \text{km}^{-1}$)	交通流量/ ($\text{pcu} \cdot \text{h}^{-1}$)
实验路段	72.00	73.50	5 200.00
测量结果	64.15	67.76	4 816.32
误差/%	10.60	7.80	7.40

由表4可知,在稳定交通流条件下,通行速度、交通流密度和交通流量误差分别在10.6%、7.8%以及7.4%,均具有较高的检测精度。受到实验条件的限制,本文并未获得实验路段的真实交通流量数据,但是通过交通信息检测结果对比与分析可知,本文所提方法的检测结果能较好地反映城市道路交通运行状况。

5 结束语

本文提出了一种基于车载相机和GPS信息融合的城市道路交通检测方法,通过对目标车辆的检测跟踪,计算城市道路车辆的纵向间距和纵向速度,再结合GPS信息获得目标车辆的绝对速度,最终获取待测路段的通行速度、交通流密度和交通流量参数。实验结果与分析表明,本文所提的移动式交通信息检测方法能为智能车辆或智能交通系统提供可靠的数据支撑。由于城市道路交通运行环境比较复杂,实验中发现车辆遮挡问题会对检测结果造成一定的影响,未来研究将重点关注车辆遮挡问题对城市道路交通信息检测结果的影响。此外,本文假设试验车在采集数据过程中,路段的交通流特性没有变化,试验车以一条扫描线的形式对实验道路进行检测。因此,本文所述方法比较适用于交通流特性比较稳定的路段,如果路段的交通流特性出现较大波动,检测误差可能会相应增大。

参考文献

- [1] GAKENHEIMER R. Urban mobility in the developing world[J]. Transportation Research Part A: Policy and Practice, 1999,33(7-8): 671-689.
- [2] SHEPELEV V, ALIUKOV S, et al. The use of multi-sensor video

- surveillance system to assess the capacity of the road network[J]. Transport and Telecommunication, 2020, 21(1): 15-31.
- [3] CHEN Xiaoquan, ZHANG Jihong, QIAN Shao, et al. Applied research on traffic information collection based on wireless sensor networks[J]. Energy Procedia, 2012, 17: 602-606.
- [4] 叶青,刘剑雄,刘铮,等. 基于电磁感应的道路车辆车型在线分类方法研究[J]. 湖南大学学报(自然科学版),2019,46(12): 41-49.
- [5] CHEOL O, PARK S, RITCHIE S G. A method for identifying rear-end collision risks using inductive loop detectors[J]. Accid Anal Prev, 2006, 38(2): 295-301.
- [6] 曹林,李佳,张鑫怡,等. 一种基于微波雷达回波信号的车辆分类方法[J]. 电讯技术,2020,60(05):542-548.
- [7] LI S, CHANG F, LIU C, et al. Vehicle counting and traffic flow parameter estimation for dense traffic scenes[J]. IET Intelligent Transport Systems, 2020, 14(12): 1517-1523.
- [8] 袁向红. 基于浮动车数据的高速公路交通量预测模型研究[D]. 北京:北京交通大学,2019.
- [9] KONG Xiangjie, FENG Xia, NING Zhaoliu, et al. Mobility dataset generation for vehicular social networks based on floating car data[J]. IEEE Transactions on Vehicular Technology, 2018, 67(5): 3874-3886.
- [10] SHAN D, LEI T, YIN X, et al. Extracting key traffic parameters from UAV video with on-board vehicle data validation[J]. Sensors (Basel), 2021, 21(16):5620.
- [11] 关闯,魏朗,乔洁,等. 一种基于消隐点的单目视觉车辆测距方法[J]. 电子测量技术,2018,41(11):83-87.
- [12] GODHA S. On-road obstacle detection system for driver assistance[J]. Asia Pacific Journal of Engineering Science and Technology, 2017, 3(1): 16-21.
- [13] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas:IEEE, 2016: 779-788.
- [14] WOJKE N, BEWLEY A, PAULUS D. Simple online and realtime tracking with a deep association metric[C]//2017 IEEE International Conference on Image Processing (ICIP). Beijing, China:IEEE, 2017: 3645-3649.

(上接第16页)

- [9] HOCHREITER S and SCHMIDHUBER J. Long short-term memory[J]. Neural Computation, 1997, 29(1): 124-131.
- [10] LI Ruiyu, LI Kaican, KUO Yichun, et al. Referring image segmentation via recurrent refinement networks[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, USA: IEEE, 2018: 4461-4468
- [11] PENNINGTON J, SOCHER R, MANING C D. GloVe: Global vectors for word representation[C]// Conference on Empirical Methods in Natural Language Processing. Doha, Qatar: dblp, 2014: 3304-3311.
- [12] CORDTS M, OMRAN M, RAMOS S, et al. The cityscapes dataset for semantic urban scene understanding[C]// Conference on Computer Vision and Pattern Recognition. Las Vegas:IEEE, 2016: 3213-3223.

- [13] RICHTER S R, VINEET V, ROTH S, et al. Playing for data: Ground truth from computer games[C]// European Conference on Computer Vision. Amsterdam:dblp, 2016: 102-118.
- [14] ZOU Y, YU Z D, KUMAR B V, and WANG J. Glove. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training[C]// European Conference on Computer Vision. Munich, Germany:dblp, 2018: 289-305.
- [15] QBAL J, ALI M. Misl: Multi-level self-supervised learning for domain adaptation with spatially independent and semantically consistent labeling[J]. arXiv preprint arXiv: 1909.13776,2020.
- [16] PAN Fei, SHIN I, RAMEAU F, et al. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle, WA, USA: IEEE, 2020: 3764-3773.