

# 基于多帧叠加和窗口搜索的快速车道检测

陈涵深<sup>1,2</sup> 姚明海<sup>1</sup> 陈志浩<sup>1</sup> 杨 圳<sup>1</sup>

(浙江工业大学信息工程学院 杭州 310023)<sup>1</sup> (浙江交通职业技术学院 杭州 311112)<sup>2</sup>

**摘要** 车道检测是辅助驾驶和自动驾驶的重要研究内容。针对现有车道检测算法的鲁棒性和复杂度较难均衡等问题,提出一种基于多帧叠加和窗口搜索的快速车道检测算法。首先,通过逆透视变换(IPM)把指定的感兴趣区域(ROI)转换成鸟瞰图,结合多帧叠加的方法把RGB图像转化成二值图。其次,根据近视场中的像素密度分布,计算当前帧的车道线起始点,并采用滑动窗口搜索的方法提取整个车道线。最后,根据车道线的特征,选择不同的车道模型,使用最小二乘法(LSE)拟合得到模型参数。大量的实际道路行驶测试结果表明,该算法能快速地检测车道线,并具有一定的鲁棒性和准确性。

**关键词** 辅助驾驶,车道检测,多帧叠加,窗口搜索

中图分类号 TP391 文献标识码 A DOI 10.11896/j.issn.1002-137X.2018.10.047

## Efficient Method of Lane Detection Based on Multi-frame Blending and Windows Searching

CHEN Han-shen<sup>1,2</sup> YAO Ming-hai<sup>1</sup> CHEN Zhi-hao<sup>1</sup> YANG Zhen<sup>1</sup>

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)<sup>1</sup>

(Zhejiang Institute of Communications, Hangzhou 311112, China)<sup>2</sup>

**Abstract** Lane detection is one of the most important research areas in assistance driving and automated driving. Many efficient lane detection algorithms have been proposed recently, but most of them are still hard to achieve a balance between computational efficiency and accuracy. This paper presented a real-time and robust approach for lane detection based on multi-frame blending and windows searching. Firstly, the image is cropped and mapped to create a bird's-eye view of the road. Then, the RGB image is converted to a binary image based on a threshold of multi-frame blending. In the next step, the starting point of the lane line is calculated by using the pixel density distribution in the near field of view, and the whole lane is extracted by the method of sliding window search. Finally, according to the feature of candidate lane, different lane models are defined and chosen, and the model parameters are obtained by Least Square Estimation (LSE). The proposed algorithm shows good performance when tested on real-world data containing various lane conditions.

**Keywords** ADAS, Lane detection, Multi-frame blending, Windows searching

## 1 引言

科技的不断创新和经济的持续增长带动了交通事业的快速发展,人们在注重驾驶的安全性的同时也开始关注行驶的舒适性和高效性,因此辅助驾驶和自动驾驶汽车成为了重要的研究课题。车道检测是辅助驾驶和自动驾驶的基本功能,用于确定车辆在当前车道内的位置,为车道偏离决策提供判断依据。

目前基于视觉的车道检测方法主要有两大类:1)基于深度学习的车道检测技术<sup>[1-3]</sup>,主要采用多层的卷积神经网络,无需人为定义特征,能在行驶过程中通过不断学习完成车道的识别。该技术具有检测精度高、鲁棒性强等优点,美中不足

的是计算复杂度高,需要性能较强的GPU才能完成工作。2)基于传统方法的车道检测技术,主要有图像特征法和模型匹配法。图像特征法通过分析底层的图像特征,如车道颜色、边缘和纹理等,作为车道线分割和提取的依据<sup>[4-5]</sup>。模型匹配法主要采用不同的道路模型(如直线模型<sup>[6]</sup>、曲线模型<sup>[7]</sup>、B样条模型<sup>[8]</sup>等)来实现结构化的车道检测,此方法对固定结构的车道具有较好的识别效果,但适应性差,在光照变化、噪声干扰或车道线不连续的情况下达不到预期的检测效果。Borkar等<sup>[9]</sup>提出了一种随机抽样一致(Density-Based Spatial Clustering of Application with Noise, RANSAC)方法来排除车道线外的干扰,首先使用最小二乘法(Least Squares Method, LSE)拟合车道线,然后用卡尔曼滤波器(Kalman Filter)

到稿日期:2017-08-12 返修日期:2017-11-24 本文受浙江省教育厅项目(Y201635456),浙江省自然科学基金项目(LZ14F030001)资助。

陈涵深(1983-),男,博士生,工程师,主要研究方向为计算机视觉、机器学习, E-mail: chs9811@163.com; 姚明海(1963-),男,博士,教授,主要研究方向为机器学习、模式识别, E-mail: yhm@zjut.edu.cn(通信作者); 陈志浩(1993-),男,硕士生,主要研究方向为计算机视觉、机器学习; 杨 圳(1994-),男,硕士生,主要研究方向为机器学习。

对车道线的参数进行跟踪估计。Lee等<sup>[10]</sup>结合了模型分解方法,提出了采用级联粒子滤波器的鲁棒车道检测算法。Ju等<sup>[11]</sup>提出了一种基于消失点估计的车道检测方法,该方法从图像中提取线段的交叉点,采用概率投票方法解决消失点的问题。上述3种算法虽然精度较高,但存在计算量大、处理耗时较长等问题,并不适合在嵌入式系统中应用。

针对现有车道检测算法较难均衡鲁棒性和复杂度的问题,本文提出一种基于多帧叠加和滑动窗口搜索的快速车道检测算法。该算法具有以下优势:1)采用指定区域的逆透变换(Inverse Perspective Mapping, IPM)和多帧叠加的预处理方法,既减少了图像处理的数据量,又降低了周围环境的干扰,使得转换后的图像具有鲁棒性强等特点;2)利用像素密度分布、滑动窗口搜索、不同车道模型定义的方法检测车道线,算法不仅时间复杂度低,而且在汽车震动、车道标识线干扰、变更车道等条件下也能稳定地工作。

## 2 快速车道检测算法的流程

本文算法主要包括图像预处理、车道线识别、车道线拟合和车道线输出四大部分,如图1所示。图像预处理的主要流程为:首先,通过逆透变换(IPM)把指定的感兴趣区域转换成鸟瞰图(bird's-eye view);其次,为了提高边缘检测的效果,将黄色车道线和普通车道线分开检测,各自结合多帧叠加的方法把RGB图转换成二值图,并进行像素的“或运算”,合并两者的输出结果;最后,采用形态学开运算,删除不能包含结构元素的对象区域,平滑对象的轮廓,以排除二值化后的干扰和噪声信号。

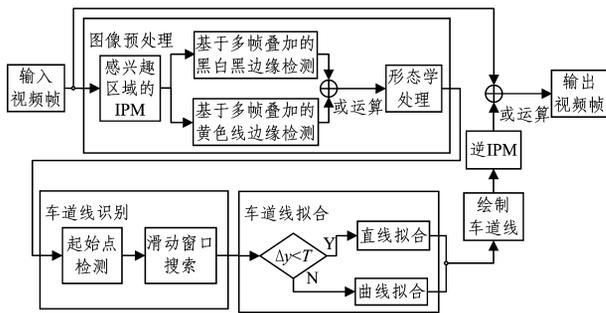


图1 快速车道检测算法流程图

Fig. 1 Flow chart of efficient lane detection algorithm

车道线识别是提取车道线像素点的过程,根据近视场中像素密度的分布,计算当前帧的车道线起始点,并通过滑动窗口搜索的方法获取车道线图像。车道线拟合采用最小二乘法(LSE),根据提取的车道线特征,动态选择直线模型或曲线模型来拟合车道线,输出车道模型函数。绘制车道线需要先在IPM坐标系上描绘车道线,再将其逆IPM到原始图像坐标系,采用像素“或运算”在原图像中标识出识别的车道线。

## 3 图像预处理

### 3.1 感兴趣区域的IPM

摄像头拍摄的图像包括路面和路面上方的信息,车道识别中只需要关注图像底部的路面信息,因此在原始的输入视频图像中选取梯形的感兴趣区域(Region of Interest, ROI)作

为IPM的输入图像。ROI的选择方法主要有类似 saliency 感兴趣区域<sup>[12-13]</sup>的自动检测法和固定参数<sup>[14]</sup>的选取法,本文选用固定参数的方法,如图2(a)所示,其中ROI的纵坐标范围为车前3~30m,去除路面上方的图像信息,ROI的宽度为我国国家规定的车道线宽度的3倍。只针对指定图像区域进行处理,不仅能减少非车道信息的干扰,而且能降低计算的时间复杂度。车道线图像具有一定的规律性,但由于拍摄视角的原因,车道线出现不平行和近宽远窄的特征。车道检测中广泛采用逆透视变换(IPM)<sup>[15]</sup>将相机获取的投影图像转换成鸟瞰图,本文利用相机内部参数和外部参数,通过4个角点的逆透视变换方法把车道线还原成一对等距离的平行线,并且车道线粗细均匀,图2(b)给出了IPM转换的结果。考虑到普遍适用性,实验中将IPM后的图像输出的分辨率统一设定为260×360,以便使后续算法能直接对固定大小的图像进行操作。



图2 感兴趣区域的IPM样例图

Fig. 2 Example of ROI-based IPM

### 3.2 基于多帧叠加的黑白边缘检测

#### 3.2.1 多帧叠加

多帧叠加前,需要把IPM输出的RGB图像转换成灰度图像,然后进行多帧叠加,即把每次输入的N帧图像通过算法叠加后输出一帧图像。该算法具有以下优势:1)对于车辆行驶中拍摄的视频,相邻帧之间的图像变化较少,不需要处理每一帧图像,采用多帧叠加的方法可以减少部分重复计算,保证了算法的实时性;2)多帧叠加后,可减轻消失的车道线对检测结果的影响,提高了车道识别的鲁棒性。

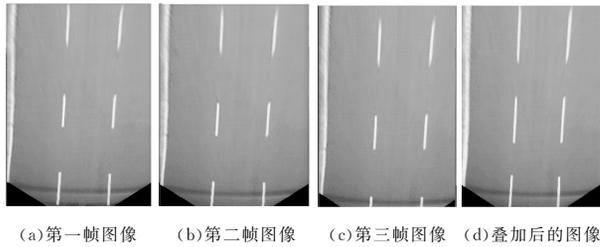
该算法的思想是保留第一帧和之后相邻帧之间亮度差较大的像素点,采用迭代计算,迭代次数为需要叠加的帧数。具体计算公式如下:

$$\begin{cases} L^{(1)} = B_1 \\ L^{(n)} = \Delta_{L^{(n-1)} B_n} \end{cases}, n=1, 2, \dots, N \quad (1)$$

其中,  $B_1$  代表第一帧图像,  $\Delta_{L^{(n-1)} B_n}$  为前  $n-1$  帧叠加的结果和第  $n$  帧图像的“叠加”操作。设第  $i$  帧和第  $i-1$  帧中每个像素点的亮度分别为  $b_i(x, y)$ ,  $b_{i-1}(x, y)$ , 亮度差阈值为  $T$ , 则两帧“叠加”的计算公式如下:

$$\Delta_{B_i B_{i-1}} = \begin{cases} b_i(x, y), & b_i(x, y) - b_{i-1}(x, y) > T \\ b_{i-1}(x, y), & \text{otherwise} \end{cases} \quad (2)$$

通常汽车的前向速度明显大于横向速度,换道时,在几毫秒内横向的行驶距离也非常小,因此车道标记的宽度在模糊后可以忽略不计。在弯曲道路上,车道的曲率逐渐变化,而不是突然变化。叠加帧数  $N$  的具体数值需要根据汽车行驶的车速、相机拍摄的帧数等动态变化。在实验中发现,在车速为30km/h时,  $N$  取3对车道线的影响非常轻微,其效果如图3所示。



(a)第一帧图像 (b)第二帧图像 (c)第三帧图像 (d)叠加后的图像

图 3 多帧叠加的效果图

Fig. 3 Pictures of multi-frame blending

### 3.2.2 黑白黑边缘检测

考虑到大部分车道线是白色,车道线相对其两侧的亮度较高,并且两侧亮度差值之和在一定范围内,采用黑白黑边缘检测算法<sup>[15]</sup>对其做二值化处理。通过实验发现,加入式(5)后,检测的效果会更好。

$$d_{+m} = b(x, y) - b(x, y + B) \quad (3)$$

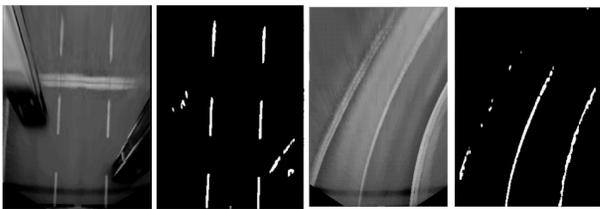
$$d_{-m} = b(x, y) - b(x, y - B) \quad (4)$$

$$D = d_{+m} + d_{-m} - |b(x, y + B) - b(x, y - B)| \quad (5)$$

其中,  $d_{+m}$  和  $d_{-m}$  代表水平亮度梯度;  $B$  表示车道线宽度在具体图像中的像素点个数,根据车道线宽度调整  $B$  值,可以使算法适应各种车道线的宽度。输出  $L(x, y)$  为:

$$L(x, y) = \begin{cases} 1, & \text{if } d_{+m} > 0 \text{ and } d_{-m} > 0 \text{ and } D > K * b(x, y) \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

其中,  $K$  为亮度差系数。通过实验对千帧不同道路、不同光照条件下的图像进行统计分析,可得  $B$  设定在 6~12 之间且亮度差  $K$  选取 0.15~0.3 时算法具有较好的效果。图 4 展示了算法的检测效果,其中图 4(a) 和图 4(c) 为原图,图 4(b) 和图 4(d) 分别为图 4(a) 和图 4(c) 经过黑白黑边缘检测算法检测后的结果。可以看到,该算法能排除部分汽车和水平标识线的干扰,也能去除大块明亮的区域。



(a)汽车干扰 (b)图(a)的二值化结果 (c)弯曲车道 (d)图(c)的二值化结果

图 4 黑白黑边缘检测效果图

Fig. 4 Results of black-white-black edges detection

### 3.3 基于多帧叠加的黄色线边缘检测

对车道线图像进行研究后可以发现,在道路光照较强时,黄色车道线经灰度变换后的亮度反而比两侧的暗,因此无法采用黑白黑边缘检测算法进行分割,需要结合其他方法进行弥补。本文先将 IPM 输出的 RGB 图像转换成 HSV 图像,然后利用阈值分割的方法提取黄色线。HSV 模型侧重于色彩表示,在该空间内可以设定一定的色调(H)范围、饱和度(S)范围、明度(V)范围。将 HSV 模型用于指定颜色的分割具有较好的效果,具体计算公式如下:

$$L(x, y) = \begin{cases} 1, & \text{if } H_l < h(x, y) < H_u \text{ and } S_l < s(x, y) < S_u \\ & \text{and } V_l < v(x, y) < V_u \\ 0, & \text{otherwise} \end{cases} \quad (7)$$

其中,  $H_l, S_l, V_l$  和  $H_u, S_u, V_u$  分别是色调、饱和度、明度的下限和上限。完成黄色线边缘检测后,采用多帧叠加的方法来延长车道线并减少处理时间。与已有研究不同,本文先进行 HSV 二值化,再进行多帧叠加,叠加方法采用像素“或运算”。因为黄色车道线在整个图像中往往表现得比较单一,很少存在其他非车道线的黄色图像,所以对二值化后的像素直接进行操作不会产生干扰噪声。

## 4 车道线识别

### 4.1 车道线起始点检测

车道线具有以下 3 个特点:1)图像通过 IPM 转换成鸟瞰图后,车道线被还原成一对等距离的平行线,并且起点处的线段往往垂直于横坐标轴;2)国家对车道的宽度有明确的规定,左右车道线之间的距离在一定范围内;3)对于车辆行驶中拍摄的视频,车道线起始点在前后帧之间的位置变化较小。本文根据上述特点,提出了一种基于像素密度分布的方法来获取左右两边的车道线起始点。该算法不仅能识别虚实线和实线车道线的起始点,而且能有效地排除周边多车道线、汽车和地标线等的干扰。算法的具体步骤如下:

Step1 在近视场部分截取图像,如图 5 所示。沿  $y$  轴计算图像中的像素密度分布,即按一个像素宽度进行垂直切片,计算其中的白色像素点个数。设  $d(y)$  为  $y$  点坐标处的垂直像素密度,  $k$  为纵坐标近视场和远视场的分割点,则:

$$d(y) = \sum_{x=k}^{height-1} d(x, y), y=0, 1, 2, \dots, width-1$$

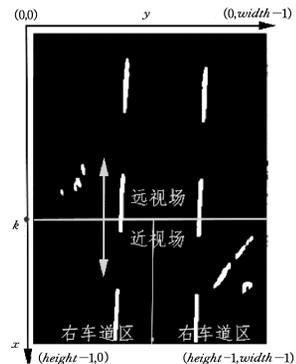


图 5 图像坐标系与近视场和远视场的定义

Fig. 5 Image coordinate system and definition of near field and far field

Step2 根据像素密度值是否连续大于零,即  $d(y) \neq 0$  并且  $d(y+1) \neq 0$ ,定义每个独立区域块为  $B_1, B_2, \dots$ ;然后求每个独立区域的局部最大值,即  $m(y)_i = \max(d(y)), d(x) \in B_i$ 。

Step3 给定一个阈值  $T$ ,排除像素密度较小的干扰信号,公式如下:

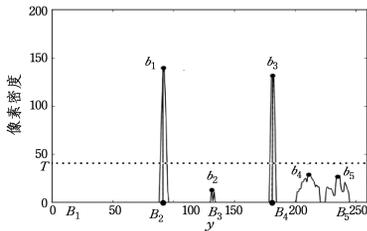
$$m(y)_j = \begin{cases} m(y)_i, & \text{if } m(y)_i > T \\ 0, & \text{otherwise} \end{cases} \quad (8)$$

Step4 对于余下的局部最大值,根据横坐标位置,在左车道区域内选取最多  $P$  个最大值点作为左车道起始点的候选点。同样,右车道起始点的候选点在右车道区域选取,  $P$  一般设定为 2。

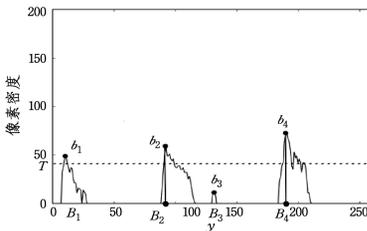
Step5 计算每个点的置信度。依据左右车道线之间的距离在一定范围内的特点,计算左右候选点之间的相互距离,符合距离要求的点加1分。根据车道线起始点在前后帧之间的位置变化较小的要求,分别在左车道区域和右车道区域内计算前一帧起始点和候选点之间的距离,距离较小的点加1分。最后得到每个候选点的置信度。

Step6 在左车道候选点中,将置信度得分最高的点作为左车道线的起始点,如果有多个相同的最大置信度点,那么使用其中像素密度最大的点作为起始点。右车道线的起始点也采用同样的方法选取。

在应用该算法时,如果左车道或者右车道的起始点检测失败,则可以采用上一帧的起始点作为当前的起始点,如果下一帧还是失败,就认为未检测到车道线。使用该技巧可以提高检测的鲁棒性。图6展示了该算法的效果。



(a)图4(b)的结果



(b)图4(d)的结果

图6 像素密度分布图

Fig. 6 Density distribution of pixel density

图6(a)和图6(b)分别为计算得到的图4(b)和图4(d)的像素密度分布图,其中横坐标上左右两个圆点分别为计算得到的左车道起始点和右车道起始点的横坐标位置。

#### 4.2 基于窗口搜索的车道线识别

得到车道线的起始点后,采用滑动窗口纵向搜索的方式提取车道线图像。为了避免获取过多的车道线外围干扰点,将滑动窗口的宽度设定为车道线宽度的3倍,滑动窗口高度的设定需要考虑车道线的最大斜率,本文中取滑动窗口宽度的2/3作为滑动窗口的高度。搜索算法首先从图像底部开始,在左车道线和右车道线分别搜索,把4.1节得到的左车道起始点作为左车道第一个滑动窗口的种子点  $S_1$ ,以种子点  $S_1$  为中心、1/2 滑动窗口宽度为半径获取窗口内的图像。然后计算该图像数据得到下一个滑动窗口的种子点  $S_2$ ,以此遍历,直到窗口到达图像的顶部或者图像的边缘区域为止。滑动窗口位置的具体计算如下。

如图7所示,设当前第  $i$  个滑动窗口的种子点为  $S_i$ ,下一个滑动窗口的种子点为  $S_{i+1}$ ,在当前滑动窗口的底部,从下往上逐行扫描,直到扫描到第一个白色像素点,设其  $y$  轴坐标值为  $y_{i1}$ ;同样在滑动窗口的顶部,自上而下逐行扫描得到第一个白色像素点,设其坐标值为  $y_{i2}$ ,则种子点  $S_{i+1}$  的坐标值  $x_{s_{i+1}}$  和  $y_{s_{i+1}}$  分别为:

$$x_{s_{i+1}} = x_{s_i} - W_h \tag{9}$$

$$y_{s_{i+1}} = \begin{cases} y_{s_i}, & \text{if } y_{i1} = 0 \text{ and } y_{i2} = 0 \\ y_{i2} + \frac{B}{2} + \frac{|y_{i2} - y_{i1}|}{2}, & \text{otherwise} \end{cases} \tag{10}$$

其中,  $B$  为车道线宽度,  $W_h$  为滑动窗口的高度,实验中选用  $W_h = 18$ 。根据图7可以推出式(10)的结果,其中  $y_{i1} = 0$  and  $y_{i2} = 0$  表示未找到  $y_{i1}$  点和  $y_{i2}$  点,说明滑动窗口内没有任何像素点,下一个滑动窗口按照当前位置的中心点向上搜索。其他情况分为3类,分别为斜率大于零、斜率等于零、斜率小于零,下一个种子点的横坐标位置等于  $y_{i2}$  加上偏移量。该算法的实际效果如图8所示。

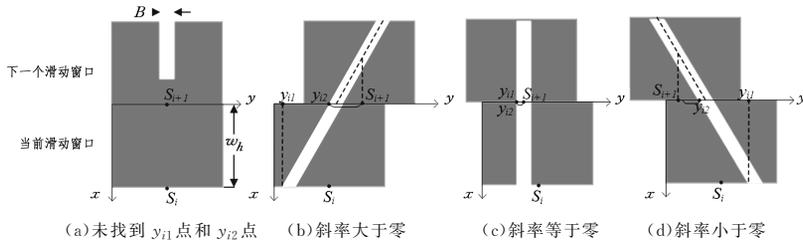
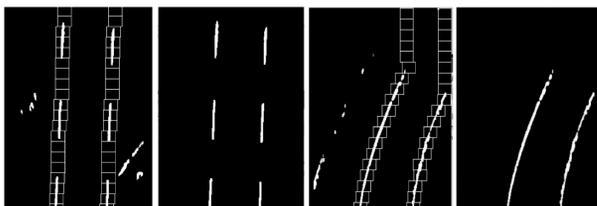


图7 滑动窗口算法

Fig. 7 Sliding window algorithm



(a)图4(b)的滑动窗口 (b)图(a)的车道线识别结果 (c)图4(d)的滑动窗口 (d)图(c)的车道线识别结果

图8 车道线的识别过程

Fig. 8 Process of lane detection

#### 5 车道线拟合

对于车道线,常采用最小二乘法进行拟合,但最小二乘法对噪声敏感,只有高质量的数据才能保证拟合精度。随机抽样一致(RANSAC)算法<sup>[16]</sup>能克服最小二乘法易受外点影响的缺点,但需要迭代计算,处理速度相对较慢。为了提高车道检测系统的鲁棒性和实时性,除了采用较小宽度的滑动窗口来排除外点干扰外,本文还将根据车道线像素点的横坐标位置差的大小,动态地选择直线模型或曲线模型来拟合车道线。如果是直线车道,车道线通过IPM转换成鸟瞰图后往往垂直

于横坐标,车道线像素点的横坐标值  $\Delta y$  变化较小,车道模型选用线性方程;如果是弯曲车道,IPM 后的车道线斜率为非零,车道线像素点在横坐标上值  $\Delta y$  变化较大。因此,将车道模型定义为二次多项式方程,具体函数如下:

$$x = \begin{cases} \frac{1}{2}C_0y^2 + my + b, & \Delta y \geq T \\ ay + b, & \Delta y < T \end{cases} \quad (11)$$

其中,  $C_0$  是车道线的曲率,  $b$  是偏移量,  $a$  和  $m$  是斜率。这些参数根据 4.2 节所得到的车道线像素点,采用最小二乘法拟合获取。

## 6 实验分析

为了验证本文方法的实际效果,采用 C++ 语言,基于 OpenCV3.1.0 库在 ubuntu16.04 操作系统下进行实验。所用微机 CPU 为 Intel i5-4202Y 双核 1.50GHz,内存为 2GDDR3,摄像头采用罗技 C525,安装在车辆挡风玻璃的正前方。获取的视频参数为  $1028 \times 720$  像素,15 帧/秒。测试硬件如图 9 所示。实验样本为实采于杭州市内的古墩路、莫干山路、石祥路和绕城高速的道路视频。



图 9 测试设备的安装位置

Fig. 9 Installation location of test set system

### 6.1 实际车道线检测

根据道路类型对采集的视频进行剪辑和分类,得到 5 种类型,然后对其中的一类添加随机高斯噪声。实验结果如表 1 和图 10 所示。结果表明,汽车在  $30 \sim 90$  km/h 的速度下行驶时,算法受噪声干扰、标识干扰、路面颠簸、行车干扰和车道变更的影响较小,具有一定的正确性和鲁棒性。

表 1 不同道路类型的检测正确率

Table 1 Detection accuracy under various lane conditions

道路类别	有效视频帧数	漏检率/%	误检率/%	正确率/%
典型道路	4080	1.64	0.91	97.45
典型道路(加噪)	4080	6.15	2.96	91.18
标识干扰	900	2.11	4.33	93.55
路面颠簸	200	3.50	2.50	94.00
行车干扰	290	4.48	7.93	87.59
车道变更	96	2.08	4.17	93.75

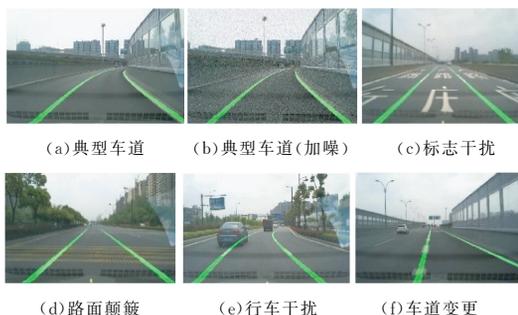


图 10 不同道路类型下的车道线检测结果

Fig. 10 Lane detection results under various lane conditions

### 6.2 与其他车道线检测方法的分析和比较

现有快速车道检测算法多采用直线模型作为车道线模型,如文献[17-19]。直线模型降低了特征点提取的要求和拟合的时间复杂度,具有实时性高、直道检测鲁棒性好等优点,但在弯道检测中难以保证拟合精度,易出现车道的误检。而本文算法动态选择直线模型或曲线模型来拟合车道线,既能保留直线模型的优点,又能保证弯道检测的正确性。

为了验证本文算法的性能,采用相同的样本(典型道路的 4080 帧有效视频),与文献[14]的方法进行对比测试。由表 2 的实验结果可知,本文方法略优于文献[14]的方法,而且检测速度提升明显。同时,通过统计对比文献[20-21]的实验结果可知,本文算法处理每一帧的平均时间为 16 ms,少于文献[20]利用 Otsu 法的 46 ms 和文献[21]的 29 ms。

表 2 本文方法和其他方法的性能比较

Table 2 Performance comparison between proposed method and other methods

方法	平均测试精度/%	耗时/ms
文献[14]的方法	96.10	209
本文方法(多帧叠加)	97.45	16
本文方法(全帧率)	97.32	27
跳帧检测+本文方法	96.74	15

为了验证多帧叠加的效果,分别用多帧叠加法、全帧率法和文献[20]中的跳帧方法进行图像的预处理,然后使用本文的车道检测和拟合方法对识别结果进行比较,统计结果如表 2 所列。可以看出,相比全帧率方法,采用多帧叠加( $n=2$  帧)的车道线检测精度提升了 0.13%,耗时减少了 40.7%;相比跳帧( $n=2$  帧)的方法,精度提升了 0.6%。

**结束语** 本文通过实验验证了所提算法具有很好的实时性、正确性和鲁棒性,可应用于嵌入式设备,具有一定的实际应用价值。本文的主要创新点如下:1)在预处理阶段提出了结合多帧叠加的二值化方法,其既能减小消失车道线对检测的影响,又能保证算法的实时性;2)在车道线识别环节提出了基于像素密度分布的车道线起始点检测方法,并利用滑动窗口搜索的方法提取车道线,该算法具有时间复杂度低、鲁棒性高、实现简单等特点;3)在车道模型定义上,动态选择线性模型或二次多项式模型,采用最小二乘法拟合获取参数,以提升算法的实时性。虽然本文算法已取得不错的效果,但在多帧叠加时未考虑车辆在高速行驶或急转时可能出现的视频模糊问题,如果能计算出下一帧的偏移量后再进行位移叠加,检测效果可能会更好,这将作为下一步研究的重点。

## 参考文献

[1] GURGHIAN A, KODURI T, BAILUR S V, et al. DeepLanes: End-To-End Lane Position Estimation using Deep Neural Networks [C]// Computer Vision & Pattern Recognition Workshops, 2016:38-45.

[2] LI J, MEI X, PROKHOROV D. Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene [J]. IEEE Transactions on Neural Networks & Learning Systems, 2016, 28(3):1-14.

[3] CHEN C, SEFF A, KORNHAUSER A, et al. DeepDriving:

- Learning Affordance for Direct Perception in Autonomous Driving [C]//IEEE International Conference on Computer Vision, 2015;2722-2730.
- [4] JEONG P, NEDEVSKI S. Efficient and robust classification method using combined feature vector for lane detection [J]. Circuits and Systems for Video Technology, 2005, 15(4): 528-537.
- [5] GOPALAN R, HONG T, SHNEIER M, et al. A learning approach towards detection and tracking of lane markings [J]. Intelligent Transportation Systems, 2012, 13(3): 1088-1098.
- [6] SATZODA R K, SATHYANARAYANA S, SRIKANTHAN T, et al. Hierarchical additive hough transform for lane detection [J]. Embedded Systems Letters, 2010, 2(2): 23-26.
- [7] CHEN Q, WANG H. real-time lane detection algorithm based on a hyperbola-pair model [C]// Intelligent Vehicles Symposium, Tokyo; IEEE, 2006; 510-515.
- [8] WU B F, LIN C T, CHEN Y L. Dynamic calibration and occlusion handling algorithms for lane tracking [J]. IEEE Transactions on Industrial Electronics, 2009, 56(5): 1757-1773.
- [9] BORKAR A, HAYES M, SMITH M T. Robust lane detection and tracking with ransac and Kalman filter [C]//IEEE International Conference on Image Processing, 2010; 3225-3228.
- [10] LEE M, JANG C, SUNWOO M. Probabilistic lane detection and lane tracking for autonomous vehicles using a cascade particle filter [J]. Proceedings of the Institution of Mechanical Engineers Part D Journal of Automobile Engineering, 2015, 229(12): 1-15.
- [11] JU H Y, LEE S W, PARK S K, et al. A Robust Lane Detection Method Based on Vanishing Point Estimation Using the Relevance of Line Segments [J]. IEEE Transactions on Intelligent Transportation Systems, 2017, PP(99): 1-13.
- [12] WANG W, SHEN J, SHAO L. Consistent video saliency using local gradient flow optimization and global refinement [J]. IEEE Transactions on Image Processing, 2015, 24(11): 4185-4196.
- [13] WANG W, SHEN J, PORIKLI F. Saliency-aware geodesic video object segmentation [C]// Computer Vision & Pattern Recognition, 2015; 3395-3402.
- [14] ALY M. Real time detection of lane markers in urban streets [C]// IEEE Intelligent Vehicles Symposium, 2008; 7-12.
- [15] BERTOZZI M, BROGGI A. GOLD; A parallel real-time stereo vision system for generic obstacle and lane detection [J]. IEEE Transactions on Image Processing, 1998, 7(1): 62-81.
- [16] FISCHLER M A, BOLLES R C. Random sample consensus; A paradigm for model fitting with applications to image analysis and automated cartography [J]. Communications of ACM, 1981, 24(6): 381-395.
- [17] HAN G F, LI X M, WU X. Research of lane line detection in the vision navigation of unmanned vehicle [J]. Fire Control Command Control, 2015, 40(6): 152-158. (in Chinese)  
韩广飞, 李晓明, 武潇. 无人驾驶汽车视觉导航中车道线检测的研究 [J]. 火力与指挥控制, 2015, 40(6): 152-158.
- [18] LI C, LIU H Z, YUAN J Z, et al. Real-time Lane Detection Algorithm Based on Inter-frame Correlation [J]. Computer Science, 2017, 44(2): 317-323. (in Chinese)  
李超, 刘宏哲, 袁家政, 等. 一种基于帧间关联的实时车道线检测算法 [J]. 计算机科学, 2017, 44(2): 317-323.
- [19] CHEN W W, JIANG Y T, TAN D K. A Fast Lane Marking Recognition Algorithm Based on Edge Projection [J]. Automotive Engineering, 2017, 39(3): 357-363. (in Chinese)  
陈无畏, 蒋玉亭, 谈东奎. 一种基于边缘点投影的车道线快速识别算法 [J]. 汽车工程, 2017, 39(3): 357-363.
- [20] SONG R, CHEN H, XIAO Z G, et al. Lane detection algorithm based on geometric moment sampling [J]. Computer Science, 2017, 44(2): 455-467. (in Chinese)  
宋锐, 陈辉, 肖志光, 等. 基于几何矩采样的车道线检测算法 [J]. 中国科学: 信息科学, 2017, 47(4): 455-467.
- [21] HALOIM J, JAYAGOPI D B. A robust lane detection and departure warning system [C]// IEEE Intelligent Vehicles Symposium (IV), 2015: 126-131.

(上接第 254 页)

- [9] BROWNRIGG D. The weighted median filter [J]. Image Processing and Computer Vision, 1984, 27(8): 807-818.
- [10] ROY A, SINGHA J, MANAM L, et al. Combination of adaptive vector median filter and weighted mean filter for removal of high-density impulse noise from colour images [J]. IET Image Processing, 2017, 11(6): 352-361.
- [11] ZHANG Z, HAN D Q, DEZERT J, et al. A new adaptive switching median filter for impulse noise reduction with pre-detection based on evidential reasoning [J]. Signal Processing, 2018, 147: 173-189.
- [12] ZENG X Y, HUANG Z H, ZHOU J Z. Switching median filter with boundary discriminative noise detection [J]. Computer Engineering and Applications, 2014, 50(14): 176-179. (in Chinese)  
曾宪佑, 黄佐华, 周进朝. 基于差分分层噪声检测的开关中值滤波算法 [J]. 计算机工程与应用, 2014, 50(14): 176-179.
- [13] ZHANG S, KARIM M. A new impulse detector for switching median filters [J]. IEEE Signal Processing Letters, 2002, 9(11): 360-363.
- [14] NG P, MA K. A switching median filter with boundary discriminative noise detection for extremely corrupted images [J]. IEEE Transactions on Image Processing, 2006, 15(6): 1506-1516.
- [15] DONG Y Q, XU S F. A new directional weighted median filter for removal of random-valued impulse noise [J]. IEEE Signal Processing Letters, 2007, 14(3): 193-196.
- [16] LU C T, CHOU T C. Denoising of salt-and-pepper noise corrupted image using modified directional weighted median filter [J]. Pattern Recognition Letters, 2012, 3(10): 1287-1295.
- [17] LI Z Y, LIU G H, XU Y, et al. Modified directional weighted filter for removal of salt & pepper noise [J]. Pattern Recognition Letters, 2014, 40(15): 113-120.
- [18] LU C T, CHEN Y Y, WANG L L, et al. Removal of salt-and-pepper noise in corrupted image using three-values-weighted approach with variable-size window [J]. Pattern Recognition Letters, 2016, 80(C): 188-199.