

# 基于改进 SIS 算法和顺序 RANSAC 的车道线检测方法研究

段建民, 李岳, 庄博阳

(北京工业大学 信息学部, 北京 100124)

**摘要:** 为了降低车道线识别算法在车道线存在阴影遮挡、路面出现泛白现象等不同道路环境下的误检率, 提出了一种基于改进简单图像统计 (Simple Image Statistics, SIS) 阈值算法和改进的顺序随机抽样一致性 (Random Sample Consensus, RANSAC) 的车道线检测算法; 首先, 在图像预处理阶段采用改进的 SIS 阈值算法进行二值化; 然后构建车道线模型, 并简化车道线模型为双曲线模型; 之后采用改进的顺序 RANSAC 算法拟合车道线; 最后根据两边的车道线模型进行模型配对, 通过选取最多支持数据点的组合确定车道线; 分别采用加州理工学院的车道数据集和实际采集的城市快速路视频对所提出方法进行了实验验证, 实验结果表明, 该算法误检率低, 为 2.32%, 鲁棒性高, 能在复杂环境下快速、准确识别车道线。

**关键词:** 阈值分割; 车道线模型; 随机抽样一致性; 模型配对; 误检率

## Lane Line Detection Method Research Based on Improved Algorithm of SIS and Sequential RANSAC

Duan Jianmin, Li Yue, Zhuang Boyang

(Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China)

**Abstract:** In order to reduce the false positive rate of the lane recognition algorithm in different road circumstances such as the lane is prone to the occlusion of shadow and the pavements appear blanched. A lane detection algorithm based on improved SIS threshold algorithm and modified version of sequential RANSAC is proposed. First of all, the improved SIS threshold algorithm is used to transform the image data into binary data, during the image preprocessing stage. Then, a lane line model is constructed, and it is simplified to a hyperbolic model. Moreover, the modified sequential RANSAC algorithm is applied to detect whether there exists lane line in the image. Finally, model pairing is carried out according to the lane line model of both sides, and lane line is determined by selecting the combination of the most data points. The method is verified by using California institute of technology's driveway data set and the actual city fast road video. The results show that the algorithm has low false positive rate which is 2.32% and high robustness. Moreover, the lane line can be identified precisely and quickly in the complex environment.

**Keywords:** threshold segmentation; lane model; random sample consensus; model pairing; false positive rate

## 0 引言

近年来, 由于汽车保有量与使用率的增加, 交通事故的发生呈现逐年上升的趋势。其中汽车作为交通道路中的主要参与者, 保证汽车的行车安全是智能车辆研究领域的重要课题。随着先进驾驶辅助系统 (Advanced Driver Assistance Systems, ADAS) 的有效应用, 智能车辆在辅助驾驶和遇险报警等方面的功能已经日趋完善。如何快速、准确的检测出城市道路中的车道线, 有效提升驾驶系统的鲁棒性及安全性, 为后续的车道保持、车辆避障与路径规划提供可靠保障是智能车辆自动驾驶技术亟需解决的问题。

许多研究者对车道线检测进行了深入的研究。常见的方法是先进行图像预处理, 包括灰度化、图像降噪和二值

化等, 然后采用边缘检测算法<sup>[1]</sup>和 Hough 变换<sup>[2]</sup>检测车道线。为了提高算法的实时性, Liu 等<sup>[3]</sup>采用 LSD 算法<sup>[4]</sup>检测车道线, 但没有考虑线段的相关性, 所以在噪声环境中提取车道线时会受到干扰。Guo 等<sup>[5]</sup>采用 LDA 对道路图像进行有针对性的灰度化, 然后再利用 LSD 提取直线, 但没有考虑道路其它标志的干扰。Lee<sup>[6-7]</sup>等提出一种基于特征的车道线估计方法, 采用直线模型匹配道路边界, 抗干扰能力强, 但弯曲的道路边界会提取失败。还有一些研究者利用卡尔曼滤波<sup>[8]</sup>和粒子滤波<sup>[9]</sup>利用当前帧对下一帧进行估计, 但是由于算法复杂度高, 实时性会有所下降。

## 1 改进 SIS 阈值算法

通过阈值化对图像进行处理, 可去除大量的无用信息, 提高车道线拟合与检测的准确性。根据路面环境特点, 本文提出一种改进 SIS 阈值<sup>[10]</sup>分割算法和改进的顺序 RANSAC 算法用于车道线拟合与检测。

本文算法主要包括以下几步, 算法主要流程如图 1 所示。第一步, 如图 2 所示, 截取车道线感兴趣区域 (ROI), 并进行灰度化和图像降噪处理; 第二步, 采用改进的 SIS 阈

收稿日期:2018-05-24; 修回日期:2018-07-08。

基金项目:国家自然科学基金项目(61573029);北京市自然科学基金项目(4162012)。

作者简介:段建民(1959-),男,河北石家庄人,博士,教授,主要从事自动驾驶、车辆控制方向的研究。

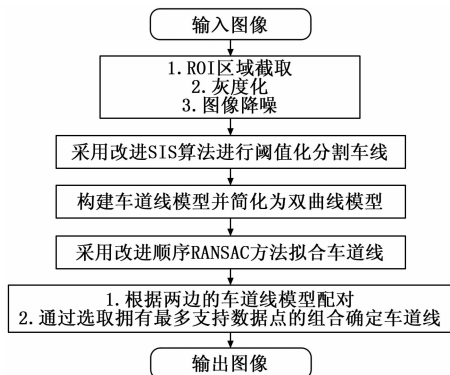


图 1 改进 SIS 算法与改进顺序 RANSAC 算法流程图

值算法对图像进行阈值化; 第三步, 构建车道线模型并简化为双曲线模型; 第四步, 利用改进顺序随机抽样一致性 (RANSAC) 算法拟合车道线, 然后根据两边的车道线模型进行模型配对; 最后, 通过选取包含最多支持数据点的组合确定车道线, 与传统的 Hough 变换和 LSD 等方法相比, 提出的算法不仅可以检测直线车道线, 还可以进行曲线车道线的拟合, 而且在有效提高车道线识别准确率的同时也满足实时性要求, 鲁棒性也有所提高。



图 2 车道线感兴趣区 (ROI)

SIS 算法是一种基于图像本身统计特性的阈值算法, 它执行速度快, 可以自动选择阈值, 在目标区域与背景区域有明显梯度的图像中有很好的效果。假设图像为  $f(u, v)$ , 算法过程如下:

1) 计算每个像素的水平方向梯度和垂直方向梯度, 如式 (1) 所示:

$$\begin{cases} e_u = |f(u-1, v) - f(u+1, v)| \\ e_v = |f(u, v-1) - f(u, v+1)| \end{cases} \quad (1)$$

2) 取水平方向梯度和垂直方向梯度的最大值作为这个像素的权重, 然后计算权重之和、加权像素之和, 如式 (2) 所示:

$$\begin{cases} S_w = \sum_u \sum_v \max(e_u, e_v) \\ S_{wd} = \sum_u \sum_v \max(e_u, e_v) f(u, v) \end{cases} \quad (2)$$

3) 加权像素之和和权重之和的比值即为最终的阈值, 如式 (3) 所示:

$$T = \frac{\sum_u \sum_v \max(e_u, e_v) f(u, v)}{\sum_u \sum_v \max(e_u, e_v)} \quad (3)$$

其中:  $f(u, v)$  是像素点  $(u, v)$  的灰度值。

SIS 算法在寻找阈值时, 降低非边缘区域的像素权值, 提高梯度边缘的像素权值, 在处理噪声低的图像时效果良好。但在噪声多的图像中, 并且图像中的噪声灰度值介于目标灰度值和背景灰度值, 那么得到的阈值会变小。如果图像中的噪声灰度值大于目标区域的灰度, 得到的阈值会变大。所以在一些复杂情况下该算法效果一般。

所以本文对 SIS 算法进行了改进, 主要思想是由于车道线的亮度总是高于其周围的道路, 所以将亮于邻近点的区域分离出来, 也就相当于将车道线部分分离出来。首先计算像素点周围  $r$  邻域内的灰度值, 并对灰度值做加权平均, 然后用原灰度值减去加权平均的灰度值, 如果小于零, 将该点置零。

这种方法受  $r$  的影响很大, 如果  $r$  取的过大, 不会影响计算速度, 且分离效果不理想。如果  $r$  取的过小则只会把图像中的边缘部分突出, 无法有效去除图像中的噪声。结合车道线检测的道路环境, 对  $r$  取不同的值进行多次计算对比, 当  $r = 7$  时可取得最佳效果, 灰度图像的处理结果如图 3 所示。

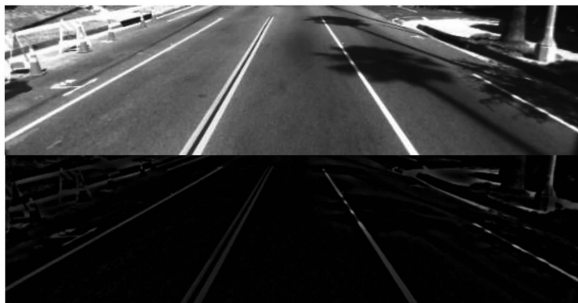


图 3 灰度原图以及高亮区分离图

图 4 (a) 是图 3 的灰度直方图曲线, 在这种情况下, 一方面由于分离出的高亮车道线区域周围像素灰度值均为 0, 并且这些像素处在高亮区域边缘, 权值较大, 那么阈值会变小; 另一方面存在高亮度噪点, 这些噪点的权值也较大, 那么阈值会变大。所以很难通过基于直方图的算法求得理想的阈值。

为降低这些因素的影响, 本文采用像素本身灰度值的平方根作为该像素的权值。这样可以在去除灰度值为 0 的像素的影响的同时降低高亮噪点对阈值的影响。阈值的计算公式如式 (4) 所示:

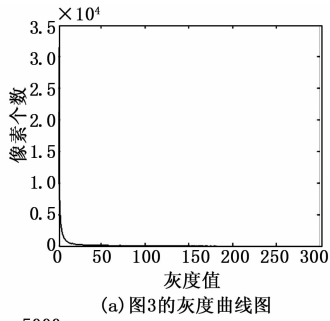
$$T = \frac{\sum_x \sum_y f(x, y) \times \sqrt{f(x, y)}}{\sum_x \sum_y \sqrt{f(x, y)}} \quad (4)$$

改进算法与比原算法节省大量计算, 因为只需要对原直方图进行计算便可以得到阈值, 而不再需要对每个像素进行计算。根据直方图计算阈值的公式为:

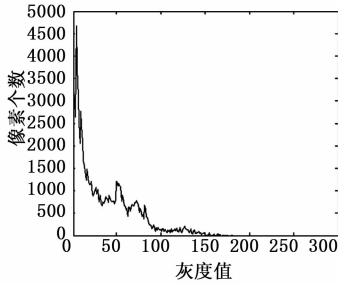
$$T = \frac{\sum_i i \times \sqrt{i} \times P_i}{\sum_i \sqrt{i} \times P_i} \quad (5)$$

式中,  $i$  为灰度值,  $P_i$  为该灰度值的像素个数。

加权后的直方图曲线如图 4 (b) 所示, 改进的算法与



(a) 图3的灰度曲线图



(b) 加权后的灰度曲线图

图 4 灰度曲线图

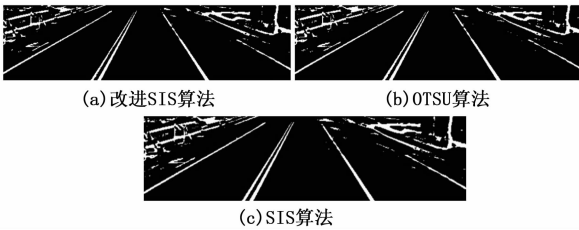


图 5 路面泛白情形下二值化对比

原算法及 OTSU 算法的效果对比如图 5~7 所示，其中图 5 的原图为图 2。图 5 是有阴影干扰和路面泛白的情形下，SIS 算法、OTSU 算法和本文的算法都取得了好的效果。图 6 是在低亮度和有阴影干扰的情况下，本文的算法在保留了车道线前提下，滤除了噪声，效果最好。改进算法能更好地响应图像亮度和对比度的变化，对环境的适应性更好。

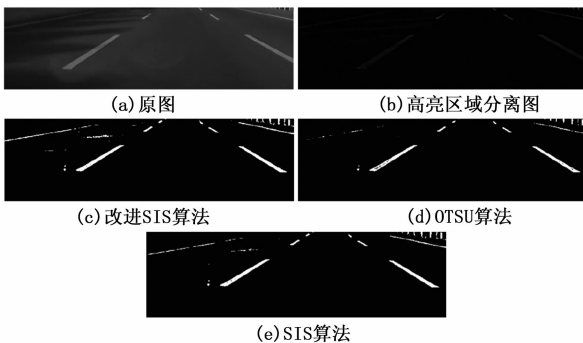


图 6 阴影遮挡情形下二值化对比

通过实验测试，SIS 算法处理一幅图片需要 2.45 ms，OTSU 算法需要 1.49 ms，本文提出的算法仅需 0.98 ms。

## 2 车道线模型的构建与简化

假设路面为平面，并且车道线都是平行的，那么根据

摄像机的位置和朝向，可以确定路面车道线和投影车道线的几何关系，这里采用如图 7 所示的道路模型，那么左右两条车道线可分别由式 (6)、(7) 表示：

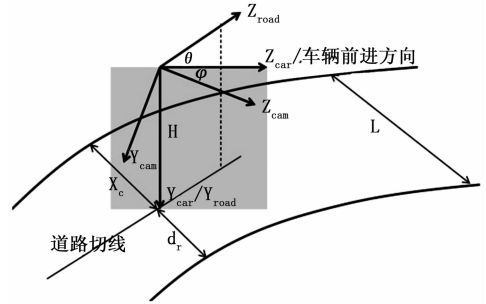


图 7 道路模型

$$u_l = E_u \left( \frac{\theta}{\cos\varphi} + \frac{\cos\varphi}{HE_v} x_c (v_l + E_v \tan\varphi) + \frac{E_v HC_0 / \cos^3\varphi}{4(v_l + E_v \tan\varphi)} \right) \quad (6)$$

$$u_r = E_u \left( \frac{\theta}{\cos\varphi} - \frac{\cos\varphi}{HE_v} d_r (v_r + E_v \tan\varphi) + \frac{E_v HC_0 / \cos^3\varphi}{4(v_r + E_v \tan\varphi)} \right) \quad (7)$$

为了更好地说明这个模型及其中的参数，首先明确 3 个坐标系：大地坐标系（路面坐标系）、汽车坐标系和摄像机坐标系。 $\theta$  和  $\varphi$  是摄像机对于大地坐标系的偏航角和俯仰角。 $(u, v)$  表示像素点相对于投影中心水平方向和垂直方向的位置， $E_u$  和  $E_v$  表示摄像机在像素单元上水平方向与垂直方向的焦距，这些参数（投影中心、 $E_u$  和  $E_v$ ）都可以通过摄像机的标定来获得。 $H$  是摄像机距离道路表面的距离。 $C_0$  是道路的横向曲率，如果  $C_0 = 0$  那么道路为直的，即式 (6)、(7) 表示直线。 $x_c$  和  $d_r$  分别表示左右车道线距车辆的距离， $L = x_c + d_r$  为车道宽度。

为了简化拟合过程，这里将车道线模型进一步简化为以  $A$ 、 $B$ 、 $C$ 、 $D$  为未知量的双曲线形式，即：

$$u = \frac{A}{v - D} + Bv + C \quad (8)$$

以矩阵形式表示为

$$P^T \cdot C_r \cdot P = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}^T \begin{bmatrix} 0 & -0.5 & 0.5D \\ -0.5 & B & 0.5E \\ 0.5D & 0.5E & F \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = 0 \quad (9)$$

其中： $E = C - BD$ ， $F = A - CD$ ， $C_r$  是对称的， $C_r \cdot P$  表示双曲线在  $P$  点的切线。

## 3 改进顺序 RANSAC 方法拟合车道线

要确定一个模型需要 4 个像素点，但多数情况下会存在超过 4 个候选像素，并且还可能存在一些异常像素，从而导致可能存在多模型问题，这里采用顺序随机抽样一致性<sup>[11]</sup> (Sequential RANSAC) 算法来解决此问题。

顺序 RANSAC 是将 RANSAC 算法运行多次，直到找到所有的模型或者执行一定次数的迭代。一旦确定一个模型后，它包含的支持数据点就会从数据集中清除，然后 RANSAC 再在数据集剩余数据点中确定其它模型，也就是

说如果前面确定的模型不准确, 会导致数据集中的数据点误消除, 这种情况在道路弯曲时尤为明显。

为了解决这一关键问题, 更准确地检测车道线, 本文提出了一种改进的顺序 RANSAC 算法, 加入了一种融合机制, 该算法流程如图 8 所示。

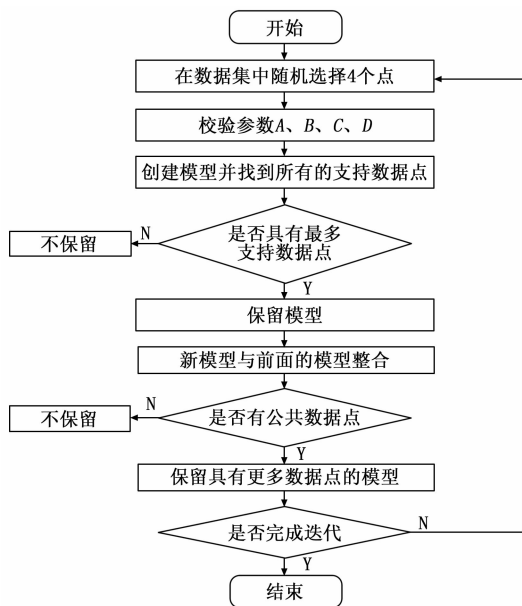


图 8 改进的顺序 RANSAC 算法流程图

该方法分为以下 6 个步骤:

1) 在像素集中随机选择 4 个像素数据点。

2) 创建模型并找到所有的支持数据点。尽管 4 个不在一条线上的任一点可以产生一个唯一的双曲线, 但并不是每一个结果都能正确地描述车道线, 所以这一步在寻找支持数据点之前, 需要对模型进行校验, 式 (8) 中 A、B、C、D 的显式解如式 (10) ~ (13) 所示:

$$A = \frac{E_c C_0 H E_u E_v}{4 \cos^3 \varphi} \quad (10)$$

$$B = \frac{x_c E_u \cos \varphi}{H E_v} \quad (11)$$

$$C = \frac{\theta E_u}{\cos \varphi} + \frac{x_c E_u \sin \varphi}{H} \quad (12)$$

$$D = -\tan \varphi E_v \quad (13)$$

首先校验 D, D 的值随  $\varphi$  的变化而变化,  $\varphi$  的变化是汽车悬架系统的压缩和伸长变化决定的。当车辆前悬架压缩, 后悬架伸长时, 摄像机向前下倾斜,  $\varphi$  可达到最大值; 反之车辆前悬架伸长, 后悬架压缩时, 摄像机前倾减少甚至会向上倾斜, 此时  $\varphi$  可达到最小值。通过实际车辆测试可知得  $\varphi$  的值浮动变化为  $\pm 3^\circ$ , 即  $\pm 0.035 \text{ rad}$ , 所以  $\varphi$  值范围为  $[\varphi_n - 0.035, \varphi_n + 0.035]$ 。D 确定后可以根据道路曲率校验 A, A 的大小取决于  $|C_0|$ , 一般车辆的最小转弯半径为 10 米, 那么这里  $|C_0|$  设置为  $0.1 \text{ m}^{-1}$ , 对于直线,  $A=0$ 。B 和 C 与车辆的行驶姿态有关, 由于车辆可以在道路的任何位置行驶, 所以对 B、C 不约束。如果没有进行 D 和 A 的校验就开始下一次迭代, 会出现错误的拟合结果。

3) 4 个随机点可以产生一个唯一的双曲线和六条直线, 共 7 个模型, 这 7 个模型中只有包含最多支持数据点的模型会被保留, 支持数据点由  $d_s$  确定, 如式 (14) 所示, 如果  $d_s$  小于一个阈值, 那么数据点将被判定为支持数据点。

$$d_s = \frac{(P^T \cdot C_r \cdot P)^2}{4((C_r \cdot P)_1^2 + (C_r \cdot P)_2^2)} \quad (14)$$

其中:  $C_r$  已由式 (9) 给出,  $(C_r \cdot P)_n$  表示向量的第 n 个元素。

4) 将新模型与之前确定的模型进行融合, 如果新模型与之前的模型中有公共数据点, 那么保留具有更多数据点的模型, 另一个模型移除释放。如果没有公共数据点, 则此模型被标记为一个新的模型。由于车道线不存在交叉点, 所以此方法在此适用。

5) 重复步骤 1) ~ 4), 进行一定次数的迭代, 直到没有数据点被消除, 从而不会影响后续的拟合结果。

6) 到本步骤为止, 已经得到了多个互不交叉的模型, 为了提高执行速度, 消除这些已经得到的模型的支持数据点, 然后在剩余像素数据集中重复步骤 1) ~ 4), 进行一定次数的迭代。拟合结果如图 9 所示。



图 9 拟合结果

## 4 最终模型和车道线的确定

### 4.1 车道线连续性

如图 10 所示, 以图中的标志为例, 车道线与直行标志最大区别是车道线由多条连续线段组成, 所以可以计算出箭头在感兴趣区域内连续出现的帧数 P。利用标志位置到车辆的距离、车辆的行驶速度和视频的帧率就可以计算出 P 值, 如式 (15) 所示:

$$P = \frac{SF}{v} \quad (15)$$

式中, S 为箭头、文字等被检测的位置到车辆的距离, F 为视频帧率, v 为车辆的行驶速度。



图 10 车道线与直行标志

在连续的视频帧中, P 值可以作为判断车道线是否存在的阈值。如果一条直线在 P 帧中连续出现, 那么就应该加上这条车道线, 反之如果在连续 P 帧中没有出现, 那么就去除这条车道线。

### 4.2 确定最终车道线模型

确定单个车道线模型后要要进行车道线模型的配对, 从而得到一条车道, 理想的配对结果取决于模型所具有的支持数据点数量以及模型拟合出的车道宽度是否符合正常的数值。

逆透视映射 (Inverse Perspective Mapping, IPM) 变换图像是将摄像机视图转换为从上到下的俯视视图, 如图 11

所示。假设地面为零，相机分辨率为  $m \times n$  像素，视场角为  $2\alpha \times 2\beta$  弧度， $\theta$  为摄像机俯仰角，相机到地面距离为  $H$ 。大地坐标系  $[u, v]$  到图像坐标系  $[x, y]$  转换关系如下<sup>[12]</sup>：

$$\begin{cases} u = \frac{H(m \tan \theta + 2x \tan \alpha)}{m - 2x \tan \theta \tan \alpha} \\ v = \frac{\sqrt{H^2 + u^2} y \tan \beta}{n} \end{cases} \quad (16)$$

图像中的点可通过这个方程得到大地坐标系中的对应点。



图 11 IPM 变换图

根据 IPM 变换图结合式 (6)、(7) 可以得到车道宽度  $L$ ，在现实情况中，机动车道的宽度在 2.45~3.75 m 之间，可以由此建立车道宽度约束。一般情况下，两条车道线独立拟合的结果不会完全遵循平行的关系，所以这里会得到一个非恒定的  $L$ ，为了获得更加准确的模型，需要对配对的两个模型进行微调，使其遵循平行关系。根据式 (6)、(7) 可知，如果一个模型的形式与式 (8) 相同，那么另一个模型就如式 (17) 所示：

$$u = \frac{A}{v - D} + B'v + C + D(B - B') \quad (17)$$

式 (8) 和 (17) 表明，要对这两个模型进行微调需要 5 个点来确定。但是这对车道线模型是由拟合步骤中的 8 个点（每条车道线模型各 4 个）拟合推导出得到的，这里在其中一个模型的 4 个点中选取 3 个点，另一个模型的 4 个点中选取 2 个点，总的来说有 48 种选取组合，这 48 个组合中选取规定范围内具有最多支持数据点的组合为最终模型。之后，利用式 (16) 将车道线的端点再转换到原图像中。

### 5 实验验证

程序基于 Visual Studio 2010 和 OpenCV2.4.10，硬件平台为 Intel (R) Core (TM) i5-7200CPU、GF920MX 显卡和 4GB 内存，本文的算法基于单目视觉，为了验证本文所提出的基于改进 SIS 算法和改进顺序 RANSAC 算法的车道线检测方法的检测性能，共采用 6 段视频数据集进行验证，其中四段视频是加州理工学院的车道数据集，另外两段为城市快速路、高速公路的视频数据，这 6 段视频分辨率为  $640 \times 480$ ，涵盖多种工况。其中加州理工学院的车道数据集与文献 [13] 作了对比，结果如表 1 所示。高速路的检测结果如表 2 所示。

其中，车道线检测的正确率就是指检测到的正确的车道线数目与实际车道线总数的比值，车道线的误检率是指

本身图像中不是车道线的信息被误检测成为了车道线，也

表 1 城市道路实验对比 %

算法	多车道	多车道	单车道	单车道
	正确率	误检率	正确率	误检率
提出的算法	91.58	7.32	96.24	2.32
文献[13]	91.03	8.08	95.06	5.79

表 2 高速公路测试结果

视频	帧数	多车道	多车道	单车道	单车道
		正确率/%	误检率/%	正确率/%	误检率/%
1	2400	97.45	0.68	98.23	0.22
2	3600	97.98	0.55	98.52	0.23

就是误检测的直线与实际车道线总数的比值。单车道只考虑车辆当前行驶的车道，多车道表示视车辆前方包括当前车道在内的多条车道。算法在多种工况下的拟合与检测效果见图 12，可以看到效果图清晰地展示了阴影、车辆干扰、车道标志干扰等多种工况下良好的检测效果。



图 12 各种工况下的车道线拟合

提出的算法识别准确率高，误检率低，对不同的道路环境有较强的适应性，在单车道即两边车道线的情况下有很高的鲁棒性。算法每帧耗时为 26.65 ms，实时性较强，能满足每秒 30 帧的摄像头检测。

### 6 结束语

本文针对车道线检测时会遇到的光照变化、路面状况不同以及阴影干扰等复杂环境，提出了改进的 SIS 阈值算法和改进顺序 RANSAC 算法相结合的方法来拟合与检测车道线。提出了改进的 SIS 阈值算法，相比其他的阈值算法不易受阴影车辆干扰的影响，抗干扰能力强，鲁棒性好，而且算法的计算复杂度低，时间消耗低。采用改进的顺序 RANSAC 算法检测车道线比采用传统的 RANSAC 的检测方法相比准确率有了一定的提升，误检率有所减少。实验结果表明，本文提出的算法在满足了实时性与鲁棒性要求的基础上，提高了检测的准确率，并且能适应多种工况以及复杂环境的干扰。