

# Research on the Lane Line Detection Algorithm Based on the Inverse Perspective Projection

Fu Hu<sup>1</sup>, Meisong Ye<sup>2</sup>, Shide Xiao<sup>2</sup>

<sup>1</sup>Guangzhou Dingxing Asset Management Co., Ltd., Guangzhou Guangdong

<sup>2</sup>College of the Mechanical Engineering, Southwest Jiaotong University, Chengdu Sichuan

Email: 1157212987@qq.com

Received: May 18<sup>th</sup>, 2020; accepted: Jun. 1<sup>st</sup>, 2020; published: Jun. 8<sup>th</sup>, 2020

---

## Abstract

Based on machine vision, the improved lane detection algorithm has been studied with the processing of the image, IPM view conversion, filtering, Hough linear detection, RANSAC and Bessel spline curve fitting and inverse perspective transformation. The test results show that this improved lane detection algorithm is stable and accurate, and can accurately detect the lane line, for vehicle occlusion, shadows of trees, uncleared lane, obviously curved lane and other typical road conditions, providing the feasibility of intelligent electric independent car driving in the park environment.

## Keywords

Intelligent Vehicle, Lane Detection, Algorithm, Machine Vision

---

# 基于逆透视投影的车道线识别算法研究

胡 拂<sup>1</sup>, 叶美松<sup>2</sup>, 肖世德<sup>2</sup>

<sup>1</sup>广州鼎兴资产管理有限公司, 广东 广州

<sup>2</sup>西南交通大学, 机械工程学院, 四川 成都

Email: 1157212987@qq.com

收稿日期: 2020年5月18日; 录用日期: 2020年6月1日; 发布日期: 2020年6月8日

---

## 摘要

本文立足机器视觉, 对智能电动车单目视觉的车道线图像检测和识别算法进行研究, 主要包括原始图像

获取、IPM逆透视图像的转换、滤波降噪(A)、霍夫直线检测、RANSAC和贝塞尔样条曲线拟合和反逆透视变换。实验测试结果表明,本文所改进的车道线检测和识别算法比较准确稳定,对于车辆的遮挡、树木的阴影、不清晰的车道线、明显弯曲的车道线等典型道路情景能够准确检测出车道线,为智能电动车在园区道路环境下自主行驶提供了可行性。

## 关键词

智能电动车, 车道线识别, 算法, 机器视觉

Copyright © 2020 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

车道线智能检测识别是智能电动车的一项关键技术,它通过图像传感器感知地面的车道线信息找到最适合行驶区域行进,减少了交通事故发生几率。随着计算机硬件系统对图像处理速度提升飞跃,基于机器视觉的车道线图像检测算法处理速度快,更容易满足实时性要求。本文主要研究道路车道线图像预处理和检测识别提取算法,实现对车道线的跟踪以及车道发生偏离时进行预警,为无人驾驶智能电动车实现自主导航增加技术可能性。

## 2. 车道线检测算法的研究现状

从上世纪90年代开始机器视觉检测车道线的技术研究就已经开始[1]-[7],对CCD摄像头拍摄道路上的车道线进行检测,车道线的纹理特征、颜色与那些色彩分明纹理特征比较明显的物体相比,前者属于比较简单的类型。因为其特征不够鲜明因此对车道线特征的提取和辨别带来了一定的难度。迄今为止车道线的检测算法主要分为三大类:基于特征、基于区域、基于模板。

### 1) 基于特征的车道线检测算法

该检测算法是根据车道线的颜色特征和它的边缘来定位图片中的车道线,根据车道线的特征颜色在HSV模型下进行约束,对满足约束条件的像素点进行保留。这种检测方法的前提是车道线的颜色变化小并且车道线的边缘部分要足够的清晰,而对于外界环境的光照或者阴影下颜色变化比较明显的时候这种算法就会失效[8]。因此这种检测算法只能作为一种辅助的手段,不能满足现实中车道线实时检测的要求。

### 2) 基于道路模型的车道线检测算法

道路模型一般可以分为:直线、线性双曲线、抛物线和B-样条曲线模型等。这类算法是通过用一些几何参数来表示这些道路模型[9]。直线模型表示的方法比如有霍夫变换求出直线的参数,Wang Lei在他的论文中提出了一种基于Hough变换和Kalman滤波的分段双曲线模型车道线检测方法,该算法是对所感兴趣的区域用Sobel算子对某个特定的方向进行边缘检测,对近视场区域采用Hough变换对检测的边缘部分进行直线检测,对远视场区域采用加权双曲线模型对车道线进行拟合,因为将车道线分为了多个段落所以在对与弯曲的车道线能够有比较准确的判断,弯道的方向确定之后可以通过改变双曲线模型的参数进行比较精确的曲线拟合,该算法的优点在于对弯道的检测也会有很好的检测效果。加州理工大学计算机视觉实验室的Mohamed Aly在其论文中使用B-样条曲线模型进行车道线的检测。他的做法是通过RANSAC算法拟合B-样条曲线获取车道线的参数,接着对拟合的车道线进行扩充和延伸等后处理过程,

实验检测效果表明该算法的检测的效果比较好，弊端就是在直线的倾角比较大时在所检测的效果就不是很理想，因此本文是在他的算法的基础上进行了一些调整和改变，对车道线有着比较稳定的检测效果。

### 3) 基于区域的车道线检测算法

该检测算法首先是需要确定车道线的大概位置，这与加州理工大学所提供的算法有一些类似，都是先进行直线检测，确定车道线在图片中的大概位置，然后对每一条车道线所确定的区域进行比较精确的检测。该算法的优点在于可以避开图片中其它的干扰因素，提高算法的实时性。

基于道路模型的车道线检测方法比较实用，噪声对该检测算法不会造成太大的干扰，检测效果稳定可靠，得到了比较广泛的应用，目前检测方式主要分为以下几类：

#### 1) 基于双摄像头的车道线检测[10]

双摄像头检测车道线的原理是通过双摄像头所拍摄的图片中车道线对应点的位置之差来获取车道线的三维几何信息。用双摄像头检测直线的优势在于能过获取图片中的大量信息，对车道线的定位精度高，检测的结果比较可靠，但是双摄像头的车道线检测方法增加了难度，相比之下，随着数字图像处理的能力的提高，单摄像头的检测效果已经能够满足实际的需求。

#### 2) 基于逆透视变换的车道线检测[11]

对视图直接进行边缘检测的，会有很多的干扰因素，比如图片中的树木、行驶的车辆、行人，天空等干扰因素多会对车道线的检测造成影响，但是经过逆透视变换之后的视图就不存在这一方面的问题，逆透视变换之后的图像只是对原视图的所感兴趣的区域进行检测减少了不必要的处理耗时，提高了算法的实时性，因此被广泛的应用到单目视觉之中，但是进行此类车道线检测比如保证相机的内部参数不会发生变化并且摄像头的与地面的倾斜角度是固定的，由于无人驾驶车的摄像头基本上是在车顶上保持不动的，所以基于逆透视变换车道线检测算法是可行的。

#### 3) 基于 Gabor 滤波器的车道线检测[12]

该检测算法是根据车道线的纹理信息确定它的消失点的位置，然后以该点作为起始点的位置，以这个起始点辐射状的形式画线，根据辐射的线段上的每一个像素点计算它们的梯度方向的一致性，获取的最大的值即为第一主边界，根据所确定的第一边界的基础上求出第二边界，以此类推可以求出其它的边界，但是该算法的实时性比较差，因此对于不适于本文的检测算法。

#### 4) 基于随机抽样一致性的车道线检测[13] [14] [15]

此方法进行车道线检测是在逆透视变换之后的视图图中进行的，根据检测的直线获取感兴趣区域，该区域视为一组待观察的数据，这些数据中往往会有比较大的噪声或者是一些无效的点，该算法的检测原理是随机的选取待观察数据的一些点，通过这些点进行样条曲线模型拟合，反复的随机选取局内点和样条曲线拟合，找出多次选取中权值最大的拟合样条作为车道线的模型参数，该算法的优点在于即使在复杂的环境里也有很好的拟合效果。

## 3. 图像采集系统及摄像机标定

### 3.1. 图像采集系统构成

由于前视系统视角大，图像信息量广，并且易于安装，本文研究采用前视系统，将 USB 高清摄像头尽可能地安装在智能电动车较高位置，扩大视野范围。为了更好的查看地面情况，单目摄像机工作时一般都会会有一个下倾角度，不同的角度摄像机所观察的视野有很大的变化，本文中所学的 USB 高清摄像头是可调的，调动的范围是 0~45 度，而在电动车行驶的过程中其下倾的角度是固定的，这样就可以稳定的进行视图转换。考虑到车辆在行驶的过程中是否会偏离原先的车道，因此在行驶的过程需要判断车身相对于车道线的位置，因此本文中的摄像头安装是在智能电动车正中的位置上。



Figure 1. Image acquisition cameras  
图 1. 图像采集摄像头

USB 高清摄像头信息的是靠 4G 内存 i5 处理器的笔记本电脑所采集的, 软件开发环境为 Microsoft Visual C++2010, 开发语言为 C/C++ (OPENCV), 图像传感器的实物图如图 1 所示, 它的相关参数如表 1 所示。

Table 1. Correlation parameter of image acquisition sensor  
表 1. 图像采集传感器相关参数

参数		取值或可变范围
摄像机参数	摄像机个数	1
	分辨率	640*480
	帧率	7.5 fps
安装参数	俯仰角 $\alpha$	0~45
	航向角 $\beta$	0
	摄像头距地面高度	1200 mm

### 3.2. 基于 OPENCV 摄像机标定

相机拍摄图片的过程可以看作为一个三维场景转换为二维场景的过程。常用线性摄像机成像模型进行描述。图 2 描述了线性摄像机模型对空间里的一点的透视投影成像过程。假设  $(x, y)$  为  $p$  点的图像坐标,  $(X_c, Y_c, Z_c)$  为空间点  $p$  在摄像机坐标系下的坐标。

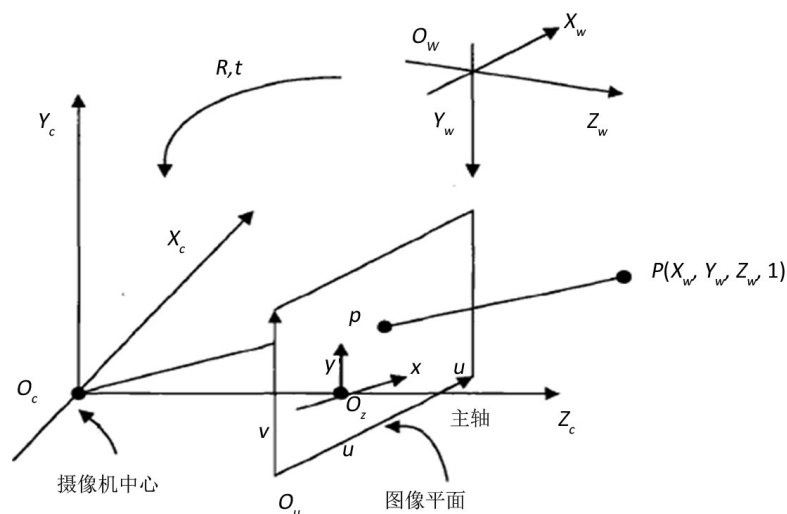


Figure 2. Linear camera model  
图 2. 线性摄像机模型

$p$  点的世界坐标与其在图像坐标系上的投影点  $p$  的关系:

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{d_x} & 0 & u_0 \\ 0 & \frac{1}{d_y} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} R & t \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} a_x & 0 & u_0 & 0 \\ 0 & a_y & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} R & t \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = M_1 M_2 X_w = M X_w$$

其中  $a_x = f/d_x$ ,  $a_y = f/d_y$ ,  $R$  是一个  $3 \times 3$  的正交矩阵,  $t$  是一个三位平移向量,  $M$  为一个  $3 \times 4$  矩阵, 称为投影矩阵,  $a_x, a_y, u_0, v_0$  构成了  $M_1$ , 因为  $a_x, a_y, u_0, v_0$  取决于内部结构所以被称之为摄像机的内参数;  $M_2$  摄像机相对于世界坐标系的位置信息决定了  $M_2$  矩阵, 所以称  $M_2$  为摄像机的外参数。求解这些几何模型的参数的过程称之为相机的标定过程。求出的相机内参数可以有效的消除畸变。而不同的镜头类型求出的畸变参数是不同的。内标定计算 4 个内部线性参数 ( $f_x, f_y, u_x, u_y$ ), 和 5 个畸变参数 ( $k_1, k_2, k_3, p_1, p_2$ ), 它们合称为摄像机的内部参数。外标定则是计算从三维世界坐标系到三维摄像机坐标系的变换关系, 包括旋转矩阵  $R$  和平移向量  $t$ , 它们合称为外部参数。

采用棋盘格模板对摄像机进行标定。本文从用摄像机拍摄的图片中选取了 12 张不同角度的棋盘格图片。棋盘格大小为  $9 \times 6$ , 利用 *OpenCV* 的库函数对它进行标定。具体实现过程如下:

- 1) 根据实际需要, 读取摄像机拍摄的棋盘图。选取的棋盘图如图 3 所示。
- 2) 调用 `findChessBoardCorners()` 函数提取每一幅棋盘格图片的角点坐标值, 角点坐标提取的精度是关键, 角点精度越高其标定的结果越准确。
- 3) 调用函数 `findConnersPix()` 能够很精确的求出角点的位置坐标, 可以达到亚像素级别。如图 4 所示。
- 4) 调用 `DrawChessboardComers()` 函数把检测到的所有的角点信息标记在棋盘上。检测的所有图片的角点位置如图 5 所示。
- 5) 得到多个视标图像的角点, 通过对函数 `calibrateCamera()` 的调用可以到的到相机的内外部参数如图 6 所示。



Figure 3. Selected twelve checkerboard graph  
图 3. 选取的 12 张棋盘格图片





- 6) 调用 `undistort()` 函数得出矫正后的图像。
- 7) 以 XML 文本格式保存标定的参数。

#### 4. 逆透视映射(IPM)

逆透视变换是透视变换的逆过程，根据上面标定过程求取的相机内外参数，将图像从图像坐标系映射到世界坐标系的过程就是逆透视变换的过程。通过逆投影变换得到的 IPM 图像，有两个好处：

- 1) 有效消除非目标因素对车道线检测的干扰

直接对原视图进行车道线检测会增加大量的非目标因素，比如道路上的树木、远处的天空、行驶中的车辆与行人、地面上的一些交通标识都会对车道线的提取造成了干扰。在 IPM 视图上进行选定道路区域的边缘提取就几乎不存在这类问题，我们只对车辆行驶过程中的感兴趣区域进行检测，减少了不必要的处理耗时，提高了算法的实时性，图 7 中左边为原始图片，红色方框为感兴趣的区域，右边图片为感兴趣区域得到的俯视图图像。



Figure 7. Schematic diagram of inverse perspective projection (Image Credit: Cal tech lane data set)

图 7. 逆透视映射示意图(图片来源：加州理工车道数据集)

- 2) 原本道路上平行的两条车道线在经过相机的透视变换之后变为了两条相交的线段，通过逆透视变换之后的图像可以消除透视变换带来的干扰，可以很直观有效的判断车道线在世界坐标系的位置，更加方便的求解出智能电动车与车道线的相对位置。

用 `OpenCV` 的库函数实现逆透视变换的过程。具体实现过程如下：

- 1) 从标定时保存的 XML 文件中提出相机内外参数。
- 2) 读取摄像机拍摄的棋盘图，如图 8(左)所示。
- 3) 调用 `initUndistortRectifyMap()` 函数，输入标定时得到的内外参数求出新的摄像机参数矩阵，以及  $xy$  坐标的重映射参数。
- 4) 根据  $xy$  坐标的重映射参数调用 `remap()` 函数对图片进行校正。
- 5) 调用 `findChessBoardCorners()` 和 `FindComerSubPix()` 函数求出棋盘格图片的角点坐标。如图 8(右)所示。
- 6) 确定四个角点位置的坐标，四个角点为所求的角点的顶点位置，图 8(右)中四个顶点的位置。物理坐标系的坐标为  $(0,0)(8,0)(0,5)(8,5)$ ，根据这四组坐标值调用 `getPerspectiveTransform()` 函数求出图像透视变换的矩阵  $H$ 。

- 7) 根据求出的矩阵  $H$ ，调用 `warpPerspective()` 函数即可求出逆透视变换之后的图像。对于所求的矩阵  $H$ ， $H.at(2,2) = k$ ， $k$  的值可以控制俯视图图片的大小。对于不同类型的摄像机所取的  $k$  值是不同的，根据所需求的俯视图的大小来控制  $k$  的值，对于同一类型的摄像机在一个固定的倾角拍摄图片时， $k$  值确

定之后对于以后检测的俯视图的图片尺寸是一样的。不同的  $k$  值对应的俯视图的如图 9 所示。



Figure 8. Read original diagram

图 8. 读取的原始图

8) 以 XML 文本格式保存逆透视变换之后的矩阵以及它的反逆透视变换矩阵  $H.inv()$ , 得到它的单应矩阵和高度参数后, 就能不用棋盘的情况下也能得到它的鸟瞰图图像。

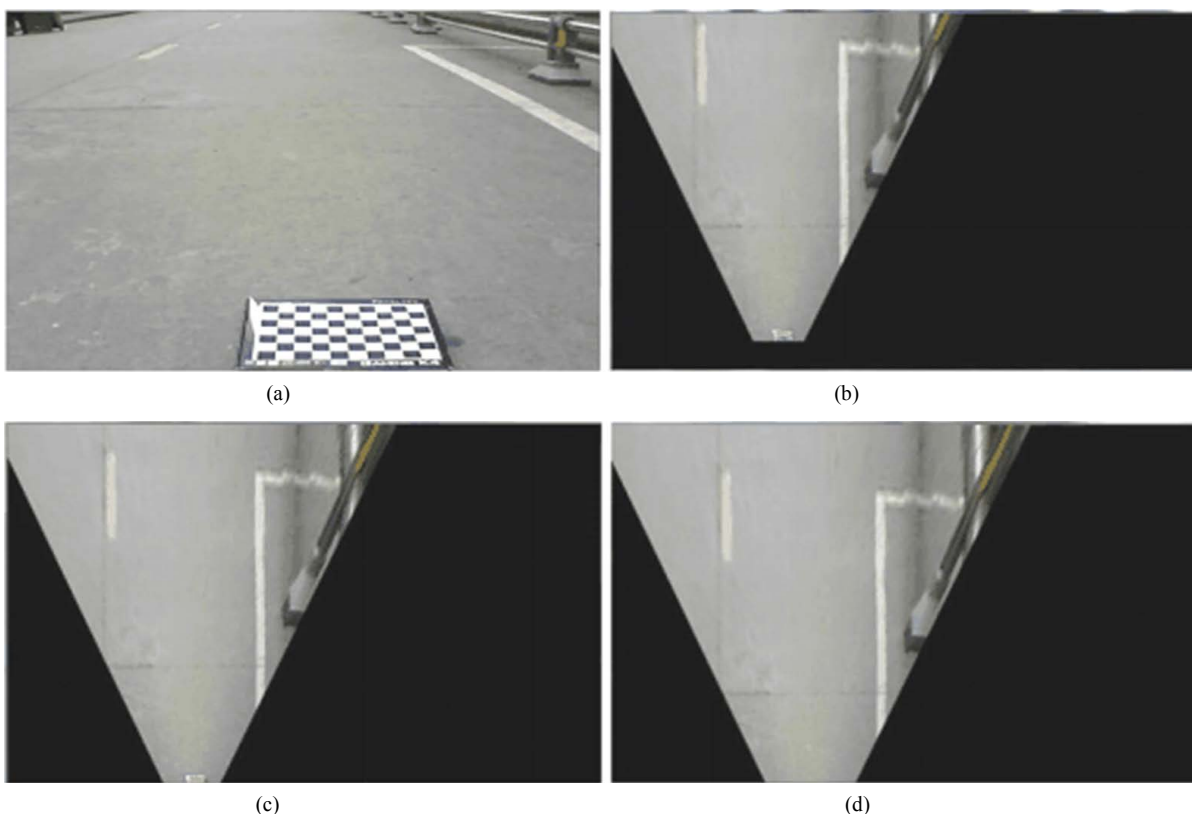


Figure 9. Inverse perspective transformation. (a) Original image; (b) Image after the inverse perspective transformation when  $H = 11$ ; (c) Image after the inverse perspective transformation when  $H = 12$ ; (d) Image after the inverse perspective transformation when  $H = 13$

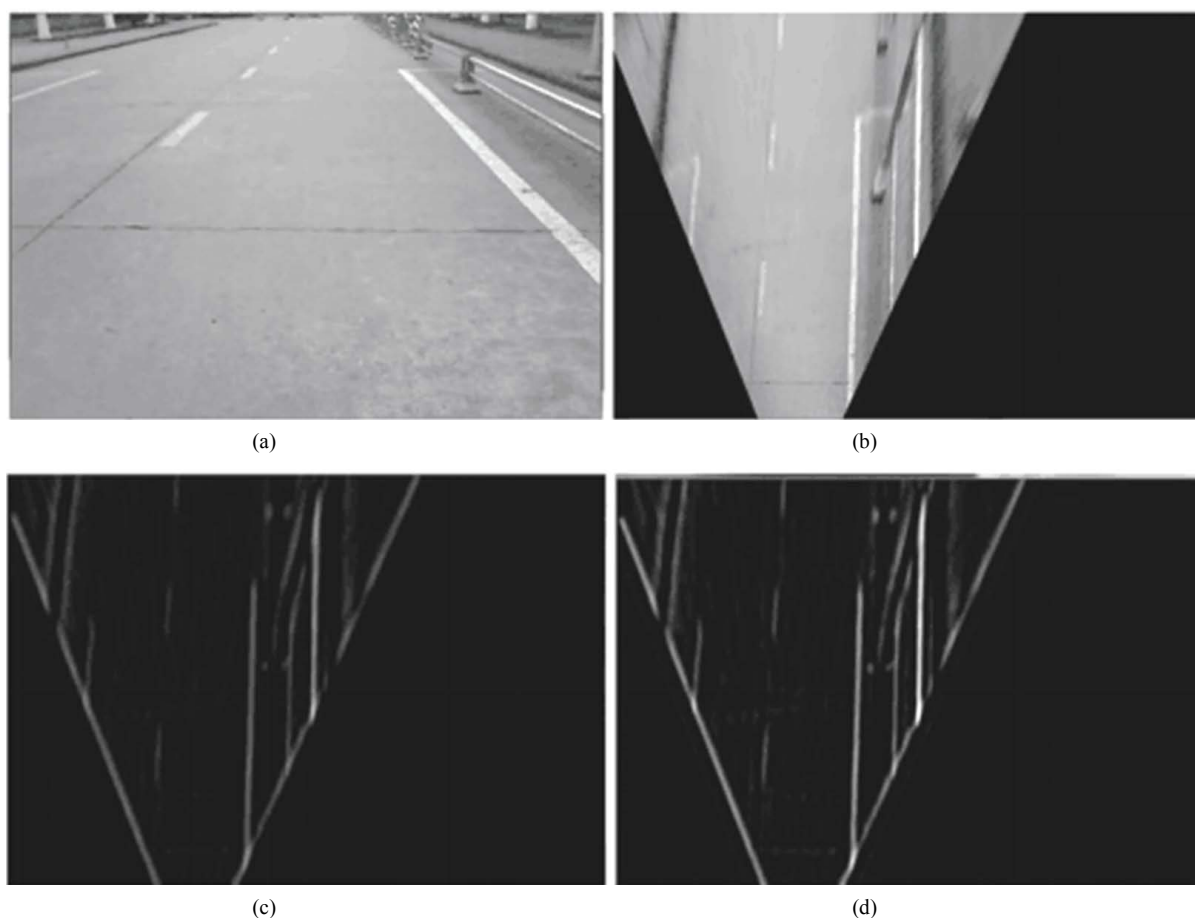
图 9. 逆透视变换。(a) 原始图像; (b)  $H = 11$  的逆透视变换后的图像; (c)  $H = 12$  的逆透视变换后的图像; (d)  $H = 13$  的逆透视变换后的图像



## 5. IPM 图像数据预处理

获取 IPM 图像后,需要对数据进行预处理。首先是数据归一化,把 IPM 图像中的每个像素缩放到[0, 1]之中,并且将 IPM 图像中的每个像素值减去所有像素值的平均值,相当于移去了图像的平均亮度值,对于逆透视映射(IPM)后的图像我们感兴趣的地方是车道线的位置,而不是它的光照度,因此这样的预处理是比较合理的。然后进行滤波处理,在不破坏图像细节的基础上对噪声进行消除或者抑制。常见三种线性滤波为方框滤波、均值滤波、高斯滤波,非线性滤波为中值滤波和双边滤波。本文采用高斯滤波。高斯滤波就是对整幅图像进行加权平均的过程,每一个像素点的值,都由其本身和邻域内的其他像素值经过加权平均后得到。高斯滤波的具体操作是:用一个模板(或称卷积、掩模)扫描图像中的每一个像素,用模板确定的邻域内像素的加权平均灰度值去替代模板中心像素点的值。对于抑制服从正态分布的噪声非常有效。

为减少各种噪声对特征检测造成干扰,比如行人、车辆、树荫、等干扰因素,在特征提取之前首先的处理工作是对 IPM 图像进行高斯滤波,这个处理工作的目的是在竖直的方向上检测直线,所以在 y 方向我们对其做一维高斯平滑滤波,经过实验比选,本文中所取纵向滤波尺度  $\sigma_y$  的值为 5,横向滤波尺度  $\sigma_x$  的值为 7,能够有效的减少很多的噪声干扰。如图 10 所示,可以看出  $\sigma_x$  的滤波尺度为 7 时明显比滤波尺度为 5 时的噪声要小。

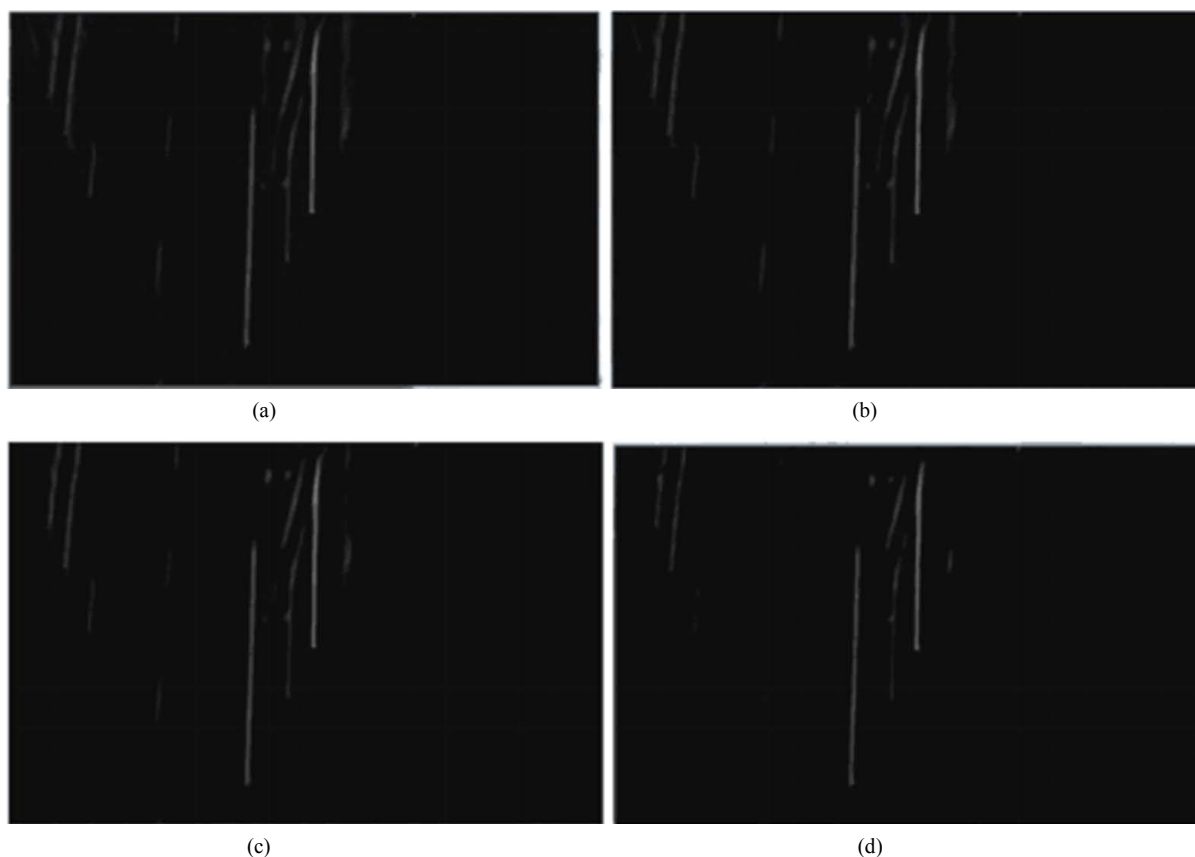


**Figure10.** Gauss filtering. (a) Original image; (b) Image after the inverse perspective transformation; (c) Filtered image when  $\sigma_x = 7, \sigma_y = 5$ ; (d) Filtered image when  $\sigma_x = 5, \sigma_y = 5$

**图 10.** 高斯滤波。(a) 原始图像; (b) 逆透视变换后的图像; (c)  $\sigma_x = 7, \sigma_y = 5$  时滤波后的图像; (d)  $\sigma_x = 5, \sigma_y = 5$  时滤波后的图像

由于逆透视变换形成的左右边界经过高斯滤波后并不能有效的剔除掉，而这部分信息是原视图中是不存在的。本文研究在边界没有干扰因素时求出 IPM 图像中的左右边界线的坐标值，根据这两条直线将这两条直线划分为多个掩码区域进行边界的剔除，对每一帧图片都进行处理。

滤波处理后阈值化，通过从滤波后的图片中选择  $q\%$  的分位点的值来实现的，对于低于这个分位点的值全部变为 0，可以有效的剔除图片中的噪声，本文中  $q$  的取值为 96.5，其不同的  $q$  值的处理效果图如图 11 所示。



**Figure 11.** IPM image after threshold conversion. (a) Percentile point  $q = 95.5$ ; (b) Percentile point  $q = 96.5$ ; (c) Percentile point  $q = 97.5$ ; (d) Percentile point  $q = 98.5$

**图 11.** 阈值化后的 IPM 图像。(a) 分位点  $q = 95.5$ ; (b) 分位点  $q = 96.5$ ; (c) 分位点  $q = 97.5$ ; (d) 分位点  $q = 98.5$

阈值化并不是二值化，对于图片的值并不是单纯的 0 和 1，保留了图片的原有像素值，今后样条曲线拟合会用到它的像素值。

## 6. 总结

本文叙述车道线检测图像获取和预处理全部过程算法，包括摄像头的安装和标定，相机的内外参数获取，对标定后的图像进行逆透视变换，得到 IPM 图，对 IPM 图进行高斯滤波，分析了滤波尺度的大小对噪声的影响。对 IPM 图滤波后形成的边界线进行处理，进而对边界线处理后残留的毛刺信号进行腐蚀处理。最后介绍了图像的分位点的大小对阈值化后图像车道线的影响。

## 参考文献

- [1] 鞠乾翊. 基于机器视觉的快速车道线辨识研究[D]: [硕士学位论文]. 上海: 上海交通大学, 2013.

- 
- [2] 许茂鹏. 车道线检测系统的研究与实现[D]: [硕士学位论文]. 成都: 电子科技大学, 2015.
- [3] Goldbeck, J., Huertgen, B., Ernst, S., *et al.* (2000) Lane Following Combining Vision and DGPS. *Image and Vision Computing*, **18**, 425-433. [https://doi.org/10.1016/S0262-8856\(99\)00037-2](https://doi.org/10.1016/S0262-8856(99)00037-2)
- [4] Jung, C.R. and Kelber, C.R. (2005) Lane Following and Lane Departure Using a Linear-Parabolic Model. *Image and Vision Computing*, **23**, 1192-1202. <https://doi.org/10.1016/j.imavis.2005.07.018>
- [5] Bertozzi, M. and Broggi, A. (1998) GOLD: A Parallel Real-Time Stereo Vision System for Generic Obstacle and Lane Detection. *IEEE Transactions on Image Processing*, **7**, 62-81. <https://doi.org/10.1109/83.650851>
- [6] Lookingbill, A., Lieb, D. and Thrun, S. (2007) Optical Flow Approaches for Self-Supervised Learning in Autonomous Mobile Robot Navigation. In: *Autonomous Navigation in Dynamic Environments*, Springer, New York, 29-44.
- [7] Wang, Y., Teoh, E.K. and Shen, D. (2004) Lane Detection and Tracking Using B-Snake. *Image and Vision Computing*, **22**, 269-280. <https://doi.org/10.1016/j.imavis.2003.10.003>
- [8] Chiu, K.-Y. and Lin, S.-F. (2005) Lane Detection Using Color-Based Segmentation. *IEEE Proceedings of Intelligent Vehicles Symposium*, Las Vegas, 6-8 June 2005, 706-711.
- [9] 陈胜勇. 基于 OpenCV 的计算机视觉技术实现[M]. 北京: 科学出版社, 2008.
- [10] Bradski, G., Kaehler, A. 学习 OpenCV: 中文版[M]. 北京: 清华大学出版社, 2009.
- [11] 刘瑞祯, 于仕琪. OpenCV 教程[M]. 北京: 北京航空航天大学出版社, 2007.
- [12] 牛发发, 陈莉, 张永新, 等. 基于鲁棒主成分分析的 Canny 边缘检测算法[J]. *计算机应用*, 2014, 34(6): 1727-1730.
- [13] 赵军. 视觉显著性直线的检测算法研究[D]: [硕士学位论文]. 武汉: 华中师范大学, 2015.
- [14] 化雅静. 智能车辆辅助驾驶系统中车道偏离预警算法的研究[D]: [硕士学位论文]. 长沙: 湖南大学, 2015.
- [15] 彭湃. 自主车辆道路线检测与偏离预警方法研究[D]: [硕士学位论文]. 长沙: 湖南大学, 2015.