**中文图书分类号：TP\*\*\***

**密 级：公开**

**UDC：\*\*\*\*\***

**学 校 代 码：10005**



**工程硕士学位论文**

**M.E. DISSERTATION**

**论 文 题 目：基于Hough变换和神经网络的智能车辆车道线识别**

**论 文 作 者：王嘉雯**

**领 域：软件工程**

**指 导 教 师：严海蓉**

**论文 提交 日期: 2018年\*月**

**独 创 性 声 明**

本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京工业大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签 名：

日 期：2017年\*月\*日

**关于论文使用授权的说明**

本人完全了解北京工业大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅和借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

（保密的论文在解密后应遵守此规定）

签 名： 日 期：2017年\*月\*日

导师签名： 日 期：2017年\*月\*日

摘 要

随着人工智能（Artificial Intelligent）越来越受大众关注，将机器智能化已成为最终目标。其中，智能车辆（Intelligent vehicle）在近年来成为各国在智能交通中潜心研究的一个重要领域，其中，基于视觉的结构化道路环境识别研究成为研究的重点。结构化道路环境的检测与识别的一个主要方面就是对路面中的车道线进行检测与识别，确定车道的边界与类型，为智能车辆提供保持车道行驶和变换车道行驶必须的车道信息，使智能车辆能实现平稳、安全的自动驾驶。

无论是通过红外识别，图像识别还是最终的机器学习，智能车辆的首要关键都是为了让车辆能够准确的识别出车道线，只有准确识别出车道线，车辆才能根据车道线信息来进行下一步控制的判断。本课题以智能车辆中的车道线识别和智能车辆的自动驾驶为背景，主要针对卷积神经网络在车道线识别中的应用进行设计，实现智能车辆稳定实时的对道路结构进行识别为目标，提出了一种基于Hough变换和卷积神经网络的道路识别方法。

本课题设计的道路识别方法主要分为图像预处理，图像分类，车辆控制。因道路环境易受天气，光照等影响，所以提取的道路的图像会由于天气光照的不同而产生一定的噪声，为了让卷积神经网络能更好的识别到图像中的车道线，本课题首先通过图像的预处理将图像进行去噪，直线的提取。然后，针对处理后的图像，构建出卷积神经网络对原始图像进行训练，达到机器能正确将车道线提取并且进行正确的分类。最后，结合分类的结果和强化学习来进行车辆的控制。

………

关键词：机器视觉；图像处理；机器学习；卷积神经网络；强化学习；

Abstract

目录

[摘 要 I](#_Toc490218664)

[第1章 绪 论 3](#_Toc490218665)

[1.1 课题研究背景及意义 3](#_Toc490218666)

[1.2 车道线检测国内外研究现状 2](#_Toc490218667)

[1.2.1 车道线检测国内研究现状 3](#_Toc490218668)

[1.2.2 车道线检测国外研究现状 3](#_Toc490218669)

[1.3 论文研究内容及论文结构 4](#_Toc490218670)

[第2章 相关技术介绍 6](#_Toc490218671)

[2.1 图像预处理算法概述 6](#_Toc490218672)

[2.1.1 Canny边缘检测 6](#_Toc490218673)

[2.1.2 Hough Transform 直线提取 6](#_Toc490218674)

[2.2 神经网络概述 6](#_Toc490218675)

[2.2.1 普通人工神经网络概述 6](#_Toc490218676)

[2.2.2 卷积人工神经网络概述 6](#_Toc490218677)

[2.3 强化学习概述 6](#_Toc490218678)

[2.3.1 马可夫决策过程 6](#_Toc490218679)

[2.4 本章小结 6](#_Toc490218680)

[第3章 车道线检测需求分析 7](#_Toc490218681)

[3.1 车道线检测系统需求分析 7](#_Toc490218682)

[3.2 功能需求 7](#_Toc490218683)

[3.2.1 车道线标注 7](#_Toc490218684)

[3.2.2 图像预处理 7](#_Toc490218685)

[3.2.3 车道线检测 7](#_Toc490218686)

[3.2.4 结果显示 7](#_Toc490218687)

[3.2.5 车辆控制 7](#_Toc490218688)

[3.3 非功能需求 7](#_Toc490218689)

[3.3.1 性能需求 7](#_Toc490218690)

[3.3.2 鲁棒性 7](#_Toc490218691)

[3.4 本章小结 7](#_Toc490218692)

[第4章 车道线检测算法设计 8](#_Toc490218693)

[4.1 车道线图像数据来源与筛选 8](#_Toc490218694)

[4.2 车道线图像预处理 8](#_Toc490218695)

[4.2.1 ROI提取范围设计 8](#_Toc490218696)

[4.2.2 边缘提取设计 8](#_Toc490218697)

[4.2.3 直线提取设计 8](#_Toc490218698)

[4.3 基于CNN的车道线检测算法 8](#_Toc490218699)

[4.3.1 CNN结构及参数设计 8](#_Toc490218700)

[4.3.2 CNN的训练及测试 8](#_Toc490218701)

[4.4 基于强化学习的车辆控制算法设计 8](#_Toc490218702)

[4.4.1 Q学习方法运用设计 8](#_Toc490218703)

[4.5 本章小结 8](#_Toc490218704)

[第5章 车道线检测算法实现 9](#_Toc490218705)

[5.1 车道线图像数据预处理实现 9](#_Toc490218706)

[5.1.1 ROI区提取实现 9](#_Toc490218707)

[5.1.2 边缘检测实现 9](#_Toc490218708)

[5.1.3 直线提取实现 9](#_Toc490218709)

[5.2 基于CNN的车道线检测算法实现 9](#_Toc490218710)

[5.2.1 CNN整体架构实现 9](#_Toc490218711)

[5.2.2 CNN训练及测试 9](#_Toc490218712)

[5.3 基于强化学习的车辆控制算法实现 9](#_Toc490218713)

[5.3.1 Q学习方法实现 9](#_Toc490218714)

[5.4 本章小结 9](#_Toc490218715)

[第6章 车道线检测方法实现及测试 10](#_Toc490218716)

[6.1 系统实现测试环境 10](#_Toc490218717)

[6.2 测试实例 10](#_Toc490218718)

[6.3 测试结果 10](#_Toc490218719)

[6.4 本章小结 10](#_Toc490218720)

[结 论 11](#_Toc490218721)

[参 考 文 献 67](#_Toc490218722)

绪 论

课题研究背景及意义

近年来，由于硬件技术和软件技术的发展，人工智能得到了越来越多的关注。随着各种高级算法不断的提高，人工智能逐渐由单一的模式识别发展为机器学习再到深度学习。其中，无人驾驶近年来成为人工智能中的一个关注热点。无人驾驶主要基于智能车辆，让车辆通过机器的识别和控制，自主的实现安全准确的驾驶。

据统计，大多数车辆事故都由人为造成，而不是机器故障。大多数车辆事故原因都是疲劳驾驶，酒驾以及判断失误和来不及反应。根据交通部门的调查来看，在事故发生之前，只有在0.5s内做出相应的应对措施才能避免交通事故的发生。对于人来说，0.5s的时间往往不够，但是，对于按毫秒甚至微秒来进行计算的机器来说，0.5s的时间相当充足了。而智能车辆中的无人驾驶的出现就是为了解决这个问题。如果能让机器自主操作，而不加与人工干涉，交通事故的发生率会减小。除了在交通方面，在工业生产中，运送和放置材料的工作往往会交给机器来完成。但是在以往的工业生产中，运送和放置材料使用的工具大致为两种：一是传送带，二是由磁感线引导的智能小车。第一种情况，传送带可以保证运送的准确度，但是却不能保证转送的灵活度，而且传送带在一定时间内就需要保养和上油等工作，这样对于生产商来说既降低了灵活度，又增大了生产开销。而第二种由磁感线引导的小车，受环境的影响较大，并且短时间内磁感线需要不时的保养和修复，对于生产商来说又是一项不必要的开销。而采用无人驾驶的智能小车进行物料的运输，既能增加其灵活度，又能减少不必要的周围设备维护的开销。因此，无论从交通上还是工业生产上来看，智能车辆的无人驾驶研究实现都具有一定的现实意义。

智能车辆是一个复杂的系统，由感知模块，路径规划模块，机械控制模块等构成。其中，感知模块用于对汽车行驶中周围环境的感知，感知周围是否有障碍物，是否可以行驶等。目前感知模块主要基于两种：一种为传感器感知，一种为视觉感知。传感器感知主要为雷达，激光，红外等，主要用于障碍物的检测。而第二种视觉感知，一般都使用工业级的高清摄像头，双目摄像头结合图像处理来进行环境实景的感知。车道线属于交通标志，规定了车辆的行驶规则，一般都是直接融于环境实景中了，所以为了实现智能小车能按照交通规矩行驶，能有效的通过视觉感知模块将车道线提取出来是一个关键。

车道线检测国内外研究现状

为了能有效并且准确提取出车道线，国内外的研究人员已经提出了各式各样的技术解决方案。由最初的运用传感器，如雷达，红外进行检测提取，到由视觉感知系统与图像处理技术来进行车道线提取，再到由视觉感知系统与机器学习结合进行车道线提取，到目前由视觉感知系统与深度学习结合进行车道线提取。

### 车道线检测国内研究现状

论文[1]提出了一种基于三维道路模型的车道线检测算法，根据车道线颜色突变，检测车道线的边界，并使用卡尔曼滤波，实现车道线的跟踪。方法鲁棒性强，在路况复杂车辆较多时仍能取得优质的检测效果，但是由于算法的复杂性，算法比较耗时。

论文[2]为了避免车道线参数估计以及最终的车道线参数方程拟合问题，提出了一种基于视觉和车辆定位系统结合的车道线检测算法，该方法先通过视觉提取出车辆近端部分的特征点，然后利用提取出的特征点更新定位系统的采样参数，实现了一种利用参数追踪的车道线检测方案。该方法的检测效率很高，但很容易受到噪声的干扰。

论文[3]在视觉的基础上提出了一种基于贝叶斯关注的机制，利用该机制对图像数据进行自顶向下以及自底向上的扫描计算，实现了一种高效的车道线检测算法。但这种方法是建立于道路线完整突出的假设之上，因此对于一些有破损的道路效果欠佳。

论文[4] 利用模型匹配方法，对车辆行驶前方的两条主要车道线进行检测，并确定其位置及曲率，该方法首先需使用逆透视变换(IPM)，将行车图像中的正视图投影到俯视图中，去除近大远小的透视效果，使得车道线看上去较为平行。逆透视变换可以将行车图像中的透视效果消除，将具有近大远小特点的行车正视图转换成俯视图效果，转换的矩阵一般可以通过照相机内参和外参标定计算获得，一般对比较平的路面有效，因为这样行车图像逆透视变换之后图中车道线是平行状态，但若道路有一定的坡度，则逆透视变换后车道线会有一定交汇，这对后面的寻找同一条车道线像素点有一定不利。因此，该方法只对平坦道路有效，具有一定局限性。

### 车道线检测国外研究现状

论文[5]提出了一种实时性和鲁棒的方法来检测城市道路车道线标志。该方法首先使用逆透视映射，以避免道路图像行车产生的近大远小效果，之后使用高斯卷积滤波器，对逆透视变换得到的俯视图滤波。该滤波器是针对黑暗背景中的亮线以及车道线宽度进行了专门的调整。因此其能很好的提取出车道线保留图像中超过阈值的部分并去掉噪声。然后，使用简化的 Hough 变换，对滤波之后的结果进行直线检测，随后利用 RANSAC 进行样条曲线拟合。最后，利用原始图像对车道线进行定位。该算法没有使用跟踪。

论文[6] 提出了一种使用改进版随机霍夫变换对车道线检测的方案，计算效率得到优化，内存开销也得到减少，效率比传统霍夫变换高。然而，这种方法强烈依赖于车道线点的特征，且路面必须干净，当车辆较多或其他障碍物较多时，该方法容易失效。

论文[7] 提出了一种将几何投影与 Adaboost 算法结合以寻找可行驶区域的方法，该方法需要大量的不同道路区域作为训练集以训练道路区域分类器。

论文[8] 提出了一种基于 RANSAC（Random Sample Consensus）的车道线检测算法。该方法将原始输入图像转化为灰度图，增强灰度图的对比度，获取图像二值图，利用滤波器对图像本身平滑处理，通过 Canny 算子找出车道标志线的边缘，最后利用 RANSAC 算法识别出图像中的车道线。该方法简单高效，但只适合于一些简单的场景，对于复杂度较高的道路面，该方法的识别效果并不好。

论文[9] 开发了一种基于三种特征车道检测算法。所使用的特征分别是起始位置，方向和亮度值。最初，使用 Sobel 算子获取图像的边缘信息。车道边界被表示为包含三个的载体特征。通过输入图像和先前车道上模型计算出当前车道线向量。分别对左边界和右边界使用两个窗口。假设每行包含 N 个像素，就会生成 N 条车道线向量作为候选。利用先前的车道线向量并通过加权距离矩阵计算得到的最小距离作为最佳候选。 为了均衡每个特征被分配一个不同的权重。利用候选预测新车道向量。如果道路的宽度突然改变，当前车道线向量的计算结果将舍弃，并利用先前向量作为当前结果。

论文研究内容及论文结构

智能车辆驾驶准确性和安全性的保障完全依赖于车辆对道路的识别和认知，只有当车辆能正确识别实时的道路状况，才能及时做出相应的反应以应对多变的道路状况。针对如何提高机器视觉对图像的处理和识别，本课题需要对一下内容展开研究：

1. 图像预处理算法设计

由于道路背景的复杂性，除了车道线以外，道路图像还常常包含着大量的干扰信息，例如道路旁边的障碍物，树木等。如果将一张图像直接进行识别，这不仅会造成部分车道线无法识别，还会严重影响机器识别图像的效果和效率，所以在进行车道线识别之前必须对道路进行图像的预处理。

1. 图像识别的人工神经网络训练方法

图像经过预处理后， 为了让机器（智能车辆）能够准确的识别出图像中的车道线，一般都会对车辆进行训练，让其在日后的使用中能自主的判断出车道线的位置以及转向，目前，运用的最多的训练方法就是运用人工神经网络进行训练，而在众多的神经网络中，本研究拟选的为卷积神经网络。

1. 小车车道识别算法与控制设计

通过机器视觉识别出了道路，如何根据识别出的图像进行小车的控制和判断也是本课题的一个研究内容。根据研究内容，本论文结构如下：

第1章 绪论。主要介绍了论文的研究背景及意义、车道线检测的国内外研究现状、论文研究内容。

第2章 相关技术。本章主要针对本系统所使用的算法相关技术进行介绍，为后续工作提供理论依据。

第3章 车道线检测系统需求分析。本章主要针对车道线检测方法进行需求分析。介绍车道线检测的重要性，分析了智能车辆中车道线识别的整体结构，明确了本课题需要完成的目标，最后进行用例分析。

第4章 车道线检测算法设计。本章主要针对车道线检测算法模型进行设计与仿真。根据第三章的需求分析、道路图像的特点进行算法的总体设计，之后分别针对图像预处理，卷积神经网络结构，强化学习算法进行详细设计，最后进行仿真验证。

第5章 车道线检测方法设计。本章主要针对智能车辆中车道线检测进行设计。根据第三章的需求分析、第四章的算法设计，针对总体结构进行分析及设计，对车道线检测进行了功能模块的划分，之后针对各个模块进行了详细的设计描述。

第6章 车道线检测方法的实现及测试。本章主要针对车道线检测方法进行实现与测试。根据第五章的车道线检测方法设计，首先介绍系统实现配置环境，其次分模块详细介绍各模块的实现，最后进行仿真测试。

相关技术介绍

## 图像预处理算法概述

### Canny边缘检测

### Hough Transform 直线提取

## 神经网络概述

### 普通人工神经网络概述

### 卷积人工神经网络概述

## 强化学习概述

### 马可夫决策过程

## 本章小结

车道线检测需求分析

无论在工厂中还是在道路中，对于无人车的需求都是非常大的。尤其在工厂中，当工厂环境过于恶劣时，或者工厂环境不易于人工操作时，企业往往会选择用无人车实现物料的运送。在机器视觉之前的无人车采用的大致是磁感线感应无人车和红外感应无人车，对于磁感线感应无人车，厂区需要大量铺设磁感线，而且一旦确定了路线便不能改动，磁感线也随时需要更换和维护，这样的无人车不仅没有灵活性，而且耗费巨大。第二种红外感应的无人车，红外往往易受外部环境的影响，而且对于红外感应的无人车厂区也需要对无人车所需要行驶的路线进行一些处理以便红外区别不同的颜色。相比较，基于机器视觉的无人车对环境没有特殊的要求，而且因为采用的是高分辨率摄像头与雷达的结合所以对于环境的变化会有自己相对应的处理方式，后续对于厂房的维护也不再需要大量的人工协助。

对于整个无人车系统，大致可分为两个模块：感知模块，决策模块。其中感知模块是决策模块的基础。机器通过感知模块对外界环境进行读取和解析，解析的结果传入决策模块，由决策模块对当前环境进行控制的决策。是否能保证无人车系统在任何环境中都能自适应的调整控制成为了如今研究的一大热点。为达到此目的，感知模块的有效设计成为了关键。在如今的感知模块中，机器视觉成为了主要的读取环境的方法，而为了保证智能车在无人操控环境下按规则行驶，准确检测出车道线变得不可或缺。

车道线检测系统需求分析

车道线检测是无人车感知模块中的不可或缺的一个重要环节，为了准确的控制无人车下一步的动作，车道线检测结果需要传递给无人车决策模块用以作为路径规划的基础信息。在本系统中，车道线最终检测结果以二进制状态00,01,10,11（停止，左转，右转，前进）返回给决策系统。

本文主要利用深度学习对车道线进行检测。深度学习是基于大规模数据驱动的一种机器学习方法，本文选择了运用卷积神经网络对车道线进行检测和分类，属于有监督的学习方法。有监督的学习方法需要大量训练集，并且每一个训练的图像都由对应的Ground Truth (标签)，及每张含车道线信息的图像的类别是明确标出的。基于以上所述，本文提出了5个功能需求：车道线标注，图像预处理，车道线检测。3.2节将对这几个功能需求进行详细介绍。对于整个车道线检测系统如图3.1所示，需要有上位机和下位机，并且上位机和下位机要有有效的通讯，下位机进行图像的采集并且传送给上位机，上位机需要对所接收的图像进行预处理，并且将处理后的图像输入CNN图像分类模型中进行车道线的识别和状态分类，最后由分类的结果产生控制指令并且传送给下位机。



3.1车道线检测系统结构图

功能需求

无人车感知模块在整个无人车系统中奠定了一切控制的基础，车道线检测属于感知任务，主要目的是从包含道路信息的图像中检测行车道路中的车道线。本文采用了卷积神经网络对海量行车图像数据进行训练，完成训练后使用已训练并通过测试的模型对图像中的道路信息进行提取和分类，最后利用分类出的各个车道线所处的状态使用强化学习进行车辆的控制。

### 车道线标注

深度学习技术是基于大规模数据集训练而习得的过程。 训练集应保证训练图像与正确分类结果一一对应。为了满足图像信息与分类结果一一对应，每一张含道路信息的图像都应该人工的一一进行标注，通过人工定义的方式决定该张图像所属于的类别。该功能需求为图像预处理的一部分，核心内容为将收集的车道线图像信息进行人工的筛选，将其按4大类别进行标记：左转，右转，停止，后退，人工将每一张图像与对应的标记（label）一一对应。已分类的图像结果在后续会用于模型的训练。

### 图像预处理

在深度学习技术中，用于训练的训练集往往规模都是上千，甚至是上万，所以，训练集的选择在深度学习中至关重要。训练集太泛化，机器则无法找到其中的规律，训练集过于详细，机器则无法将所习得的模型用于非训练集的集合中。训练集的质量的好坏往往确定于图像的质量。在不同场景下，由于天气，光照等的自然原因，每一张车道线图像的质量会参差不齐。除此之外，在原始图像中所包含的信息众多，一些不相关的信息，如道路两旁的建筑以及任何物体，天空等会对神经网络的分类增加额外的负担。因此，选取ROI（Region of interest，感兴趣区域），进行边缘检测，以及直线提取都成为了图像预处理中的一部分。通过选取ROI，筛选出图像中对于车道检测没有参考意义的范围，选取对车道线检测具有较大影响的范围作为研究对象。 边缘检测用于将图像中的边缘信息更加的突出，因为车道线的变换往往都是由边缘的变化所决定，所以正确的突出边缘会对之后的分类减少不必要的检索过程。预处理模块主要作用就是为了将图像处理为深度学习模型所需要的文件，并且通过预处理尽可能的保证能将在训练中不必要的额外负担降低到最小。

### 车道线检测

为了到达稳定安全的行驶，车道线的检测需求成为最主要的功能需求。由于图像信息易受环境的影响而变化，如何在变化的环境中保证稳定的车道线检测成为关键。车道线检测功能需求的核心内容是运用深度学习技术，训练出车道线分类模型，最终达到的效果为输入一张原始车道线图像，模型能准确的对该图像中的车道线信息分类。通过预处理的车道线图像会被输入卷积神经网络模型中进行不断的调参训练，该部分需要负责实现车道线分类模型的结果解析以及车道线提取算法的核心实现。

### 结果显示

在前人的车道线检测中，检测完的结果直接进入了决策模块参与下一步的路径规划，但是，对于研究者或者用户来说，观察车道线检测出的结果一方面能帮助研究者判断检测系统是否正确检测出车道线的位置，另一方面能帮助研究者对训练过程中每个像素点对最后分类结果所做出的贡献进行分析。该功能需求的核心内容为运用Class Activation Map 将图像中的车道线通过热感图高亮出来。该部分主要功能为通过热感图的显示将车道线信息显示出来，更加直观的体现在分类过程中特定区域对分类结果的影响。

### 车辆控制

在前人的研究中，仅仅只是将车道线识别提取出，但提取出的车道线最终目的是为了能为车辆的控制提供基础和参考信息，所以，在本文中该功能需求的核心内容为运用无监督式的强化学习来进行车辆策划模块的控制。通过结合已分类图像的状态，运用强化学习来判定车辆的下一步动作。

非功能需求

### 性能需求

训练车道线分割模型是一个非常耗时的过程，由于训练数据量大，模型训练计算复杂，通常根据神经网络模型的大小训练一次大概需要半天甚至更长的时间，因此提高系统的数据并行处理能力非常必要，本文中训练加速部分主要采用 CUDA 优化。

机器学习本来就是基于大规模数据的习得，深度学习在机器学习上又增加了学习的数量以及次数，会存在更多的超参数，由于训练数据量大，模型训练计算复杂，本文训练主要采用GPU训练。由于用于智能小车中，实时性必须得到一定的保证，因此，对于已训练好的模型，单帧的处理时间应该控制在1s以内。

### 鲁棒性

在机器视觉中，外界环境变化所带来的影响是不可避免的。道路中车道线种类繁多， 路况随着时间、地点、天气的变化而不停的变化。不同的时间 、地点天气可能不同，路面可检测区域大小也会不同，不同光照也会对检测产生影响，光照强烈时容易出现逆光现象，当光照有阴影时容易出现路面车道线被覆盖。本文最终的车道线检测算法，需在各种光照不同的场景下，能够准确识别检测出车道线，保障车辆的正常行驶

本章小结

本章主要针对车道线检测系统进行了需求分析，首先介绍了目前无人车中车道线识别的做法及车道线识别的重要性，之后分析了车道线识别系统的整体结构，明确了车道线识别系统的功能及结构，分析了车道线检测系统的功能需求和非功能需求，对每个功能需求都进行了详细的说明，并对非功能需求预期的标准进行了描述

车道线检测算法设计

根据本文所使用的车道线检测算法的特点，车道线检测系统大致可以划分为 6 个模块，每个模块分别负责每个功能需求的实现，车道线图像数据筛选模块负责筛选高质量的车道线标注数据，车道线图像预处理模块负责将标注好的车道线进行适当的预处理以简化后续的分类，在当中所使用的预处理方法有：ROI提取范围设计来提取出感兴趣区域以减少后续卷积的范围以加快识别的速度，边缘提取设计与直线提取设计用来将图像边缘突出，并过滤掉不必要的参考信息。CNN卷积网络模块主要负责车道线检测模型的训练，最后控制模块主要负责依据车道线图像分类结果控制车辆的方向以及动作。图 4-1 为系统模块图。



4.1 系统模块图

车道线图像数据标注与筛选

本文主要采用深度学习技术检测车道线。深度学习由简单神经元组成的层数较深的多层感知机模型，其利用强大的非线性特性实现高维非凸函数的数学逼近，复杂的网络结构使其能够从海量数据中学习出具代表性的特征。根据深度学习的技术特点，整个车道线检测大致可以划分为三个阶段，分别是数据准备阶段、模型训练阶段以及车道线检测阶段，每个阶段又有若干模块划分，如图 4.2 所示。数据准备阶段的核心是负责深度学习训练的样本数据与 Label 数据准备，其主要包括数据的采集、标注、筛选以及预处理。



4.2车道线检测架构图

车道线图像数据标注模块主要完成标注行车图片中车道线的状态，并将这些区域按照指定的文件格式进行保存。车道线标注的设计主要根据车道线识别算法而设计的。本文的车道线识别算法主要采用CNN卷积网络分类的方式，车道线标注采用打标签（label）的方式，将图像信息与文字信息一一对应。

本文主要采用数组的方式，将车道线状态分为四种状态：00为无车道线，01为只有右侧车道线，10位只有左侧车道线，11为两侧都有车道线。图 4-2 为车道线标注示例图。



4.2 车道线标注示例图

车道线图像预处理

图片预处理模块的核心是针对行车图片进行指定的预处理操作，将其处理为神经网络所需要的格式类型的图片。但图片预处理的方式将影响后续网络模型训练的速度与模型的准确率。本文中的车道线检测主要采用了三种预处理方案，即 ROI（Region Of Interest，感兴趣区域）提取与边缘提取方案以及直线提取方案。

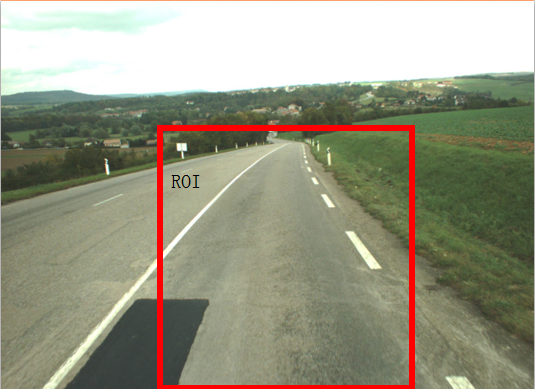
### ROI提取范围设计

ROI 提取主要是为了提取到一张图片中感兴趣的区域块，将其裁剪出来作为一幅新图像的方法。如图4-3所示，图中除了有车道线之外，还有路旁植物，天空等一系列和车道线提取不相关的信息。



4.3 实拍车道线原图

若直接将原图送入CNN网络中进行训练，该CNN网络模型会对每一个像素点进行卷积，池化等运算，这样运算量巨大，而且不相关像素的运算会导致最后对分类精度和速度影响。本文采用的行车图片原图分辨率为 2048\*2048，经过反复实验最终确定从坐标(0,1100)开始提取 2048\*320 大小的 ROI 区域供神经网络进行训练。本文直接采用 OpenCV 提供的 CVSetImageROI 接口对图像的 ROI 区域进行提取。图 4-4 中红框区域为行车图片的 ROI 区域。



4.4 处理后ROI区域

### 边缘提取设计

在车道线识别的过程中，我们只对车道的轮廓曲线感兴趣，因此需要将这些感兴趣边缘提取出来。边缘是两个具有不同灰度值的相邻区域间产生的，是灰度值不连续导致的结果，可通过一阶或二阶导数来检测边缘。在边缘提取中，用模板与图像中的像素进行卷积近似计算，根据模板的大小、元素的取值不同，可以分为多种边缘检测算子，有 Roberts算子、Prewitt 算子、Sobel 算子、拉普拉斯算子、LOG 算子等。本文采用的是Sobel算子，在技术上它是以离散型的差分算子，用来运算图像亮度函数的梯度的近似值， Sobel算子是典型的基于一阶导数的边缘检测算子，由于该算子中引入了类似局部平均的运算，因此对噪声具有平滑作用，能很好的消除噪声的影响。Sobel算子对于象素的位置的影响做了加权，与Prewitt算子、Roberts算子相比因此效果更好。

### 直线提取设计

在ROI区域中，除了需要检测的车道线之外，还有其他干扰的线型，比如道路旁花坛的弧线区域，人形的轮廓等。为了尽量减少最后分类中的卷积时间，对ROI区域进行适当的分类和划分是有必要的。鉴于大多数车道线都为直线（即使是转向时的弯道也可由多条相距较大的直线构成），本文采用了Hough变换来进行直线的提取。

预处理之后，可以对图像进行 Hough 变换，由于边缘图像中存在很多噪声，在随机选取边缘像素点的时候，有一定概率选到虚假车道线边缘，从而影响到 Hough变换后的模型匹配程度。因此 Hough 变换采用投票机制尽可能确保最终得到的函数模型最接近实际车道线。Hough 变换的设计流程如图 4-4 所示：



4.4 Hough变换设计流程

基于CNN的车道线检测算法

模型训练模块主要采用深度学习框架 Tensorflow 进行车道线的CNN模型训练。Tensorflow深度学习框架读取训练文件采用的是 numpy 等标准的数据库格式进行的数据交互，因此训练模型需首先创建相关格式文件的训练和测试样本集，然后设计相关网络结构进行CNN模型的训练。然而网络的训练并不是一次就能成功，需要反复调整网络的参数，而在调参过程中，网络的可视化工具Tensorboard能够很好的协助我们进行调参。图 4-5 为模型训练的流程图

### CNN结构及参数设计

### CNN的训练及测试

基于强化学习的车辆控制算法设计

在前人研究中，大多数都是直接采用了直线提取的方式来进行车道线检测，但是，在现实情况中，由于光照，图像质量的不同，直接采用直线识别的方法会导致误判或者无法识别的情况。除此之外，前方道路具体情况是一个不可预知的图像，所以本文采用了强化学习来进行车辆的控制。

### Q学习方法运用设计

在众多的强化学习算法中，本文选用的是Q学习算法（Q Learning）。Q learning 采用的大致思想是采用状态(s)-动作(a)对奖赏和Q（s, a）作为估计函数，提出另一种与模型无关的学习算法。在本文中，小车每行动一次都会到达一个新的状态：两条车道线都出现，只有左侧车道线出现，只有右侧车道线出现，出现横向的车道线或根本就没有车道线的出现。每一种状态下可做出不同的动作（action）。根据现实情况，本文设计了一个Q（s, a）,s为目前的状态state，a为所需要采取的动作action，此Q（s，a）作为估计函数用于根据目前的状态state来进行aciton的选择，同时，惩罚函数如公式4.1

公式4.1

根据公式4.1，当车辆通过Q（s, a）选择唯一action后，若进入了新的状态为左右车道线都存在，则r = a (a = 0)，相当于车辆选择了正确的action，得到a个奖励值；若进入的新的状态为只有左车道线或只有右车道线，则r = 0，相当于选择的action所带来的结果不是最好的，但是也不是错误的；若进入的新的状态为完全没有了车道线或出现了横向的车道线，则r = -b（b > 0），相当于选择了错误的action导致了车辆直接驶出了车道线，会被惩罚。

本章小结

本章主要针对车道线识别算法模型详细介绍了每个功能对应的模块的具体实现方式，以及实现该模块所需要使用的相关算法和工具。根据车道线识别的流程，本章首先介绍了图像预处理的方式：数据图像的标注和筛选，ROI区域的选取和直线的提取；紧接着，介绍了卷积神经网络（CNN）的设计，运用CNN来进行车道线图像中车道线的识别和状态分类，主要介绍了所采用的CNN的架构，以及参数寻优的选择。最后，本章介绍了运用强化学习来进行车辆的控制，详细介绍了强化学习中的Q learning的整体设计以配合智能车的控制

车道线检测算法实现

基于卷积神经网络的车道线检测是以数据为驱动，使用有监督学习的方式去训练一个图像分割模型，分割出图像中车道标志线区域，然后通过拟合方式，获取车道线转向趋势的检测方法。该方法的核心是数据以及网络模型的训练，数据的多少将影响深度学习模型对各式各样的环境检测的鲁棒性，网络模型的设计将直接影响深度学习模型检测的效果与性能。本章将根据车道线检测系统的整体架构与总体流程对检测中的每个模块实现的环节进行详细介绍，分析其中所使用的算法。

车道线图像数据预处理实现

图片预处理模块的核心是针对行车图片进行指定的预处理操作，将其处理为神经网络所需要的格式类型的图片。但图片预处理的方式将影响后续网络模型训练的速度与模型的准确率。本文中的车道线检测主要采用了两种预处理方案，即 ROI（Region Of Interest，感兴趣区域）提取+直线提取的方案。数据预处理模块主要完成标注行车图片中车道线区域，并将这些区域按照指定的文件格式进行保存。该模块主要用于将已知训练集图片进行部分优化用于卷积神经网络的识别。本文的数据预处理模块主要采用Canny边缘检测以及Hough变换来进行车道线的预处理。

### ROI区提取实现

ROI 提取主要是为了提取到一张图片中感兴趣的区域块，将其裁剪出来作为一幅新图像的方法。如图 5.1所示，行车图片中，天空、车辆等非车道线物体占比较高，若直接将原图送入神经网络进行分类训练，车道线可解析区域比例偏小，不利于训练，提取ROI后车道标志线区域像素占比增高，有利于提高车道线方向分类效果。本文采用的行车图片原图分辨率为 2048\*2048，经过反复实验最终确定从坐标(0,1100)开始提取 2048\*320 大小的 ROI 区域供神经网络进行训练。本文直接采用 OpenCV 提供的 cvSetImageROI 接口对图像的 ROI 区域进行提取。图 5.2 为行车图片的 ROI 区域。



5.1 车道线原始图片



5.2 车道线ROI区域

### 边缘检测实现

在机器视觉算法中，边缘检测是非常基础与重要的环节，在各类物体识别中都有广泛的应用。边缘通常以图像中某些不连续特性为表征，如灰度的突变、色彩（RGB）的变化、纹理结构的变化等。边界线或边缘表示图像中某一区域过渡到另一区域。边缘识别的任务是将图像中的边缘像素凸显标识。通过对图像边缘进行检测及优化，为后续直线提取的步骤提供了方便，在众多边缘检测算法中，本文选用了基于Canny算子的边缘检测，该算法是基于一阶微分算子的边缘检测方法，易于实现并且运算复杂度较低，有益于实现识别的实时性，本文采用Opencv中cv2.Canny()来实现边缘检测，具体步骤图5.3：



5.3 边缘检测步骤图

针对本章提出的Canny边缘检测算法，本节将进行试验比对，采用实际行驶道路图像，测试边缘检测算法的效果。本次测试图集为来自不同地区、不同路况的 100幅道路图像，图像分辨率为 640×480，测试平台为以英特尔酷睿 2 双核 1.8GHz 为核心处理器的计算机，算法实现均采用 OpenCV 开源库简单实现，未采用多线程等其他优化手段。图5.3为测试图像中的示例，给出本章所介绍Canny边缘检测算法识别效果图，(a)为图像预处理之后的灰度图像，(b)为Canny边缘检测算子的识别结果，从识别结果图像，以上算法均能检测车道线的轮廓像素.

### 直线提取实现

边缘提取处理之后，可以对图像进行 Hough 变换，由于边缘图像中存在很多噪声，在随机选取边缘像素点的时候，有一定概率选到虚假车道线边缘，从而影响到 Hough变换后的模型匹配程度。因此 Hough 变换采用投票机制尽可能确保最终得到的函数模型最接近实际车道线。Hough 变换的实现流程如图 5.5 所示：



5.5 Hough变换流程图

本文主要采用opencv中的cv2.HoughLines来进行Hough变换的实现，实现的主要过程为：

（1）选取边缘像素点。由于边缘像素中含有很多噪声点，因此在随机选取边缘点时本课题采用基于概率的选取方法，即该边缘像素被选中的概率正比于其灰度变化值，这样能更容易选到灰度变化明显的边缘像素，意味着能更大概率上选中真正的边缘像素。

（2）投票机制（Voting）。由图像空间中取两个点，即可由参数空间中两条曲线的交点得到直线的参数，经过不断地重复选点—求直线参数这一过程，对直线的参数坐标(k,b)做统计，累积票数最多的（k,b）参数对即为所求的直线参数。在投票机制中，循环统计的次数越多，最终得到的直线就越接近真实的车道线。

基于CNN的车道线检测算法实现

基于深度卷积神经网络的图片分类识别方法,因其识别正确率已经超过了传统的基于局部特征的方法,在物体识别方向上变得越来越流行。本文针对车道线图像也尝试使用卷积神经网络方法来对提取车道线特征, 并提出了基于卷积神经网络的车道线检测模型。本章详细介绍了该模型的设计和实现,最后对卷积神经网络结构设计与调参方法进行了总结

### CNN整体架构实现

本文采用的由卷积神经网络（Convolution Network）构成的网络模型，为了能将所分类出的图片的状态趋势显示于原始输入图片中，在基础的卷积神经网络中，本文将最后的全连接层换为了activation map。

1. CNN基本架构

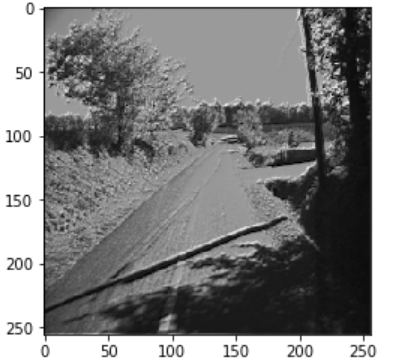
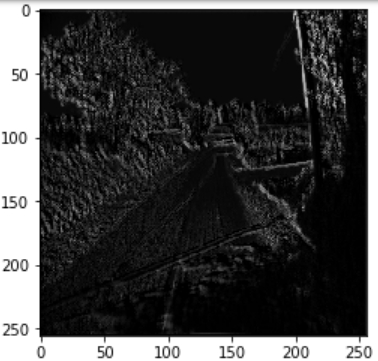
该CNN结构总共包含6层, 如图5.6所示。第1层输入为原图片: 251x 251卷积层conv1包含16个核, 核大小为5, 步长为1, 降采样层pool1层用max - pooling方式, 核大小为3, 步长为2。第2层输入为: 123x123, 卷积层conv2包含32个核, 核大小为5, 步长为1, 降采样层pool2采用max - pooling方式, 核大小为3,步长为2。第3层输入为:59 x 59, 卷积层conv3包含64个核, 核大小为5, 步长为1, 降采样层pool3采用max-pooling方式, 核大小为3, 步长为2。第4层输入为: 27x27,卷积层conv4包含128个核,核大小为5,步长为1,降采样层pool4义用max-pooling方式,核大小为3,步长为2。第5层输入为:13x13, 卷积层conv5包含256个核,核大小为3,步长为1,降采样层pool5采用max-pooling方式,核大小为3,步长为2。第6层activation map层,输入为6x6,输出是长度为1024的向量。具体结构如图5.6所示：

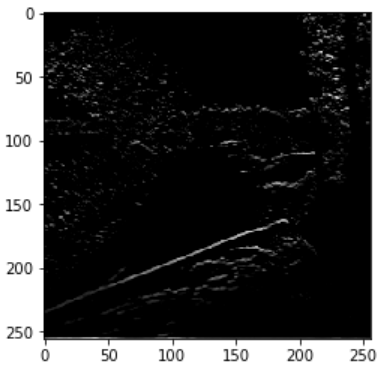


5.6 CNN整体架构图

该网络结构中,主要包含了 “picture”, “conv”, “relu”, “pool”, “activation map”节点。其中 “picture” 节点代表数据的输入或输出,每一层的开始的 “picture” 节点代表把上一层的数据作为输入, 第一层的输入为已经过预处理的车道线图像,每一层最后的 “picture” 节点代表该层数据的输出。"conv" 节点代表卷积操作,在该操作中,每一个卷积核代表对输入的图像数据提取一种特征,多个卷积核代表对图像提取多种特征。"pool" 节点代表降采样操作,当卷积特征向量太大的时候,训练分类器计算复杂度太高, 并且容易过拟合,可以对卷积特征进行聚合统计,来降低特征维度,比如取某个区域特征的最大(maxpooling)或平均值(meanpooling),这种聚合操作就叫做降采样或池化, "relu"节点代表非线性映射操作,传统CNN采用sigmoid或tanh函数来做映射,但是因为稀疏的表示效果会更好,所以使用纠正线性单元,如果卷积计算的值小于0,就让其等于0,否则保持原来的值不变。“norm”节点代表归一化操作,归一化操作分为两种:一种是减法归一化,减法归一化是针对一个卷积特征而言,反映了一个卷积特征中某个位置与邻域位置的关系,具体的操作为将目标位置与领域位畳的加权和相减,邻域位置的权值大小与领域位置距离目标位置的距离有关,在实际操作中,可以使用高斯函数来模拟。 另一种是除法归一化,除法归一化是针对不同卷积特征而言,反应不同卷积特征中相同位置的关系,具体的操作为将不同卷积特征目标位置的值相加求和并取平均,然后将各个卷积特征目标位置的值与得到均值相除。最后，本文中的CNN并未采取普通CNN最后一层所需的全连接层（fully-connect layer），为了能够直观的看出图像中每个像素点对于最后分类结果的影响，本文最后舍弃了全连接层，而采用了activation map的方式，通过该方式，各个像素点对于结果的影响会由红蓝热感图在原图中体现。

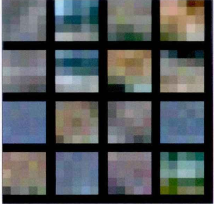
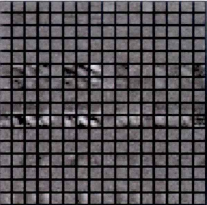
本文所采用的网络结构主要基于文献[\*]中使用的结构,差别主要在前几层。文献中针对的问题是动物识别和图像分类的问题, 类与类之间的差异较大, 因此在原文献[\*]中，其CNN结构中的前几层让图像快速的缩小, 这样也能保证最后分类的准确性, 所以在该文献中, 原始图像经过第一层之后, 图像变为原来的1/8,其中卷积操作的步长为4, 池化操作步长为2。但是本文针对的是车道线识别问题, 每一类都是车道线, 类与类之间的差异较小, 因此不能让图像在CNN结构中的前几层过快的缩小, 因为这样做会导致车道线某些细节的丢失, 最后降低分类的正确率。在本节介绍的CNN结构中, 所有层中的卷积操作的步长都为1, 池化操作的步长为2, 也就是图像每经过CNN结构中完整的一层,图像变为原来的1/2。 本节介绍的CNN结构中的参数都是基于大量的实验测试得出的, 其中最重要的是每一层中卷积核的个数, 在调试参数的过程, 具体参数可以通过可视化模型来调整。可视化是指将不直观的数据参数转化为直观的图像。模型可视化包含两个部分:数据流可视化和卷积核可视化。 数据流是指图像数据在CNN结构中的流动, 其中图像数据主要经历卷积和池化两种操作。观察数据流可视化的结果可以调整模型中卷积核的个数, 如果发现数据流可视化结果中有大量相似或者无意义的图像, 说明卷积核个数设置太多, 这时候应当减少卷积核的个数, 如果发现数据流可视化结果中的图像都各不相同, 这时候可以适当的増加卷积核的个数, 直到选出最好的参数。 卷积核代表对图像提取特征, 一个卷积核代表对图像提取一种特征, 多个卷积核代表对图像提取多个特征。低层的卷积核代表局部特征, 高层的卷积核代表全局特征, 在卷积核可视化的结果中, 有些核会表示成特别有意义的图案,比如横线、竖线、曲线等等,如果在可视化卷积的结果中没有发现含有明显图案的图像, 说明CNN结构设计不合理, 可以调整CNN结构的层数, 若当前CNN结构的层数较少, 应该增加CNN结构的层数, 若当前CNN结构的层数较多, 应该减少CNN结构的层数。

1. 模型可视化
2. 数据流可视化



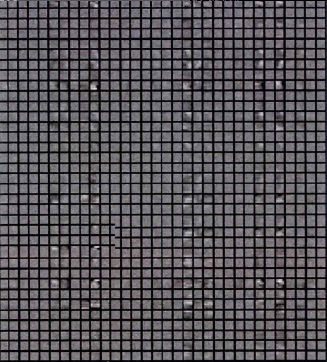
5.7 原始输入图像 5.8 Conv1卷积结果 5.9 Conv2卷积结果

将图5.7作为CNN模型的输入。该CNN结构第一层卷积节点"conv1"有16个核,所以原始输入图片对应16个输出, 并且16个输出都各不相同, 说明CNN训练得到的16个核提取的特征都各不相同, 有些核提取的横的边缘, 有些核提取的竖的边缘, 如图5.8 所示。该CNN结构第H层卷积节点"conv2"有64个核, 所以上一层的一个输入对应64个输出,送64个输出也各不相同, 但是相比于卷积节点"conv1"的结果, 该卷积层的输出更加全局化,如图5.9所示。

5.10 Conv1卷积核 5.11 Conv2卷积核

5.10为CNN结构中第一层卷积节点"conv1"中16个卷积核的可视化结果, 因为输入图片有H个通道, 因此可W把通道叠加在一起形成彩色结果。 图5.11为CNN结构中第二层卷积节点"conv2"中32个卷积核的可视化结果, 因为上一层的输出为16个通道, 所以图14是部分卷积核可视化结果, 部分图像表示为特征明显的边缘。



5.12 “Conv3”卷积核

5.12 为CNN结构中第H层卷积节点"conv3"中64个卷积核的可视化结果, 因为上一层的输出为32个通道,所以图5.12是部分卷积核可视化结果,部分图像表示为特征明显的边缘.

（3）activation map

在最后一层网络模型中，普通的CNN网络模型都会选择softmax作为全连接层，但是，本文为了让每个像素对于最后分类结果的影响能够直接显示于图片中，本文将全连接层替换为了activation map。

本文在该网络模型的最后一层进行特征卷积提取时，采用了全局平均池化（global average pooling）对特征图（feature map）进行了提取，然后运用这些特征作为全连接层的输入在原始输入的车道线图像中进行反馈显示。如图5.13所示，经过全局平均池化后的输出为每一个特征值在最后分类结果中在全图中所占的比例。所有特征值所占用的比例和最后形成一张activation map。对于给定的图片，我们令表示在空间位置（x, y）中，类别k所占的比例，对于k，表示经过全局平均池化的结果，其中：

(1)

因此，对于给定的类别C， 输入到全连接层softmax的输入为

(2)

其中，表示在C类中激活像素单元k的权重。

最后， 全连接层softmax 对于类别C的输出为

= (3)

由公式（1）与公式（2）合并，得到：

在本文的网络结构中，我们设定为类别C的class activation map, 因此，对于每一个位置（x, y）有：

因此，如图5.13所示，当Class activation map与全连接层结合使用时，每一个像素对于最后结果的影响会以热感图的形式体现出来。

5.13 Class Activation Map示意图

### CNN训练及测试

本文所构造的CNN网络模型都是建立在Tensorflow平台，通过Tensorflow中的卷积神经网络库来实现。完成网络架构之后，便是CNN的训练，本节面向ROMA车道线图像库对 CNN 的结构进行优化调参，结合图像库和卷积神经网络，根据输入图像大小和卷积核大小确定网络结构层数为 6层，整理系统结构框图如图 5.14所示。图 5.14研究过程包括两方面：训练和测试。训练过程是由若干次迭代和每次迭代中若干次训练组成。将训练图像样本分成若干组，分别设为批量样本 1、批量样本 2 直到批量样本 n。每次训练过程是由前向传播（第一次卷积以及第一次降采样为第一层、第二次卷积以及第二次降采样为第二层、第三次卷积以及降采样为第三层，第四次卷积以及降采样为第四层， 第五次卷积以及降采样为第五层， 最后Activation map和全连接结合为第六层），反向传播，权值更新三部分组成。特征提取过程即为前向传播过程。测试过程是将测试图像样本经过前向传播过程得到最终输出，输出最大值即可确定类别，最后计算错误率。



5.14 车道线识别系统训练及测试框图

（1）整个网络的训练过程

1）对车道线图像设定分类标签；

2）将权值 W 和偏置 b 置成[0，1]的随机值，并初始化参数和学习率，通过输入图像的大小设定卷积核大小为 5×5 和网络结构层数为6层，并使用实验的方式设定批量样本数为 40 和迭代次数为 50 次；

3）从车道线图像训练集中取第一个批量样本输入到训练网络中；

4）计算中间层输出向量和整个网络实际的输出向量；

5）将输出向量中的元素和标签中的向量进行比较，计算输出误差，以及中间隐层的误差；

6）计算出各权值的调整量△W 和偏置△b；

7）更新权值和偏置；

8）当迭代次数满足设定即转入（9）。如果不满足返回（3），继续迭代；

9）训练结束。

基于强化学习的车辆控制算法实现

### Q学习方法实现

本章小结

车道线检测方法实现及测试

系统实现测试环境

测试实例

测试结果

本章小结

结 论

参 考 文 献

1. 刘富强，田敏，胡振程. 智能汽车中基于是觉的道路检测与跟踪算法[J].同济大学学报（自然科学版）.2007，35：1535-1537.
2. Wang, Chenhao, Zhencheng Hu, and Keiichi Uchimura. "A novel lane detection approach fusion by vehicle localization." In Intelligent Control and Automation (WCICA), 2011 9th World Congress on, pp. 1218-1223. IEEE, 2011.
3. Li, Jian, Xiangjing an, and Hangen He. "Lane Detection Based on Visual Attention." In Image and Graphics (ICIG), 2011 Sixth International Conference on, pp. 570-575. IEEE, 2011.
4. S. Zhou, Y. Jiang, J. Xi, J. Gong, G. Xiong, and H. Chen, “Anovel lane detection based on geometrical model and gabor filter,” in Intelligent Vehicles Symposium (IV), 2010 IEEE, 2010, pp. 59–64..
5. Aly, M. (2008). Real time Detection of Lane Markers in Urban Streets. In:Proceedings of IEEE International Vehicles Symposium, pp. 7-12.
6. A.Borkar, M.Hayes, and M.T.Smith, "Polar randomized hough transform for lane detection using loose constraints of parallel lines." Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on. IEEE, 2011.
7. Alon, Y., Ferencz, A., and Shashua, A. (2006). Offroad path following using region classification and geometric projection constraints. In: Proceedings of International Conf. on Computer Vision and Pattern Recognition, vol. 1, pp.689-696.
8. Keyou, Guo, Li Na, and Zhang Mo. "Lane detection based on the random sample consensus." In Information Technology, Computer Engineering and Management Sciences (ICM), 2011 International Conference on, vol. 3, pp. 38-41. IEEE, 2011.
9. Y.U. Yim and S.- Y. Oh, "Three-feature based automatic lane detection algorithm (TFALDA) for autonomous driving," IEEE Trans. Intell. Transp. Syst. , vol. 4, no. 4, pp. 219-225, Dec. 2003.