

嵌入式系统设计综合项目

开发文档

基于关键点的全球关联网络车道检测

第六组

组员：2050235 倪韵旖

2054402 罗敏琪

2050277 林晨

2051826 杨正青

2053181 邱继正

目录

- 1.引言 3
- 2.需求分析 3
 - 2.1.功能需求 3
 - 2.2.性能需求 3
- 3.模型架构 4
- 4.数据集和预处理 4
 - 4.1.数据集 4
 - 4.2.预处理 5
- 5.模型实现 5
 - 5.1.模型设计 5
 - 5.2.模型训练 6
 - 5.3.模型测试 6
- 6.车道线检测系统实现 7
 - 6.1.开发板移植 7
 - 6.2.输入处理 8
 - 6.3.输出处理 8
- 7.总结 9
 - 7.1.系统不足 9
 - 7.2.展望与未来 9

1. 引言

车道线检测是自动驾驶系统中的一个关键组成部分,它可以帮助车辆在道路上保持正确的行驶方向和车道位置。本项目旨在开发一种基于关键点的全球关联网车道检测嵌入式系统,以 Atlas 200DK 为开发板,搭载树莓派 v2.1 摄像头,web 端展示输出图像。该系统可以在不同的场景下准确地检测车道线,包括不同类型和角度的车道线,以及不同的光照和天气条件。本文档将详细介绍该系统的需求分析、模型架构、数据集和预处理、模型设计和实现以及车道线检测系统。

2. 需求分析

2.1. 功能需求

1. 支持树莓派相机进行图像拍摄

系统需要支持树莓派 V2.1 相机的连接,并通过相关程序调用相机使其能够实时拍摄车道线图像。

2. 支持图像预处理

系统需要能够对摄像头拍摄的图像进行预处理,包括裁剪、缩放等操作,以便输入到模型中进行处理。

3. 实现车道线的检测

系统需要能够实现车道线的检测,即根据输入的图像,生成车道线的关键点坐标,并根据坐标构建结果图像。

4. 支持输出结果传输

系统需要将模型生成的结果图像通过 WebSocket 传输到网页上,使得用户可以在任何 PC 上能够实时查看开发板传来的生成图片,达到可视化的输出效果。

2.2. 性能需求

1. 准确率

车道线检测系统需要具有较高的准确率,能够准确地检测出道路上的车道线。

2. 响应时间

车道线检测系统需要具有较快的响应时间,尤其是在高速公路等道路上,需要能够及时地检测到车道线的位置。

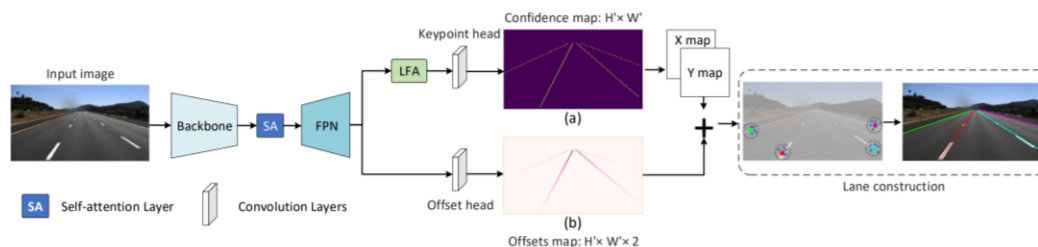
3. 系统性能

由于 Atlas 200 DK 开发板不支持多线程运行,因此需要使用单线程模式运行模型,可能会影响系统性能。因此,需要对模型进行优化,以提高系统的性能。

4. 系统稳定性

车道线检测系统需要具有较高的稳定性,能够在长时间的运行中保持良好的性能。

3. 模型架构



图一：GANet 模型架构

GANet（基于关键点的全局关联网络）模型可拆分为 4 个步骤。

Step1: 图像特征提取

ResNet18 算法对图像进行卷积，并将每一层卷积后的特征图输入 SA（self-attention）层，增加图像某个像素与全局的关联性，再将 SA 的输出输入 FPN，通过自上而下、横向链接和卷积融合得到最后的特征图像。

Step2: LFA 与关键点估计

首先利用 LFA 一个变形卷积层预测点周围 M 个点的偏移量：参数是周围点数量 M 以及与卷积层相关的参数。引入了两个损失函数对参数进行优化。然后利用该卷积层得到的周围点偏移量聚合该点的新特征。再进行关键点估计：利用 LFA 层的输出，得到一个置信图。利用高斯核区分关键点和非关键点，利用损失函数解决样本不平衡问题，并且优化高斯核参数 σ 。置信图表示点为关键点概率。

Step3: 通过偏移量层卷积得到偏移量图，其表示每一个关键点到其所属车道点的矢量的集合。

Step4: 讲关键点置信图和偏移量图做矩阵加法，从而将属于同一条车道线的关键点依次相连，形成完整的车道线

4. 数据集和预处理

4.1. 数据集

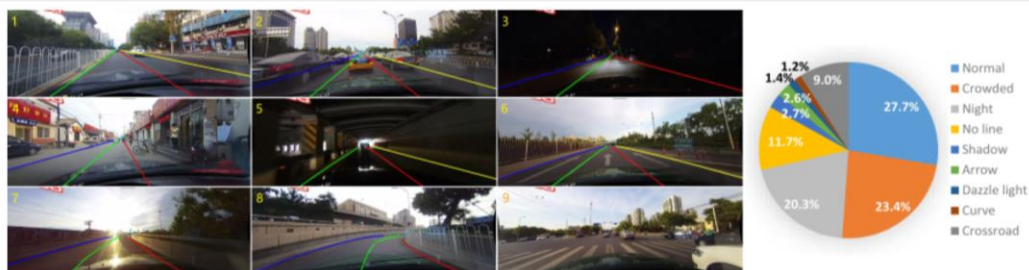
在模型的训练过程中，用到的数据集为 CULane，共计 8 万+张图像，数据集覆盖了不同类型例如直道、弯道及不同光照和天气情况下的车道线，使得模型在真实物理环境下的准确率都能达到较高的标准。

CULane

CULane 是一个用于交通车道检测学术研究的大规模挑战性数据集。该数据集是通过在北京不同驾驶员驾驶的六辆车上安装摄像头进行采集的。共采集了 55 多小时的视频，提取了 133,235 帧图像。数据集被划分为 88880 个训练集、9675 个验证集和 34680 个测试集。

CULane Dataset

Multimedia Laboratory, The Chinese University of Hong Kong



图二：CULane 数据集简介

数据集官网: [CULane Dataset \(xingangpan.github.io\)](https://xingangpan.github.io/CULane-Dataset/)

4.2. 预处理

图像的预处理包括调整图像大小，标准化像素值，数据增加，去噪。

1. 调整图像大小

树莓派 V2.1 相机拍摄的图像标准大小为 3280 × 2464 像素，这超过了我们预设的模型输入图像大小，需要将图像大小调整为 1640*590 像素。

2. 标准化像素值

将图像的像素值标准化到一定的范围内，例如[0,1]或[-1,1]，从而提高模型的训练速度和准确性。

3. 数据增强

对图像进行各种变换，例如旋转、翻转、平移等，以增加数据的多样性，从而提高模型的鲁棒性和泛化能力。

4. 去噪

去噪是指对图像进行去噪处理，例如使用高斯滤波器或中值滤波器。去噪可以帮助模型更准确地识别图像中的特征，提高模型的准确性。

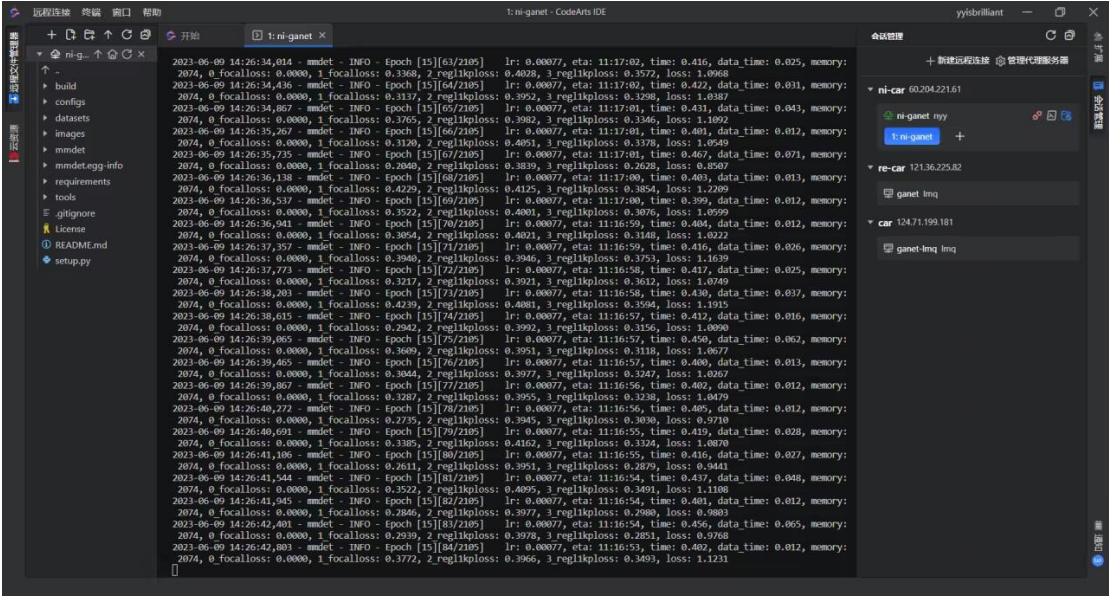
5. 模型实现

5.1. 模型设计

考虑到 Atlas 200DK 开发板不具有 gpu 的环境，而原模型架构中 LFA 与关键点估计需要依赖一个可变形卷积预测偏移量，所以为了训练一个开发板已有硬件可运行的模型，我们将原模型中的可变形卷积层删去，改换为普通卷积层，并修改了涉及到的代码中参数，为下一步模型训练做准备。

5.2. 模型训练

我们用准备的数据集在华为云上对模型进行了训练，在训练过程中，我们监控了若干个 epoch，以便更好地调整模型的训练。



图三：模型训练过程截图

5.3. 模型测试

经过训练后的模型可以通过测试集图像的测试，成功构建出车道线，并且达到较高的准确率。在 PC 端测试中，在 400 张测试集图像（包含 300 张白天正常光照下直线车道，100 张其他环境例如夜晚、弯道、阴雨、遮挡）中，共 373 张识别准确，27 张识别有误，在识别有误的图像中，21 张图像未能准确识别出车道线，6 张识别出错误的车道线，总体准确率达到了 93.25%，总体测试准确率较高。但由于训练时选取的不同类型车道及不同光照及天气的训练集较少，故在测试其他环境时系统的表现欠佳。

| 车道环境 | 总数 | 正确数 | 错误数 | 准确率 |
|--------------|-----|-----|-----|--------|
| 白天光照直线 车道 | 300 | 295 | 5 | 98.33% |
| 其他环境 | 100 | 78 | 22 | 78% |
| 综合 | 400 | 273 | 27 | 93.25% |

表一：PC 端测试准确性

模型测试结果案例：



图四：模型测试结果输出（起始点）



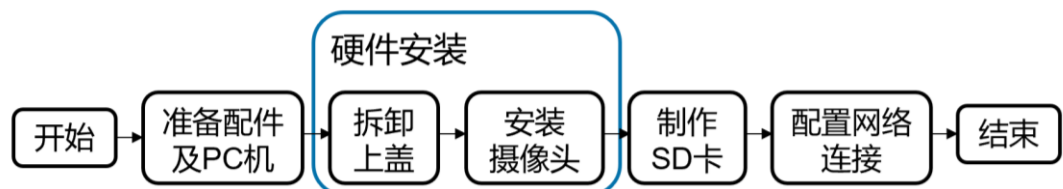
图五：模型测试结果输出（车道线）

更多模型测试相关的数据请见《测试文档》。

6. 车道线检测系统实现

6.1. 开发板移植

开发板移植之前，需先进行环境部署，其中包括准备配件及 PC 机、硬件安装、制作 SD 卡、配置网络连接等步骤。



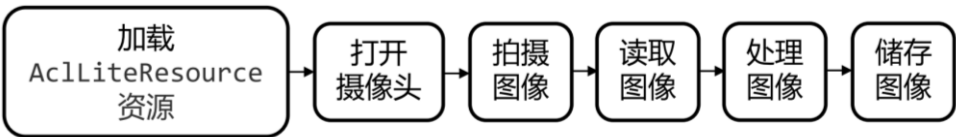
图六：开发板安装部署流程

环境部署后需要在 Atlas 200DK 开发板上安装运行测试程序需要的相关 Python 依赖及调用摄像头需要的 Ascend 相关库文件。

最后将项目传输至开发板上, 运行测试程序系统无报错且能够准确通过测试, 构建出车道线, 至此开发板移植流程结束。

6.2. 输入处理

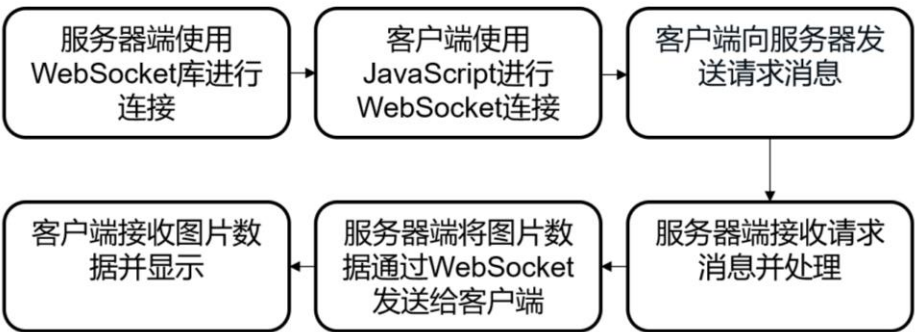
使用树莓派 V2.1 相机拍摄的图像作为输入, 为了使 Atlas 200DK 可以调用树莓派摄像头, 需要利用 Ascend 官方的 python 库, 执行相关代码加载资源-打开摄像头-拍摄图像-读取图像-处理图像-储存图像, 以一定周期循环执行上述步骤不断为模型获取输入图像。



图七：输入处理流程

6.3. 输出处理

模型输出的结果图像将通过 WebSocket 协议传输。WebSocket 是一种全双工通信协议, 可以在客户端和服务端之间建立持久性的连接, 并且可以进行双向数据传输。处理流程为 Atlas 200DK 作为服务器端使用 WebSocket 库进行连接-PC 作为客户端使用 JavaScript 进行 WebSocket 连接-客户端向服务器发送请求消息-服务器端接收请求消息并处理-服务器端将图片数据通过 WebSocket 发送给客户端-客户端接收图片数据并显示。



图八：输出处理流程

7. 总结

7.1. 系统不足

1. 为了在非 GPU 环境下进行模型训练，不得不替换可变形卷积操作。这导致无法使用车道特征对齐 (LFA)，进而影响了最终车道线的划线结果的准确性。
2. 训练数据集仅包含 8 万+张图像。这个数据集规模可能不足以覆盖所有情况，导致训练得到的模型仍然存在一些误差。
3. 尽管我们对代码进行了优化，但在开发板上运行时仍然可能出现一些卡顿现象。这可能是由于开发板的计算能力限制（不支持多线程）或代码中的一些瓶颈引起的。

7.2. 展望与未来

该项目基于关键点的全球关联网车道检测系统，通过使用深度学习技术对车道线进行检测，可以为自动驾驶和辅助驾驶系统提供重要的视觉信息，具有重要的应用前景和未来发展潜力。

在未来，还可以对该系统进行以下展望：

1. 提高算法的准确率和鲁棒性

当前的车道检测算法虽然已经能够在各种路况和光照条件下进行准确的检测，但仍然存在一些误检和漏检的情况。未来的研究可以通过引入更多的数据、优化算法结构和参数、提高数据增强的方法等方式来提高算法的准确率和鲁棒性。

2. 优化算法的速度和效率

当前的车道检测算法需要较长的处理时间和较高的计算资源，这限制了其在实际应用中的应用范围和效率。未来的研究可以通过使用更快的计算设备、优化算法实现、使用轻量级的模型等方式来提高算法的速度和效率，以适应实际应用的需求。

结合其他传感器和数据源来提高检测效果

3. 多传感器结合

车道检测算法可以结合其他传感器和数据源，例如 GPS、雷达、激光雷达等，来提高检测的效果和鲁棒性。未来的研究可以探讨如何将多种传感器和数据源进行融合，以提高车道检测的效果和可靠性。