



2022 年（第 15 届） 中国大学生计算机设计大赛

人工智能挑战赛作品报告

作品编号: 2022036004

作品名称: 边缘计算与边缘智能应用

填写日期: 2022.8.9

目 录

| | |
|---------------------------------|----|
| 第 1 章 作品概述 | 1 |
| 第 2 章 问题分析 | 1 |
| 2.1 计算能力 | 1 |
| 2.2 数据集 | 2 |
| 2.3 车道线检测 | 2 |
| 2.4 摄像头的远程连接 | 2 |
| 2.5 疲劳状态识别 | 3 |
| 第 3 章 技术方案 | 4 |
| 3.1 车道线检测: | 4 |
| 3.1.1 调用 OpenCV 库函数进行图像处理 | 4 |
| 3.1.2 ROI mask | 4 |
| 3.1.3 霍夫变换 | 4 |
| 3.1.4 离群值过滤 | 4 |
| 3.1.5 最小二值拟合 | 4 |
| 3.2 物体检测 | 5 |
| 3.2.1 YOLOv5 算法模型 | 5 |
| 3.3 单目测距: | 6 |
| 3.3.1 测距原理 | 6 |
| 3.4 远程连接摄像头 | 7 |
| 第 4 章 系统实现 | 7 |
| 4.1 模型训练及 RKNN 模型转换 | 7 |
| 4.2 相机标定 | 9 |
| 4.2.1 准备工作 | 9 |
| 4.2.2 进行标定 | 9 |
| 4.3 距离计算 | 10 |
| 4.4 疲劳驾驶检测 | 10 |
| 4.4.1 眼睛轮廓曲线拟合 | 10 |
| 4.4.2 眼睛开合度归一化计算 | 12 |
| 4.5 数据来源 | 12 |
| 4.6 数据规模 | 13 |
| 4.6.1 环境配置 | 13 |
| 4.6.2 测试过程 | 13 |
| 4.6.3 分析与结论 | 15 |
| 第 5 章 作品总结 | 15 |
| 5.1 作品特色与创新点 | 15 |
| 5.2 作品展望 | 16 |
| 参考文献 | 16 |

第1章 作品概述

本作品设计了一款基于卷积神经网络和 http 数据传输协议设计的对交通场景下，用单目摄像头获取场景图像，对小汽车、面包车、卡车、行人和驾驶司机等物体进行识别和定位，通过一种单目测距原理对识别目标进行定位和测速，并且对车道线进行检测。

作品对司机的危险驾驶行为进行分析，通过摄像头等图像传感器获取到的驾驶员面部图像为基础，运用机器视觉中人脸检测、面部特征点定位等算法技术，对驾驶员的脸部变化如眼睛睁闭、嘴巴张合以及头部姿态等特征进行提取和分析，从而实现对驾驶员的疲劳状态的分析判断。

通过分析驾驶员的疲劳状态、车距、车速和车辆于车道线的位置，可以对车辆超速进、车辆碰撞进行预警。当驾驶员有疲劳行为并且与前车车距较小时，可以通过扬声器进行声音报警，在车辆后方的警示灯亮起对后方车辆进行警示。

本作品可以在成本很低的边缘端进行计算，应用于如长途卡车司机，车厢中环境差，无法应用也没用资金去应用自动驾驶系统。在高速路况下，疲劳驾驶时，本作品可以进行碰撞预警和车道线偏移警示。并且可以适用于及其他低成本的应用场景。相比于我们熟知的自动驾驶系统来说，本作品的成本低，并且也可以进行有效的预警。

并且本作品致力于移植到新大陆人工智能开发板上进行教学活动，可以通过有线或无线传输连接摄像头进行实时处理。

第2章 问题分析

2.1 计算能力

因为本作品致力于在算力很有限的情况下解决问题，所以仅采用图像处理技术解决问题。采集图像的测距方式可以分为单目测距和双目测距。单目的优势在于成本较低，对计算资源的要求不高。在于必须不断更新和维护一个庞大的样本

数据库，才能保证系统达到较高的识别率因算力有限，所以本作品采用了单目测距，并且也具有较高的精度，且计算时间可以达到 17FPS 左右。

并且 YOLO 的 One-Stage 模型检测速度大大高于其他双阶段检测模型。可以在算力有限的边缘端达到较高检测速度。

2.2 数据集

单目测距难点在于必须不断更新和维护一个庞大的样本数据库，才能保证系统达到较高的识别率单目测距原理，对于物体的定位具有较高的精度要求，所以我们需要搜集标注较为精确且数量较大的数据集。并且我们针对的场景是在行驶的道路上的汽车拍摄的数据集，所以我们需要搜集满足此场景的数据集。本作品采用了德国卡尔斯鲁厄理工学院和丰田美国技术研究院的 KITII 数据集。该数据集是目前国际上最大的自动驾驶场景下的计算机视觉算法评测数据集。该数据集用于评测立体图像(stereo)，光流(optical flow)，视觉测距(visual odometry)，3D 物体检测(object detection)和 3D 跟踪(tracking)等计算机视觉技术在车载环境下的性能。KITTI 数据集的数据采集平台装配有 2 个灰度摄像机，2 个彩色摄像机，一个 Velodyne 64 线 3D 激光雷达，4 个光学镜头，以及 1 个 GPS 导航系统。本作品可在算力有限情况下检测物体车距，所以只需要摄像头拍摄视觉测距部分即可。并且将数据集转化为 YOLO 格。

2.3 车道线检测

车道线检测同样在算力有限的情况下进行，车道线检测利用了传统图像处理算法中的滤波算法、Canny 边缘检测算法和 Hough 直线检测算法作为基本算法模型,采用只对 ROI 中进行检测的措施来满足对于前方车道线的准确检测。在检测中,使用了 OpenCV 开源图像处理库来对进行上述方法进行实现。此方法可极大减少对前方车道线检测的外界干扰,在汽车实验场中利用该方法,可以比较准确地检测出车辆前方的车道线,并且该算法在一般机器上能够实现实时级的车道线检测进行图像处理，保证了较高的速度。

2.4 摄像头的远程连接

Flask 实现视频流媒体，同时为了拓展开发板的教学功能，本作品远程连接

摄像头用于拓展开发板的演示内容。可以在开发板上远程展示模拟小车的路况分析。

2.5 疲劳状态识别

对眨眼频率、眼睛闭合总时长疲劳识别特征数据进行阈值分析可以对驾驶员疲劳状态进行识别，由于阈值分析算法简单，因此识别速度快，对设备性能要求不高，系统实时性较好。

第3章 技术方案

3.1 车道线检测：

3.1.1 调用 OpenCV 库函数进行图像处理

Canny 边缘检测步骤：

- 1：应用高斯滤波去平滑图像为了去除噪音的影响。
- 2：计算图像的 x 轴和 y 轴的梯度，并计算梯度的合方向
- 3：使用非极大值抑制，对那些伪边界点抑制
- 4：把 min_max 应用到上面得到的图像
- 5：通过滞后跟踪边缘：通过抑制所有其他弱且未连接到强边缘的边缘来完成边缘的检测。

3.1.2 ROI mask

ROI 是选择感兴趣的区域并截取保存在内存中；mask 是掩码或者掩膜，掩码的位置如果为 0，则在这个位置上不做修改；如果不为 0，则做与运算；

3.1.3 霍夫变换

采用检测间断点边界形状的方法霍夫变换，通过将图像坐标空间变换到参数空间，来实现直线与曲线的拟合。

3.1.4 离群值过滤

特征点匹配的结果可能存在离群点，即某些特征点由于受环境的影响会远离大多数匹配上的特征点，所以要将离群点过滤掉防止影响结果。

3.1.5 最小二值拟合

采用最小二乘法使得拟合直线的误差值最小。



图2 车道线检测流程图

3.2 物体检测

3.2.1 YOLOv5 算法模型

YOLOv5 是一种单阶段检测模型，在 YOLOv4 的基础上进行了改进。输入端图像大小为 640 像素，该阶段包含一个图像预处理阶段，将输入的图片缩放到网络的输入大小，并进行归一化操作。在网络训练阶段，YOLOv5 使用 Mosaic 数据增强操作提升模型的训练速度和网络的精度。YOLOv5 使用了 CSPDarknet53 结构，并使用了 Focus 结构为基准网络。YOLOv5s 的基本网络结构如图 2 所示：

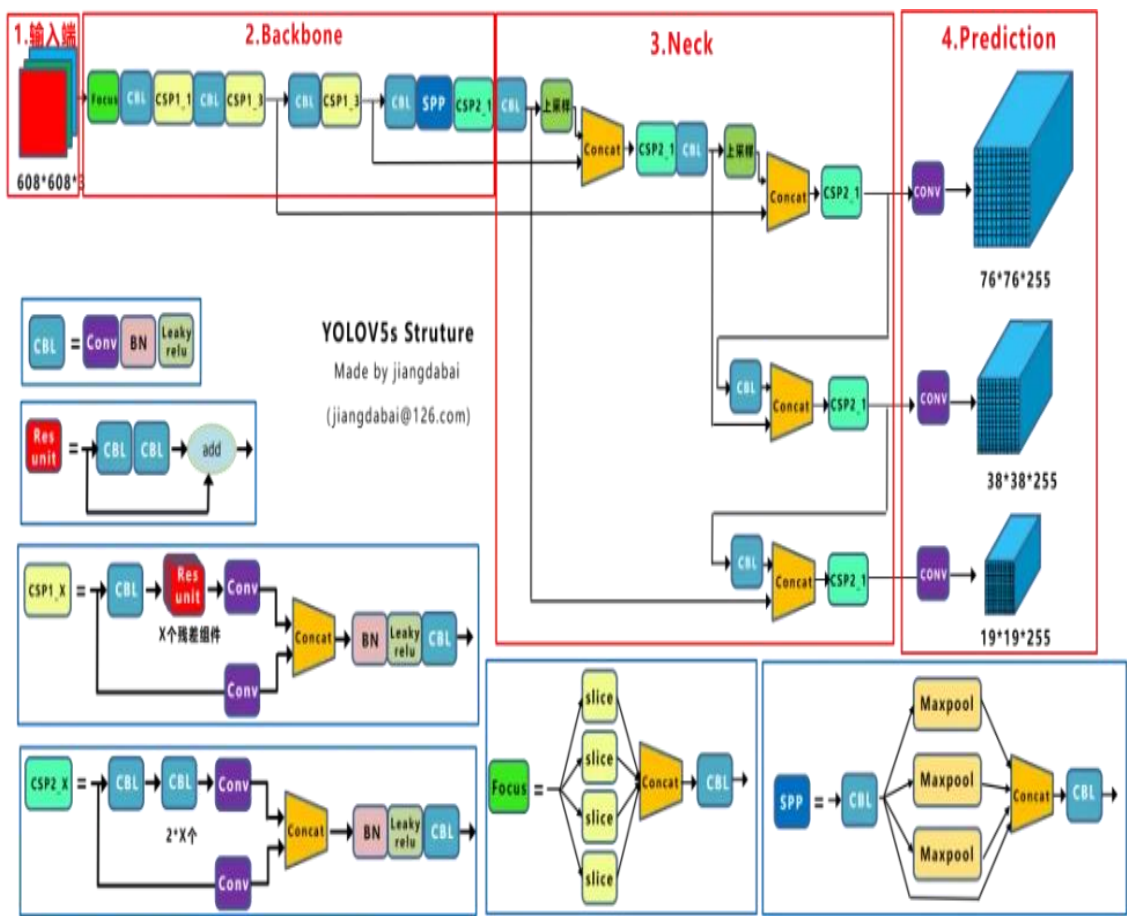


图3 YOLOv5s基本网络结构

Yolov5s网络是Yolov5系列中深度最小，特征图的宽度最小的网络。后面的3种都是在此基础上不断加深，不断加宽。

本项目第一步基于 YOLOv5 训练出 Pytorch 模型，再将 Pytorch 模型转化为 ONNX 格式便于在不同的人工智能框架中进行转换，最后采用官方的安装文件将 ONNX 转化为 RKNN 模型。

步骤分为三大模块：

- (1) 基于 YOLOv5 训练出 pytorch 模型；
- (2) pytorch 模型转化为 onnx 模型；
- (3) onnx 模型转化为 rknn 模型。

3.3 单目测距：

3.3.1 测距原理

对于单目测距，我们主要是有两个信息可以利用：图像中车辆的尺寸，图像中车辆的位置。

主要是利用图像中车辆位置进行测距的（确切地说，是利用车与路面的接触点）。如下图，A,B,C 三辆车在路面上，A 车上 p 处有个相机。有两个假设：

- 道路是平面。
- 相机光轴与路面平行。

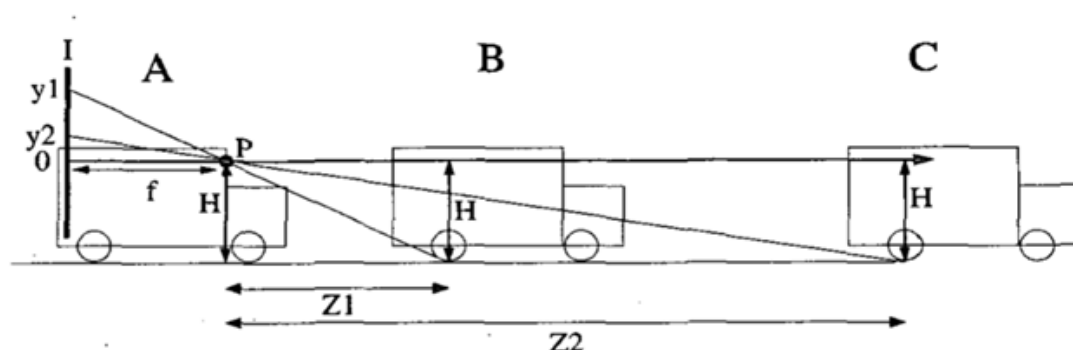


图 4 目成像示意图

道路上与本车（A）距离为 Z 的一点，投影到图像上高度为 y 。 f 为焦距， H 为相机高度，则满足：
$$y = \frac{fH}{Z}$$

为了测距，首先需要检测前车与路面的接触点，然后就可以计算距离了：

$$Z = \frac{fH}{y}$$

也就是，根据焦距、相机高度，图像中接触点位置三个量，即可计算距离。

考虑到这种测定方法会产生一定误差，且测距误差是随着距离而二次方地增长，而误差比例则随着距离线性增长。在 640x480 分辨率的图像上，90m 的测距

误差比例为 10%左右，45m 的测距误差比例为 5%左右。除了距离，相对速度或许更重要，因为它告诉我们是否在保持一个恒定的距离。所以此方法可以较为理想地解决我们的问题。

3.4 远程连接摄像头

Flask 是一个使用 Python 编写的轻量级 Web 应用框架。其“WSGI”工具箱采用 Werkzeug，模板引擎则使用 Jinja2。Flask 使用 BSD 授权。

Flask 实现视频流媒体：搭建 flask 微型网络框架，作为前端和后端之间的桥梁，从 flask 中导入`Response`和`request`模块来处理 HTTP 响应请求，使用了 python 的 yield 生成器方式，将摄像头采集实时图像一帧一帧的数据以 yield 动态获取的方式拿到然后传递给网页端。在浏览器输入 `http://localhost:666` 即可看到摄像头画面

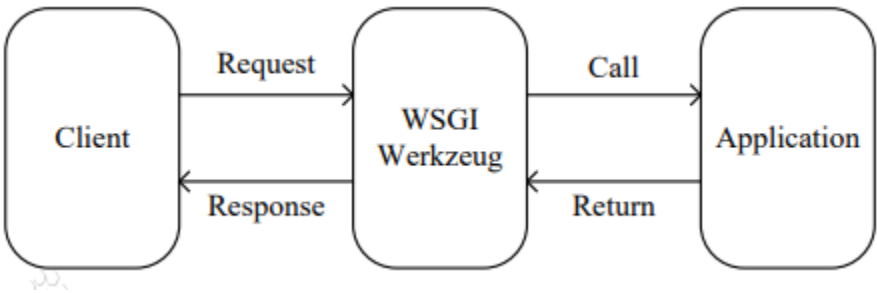


图 5 flask原理

第4章 系统实现

4.1 模型训练及 RKNN 模型转换

项目步骤流程图如图所示：

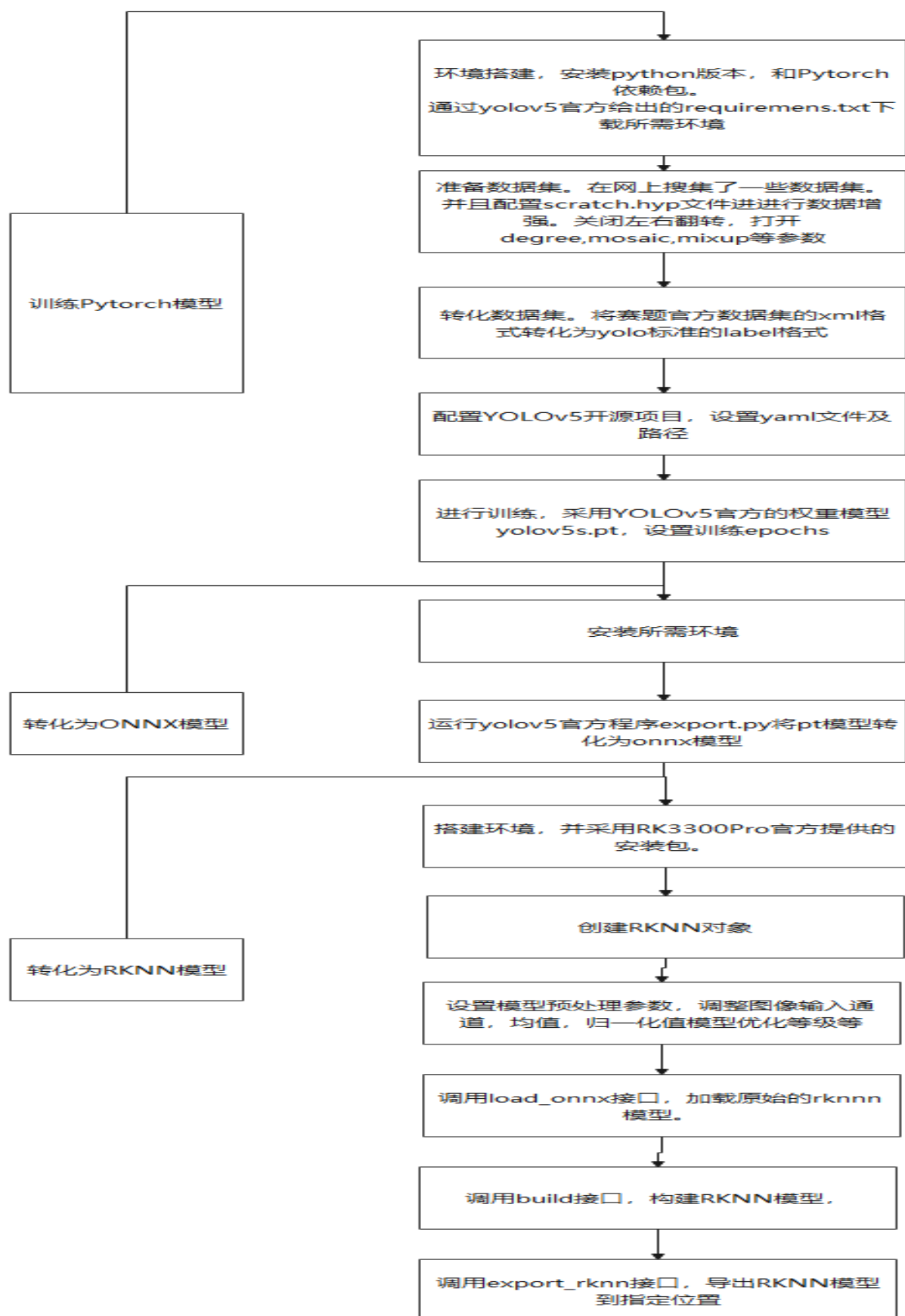


图6 模型流程图

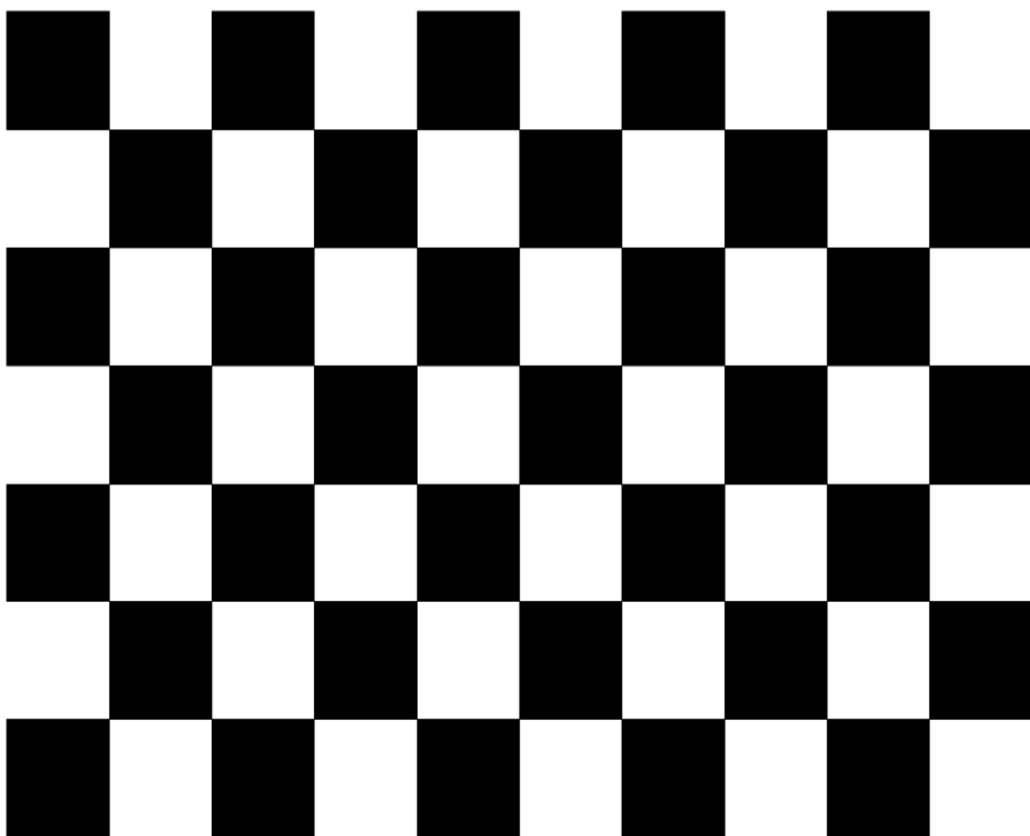


图7 棋盘格图片

4.2 相机标定

4.2.1 准备工作

我们首先使用 **MATLAB** 对相机拍摄图片进行标定得到内外参数，再使用 **OPENCV** 通过已知内外参数进行图片矫正。

使用最简单的黑白棋盘格标定板，可以直接从 **OPENCV** 官网下载得到：

使用系统所用的单目相机拍摄带有棋盘格标定板各个角度的照片，保存大概 15-20 张即可。需要测量记录好棋盘格方格的大小，在标定中会使用。按照 **OPENCV** 官网提供的图片下载打印出的方格大小大约为 25mm。

4.2.2 进行标定

使用 **MATLAB** 进行相机标定：在 APP 中找到 **CameraCalibrator** 标定应用，详细方法会在演示视频中演示

4.3 距离计算

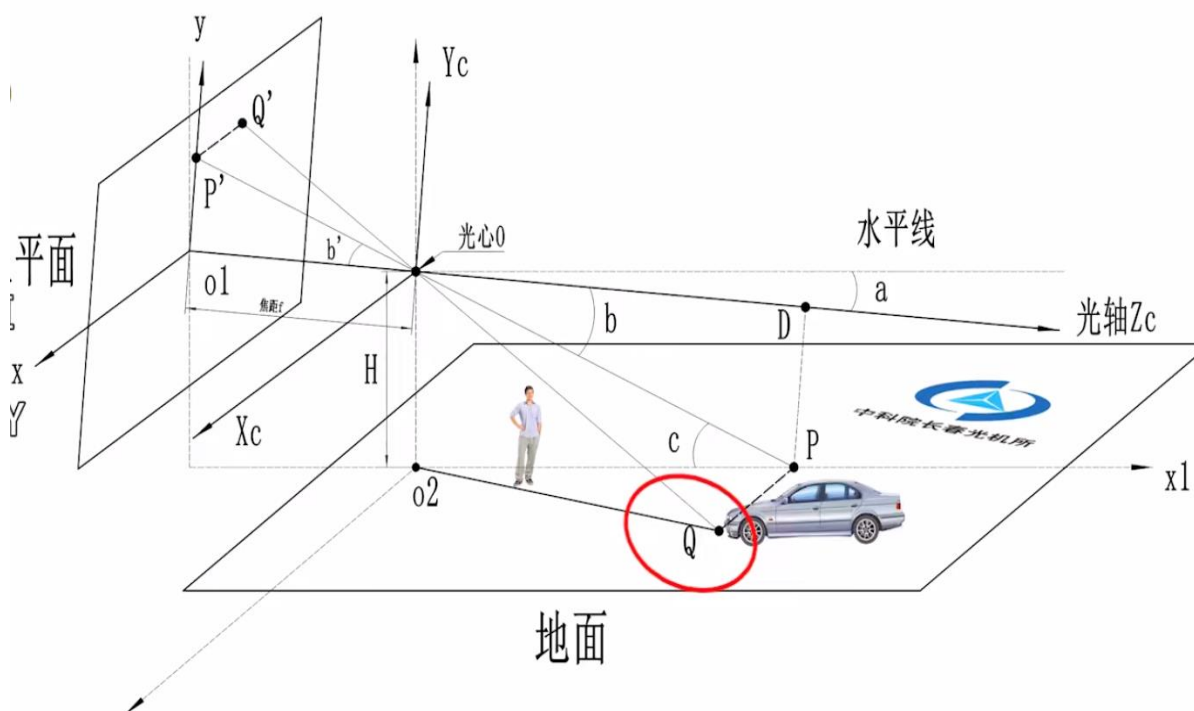


图8 成像原理图

相机模型可以简单看成一个凸透镜成像的模型。下图中， $X_c Y_c Z_c$ 是相机坐标系，其原点为光心 O ，是相机凸透镜的中心点。 $x-o1-y$ 坐标系是图像坐标系。

在 Detect 函数中调用 yolov5 检测得到目标框后,便可以提取目标框的底边的中点作为上述述的测距所需的 Q' 点。然后按照 3.3.2 中所述述原理,便可以求得得到 Q 点的 X_w 和 Y_w 坐标。取 X_w 和 Y_w 的坐标的平方和,再开根号便得到了目标的直线距离。

4.4 疲劳驾驶检测

4.4.1 眼睛轮廓曲线拟合

利用 Dlib 官方训练好的“shape_predictor_68_face_landmarks.dat”模型进行标注。输入标注人脸关键点的图像数据，先将脸提取处理，由于脸的尺寸不一，所以利用仿射变换将人脸关键点仿射到单位空间，统一尺寸和坐标系。将数据的人脸关键点做下平均，作为初始人脸形状，基于这个初始形状再

进行残差计算拟合人脸关键点。

首先在初始关键点的范围内随机采样像素作为对应的特征像素点。特征像素点选择最接近的初始关键点作为 anchor，并计算偏差。当前像素点通过旋转、转换、伸缩后的坐标系要与初始关键点(关键点的平均位置)接近，即最小化之间的距离平方，得到最优变换 tform。tform 作用于偏差，加上自身的位置信息，得到当前关键点的特征像素点。得到特征像素点后开始构建残差树，计算出当前关键点与目标关键点的偏差。通过特征像素点，利用退火的方法选择多个分割点，进行左右树划分，选择最小化划分后的偏差为最优分割点。分割样本，基于样本的平均残差更新当前关键点位置。回到上一步骤，重新选择出特征关键点，拟合下一颗残差树，最终综合所有残差树的结果得到关键点位置。

右眼轮廓曲线以 53 号和 56 号节点为界，可分成上下 2 部分，左眼轮廓曲线以 20 号和 23 号节点为界可分成上下 2 部分。 分别将左眼、右眼和嘴巴的上下 2 部分节点数据带入 轮廓曲线，计算其误差平方和，找到使误差平方和最小的函数作为轮廓拟合曲线。

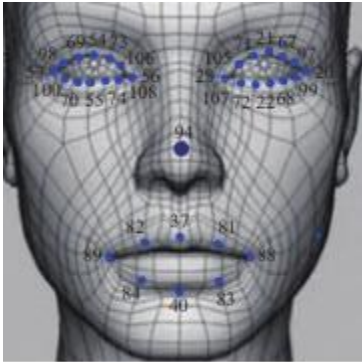


图 9 疲劳识别面部特征点分布

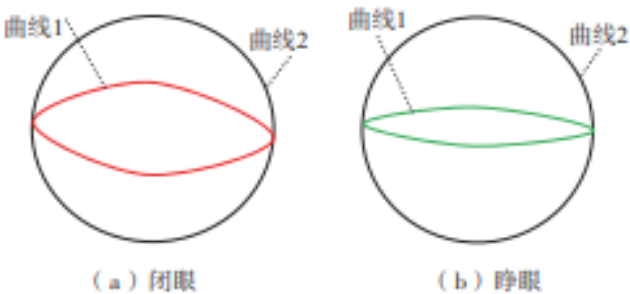


图 10 眼睛轮廓拟合曲线

由图 11 可知：眼睛在睁开状态和闭合状态下的 η_{NAE} 值差别明显，眼睛在睁开状态下 η_{NAE} 的值大于 0.25，而在完全闭合状态下 η_{NAE} 的值小于 0.2。

4.4.2 眼睛开合度归一化计算

驾驶员的疲劳状态很容易通过眼睛的形态变化反映出来，因此如何对眼睛的开度大小进行标定至关重要。对于不同的驾驶员，眼睛的开合状态差异较大，所以需要一种归一化的指标对眼睛开度大小进行标定，以提高检测的准确性和鲁棒性。由 $\eta_{NA} = A_1 / A_2$ 计算眼睛开合程度，并将其作为开度大小标定的归一化指标。

A_1 为眼睛轮廓拟合曲线所围成的面积， A_2 为其外接圆的面积大小。如图 10 中曲线 2 为眼睛在闭合和睁开状态下各特征点的外接圆。眼睛开合度归一化指标 η_{NAE} 如图 11 所示。由图 11 可知：眼睛在睁开状态和闭合状态下的 η_{NAE} 值差别明显，眼睛在睁开状态下 η_{NAE} 的值大于 0.25，而在完全闭合状态下 η_{NAE} 的值小于 0.2。

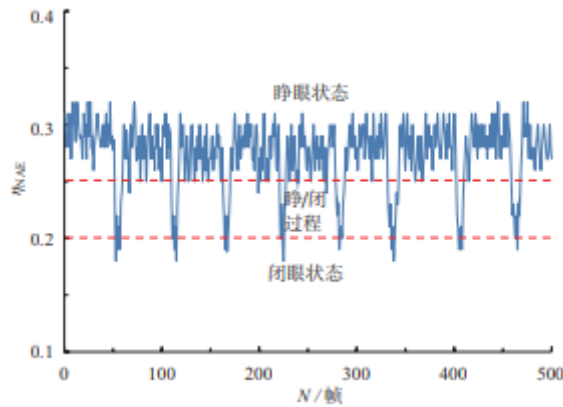


图 11 眼睛开合度归一化指标图

4.5 数据来源

数据集采用了德国 KITTI 数据集，KITTI 数据集由德国卡尔斯鲁厄理工学院和丰田美国技术研究院联合创办，是目前国际上最大的自动驾驶场景下的计算机视觉算法评测数据集。具有较高地可靠性。

4.6 数据规模

该数据集用于评测立体图像(stereo)，光流(optical flow)，视觉测距(visual odometry)，3D 物体检测(object detection)和 3D 跟踪(tracking)等计算机视觉技术在车载环境下的性能

数据集中用了图像采集模块中的部分图片，包含了七千四百张适合于问题交通场景下的图片和标签。

4.6.1 环境配置

此作品在新大陆人工智能开发板上运行，采用 Linux 系统内核。训练过程在新大陆平台上进行，训练出 Pytorch 模型。

4.6.2 测试过程

将摄像头远程连接至开发板，通过 Flask 实现视频流媒体进行传输

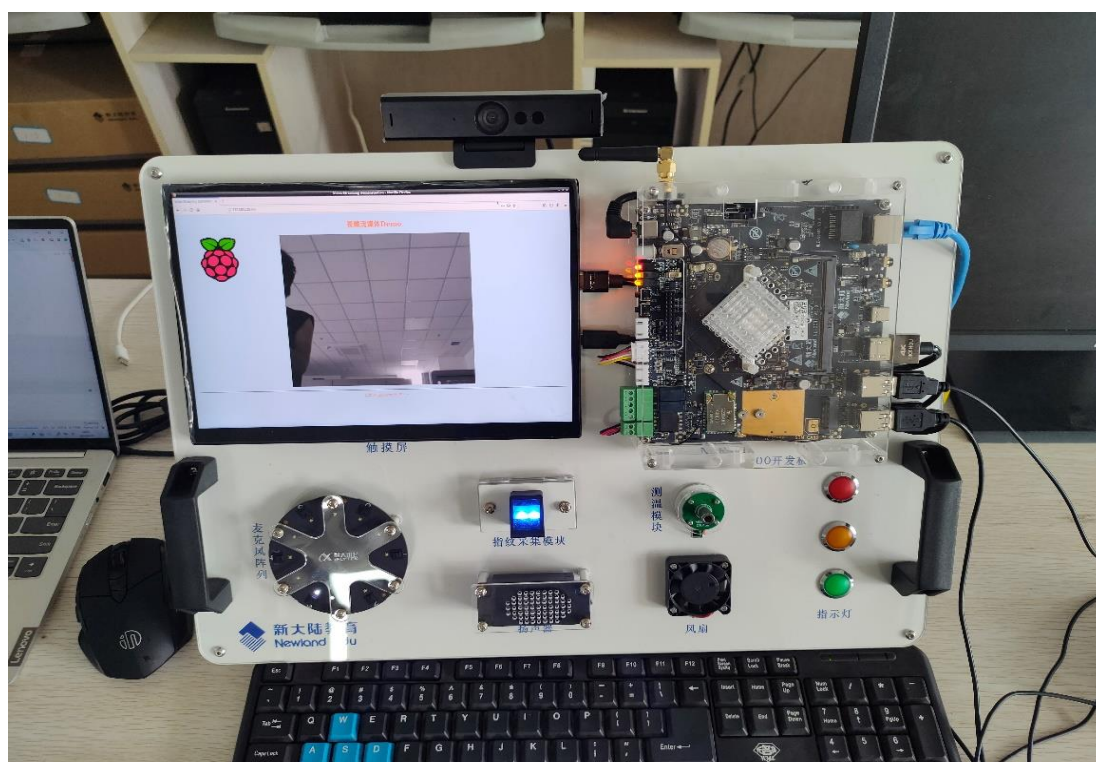


图12 开发板远程连接摄像头

用高速公路上的交通场景视频进行检测，可以对车道线进行检测，识别出前方车辆和行人等，并且对目标进行测距。



图13 实时车况检测

当速度和距离不符合标准时，结合驾驶员有疲劳驾驶行为，会进行警报，扬声器发出声音警报、计划置于车辆后方的警报灯也会亮起，对后方车辆进行警示：请注意前方车辆。检测速度在 17FPS 左右，在算力很低的边缘端也具有很强的应用性。

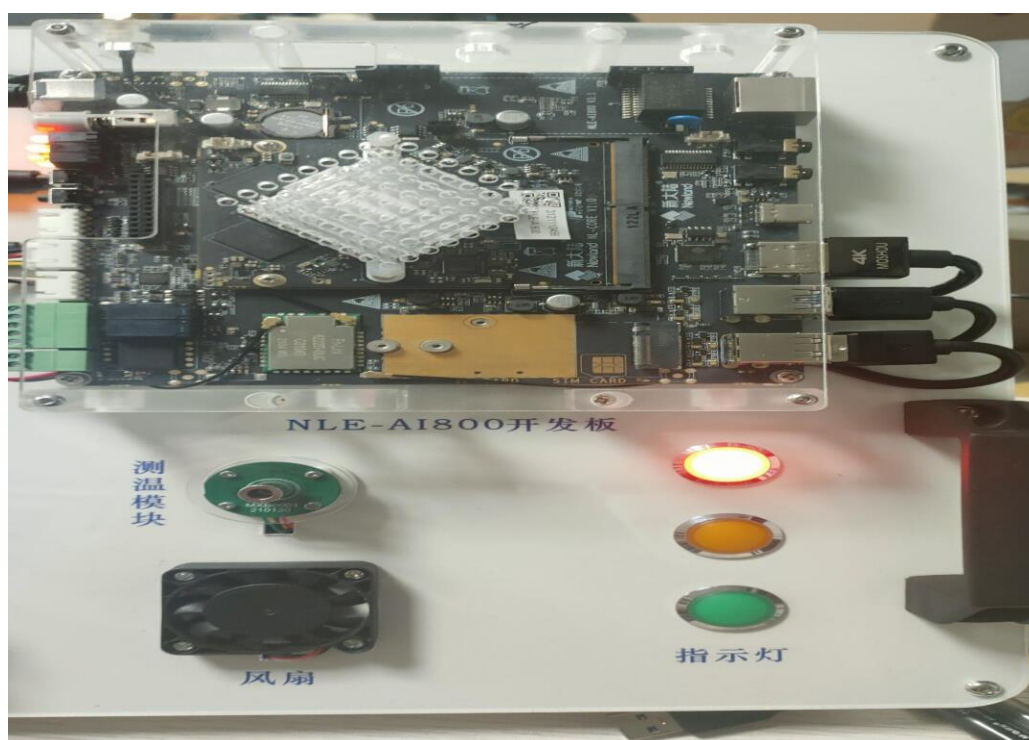


图14 警示灯与报警器

4.6.3 分析与结论

该项目在算力较为有限的边缘端有较好的应用性，检测速度较快并且在可以接受的范围内符合预期误差。

该项目可以移植于新大陆人工智能开发板，并且可以远程连接摄像头用于教学展示功能

根据车辆底部位置能准确有效地检测出车辆的位置，为单目立体视觉测距系统的下一步研究奠定了基础。

通过摄像头采集前方道路的实时路况信息，提取前方车辆在图像中的位置和距离信息，并且将提取到的前方车辆信息反馈给驾驶员，这样驾驶员可以根据反馈的信息避免潜在的危险，提高车辆行驶的安全性。而且还可以将采集的道路视频保留作为交通违章的证据，有助于减少交通违章现象，比如占用快速专用车道等，更好地发挥快速专用车道的作用。

先进驾驶员辅助驾驶系统中基于计算机视觉的应用主要是在障碍物的检测与分类，障碍物的类别包括有：车辆、行人、各种障碍物和交通指示牌等。在驾驶辅助系统中的前车防撞预警系统，其主要是检测出交通参与者中的前方车辆，首先由摄像头实时感知车辆外部环境。其次通过相关的算法实现前方的车辆检测与车距测量，从而提供相关的行驶数据供驾驶员参考。当发生车辆与前方车辆距离过近等危险工况时，系统通过信号警示驾驶员，阻止可能将发生的危险碰撞。防撞预警系统主要组成部分有前车检测模块、车辆测距模块、车辆预警模块等组成，鲁棒性强、精确性高地检测出前方车辆是保证前向安全行驶、进行危险预警和避免碰撞的基础前提。

第5章 作品总结

5.1 作品特色与创新点

根据图像检测信息准确有效地计算出车辆的位置，为单目立体视觉测距系统的下一步研究奠定了基础。

通过单目摄像头采集前方道路的实时路况信息，提取前方车辆在图像中

的位置和距离信息，并且将提取到的前方车辆信息反馈给驾驶员，这样驾驶员可以根据反馈的信息避免潜在的危险，提高车辆行驶的安全性。而且还可以将采集的道路视频保留作为交通违章的证据，有助于减少交通违章现象，比如占用快速专用车道等，更好地发挥快速专用车道的作用。

实现了实时疲劳告警的功能需求，因其设备便携度高，可以广泛应用于汽车出行。该系统围绕深度学习和物联网知识体系，具备良好的可维护性和可拓展性。经过进一步训练与改进，能够成为一个功能更加完备，人机交互更加舒适的疲劳驾驶检测系统。

5.2 作品展望

1.该作品成本很低，并且使用很方便。完全符合问题运用于边缘端算力有限的情况下。该装置尤其适用于长途车驾驶员，只需将设备摄像头固定在车辆上，将警示灯放于车辆后方。即可对驾驶员疲劳驾驶行为和前方路况信息进行分析，对司机警报和对后车进行警示。

2.此装置的分析数据也可保留用作交通事故的取证。

3.该项目可以对人工智能开发版内容进行开发，可以在不添加其他设备的情况下，加入疲劳驾驶分析和路况分析；也可以添加一个终端并对终端设备摄像头进行远程连接。

4. 该系统围绕深度学习和物联网知识体系，具备良好的可维护性和可拓展性。经过进一步训练与改进，能够成为一个功能更加完备，人机交互更加舒适的疲劳驾驶与车辆检测系统。此举将大大的减少交通事故的发生。

参考文献

[1]Gideon P.stein,Ofer Mano Amnon Shashua.Vision-based ACC with a Single Camera: Bounds on Range and Range Rate Accuracy[J].2003

[2]任述明, 向怀坤, 刘建伟,等. 基于视频图像的车速检测研究[J]. 交通信息与安全, 2007, 025(001):90-93.

[3]刘礼. 车速标示牌检测与识别的研究[D]. 天津工业大学, 2011.

单美静,秦龙飞,张会兵. L-YOLO:适用于车载边缘计算的实时交通标识检测模型[J]. 计算机科学,2021,48(1):89-95. DOI:10.11896/jsjcx.200800034.

[4]尹俊超, 刘直芳. 基于 OpenCV 的运动目标检测与跟踪[J]. 计算机工程与设计, 2011, 32(8):4.

[5] Wang J , Chen Y , Gao M , et al. Improved YOLOv5 network for real-time multi-scale traffic sign detection[J]. 2021.

[6]韦庭. 基于单目视觉的辅助驾驶系统中的图像处理研究[D]. 电子科技大学.

[7]郭晓东. 轿车前方行人识别及碰撞预警系统研究[D]. 东北大学.

[8] 黄同愿、杨雪姣、向国徽、陈辽. 基于单目视觉的小目标行人检测与测距研究[J]. 计算机科学, 2020, 47(11):7.

[9]朱艳, 谢忠志, 于雯,等. 低光照环境下基于面部特征点的疲劳驾驶检测技术[J]. 汽车安全与节能学报, 2022, 13(2):8.