



2019 年 SEU-Xilinx 国际暑期学校团队项目设计文档

(Project Paper Submission Template)

作品名称	红绿灯识别及反馈
组员姓名	吴玮琳 刘鹏晨
房间号及桌号	720-31



第一部分

小组成员分工

(各成员任务分配)

	姓名	任务分配
组长	吴玮琳	实现 BNN 算法及调试上板
组员 1	刘鹏晨	实现 YOLOv3 算法
组员 2		
组员 3		

第二部分

设计概述 /Design Introduction

(请简要描述一下你的设计：1. 功能说明；2. 所用到的设备清单)

1. 通过摄像头传输实时数据，在开发板上运行 YOLOv3 算法对红绿灯进行定位检测，再对框住的图片进行提取，使用 BNN 算法进行识别，由 PYNQ-Z2 上的 LED 灯反馈红绿灯颜色。

2. PYNQ-Z2 开发板

摄像头

HDMI 线

第三部分

详细设计 /Detailed Design

(请详细说明你作品要实现的所有功能以及如何组建系统以实现该功能，还包括为实现该功能需要用到的所有参数和所有操作的详细说明，必要的地方多用图表形式表述)

如因文档篇幅限制无法详述，可提供附件。

1. 摄像头数据采集以及视频输出：摄像头作为输入端传输视频数据进行检测，处理后的视频通过 HDMI 线传输至显示屏中，有官方提供的 Demo 可以使用；

2. YOLOv3 定位检测红绿灯：

(1) 数据集预处理：找寻数据集，使用图像处理工具将要识别的目标框出，并导出 xml 文件，使用脚本提取关键文件信息形成 txt 文件，txt 文件包含数据集图片中识别目标的位置信息以及标签信息；

(2) 下载 YOLOv3 源代码，使用 Visual Studio 2019 打开，编译生成 darknet.exe 文件，为训练测试做准备；

(3) 修改配置信息，识别类数设置为 2 类，训练图片、txt 参数文件、以及类名保存放入相应路径；

(4) 使用 YOLOv3 算法框架对其进行训练，大概在训练 3000 次左右 loss 达到 0.8，生成权重文件 yolo-obj_3000.weights；

```
current avg loss = 0.8176    iteration = 2872
Press 's' to save : chart.png -- Saved
```

Iteration number

in cfg max_batches=500200

(5) 对图片视频数据进行定位检测；

(6) 移植已完成算法到 PYNQ-Z2 板上；



3. BNN 图像识别:

- (1) 数据集预处理: 制作算法需要的数据集;
- (2) 下载 BNN 算法框架, 修改参数, 对数据集进行训练, 生成 bin 文件;
- (3) 对图片进行识别测试;
- (4) 移植已完成算法到 PYNQ-Z2 板上;

第四部分

完成情况及性能参数 /Final Design & Performance Parameters

(作品完成情况, 附照片/视频链接)

完成 YOLOv3 算法软件实现, 仅识别红绿灯, loss, 没有误报, 但会有少数漏报, loss=0.4042;
识别红灯和绿灯两类, 误报率偏高, loss=0.8176, 没有完成硬件的算法写入;
完成 BNN 算法软件和 PYNQ-Z2 实现, 识别红绿灯图片, 准确率达到 80%。
YOLOv3 视频:

第五部分

项目总结 /Conclusions

(项目中所用到的知识点, 项目收获及心得)

在本次项目中, 我们使用了 YOLOv3 和 BNN 两种神经网络算法, 前期走了许多的错路, 从最初的开始都不知道开始, 查阅各种资料, 一步一步开始往前走, 走的很坎坷, 但走的过程中通过不断的解决问题, 查阅资料也让我们受益良多, 本次项目是我们小组第一次正式接触 FPGA 开发, 深深感受到了自己的缺陷与不足, 凡是遇到 FGPGA 相关的, 都是艰难险阻, 给了我们强烈的无力感, 熬夜不可怕, 可怕的是熬了夜还是做不出来, 不过, 这强烈的无力感也让我们痛定思痛, 这次回学校一定会加强对 FPGA 开发的学习。

第六部分

源代码/Source code

(作品源码 Github 链接, github 操作步骤见 *Github 项目上传简单教程*)

<https://github.com/51184501129/Traffic-Light-Detection.git>



第七部分（技术总结/心得/笔记等分享）

（可分享的技术文档附件或已发表的文章链接）

BNN 可使用 pynq.io 里的 BNN Overlay 进行重训练。

BNN 教程：

<https://www.hackster.io/adam-taylor/training-implementing-a-bnn-using-pynq-1210b9>