



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	变色相机
组员姓名	黄婧雯 唐轩 吴欣茹 仲海兵
房间号及桌号	715 房 —6 号



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	吴欣茹	学习在 HLS 中调用 xfpencv 库，编写滤镜代码，测试并仿真；生成 ip 核并在 vivado 中综合仿真，生成二进制文件；使用 SDK 编写 main.c 函数，对项目设计的所有工程进行测试。
组员 1	黄婧雯	学习在 HLS 中调用 xfpencv 库，编写滤镜代码，测试并仿真；生成 ip 核并在 vivado 中综合仿真，生成二进制文件；使用 SDK 编写 main.c 函数，对项目设计的所有工程进行测试。
组员 2	唐轩	学习在 HLS 中调用 xfpencv 库，编写滤镜代码，测试并仿真；生成 ip 核并在 vivado 中综合仿真，生成二进制文件；使用 SDK 编写 main.c 函数，对项目设计的所有工程进行测试。
组员 3	仲海兵	学习在 HLS 中调用 xfpencv 库，编写滤镜代码，测试并仿真；生成 ip 核并在 vivado 中综合仿真，生成二进制文件；使用 SDK 编写 main.c 函数，对项目设计的所有工程进行测试。

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. 功能说明：

- 1) 数据输入为摄像头拍摄的场景，数据输出为 HDMI 输出添加滤镜后场景，实现场景自然变换。
- 2) 满足实时性要求，包括帧率及延时。

2. 所用设备清单：

Camera (OV5640)、PYNQ-Z2、HDMI Out

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. 项目要实现的功能：

- 1) 数据输入为摄像头拍摄的场景。
- 2) 数据输出为 HDMI 输出添加滤镜后场景。



- 3) 使用 HLS 及 xfOpenCV 编写处理代码。
- 4) 实现场景自然变换。
- 5) 满足实时性要求，包括帧率及延时。

2. 系统组建：

HLS 中滤镜效果的 ip 核设计、vivado 中的模块设计，综合仿真生成二进制文件、SDK 中对不同滤镜效果的调用。

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）

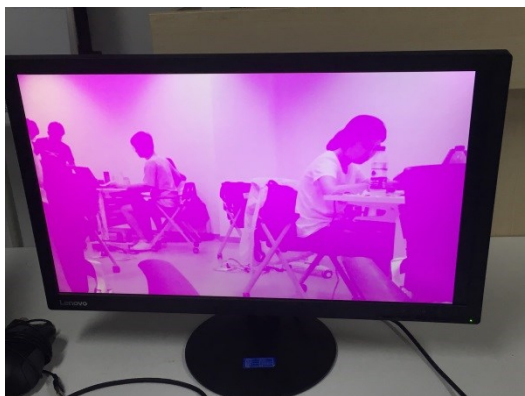
实现结果：实现了灰白，红，绿，绿色轮廓，磨皮（效果不明显）五种滤镜。原图以及加了滤镜效果的图片如下：



原图



灰白



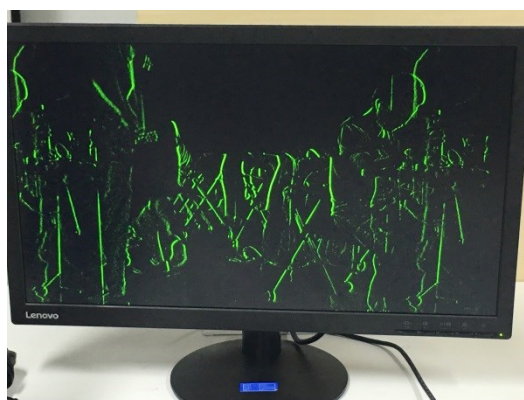
红色



绿色



黑板画



鬼脸滤镜

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)

1. 所用到的知识点：

HLS 处理；xfopencv 库；SDK 的运用

2. 项目收获：

学习了 HLS 如何调用 xfopencv 库，编写滤镜代码，实现了在电脑中读取图片并生成仿真结果，并生成了 ip 核；学会了使用 SDK 编写 main.c 函数实现滤镜的功能。

3. 心得：

我们对项目的实现方法有以下三种思考：

1) opencv+cmake+visual studio

一开始我们完全不了解 HLS 和 opencv 的相关知识，所以在查阅了相关资料后，发现了相对较多的关于使用 opencv 的资料，于是我们大胆提出了 opencv+cmake+visual studio 的



实现方法。经过对方案 1 更深入的了解和思考后，我们认识到了这种方案的实现难度。又由于回忆到了之前

- 2) jupyter notebook+python +opencv +USB 摄像头（帧率太低，无法实现项目的实时要求）

由于回忆到了之前的讲座“融入 python 生态的 zynq 软硬件设计框架”，我们又提出了方案 2: jupyter notebook+python +opencv +USB 摄像头。咨询老师后，老师提出此种方案帧率太低，无法实现项目实时性要求的建议。

- 3) HLS+xfopencv

- 4) 最后在老师的建议下，我们最终选择了方案 3: HLS+xfopencv，积极查阅相关资料进行学习和讨论。

通过这次做项目的经历，我们又一次体会到了团结协作的重要性。三个臭皮匠，顶个诸葛亮，人难免有疏忽大意的时候，如果两个人一起合作，互相提醒，就会有效避免此类问题的发生。另外，我们也明白了拥有一台性能好，速度快的电脑对软件学习和项目制作是多么重要。

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/wuxinruru/HLS_xfopencv.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）

项目完成进程：

7.12

查找 xfopencv 和 opencv 的资料；

opencv 里关于图片滤镜代码的研究；

学习怎样在 HLS 里调用 xfopencv 并生成 ip 核；

修改 xfopencv 的环境 运行 xfopencv 的例程。

7.13

完成了 camera_in_hdmi_out 的 demo 仿真；

完成了 opencv 和 xfopencv 的图片滤镜算法的仿真；

了解了整个项目重点为 ip 核的生成及并列；

完成了后续项目进展计划。

7.14



在电脑上仿真测试了各种滤镜的效果，实现了在电脑中读取图片并生成仿真结果。

7.15

中期验收。了解了 ip 核如何生成，以及在 sdk 中如何编写 main.c 函数。

7.16

实现了各种滤镜的 ip 核的生成，并进行了测试。

7.17

利用 sobel 算法实现了边缘检测滤镜的设计；调试不同滤镜，准备验收文件，编写验收文档。