**潘多拉——基于PGL50H的多媒体视频处理魔盒**

袁华阳；傅佳木；戴晨康

# 设计概述

## 设计目的

随着图像处理算法和FPGA硬件的不断发展，有越来越多的图像处理算法使用FPGA平台实现。为了达到让多种图像处理算法能够在硬件平台上更加直观的展示的目的，本作品以国产紫光FPGA硬件平台为载体，整合了图像的拼接、融合、去噪、缩放、调整亮度色度等在内的一系列图像处理算法，实现一个能够运行多种图像处理算法的原型验证和效果展示的魔盒。

本作品设计了一系列的图像处理功能模块，并综合在一起，以达到预期功能。

## 应用领域

本作品以紫光FPGA平台为载体，允许开发人员根据特定应用需求设计定制的处理流程和算法并加以展示。本作品具有高度可定制化、并行处理能力强、低延迟等优势，适用于许多图像处理应用的领域，FPGA的并行性和低延迟使其能够处理实时图像数据，适用于自动驾驶、视频监控、工业视觉等需要快速响应的领域。

## 主要技术特点

硬件部分采用紫光同创Pango PGL50H6FBG484，OV5640双目摄像头模块， HDMI接口，以太网接口。

本作品能够实现以下功能：

1. OV5640双目摄像头的视觉融合。

2. 两路HDMI图像拼接整合。

3. 对拼接后图像进行缩放。

4. 对摄像头图像进行亮度、色度的调整。

5. 对摄像头图像进行去噪。

6. 通过以太网传输字符并显示。

## 关键性能指标

1. OV5640单个摄像头分辨率采用320\*720@30FPS，双目融合后分辨率为640\*720@30Hz，作为第一路视频源。

2. 通过以太网传输单个字符并使用UDP接口进行传输，通过OSD叠加显示。

3. 第二路视频源采用电脑自带的HDMI输入，输入分辨率为1920\*1080@60Hz。

4. HDMI输出分辨率为1280\*720@30Hz。

## 主要创新点

(1) 实现基础部分的所有功能。

(2) 实现双目摄像头融合去噪功能。

(3) 实现按键调整图像亮度和色度。

(4) 实现通过网口输入字符并叠加显示到拼接处理后的图像中。

(5) 实现通过按键控制图像的旋转180度。

(6) 实现按键控制图像的缩放。

# 系统组成及功能说明

## 整体介绍

给出芯片或系统整体框图，各子模块标注清楚，并进行整体的文字说明，需要表达出各模块之间的关系。

我们所使用的 FPGA 型号为 PGL50H6FBG484，属于紫光同创公司的 Logos 系列产品。

系统一共有三个输入端，一个输出端。三路输入分别为：一是HDMI\_IN输入：接收PC端视频；二是以太网输入：接收以太网口传输来的字符；三是双目摄像头输入：接收双目摄像头视频。一路输出为：HDMI\_OUT：将FPGA处理后的视频输出。

核心为紫光同创PGL50H FPGA平台，将第一路输入（HDMI\_IN）和第二路输入（双目摄像头）数据流进行图像拼接，同时进行图像处理，去噪，旋转，缩小，亮度和色度调节之后生成视频流。第三路输入（以太网口），将字符叠加到画面上生成最终的视频流，最终通过输出HDMI\_OUT输出显示到外接显示屏上。

## 各模块介绍

根据总体系统框图，给出各模块的具体设计说明。

### OV5640双目摄像头的初始化配置

OV5640 双目摄像头模组采用美国 OmniVision(豪威)公司的CMOS图像传感器 OV5640，支持自动对焦的功能。OV5640 芯片支持 DVP 和 MIPI 接口。

OV5640 的寄存器配置是通过 FPGA 的 I2C（也称为 SCCB 接口）接口来配置。需要正确地配置寄存器， 使OV5640 输出我们需要的图像大小和格式，OV5640的配置时序如图 1所示。为适应HDMI输出和双目摄像头融合的需要，我们将OV5640配置为640\*720的图像输出。

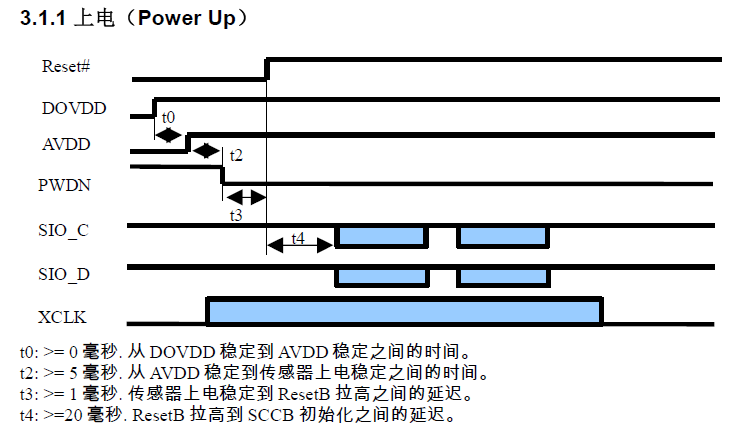


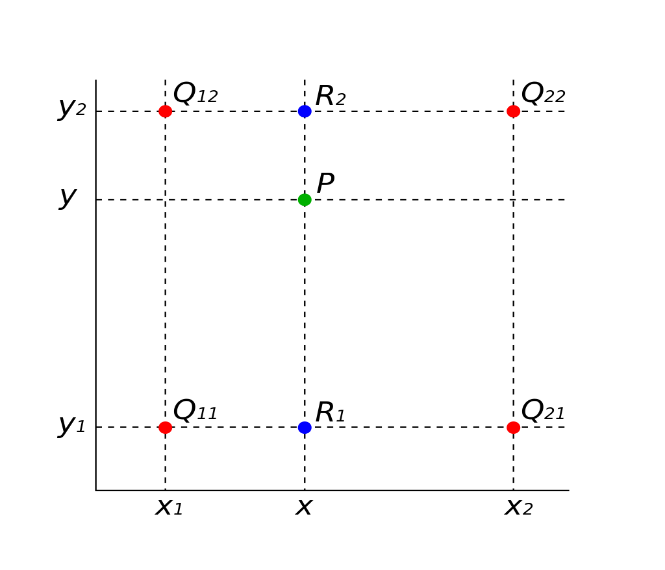
图 1 OV5640配置时序

### 图像缩放方案

在原图显示的基础上，我们添加了图像缩放的功能，能将原始画面缩小到原来的四分之一（图像宽高各为原图的一半），下面具体分析算法如下：

对于图像缩放，方案采用**双线性插值算法：**

算法原理如下：



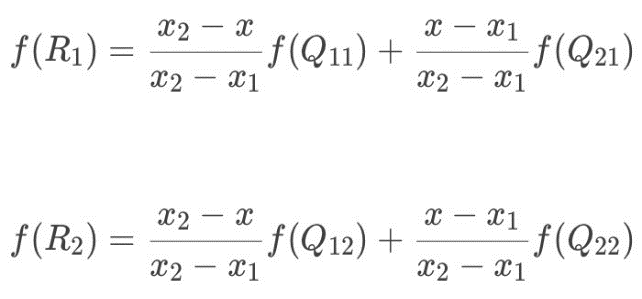


图 2 双线性插值原理

原图像数据从HDMI输入

对于算法的数据缓存，方案采用两个RAM，地址宽度均为10，数据位宽24（RGB888）：ram0和ram1，同时对它们进行分页page0和page1，用于缓存四行像素，地址的最高位为0时为page0，地址的最高位为1时为page1；ram0的page0和page1分别存储第1行和第3行，ram1的page0和page1分别存储第2行和第4行。像素数据逐行逐个按顺序输入，每输入两个像素进行一次X方向的线性插值，并将算好的像素存入ram中，同时对完成第一次插值的像素进行计数，当计数值达到目标图像的宽度时，一行的第一次线性插值结束。当两行的第一次线性插值完成后，结果均保存在ram中，第二次线性插值将读取ram 中的数据进行第二次线性插值。

对于算法的运算流程，采用二级流水线结构，先对输入的第1，2行进行第一次插值（x方向上的插值），1，2行完成第一次插值之后，紧接着3，4行的第一次插值运算开始，同时进行1，2行的第二次插值（y方向上的插值）。第二次线性插值运算所得像素均为有效像素，按运算完成的顺序逐个输出。

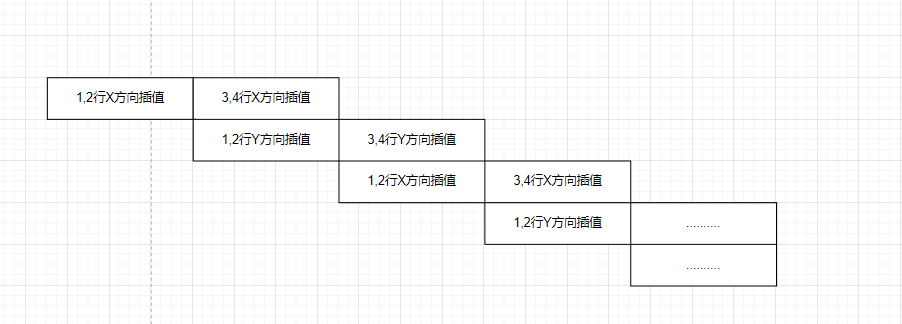


图 3 双线性插值运算二级流水线结构

# 完成情况及性能参数

具体完成的情况，可图文结合，具体的性能参数等量化指标。

## 对两路输入视频源进行拼接

## 对拼接后的图像进行缩小

## 实现180度的图像旋转

## 双目摄像头去噪

## 通过按键调整图像的亮度和色度

## 双目摄像头融合

## 通过网口输入字符，将字符叠加到图像中

## 将凭借后的图像通过HDMI实时显示输出

# 总结

## 可扩展之处

### 实现图像任意比例缩放：目前只实现了对图像一个比例的缩小，在以后的改进中，将扩展任意比例的图像缩放。

### 实现更多视频图像处理操作：例如图像阈值分割，灰度转换，形态学操作等。

### 图像处理算法的优化：目前方案中所采用的部分图像处理算法仍有优化空间，双线性插值算法可以进行数据缓存方案的优化，可以实现加快运算的速度。

### 实现图像任意角度旋转：目前只实现了对图像进行180度的旋转。

## 心得体会

首先感谢大赛组委会提供这次宝贵的比赛机会，感谢紫光同创官方对本次比赛的大力支持。通过本次的FPGA竞赛，我们深刻领略到了FPGA的魅力，仅凭一块板子就能完成许多非常有意思的图像处理操作。

在FPGA图像处理的实践中，我们发现，相比于计算机软件对于图像的处理，或许有时候并不需要对图像处理原理的细节有很深的了解，也可以通过调用库函数或者简单地几行高级语言代码完成，但是在FPGA中想要完成图像的处理，如果编码者对相关的图像处理算法的细节不熟悉的话，编码的工作将寸步难行。这其中不仅涉及到每个像素的RGB格式（常用的有RGB888,RGB565），不同的RGB格式的像素的数据位宽不同，导致运算过程也会有所差别，而且即使对于同一种图像处理操作，不同的算法的运算方式有其不同的特点，因此我们还要根据其特点去针对性的设计数据缓存方案，是用RAM，还是FIFO？如果用RAM，那应该用几个合适？是否需要对其进行分页操作？数据的读写地址该按照什么规则去计算出来？等等一系列的问题需要我们仔细考量。而以上这些需要我们仔细考虑的问题，在计算机软件中，大部分不需要我们特别关注。但是也正是因为FPGA这个特点，使得我们对一些图像处理算法的细节相比于没接触FPGA之前有了更深入的理解。

同时，通过动手实践，使得我们所学的知识不只停留在学校课堂中或者课本上所教授的理论上。在实践中，反复的调试、一次又一次地修改是常态，也正是因为这占据了绝大部分时间的调试才使得我们距离最终的功能实现越来越近。

总而言之通过这次比赛，我们受益匪浅，不仅团队协作能力有了极大的提高，而且工程实践能力也都有了很大的进步。同时，FPGA的前景也给予我们极大鼓舞，希望以后也能为我国的FPGA行业的发展贡献自己的力量。

最后，再次感谢大赛组委会以及紫光同创公司对本次比赛的大力支持。希望今后的全国大学生FPGA创新设计竞赛的越办越好。

# 参考文献

[1] [美]冈萨雷斯. 数字图像处理[M]. 阮秋琦，译.北京：电子工业出版社，2011.

[2] 阎石. 数字电子技术基础[M]. 北京：高等教育出版社，1998.

[3] 夏宇闻 ，韩彬. Verilog数字系统设计教程（第4版）[M]. 北京：北京航空航天大学出版社，2017.

[4] 牟新刚. 基于FPGA的数字图像处理原理及应用[M]. 北京：电子工业出版社，2017.

# 附录

重要代码、推导过程等不便于在正文中体现的内容