**基于Zynq的边缘滤波实现**

梁盛喜，王誉珩，刘嗣隆

(1. 电子科技大学, 成都 400060)

摘 要： 介绍了一种基于Zynq的边缘滤波的实现算法，首先基于Sobel算子对单张图片进行卷积滤波，后通过腐蚀与膨胀算法对噪声进行过滤，最后将得到的结果通过HDMI模块转接到HDMI显示屏实现单张图像的滤波。在功能实现的基础上，我们通过对滤波器中加法器的优化实现了更高效的卷积滤波并通过ASIC综合结果对两个滤波器进行比较。最后利用OV5640摄像头实现了Zynq中PL与PS端的数据交互，并实现了对图像的实时边缘检测。

关键词：边缘检测；FPGA；Zynq；Sobel算子；卷积滤波

Implementation of edge filtering based on Zynq

WANG Yuheng, LIU Silong, LIANG Shengxi

(1. University of Electronic Science and Technology of China, Chengdu 400060)

Abstract：This paper introduces an implementation algorithm of edge filtering based on Zynq. Firstly, the convolution filter based on the Sobel operator, and the noise is filtered by the corrosion and expansion algorithm. Finally, the resulting results are transferred to the HDMI module to the HDMI screen. Based on the functional implementation, we achieve more efficient convolution filtering by optimizing the adder in the filter and compare the two filters by ASIC synthesis results. Finally, OV5640 camera is used to realize the data interaction between PL and PS in Zynq, and realize the self-detection of real-time edge of images.

**Key words**： Edge detection; FPGA; Zynq; Sobel operator; convolution filtering

0 引 言

索伯[算子](https://so.csdn.net/so/search?q=%E7%AE%97%E5%AD%90&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/qq_55025358/article/details/_blank)（sobel operator）常用于边缘检测，在粗精度下，是最常用的边缘检测算子，以广泛应用几十年。sobel算子由两个3X3的卷积核构成，分别用于计算中心像素邻域的灰度加权差。分为垂直方向和水平方向的索伯滤波器Gx and Gy。

[sobel](https://so.csdn.net/so/search?q=sobel&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/qq_55025358/article/details/_blank) 算子的用途主要为：边缘检测时： Gx用于检测纵向边缘,，Gy用于检测横向边缘。计算法线时：Gx用于计算法线的横向偏移，Gy用于计算法线的纵向偏移。

在摄像头实时图像滤波的实验中，我们使用了Zynq的PL与PS端进行数据交互，而我们使用的摄像头是OV5640型号。在该实验中我们调用了如下IP：VDMA，AXI VIDEO INTO AXI-Stream视频流AXI-Stream INTO VIDEO OUT：AXI-Stream转视频流VIDEO TIMING CONTROLLER:配置视频流控制信号。

文一律五号宋体

[sobel](https://so.csdn.net/so/search?q=sobel&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/qq_55025358/article/details/_blank) 算子的核心计算单元为3X3的卷积结构，传统的滤波计算速度慢，功耗高，占用面积大，我们需要更高效的滤波运算结构实现边缘检测滤波器的优化。

文章第1节主要介绍单张图片的Sobel边缘滤波基本原理；第2节介绍卷积运算结构优化设计；第3节介绍；第4节介绍基于OV5640的Sobel实时边缘滤波。

1 单张图片的Sobel边缘滤波的基本原理

1.1 Sobel滤波算子与基本原理

sobel算子应用时进行给定图像的卷积操作，卷积为计算图像大矩阵周围像素和滤波器矩阵对应位置元素的乘积, 然后把结果相加到一起, 最终得到的值就作为该像素的新值, 这样就完成了一次卷积，之后继续移动卷积核，直到把大矩阵每个位置都运算完毕，因为相邻像素卷积结果一般具有相似输出，会产生大量冗余信息，一般为减少输出值会进行求取最大小值或者平均值的池化操作，Sobel卷积算子如图1所示。

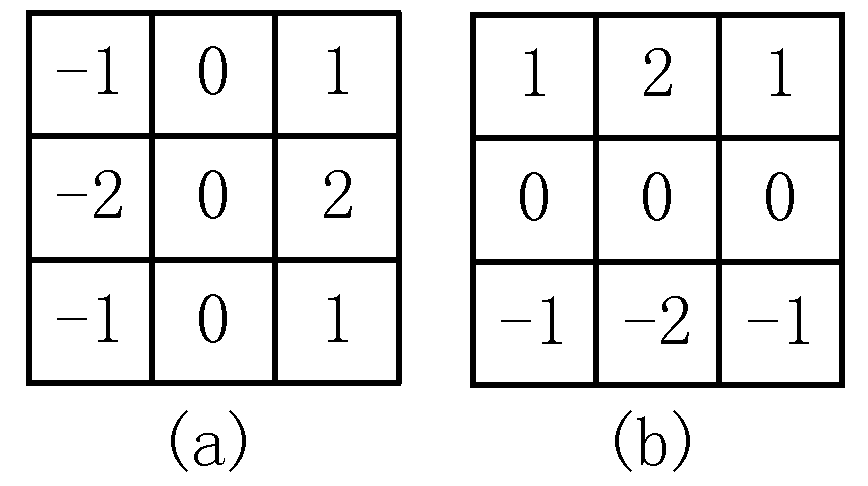


图1 Sobel卷积算子

1.2 Sobel边缘滤波的系统实现

实验一的顶层设计由以下四个IP组成：

CLK\_WIZ:MMCM的时钟分频器，为HDMI接口提供时钟。

Video\_drive：视频流驱动IP，用于用于配置视频流LCD(VGA)的主要的所需的控制时序信号，如HSYNC,VSYNC,clken等信号，其中VSYNC为场同步信号，用于使能某一帧的信号指示，HSYNC为行同步信号，用于使能某一行的信号指示，clken为数据的使能信号。

HDMI\_interface:LCD转HDMI接口，用于将LCD协议转HDMI协议。

Video\_display:图像处理IP，内部定义了一个ROM的IP用于存储一张200x200分辨率图片的COE文件，ROM中的图像数据会经过以下三层图像处理设计。

整个滤波系统的数据流是由Video\_drive模块LCD控制信号与像素的位置信息，发往Video\_display模块，在该模块内部会对ROM存储器中的内置图片实现滤波操作并将滤波结果发往Video\_driver模块，通过HDMI接口转成HDMI协议传往HDMI显示屏，顶层设计见图2。

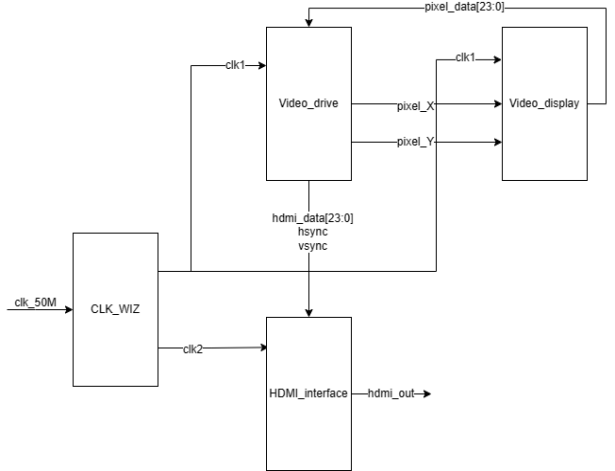
****

图2 顶层设计

Video\_display模块内部的图像处理流程如图3所示，首先ROM内部的单张图片先经过Sobel滤波，并通过二值化后滤波的结果会存在白色噪声，需要经过腐蚀算法滤除白色的散点噪声，由于腐蚀操作会使边缘的信息受到腐蚀，所以仍然需要膨胀算法对边界信息进行膨胀操作。

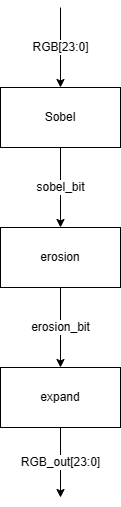


图3 图像处理流程

Sobel模块的数据扫描如图4所示，基HDMI与LCD的协议原理，整个图像的扫描首先由场同步信号使能，使能后的场同步信号后系统从第一行的第一个像素开启扫描，扫描到最后一个像素后切换到第二行信号继续进行扫描，直到所有的像素都被扫描完毕。

根据显示屏的像素扫描原理，我们可以得到当像素扫描到第三行的第三列时才可以进行卷积操作，因为这时我们才能获得3X3卷积所需的所有像素信息，9个像素分别来自line1(寄存前前一行的像素)，line2(寄存前一行的像素)，p0(寄存前前一个数据)，p1(寄存前一个数据)和当前像素数据。

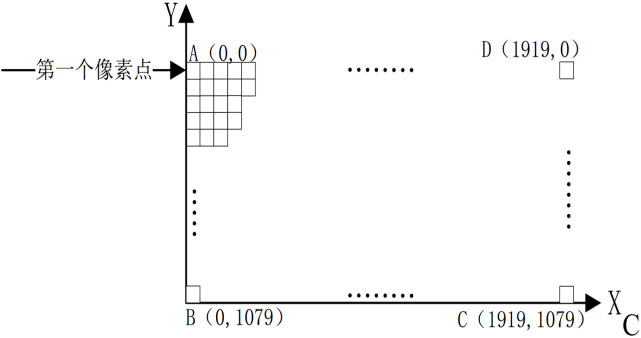


图4 图像扫描原理

滤波结果如图5所示，在Zynq的PL端的ROM存储了一张200x200的分辨率图片，可以看出最后的滤波结果非常明显，并且在噪声的滤波效果也很好。

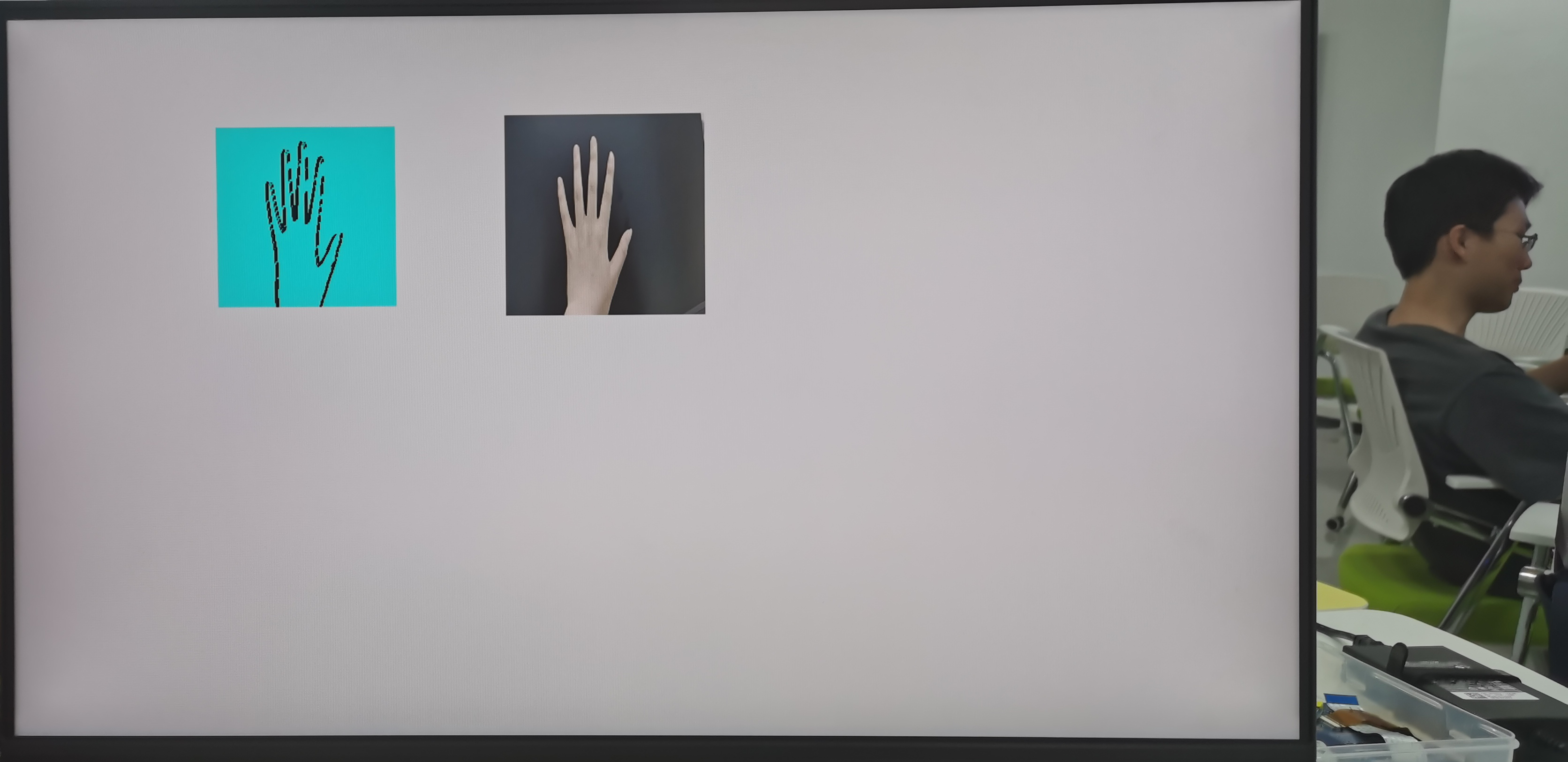


图5 手掌的边缘检测结果

2 卷积运算结构优化设计

2.1 卷积运算结构优化设计原理

Sobel卷积运算：num1[7:0]+2\*num2[7:0]+num3[7:0]

-num4[7:0]-2\*num5[7:0]-num6[7:0]

Sobel卷积运算结构为6组8位无符号二进制数的加法运算。首先将多组数中的减数取补码，利用Wallace结构通过全加器将多组数压缩至两组数，最后通过CLA超前进位加法器得到结果。

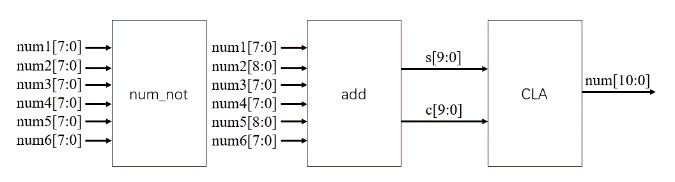


图5 卷积运算结构顶层设计

图5中共3跟模块，首先通过num\_not处理初始数据，将减数num4、num5、num6按位取反，并将num2，num5左移1位；再通过add将6组数压缩为两组；最后通过CLA得到结果。

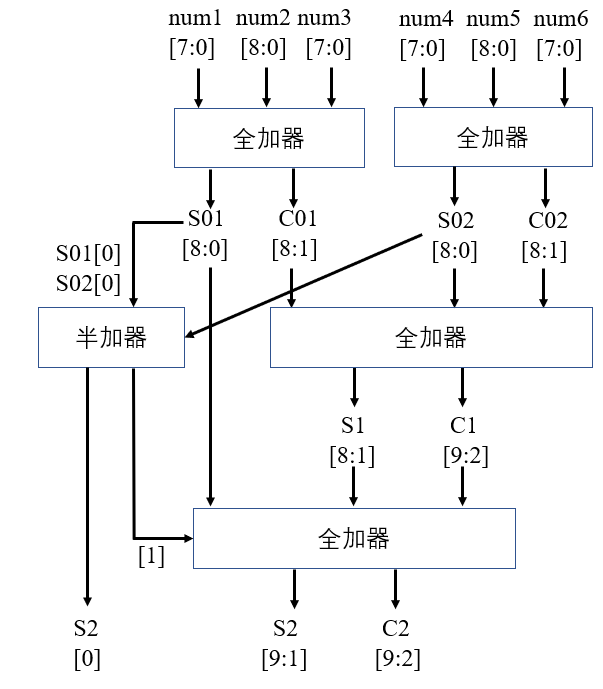


图6 add压缩模块结构设计

图6中add模块中共3级压缩。第一级为2组8位加法器级联的3-2压缩，将num1、num2、num3、num4、num5、num6分别压缩为S01，C01，S02，C02，其中[8:1]为数的权重；第二级为一个半加器和8位加法器，8位加法器将C01，C02以及S02[8:1]进行压缩，半加器则把S01[0]和S02[0]单独运算，目的是减少下一级待压缩的数位宽，实现8位3-2加法器的结构复用，同时减少后续加法的位宽。得到S1，C1；最后将S1，C1[8:2]，S01[8:1]以及半加器的输出[1]进行压缩，得到S2，C2。最后得到结果为S[9:0]，C[9:2]，再通过8位CLA结构加法器计算，并将加法进位置1，实际上是+4，实现取补码操作中的加一。

3 滤波器ASIC综合结果分析

3.1 滤波器模块版图

我们成功地得到了基于Sobel算子的卷积滤波模块的电路版图。版图设计符合工艺规则，通过了DRC和LVS验证。为了方便观察，我提供了整体电路板图，时钟线，标准单元三幅图。

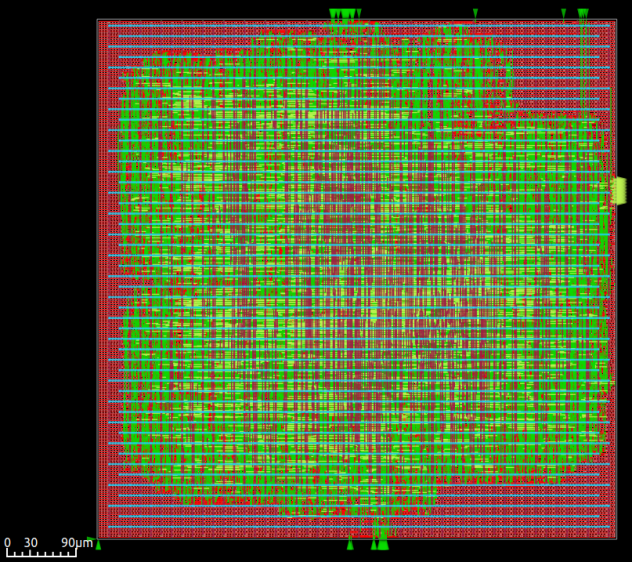


图9 电路版图（包含信号，时钟，电源，所有单元）



图10 时钟线

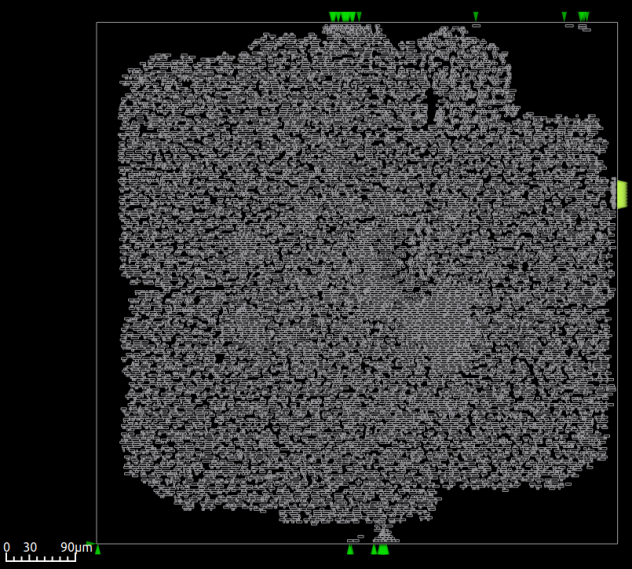


图11 基本单元

3.2 滤波器模块面积分析

我们在25MHz的工作频率下成功地得到了基于Sobel算子的卷积滤波模块的电路版图，优化后的总设计面积为212265 u²，利用率为47%，总体上该设计面积比较合理。

对于加法器部分，优化前的设计面积为1460 u²，利用率为53%，经过优化后，设计面积缩小至1231 u²，利用率为51%。优化后的加法器设计面积减小，利用率也有所下降，热分布更加合理，说明我们的优化工作是有效的。

3.3 滤波器模块功率分析

在25MHz的工作频率下，优化后的滤波器模块的总功耗为8.16mW，其中内部功耗4.90mW，占60.1%，开关功耗3.26mW，占39.9%，静态功耗较小，为7.83e-05mW。这是一个比较均衡的功耗分布，说明设计在功耗管理方面做得较好。

对于滤波器中的加法器部分，优化前的总功耗为0.490μW，内部功耗 0.243μW占49.5%，开关功耗0.248μW占50.5%，静态功耗较小4.88e-4μW。优化后的总功耗减小至0.236μW，内部功耗0.126μW占53.5%，开关功耗0.110μW占46.5%，静态功耗较小，为5.02e-4μW。优化后的加法器功耗明显下降，内部和开关功耗比例更为均衡。

3.4 滤波器模块延时分析

在25MHz的工作频率下，优化后的滤波器模块的最大延时为19.27ns，满足了工作频率的要求。

对于滤波器加法器部分，优化前的最大延时为12.97ns，优化后的最大延时为11.18ns。

总体来说，优化后的加法器设计在时序、面积和功耗上都得到了显著的改善，说明这次优化是成功的。

表1 优化前后加法器性能对比(25 MHz下)

|  |  |  |
| --- | --- | --- |
| 参数 | 优化前 | 优化后 |
| 关键路径延时/ns | *12.97* | 1.28 |
| 总功率/10-5W | *4.90* | 2.36 |
| 面积/u^2 | *1460* | 1231 |

4 OV5640实时滤波的实现

4.1 OV5640实时滤波设计

为了实现对摄像头的数据进行实时滤波与Sobel边缘检测，我们在相关例程的基础上加上 Sobel 边缘检测算法的模块，实现了对摄像头数据的实时滤波处理，整个顶层模块见图6。



图6 OV5640顶层模块

在摄像头实时图像滤波的实验中，我们使用了Zynq的PL与PS端进行数据交互，而我们使用的摄像头是OV5640型号，整个数据流如下:首先Zynq端会初始化一个GPIO口，该GPIO会以SCCB总线的协议往摄像头的寄存器进行配置初始化，初始化之后，摄像头的数据流信息会经过PL端的OV5640端口实现对是控制信号的延迟(因为摄像头的初始数据不稳定)，之后视频流数据通过VIDEO INTO AXI-Stream的IP模块将视频流数据转化为AXI-Stream协议的数据，之后同股票VDMA的IP模块将AXI-Stream协议转化为AXI协议后通过PL与PS端的GP口进入PS端，并将数据从DDR接口存储到DDR，DDR内的数据同样需要在GP口传出PS端，通过VDMA与AXI-Stream TO VIDEO OUT的IP模块将AXI数据流转化为视频流并通过HDMI接口发往显示屏。

4.2 实现过程的问题

在整个实验过程中困扰我们的主要问题是以下两个：

第一个问题是在OV5640的图像显示中对与VIDEO INTO AXI-Stream IP与Sobel处理数据IP直接的连线出现问题，这是由于两个IP之间的接口信号线的个数不对称，最后我们通过了网上查找相关的资料解决了。

第二个问题是显示屏显示图像出现了重叠：



图7 图像出现重叠

这是因为我们在Sobel滤波的行寄存器的配置时忽略了对地址位宽的修改，由于显示器的分辨率是1280，需要我们寄存器的地址位宽为11以上，而初始位宽为10，导致显示出现显示的重叠。

5 实验结果

实验一：实现单张图片的Sobel边缘检测如下所示，实现了对手的滤波，最终的滤波效果非常好，没有明显的噪声点，并且边缘信息得到了完整提取。

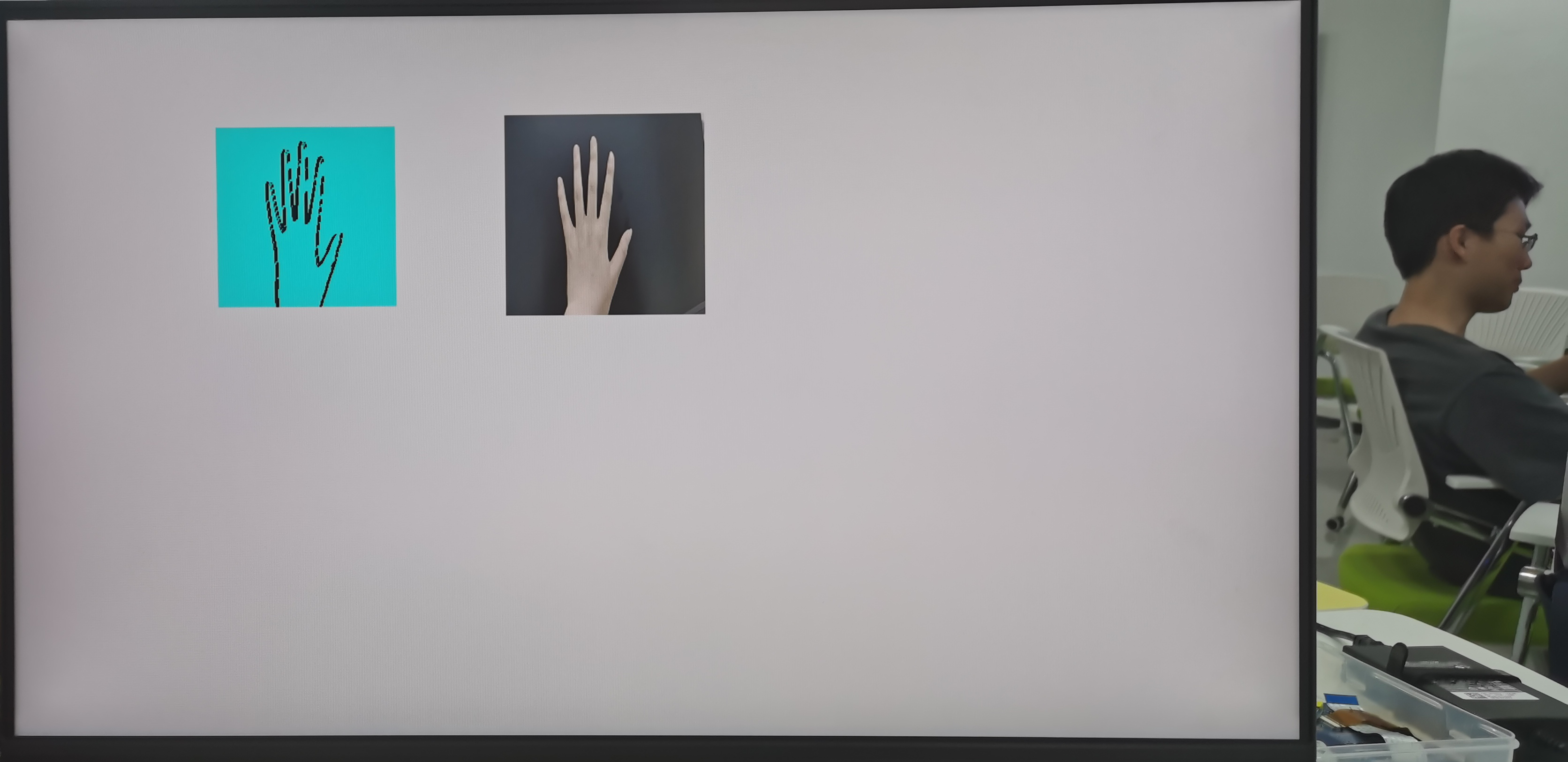


图8 单张图像滤波

实验二：实现实时的Sobel边缘检测如下所示，视频的流畅度很高，缺点是存在少数噪声。



图9 对矿泉水实时滤波

参 考 文 献：

[1] David H. Jacobsohn. A Suggestion for a Fast Multiplier.[J]. IEEE Trans. Electronic Computers,1964,13(6).

[2] 李迪,吴奇,杨浩森.基于改进Sobel算子的边缘检测系统的设计与实现[J].信息技术与网络安全,2022,41(03):13-17.DOI:10.19358/j.issn.2096-5133.2022.03.003.

[3] 罗紫阳,李雪梅,陈鹏.基于FPGA的Sobel算子边缘检测算法的实现[J].电子制作,2020,No.407(21):6-7.DOI:10.16589/j.cnki.cn11-3571/tn.2020.21.002.