

DOI:10.19651/j.cnki.emt.2209974

基于 Vivado HLS 的内窥镜实时暗部增强算法设计^{*}

王云峰 范正吉 何鑫 景金荣 洪应平

(中北大学仪器科学与动态测试教育部重点实验室 太原 030051)

摘要: 医用内窥镜的探头体积有限,使用环境特殊,照明条件有限,故图像的暗部处理显得尤为重要。本设计提出一种快速的彩色图像暗部增强算法,首先对采集到的 RGB888 格式视频流图像进行通道分离,然后通过卷积的方式对每个通道进行特定窗口大小的均值滤波来进行区域特征提取。最后对滤波后三个通道的每个对应点取平均值,代入特定参数的 Logistic 函数,得到的结果为该帧图像该位置像素点的增益,并将其应用于原图。本设计以 Xilinx 公司推出的 Zynq 系列 ARM+FPGA SoC 平台为载体,使用 Vivado HLS 进行 AXI-Stream 接口视频流处理算法的开发,并生成 IP 在 FPGA 端运行。经过实验,本算法在 Zynq7020 平台处理 $400 \times 400 @ 30$ fps 的一帧图像仅需 1.6 ms,保证了视频流输出的实时性。同时把内窥镜伸入口腔观察算法处理前后的视频流图像可知,本算法在保证区域对比度的同时增强了暗部亮度,较好的提升了医用内窥镜视频流的暗部质量。

关键词: 内窥镜;暗部增强;ZYNQ;Vivado HLS;视频图像处理

中图分类号: TN919.5 **文献标识码:** A **国家标准学科分类代码:** 510.40

Design of endoscope real-time dark part enhancement algorithm based on Vivado HLS

Wang Yunfeng Fan Zhengji He Xin Jing Jinrong Hong Yingping

(Key Laboratory of Instrumentation Science and Dynamic Measurement, Ministry of Education, North University of China, Taiyuan 030051, China)

Abstract: The probe volume of medical endoscope is limited, the use environment is special, and the lighting conditions are limited, so the dark part processing of image is particularly important. This design proposes a fast dark part enhancement algorithm for color image. Firstly, the collected RGB888 format video stream images are channel separated, and then the mean filter of specific window size is carried out for each channel through convolution to extract the regional features. Finally, the average value of each corresponding point of the three channels after filtering is taken and substitute it into the Logistic function of specific parameters. The result is the gain of the pixel point at that position of the frame image, which is applied to the original image. The design takes the Zynq series ARM+FPGA SoC platform launched by Xilinx as the carrier, uses Vivado HLS to develop the AXI-Stream interface video stream processing algorithm, and generates IP to run on the FPGA. After experiments, this algorithm only needs 1.6 milliseconds to process a frame of $400 \times 400 @ 30$ fps on the Zynq7020 platform, which ensures the real-time output of the video stream. At the same time, the endoscope is inserted into the oral cavity to observe the video stream images before and after the algorithm processing. The algorithm enhances the brightness of the dark part while ensuring the regional contrast, and improves the dark part quality of medical endoscope video stream.

Keywords: endoscope; dark enhancement; ZYNQ; Vivado HLS; video image processing

0 引言

经过几千年的发展,内窥镜得以在医学上发挥作用,但是研究者对内窥镜能得到更清晰的图像和更方便的操作的

热情并未停歇。相比于传统的外科手术,内窥镜手术产生更少的痛苦,更短的康复时间和减少伤疤^[1]。随着科技的不断进步和应用的实际需求,内窥镜镜体的尺寸也越来越小。带来的一个结果就是镜头的进光量越来越低,从而导

收稿日期:2022-05-17

^{*} 基金项目:国家自然科学基金青年科学基金(51705475)项目资助

致成像效果整体偏暗。由于应用环境大多在人体内部,故辅助光源的选用条件也及其严苛。辅助光源既要满足尺寸要求,又要满足照明要求,且亮度不能太高,否则会因为功率问题导致光源器件发热进而对人体造成不适。所以要在上述条件下获得高质量的图像,暗部增强显得尤为关键。

由于医用内窥镜的特殊用途,必须保证采集回的视频流的实时性,否则会对医生的操作产生影响。故执行算法处理必须最大限度的保证实时性。本设计中选用 Xilinx 公司的 Zynq 平台为载体,进行设计与实现。本算法固化在 FPGA 中,利用硬件并行化的思想改进传统算法的处理模式^[2]。由于 FPGA 的并行运算机制,决定了在实时性方面比 DSP 和 ARM 强很多。并且 Xilinx 提供的 Vivado HLS, 给在 FPGA 上开发图像处理算法开辟了另一条道路。

ZYNQ 是 Xilinx 公司推出的一款全可编程片上系统处理器,集成了处理器(processing system, PS)和可编程逻辑(programmable logic, PL),是一种 ARM+FPGA 的架构。其中 PS 处理器部分包含了完整的 ARM 处理子系统,而 PL 逻辑部分中的资源会随着芯片的级别的不同而不同。这两个部分相互联系又可以独立使用^[3]。

目前效果较好的暗态图像增强算法主要有 Retinex 算法、Lime 算法、MBLLEN 算法、KinD 算法等。Retinex 是一种常用的建立在科学实验和科学分析基础上的图像增强方法,它是由 Edwin. H. Land 于 1963 年提出的。Retinex 算法的核心思想是以色感一致性(颜色恒常性)为基础的。该算法不仅能够增强图像的亮度信息,同时可以去除图片中的部分阴影信息。但是该算法的运算速度慢,不能应用到一些实时的场景中;LIME 算法是一个简单而高效的低光照图像增强算法。该算法首先通过在 RGB 通道中找到最大值来单独估计每个像素的照。然后通过初始光照图上施加一个结构先验来细化,作为最终的光照映射。LIME 算法能够较好的处理一些低光照图片,但是处理后的图像与原图像相比色差较大;MBLLEN 算法是一个多分支低光照图像增强网络,该算法的核心思想是在不同等级中提取出丰富的图像特征,该算法可以获得更加自然的增强效果,缺点是光对比度强时亮部较易过曝。KinD 算法是将原始的图像空间分解为两个比较相似的子空间,然后使用不同曝光程度的图片块来进行训练,经过该算法处理的图像暗部增强效果好,色彩还原性好,缺点就是运算量极大,不适用于实时视频流的处理。

针对上述情况,本文设计了一种基于 Vivado HLS 的医用内窥镜彩色视频流实时暗部增强算法,以 Zynq 平台为载体。首先对采集回的 RGB888 帧图像进行通道分离,然后对每个通道进行滑窗均值滤波,再对滤波后的三个通道每个对应点取平均值后,代入特定参数的 Logistic 函数,得到的结果就是该帧图像该位置像素点的增益,把增益值按坐标代入原图进行乘积计算,得到处理后的图像,再把各个通道合并,还原回 RGB888 格式。实现了医用内窥镜视

频流实时暗部增强处理。

1 基于区域特征的暗部增强算法

由于我们的目的是增强图像暗部的同时,不让亮部过曝,所以不能用常规的亮度调整来进行图像的整体调节。相较于类似虚拟多曝光融合的图像增强方法^[4]、Retinex 算法、KinD 算法等虽然有效果,但是这些算法计算量过大,或者需要全图信息,很难做到实时处理,且占用资源过高。本算法为点对点处理,其基本思路就是对低灰度级的像素点给高增益,高灰度级的像素点给低增益。但是直接以当前像素点的灰度级进行增益调制,会出现区域内由于亮点给了低增益而暗点给了高增益,造成区域对比度下降的问题。故本设计提出以该像素点为中心划一正方形窗口,该窗口内像素点的平均值作为该像素点的灰度级特征,代入增益曲线,求得增益。这样的话由于局部平均值变化缓慢,可基本视为恒定,故相邻的像素点的增益不会差很多,避免了区域内对比度骤降从而导致图像细节丢失的问题。算法总体流程如图 1 所示。

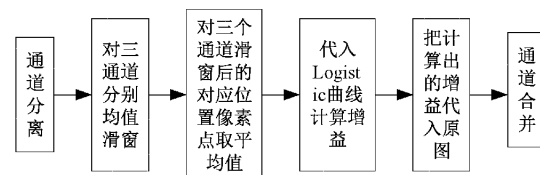


图 1 算法整体流程

1.1 区域特征提取

区域特征的提取本质上相当于对原始图像进行均值滑窗,是一个卷积的过程。即一个像素点的灰度值由其邻域内像素值的平均值代替。通常采用一个滤波窗口来选取邻域像素^[5]。以待处理的当前像素点 (x, y) 为中心,选择一个尺寸窗口 $\Omega(i, j)$,该窗口由其近邻的若干像素组成,求窗口中所有像素的均值,再把该均值赋予当前像素点,3 个通道进行完式(1)的运算,再对应点取平均值后,作为要代入增益曲线的输入值,即每个 RGB888 像素点的每个通道增益一致。

$$\bar{g}(i, j) = \sum_{(x, y) \in \Omega_{i, j}} g(x, y) / M \quad (1)$$

式(1)中的 M 为该窗口中包含当前像素在内的像素总个数。以 3×3 的窗口为例,下图展示了区域特征的提取方法,以该窗口为模板,与原图进行卷积运算。如图 2 所示。

在本设计中,实物的内窥镜型号为 OV6946,分辨率为 400×400 ,帧率 30 fps。经过实验,选择 15×15 的窗口较为合适。以下的代码设计与仿真、实现,均取 15×15 的滑动窗口大小。

1.2 增益的计算

在进行完 1.1 中的所有步骤后,得到图像每个像素点的区域特征灰度值,代入特定参数的 Logistic 曲线方程。Logistic 曲线方程既应用于社会经济现象研究,也广泛应

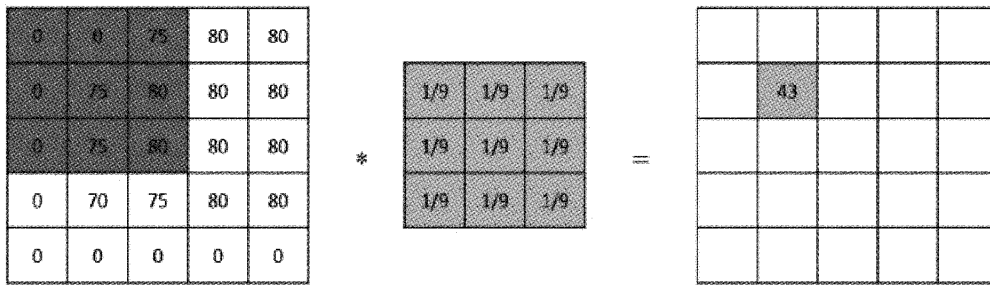


图 2 3×3 窗口条件的单通道区域特征提取

用于动植物生长发育或繁殖过程等研究。其主要特点是开始增长缓慢,而在之后的某一范围内迅速增长,达到一定限度后,增长又缓慢下来。即为一平滑的“S”型曲线。其基本公式如式(2),图像如图 3 所示。

$$S(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

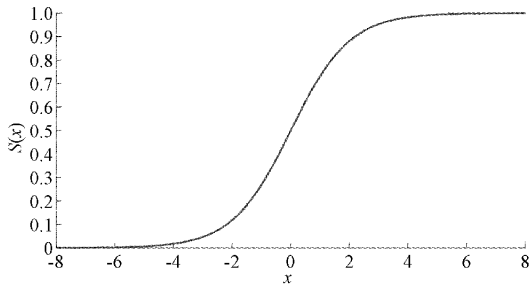


图 3 Logistic 曲线方程基本图像

该曲线过渡平滑,且两端对称趋近于极值,本设计正是需要低灰度值给高增益,高灰度值给低增益,且过渡平滑。基于此思想,提出了下面的函数式(3),图像如图 4。同时还可以根据不同内窥镜的使用环境来调节增益曲线的纵坐标的极值,来达到适配的目的。

$$y = \frac{-1}{1 + e^{10(\frac{x}{255} - 0.5)}} + 2 \quad (3)$$

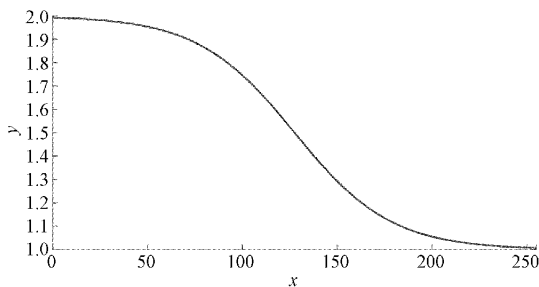


图 4 特定参数的 Logistic 曲线方程图像

式(3)即为本设计的增益曲线公式,其中横坐标为 0~255,因为像素点灰度值的范围为 0~255,在该像素点的区域特征灰度值接近于 255 时,代表该点的亮度高,增益平滑趋近于 1,即不变。在该像素点的区域特征灰度值接近于 0 时,代表该点亮度低,增益平滑趋近于 2,即两倍增益。

实现了暗部增强,且增益公式的输入为区域特征灰度值,保证了区域对比度不会明显下降。

2 基于 Vivado HLS 的设计与实现

本设计的算法是通过 Vivado HLS 开发完成。Vivado HLS 是最具创新性的可编程逻辑设计工具之一。使用 Vivado HLS 生成 IP 核可以大大缩短 FPGA 的开发周期。只需要使用高级语言,如 C、C++ 或 System C 来描述逻辑,然后将逻辑转换为 RTL 级别的实现。利用 Vivado HLS 可以缩短 1/3 的 RTL 仿真时间,使算法验证速度提高 10 倍以上^[6]。Vivado HLS 的功能简单地来说就是把 C、C++ 或 SystemC 的设计转换成 RTL 实现,然后就可以在 Xilinx FPGA 或 Zynq 芯片的可编程逻辑中综合并实现,但是不同于纯粹的高级语言开发,在使用 Vivado HLS 开发时,编写代码仍要基于硬件描述语言的思想,最后进行协同仿真与高层次综合,根据所编写或配置约束生成 RTL 文件。其开发流程如图 5 所示。

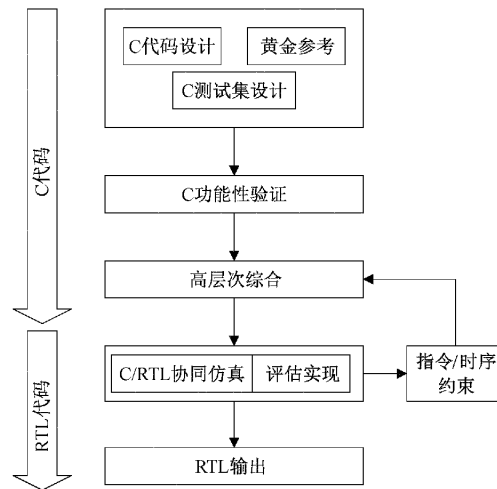


图 5 Vivado HLS 设计流程

2.1 区域特征提取的 Vivado HLS 实现

区域特征的提取主要应用 Vivado HLS 的 OpenCV 库。VivadoHLS 对 OpenCV 的支持,不是指可以将 OpenCV 的函数库直接综合成 RTL 代码,而是需要将代码转换为可综合的代码,这些可综合的视频库称 HLS 视频

库,由 VivadoHLS 提供^[7]。对于本设计中的区域特征提取,首先定义一个 8 位的 3 通道 mat 作为该函数的输入,即 RGB888 格式的彩色图像帧数据,16 位的 3 通道 mat 格式作为输出,高 8 位存储卷积滑窗运算后的数据,低 8 位作为原图的缓存。之所以这样做的原因是由于 Vivado HLS 中的 mat 类型,实际上为 stream 类型,即流类型,所有函数间都是通过 mat 的形式作为输入输出,实现并行流水线操作。只可以顺序存取,不可以随机存取,因此无法进行数据的复用。“流”的思想必须贯穿整个工程,这样才能保证

高层次综合成功且达到较高的实时性与较低的资源占用。这也是用 Vivado HLS 进行开发与普通高级语言编写算法不同的地方。然后定义一个 15x15 的内核,利用内核实现卷积运算,内核的基准点(anchor),定义为(-1,-1)说明位于内核的中心位置。基准点即内核中与进行处理的像素点重合的点。卷积核内权值都为 1,求核内所有点的和并除以核内点的个数即完成了区域特征提取并存储下来原始数据,分别给计算增益和应用增益做准备。实现简化流程如图 6 所示。

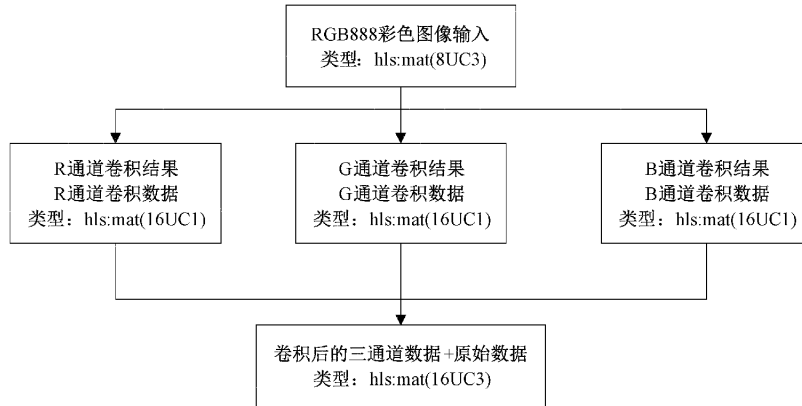


图 6 区域特征提取 Vivado HLS 设计简化流程

2.2 增益计算与应用的 Vivado HLS 实现

由于区域特征灰度值范围为 0~255,由式(3)可知,实际上增益也只有 256 种情况。由于增益计算函数的计算量较大,故采用查找表的方法来提高代码运行效率与实时性,即把 256 个增益数据定义为全局数组。增益数

值取出后直接与缓存的原始图像数据相乘,得到结果,该函数的输入为 16 位的 3 通道 mat 格式,即 2.1 中区域特征提取函数的输出。该函数的输出为 8 位的 3 通道 mat 格式,即处理完的 RGB888 彩色图像帧数据。简化流程如图 7 所示。

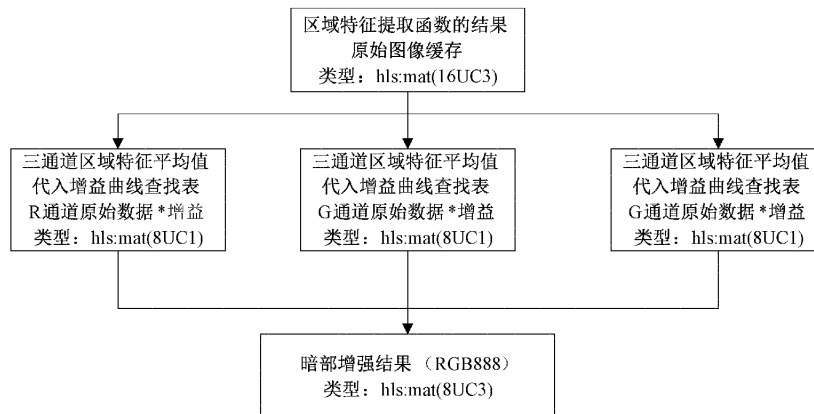


图 7 增益计算与应用 Vivado HLS 设计简化流程

2.3 流接口封装的 Vivado HLS 实现

上述计算流程的实现都是基于单帧图像数据的处理,要使其能够处理视频流,即固定频率与时序来的连续帧图像数据,需要借助 AXI4-Stream。AXI4-Stream 总线协议由 ARM 公司提出,该协议专门针对视频,音频,数组等数据在片内通信设计^[8]。

AXI4 总线包含 3 种类型的接口:AXI4 接口、AXI4-

Lite 接口以及 AXI4-Stream 接口。AXI4 接口主要面向高性能地址映射通信的需要,功能完整,在单地址传输的情况下最大允许 256 轮的数据突发长度^[9]。其中 AXI4-Stream 适用于高速数据流,去掉了地址项,允许无限制的数据突发传输。除了总线时钟和总线复位,其他的接口信号都是以字母 t 开头。log_clk 全局时钟信号,在上升沿时对信号采样。所有的输入信号都通过上升沿采集,所有的

输出信号都在上升沿时变化。rst_n 全局复位信号,低电平有效。tvalid 握手信号。tlast 表示传输的最后一个字节,置高有效。tdata 为数据信号。tkeep 为 tdata 的内容修饰符,用于指示是此字节是否为有效字节。tuser 为用户自定义信号。其时序图如图 8 所示。

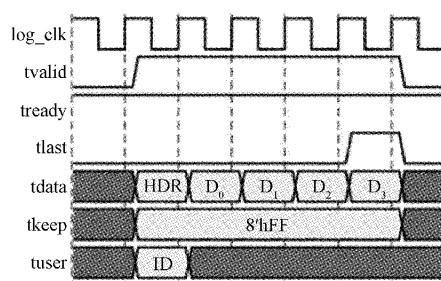


图 8 AXI4-Stream 时序图

在 Vivado HLS 中,使用 AXIvideo2Mat 函数,即可把输入的流变换为单帧的 mat 格式进行处理,处理后使用 Mat2AXIvideo,变换为流输出。故算法处理函数最后需要封装在流输入和流输出的总函数内。并把流作为最后生成 IP 核的输入输出接口。

3 仿真与综合验证

3.1 算法仿真的 Vivado HLS 实现

Vivado HLS 除了将高效的编程代码转换为 IP 以外,还提供了另一项重要功能:C 语言仿真和 C/RTL 协同仿真^[10]。在对 HLS 设计进行综合之前,要先对其进行功能性验证,也就是 C 仿真,其目的是验证 HLS 输入的 C 代码的功能是否正确,实现效果是否达到需求。验证的方式就是编写 TestBench,也就是激励文件。在激励文件中首先将本地用于测试的 400×400 分辨率的图片利用 IplImage2AXIvideo 函数转换为 AXI4-Stream 接口协议的数据,作为工程中写好的算法函数的输入,最后把算法输出的 AXI4-Stream 接口协议的数据利用 AXIvideo2IplImage 函数进行转换,转换为图片数据,再利用 cvShowImage 函数进行原始图像与算法处理后图像的显示与对比。图 9 显示了原图和几种处理方式的对比。图 9 中的(a)为原始图像,图 9(b)为经过区域特征提取,代入增益曲线,求得增益后再与原图各像素点相乘后的结果。图 9(c)为不求区域特征灰度值,直接把该像素点的灰度值代入增益曲线后,与原图各像素点相乘。图 9(d)为直接调节图片亮度,使暗部的建筑物和物体的亮度与经过本文算法处理的暗部建筑物和物体基本相同。

经过对比可知,图 9 (b)对比于图 9 (c),在蓝天和云彩的交界处看的比较明显,在增强了原图中暗部的同时,较好的保留了细节,图 9 (c)的对比度下降较为严重,图像整体模糊。图 9 (d)为普通的图片亮度调整,可以看到在暗部增强的同时。亮部已经严重失真。



图 9 Vivado HLS 算法仿真结果对比图

综上,本设计的算法,即图 9 (b)效果最优。在增强暗部的同时,避免了亮部失真过曝,同时较好的保留了对比度,细节几乎没有损失。

3.2 算法综合的 Vivado HLS 实现

经过上面的仿真,下一步需要进行高层次综合,即把 C 编程代码转化为 RTL 文件,生成可固化在 FPGA 端的 IP 核。这个过程 Vivado HLS 会对 C 工程进行优化,以便实现更健壮的时序、更低的延时与更少的资源占用。Vivado HLS 有两种优化方式,一种是在 Directive 控制栏设置优化变量和参数,另一种方法是在代码中使用 #pragma 命令来定义变量实现类型和结构^[11]。本设计通过上述两种方式对 C 工程进行了优化,通过 #pragma 命令,主要指示 Vivado HLS 通过并行方式安排执行该函数的所有子函数,从而达到缩短延时、提高吞吐率的效果^[12]。通过 Directive 设置,指示 Vivado HLS 把工程中用到的变量定义为何类资源,从而达到合理资源分配、降低资源占用的效果。本设计的算法经过 Vivado HLS 综合后,统计了其在 XC7Z020-CLG400-2 器件 FPGA 部分的资源占用率,如表 1 所示。Vivado HLS 综合后的时钟和延时如图 10 所示。由图 10 可知。目标时钟周期为 10 ns,综合后的设计所估计的时钟周期为 8.631 ns,满足时序要求。最高延时时为 169 724 个时钟周期,即执行本设计的算法处理一帧图像的时间为 1.697 24 ms。由于 30 fps 的内窥镜,图像帧间隔为 33.3 ms,故完全满足实时处理 OV6946 内窥镜的 400×400 分辨率,每秒 30 帧图像的视频流。

3.3 IP 核的生成与系统实现

本算法经上述的仿真、优化与综合后,在 Vivado HLS

表 1 PL 资源占用统计表

资源	BRAM	DSP	触发器	查找表
FIFO	0	0	45	252
接口	46	4	22 143	20 027
寄存器	0	0	0	0
共用	46	4	22 188	20 279
可用	280	220	106 400	53 200
百分比	16%	1%	20%	38%

Timing (ns)

Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	8.631	1.25

Latency (clock cycles)

Summary

Latency		Interval		Type
min	max	min	max	
169724	169724	169723	169723	dataflow

图 10 时钟与延时

中可直接生成 IP 核,用于固化到 FPGA 中。通过图 10 的数据可知,结合 FPGA 并行计算的特点和资源约束^[13],利用 FPGA 设计底层 IP 核是提升图像处理速度的有效途径之一^[14]。本设计生成 IP 的数据输入输出接口均为 AXI4-Stream 协议接口,通用性高,可以方便的移植到其它 AXI4-Stream 协议的视频流处理工程中,最后生成的 IP 核顶层如图 11 所示。

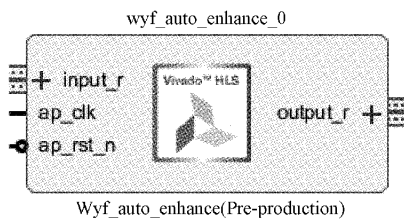


图 11 生成的 IP 核在 Block Design 中的形态

其中,ap_clk 为输入时钟。ap_rst_n 为复位信号,低电平有效。input_r 和 output_r 为 AXI4-Stream 协议的视频流输入和处理后的视频流输出。把该 IP 核添加进 Vivado 工程的 Block Design 中后,通过连线的方式即可使该 IP 核与其它设计进行数据交互,简洁明了。

系统以 Zynq7020 SOC 为核心处理平台^[15],可编程逻辑与处理系统之间通过 AXI 总线互联^[16]。实现 OV6946 内窥镜的视频流采集、处理、输出与显示。采用 Zynq-7000 实现的嵌入式图像处理系统,既能够通过 FPGA 获得较高的计算性能,又能够依靠 ARM 处理器提供完善的应用功能^[17]。实物图如图 12 所示。

本系统由 FPGA 端采集 OV6946 内窥镜经 OV426 桥

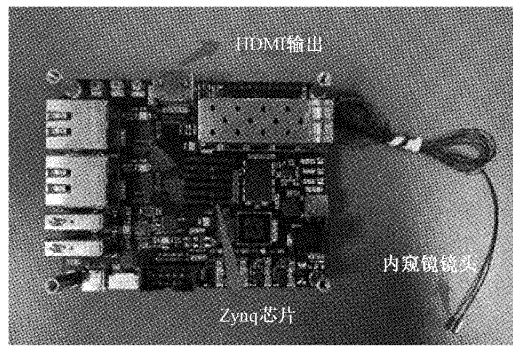


图 12 测试系统-测试装置实物图

接后输出的 DVP 协议数字信号图像,通过 AXI4 总线缓存到 ARM 端的 DDR3,经 ARM 做部分处理后再次缓存并由 VDMA 读出并转换为 AXI4-STREAM。通过 Vivado HLS 编写输入输出接口均为 AXI4-STREAM 的暗部增强处理算法,固化到 FPGA 端运行,通过 HDMI 接口显示到显示器。最后实物效果如图 13 所示。把 OV6946 内窥镜握在手中以模拟进入体腔内的情况。其中左为本设计中的算法处理前,右为处理后。

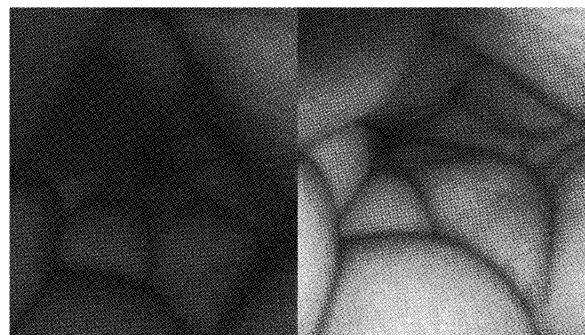


图 13 暗部增强前(左)后(右)对比图

把图 12 系统中的内窥镜镜头放入口腔内观察喉部,图 14 为未经本文算法处理,直接对采集后的视频流进行 HDMI 输出后截取的图像,图 15 为经过本文算法处理后经 HDMI 输出截取的图像。

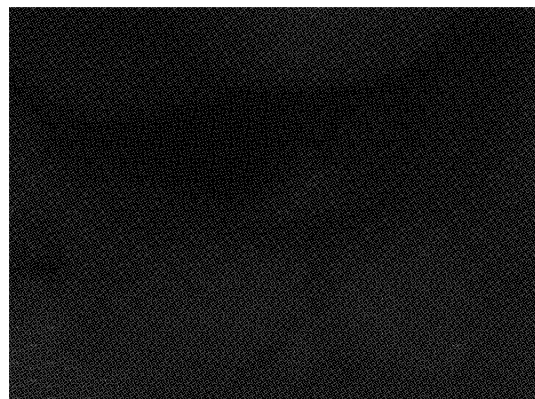


图 14 未经算法处理的原始图像截取



图 15 经过本文算法处理的图像截取

由图 14 和 15 可知,经本算法处理后的视频流暗部图像质量得到了明显增强,且色彩真实,对比度几乎没有损失,血管清晰可见。并且由图 10 可知,处理后的视频流相较于原始视频流输出无延时,在视频流帧间隔即可处理完成并输出。

4 结 论

本文从电子内窥镜设备视频流处理需求出发,设计了基于 Vivado HLS 的内窥镜彩色视频流实时暗部增强算法,即提升了暗部质量,又避免了亮区过曝,同时还保证了区域对比度,避免了邻近像素点在灰度值差异过大时因增益差异过大而导致的图像细节丢失。在满足需求的前提下,对各个指标均进行了优化设计,保证了处理的实时性。同时,该系统已投入到实际使用中,具有较好的实用价值。

参考文献

- [1] 章科建. 高清电子医用内窥镜图像增强技术研究[D]. 杭州:浙江大学, 2018.
- [2] 张妮娜, 窦衡. 并行化改进遗传算法的 FPGA 高速实现方法[J]. 太赫兹科学与电子信息学报, 2012, 10(1):107-109.
- [3] 孙超. 基于嵌入式系统的实时成像处理研究[D]. 陕西:西安电子科技大学, 2017.
- [4] 徐望明, 刘召, 伍世虔, 等. 基于虚拟多曝光融合的不均匀光照图像增强方法[J]. 华中科技大学学报:自然科学版, 2020, 48(8):1-7.
- [5] 张新明, 程金凤, 康强, 等. 迭代自适应权重均值滤波的图像去噪[J]. 计算机应用, 2017, 37(11): 3168-3175.

- [6] SHARAFEDDIN M, SAGHIR M A R, AKKARY H H, et al. On the effectiveness of accelerating MapReduce functions using the Xilinx Vivado HLS tool [J]. International Journal of High Performance Systems Architecture, 2016, 6(1):1-12.
- [7] 郭丰收. Xilinx FPGA/Zynq 设计中使用 HLS 实现 OpenCV 的开发流程[J]. 电子产品世界, 2014, 21(2):50-52.
- [8] 贺理, 赵鹤鸣, 邵雷. AXI4-Stream 总线的 FPGA 视频系统的开发研究[J]. 单片机与嵌入式系统应用, 2015, 15(12):42-45.
- [9] 杨博, 楚要钦, 景德胜, 等. 基于 AXI4 总线的 SoC 中离散量信号控制 IP 核的设计与验证[J]. 信息通信, 2020, 207(3):95-97.
- [10] 张艳辉, 郭洛宇, 何宾. Vivado HLS 嵌入式实时图像处理系统的构建与实现[J]. 电子技术应用, 2016, 42(9):115-121.
- [11] 彭习武, 张涛. 基于 Vivado HLS 的边缘检测硬件加速应用[J]. 电子技术应用, 2017, 43(5):70-73.
- [12] 何凯, 梁蓓, 杨发顺. 基于 Vivado HLS 的求取特征点图像坐标的设计[J]. 电子科技, 2018, 31(4):87-90.
- [13] 周彦臻, 吴瑞东, 于潇, 等. 面向 FPGA 部署的 CNN-SVM 算法研究与实现[J]. 电子测量与仪器学报, 2021, 35(4):90-98.
- [14] 张壮. 基于 HLS 的智能相机图像处理底层 IP 核设计与实现[D]. 南京:南京邮电大学, 2020.
- [15] 卫建华, 刘润利, 许佳豪, 等. 基于 PYNQ 框架的人体目标跟踪系统[J]. 国外电子测量技术, 2021, 40(12): 89-95.
- [16] 刘阳, 冯进良, 黄伟, 等. 基于 FPGA 的三目半全局匹配算法设计与实现[J]. 仪器仪表学报, 2021, 42(12):202-210.
- [17] 焦再强. 基于 Zynq-7000 的嵌入式数字图像处理系统设计及实现[D]. 太原:太原理工大学, 2015.

作者简介

王云峰, 硕士研究生, 主要研究方向为 FPGA 数字信号处理与图像处理。

E-mail: 941157912@qq.com

洪应平(通信作者), 讲师, 主要研究方向为极端环境下测试技术及仪器研究。

E-mail: hongyingping@nuc.edu.cn