

〈图像处理与仿真〉

## 基于 ZYNQ 的多传感器实时图像拼接技术研究

冯 鑫，汪陈跃，孙爱平，赵 强，雷旭峰，董海翔，康丽珠，杨增鹏  
(昆明物理研究所，云南 昆明 650223)

**摘要：**为满足高速多传感器大视场图像处理系统结构紧凑、高实时性的迫切要求，进行了以全可编程平台 ZYNQ 为基础的多传感器实时图像拼接技术研究。重点介绍了该设计的基本架构、工作原理及 ZYNQ 的功能模块设计等，通过对图像处理的结果表明，该技术满足高速实时图像处理的要求，同时可以扩展到更为复杂的多传感器大视场实时图像处理任务。

**关键词：**ZYNQ；图像拼接；实时图像；多传感器

中图分类号：TP311 文献标识码：A 文章编号：1001-8891(2016)03-0207-04

## Research on Real-time Multi-sensor Image Mosaic Technology Based on ZYNQ

FENG Xin, WANG Chenyue, SUN Aiping, ZHAO Qiang,  
LEI Xufeng, DONG Haixiang, KANG Lizhu, YANG Zengpeng  
(Kunming Institute of Physics, Kunming 650223, China)

**Abstract:** This paper conducts a research on a real-time multi-sensor image mosaic technology based on the all programmable platform, ZYNQ, in order to satisfy the urgent requirement of the compact structure and high level real-time in high-speed multi-sensor image processing system. This paper focuses on the basic architecture, operation principle and functional module design in ZYNQ. The result of processing digital images proves that the technology satisfies the requirement of high-speed real-time image processing. It can expand into the task for more complicated multi-sensor real-time image processing.

**Key words:** ZYNQ, image mosaic, real-time image, multi-sensor

### 0 引言

随着军事分离口径成像系统的发展，在多传感器实时图像处理系统中，如何在规定时间内完成大量图像数据的处理，满足系统的实时性要求一直是人们重点关注的问题。随着大规模集成电路的发展，各种处理器及专用集成电路的性能不断得到提高，人们对图像处理系统的功能、性能、功耗、体积等都提出了更高的要求。常见的多传感器大视场图像处理系统大多采用多 DSP+FPGA 架构<sup>[1-2]</sup>或者 CPU+GPU 架构<sup>[3]</sup>，这样的架构造成了该系统成本高、体积大、功耗高、设计相对不灵活、设计复杂，增加系统开发周期等缺点。本系统鉴于满足多传感器图像处理系统中数据运算量大、实时性强，数据传输率高，尤其是对体积、功耗要求严苛的情况下，开展了以全可编程 SOPC 为

基础的多传感器实时图像拼接技术研究，不仅可以通过软件增加系统的灵活性和可扩展性，同时还能通过可编程硬件实现执行更多的数据处理和决策功能<sup>[4]</sup>。

### 1 系统框架

本系统采用美国 Xilinx 公司推出的业界第一款全可编程平台 ZYNQ 作为图像拼接技术研究的主要硬件平台。主要包括图像采集、图像预处理、图像传输、图像配准及图像拼接显示几个模块。系统设计框图如图 1 所示。

系统通过 2 个图像传感器采集图像，经过 ZYNQ 内部的图像采集及预处理，图像传输，图像存储，图像配准后，最终图像显示。ZYNQ 片内的 PL（可编程逻辑）和 PS（处理器系统）之间的数据是通过基于 AXI4.0 协议的片内总线传输的，AXI4.0 是由 ARM 和

收稿日期：2015-09-09；修订日期：2016-01-12。

作者简介：冯鑫（1989-），吉林通化人，硕士研究生，主要研究方向为图像处理。

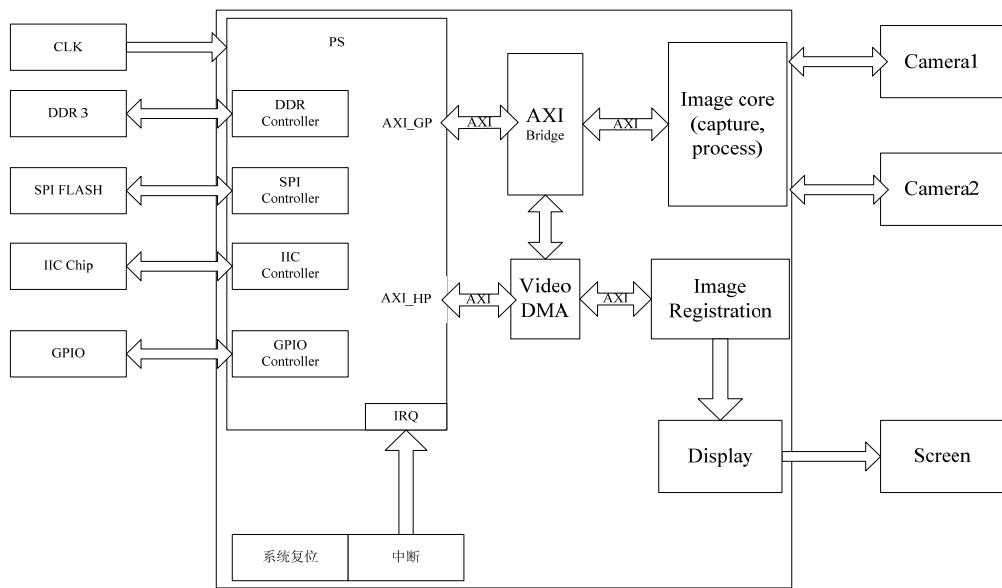


图1 系统框图

Fig.1 The chart of System

Xilinx 公司基于 ARM 公司的 AMBA3.0 总线协议共同提出的，是一种高性能、高带宽、低延迟的片内总线<sup>[5]</sup>，通过 AXI 总线将 Zynq-7000 内 PS 和 PL 紧密地耦合在一起。通过采集外部摄像头 1 和摄像头 2 的图像数据，经过图像预处理模块，将数据打包成 AXI4-Stream 总线格式数据，然后经过 Video DMA 控制器，将数据从 AXI\_HP 接口写入 DDR3 控制器后，写入外部 DDR3 存储器，再将数据从 DDR3 中读出，完成数据的高速率吞吐。数据以较高速率流入图像配准模块，最后至图像显示模块，完成跨时钟域图像处理，将数据解调成符合显示时序的数据格式。

## 2 功能模块设计

### 2.1 图像捕获及预处理模块

本文采用 CMOS 传感器，传感器阵列为  $640 \times 480$ 。CMOS 传感器具有高帧频、可配置、功耗低、成本低等优点，在众多领域都有应用。本系统通过 I<sup>2</sup>C 总线对 CMOS 进行配置，输出 RAW RGB 格式的图像数据，每个像素为 8 bit。其配置电路，捕获及预处理模块如图 2、图 3 所示。

图像传感器通过 I<sup>2</sup>C 接口配置后，给出配置结束信号 i2c\_cfg\_done，接收此信号后才开始进行完整帧图像采集。在像素时钟 cmos\_pclk 的节拍下，图像数据 cmos\_data[7:0] 在图像传感器的行有效信号 cmos\_href 及场有效信号 cmos\_vsync 的作用下，将数据输入至 ZYNQ。在 ZYNQ 内部，对 2 路视频流同时捕获，将数据送至图像预处理模块。图像预处理模块接收到图像采集模块的数据，通过插值算法将 RAW RGB 格式的数据转换为 RGB888 格式的数据，进而得

到了每秒 60 帧 24 位  $640 \times 480$  像素。

### 2.2 图像传输模块

图像数据是基于 AXI4-Stream 总线协议进行传输的，通过调用 Xilinx 公司提供的 AXI Video DMA IP 软核实现数据传输。CPU 通过接受来自图像捕获和预处理模块的完成标志信号，通过 AXI4-Lite 总线配置 AXI Video DMA 模块，初始化结束后开始传输数据。图像预处理模块的输出数据要转化为基于 AXI4-Stream 总线协议格式，数据经过 AXI-HP 接口，以 DMA (Direct Memory Access) 的方式将数据存储至外部数据存储器 DDR3，在适当的时候再以 DMA 的方式从 DDR3 中读出，再进行后续的图像配准处理。DMA 搬移数据的方式是 PL 与外部数据存储器数据交互的时间最快、吞吐率最高的一种方式，可以极大的满足系统的实时性，数据传输的过程中并不占用 CPU，进而提高了系统性能。

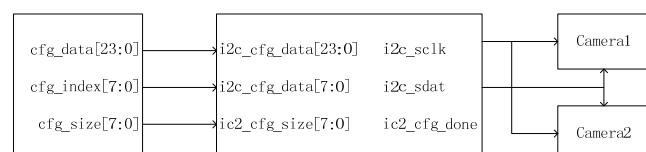


图2 图像传感器配置电路连接框图

Fig.2 Diagram for image sensors configuration circuit connection

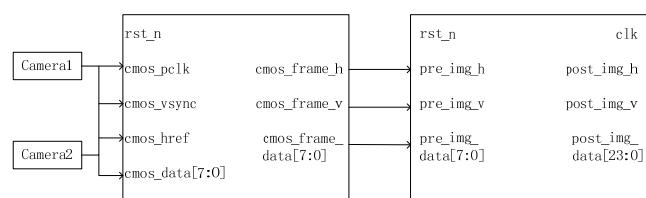


图3 图像捕获及预处理框图

Fig.3 Diagram for image capture and image pre-process

### 2.3 图像配准模块

图像拼接算法的性能主要取决于图像配准的精度, 配准算法既要保证配准精度算法的运算时间又要满足实时性的要求<sup>[6]</sup>。系统通过机械结构固定好图像传感器, 保证所成的图像之间只存在水平方向的位移, 所以这里采用可以有效提取出图像位移的图像配准算法——相位相关法进行图像配准点的提取。图像的相位相关配准算法, 主要是将图像进行傅里叶变换, 计算2幅图像的互功率谱, 再对互功率谱求傅里叶反变换得到冲激函数, 找出使该冲激函数取得最大值的位置, 即为图像的平移量。具体原理描述如下:

其中  $f_1(x, y)$  和  $f_2(x, y)$  分别代表2幅图像, 那么2幅图像的变换关系可以表示为:

$$f_2(x, y) = f_1(x - x_0, y - y_0) \quad (1)$$

根据傅里叶变换性质可得:

$$F_2(u, v) = F_1(u, v) e^{-j(ux_0 + vy_0)} \quad (2)$$

式中:  $F_2(u, v)$  和  $F_1(u, v)$  是  $f_2(x, y)$  和  $f_1(x, y)$  的傅里叶变换。它们的互功率谱为:

$$\frac{F_1^*(u, v) F_2(u, v)}{|F_1^*(u, v) F_2(u, v)|} = e^{-j(ux_0 + vy_0)} \quad (3)$$

式中:  $F_1^*(u, v)$  是  $F_1(u, v)$  的复共轭, 对互功率谱求傅里叶反变换, 找到峰值的位置, 即可以确定平移参数  $x_0$  与  $y_0$ 。

计算一个  $N$  点基2的快速傅里叶变换, 需要有  $(N/2) \times \log_2 N$  次蝶形运算和  $2N \times \log_2 N$  次数据访问。图像数据的傅里叶变换是对行列数据分别做傅里叶变换。一次复数的乘法运算包括包含4次乘法运算及3次加法运算。一幅  $N \times N$  的图像经过傅里叶变换后产生  $N^2$  个复数, 经过开方及除法运算后, 算法的复杂度大大提高。考虑到硬件实现的局限性及硬件算法的精度问题, 这里将采用 Matlab 对图像进行相位相关运算<sup>[7]</sup>。固定好图像传感器之后, 采集图像, 提取平移量, 利用此平移量参与 ZYNQ 配准控制逻辑。

### 2.4 图像拼接显示模块

图像拼接显示模块的主要功能是完成基于 AXI4-Stream 总线数据的跨时钟域处理。本文设计了图像拼接显示模块用于完成 AXI4-Stream 总线的控制, 发出 DMA 占用总线请求, 访问 DDR3 存储器, 同时将基于 AXI4-Stream 总线数据解调为符合终端显示的时序的数据。图像拼接显示模块如图 5 所示。

总线请求及数据解耦模块通过发动总线占用请求, 发起读取内存请求, 接收来自 DMA 控制器的数据, 并将总线数据解码, 将数据进行行缓存。图像显示控制模块根据接收到的 coordinate[10:0] 配准点信息, 发出控制地址 pixel\_addr\_0[9:0] 和 pixel\_addr\_1[9:0], 读取相应的像素数据 frame\_pixel\_0[23:0] 和 frame\_pixel\_1[23:0]。最终经过图像显示模块, 可以输出2路视频图像的所有信息。

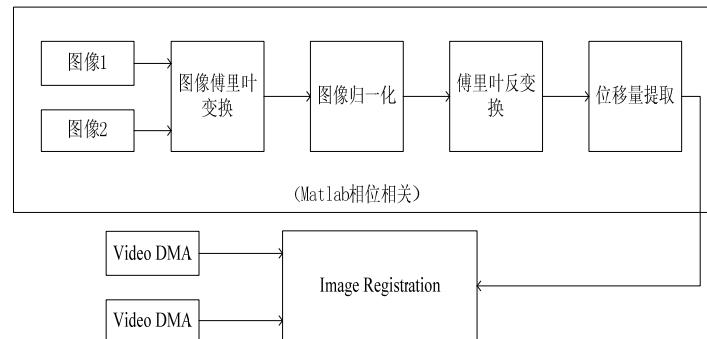


图 4 图像配准模块功能图

Fig.4 Function diagram for image registration

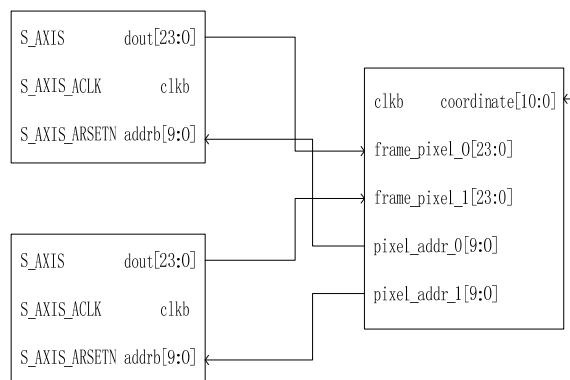


图 5 图像拼接显示模块框图

Fig.5 Diagram for image display module

### 3 硬件资源消耗分析

表1表明在处理2路分辨率为 $640\times480$ , 帧率每秒60帧的数据所消耗的硬件资源数量。利用片内PS的硬核处理器(ARM CortexA9处理器)完成相应的硬件控制, 极大地节省了相应的逻辑资源。同时借助于外部存储器及片内总线数据传输结构, 最大限度地节省PL的存储资源。

表1 硬件资源消耗分析

Table 1 Hardware resource consumption analysis

资源	使用量	可用量	消耗率/%
FF	9353	106400	8.79
LUT	8227	53200	15.46
MemoryLUT	1154	17400	6.63
BRAM	24.5	140	17.5
ARM CortexA9	1	2	50



(a) 摄像头1图像

图6 两路采集图像



(b) 摄像头2图像

Fig.6 Images collected by two cameras



图7 ZYNQ拼接结果图

Fig.7 Results of image mosaic based on ZYNQ

表2 拼接前后参数对比

Table 2 Parameters comparison before and after mosaic

参数	拼接前	拼接后
像素	$640\times480$	$1180\times480$
水平角度	$30^\circ$	$58.5^\circ$
垂直角度	$24^\circ$	$24^\circ$
时钟频率/(MHz)	24	75
帧数/(fp/s)	60	60

### 4 实验测试结果

设计要求2幅图像的重叠区不大于7%。重叠区相对大, 监控的目标视场角相对变小, 信息损失, 拼接起来相对容易。重叠区相对小, 监控的目标视场角相对变大, 信息丰富, 拼接起来相对困难。图6给出了2个具有重叠区的摄像头所采集的图像。图7给出了拼接后的图像。拼接后的图像信息量大, 没有冗余信息, 经过实时处理之后, 图像的大小为 $1180\times480$ , 帧率为每秒60帧, 图像成像清晰。

表2给出了2路图像的拼接前后的参数, 通过拼接前后的参数对比, 可以看出每个图像传感器的水平视场角度为 $30^\circ$ , 拼接后的水平视场角为 $58.5^\circ$ , 有效地增大了目标观测的视场范围, 且重叠区保持在5%左右, 满足重叠区小于7%的设计要求。具备较好的成像效果, 保证了系统的实时性, 达到了预期的效果。



(b) 摄像头2图像

Fig.6 Images collected by two cameras

### 5 结论

基于ZYNQ全可编程平台进行图像拼接技术研究, 进行了实时图像处理系统的采集、预处理、传输、拼接显示的模块设计, 完成了每秒60帧 $640\times480$ p的图像处理, 与传统设计方案相比, 可以使多传感器图像处理系统体积更小, 成本更低, 同时具有较好的实时性, 更符合未来实时图像处理系统的发展需求。

(下转第217页)

- 2013.
- Zhu Jinhua. The study of target tracking algorithm based on information fusion[D]. Nanjing: Nanjing University of Science and Technology, 2013.
- [5] 赵鹏, 浦昭邦, 张田文, 等. 一种新的红外与可见光图像融合与跟踪方法[J]. 光电工程, 2005, 32(2): 37-40, 92.  
Zhao Peng, Pu Zhaobang, Zhang Tianwen. Fusion and tracking for IR and visible images[J]. *Opto-Electronic Engineering*, 2005, 32(2): 37-40, 92.
- [6] 陈少华. 基于红外与可见光图像融合的目标跟踪[D]. 南京: 南京航空航天大学, 2012.  
Chen Shaohua. Target tracking based on infrared and visible image fusion[D]. Nanjing: Nanjing University of Aeronautics and Astronautics, 2012.
- [7] Birchfield S T, Rangarajan S. Spatiograms versus histograms for region-based tracking[C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005, 2: 1158-1163.
- [8] Zivkovic Z, Krose B. An EM-like algorithm for color-histogram-based object tracking[C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004, 1: 798-803.
- [9] Collins R T, Liu Y, Leordeanu M. Online selection of discriminative tracking features[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, 27(10): 1631-43.
- [10] 杨悦, 刘兴森, 郭启旺, 等. 基于改进互信息的红外目标匹配跟踪算法[J]. 红外技术, 2013, 35(6): 350-354.  
Yang Yue, Liu Xingmiao, Guo Qiwing, et al. Infrared object matching tracking algorithm based on improved mutual information[J]. *Infrared Technology*, 2013, 35(6): 350-354.
- [11] 高文, 朱明, 贺柏根, 等. 目标跟踪技术综述[J]. 中国光学, 2014, 7(3): 365-275.  
Gao Wen, Zhu Ming, He Baigen, et al. Overview of target tracking technology[J]. *Chinese Optical*, 2014, 7(3): 365-275.
- [12] Otsu N. A threshold selection method from gray-level histograms[J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 1979, 9(1): 62-66

(上接第 210 页)

## 参考文献:

- [1] HAO Wei. Designing and development of multi-DSP real-time image processing system based on FPGA[C]//IEEE International Conference on Computer Science and Network Technology, 2011: 1263-1265.
- [2] 银志军. 基于多 DSP 和 FPGA 的实时图像处理系统设计[J]. 光电技术应用, 2012, 27(1): 76-80.  
YIN Zhijun. Design of real-time image processing system based on FPGA and multi-DSP[J]. *Electro-optic technology application*, 2012, 27(1): 76-80.
- [3] 王雨曦, 王义坤, 葛明峰. 基于点特征算子的红外图像实时拼接系统[J]. 红外技术, 2015, 37(3): 204-209.  
WANG Yuxi, WANG Yikun, GE Mingfeng. Point feature-based real-time infrared image mosaicking system[J]. *Infrared Technology*, 2015, 37(3): 204-209.
- [4] 何宾. Xilinx All Programmable Zynq-7000 Soc 设计指南[M]. 北京: 清华大学出版社, 2012: 13.
- HE Bin. *Xilinx All Programmable Zynq-7000 Soc*[M]. Beijing: Tsinghua university press, 2012: 13.
- [5] Xilinx Inc. Xilinx AXI Reference Guide[R/OL]. [2015-09-09]. [http://www.xilinx.com/support/documentation/data\\_sheets/axi\\_reference\\_guide](http://www.xilinx.com/support/documentation/data_sheets/axi_reference_guide).
- [6] 张云峰. 基于 DSP 的实时图像拼接技术[J]. 液晶与显示, 2013, 28(6): 963-966.  
ZHANG Yunfeng. Real-time image mosaic technology based on DSP[J]. *Chinese Journal of Liquid Crystals and Displays*, 2013, 28(6): 963-966.
- [7] 杨磊, 任龙, 刘庆, 等. 基于 FPGA 的大视场图像实时拼接技术的研究与实现[J]. 红外与激光工程, 2015, 44(6): 1929-1935.  
YANG Lei, REN Long, Liu Qing, et al. Research and implementation of large field image real-time mosaic technology based on FPGA[J]. *Infrared and Laser Engineering*, 2015, 44(6): 1929-1935.