

图像灰度级拉伸算法的 FPGA 实现

唐耀飞

(中国空空导弹研究院, 河南 洛阳 471009)

摘要: 为了调整图像数据灰度, 介绍了一种图像灰度级拉伸算法的 FPGA 实现方法, 并针对 FPGA 的特点对算法的实现方法进行了研究, 从而解决了其在导引系统应用中的实时性问题。仿真验证结果表明: 基于 FPGA 的图像拉伸算法具有运算速度快、可靠性高、功耗低等特点, 非常适合成像系统使用。

关键词: 图像灰度级拉伸算法; FPGA; 图像处理

中图分类号: TP391.9 **文献标识码:** A **文章编号:** 1673-5048(2010)02-0021-03

The Implementation of Image Gray-Lever Stretch Algorithm Based on FPGA

TANG Yao-fei

(China Airborne Missile Academy, Luoyang 471009, China)

Abstract: To adjust the gray of image, an image gray-lever stretch algorithm which is based on FPGA is presented. Aiming at the characteristic of FPGA, the implementation of the algorithm is studied, and the real-time requirement problem of imaging system is sloved. The simulation results show that the FPGA-based algorithm with features high speed, high reliability and low power consumption is very suitable for imaging system.

Key words: image gray-lever stretch algorithm; FPGA; image processing

0 引言

在精确制导武器领域成像电路系统研制过程中, 经常会对图像灰度数据做变换处理操作。能否采用一种合理有效的算法对图像灰度数据进行变换, 对成像系统的最终效果影响很大。在实际应用中, 常用一个图像灰度数据映射到不同宽度的图像灰度数据中, 并且变换后要求既能真实地反映图像数据, 又能对整幅图像作一定的有效处理。以往通过简单的数据右移缩位、左移扩位来完成图像数据宽度变换, 但是这种方法存在明显的弊端,

主要表现在图像数据的动态范围上, 例如一幅 16 位图像数据变换为 8 位图像数据, 当背景及目标灰度均较小时, 通过简单的右移 8 位方法, 不仅会造成目标信息丢失, 同时图像目标与背景的对比度将不能进行有效控制。针对这一缺点, 本文将采用图像灰度级拉伸算法来完成图像数据位数的转换, 以使小灰度目标得到增强, 图像对比度得到提高, 并且可以调整图像灰度范围。目前图像灰度级拉伸算法都采用 DSP 来实现, 其原理为: 图像数据实时传输给 DSP, DSP 接收到块完整数据后, 再对整块数据进行拉伸处理。这样势必会造成时间的延迟, 不能满足精确制导武器系统实时性的要求。

可编程逻辑器件 FPGA 具有容量大、功能强、可靠性高、开发过程投资少、周期短、可反复修改和保密性能好等特点。采用 FPGA 进行硬件电路设

收稿日期: 2008-07-09; **修回日期:** 2008-10-23

作者简介: 唐耀飞(1981-), 男, 河南濮阳人, 工程师, 研究方向是图像处理技术。

计,不但运算速度快、电路规模小、可编程性强,而且其智能化的开发工具使整个系统的设计周期大大缩短,因此,FPGA 已成为复杂数字硬件电路设计的理想之选。本文采用 FPGA 实现图像灰度级拉伸算法,可满足图像的即时拉伸处理,使其适用于导引系统的实时性要求,解决了 FPGA 硬件电路设计两个技术难点:逼近法实现除法操作和图像数据直方图统计。

1 图像灰度级拉伸算法原理介绍

图像灰度级拉伸算法表达式为

$$Y(i, j) = \frac{X(i, j) - X_{\min}}{X_{\max} - X_{\min}} \times Z_{\max} \quad (1)$$

式中: i 为图像数据行号; j 为图像数据列号; $Y(i, j)$ 为输出图像数据; $X(i, j)$ 为输入图像数据; X_{\min} 为图像数据的最小值; X_{\max} 为图像数据的最大值; Z_{\max} 为输出图像数据全‘1’对应的最大值。

采用 FPGA 实现上述操作,可以针对实时图像数据流的输入,在流水节拍的控制下,几乎实时输出拉伸后的图像数据,满足系统的实时性要求。直方图统计出一幅图像 5% 的最大值和 5% 的最小值,用图像 5% 的最大值中的最小值作为最大值 X_{\max} , 5% 的最小值中的最大值作为最小值 X_{\min} , 考虑到连续帧图像间的相关性,得到最大值和最小值,供后帧数据拉伸处理。逼近法实现算法表达式中的除法操作, Z_{\max} 为输出图像数据全‘1’对应的最大值,如输出为 8 位对应 255, 输出为 14 位对应 16 383。相邻两帧理想的无变化的图像,图像拉伸后最小值(背景)变为‘0’,最大值(目标)变为 Z_{\max} 。这样提高了图像对比度,增强目标。

2 图像灰度级拉伸算法 FPGA 实现

2.1 算法硬件设计任务分析

图像灰度级拉伸算法要求每帧求出最大值和最小值,供后帧图像数据操作处理使用。考虑到一幅图像可能有盲元等非图像信息,本文采用直方图统计法得到有效的最大值和最小值。利用可编程逻辑器件 FPGA 内部嵌入的丰富 BLOCKRAM 存储器资源实现直方图统计。实现过程中用到了一个巧妙的处理就是将数据线接到 BLOCKRAM 存储器的地址线上,利用像素时钟将数据大小对应存储空间的数据读出来自动加 1,再把得到的结果写入对应的存储空间中去,这样依次操作,完成一幅图像的直方图统计。在图像帧逆程利用 BLOCKRAM 的双口操作,分别从大地址空间中得到 5%

的最大值,小地址空间中得到 5% 的最小值,将 5% 的最大值中的最小值作为最大值,将 5% 的最小值中的最大值作为最小值,供后帧数据操作处理,这样避免了盲元成为最大值和最小值的问题。当然根据不同导引系统的要求,统计出一幅图像 5% 的最大值和 5% 的最小值这一参数是可以灵活修改的。在求出最值后下帧正程到来之前,对整个存储器的地址空间清零,然后在下帧正程时边对下帧图像数据做直方图统计,边对图像数据做拉伸操作。依次上述操作流水式处理。

此外,在式(1)中还存在除法操作,但目前的 FPGA 器件只支持分母为 2 的 n 次方的直接除法运算,而最大值减最小值一般都不会是 2 的几次方的数据,因此采用逼近法来解决此问题。首先判断出最大值减最小值的区间范围,根据此区间范围的不同,不断地使分母的大小逼近为 1,如式(2),分子的大小即为除法的结果。图 1 给出了图像拉伸算法硬件电路设计的整体系统框图。

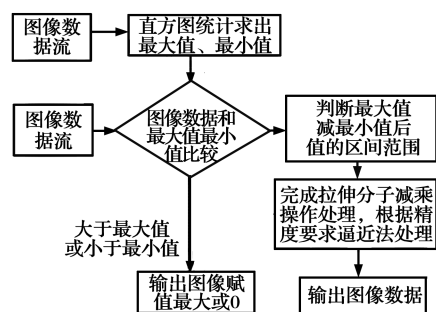


图1 图像拉伸算法硬件电路设计系统框图

从图1可以看出电路设计要完成任务包括:

- 直方图统计操作处理;
- 求出每帧的最大值、最小值;
- 根据图像数据和最大值、最小值进行分情况处理;
- 用最大值、最小值的差值进行逼近法处理;
- 生成各操作的时序控制信号。

2.2 硬件电路设计

本文选用 XILINX 公司两百万系统门规模的 VIRTEX - II XC2V2000 器件, XC2V2000 片内资源丰富,有 56 个 SelectRam BLOCK 存储模块、56 个 18×18 乘法器, 8 个 DCM 时钟管理模块、624 个 I/O 端口, 每个 BLOCK 模块容量高达 18 kbit, 器件资源丰富。

硬件电路实现重点在于用逼近法完成除法操作。由于 FPGA 数字电路中只支持移位而不支持除法操作,为了提高运算精度,采用逼近法来最大的提高运算结果精度。一个数据如 $99/5$, 如果采用近

似法移位操作, 把 5 近似为 4 则结果为 24, 这与实际结果 19.8 显然差距太大。采用逼近法使分母接近于 1:

$$\frac{99}{5} = \left(\frac{99}{8} + \frac{99}{16} + \frac{99}{32} - \frac{99}{64} \right) / \left(\frac{5}{8} + \frac{5}{16} - \frac{5}{32} - \frac{5}{64} \right) \quad (2)$$

式(2)中分母为 $\frac{65}{64}$, 比较接近于 1, 结果为 20。

一般情况下 3~4 个分式的精度都可以接受, 如果不够还可以加长为 $\frac{5}{8} + \frac{5}{16} + \frac{5}{32} - \frac{5}{64} - \frac{5}{128} \dots$, 由于式(1)中最大值和最小值的差值是未知的, 需要判断出此值的区间范围, 用到的数据流为 14 位图像数据, 如果最大值和最小值的差值为 2^n , 1, 2, 4, 8... 等的数时, 就可以直接执行移位操作。当最大值和最小值的差值不为 2^n 的数据时, 另外划分了 25 个区间如图 2 所示。

$2 < X < 4$	$48 \leq X < 64$	$768 \leq X < 1024$	$12288 \leq X < 16384$
$4 < X < 6$	$64 < X < 96$	$1024 < X < 1536$	
$6 \leq X < 8$	$96 \leq X < 128$	$1536 \leq X < 2048$	
$8 < X < 12$	$128 < X < 192$	$2048 < X < 3072$	
$12 \leq X < 16$	$192 \leq X < 256$	$3072 \leq X < 4096$	
$16 < X < 24$	$256 < X < 384$	$4096 < X < 6144$	
$24 \leq X < 32$	$384 \leq X < 512$	$6144 \leq X < 8192$	
$32 < X < 48$	$512 < X < 768$	$8192 < X < 12288$	

图 2 最大值和最小值的差值的区间划分

如图 2 所示从左到右, 从上到下按数值大小依次排序为 1, 2, ..., 25, 分别给控制信号赋值。数据这边同时分别给出图像拉伸的分子 DATA 右移 2 位(高位补零, 相当于除以 4)为 DATA2, 右移 3 位(高位补零, 相当于除以 8)为 DATA3, 依

次类推一直到右移 15 位(高位补零, 相当于除以 32 768)为 DATA15, 图像拉伸分子各种移位结果如图 3 所示。

DATA2=DATA/4	DATA9=DATA/512
DATA3=DATA/8	DATA10=DATA/1 024
DATA4=DATA/16	DATA11=DATA/2 048
DATA5=DATA/32	DATA12=DATA/4 096
DATA6=DATA/64	DATA13=DATA/8 192
DATA7=DATA/128	DATA14=DATA/16 384
DATA8=DATA/256	DATA15=DATA/32 768

图 3 图像拉伸分子各种移位结果

根据逼近法原理, 当最大值和最小值的差值位于区间 1 时, 得到结果为 DATA2 + DATA3 - DATA4。最大值和最小值的差值位于区间 2 时, 得到结果为 DATA3 + DATA4 + DATA5。最大值和最小值的差值位于区间 3 时, 得到结果为 DATA3 + DATA4 - DATA5, 依次类推进行操作使分母逼近 1, 也可增加式子长度提高精度。最后对结果阈值判断后输出拉伸数据, 根据拉伸运算占用的时钟周期数把时序和图像数据匹配上供后续电路使用。

3 算法仿真验证

本文实现的图像灰度级拉伸算法在 Xilinx 的集成开发环境 ISE 下利用 ModelSim 进行了仿真, 仿真结果正确, 并取得了预期的效果。为进一步证实该算法的正确性, 在实际红外系统中进行了验证, 得到了验证结果: 如图 4(a)为原始 14 位图像数据; (b)为采用简单的移位算法即图像数据右移 6 位完成 14 位数据到 8 位数据的变换产生的结果, 图像中目标信息丢失; (c)为图像灰度级拉伸算法



图 4 采用基于 FPGA 的图像拉伸试验结果

拉伸后的 8 位图像数据。

由图 4 可以看到, 利用可编程逻辑器件 FPGA 实现图像灰度级拉伸算法, 可以有效提高图像的对比度, 并且图像数据处理时间从 DSP 软件进行拉伸处理的 100 μ s 缩短为 200 ns。

4 结 论

利用 FPGA 实现图像灰度级拉伸算法大大提

高了图像处理速度, 满足了系统实时性的要求, 并且很好地改善了图像质量, 具有较高的应用价值。

参考文献:

- [1] 王琳, 向健勇, 刘安. 红外图像增强在 CPLD 中的实时实现技术[J]. 红外技术, 2006, 28(1): 39-42.
- [2] 唐耀飞, 李杰. 红外复杂背景滤波算法的 FPGA 实现[J]. 航空兵器, 2007, (2): 25-28.

图像灰度级拉伸算法的FPGA实现

作者: [唐耀飞](#), [TANG Yao-fei](#)
作者单位: [中国空空导弹研究院, 河南, 洛阳, 471009](#)
刊名: [航空兵器](#)
英文刊名: [AERO WEAPONRY](#)
年, 卷(期): 2010, "" (2)
被引用次数: 0次

参考文献(2条)

1. [王琳](#), [向健勇](#), [刘安](#) [红外图像增强在CPLD中的实时实现技术](#) 2006(1)
2. [唐耀飞](#), [李杰](#) [红外复杂背景滤波算法的FPGA实现](#) 2007(2)

本文链接: http://d.g.wanfangdata.com.cn/Periodical_hkbq201002005.aspx

授权使用: 陕西理工学院(sxlgxy), 授权号: 8c845c51-27df-4105-932a-9df2010a95de

下载时间: 2010年9月15日