

基于 FPGA 的快速中值滤波算法^{*}

王宇新, 贺圆圆, 郭 禾, 龙 珠

(大连理工大学 计算机科学与技术系, 辽宁 大连 116023)

摘 要: 针对传统中值滤波算法排序量多、速度慢的缺点,提出了一种基于 FPGA 的中值滤波快速算法。充分利用两个相邻滤波窗口中的相关排序信息,随着一系列新像素的移入,同时更新已有的排序信息,从而完成中值滤波处理。该算法将每个窗口查找中值的比较次数降到很低,达到了快速抑制噪声及保持图像细节的目的。

关键词: 现场可编程门阵列; 中值滤波; Verilog; 实时图像处理

中图分类号: TP391

文献标志码: A

文章编号: 1001-3695(2009)01-0224-03

FPGA-based algorithm of fast median filter

WANG Yu-xin, HE Yuan-yuan, GUO He, LONG Zhu

(Dept. of Computer Science & Technology, Dalian University of Technology, Dalian Liaoning 116023, China)

Abstract: In order to solve the problem that the speed of classical median filter was slow because of a lot of sorting, this paper proposed a new algorithm of median filter based on FPGA. It made full use of the coherence of data adjacent windows and completed the median filter processing by adding the new column of pixels while updating the rest of the arranged pixels. The algorithm can reduce the number of comparisons, and achieve the goal of noise suppression and image details keeping in a fast speed.

Key words: FPGA; median filter; Verilog; real-time image processing

中值滤波是一种非线性滤波方法,既可以消除随机噪声和脉冲干扰又可以很大程度地保留图像的边缘信息,近年来在图像平滑和数据分析与处理等多个领域中得到广泛应用。随着微电子技术工艺的迅猛发展,现场可编程门阵列(FPGA)以其可编程性、低成本性、高逻辑密度和高可靠性,得到了越来越广泛的应用。本文介绍了一种高效中值滤波器的设计及其在FPGA上的实现方案,并用Xilinx公司的开发工具ISE 8.2和Verilog硬件开发语言进行了仿真验证。

1 传统中值滤波原理

传统中值滤波算法^[1,2](TMF)的基本原理是通过将一幅图像中每一个合法像素点邻域中的像素按灰度级进行排序,然后选择该组的中间值作为输出像素值。传统中值滤波可以定义为

$$g(x, y) = \text{median}\{f(x-i, y-j) \mid (i, j) \in W\}$$

其中: $g(x, y)$ 和 $f(x-i, y-j)$ 分别为输出和输入像素灰度值; W 为模板窗口。模板窗口 W 可以取线状、方形、十字形、圆形等。

在传统算法中,需要对邻域内所有像素进行排序获得其中值。排序的过程就是对像素进行比较和交换的过程。序列中像素之间的比较次数是影响排序速度的重要原因。传统中值滤波算法随着窗口的移动,需要大量的比较运算。对于一个像素数为 N 的窗口数列,需要执行 $N(N-1)/2$ 次像素比较操作,时间复杂度为 $O(N^2)$ 。以 3×3 方形窗为例,对一幅 128×128 的灰度图像进行中值滤波处理,共需要处理16 384个像素,

对每个像素点及其邻域取中值需要进行36次比较排序运算。可见,传统中值滤波算法虽然比较简单,但是运算量大,在FPGA中实现时需要消耗大量片上资源,速度慢,无法满足实时性要求。

2 算法分析

针对传统实现中值滤波方法的缺点,文献[3]中应用快速排序算法,以滤波窗口中所有元素值的平均值为界进行划分(FSMF)。每次选取大集合以其均值进行划分,直到分解的两个集合元素个数均小于滤波窗口总元素个数的一半,再对两个子集合之一进行快速排序。虽然可以减少比较次数,但时间复杂度仍然是 $O(N \ln N)$ 。

朱捷等人^[4]提出了一种不完全排序的思想(NSMF)。对窗口像素排序时采用冒泡排序,当比较到中间位置即第 $n/2$ 个像素时停止排序即可获得中值。对于 3×3 方形窗,每次对窗口中的9个元素排序到第5个停止,需要 $8+7+6+5+4=30$ 次比较次数。

文献[5,6]提出了基于统计思想的中值滤波算法(SMF)。将包含 n 个像素模板中的某一个灰度值 D_0 与其他所有值相比较,统计出大于 D_0 和小于 D_0 的像素个数分别表示为large和little,如果两者相等则可确定该值即为所求中值,否则选择一个像素值重复上述过程,直到找到中值为止。该算法在进行统计过程中,需要将待判断像素与窗口中的所有其余像素都进行比较,利用硬件资源的可重复性定义9个比较统计单元并行

收稿日期: 2008-03-04; 修回日期: 2008-05-24 基金项目: 国家自然科学基金资助项目(60675008)

作者简介: 王宇新(1973-),男,辽宁大连人,硕士,主要研究方向为计算机系统结构、图像处理(wangyuxin@263.net);贺圆圆(1983-),女,硕士研究生,主要研究方向为计算机系统结构、图像处理;郭禾(1955-),男,教授,硕士,主要研究方向为计算机系统结构;龙珠(1985-),女,硕士研究生,主要研究方向为FPGA理论与实践。

计算,但该过程仍然需要 9 个时钟周期才能得到统计值,不适合用于实时性要求比较高的场合。文献[5]中的统计法设置了阈值 threshold,满足 $|large - little| \leq threshold$ 则可认为该值为中值近似值。虽然可以减少比较次数,但找出的像素值并不一定是滤波窗口真正中间值,从而在提高速度的同时带入了误差。

付显强^[7]提出了一种快速算法(FMF),将有 $m \times m$ 像素的二维阵列的每一列进行排序,再对每一行排序,最后取对角线像素的中值即为所求的中值。对于 3×3 方形窗,列排序运算需要 $3 \times 3 = 9$ 次比较,行排序也同样需要 9 次比较,将对角线上的三个像素进行比较取中值需要 3 次比较运算。对一组 9 个像素取中值一共需要进行 21 次比较运算。

文献[8,9]提出了利用相邻窗口间相关信息的快速算法(NWMF),充分利用当前窗口 $W(i, j)$ 其左边和上边相差一个像素位置的已排序窗口 $W(i, j-1)$ 和 $W(i-1, j)$ 中的信息获得当前窗口 $W(i, j)$ 的中值。将 $W(i, j)$ 内的像素分为 G_1 和 G_2 ,分别为含有其值最小的 $(m^2 + 1)/2$ 个像素及其值最大的 $(m^2 - 1)/2$ 个像素的集合。将新输入的一列像素进行排序,并将它们按 $W(i-1, j)$ 的中值为界归入 G_1 和 G_2 中,再对两集合中的边界像素进行比较调整,直到两集合像素个数相差 1 即可找到中值。对于 3×3 方形窗,最坏情况下可以将比较次数降低到 20 次。但是应用到 FPGA 中时,需要消耗大量的硬件资源来记录 $W(i, j-1)$ 和 $W(i-1, j)$ 的排序信息。

3 中值滤波快速算法

为了以最少的比较次数找到中值,可以充分利用已排序窗口中原有像素的序列值。本文只利用左边前一个窗口中的排序信息,通过移出一列旧像素和移入一列新像素,保留原窗口中剩余像素的排序信息,将新移入的像素与剩余像素进行比较的同时更新原窗口中剩余像素的排序信息,从而完成中值滤波处理。

设 WW 代表滤波窗口的宽度, WH 代表滤波窗口的高度, $window$ 代表窗口中的像素个数,则 $window = WW \times WH$ 。为了记录原窗口中像素的排序序列,在读入新像素时能够利用已有排序序列减少比较次数,为滤波窗口中的每一个像素都附加了一个序列值寄存器 $CR[window - 1 \cdots 0]$,用来保存当前窗口中不比它小的像素个数,新像素的序列值存放在 CN 寄存器中。窗口中像素序列值的变化范围是从 1(代表最小的像素)到 $window$ (代表最大的像素)。随着新像素进入滤波窗口,已存在于窗口中的像素和对应的序列值同时移动。寄存器 $D[window - 1 \cdots 0]$ 存放当前窗口中的像素,寄存器 ND 存放新读入的像素,比较寄存器 $C[window - 1 \cdots 0]$ 存放新像素与窗口中的相应像素的比较结果。

当窗口读入一个新像素 CN 时,将原窗口中的每一个像素 $D[window - 1 \cdots 0]$ 与新像素进行比较,若不小于 CN 则相应的比较寄存器 $C[window - 1 \cdots 0]$ 赋值为 1;否则为 0。根据比较结果,更新序列值寄存器 CN 和 $CR[window - 1 \cdots 0]$ 。 CN 的最低位置为 1,其余位由 $C[window - 1 \cdots 0]$ 的逆序生成。 $CR[window - 1 \cdots 0]$ 的最低位由对应的 $C[window - 1 \cdots 0]$ 寄存器提供,其余位由前一个像素的 $CR[window - 1 \cdots 0]$ 提供,即

$$CR[k] = \{ CR[k-1](window-2:0), C[k] \}$$

其中:(:)代表位选择;{}代表组合。这样,在任意时刻数据寄

存器 $D[k]$ 和对应的序列寄存器 $CR[k]$ 分别存储了一个输入像素的值和它与窗口中其他像素的比较结果。计算 $CR[k]$ 寄存器的各位之和就可以得到对应像素 $D[k]$ 的排序信息,从而输出中值。

图 1 表明了 $window = 5$ 时在一个周期内各个寄存器的变化情况,新像素从右侧输入,表中所示为寄存器(ND, D)的内容和对应的比较结果寄存器(CR, CN)的变化情况。寄存器 $D[4 \cdots 0]$ 存储了当前窗口中的像素值,当读入新像素 $ND = 5$ 时,将原窗口中每一个像素均与新像素进行比较,若 $D[i]$ 不小于 5,则寄存器 $C[i] = 1$;否则 $C[i] = 0$ 。 $C[3 \cdots 0]$ 提供了序列值寄存器 $CR[4 \cdots 1]$ 的最低位,其余位由 $CR[3 \cdots 0]$ 的高 8 位提供。 CN 的最低位置为 1,其余位由 $C[3 \cdots 0]$ 的逆序构成。

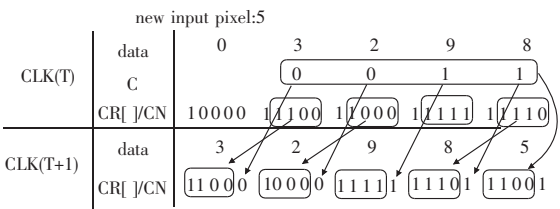


图 1 滤波算法例子(window=5)

4 硬件实现

4.1 总体设计方案

为了便于在 FPGA 硬件上实现中值滤波,本文以 3×3 窗口($WW = 3, WH = 3, window = 9$)处理 $128 \times 128 \times 8$ 的灰度图像为例进行了硬件设计实现。设计方案包括四个主要部分,如图 2 所示。

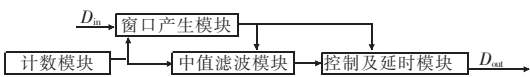


图 2 总体设计

4.2 窗口产生模块

该模块用于生成窗口数据,实现了图像数据的串入并出。在 FPGA 中定义了 9 个寄存器($w_{11}, w_{12}, w_{13}, w_{21}, w_{22}, w_{23}, w_{31}, w_{32}, w_{33}$)组成了 3×3 窗口寄存器组来暂时存放 3×3 窗口中的数据。原理图如图 3 所示。

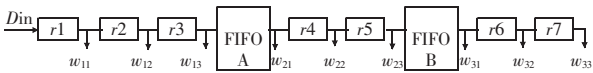


图 3 窗口生成模块原理图

在图 3 中, r 代表移位寄存器;FIFO 代表先进先出存储器。图像数据按时钟节拍从数据输入端 D_{in} 依次输入,FIFO 用来存储一行的数据,以便使 $w_{11}, w_{12}, \dots, w_{33}$ 存放的正好是 3×3 模板所对应的图像数据。当数据流不断从数据输入端输入时, 3×3 模板对应的图像数据不断地跟着变化,这就可以对图像的所有像素都进行 3×3 模板处理。

4.3 中值滤波模块

中值滤波模块的原理图如图 4 所示。算法已在第 1 章中进行了说明。随着新像素从 ND 输入,窗口像素及其对应的序列值依次左移。当一列 3 个新像素输入完毕后,此时输出有效中值数据。对于 3×3 的方形窗,每次窗口移动有 3 个新数据读入,3 个旧数据移出,只需将 3 个新数据和原窗口中剩余 6 个数据比较,比较次数减少为 $3 \times 6 = 18$ 次。

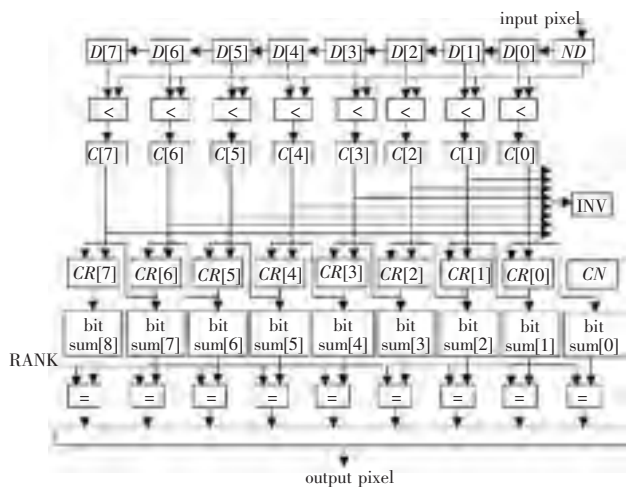


图 4 滤波原理图(window=9)

4.4 计数模块

当中值滤波执行到图像边缘时,用 3×3 的模板无法盖住图像或会盖住图像外的一部分,这样无法正确执行中值滤波模块,需要计数模块对滤波操作进行行列计数控制,在窗口移动到图像边缘时,置输出为 0。通过该模块可以确定一幅图像是否到达边缘或者是否传输完毕,从而进行边缘控制。模块结构如图 5 所示。

图 5 中,RSTn 为复位信号与全局复位信号相连;EN 为使能端;CLK 为时钟输入端;ColPos 为图像列标志;RowPos 为图像行标志。该模块只起到计数功能,用来确定数据在图像中的阵列位置。

5 测试结果及分析

本文采用 Xilinx 的 Spartan-3E 开发板 XC3S500E 芯片完成算法的设计。将本文的快速算法与文中提到的几种方法在最坏情况下的比较次数进行了比较。结果如图 6 所示。

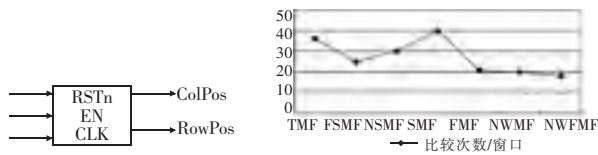


图 6 几种方法的比较结果

图 5 行列计数模块结构

本文对软硬件处理同一幅 128×128 的图片进行了对比分析。硬件板上使用 50 MB 的时钟;软件上在 Pentium4 CPU,时钟频率为 2.8 GHz,RAM 为 512 MB 的系统上采用 MATLAB 仿真实现。比较结果如表 1 所示。

表 1 中值滤波的性能比较

处理器	时钟频率/MHz	处理时间/ms
Spartan-3E	50	0.333
Intel Pentium4	2 790	30

仿真滤波效果如图 7 所示。通过比较,该快速滤波算法达到了比较好的滤波效果,图像边缘设置为 0,所以有一圈黑线。



图 7 仿真滤波效果

6 结束语

本文首先介绍了传统中值滤波算法以及已有的一些改进算法,并对它们进行了分析。在此基础上提出了一种利用相邻窗口中的排序信息得出中值的方法,将时间复杂度由 $O(N^2)$ 降到 $O(N)$,对于每个 3×3 窗口仅需要 18 次比较便可得出中值,避免求中值过程中繁琐的排序过程,提高了中值滤波的计算速度,易于在 FPGA 中实现,适合于实时图像处理。本设计现仅应用于 3×3 模板,可以进行大模板的测试,如 5×5 窗口。今后可以将该算法应用到实时图像处理应用中,以提高中值滤波的处理速度。

参考文献:

[1] ANTHONY E N. Implementation of image processing algorithms on FPGA hardware[D]. Nashville: Vanderbilt University, 2000.

[2] 李雷鸣,张焕春,张波. 一种基于 FPGA 的图像中值滤波器的硬件实现[J]. 电子工程师,2004,30(2):48-50.

[3] 张丽,陈志强,高文煊,等. 均值加速的快速中值滤波算法[J]. 清华大学学报,2004,44(9):1157-1159.

[4] 朱捷,朱小娟,贺明. 基于 FPGA 的实时性的图像处理中值滤波器设计实现[J]. 计算机测量与控制,2007,15(6):798-800.

[5] 李元帅,张勇,周国忠,等. 图像中值滤波硬件算法及其在 FPGA 中的实现[J]. 计算机应用,2006,26(6):62-63,75.

[6] 董付国,原达,王金鹏. 中值滤波快速算法的进一步思考[J]. 计算机工程与应用,2007,43(26):48-49,64.

[7] 付显强. 基于 FPGA 的图像处理算法的研究与硬件设计[D]. 南昌:南昌大学,2006.

[8] CHEN Tao, WU Hong-ren. Application of partition-based median type filters for suppressing noise in images[J]. IEEE Trans on Image Processing,2001,10(6):829-836.

[9] 刘立宏,胡可刚,刘立欣. 目标检测中的快速中值滤波法[J]. 吉林大学学报,2004,22(3):232-235.

(上接第 194 页)

[23] 胡耀华,贾欣乐. 广义预测控制综述[J]. 信息与控制,2000,29(3):248-256.

[24] BUCKINX W, VERSTRAETEN G, POEL D. Predicting customer loyalty using the internal transactional database[J]. Expert Systems with Applications, 2007,30(2):71-85.

[25] HAN Jia-wei, KAMBER M. Data mining concepts and techniques[M]. San Mateo: Morgan Kaufmann Publishers,2001.

[26] JOOS P, VANHOOF K, OOGHE H. Credit classification: a comparison of logic models and decision[C]//Proc of European Conference on Machine Learning. Chemnitz:[s. n.], 1999.

[27] 蔡越江. 论假设检验中两类错误[J]. 数理统计与管理,1999,18(3):30-35.

[28] SEBALD D J, BUCHLEW J A. Support vector machines and the multiple hypothesis test problem[J]. IEEE Trans on Signal Processing,2001,49(11):2865-2872.

[29] JONES T O, Jr SASSER E W. Determinants of customer loyalty and share of wallet in retail banking[J]. Journal of Financial Services Marketing, 1995,9(3):231-248.

[30] ZEITHAML V A, BERRY L L, PARASURAMAN A. The behavioral consequences of service quality: an examination of moderator effects in the four-stage loyalty model[J]. Journal of Marketing, 1996,60(2):31-46.

[31] 吴建彰. 产品契合度与品牌认同度对品牌延伸策略之探讨[D]. 中坜:国立中央大学,2001.