

〈系统与设计〉

基于 FPGA 的双边滤波器设计

张爱杰^{1,2,3}, 刘士建^{1,2,3}, 张瑞^{1,2,3}, 李冰^{1,2,3}

(1. 中国科学院上海技术物理研究所, 上海 200083; 2. 中国科学院大学, 北京 100049;
3. 中国科学院红外探测与成像技术重点实验室, 上海 200083)

摘要: 针对串行架构的处理器如 ARM 和 DSP 等实现双边滤波算法消耗资源过多的情况, 提出了一种基于 FPGA 加速的双边滤波去噪的实现方法。根据双边滤波算法的原理以及 FPGA 的硬件资源, 充分发挥 FPGA 在并行计算上的优势, 在滤波器的系数生成和除法操作上进行了优化。在减少了 FPGA 所消耗的资源的同时, 保证了图像去除噪声的实时性, 从而为后期的视频采集与传输系统奠定了一个良好的开端。

关键词: 双边滤波; FPGA 加速; 实时图像处理

中图分类号: TN215

文献标识码: A

文章编号: 1001-8891(2019)01-0013-05

Design of a Bilateral Filter Based on FPGA

ZHANG Ajie^{1,2,3}, LIU Shijian^{1,2,3}, ZHANG Rui^{1,2,3}, LI Bing^{1,2,3}

(1. Shanghai Institute of Technical Physics, Shanghai 200083, China;

2. University of Chinese Academy of Sciences, Beijing 100049, China;

3. Key Laboratory of Infrared System Detection and Imaging Technology, Chinese Academy of Sciences, Shanghai 200083, China)

Abstract: Considering that general serial processors, such as ARM and DSP, consume too many resources, we propose a method for a bilateral filter based on FPGA acceleration. According to the theory of a bilateral filter and the hardware resources that an FPGA can provide, we optimize the coefficient generation and division operation of the filter in the FPGA, which takes full advantage of the parallel computing capabilities in the FPGA. This ensures that the image is filtered in real-time, while reducing the resources consumed by the FPGA, and provides a good foundation for the later design of the video capture and transmission systems.

Key words: bilateral filtering, FPGA accelerating, real-time image process

0 引言

图像的产生和传输过程中, 传感器自身的噪声、量化噪声或外部环境干扰等都会对图像产生噪声干扰, 为了提高图像的识别效果, 必须对图像进行滤波处理^[1]。常用的滤波算法有高斯滤波、中值滤波和双边滤波等。而双边滤波算法结合邻域空间信息和像素值相似性对图像进行滤波处理, 在平滑滤波的同时能够保留图像丰富的边缘特征, 具有简单、非迭代、局部的特点^[2]。但是在 ARM 或者 DSP 这类的串行结构的处理器上实现实时滤波的算法将会十分耗时, 对于

实时性要求比较高的系统, 如果用软件实现会比较慢, 因此采用硬件实现比较合适^[1]。

随着芯片制程、工艺和性能的提升, FPGA 在功耗、性能和成本上的优势显著, 用户可根据实际需求设计出所需的逻辑, 也可以随时修改, 大大降低了开发难度。同时由于 FPGA 并行和流水线的特点, 可以有效提升算法的运行速度^[3]。

我们的工作主要创新内容是在 FPGA 上实现了双边滤波, 提出了该算法在结合 FPGA 硬件的优化方法, 提高图像处理的实时性, 为未来实时图像采集与处理做了良好的铺垫。

收稿日期: 2017-12-22; 修訂日期: 2018-05-11。

作者简介: 张爱杰(1992-), 男, 硕士研究生, 主要从事基于 FPGA 的图像处理。E-mail: zajlr@163.com

基金项目: 国家“十二五”国防预研项目, 上海物证重点实验室基金(2011xcwzk04), 中国科学院青年创新促进会资助(2014216)。

1 双边滤波原理

Tomasi 和 Manduchi 于 1998 年提出图像双边滤波算法 (Bilateral filtering, BF) [4]。双边滤波算法是基于高斯滤波方法, 旨在解决高斯滤波带来的图像边缘模糊问题, 利用强度的变化来保存边缘信息^[5]。相比于其他滤波器, 双边滤波器在平滑噪声的同时也保留了图像的边缘信息, 因此非常适用于图像的噪声处理。

双边滤波器的数学公式为:

$$g(i, j) = \frac{\sum_{k, l} f(k, l) \times c(k, l, i, j) \times s(k, l, i, j)}{\sum_{k, l} c(k, l, i, j) \times s(k, l, i, j)} \quad (1)$$

式中: $c(k, l, i, j) = e^{-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2}}$, $s(k, l, i, j) = e^{-\frac{\|f(i, j) - f(k, l)\|^2}{2\sigma_r^2}}$;

(i, j) 为中央点; (k, l) 为中央点的邻域像素; $f(i, j)$ 为中央点的像素值; $f(k, l)$ 为中央点 (i, j) 的 (k, l) 邻域的像素值; σ_d 、 σ_r 两个系数分别决定了空间位置对中央点的

影响力度和像素差值对中心点的影响力度。

双边滤波器的内核由 $c(k, l, i, j)$ 空间域的滤波核和 $s(k, l, i, j)$ 值域滤波核组成, 前者表示距离中心像素越远的像素占得权重越少, 后者表示灰度差异越大的像素占的权重越少, 因此, 双边滤波算法能更好地保护边缘信息^[2]。

2 双边滤波器硬件实现

滤波器的实现过程需要复杂的指数运算和除法运算, 这两个部分将消耗许多的逻辑资源, 因此本文提出了基于查找表的快速指数运算和移位相减的除法实现。指数计算需要涉及浮点数运算, 但是根据指数的性质可知 $0 < c(k, l, i, j) \leq 1$, $0 < s(k, l, i, j) \leq 1$, 因此可用只有小数部分的定点数代替浮点数来表示, 这样可以直接使用整数乘法器和移位操作实现指数的运算, 可以节省很多的逻辑资源。算法实现的总体框图如图 1 所示, 输入的图像 I 经过 $N-1$ 个行缓存后进入 $N \times N$ 的窗口寄存器中, 经过行计算单元生成滤波器核系数与邻域像素值的积 $c(k, l, i, j) \times s(k, l, i, j) \times f(k, l)$ 及核系数 $c(k, l, i, j) \times s(k, l, i, j)$, 最后对这两项分别求和后再相除, 得到滤波后的像素值 Q 。

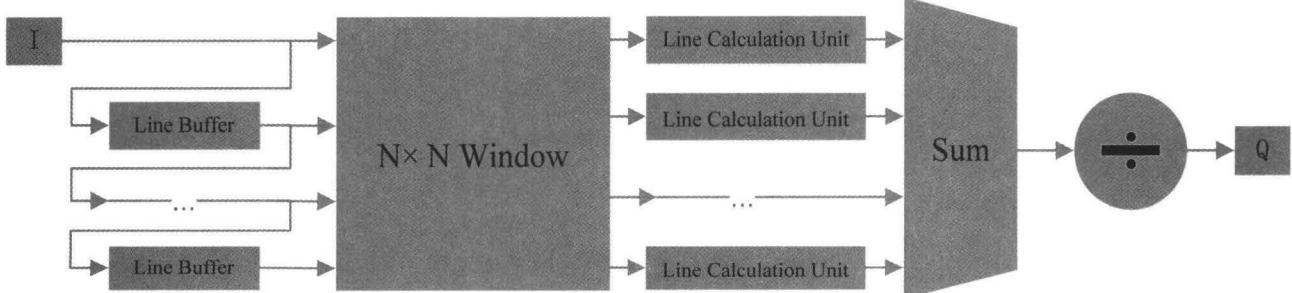


图 1 算法实现总体概览

Fig.1 Overview of algorithm implementation

FPGA 内部拥有丰富的存储器资源, 这些存储器资源可以配置成 ROM, RAM 和 FIFO 等, 在处理指数时, 可以预先将计算好的结果存入 ROM 中, 待需要的时候可快速地从 ROM 中读取出来, 这样不仅可以节省资源, 同时还能提高计算速度。对于空间域的

滤波器核 $c(k, l, i, j) = e^{-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2}}$, 可以直接把

$e^{-\frac{(i-k)^2 + (j-l)^2}{2\sigma_d^2}}$ 的计算结果存入 ROM 中, 在使用 17 位的定点浮点数的情况下, 消耗 $k \times l \times 17$ bit 的存储器资源。但对于像素值位宽为 n 的值域的滤波器核

$s(k, l, i, j) = e^{-\frac{\|f(i, j) - f(k, l)\|^2}{2\sigma_r^2}}$, 直接将计算结果存入 ROM

中则会消耗 $2^n \times 17$ bit 的存储器资源。当位宽为 8 时, 需要 4352 bit 的资源, 但是当位宽为 16 bit 时, 存储器资源的消耗将达 1.1 Mbit, 超过了很多 FPGA 所能提供的存储器资源, 因此需要对值域部分的算法进行改进, 减少资源的消耗。

值域的滤波器核系数计算部分的简化原理就是将指数运算的加法运算改为指数的乘法运算, 假设 $f(i, j)$ 的位宽为 16, 则 $\|f(i, j) - f(k, l)\|^2$ 的位宽为 32, 将其拆分为 4 个 8 位宽的数表示:

$$\|f(i, j) - f(k, l)\|^2 = 256^3 \times a + 256^2 \times b + 256 \times c + d \quad (2)$$

式中: a, b, c, d 分别为 $\|f(i, j) - f(k, l)\|^2$ 的 24 到 31 位, 16 到 23 位, 8 到 15 位和 0 到 7 位。则值域的滤波器核系数计算公式可以表示为:

$$e^{-\frac{\|f(i,j)-f(k,l)\|^2}{2\sigma_r^2}} = e^{-\frac{256^3*a+256^2*b+256*c+d}{2\sigma_r^2}} \quad (3)$$

由指数运算的性质 $e^{a+b} = e^a \times e^b$ 继续转变为指数乘积的形式:

$$e^{-\frac{\|f(i,j)-f(k,l)\|^2}{2\sigma_r^2}} = e^{-\frac{256^3*a}{2\sigma_r^2}} \times e^{-\frac{256^2*b}{2\sigma_r^2}} \times e^{-\frac{256*c+d}{2\sigma_r^2}} \times e^{-\frac{d}{2\sigma_r^2}} \quad (4)$$

这样值域滤波器核的系数就等于 4 个 8 位输入的查找表存储的 $e^{-\frac{256^3*a}{2\sigma_r^2}}$, $e^{-\frac{256^2*b}{2\sigma_r^2}}$, $e^{-\frac{256*c+d}{2\sigma_r^2}}$, $e^{-\frac{d}{2\sigma_r^2}}$ 的结果的乘积。相比于 16 位的查找表, 存储器资源消耗减少至 1/64。同时, 由于指数运算的衰减很快, e 的 -16 次方用 17 位的定点数表示已经为 0 了, 因此上述的计算中 a, b 项可以省略掉, 当 a 或 b 大于 0 时 $s(k, l, i, j)$ 直接取 0。最终的等效公式为:

$$e^{-\frac{\|f(i,j)-f(k,l)\|^2}{2\sigma_r^2}} = \begin{cases} e^{-\frac{256c}{\sigma_r^2}} \times e^{-\frac{d}{\sigma_r^2}} & (a=0 \& \& b=0) \\ 0 & (a>0 \& \& b>0) \end{cases} \quad (5)$$

图 2 表示为值域和空间域的滤波器核函数生成的原理框图。图中 $f(k, l)$ 为邻域像素点的像素值, $f(i, j)$ 为中央像素点的像素值。空间域的系数可由邻域像素点相对中心点的空间位置直接查找得到, 值域的系数由邻域像素值和中心像素值的差平方后, 对低 16 位分高低 8 位查找后相乘得到。最后对两部分的系数相乘得到滤波器的核系数 $w(i, j, k, l)$ 。由于 FPGA 实现加法操作是不需要耗费额外的时钟周期的^[6], 因此滤波器核函数生成器只需要 3 个时钟周期的延时就可以输出结果了, 计算速率大大提高。

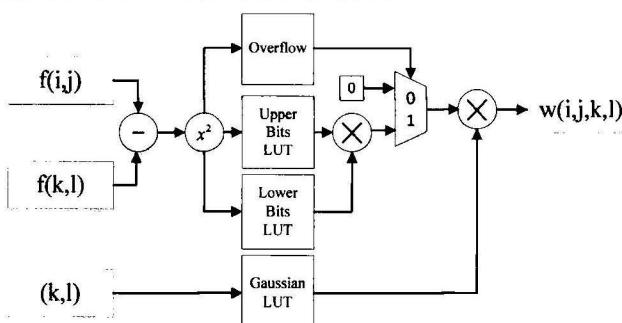


图 2 滤波器核函数生成器的硬件实现

Fig.2 Hardware implementation of filter Kernel function generator

双边滤波器分别对长宽为 k, l (k, l 为奇数) 的窗口内的所有像素点进行上述的运算, 求出窗口内每个邻域像素点的滤波系数 $W(k, l, i, j)$, 最后对窗口内的像素点进行加权平均, 得出结果。FPGA 可以使用 $k \times l$ 个上述的核函数生成器对 $k \times l$ 的矩形框内的像素点同时进行计算, 这样只需一个时钟周期就可以计算出来, 但是这样将消耗很多的 FPGA 资源。综合考虑

面积和速度的关系, 可以考虑使用 k 个核函数生成器, 每个核函数生成器对所在行的 l 个像素进行计算, 最后把计算的结果累加起来。这种做法可以减少 l 倍的核函数生成器, 但是每个生成器的计算速度也要增加 l 倍, 这样才能在一个像素周期内完成像素点的滤波而不影响滤波效果。同时, 在平衡算法性能和运算速度之间的关系时, 需要综合考虑 FPGA 的逻辑资源、速度等级以及功耗等的限制, 在满足算法性能要求的同时调整消耗的 FPGA 资源和工作频率, 使得整体的功耗最低, 性能稳定性最强。

图 3 是滤波窗口的宽度 $N=3$ 时的双边滤波器原理框图, 图中每个核函数生成器 Function Generator 分别对每行的像素点进行计算, 生成器的输入由计数器 Counter 进行控制, 同时计数器也控制每行累加器的清零。最后将每行的产生的结果 $w(i, j, k, l) \times f(k, l)$ 和 $w(i, j, k, l)$ 相加再相除, 得到最终的结果。

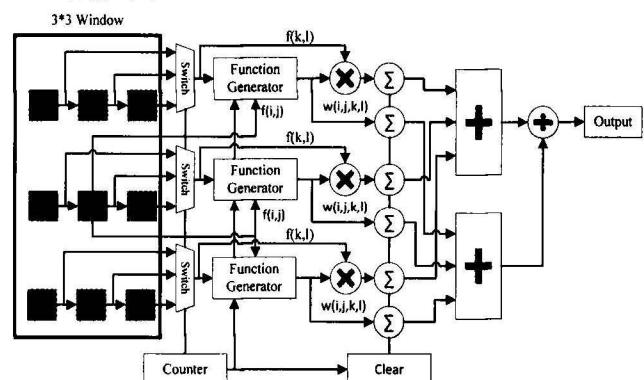


图 3 $l=3$ 时的双边滤波器

Fig.3 Bilateral Filter with $l=3$

FPGA 的四则运算中, 除法是最为复杂的, 实现除法将消耗很多的资源而且需要的时钟周期随着位宽的增加而增加, 不过双边滤波器的除数 $sum(w)$ 和被除数 $sum(f \times w)$ 之间的位宽刚好差了 f 的位宽, 因此 $sum(w)$ 最多只需右移 f 的位宽就可以计算出商, 因此可以采用移位相减的方法提取出商, 这样只需要 f 的位宽个周期过后就可以得出结果。

如图 4 所示, 首先分别将被除数 A 加载到寄存器中, 将除数 B 加载到移位寄存器中, 移位寄存器最开始右移 16 位, 然后每计算一次向左位移一位。将寄存器和移位器中的数相减并比较, 如果不小于零则比较器输出 1 给串并转换模块, 同时选择器选通相减后的数到差值寄存器中, 否则, 比较器输出 0 给串并转换模块, 寄存器中的值保持不变。这样经过 16 个周期的计算过后就可以得到 A 除以 B 的值了。

3 实验结果

采用 Xilinx 的 FPGA xc7a50t-1ftg256 对以上的算

法进行硬件实现,总共消耗了1695个寄存器资源,1838个LUT资源,5个36kB RAMS资源和15个乘法器资源。

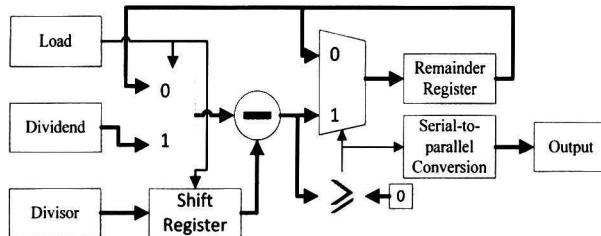


图4 移位相减除法器

Fig.4 Shift Subtraction Divider

在FPGA上实现的双边滤波,验证了两种图像源(灰度图像a与红外图像c)如图5所示,两种图像在经过滤波处理后,都展现出较好的视觉效果,图中明显看到噪声被滤除。

采用峰值信噪比(peak signal to noise ratio, PSNR)和结构相似性指数(structural similarity index, SSIM)对双边滤波器的效果进行评估,其中PSNR的计算方法为:

$$\text{PSNR} = 10 \times \lg \left(\frac{(2^n - 1)^2}{\text{MSE}} \right), \quad (6)$$

$$\text{MSE} = \frac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} \|I(i, j) - K(i, j)\|^2$$

式中: $I(i, j)$ 待测图像的像素值; $K(i, j)$ 是原图的像素值; n 是像素值的位宽; H, W 为图像的高度和宽度。PSNR可以反映出两幅图像之间像素点的误差大小,PSNR越大,两幅图之间的误差越小。

结构相似性的计算方法为:

$$\text{SSIM} = \frac{2u_x u_y + C_1}{u_x^2 + u_y^2 + C_1} \times \frac{2\sigma_x \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \times \frac{\sigma_{xy} + C_3}{\sigma_x \sigma_y + C_3} \quad (7)$$

式中: u_x, u_y 分别表示图像 X, Y 的均值; σ_x, σ_y 为图像

X, Y 的方差, σ_{xy} 表示图像 X, Y 的协方差。

$$u_x = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W X(i, j) \quad (8)$$

$$\sigma_x^2 = \frac{1}{H \times W - 1} \sum_{i=1}^H \sum_{j=1}^W (X(i, j) - u_x)^2 \quad (9)$$

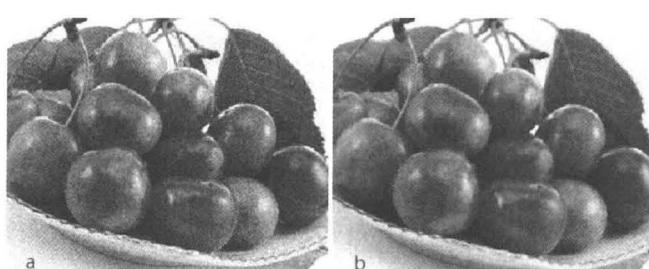
$$\sigma_{xy} = \frac{1}{H \times W - 1} \sum_{i=1}^H \sum_{j=1}^W (X(i, j) - u_x)(Y(i, j) - u_y) \quad (10)$$

式中: H, W 表示图像的高度和宽度,SSIM从亮度、对比度和结构3个方面评估两幅图像的相似性,SSIM越大表示两幅图片的相似度越高。

实验结果表明在FPGA平台上可以对8位50Hz的640×512分辨率的红外图像(包括灰度图像)进行滤波去噪,滤波效果良好,消除了由大部分噪声带来的高频成分,更由于FPGA出色的并行计算能力,使得系统满足实时性的要求,可用于实时图像处理系统。从表1可以看到,同样的算法在计算机上实现单帧耗时已经超过了1.2s,远不能满足实时性的要求。当然由于我们没有增加一些细节补强的算法,使得图像的细节被弱化,这也是后期需要改进的地方。

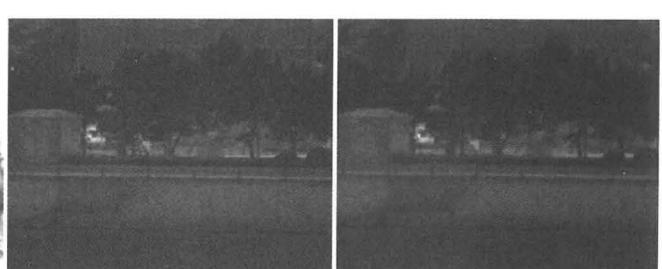
4 结论

双边滤波器具有平滑图像的同时还能保持边缘的特征,广泛应用于图像处理领域,本文充分利用FPGA的资源,结合双边滤波器的算法特点,改进了双边滤波算法在FPGA上的实现方法,实现了双边滤波器的加速,提高了图像处理系统的处理性能和实时性。我们在设计系统时只考虑了对算法的验证,在工程实现中还有一些不足,比如算法没有实现参数实时调节,这也是后期工作的重点。



(a) 包含高斯噪声的灰度图像
(a) Grayscale image with Gaussian Noise
(b) FPGA 处理后的(a)图像
(b) Grayscale image after FPGA processing

图5 使用FPGA实现双边滤波算法的效果



(c) 包含高斯噪声的红外图像
(c) Infrared image with Gaussian Noise
(d) Infrared image after FPGA processing

Fig.5 Results of bilateral filter using FPGA

表 1 两张图像在 PC 机与 FPGA 上的处理对比

Table 1 Comparison of two images processed on PC and FPGA

	Processing time in PC/s	Image with noise PSNR/dB	Image after BF PSNR/dB	Processing time in FPGA/ms	Image with noise SSIM	Image after BF SSIM
Infrared image	1.252	29.32	35.33	20	0.74	0.97
Grayscale image	1.282	28.51	30.17	20	0.90	0.98

参考文献:

- [1] 王萍, 白光远, 唐晓燕. 基于 FPGA 的图像中值滤波算法的优化及实现[J]. 电子与电脑, 2009(10): 94-96.
WANG Ping, BAI Guangyuan, TANG Xiaoyang. Median filter algorithm optimize and execute based on FPGA[J]. *Compotech China*, 2009(10): 94-96.
- [2] 梁广顺, 汪日伟, 温显斌. 基于双边滤波与非局部均值的图像去噪研究[J]. 光电子·激光, 2015, 26(11): 2231-2235.
LIANG Guangshun, WANG Riwei, WEN Xianbin. Image denoising based on bilateral filtering and non-local means[J]. *Journal of Optoelectronics · Laser*, 2015, 26(11): 2231-2235.
- [3] 白书华. 基于 FPGA 的实时图像采集和去噪系统设计[J]. 航天器环境工程, 2016, 33(2): 163-166.
BAI Shuhua. Design of real-time image acquisition and denoising based on FPGA[J]. *Spacecraft Environment Engineering*, 2016, 33(2): 163-166.
- [4] 李红平, 陈祥望, 徐丹. 基于双边滤波的 HDR 图像色调映射算法[J]. 云南大学学报: 自然科学版, 2014, 36(3): 341-346.
LI Hongping, CHEN Xiangwang, XU Dan. HDR image tone mapping algorithms based on bilateral filtering[J]. *Journal of Yunnan University: Natural science*, 2014, 36(3): 341-346.
- [5] 田小平, 姜薇, 吴成茂. 基于双边滤波的低亮度图像增强算法[J]. 西安邮电大学学报, 2016, 21(4): 43-47.
TIAN Xiaoping, JIANG Wei, WU Chengmao. Low luminance image enhancement based on bilateral filtering[J]. *Journal of Xi'an university of Posts and Telecommunications*, 2016, 21(4): 43-47.
- [6] 梁超, 马天翔. 基于黑体标定的红外图像非均匀性校正系统设计[J]. 中国光学, 2016, 9(3): 385-393.
LIANG Chao, MA Tianxiang. Design of infrared imaging nonuniformity correction system based on black body calibration[J]. *Chinese Optics*, 2016, 9(3): 385-393.