

一种改进的 Bayer 图像彩色恢复差值算法

尹 勇 胡 磊

(武汉理工大学信息工程学院 武汉 430070)

摘 要 图像传感器采集到的都是经过滤波镜头后的 Bayer 格式马赛克图像,为了恢复丢失掉的色彩信息,必须经过插值计算。针对图像采集系统处理的实时性要求,结合自适应插值法和三角函数法则,提出了一种适用于硬件实现的新算法。给出对算法性能的评估数据和 MATLAB 仿真图片,对比了算法的处理效果,最后基于实际应用的需求对该算法的硬件实现难度进行了分析。实验结果证明,改进算法在图像边缘的处理上效果更好,基于硬件的实现方式也能保证系统的实时性。

关键词 插值算法,彩色滤波阵列,Bayer 格式,硬件,FPGA

中图分类号 TP391.41 **文献标识码** A

Improved Interpolation Algorithm for Demosaicing Bayer Pattern Images

YIN Yong HU Lei

(School of Information Engineering, Wuhan University of Technology, Wuhan 430070, China)

Abstract Image sensors can only gain the Bayer pattern pictures which are filtered by CFA. In order to get the full colour information, the other two missing colour samples must be interpolated. An algorithm based on Hamilton-Adam interpolation and trigonometric function rule was presented, which can satisfy the demand of real-time system, and can be implemented by hardware easily. Then statics and simulation pictures produced by matlab were given to estimate this algorithm, and the difficulty of the implement for the algorithm on hardware was discussed at the end. The results of experiments proved that the algorithm can restore the edges of the pictures better, and can provide satisfying speed when implied on hardware.

Keywords Interpolation algorithm, Colour filter array, Bayer pattern, Hardware, FPGA

1 引言

目前的数字图像采集系统都采用一块 CCD/CMOS 作为接收图像的传感器,在感光表面覆盖颜色滤波阵列(CFA),每点仅允许通过一种颜色分量,产生一种马赛克图像。对经过滤波阵列 CFA 的原始马赛克图像进行彩色恢复和图像重建,在图像压缩、解码和数字图像采集系统彩色实时显示方面有着非常重要的应用。由于多数彩色恢复算法都是基于 DSP 来实现,受限于 CPU 的运算能力,因此,充分利用 FPGA 的并行运算优势,选择和研究适用于硬件的算法,成为了当今图像处理的一个发展趋势。

关于去马赛克的颜色插值方法有很多,常用的算法有邻域插值法、线性插值法、立方卷积插值法、神经网络插值法等。选用方法越复杂,插值产生的图像质量越高,但需要的处理时间也会增加。其中,邻域插值、线性插值法能在图像相对平滑的区域得到较好的结果。但是,在图像高频区域,如图像中的边缘处和色差大的区域,走样现象不能得到很好的控制;而立方卷积插值法和神经网络插值算法在计算过程中会用到大量的卷积等复杂运算,比较适合于计算机处理,但采用硬件实

现的难度很大。因此,本文选择对线性插值算法进行改进,在保留其运算快的特点的同时尽量提高图像质量。

为了更好地解决边界问题,本文借鉴文献[2]中的三角函数法则,在插值过程中引入边界方向的概念。针对 FPGA 的特点设计了一种取近似的简化计算方法;在边界的判断和检测时,则采用了一种平滑的仲裁方式,以避免插值过界。

2 适用于硬件的改进算法

考虑相关性的差值算法,其核心思想是通过研究邻域像素的水平 and 垂直的相关性,选择在相关性大的方向的两个像素进行线性插值,在相关性一样的情况下,此法与线性插值一致。图 1(a)是一个 3×3 窗口的 Bayer 格式彩色阵列,由图中可以看出,绿色像素点占到了图像的一半。由于绿色分量数量最多,其中包含的景象息也比红色分量和蓝色分量多,因此一般将绿色分量作为考察相关性的根据,首先恢复出绿色分量的信号,再根据不同颜色通道的相关性恢复出红色和蓝色信号。

由于人的视觉系统对色彩的变化以及边界信息敏感,错误的色彩、模糊的边界都会直接影响人眼观察到的图片的质

到稿日期:2009-05-22 返修日期:2009-08-05

尹 勇(1976—),男,硕士生导师,研究方向为智能制造、网络化控制与节能,E-mail: yiyng_hust@126.com; 胡 磊(1985—),男,硕士生,研究方向为嵌入式智能控制。

量,因此高效的插值算法都会结合边界进行插值。基于边界的插值算法均是采用沿水平或是竖直方向进行插值,即认为边界位于最理想的方向,但实际上,边界是有一定的角度的。更精确的做法是计算出每个像素点在水平和竖直方向的梯度,然后根据梯度值分别在两个方向给出加权系数,再计算出最终的G通道的插值结果。例如文献[2]中的三角函数法就是这样处理的。然而,由于三角函数的计算过程中有开根号和除法运算,在FPGA硬件系统中实现起来会相当困难。本文通过一种求近似方法,对三角函数的计算进行简化。



图1 Bayer格式及Sobel算子矩阵

首先引入方向标志位的概念。由于每个像素与它邻域内的像素具有相关性,我们认为每一像素都存在方向标志位,定义符号 $flag_{mn}$ 为位于第 m 行 n 列的像素 P_{mn} 的标志位。如果 $flag_{mn}=1$,则认为像素 P_{mn} 与水平方向的像素的相关性大,在该方向存在边界的概率比较大;反之 $flag_{mn}=0$,则认为该像素在竖直方向存在边界的概率比较大。在恢复绿色分量时,综合考虑亮度分量与色度分量的影响来确定边界方向。以图1的 B_{44} 像素点为例,分别计算其水平与竖直方向的梯度 G_h, G_v 。

索贝尔算子^[6](Sobel operator)是图像处理中的算子之一,主要用作边缘检测。在技术上,它是一离散性差分算子,用来运算图像亮度函数的梯度之近似值。在图像的任何一点使用此算子,都会产生对应的梯度矢量或是其法矢量。Sobel算子包含两组 3×3 的矩阵,如图1(b)和图1(c)所示,分别对应横向及纵向,将之与图像作平面卷积,即可分别得出横向及纵向的亮度差分近似值。如果以 A 代表原始图像, G_x 及 G_y 分别代表经横向及纵向边缘检测的图像,则计算公式如下:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} A, G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} A \quad (1)$$

套用索贝尔算子,则像素点 B_{44} 处的梯度计算表达式为:

$$G_h = R_{35} + R_{55} + 2G_{45} - (R_{33} + R_{53} + 2G_{43}) \quad (2)$$

$$G_v = R_{53} + R_{66} + 2G_{54} - (R_{33} + R_{35} + 2G_{34}) \quad (3)$$

如果 $G_h < G_v$,取 $flag=1$,否则 $flag=0$ 。使用上面的规则,可以获得每个像素的方向标志位,判断出边界可能存在的方向。但是,本文并不采用方向标志位作为判断边界的标准,而是采用一种平滑的仲裁法来检测边界。对于 B_{44} ,取其邻域内 3×3 窗口,统计出该窗口内 $flag$ 的总数:

$$total_flag_{44} = flag_{33} + flag_{34} + flag_{35} + flag_{43} + flag_{44} + flag_{45} + flag_{53} + flag_{54} + flag_{55} \quad (4)$$

由式(4)可知, $total_flag_{44}$ 的取值范围在 $0 \sim 9$ 之间,且为整数。为简化计算过程,本文给标志位的每个取值分配一个“近似”的角度值 θ ,也就是将 $0 \sim 90$ 度的角度进行9等分,每10度为一个步进,然后用三角函数法确定水平方向和竖直方向的加权系数,最后计算得到插值结果。

这样,第4行第4列像素的绿色分量计算表达式为:

$$G_{44} = \left(\frac{G_{43} + G_{45} + B_{33}}{2} - \frac{B_{42} + B_{46}}{4} \right) \sin\theta +$$

$$\left(\frac{G_{34} + G_{54} + B_{44}}{2} - \frac{B_{24} + B_{64}}{4} \right) \cos\theta \quad (5)$$

在FPGA中进行浮点乘法运算比较复杂,占用资源多,会对运行速度造成影响。因此考虑采用一种比较简单也比较常用的办法,将浮点乘法运算转换成整数运算。例如,当 $\theta=30^\circ$ 时, $\sin\theta=0.5$, $\cos\theta \approx 0.866$,则取水平方向的加权系数 e_h 为500,竖直方向加权系数 e_v 为866,计算得到的结果再左移10位(即除以1024),就可以得到近似结果。该点绿色分量的计算表达式变为:

$$G_{44} = \left[\left(\frac{G_{43} + G_{45} + B_{33}}{2} - \frac{B_{42} + B_{46}}{4} \right) e_h + \left(\frac{G_{34} + G_{54} + B_{44}}{2} - \frac{B_{24} + B_{64}}{4} \right) e_v \right] \ll 10 \quad (6)$$

其中, e_h, e_v 的取值和 $total_flag$ 以及 θ 的对应关系如表1所列。

表1 标志位与加权系数对应表

total_flag	θ	e_h	e_v
0	0	0	1024
1	10	173	984
2	20	342	940
3	30	512	866
4	40	642	766
5	50	766	642
6	60	866	512
7	70	940	342
8	80	984	173
9	90	1024	0

由表1可知,水平方向加权系数 e_h 的值为对应角度 θ 的正弦值乘以1000后取整的结果(部分值稍微调整),竖直方向加权值 e_v 的处理方法相同。

至此就已经完成了对图片绿色分量的恢复,接下来要恢复R通道和B通道的颜色分量。在对R和B通道插值时,用到的基本思想是基于色差的算法。假设现在需要恢复R通道信息,根据阵列的排列格式,总共有3种不同的情况。

像素点 G_{43} 在上下两个位置有已知的R色彩值,其红色分量计算式为:

$$R_{43} = \frac{R_{33} + R_{53} + G_{43}}{2} - \frac{G_{33} + G_{53}}{4} \quad (7)$$

在像素点 G_{34} 中插值 R_{34} ,在左右两个位置有已知的R色彩值:

$$R_{34} = \frac{R_{33} + R_{35} + G_{34}}{2} - \frac{G_{33} + G_{35}}{4} \quad (8)$$

还有一种情况如 B_{44} ,该点的两个对角线上有已知的R色彩值,由于前面的G通道恢复过程中已经考虑过边界的方向和角度问题,此时就不再进行梯度的计算和比较,直接取邻域像素点信息的平均值:

$$R_{44} = \frac{R_{33} + R_{35} + R_{53} + R_{55}}{4} - \frac{G_{33} + G_{35} + G_{53} + G_{55}}{4} + G_{44} \quad (9)$$

B通道与R通道的处理一致,经过上面两个步骤的处理,就完成了图片的插值过程。

3 算法评估与MATLAB仿真

3.1 实验数据及仿真图片分析

MSE与PSNR是图像处理中评价重构图像质量最常用的方法。其中,均方误差MSE(mean squared error)定义为:

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (f_a(i,j) - f_b(i,j))^2 \quad (10)$$

在应用中,均方误差经常根据与它等价的倒数度量,即峰值信噪比 PSNR(peak signal to noise ratio)来引用:

$$PSNR = 10 * \log_{10}(255^2 / MSE) \quad (11)$$

其中, M 和 N 代表图像的尺寸为 M 行 N 列, $f_a(i,j)$ 和 $f_b(i,j)$ 分别代表原始图像和重构图像在第 i 行、第 j 列的强度分量。这里, PSNR 是单个通道的值,即真彩色图像可以分成 R, G, B 三通道独立比较,它反映了原始图像与重构图像相符合的程度, PSNR 越大,则两者越接近。表 2 给出了 3 种不同插值算法对实验图片进行处理后的 PSNR 值。

表 2 3 种算法性能比较

差值方法	B 分量 PSNR 值	G 分量 PSNR 值	R 分量 PSNR 值
双线性插值算法	30.58	36.18	31.44
自适应插值算法	33.24	39.86	36.48
改进的差值算法	33.53	41.18	36.86

本文给出了不同算法对同一图片的插值结果,其采用 MATLAB 软件仿实现,如图 2 所示。结合实验数据和图片效果可以看出,普通的双线性插值算法效果较差,会产生比较多的伪彩色条纹;基于边界的自适应插值算法由于考虑了颜色的相关性,质量有很大改善;而本文提出的算法与自适应算法相比,在对每个像素点做插值运算时都参照了整个领域内的 8 个像素点,色彩的过渡更平滑,从而进一步减少了伪彩色斑纹(例如图 2(a)中间两扇窗分界部分);仔细对比图 2(c)和图 2(b)可以看出,图 2(c)上的细密窗叶纹路更清晰,证明了本文算法对图像高频部分恢复效果更好。

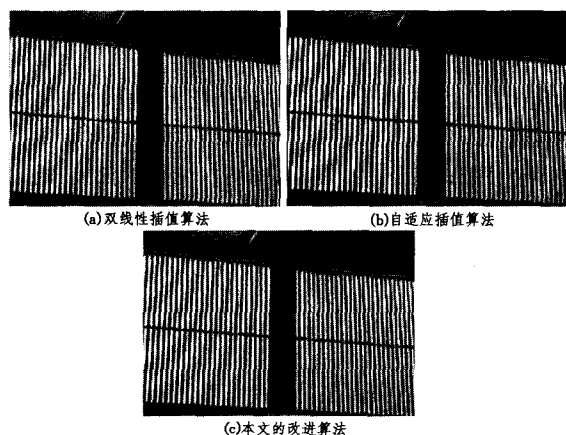


图 2 几种不同算法的仿真结果

3.2 硬件可行性评估

(上接第 207 页)

- [7] Guan Yan-yong, Wang Hong-kai. Set-valued information systems [J]. Information Sciences, 2006, 176: 2507-2525
- [8] Slowinski R, Vanderpooten D. A generalized definition of rough approximations based on similarity [J]. IEEE Transactions Data Knowledge Engineering, 2000, 2: 331-336
- [9] 陈子春, 秦克云. 区间值信息系统在变精度相容关系下的属性约简[J]. 计算机科学, 2009, 36(3): 163-165
- [10] 陈子春, 秦克云. 集值信息系统基于变精度相容关系的知识约简[J]. 计算机工程与应用, 2008, 44: 27-29
- [11] Zhu W, Wang Fei-yue. Some results on the covering generalized rough sets [J]. Pattern Recognition Artificial Intelligence, 2002,

本文拟采用 Altera 公司的 cyclone II 系列芯片来实现该算法。从存储容量来说,由于改进的插值法是基于 3×3 掩模窗口来计算的,计算一个像素点会用到相邻 3 行内的像素数据,而图像数据是连续输入的,故需要缓存 3 行数据。输入缓冲可以采用 3 个 16kbit 的片内 RAM(称缓存 1 行数据的 RAM 块为 1 条 line buffer),用于存储 3 行数据,输出缓冲用 1 个 4kbit 的 FIFO,而为了存储 $flag$ 标志变量,需要再增加 1 条 buffer,加权系数的取值固定,可直接采用查表方式实现。这样,总共需要的片内 RAM 在 70k 以下,不到 cyclone II 系列芯片内存总量的 30%,故无须使用外部存储器。在计算复杂性方面,本文提出的算法仅限于四则运算,由于 FPGA 芯片内部提供了大量的乘法器资源,文中所提到的 10 位以内的乘法运算可以顺利完成,不会成为系统运行的速度瓶颈。因此,本文提出的算法在硬件上是完全可行的。

结束语 从第 3 节的分析可以看出,虽然本文在对三角函数的求近似简化方面不甚精确,但相对于只做水平和竖直两个方向插值的自适应插值算法而言,本文提出的算法对图像高频部分恢复效果更好,能进一步减少伪彩色斑纹;同时算法描述简单,容易被硬件理解和实现,基本达到了预期的目的。

参考文献

- [1] Bayer B E. Color image array: U. S. Patent No. 3917065 [P]. 1976
- [2] 刘方, 杜平安. 基于 Bayer 彩色滤波阵列插值算法的研究 [M]. 西安: 电子科技大学, 2006
- [3] Cok B E. Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal: U. S. Patent, No. 4642678 [P]. 1987
- [4] 刘晓松, 杨新, 文俊. 一种用于数字图像传感器的彩色插值算法 [J]. 中国图像图形学报, 2003, 8A(5): 516-521
- [5] Hamilton J F, Adams J E. Adaptive Color Plan Interpolation in Single Sensor Color Electronic Camera: United States Patent, 5629734 [P]. 1997
- [6] Gonzalez R C, Woods R E. Digital Image Processing Second Edition [M]. 北京: 电子工业出版社, 2005: 33-43, 225-239, 467-475
- [7] 吴心然, 赵群飞, 山元康裕. 基于小波的 Bayer 颜色滤波阵列上的插值 [J]. 计算机工程与应用, 2005, 26: 64-65
- [8] 李奇, 徐之海, 冯华君. 用于数字相机的 CCD 颜色插值算法研究 [J]. 激光与红外, 2002, 32(5): 300-302
- [9] Murean D D, Park T W. Demosaicing using Optimal Recovery [J]. IEEE transactions on Image Processing, 2002

5: 6-13

- [12] Zhu W, Wang Fei-yue. Reduction and axiomization of covering generalized rough sets [J]. Information Sciences, 2003, 152: 217-230
- [13] Tsang E C C, Chen De-gang, Yeung D S. Approximations and reducts with covering generalized rough sets [J]. Computers & Mathematics with applications, 2008, 56: 279-289
- [14] 张文修, 梁怡, 吴伟志. 信息系统与知识发现 [M]. 北京: 科学出版社, 2003
- [15] Skowron A, Rauszer C. The discernibility matrices and functions in information systems [M]. Boston: Kluwer Academic Publishers, 1992: 331-362