

基金项目：漳州职业技术学院科研计划资助项目，项目编号：ZZY1107

一种基于GPU的快速Kirsch边缘检测算法

黄 轩^{1,2}

1.厦门大学智能科学与技术系，福建 厦门 361005；2.漳州职业技术学院经济管理学系，福建 漳州 363000

A Fast Kirsch Edge Detection Algorithm Based on GPU

Huang Xuan^{1,2}

1.Department of Cognitive Science, Xiamen University, Fujian, Xiamen 361005, China ;

2.Department of Economic Management, Zhangzhou Institute of Technology, Fujian, Zhangzhou 363000, China

摘 要

传统的Kirsch边缘检测算法的优化和实现都是针对常用处理器提出的。根据Kirsch算法的可并行计算的特点，本文提出了一种基于图形处理器GPU的快速Kirsch算法。快速算法根据GPU的并行结构和硬件特点，采用了纹理存储技术、多点访问技术和对称计算技术三种加速技术，优化了数据存储结构，提高了数据访问效率，降低了算法复杂度。实验表明，采用基于GPU的算法可将图像处理速度提高到传统Kirsch边缘检测算法的10倍以上。

关键词

GPU；Kirsch算子；边缘检测；GPU通用计算

Abstract

The traditional Kirsch edge detection algorithms for optimization and implementation which were designed for common processor. According to parallel computing capability of Kirsch algorithm, a fast Kirsch edge detection algorithm is presented based on GPU. On the basis of the parallel architecture and hardware characteristic of GPU, the fast algorithm introduces three methods to improve the implementation performance: Texture Storage technology optimizes the data storage structure, multiple point access technology improves the data access efficiency, and symmetry computation technology reduces the computation complex. The experiment expressed that we could get a over ten times speed effect by this method than traditional Kirsch algorithm.

Keywords

GPU；Kirsch arithmetic；edge detection；GPGPU

引言

伴随着GPU技术的飞速发展，现代的GPU已经具备很强的并行计算能力，浮点数计算能力可以达到同级CPU的10倍或10倍以上^[1]。如何充分利用GPU的高速运算及其并行计算能力实现一些复杂运算的快速求解，已经成为当今的热点问题之一。

边缘检测问题是图像图形处理、目标跟踪识别和计算机视觉等领域中最重要的研究内容之一，且已有较长的历史沿革^[2-5]。以往的边缘检测方法主要是对图像按像素的某邻域范围构造边缘检测算子，常用的边缘检测算子有：Roberts算子、Sobel算子、Prewitt算子、Robinson算子、拉普拉斯算子等等。而其中的Kirsch算子通过对原始图像进行8个方向的边缘检测，对各个方向的噪声都具有一定的平滑作用，有着比较广泛的研究和应用价值。但由于它采用8个模板来处理一幅图像的每一个像素，运算量大，计算非常耗时。

目前，利用Kirsch算子进行边缘检测

的快速算法较少^[6-8]，文献[7]提出的FKC算法，有效地解决了经典算法运算量过大的问题，是当前的主流快速算法。但这些优化和实现的技术大多数都是针对处理器本身而提出的，GPU是全新的处理技术，传统方法很难在GPU上高效运行。因此本文结合GPU的硬件结构和特点，通过OpenGL着色语言（GLSL），利用GPU来进行计算，从存储结构、数据访问和计算方法三个方面对传统的Kirsch算法进行了优化。通过仿真实验证明了基于GPU的Kirsch快速算法比传统的算法更为高效。

1 Kirsch算法与GPU通用计算简介

1.1 传统的Kirsch算法

经典的Kirsch算法分别通过对图像的8个方向进行边缘检测，把其中方向响应最大的作为边缘幅度图像的边缘。8个kirsch边缘检测算子如图1所示：

5	5	5	-3	5	5	-3	-3	5	-3	-3	-3
-3	0	-3	-3	0	5	-3	0	5	-3	0	5
-3	-3	-3	-3	-3	-3	-3	-3	5	-3	5	5
-3	-3	3	-3	-3	3	5	-3	-3	5	5	3
-3	0	-3	5	0	-3	5	0	-3	5	0	-3
5	5	5	5	5	-3	5	-3	-3	-3	-3	-3

图1 Kirsch边缘检测算子的滤波模板

假设图像中某一点C及其3x3区域的灰度，假设C₀为C左上方的像素点，其他点C_j(j=1~7)按顺时针方向排列，分别设C_j(j=1, 2,...,8)为经过Kirsch算法的第j种模板计算处理后的C点的边缘幅度，则8个方向模板中的C₁为：

$$C_1 = 5C_0 + 5C_1 + 5C_2 - 3C_3 - 3C_4 - 3C_5 - 3C_6 - 3C_7 = 5(C_0 + C_1 + C_2) - 3(C_3 + C_4 + C_5 + C_6 + C_7)$$

按照同样方法可计算其他7个方向的边缘幅度。则最终C点的边缘幅度C_a：

$$C_a = \max\{C_j\} \quad (j=1, 2, \dots, 8)$$

按照上述Kirsch算法计算边缘幅度值的方法，计算每一个模板所需要的计算量为：进行7次加法(含减法)，2次乘法。计算像素点C的边缘幅度值所需的总计算量为：进行56(7x8)次加法，16(2x8)次乘法，加上求最大幅度的7次比较运算，运算总量太大，从而不符合实时图像图形处理系统的基本要求。

Kirsch边缘检测算法的主要缺陷是运算量较大，采用Kirsch算法处理一幅AxA的图像，所需计算量为：加法56A²次，乘法16A²次，比较运算7A²次。文献[7]中的方法可以把运算速度提高到原来的3倍多；文献[8]中的算法提高得更多，但是这两种改进算法都是针对常用处理器，如果应用并行处理器，运算速度将得到进一步提升。

1.2 GPU通用计算(GPGPU)

最近几年来，GPU(Graphic Processing Unit)飞速发展，GPU本身的特点决定了其非常适合用于高效率低成本的高性能并行计算。而GPGPU(General-purpose computing on graphics processing units)则是使用处理图形任务的专业图形处理器，来从事计算任务。由于现代图形处理器具备强大的并行处理能力，流处理器处理非图形数据成为现实，且通用图形处理器在性能上大大超越了传统的中央处理器应用程序。另一方面，CPU由于受摩尔定律的限制，采用提高CPU制程和主频的办法遇到了工艺上的壁垒，暂时无法突破；而从1993年开始，GPU的性能以每年2.8倍的速度增长，如图2所示。

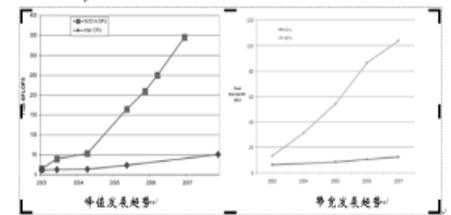


图2 GPU发展趋势

CPU和GPU都是具有运算能力的芯片，CPU更像“通才”——指令运算(执行)为数值运算，GPU则更像“专才”——并行数值计算为核心。CPU+GPU平等运算已经有五年时间，对大量运算并行处理，CPU面临越来越大的困难。更形象地说，CPU和GPU运算好比送比萨，CPU一次又一次地送，一次送一个，像卡车原理，再大但总有最后一个，到时候就凉了。唯一解决的方法是让卡车的容量再大，速度再快。但总会有一个极限，当遭遇瓶颈时，就会发热，耗电。GPU正好相反，它就像几百个小车，每个都很小，很简单，体积耗电小，做复杂的事很有限，但在1秒或者几秒内能够很好地处理信息。

GPU的并行运算性能是极为强悍的，而传统的图形API又简单的只提供了图形操作的功能，没有提供类似于CPU那样通用计算的接口，各种GPU编程语言(Cg、HSL、GLSL)的出现改变了这一状况。这些GPU编程语言编写处理的着色器Shader(有两种Shader：顶点Shader和片元shader)传统上仍是用于图形渲染上，但也可以用于并行计算中。用纹理(二维数组)作为数据模拟输入，着色器应用GPU来处理数据，渲染缓冲区(可看成二维数组)作为数据输出，就可以用模拟图形渲染的方式来并行计算。数以千计的软件开发人员正在使用GPU代替CPU来解决各种专业以及家用应用程序中的问题，为各种应用程序加速。这些应用程序从视频与音频处理和物

理效果模拟到石油天然气勘探、产品设计、医学成像以及科学研究,涵盖了各个领域^[9-11]。

2 基于GPU的Kirsch边缘检测算法

2.1 使用纹理(Texture)类型存储图像数据

Kirsch边缘检测算法在计算时要使用当前像素点的8个邻域信息,而且前后点之间的存储访问相关度很高,倘若使用普通的存储类型,每次访问都必须重新从全局存储器中访问数据,很多像素点信息被重复读取,访问效率很低;另一方面,模板运算必然会导致边界问题,即模板在图像的边缘时要有不同的处理方式,需要设置边界处理条件,这些额外的条件判断语句就会极大降低并行计算的效率,因此,必须采用一种合适的数据存储方式以提高算法效率。纹理类型是GPU定义使用的一种常用存储类型,为图像图形处理提供了专门的优化,在图像图形处理中有广泛的应用。纹理是具有高速的纹理缓存,能够保存最近访问的信息,在缓存中访问数据与访问GPU寄存器的速度差不多;通过对纹理的属性的设置,GPU在访问纹理时能自动处理边界条件,如访问G(-1, -1)可以直接返回G(0, 0)的值。

2.2 多点访问技术

边缘检测算法每次计算一个输出点,要从存储器中读取8个数据。计算结束后,这些数据点信息就被丢弃,等到下一次计算则需要重新从存储器中再次取点,许多数据被重复读取,访问效率低下。采用多点访问技术可以很大程度的提高数据的访问效率:GPU预先读取多个连续的数据放置在寄存器中,后点放置在寄存器中的前点,不必重复访问全局存储器,处理后可同时输出多个处理结果。GPU从全局存储器读取一个数据需要约400~600个时钟周期,直接从寄存器读取数据只需要4个时钟周期,这就极大提高了访问效率。

当需要连续计算4个输出点时,采用多点访问技术后只需访问18个输入点,与原来的方法的36个点相比,足足减少了50%的存储器访问时间;输出图像数据时,生成的4个存储位置连续的8位数可以看作是一个32位数据,在GPU中,传输一个8位数据和传输32位数据所需时间是一样的,因此输出4个点比输出1个点减少75%的存储器访问,垂直方向上读取多个点也是同样的道理。需要特别强调的是,线程访问多个元素能有效提高存储访问效率,但却占用了更多的寄存器,线程内部计算比较复杂时,会降低GPU的并行性,所以,在程序设计时要同时考虑访问效率和并行性程度。

2.3 并行计算管线

每个GPU含有多个渲染管线(4个以上),每个渲染管线在每个GPU时钟周期内可以处理2个像素(每个像素有RGBA4个通道,每个通道含一个float数据)。而kirsch含有8个边缘检测算子,每个边缘检测算子在计算时均不影响源输入图像,故而具有并行计算的特性。故而可基于GPU来设计kirsch边缘检测的并行算法。



图3 着色器的四个通道

我们将可以将带检测边缘的图像数据并发地放到2个管线中(根据kirsch算法特性

不需要更多)去执行计算。图像数据初始的RGBA四个通道中的值均相同,都存放着当前像素点的灰度值。在每个管线中的着色器Shader中存放8个4维向量,每个向量存放着kirsch算法的八个邻域点的权值,其中前4个算子的权值存放在第一个管线的Shader中,后4个算子的权值存放在第二个管线的Shader中。Kirsch算子的权值之所以设计成4维形式是为了和每个像素点的RGBA四个通道进行4维向量运算(在GPU中有加速),可以一次性得到4个检测算子的结果。两条管线并发执行,在每个计算管线我们能够输出4个kirsch检测算子的结果,相当于8个检测算子并发执行,最后通过比较我们能够迅速地得到每个像素点的边缘幅度。

3 实验结果与分析

本文所采用的实验平台为Intel i5-3210M,主频为2.5GHz,系统内存为2G,显卡采用的是AMD Radeon HD 7670M,显卡内存为1GB。操作系统为Windows 7,整个算法实现基于OpenGL和GLSL,实验数据为不同分辨率的8位N×N灰度图像。表1列出了在GPU上和CPU上执行Kirsch算子边缘检测的实验结果。



图4 kirsch算法边缘检测结果

表1 基于GPU的kirsch算法与传统的kirsch算法处理时间的比较

算法	图像大小	处理时间1(s)	处理时间2(s)
传统的Kirsch算法	256×256	0.061	0.062
	512×512	0.243	0.235
	1024×1024	0.712	0.708
基于GPU的kirsch算法	2048×2048	2.852	2.849
	256×256	0.005	0.005
	512×512	0.019	0.018
	1024×1024	0.075	0.074
	2048×2048	0.301	0.298

可以看出,在GPU上执行边缘检测算法,性能有比较明显提升。在处理1024×1024以上分辨率图像时,加速比大于10。一方面由于GPU具有高度的并行架构,能够同时处理多个数据;另一方面,由于边缘检测算法具有良好的可分块性和对称性,实验可以充分利用GPU的并行架构。因此,在处理类似的问题时,采用GPU可以极大的提高效率。

4 结语与展望

通过运用纹理存储技术、多点访问技术和对称计算技术,结合了边缘检测算法良好的可分块性和GPU高度并行架构的特点,在GPU上实现了基于Kirsch算子的边缘检测算法。对分辨率为2048×2048的图像进行计算处理时,速度可达200帧/秒,相比CPU,加速效果明显,实时性有很大程度的提高。算法实现采用通用性技术,对于其他类似的问题和算法也具有一定的现实借鉴意义。

参考文献

- [1]Nvidia corporation. NVIDIA CUDA Programming Guide version 1.1[OL]. http://www.nvidia.com/object/cuda_home.html, 2007-11.
- [2]Davis L S. A Survey of Edge Detection Techniques [J].CGIP, 1975(4): 248-270.
- [3]托马斯·布劳恩.并行图像处理[M].李俊山,译.西安:西安交通大学出版社,2003:29-31.
- [4]Kirsch R. Computer determination of the constituent structure of biological image. Computers and Biomedical Research, 1971,4(3):315-328.
- [5]Natalia Kazakova, Martin Margala, Nelson G Durdle. Sobel Edge Detection Processor For A Real-Time Volume Rendering System[C].Proceedings of the 2004 International Symposium, May 23-26, 2004, 2: 913-916.
- [6]康牧,王宝树.自适应Kirsch边缘检测算法[J].华中科技大学学报,2009,37(4):47-50.
- [7]郑翔,黄艺云.Kirsch边缘检测的快速算法[J].通信学报,1996,17(1):131-134.
- [8]邵平,杨路明.基于模板分解和积分图像的快速Kirsch边缘检测[J].自动化学报,2007,33(8):795-800.
- [9]左颀睿,张启衡,徐勇,赵汝进.基于GPU的快速Sobel边缘检测算法[J].光电子工程,2009,1(36):8-12.
- [10]Richard P Brent.The parallel evaluation of general arithmetic expressions [J].Journal of the Association for Computing Machinery(S0004-5411), 1974, 21: 201-206.
- [11]Zhang Renyan, Zhao Guoling,Su Li. A new edge detection method in image processing[C]. Communications and Information Technology,IEEE International Symposium on. 2005:445-448.

上接第80页

势,进行相互之间的互补和共享,例如:物联网通信信道和智能电网的功率、控制和电能采集技术等有效结合,物联网的信息交互共享优势和智能电网辅助、决策系统、人工智能系统相互结合。

每一个物品在物联网中被寻址,这就需要地质,这样就需要更多的IP地址,IPv4在耗尽的时候需要IPv6来继续支撑。而且,因为物联网的中断不但拥有自身功能,而且还有传感器和网络接入等功能,频繁的采集数据也需要对安全问题给予重视。

3 结语

虽然现在针对智能电网的解释还没有统一的说法,但是使用现金的通信手段和技术,提高电网的智能应用已经不是幻想,所以,在电网智能化的发展过程中,建立一个高速、安全的ICT平台是必经之路。智能电网的物联网领域具有较长的周期性,需要不断提高电网信息化的程度和物联网工作,这样才能使我国智能电网有效、快速的发展,降低成本和电力资源的浪费使用,实现我国社会可持续化发展的目标。

参考文献

- [1]赵婷,高昆仑,郑晓! .智能电网物联网技术架构及信息安全防护体系研究[J].中国电力, 2012, 45(5):87-90.
- [2]任卫东,李琦,康洪渡.基于物联网技术的智能电网信息反馈系统[J].电源技术, 2012, 36(6):829-831.
- [3]王广辉.中国智能用电的实践与未来展望[J].中国电力, 2012, 45(1):1-5.