

使用 OpenCL 技术的影像快速畸变纠正方法在异构平台上的应用分析

韦博文¹ 李 涛² 李广宇¹ 汪致恒¹ 何 沐¹ 师悦龄¹ 刘路遥¹ 张 瑞¹

(西南交通大学地球科学与环境工程学院 成都 611756)¹

(国家测绘地理信息局卫星测绘应用中心 北京 100048)²

摘 要 针对海量遥感数据应用中日益显著的处理效率低下和计算瓶颈问题,基于通用计算机图形处理单元的编程开发使用 OpenCL 并行处理技术对遥感数据处理及其过程进行加速,旨在为遥感影像大数据处理提供一条更为高效的途径。在不同显卡平台上对影像畸变纠正实施并行处理,结果表明,OpenCL 技术在提高影像畸变纠正的速度方面作用显著,可取得 29.1 倍的最高加速效果;与 CUDA 并行处理技术的交叉验证进一步凸显了 OpenCL 技术在异构平台上实施并行处理时所具有的通用性的优势。

关键词 OpenCL, CUDA, 并行处理, 异构平台通用性, 畸变纠正

中图分类号 TP311 文献标识码 A

Applied Analysis of Image Accelerating Distortion Correction of OpenCL Technology on Heterogeneous Platform

WEI Bo-wen¹ LI Tao² LI Guang-yu¹ WANG Zhi-heng¹ HE Mu¹ SHI Yue-ling¹ LIU Lu-yao¹ ZHANG Rui¹

(Faculty of Geosciences and Environmental Engineering, Southwest Jiaotong University, Chengdu 611756, China)¹

(Satellite Surveying and Mapping Application Center, Beijing 100048, China)²

Abstract According to the increasing problem of low processing efficiency and computational bottleneck in massive remote sensing data, for providing a more efficient approach of big data processing of remote sensing, using OpenCL parallel processing technology can accelerate the processing of remote sensing image based on program of general purpose GPU. The analyses of test results on heterogeneous GPU platforms indicate that OpenCL technology can dramatically increase velocity of distortion calibration of image and can achieve 29.1 times best speedup. Comparing to parallel processing of CUDA suggests that using OpenCL technology on heterogeneous platform has a common superiority.

Keywords OpenCL, CUDA, Parallel processing, Heterogeneous platform commonality, Distortion calibration

1 引言

随着遥感技术以及卫星技术的发展,遥感数据在空间分辨率、光谱分辨率和时间分辨率方面都获得显著提升。然而海量数据会导致处理效率的降低, CPU 硬件瓶颈在数据处理方面的制约作用日趋突出。如何适应海量数据处理的需求、结合高性能的计算机硬件和完善的算法来提升遥感数据处理的时效性,对于及时满足空间数据更新、抢险救灾等方面的需求具有重要的现实意义。

计算机硬件水平的提升以及算法的更新促进了将 GPU 用于遥感影像数据的加速处理这一技术的发展。基于 GPGPU (General-purpose Graphics Processing Units) 编程开发, OpenCL 技术在提高数据处理方面优势明显。已有研究表明, OpenCL 技术在遥感影像特征点提取^[1,2]、快速傅里叶变换^[3]、影像 NDVI 计算^[4]和直方图生成^[5]等方面都取得了较好加速效果。丁莹等人^[6]提出了数字图像处理技术对于光学系统进行畸变纠正的新算法,并且使用 CUDA 技术使得处理

速度达到了每秒 30 帧的水平。而詹总谦等人^[7]分析了 GPU 实现畸变差纠正的可能性和任务分配方法,并且提出了基于畸变纠正差修正算法的 CUDA 处理流程。但是在已有的影像加速畸变纠正算法研究中,都只涉及使用 CUDA 技术在单一 GPU 平台上取得很好的加速效果。当前计算机平台并不是朝着单一化方向发展,这使得在异构平台上使用并行技术对遥感影像进行加速处理存在着制约,影响并行处理技术在个人计算机平台上的发展。

基于上述问题和限制,本文使用 OpenCL 技术来探讨异构平台上影像畸变纠正的可行性及其加速性能,并为将来遥感影像加速处理提供一个有效的途径。

2 影像畸变纠正的实现

2.1 实验平台

OpenCL 技术是基于 GPGPU 的开放性编程平台,支持 C/C++、FORTRAN 等高级语言,允许用户调用 GPU 的高性能浮点计算资源来实现繁杂的运算,能实现数据的并行

本文受国家自然科学基金(41474003),四川省应急测绘与防灾减灾工程技术研究中心开放基金(K2015B007)资助。

韦博文(1992—),男,硕士生,主要研究方向为合成孔径雷达差分干涉, E-mail: weibowen100@163.com; 张 瑞(1981—),男,博士,讲师,主要研究方向为合成孔径雷达差分干涉测量和永久散射体理论, E-mail: zhangrui@swjtu.edu.cn(通信作者)。

处理以提高计算机数据处理速度。OpenCL 最大的特点是支持所有可用 GPU(NVIDIA、AMD、Intel 等)显卡,具有很强的平台通用性。为了与 OpenCL 技术进行比较分析,在实验中还使用了 CUDA 技术和 CPU 串行处理实现影像的畸变纠正。

因为实验中使用 OpenCL 和 CUDA 两种不同的并行处理技术,所以需要的硬件和软件有所不同。对于 CUDA 技术,首先需要一台载有 NVIDIA GPU 的计算机,再下载相应的 CUDA SDK 对计算机进行配置。而对于 OpenCL 技术,若计算机载有 NVIDIA GPU,则配置 CUDA SDK 即可直接使用;若计算机载有 AMD GPU 或 Intel GPU,则必须下载相应的 OpenCL SDK 进行配置。与此同时,计算机必须提前安装相应的集成开发环境(如 Microsoft Visual Studio 2010)。

2.2 畸变纠正模型

光学相机的镜头由于受到制作工艺的影响不可避免地存在着系统畸变。按照光线偏移状态的不同,可将系统畸变分为两大类:径向畸变和偏心畸变。径向畸变使得像点沿着镜头向径方向发生位移(见图 1);而偏心畸变使得像点沿着镜头向径方向以及垂直于镜头向径方向偏移(见图 2)。在近景摄影测量中,这两种畸变使得光学相机所获取的影像很难达到测绘应用领域较高精度的要求,因此需要对系统畸变进行纠正。

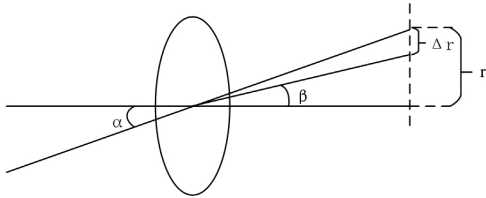


图 1 径向畸变

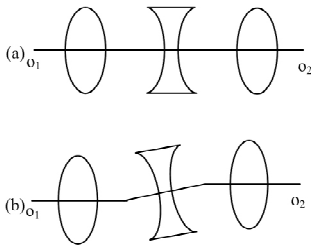


图 2 偏心畸变

数码相机总的畸变纠正模型如式(1)所示:

$$\left. \begin{aligned} \Delta x &= (x-x_0)(k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots) + p_1 [r^2 + 2(x-x_0)^2] + 2p_2 (x-x_0)(y-y_0) + \alpha(x-x_0) + \beta(y-y_0) \\ \Delta y &= (y-y_0)(k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots) + p_2 [r^2 + 2(y-y_0)^2] + 2p_1 (x-x_0)(y-y_0) \\ r &= \sqrt{(x-x_0)^2 + (y-y_0)^2} \end{aligned} \right\} \quad (1)$$

其中, Δx 、 Δy 为像点位移改正数; x 、 y 为像点坐标; x_0 、 y_0 为像主点坐标; k_1 、 k_2 为待定对称径向畸变系数; p_1 、 p_2 为待定切向畸变系数; r 为像点向径; f 为主距。

3 影像畸变纠正

首先读取影像畸变参数文件,其次使用 OpenCV (Open Computer Vision)读取影像,再将影像的第一个像素点代入畸变纠正模型进行迭代处理,计算迭代前后像素点差值,若该差值满足既定的阈值条件则停止迭代,反之继续迭代,如此遍

历整幅影像。在实际应用中,常采用双线性内插^[8]以及间接纠正^[9]的方法进行灰度值重采样,最后生成纠正后的影像。

畸变纠正编程实现的基本流程如图 3 所示。由图 3 可知,OpenCL 需要搭建一系列的并行处理环境,所以 OpenCL 的实现是比较复杂的。畸变纠正的核心部分由内核函数实现,除了内核函数是在 GPU 中执行之外,其余处理流程都在 CPU 中执行。

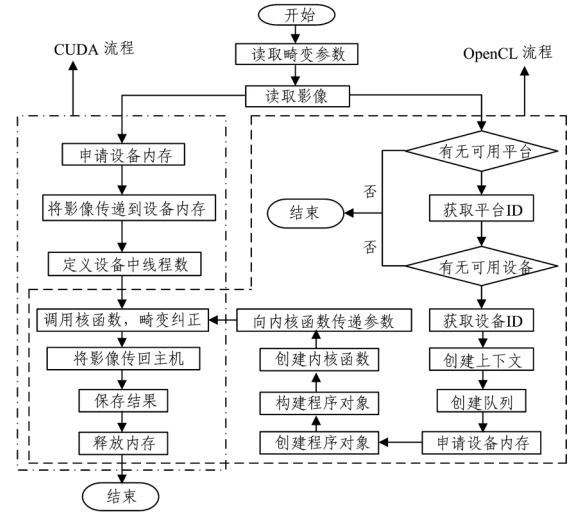


图 3 CUDA 和 OpenCL 实现流程

4 实验与结果分析

本实验采用 CUDA5.5 和 OpenCL SDK 两种并行处理技术分别在 3 种异构 GPU 平台 (NVIDIA GT640M、AMD Radeon 7500M/7600M 和 Intel(R)HD Graphics 4600)计算机上进行实验。实验平台组合如表 1 所列。

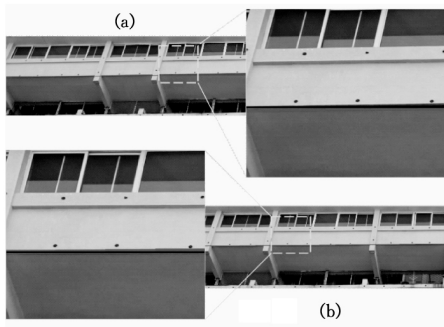
表 1 实验平台组合

实现平台	实现方式
Intel Core i5 CPU	CPU
Intel Core i5 CPU+Intel (R) HD Graphics 4600	OpenCL
Intel Core i5 CPU+AMD Radeon 7500M/7600M	OpenCL
Intel Core i5 CPU+NVIDIA GT640M	OpenCL
	CUDA

使用 Canon EOS 5D Mark II 数码相机分别从两个不同角度各获取一幅带有人工标志的大楼影像 IMG_0111.jpg 和 IMG_0123.jpg 作为实验影像。从 IMG_0111.jpg 左上角第一个像素开始分别截取尺寸为 128×128 pixels、 256×256 pixels、 512×512 pixels、 1024×1024 pixels 和 2048×2048 pixels 的 5 幅影像,再加上尺寸均为 5616×3744 pixels 的原始影像 IMG_0111.jpg 和 IMG_0123.jpg,一共 7 幅实验影像。为方便分析,将此 7 幅影像按尺寸由小到大的顺序分别命名为 Image 1—Image 7。影像畸变参数已提前由相机检校获得。

将相同的代码分别应用在 3 个不同平台上对影像进行畸变纠正,每幅影像重复纠正 100 次,计算所耗费时间的算术平均值,画出图表进行比较分析。

如图 4 所示,将原始影像和纠正后的影像中屋檐部分用白色直线勾画出来进行对比,可见原始影像由于镜头畸变差的原因发生向上的扭曲变形,与黑色直线不重合,而纠正后的影像屋檐部分则能够很好地与黑色直线相吻合,可以证明畸变纠正取得了一个较好的结果。



(a)从原始影像 IMG_0111.jpg 中截取的放大图 (b)从纠正后影像 New_IMG_0111.jpg 中截取的放大图

图4 纠正前后对比

如表2所列,所有的处理方法所消耗的时间均是随着影像尺寸的增大而增长,但增长的趋势不同。CPU处理所耗费的时间随着影像尺寸增大而呈线性增长的趋势,造成这一现象的原因是CPU只能以串行处理方式进行处理,即影像只能在CPU中逐像素进行处理。在Intel Graphics HD 4600+OpenCL平台和AMD Radeon 7500M/7600M+OpenCL平台上处理时,尺寸较小的影像所花费的总时间反而比CPU串

表2 不同大小影像在异构实验平台所耗费的时间(s)

平台	Intel Core i5 (CPU)	Intel Graphics HD 4600 (OpenCL)	AMD Radeon 7500M /7600M (OpenCL)	NVIDIA GT640M (OpenCL)	NVIDIA GT640M (CUDA)
Image 1	0.01903	2.90643	0.850401	0.04448248	0.00157576
Image 2	0.07410	2.92516	0.928269	0.04768330	0.00220202
Image 3	0.29605	3.01102	0.902335	0.05309324	0.00535354
Image 4	1.19536	3.11921	0.996391	0.07061160	0.01435350
Image 5	4.40690	3.26368	0.979363	0.15197600	0.05062663
Image 6	16.9005	4.91187	1.398390	0.58031360	0.23355600
Image 7	16.9085	4.94910	1.361990	0.58042520	0.23461600

如图5所示,相对于CPU平台来说,NVIDIA+CUDA平台加速比最高,可达87倍;其次,NVIDIA+OpenCL平台的最高加速比为29.1倍;ADM+OpenCL最高加速比可达

行处理要高,当影像尺寸继续增大到一定像素数量时才会显示出两平台的加速优势。而对于NVIDIA GT640M+OpenCL平台来说,几乎所有影像的并行处理都得到很大程度的加速。更有甚者,NVIDIA GT640M+CUDA体现出优异的加速性能。CUDA技术能在NVIDIA显卡上取得如此优异的加速效果的原因是安装CUDA SDK时,NVIDIA显卡上所有的并行处理环境都已经被搭建完毕,只需要在使用时进行简单调用即可实现,而不需要编写代码从底层搭建新的并行处理环境。即便是在NVIDIA显卡上使用OpenCL技术,在图4中创建程序对象等操作也不需要从底层开始搭建,所以避免了除并行处理之外的时间浪费。

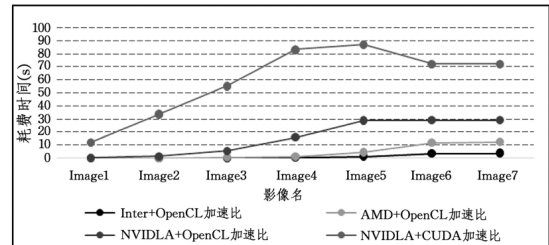
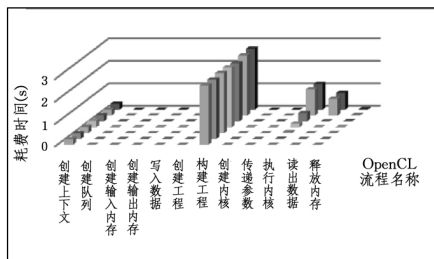
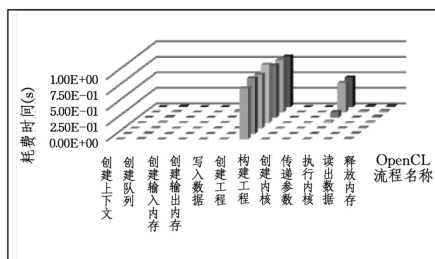


图5 4种不同平台加速比

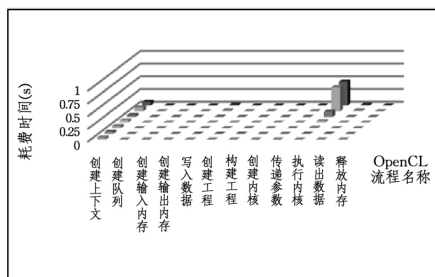
12.4倍;而Intel+OpenCL平台实现的加速最不明显,只有3.4倍。为分析其原因,将所有并行处理平台中各阶段运行时所耗费的时间进行统计分析。



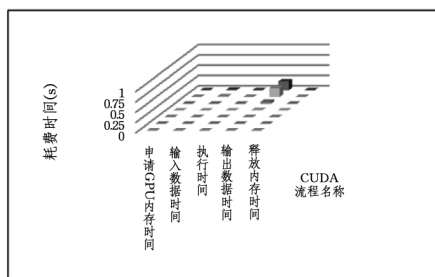
(a) Intel+OpenCL 各部分耗费时间



(b) AMD+OpenCL 各部分耗费时间



(c) NVIDIA+OpenCL 各部分耗费时间



(d) NVIDIA+CUDA 各部分耗费时间

■ Image 1 ■ Image 2 ■ Image 3 ■ Image 4 ■ Image 5 ■ Image 6 ■ Image 7

图6 不同平台耗费的时间分布

根据符号压力函数可以自动调节演化速度,避免边缘泄漏,实现对目标区域的精确分割。实验结果表明本文组合算法对医学图像中复杂目标的分割精确更高,效果更满意。

参 考 文 献

- [1] 江贵平,秦文健,周寿军,等.医学图像分割及其发展现状[J].计算机学报,2015,38(6):1222-1241
- [2] 刁智华,赵春江,郭新宇,等.分水岭算法的改进方法研究[J].计算机工程,2010,36(17):4-6
- [3] Osher S, Sethian J A. Fronts propagating with curvature dependent speed; Algorithms based on the Hamilton-Jacobi formulation [J]. Journal of Computational Physics, 1998, 79: 12-49
- [4] Li C M, Kao C Y, Gore J C, et al. Minimization of region-scalable fitting energy for image segmentation [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2008, 17 (10): 1940-1949
- [5] Li C, Xu C, Gui C, et al. Level set evolution without re-initialization: A new variational formulation [C] // IEEE Conference on Computer Vision and Patten Recognition (CVPR). 2005: 430-436

(上接第 169 页)

由图 6 示出的各平台所耗费的时间分布可知,不同的显卡平台进行并行处理时,各个阶段的运行所花费的时间不尽相同。由于 NVIDIA 显卡的 CUDA SDK 提前搭建好了相应的并行处理环境,因此 NVIDIA+OpenCL 并行处理的大部分时间被消耗在内核执行上,而不需要消耗在环境搭建上。与之相比,CUDA 并行处理内核执行所耗费时间几乎为零,而大部分时间被消耗在数据传递上,所以出现了图 5 中后两幅大尺寸影像加速比没有前两幅的高的现象。Intel+OpenCL 和 AMD+OpenCL 并行处理在工程构建上消耗了大部分时间,所以导致了表 2 中较小尺寸影像在 OpenCL 并行处理时所耗费的时间反而比在 CPU 串行处理时所耗费的时间多。正是由于每次在并行处理前进行新环境的构建,OpenCL 技术才具有 CUDA 技术所没有的异构平台并行处理的通用性。

结束语 随着卫星、无人机等技术的发展,遥感数据量亦会随之增加,提高遥感数据处理的效率以及平台兼容性有利于应对大数据、快节奏以及自动化时代的发展需求,能更有效地提高对于各种突发事件(如地震、泥石流等)的反应能力,保护人们生命财产安全。本文使用 OpenCL 技术在现今主流的计算机显卡平台上实现影像畸变纠正的并行处理,使得影像处理速度比 CPU 的串行处理提高了 3.4~29.1 倍,较大地提高了遥感数据处理的效率。但是相比 CUDA 技术来说,OpenCL 技术的处理效率仍需要进一步提高。在异构平台上,OpenCL 展示出了优良的异构平台通用性,不仅能应用于串行处理的 CPU 平台,也能应用于 Intel、AMD 和 NVIDIA 等 GPU 平台。这不仅降低了针对不同计算机平台重新编写并行处理程序的成本,还解决了因不同计算机平台所带来的并行处理程序兼容性问题,大大提高了实际工作中的效率。

- [6] Vincent L, Soilh P. Watersheds in digital space: An efficient algorithm based on immersion simulations [J]. IEEE Transactions on Pattern Analysis and Machine Interpretation, 1991, 13 (6): 583-598
- [7] 阮秋琦,阮宇智,等.数字图像处理[M].北京:电子工业出版社,2011:497-502
- [8] 赵珊,王水.结合进化规划的图像分水岭分割技术[J].计算机科学,2011,38(5):265-267
- [9] 方江雄.变分和偏微分方法在图像分割中的应用[M].北京:中国石化出版社,2015:20-23
- [10] Xu C Y, Yezzi A, Prince J L. On the relationship between parametric and geometric active contour [C] // Processing of 34th Asilomar Conference on Signals Systems and Computer. Pacific Grove: IEEE Press, 2000: 483-489
- [11] Salah M B, Mitiche A, Ayed I B. Multiregion image segmentation by parametric kernel graph cuts [J]. IEEE Transactions on Image Processing, 2011, 20(2): 545-557
- [12] Han X Z, Jian Z. A nonlinear image enhancement algorithm based on partial differential equations [C] // IEEE 10th International Conference Signal Processing. 2010: 1114-1116

同时,这也为个人计算机能更好地运用于遥感影像快速处理提供了一个有效的途径,极大地促进了并行处理技术在个人计算机上的发展。

本研究侧重于解决在遥感影像大数据条件下如何提高个人计算机的处理效率问题以及异构平台的通用性问题,但是关于影像数据存储方面有待深入研究,有效地管理影像数据的存储能更进一步提高处理效率,这是今后进一步研究和解决的重点。

参 考 文 献

- [1] 肖汉,马歌,周清雷.面向 OpenCL 架构的 Harris 角点检测算法 [J]. 计算机科学, 2014, 41(7): 306-309, 321
- [2] 王艳梅,史晓华,于湛麟. SURF 算法在通用 GPU 和 OpenCL 的优化与实现 [J]. 电子测试, 2013(23): 38-42
- [3] 李焱,张云泉,王可等.异构平台上基于 OpenCL 的 FFT 实现与优化 [J]. 计算机科学, 2011, 38(8): 284-286, 296
- [4] 熊英,罗琼.基于 OpenCL 的 NDVI 算法 [J]. 计算机光盘软件与应用, 2013, 16(18): 99-100
- [5] 安小景,张云泉,贾海鹏.基于 OpenCL 的直方图生成算法优化方法研究 [J]. 计算机科学, 2015, 42(11): 32-36
- [6] 丁莹,李文辉,范静涛,等.基于 GPU 的大视场景物畸变实时校正算法 [J]. 吉林大学学报(工学版), 2010, 40(1): 189-194
- [7] 詹总谦,卢亮.基于 GPU 并行处理技术的影像畸变差修正 [J]. 测绘信息与工程, 2011, 36(2): 1-3
- [8] Parker J A, Kenyon R V, Troxel D E. Comparison of interpolating methods for image resampling [J]. IEEE Transactions on Medical Imaging, 1983, 2(1): 31-39
- [9] 张剑清,潘励,王树根.摄影测量学(第二版) [M]. 武汉:武汉大学出版社, 2003: 213-215