

应用 MPICH 并行化图像处理算法

李海荣, 方中纯

(内蒙古科技大学 工程训练中心, 内蒙古 包头 014010)

摘 要:近年来,图像处理技术取得了巨大进步,但随之也带来了不断增长的计算量。为了提高图像处理速度,作为主流解决方案之一的基于 MPI 的并行处理技术日益受到重视。文中首先介绍了 MPI 的基本知识,然后以图像处理中的一项重要技术-图像锐化(采用梯度算法)为例,说明了采用 MPI 对图像进行并行处理的具体过程,并就其中的两个核心步骤:建立并行算法模型及程序代码的编写进行了详细的阐述。最后通过对实验数据的分析说明了并行计算在图像处理领域所起到的重要作用。

关键词: MPI; 并行计算; 图像锐化

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2013)07-0101-03

doi:10.3969/j.issn.1673-629X.2013.07.025

Parallelizing Algorithms of Image Processing Using MPICH

LI Hai-rong, FANG Zhong-chun

(Engineering Training Center, Inner Mongolia University of Science and Technology, Baotou 014010, China)

Abstract: Recently with the development of image processing technique, computational cost increases greatly. To improve the speed of image processing, parallel-computing cluster based on MPI as mainstream solution is paid more attention. First, the basic knowledge of MPI is introduced. And then the parallelization of image gradient sharpening is taken as example. In this example, the general process of parallelization for image processing is described in detail: modeling and program implementation. At last the experimental data show the important function of parallel-computing in image processing area.

Key words: MPI; parallel-computing; image sharpening

0 引言

随着科学技术的发展,越来越多的场合对计算速度提出了更高的要求。如人类基因、天气预报、石油勘探、大规模的地形匹配等等。近年来,微处理器的性能不断提高,这使得人们可以搭建并行集群计算系统来提高计算速度,并且满足大规模数据处理的需求。并行计算机系统的出现就需要对程序进行并行设计。可以说,不在并行编程方面开展大量工作就无法充分发挥多核处理器的优势,许多应用程序也就不能在高性能计算机上获得加速。目前,高性能计算机系统中较为常用的并行编程环境主要有两种:PVM(Parallel Virtual Machine,并行虚拟机)和MPI(Message Passing Interface)。

PVM是一个软件包,支持多种并行计算模型,用户使用PVM提供的函数库可进行并行程序或分布式

程序的设计工作。MPI是一个并行计算消息传递接口标准。现已成为产业界广泛支持的并行计算标准。

图像并行处理技术是图像处理中的一个重要方面,是提高图像处理速度最为有效的技术,它在处理速度上所获得的加速比是令人振奋的,其实际应用系统也将产生巨大的经济效益和社会效益^[1]。

1 MPI 简介

它是由MPI论坛推出,MPI论坛是由欧美主要的厂商、软件开发商、大学、研究中心组成的一个开放团体,并先后于1994年和1997年推出了MPI-1和MPI-2。目前,MPI已经成为国际上应用最广泛和稳定的并行程序设计技术。当然,MPI只是一个并行编程语言标准,而不是一种具体的编程语言。要编写基于MPI的并行程序,还必须借助MPI的某种具体实现。目前

收稿日期:2012-09-20

修回日期:2012-12-23

网络出版时间:2013-03-05

基金项目:内蒙古自治区高等学校科学研究项目(NJZY12113, NJZZ11142)

作者简介:李海荣(1976-),女,内蒙古包头人,副教授,博士,研究方向为云计算、数据库。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20130305.0817.041.html>

主要的 MPI 实现有: MPICH、LAMMPI、HP-MPI、MS-MPI 等等。其中, MPICH 是最重要的一种 MPI 实现, 它是由 Argonne 国家实验室和密西西比州立大学联合开发的一个免费软件, 支持多种操作系统, 如: LINUX、Windows NT/2000/XP 等, 具有更好的可移植性。当前最新版本是 2012 年 12 月 19 日发布的 MPICH2-3.0.1, 可以从相关网站免费获得 <http://www.mpich.org/downloads/>, 按照文档进行安装和配置^[2]。

MPI 支持 Fortran 和 C/C++ 语言, 即在标准串行程序设计语言 (Fortran 和 C/C++) 的基础上, 加入实现进程间通信的 MPI 消息传递的库函数, 就构成了编写 MPI 并行程序所依赖的并行编程环境。

一个 MPI 程序的基本结构如图 1 所示:

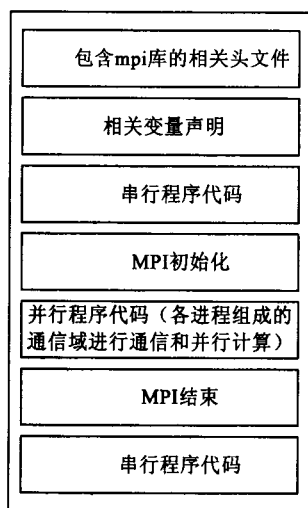


图 1 MPI 程序基本结构

2 并行化图像处理算法的一般过程

一般来说, 图像处理的并行求解过程分为以下 3 步^[3]:

- (1) 分析图像处理问题, 建立算法的串行模型。
- (2) 抽取出算法的串行模型中需要并行处理的部分 (一般均为计算量最大的部分), 确定并行实现方法, 从而建立算法的并行模型。
- (3) 选用程序设计语言实现该并行算法, 然后在已搭建好的集群系统上运行调试。

2.1 建立算法模型

数字图像处理并行算法分为几何并行、领域并行、像素-比特并行和运算并行。图像一般在正方形网格上采样而以二维数组的方式存储, 因此对像素值的逐点处理可以并行处理。所以对图像的处理过程具有天然的几何并行性质。

以在实际应用中占重要地位的图像锐化算法为例来说明进行算法模型建立的过程。

图像锐化的主要目的有两个^[4]: 一是增强图像边

缘, 使图像变得更加清晰, 颜色变得鲜明突出, 图像的质量有所改善, 更适合人眼观察和识别; 二是希望经过锐化处理后, 目标物体的边缘鲜明, 一方面可以提高图像细节的清晰度和质感, 另外也更便于进行提取目标的边缘、对图像进行分割、目标区域识别、区域形状提取等操作, 从而为进一步的图像理解与分析奠定基础。图像锐化算法一般有两种: 微分法和高通滤波法。这里选用的是微分法中常用的锐化算法之一: 罗伯特梯度法 (Robert Gradient), 这是交叉地进行差分计算的一种方法。该算法在本质上是一种局部的邻域算法, 即在处理某一像素时, 利用与该像素相邻的一组像素, 经过某种变换得到处理后图像中某一点的像素值。表示为:

$$G[f(i,j)] = \sqrt{[f(i,j) - f(i+1,j+1)]^2 + [f(i+1,j) - f(i,j+1)]^2} \quad (1)$$

为提高运算速度和便于编程, 在计算精度允许的范围内, 可采用绝对差算法近似为:

$$G[f(i,j)] = |f(i,j) - f(i+1,j+1)| + |f(i+1,j) - f(i,j+1)| \quad (2)$$

即某一点的新的灰度级别值是由它和它周围三个点来决定的。由上可见, 计算灰度值是算法中计算量最大的部分, 这样就可以把计算灰度相关值的任务分解到多个计算节点 (或进程, 以下同) 上并行执行, 从而达到缩短运行时间, 提高运行效率的目的^[5,6]。针对该锐化算法的特点, 文中采用了并行程序设计中的主从 (Master-Slave) 模式, 算法按行来进行任务的划分, 具体过程是:

首先把参考图按行进行划分 (设有 N 行), 然后根据参与计算的节点数 M , 计算出每个节点需要处理 N/M 行。接着将这些数据发给各个子节点, 各个子节点接收到主节点发来的数据, 根据锐化公式计算出新的像素的灰度值, 最后将处理后的数据发给主节点, 主节点综合所有的结果并输出, 即进行归约, 这样整个计算过程就完成了。在处理过程中, 节点之间的通信是通过 MPICH 提供的消息传递函数进行的。

从中可以看出:

- (1) 因为各个节点求灰度值与分块数据无关, 所以可以采用主从模式。
- (2) 主节点 Master 的工作是分块、发送数据、接收计算结果、归约。
- (3) 子节点 Slave 负责各个子图像的计算。

2.2 编程实现

以下程序用 C 语言在一台双核处理器 3.0GHz 的计算机上进行实现、运行^[7-9]。

- (1) 并行程序初始化。

MPI_Init(&argc, &argv);

初始化 MPI 运行环境,获取用户指定执行参数。
是 MPI 程序的第一个函数调用。

MPI_Comm_rank(MPI_COMM_WORLD, &myid);

获得调用进程在指定通信域中的编号,放入变量 myid,一般 0 为主节点。该编号是为了区分各个不同的进程,从而实现进程间的并行和合作。

MPI_Comm_size(MPI_COMM_WORLD, &numprocs);

获得指定的通信域中所包括的进程个数,放入变量 numprocs 中。

(2)处理过程的部分关键代码。

if(myid == 0)

//如果是主进程,需要做以下工作

{

for(i = 1; i < numprocs; i++)

{

MPI_Send(rSize, 3, MPI_INT, i, 99, MPI_COMM_WORLD);

//给编号为 i 的进程发送图像的范围, rSize 是一维数组,分别存储分块数据的行、列信息

MPI_Send(&arData[rSize[0] * cols], rSize[1] * rSize[2], MPI_CHAR, i, 99, MPI_COMM_WORLD);

//给编号为 i 的进程发送图像的数据

{

// 以下为接收处理后的图像块数据

for(i = 1; i < numprocs; i++)

{

MPI_Recv(rSize, 3, MPI_INT, i, 99, MPI_COMM_WORLD, &status);

//从编号为 i 的进程接收图像的范围

MPI_Recv(&arData[rSize[0] * cols], (rSize[1]-1) * rSize[2], MPI_CHAR, i, 99, MPI_COMM_WORLD, &status);

//从编号为 i 的进程接收图像的数据

{

{

else

//子进程的工作如下

{

MPI_Recv(rSize, 3, MPI_INT, 0, 99, MPI_COMM_WORLD, &status);

// 从编号为 0 的主进程接收图像的范围,放入地址为 rSize 的缓冲区

MPI_Recv(imageData, rSize[1] * rSize[2], MPI_CHAR, 0, 99, MPI_COMM_WORLD, &status);

// 从编号为 0 的主进程接收图像的数据,放入地址为 imageData 的缓冲区

for(i = 0; i < rSize[1] - 1; i++)

{for(j = 0; j < rSize[2] - 1; j++)

{imageData[i * rSize[2] + j] = fabs(imageData[i * rSize

[2] + j] - imageData[(i + 1) * rSize[2] + (j + 1)]) + fabs
(imageData[(i + 1) * rSize[2] + j] - imageData[i * rSize[2]
+ (j + 1)]);

}}

// 对接收的数据做计算,参见公式(2)

MPI_Send(rSize, 3, MPI_INT, 0, 99, MPI_COMM_WORLD);

//向主进程回传处理后图像的范围

MPI_Send(imageData, (rSize[1] - 1) * rSize[2], MPI_CHAR, 0, 99, MPI_COMM_WORLD);

// 向主进程回传处理后图像的数据

}

(3)并行程序结束。

MPI_Finalize();

结束 MPI 程序的运行,是 MPI 程序的最后一个调用,也是 MPI 程序的最后一条可执行语句,否则程序的运行结果是不可预知的。

2.3 程序运行结果分析

在一台双核处理器 3.0GHz 的计算机上对一幅大小为 4288 * 2848 的图像进行锐化处理,表 1 列出使用不同数目的进程的运行时间(当节点数目为 1 时,相当于串行运算)。

表 1 使用不同数目进程运行时间对比表

进程数目(p)	运行时间(us)	加速比
1	1250	/
2	875	1.43
3	800	1.56
4	715	1.75
5	750	1.67
6	775	1.61

从结果可以看出,运行时间随着进程的增加而减少,当进程数目为 4 时,运行时间最短且加速比最大。但是如果进程数继续增多,则用于消息传递的时间开销增加,因此导致运行时间变化不大且有增加的趋势。可见,节点之间的通信开销成为影响并行计算加速比和并行效率的主要因素。因此为了提高并行处理的效率,需要根据实际情况来进行并行算法设计和优化。总体上合适的并行处理相对串行(单节点)处理可以明显减少计算时间,提高计算速度。

3 结束语

文中讨论并详细实现了一种基于 MPI 的图像锐化处理算法,算法采用主从模式。结果表明恰当运用并行计算可以提高图像处理的运算速度和精度。以后的工作可以围绕图像处理常用算法的并行库的建立^[10]展开。希望文中可以为同行提供借鉴和参考。

(下转第 107 页)

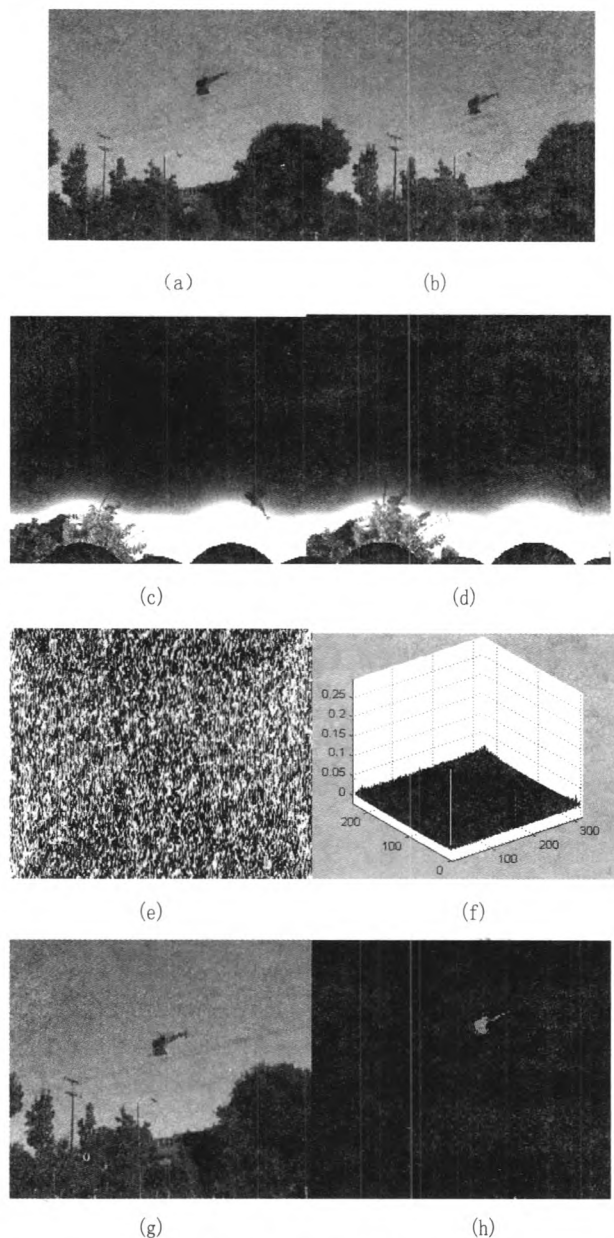


图 3 实验仿真结果

参考文献:

- [1] 黄鑫娟,周洁敏,刘伯扬. 自适应混合高斯背景模型的运动目标检测方法[J]. 计算机应用,2010,30(1):71-74.
- [2] Huang Zhongwen, Qi Feihu, Cen Feng. Background Subtraction and Frame Difference Based Moving Object Detection for Real-time Surveillance[J]. Journal of Donghua University, 2003,20(1):15-19.
- [3] 严晓明. 一种基于改进帧差法的运动目标检测[J]. 莆田学院学报,2011,18(5):69-72.
- [4] 郑志彬,叶中付. 基于相位相关的图像配准算法[J]. 数据采集与处理,2006,21(4):444-449.
- [5] 王福田,吴福虎,罗 斌,等. 基于相位相关的电力红外热像拼接[J]. 计算机技术与发展,2012,22(5):150-152.
- [6] Kuglin C, Hines D. The phase correlation image alignment method[C]//Proc. of IEEE Conference on Cybernetics and Society. New York: [s. n.], 1975.
- [7] 郭晓新,许志闻,卢奕南. 基于 Fourier-Mellin 不变量的图像配准方法[J]. 仪器仪表学报,2004,25(4):421-424.
- [8] 王 珏,肖 斌,马建峰. 基于 Radon 和解析 Fourier-Mellin 变换的尺度与旋转不变目标识别算法[J]. 中国图象图形学报,2008,13(11):2157-2162.
- [9] Chen Q S, Defrise M, Deconinck F. Symmetric Phase-only Matched Filtering of Fourier-Mellin Transforms for Image Registration and Recognition[J]. IEEE Transaction on Pattern Analysis and Machine Intelligence, 1994, 16(12):1156-1168.
- [10] Wang D, Wang L. Fast and robust algorithm for global motion estimation[C]//Proc. of SPIE Conf. on Visual Communications and Image Processing. San Jose, CA, USA: [s. n.], 1997:1144-1151.
- [11] 王思珏,赵 建,韩希珍. 基于放射变换的快速全局运动估计算法[J]. 液晶与显示,2012,27(2):263-266.
- [12] 强赞霞,彭嘉雄,王洪群. 基于傅里叶变换的遥感图像配准算法[J]. 红外与激光工程,2004,33(4):307-311.

(上接第 103 页)

参考文献:

- [1] 危疆树. 图像处理算法分析及其并行模式研究[D]. 成都:电子科技大学,2006.
- [2] MPICH Forum. MPICH[EB/OL]. 2012-12-20. <http://www.mpich.org/>.
- [3] 吕 捷,张天序,张必银. MPI 并行计算在图像处理方面的应用[J]. 红外与激光工程,2004,33(5):496-499.
- [4] Gonzalez R C. 数字图像处理[M]. 阮秋琦,译. 第 3 版. 北京:电子工业出版社,2011.
- [5] 刘振安,章守信,刘胜璞. 并行图像处理算法的设计与实现[J]. 测控技术,2003,22(5):5-6.
- [6] 余 霞,葛 红,何 俊,等. 基于 MPI 的并行医学图像处理[J]. 计算机工程与科学,2009,31(3):32-34.
- [7] 王勇超,张 璟,王新卫,等. 基于 MPICH2 的高性能计算集群系统研究[J]. 计算机技术与发展,2008,18(9):101-104.
- [8] Brauml T. Parallel Image Processing 并行图像处理[M]. 李俊山,李新社,焦 康,译. 西安:西安交通大学出版社,2003.
- [9] 孙 敏. 图像处理并行算法研究与实现[D]. 绵阳:西南科技大学,2011.
- [10] 蒋 英,雷永梅. 基于 MPI 的几种算法的并行编程通用算法[J]. 计算机工程与应用,2003,39(3):139-141.

应用MPICH并行化图像处理算法


作者:

李海荣, 方中纯, [LI Hai-rong](#), [FANG Zhong-chun](#)

作者单位:

[内蒙古科技大学工程训练中心, 内蒙古包头, 014010](#)

刊名:

[计算机技术与发展](#) 

英文刊名:

[Computer Technology and Development](#)

年, 卷(期):

2013, 23(7)

本文链接: http://d.g.wanfangdata.com.cn/Periodical_wjfz201307025.aspx