

OpenMP 多核计算技术在 SAR 图像处理中的应用

程 栋 王卫红

(中国电子科学研究院 北京 100041)

摘 要 SAR 图像数据量大,常规识别算法复杂、处理耗时,难以满足实时性要求。针对这一问题,提出一种基于 OpenMP 多核计算的 SAR 图像目标分类算法。在分析基于模板匹配的 SAR 图像目标分类算法的基础上,给出基于 OpenMP 多核计算技术的图像处理并行处理框架,实现 SAR 图像目标分类算法的并行计算。最后,采用所提方法对 3 类目标进行分类识别实验,SAR 图像分类识别的处理速度提高了 8 倍,表明了该方法是有效的。

关键词 SAR 图像,目标识别,并行计算,OpenMP

中图分类号 TP311.11 文献标识码 A

Application of OpenMP in SAR Image Processing

CHENG Dong WANG Wei-hong

(China Academy of Electronics and Information Technology, Beijing 100041, China)

Abstract The amount of a SAR image is very large usually, meanwhile, the ordinary recognition algorithm is complex and time-consuming. Aimed to solve this jam, a kind of target classification method of SAR image based on OpenMP is proposed. Firstly, the model-template based recognition algorithm is analyzed. Secondly, the OpenMP is applied to build the parallel calculation frame of SAR image target classification method. Finally, the tests are completed by classifying 3 kinds of objects. The test results show that the processing speed is improved by 8 times, and the proposed method is available.

Keywords SAR image, Target recognition, Parallel calculation, OpenMP

1 引言

CPU(Central Processing Unit)即中央处理器,是现代计算机的核心部件,对整个机器运行以及科学计算的效率起到至关重要的作用。从 1971 年 Intel 公司推出世界上第一块用于计算机的微处理器 4004 算起,CPU 已经经历了 40 余年的发展。最初的 4004 微处理器只包含了 2300 个晶体管,功能相当有限,速度也比较慢;最新的 Intel Pentium 8400EE 处理器已经包含了 2.3 亿个晶体管,集成度提高了十万倍。但是,由于制造工艺、散热和功耗等原因,单个 CPU 主频的提升已经遇到了瓶颈,于是多核架构应运而生。目前,多核处理器已经非常成熟,多个核之间的高效协作能力极大地满足了人们对计算能力的追求。

合成孔径雷达(Synthetic Aperture Radar, SAR)图像处理是数字图像处理中的一种特例,在军事侦察、地形和生态环境测绘监测和海洋遥感探测等领域发挥着巨大的作用。随着 SAR 技术的不断发展和完善,SAR 图像的分辨率越来越精细、尺寸越来越大,对 10000 * 10000 像素的 SAR 图像进行处理已成为当前的基本需求。同时,为了满足 SAR 图像处理的实时性需求,利用多核进行并行计算成了当务之急。OpenMP 是由世界上主要的计算机硬件和软件厂商提出的标准,它是一种面向共享内存的多核多线程并行编程库,提供了一套扩展标准 C/C++ 和 FORTRAN 的指令。开发者通过

OpenMP 可以不用关心线程创建、同步、分配和回收的细节,而专注于业务本身^[1-2]。

本文在 Windows 平台下,基于 VS2013+QT,在 64 位编译环境下应用 OpenMP 技术进行 SAR 图像处理并行计算的设计和实现,有效地利用了现有的多核资源,较以往的单个线程运算大大节省了处理的时间。

2 OpenMP 介绍

OpenMP 不是一种独立的语言,它是在串行编程语言上的一种扩展,是为在多核处理器上编写并行程序而设计的一个 API,可以在标准 C/C++ 和 FORTRAN 中应用,并以注释的形式出现,通过打开/关闭并行计算选项来决定其语句是否执行。OpenMP 支持多种平台,包括绝大多数 Unix, Linux 以及 Windows NT 系统。

2.1 OpenMP 并行结构

图 1 示出了 OpenMP 的体系架构^[3-4]。

OpenMP 由指导语句、环境变量和运行库函数组成。指导语句由 directive(指令)和 clause list(子句列表)组成,用来指示编译器如何将串行程序转化成并行程序;环境变量是 OpenMP 中定义的一些用来对并行计算进行控制的命令;运行库函数是 OpenMP 编译一个重要组成部分,提供了获取处理 OpenMP 线程信息、变量属性等功能,可以方便和灵活地将程序进行并行化处理。

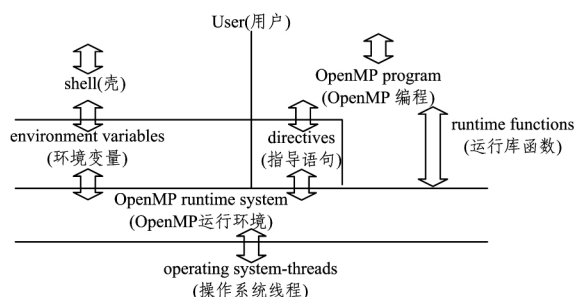


图1 OpenMp 体系结构

OpenMp 有两种最常用的并行模式:fork-join 模式和单程序多数据(Single Program Multi-Data,SPMD)模式。

fork-join 模式是最常用的一种方式,它的优点是以增量化的方式开发串行计算程序,较为简单方便。图2是并行计算的过程图,程序开始时串行执行,只有一个主线程存在,当碰到并行计算任务时,主线程进行fork操作,产生一组线程,每个线程执行一部分操作,共同完成任务;任务完成后,各线程进行join操作,返回主线程。

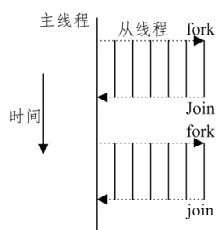


图2 fork-join 并行模式

SPMD 模式是指程序由一个或者多个具有 SPMD 特征的并行区域组成,在每个单独的并行区域中,有多个任务结构,依赖标准的 OpenMP 任务分配指导语句并借助 OpenMP 的库函数 `omp_get_num_threads()` 和 `omp_get_thread_num()` 在整个线程组中对这些任务进行分配。

相对于 fork-join 而言,SPMD 模式的程序一次创建线程,多次分配任务,减小了线程管理引发的开销。但是,编写 SPMD 类型的 OpenMP 程序比 fork-join 类型的程序更困难,SPMD 类型的程序比 fork-join 类型的程序的并行区域结构更大,需要对数据相关性、线程同步及变量共享或私有化投入更多的精力。具体在应用时,开发者要根据实际情况选择适合的形式完成并行计算^[5]。

2.2 OpenMP 主要指令集

OpenMP 不但支持 C/C++, 还支持 Fortran 语言。下面主要介绍 C/C++ 中 OpenMP 的指令集。OpenMP 指令以 `#pragma` 开始,其后面是 `omp`、指令名称和可选的子句,最后用新行结束,如下所示:

```
#pragma omp 指令[子句[子句]...]
```

常用的指令有:

`parallel`: 出现在一段代码之前,表示这段代码将被多个线程并行执行。`parallel` 指导命令包含的代码段,线程组中的所有线程都要执行。

`for`: 用于 `for` 循环之前,将循环分配到多个线程中并行执行。

`sections`: 用在可能被并行执行的代码段之前,根据 `section` 数量和线程数量的不同,可能有时一个线程执行多个 `section`。

常用的子句有:

`Private`: 表示它列出的变量对于每个线程都是局部的。

`Firstprivate`: 表示它列出的变量在每个线程都有私有的副本,并且变量要被继承主线程中的初值。

`Shared`: 表示它列出的变量对于线程组中的所有线程是共享的。

`Reduction`: 当使用指令的操作对其列表中出现变量进行操作时,每个线程都保留一份私有的拷贝在结构尾部,根据指定的操作对线程中相应变量进行规约,并更新该变量的全局数值^[6-7]。

3 基于 OpenMP 技术的并行识别算法设计

3.1 分类识别算法的流程

SAR 图像在军事侦察、地形和生态环境测绘监测以及海洋遥感探测等领域发挥着巨大的作用。通过对 SAR 图像的处理,可以从中提取出重要的目标信息,并对目标信息进行分析,从而判断目标类型、属性甚至目标型号。

在基于模板的 SAR 图像识别系统中,对 SAR 图像 ROI 切片中的目标进行分类判别时,模板比对是一种简单、常用的手段。后台模板库中存储了不同俯仰角、方位角下不同目标的特征矢量或者图像,通过将 ROI 切片与这些预存的模板进行比对,实现对目标的分类和判别。为了能够达到最佳的匹配效果,数据库中需要存储尽可能完备的模板,因此对计算性能提出了较大的需求。

目标分类识别的流程如图3所示。

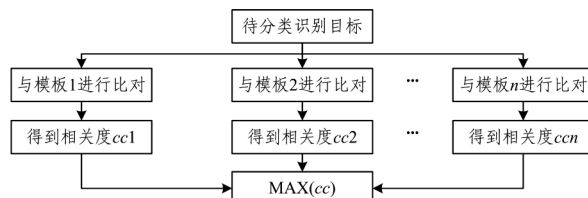


图3 目标分类识别的流程

在目标模板库中预存了 n 种不同种类目标 SAR 图像,待分类识别的 ROI 切片与这 n 个目标的模板(极限情况下是穷尽模板库,通常可以根据其他信息选定一定方位角和俯仰角范围内的模板)进行对比,得到不同的相关度 cc ,相关度最大的即为待识别目标最可能的目标类别。

而待分类识别目标与模板的对比过程如下所描述。已知待识别图像的区域大小为 $P \times Q$ 个像素,模板图像的区域大小为 $M \times N$ 个像素。为了能够更加准确、完善地实现互相关比,首先对模板图像进行扩展,模板图像的长和宽各扩展一定的像素(本次实验扩展 5 个像素),扩展后的模板为 $(M+5) \times (N+5)$ 个像素,如图4所示。

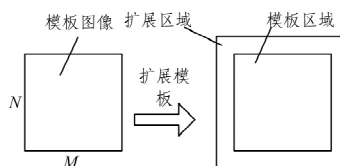


图4 扩展模板区域

按照从左到右、从上到下的顺序抽取待识别的图像和扩展后的模板图像的像素,分别形成像素值构成的数列 X 和 Y 。

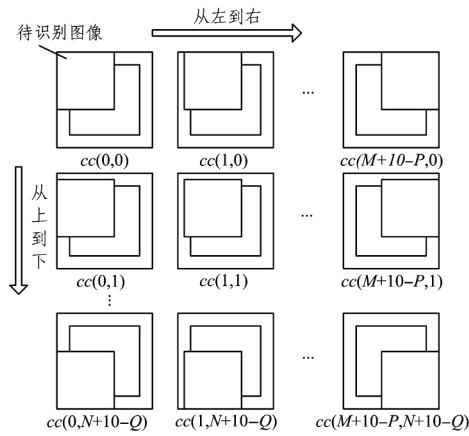


图 5 模板比对

如图 5 所示,将待识别图像的左上角与扩展后的模板图
像左上角对齐后进行像素比对,比对双方分别为待识别图

和与待识别图像区域重叠的扩展模板图像。计算两者的互
关值 $cc(0,0)$,公式如下:

$$cc(0,0) = \frac{\sum_{i=0}^n (X_i - E(X))(Y_i - E(Y))}{\sqrt{\sum_{i=0}^n (X_i - E(X))^2 \sum_{i=0}^n (Y_i - E(Y))^2}}$$

其中, n 为待识别图像的像素个数, $n = P \times Q$; X_i 为待识别图
像的像素数列中第 i 个点的像素值; Y_i 为与待识别图像区域
重叠的模板图像的像素数列中第 i 个点的像素值; $E(X)$ 为待
识别图像的像素均值, $E(X) = \frac{1}{P \times Q} \sum_{i=0}^{P \times Q} X_i$; $E(Y)$ 为与待识别
图像区域重叠的模板图像的像素均值, $E(Y) = \frac{1}{P \times Q} \sum_{i=0}^{P \times Q} Y_i$ 。

随后按照从左到右、从上到下的顺序依次移动一个像素,
并计算出相应的互相关值 $cc(i,j)$,最后取最大值作为待识别
图像和模板的相关度。

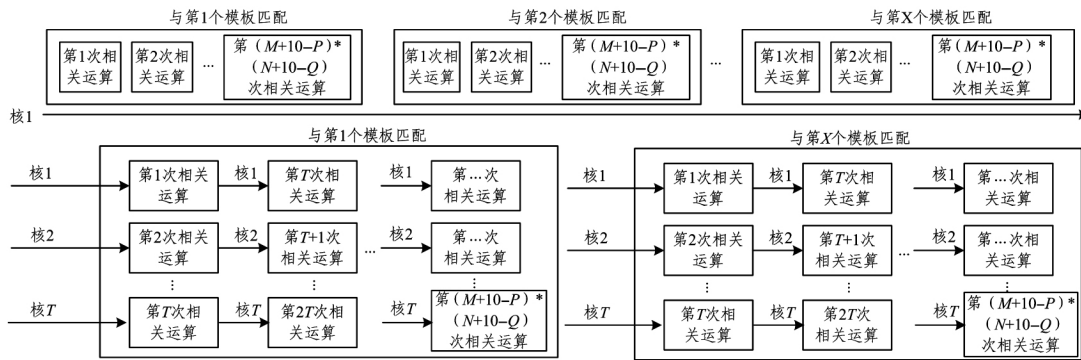


图 6 串行计算与多核并行计算流程对比

图 6 是针对分类识别算法,计算机处理的流程对比。可
以看到,如果采用原有的单核计算,每种模板以及每个模板中
的每次运算都需要串行处理;采用 OpenMP 多核技术后,对
每类模板采取并行计算任务分配,将每类模板中的每次相关
计算均匀地分配到各个不同的核进行运算,这样可以充分利
用所有的计算机处理器资源。基于模板的分类识别的最大缺
点就是耗时,通过上面的分析可以看出,并行计算在解决这一
问题时具备天然的优势。

3.2 主要实现

本文采用 SPMD 模式来动态获取计算机的处理器个数,
给每个处理器分配不同的计算任务,最后将每个处理器的计
算结果进行合并,从而实现 SAR 图像处理的并行计算。部分
主要代码如下:

```
int coreNum = omp_get_num_procs();
//当前机器的核数
float ** cutTemple = new float * [coreNum];
//二维指针数组,分配空间
for(int i=0; i<coreNum; i++)
{
    cutTemple[i] = new float[nROIWidth * nROIHeight];
}
float max_pro = 0.0;
float dProbability = 0;
//对多核进行任务分配
#pragma omp parallel for
for(int dk=0; dk<iHeightTempl-nROIHeight; dk++)
{
    for(int pk=0; pk<iWidthTempl-nROIWidth; pk++)
```

```
{
    int nMinPos = dk * iWidthTempl + pk;
    //每周 ROI 挪动到模板进行比较的起始位置
    float dProba = 0;
    int coreID = 0;
    for(int j1=0; j1<nROIHeight; j1++)
    {
        for(int i1=0; i1<nROIWidth; i1++)
        {
            int nPos1 = nMinPos + i1 * iWidthTempl + j1;
            coreID = omp_get_thread_num(); //获取当前处理该任务的核编号
            cutTemple[coreID][i1 + j1 * nROIWidth] = oTemplate.m_pTemplateData[nPos1]; //模板数据获取
        }
        dProba = Match2D(pDataFromROI, cutTemple[coreID], nROIHeight, nROIWidth); //匹配识别
    }
    #pragma omp critical
    //公用数据 max_pro, 加锁防止冲突
    {
        if(dProba > max_pro)
            max_pro = dProba;
    }
}
dProbability = max_pro; //获取最大的匹配概率
```

(下转第 187 页)

2010;943-946.

- [3] 陈双叶,周耳江,吴强. ND-GSM 模型的采样矩阵方向优化及 SAR 图像去噪[J]. 计算机科学,2015,42(6A):158-162,167.
- [4] BUADES A,COLL B. A non-local algorithm for image denoising [J]. IEEE Computer Society Conference on Computer Vision & Pattern Recognition,2005,2(7):60-65.
- [5] KERVRANN C,BOULANGE R J. Optimal spatial adaptation for patch-based image denoising[J]. IEEE Transactions on Image Processing,2006,15(10):2866-2878.
- [6] DABOV K,FOI A,KATKOVNIK V, et al. Image denoising by sparse 3-d transform-domain collaborative filtering [J]. IEEE Transactions on Image Processing,2007,16(8):2080-2095.
- [7] 范云鹏. 矩阵低秩逼近在图像压缩中的应用[D]. 西安:西安电子科技大学,2012:26-30.
- [8] 赵新斌. 一类带有核范数的优化问题的梯度算法[D]. 北京:北京工业大学,2012:13-25.
- [9] CANDÈS E J,Recht B. Exact matrix completion via convex optimization [J]. Foundations of Computational mathematics,2012,55(6):111-119.
- [10] GU S,ZHANG L,ZUO W, et al. Weighted nuclear norm minimization with application to image denoising[C]//CVPR 2014. 2014:2862-2869.
- [11] DONG W S,SHI G M,LI X. Nonlocal image restoration with bilateral variance estimation: A Low-rank Approach[J]. IEEE

Transactions on ImageProcessing,2013,22(2):700-711.

- [12] 王圳萍,张家树,陈高. 加权低秩矩阵恢复的混合噪声图像去噪[J]. 计算机科学,2016,43(1):298-301.
- [13] 黄之娟,唐超颖,陈跃庭,等. 基于非局部相似性和低秩矩阵的遥感图像重构方法[J]. 光学学报,2016,36(6):97-107.
- [14] DABOV K,FOI A,KATKOVNIK V, et al. Image restoration by sparse3D transform-domain collaborative filtering[J]. Electronic Imaging,International Society for Optics and Photonics,2008,6812(8):2080-2095.
- [15] NEJATI M,SAMAVI S,DERKSEN H. Denoising by low-rank and sparse representations[J]. Journal of Visual Communication & Image Representation,2016,36(C):28-39.
- [16] LIU X,TANAKA M,OKUTOMI M. Noise level estimation using weak textured patches of a single noisy image[C]//2012 19th IEEE International Conference. 2012:665-668.
- [17] KHAN S,JAIN A,KHARE A. Image denoising based on adaptive wavelet thresholding by using various shrinkage methods under different noise condition[J]. International Journal of Computer Application,2012,59(20):13-17.
- [18] GERARDO D M,MARIANA P,Giovanni P. Benchmarking Framework for SAR Despeckling[J]. IEEE Transactions on Geoscience and Remote Sensing,2014,52(3):1596-1615.
- [19] 郝强,周敏,郑红婵. 基于边缘检测和四点插值细分的图像去噪[J]. 计算机工程与应用,2014,50(11):184-187.

(上接第 163 页)

4 性能测试及分析

针对以上的设计,在同样的环境下分别对传统的计算方式和本文所提的 OpenMP 多核技术进行测试,测试的环境如表 1 所列。

表 1 测试环境

主机配置项	指标
CPU	16 核,主频:2GHz
内存	64GB
硬盘	2T
操作系统	WIN7 64 位
软件环境	QT+VS2013+OpenMP

在实验样本方面,分别针对小型飞机、中型飞机和舰船开展单核和多核性能的对比试验结果如图 7 所示。由于并不是所有的计算都可以采用并行化处理,同时并行化处理过程中部分任务的拆分和数据的合并也需要时间消耗,因此 16 个核满负荷工作情况下不可能产生 16 倍的效率提升。通过实验可以看到,采用 OpenMP 多核技术,整体 SAR 图像处理在速度上可以提升 8 倍左右。

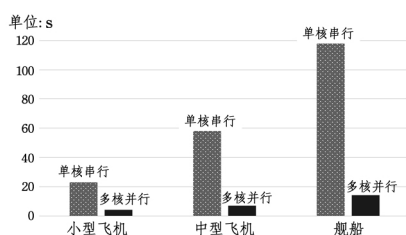


图 7 性能对比分析

结束语 随着计算机技术的发展和多核处理器的普及,OpenMP 作为多核结构编程模型的有力竞争者,在众多领域得到了越来越多的应用。SAR 图像本身数据尺寸巨大,计算过程复杂,对时效性要求高,这些特征决定了其对并行计算处理的依赖性,因此 SAR 图像处理可以和 OpenMP 技术进行良好的应用结合。

本文阐述了 OpenMP 的体系结构,分析了 OpenMP 的并行机制,并对基本的指令语句进行了介绍。本着减少程序运行时间的原则,利用 OpenMP 技术在 SAR 图像处理的模板比对过程中进行多核并行处理,并针对不同对象开展了实验验证,结果表明程序的性能较原有串行程序有了很大的提升。将来主要在并行任务优化划分上继续展开深入研究。

参 考 文 献

- [1] 张彩明,杨兴强,李学庆,等. 计算机图形学[M]. 北京:科学出版社,2005.
- [2] CHANDRA R,DAGUM L,KOHR D, et al. Parallel Programming in OpenMP[M]. Morgan Kaufmann,2000.
- [3] CHAPMAN B,JOST G,VAN DER P R. Using OpenMP[M]. The MIT Press,2007.
- [4] OpenMP Architecture Review Board[OL]. <http://www.openmp.org>.
- [5] 孙洪迪,高柱. 基于 OpenMP 技术的多核处理器程序的开发实现[J]. 北京工业职业技术学院学报,2010,9(1):19-22.
- [6] 刘晓娟,赵荣彩,韩林. 一种基于 OpenMP 的 DSWP 自动并行算法[J]. 信息工程大学学报,2015,16(2):225-233.
- [7] 马旭龙,林峰. 基于 OpenMP 的快速并行分层算法[J]. 计算机辅助设计与图形学学报,2015,27(4):747-751.