

# 任务分配均衡的双参数CFAR舰船检测并行算法

张临杰<sup>1</sup>, 张杰<sup>2</sup>, 张晰<sup>2</sup>, 郎海涛<sup>3</sup>

1. 中国海洋大学 数学科学学院, 山东 青岛 266100;

2. 国家海洋局 第一海洋研究所, 山东 青岛 266061;

3. 北京化工大学 应用物理系, 北京 100029

**摘要:** 双参数恒虚警率CFAR(Constant False Alarm Rate)是舰船目标检测中的常用算法。近年来, 合成孔径雷达(SAR)分辨率不断提高, SAR图像幅宽增大, 并且在检测时希望尽量保持舰船轮廓以便后续的舰船目标识别。双参数CFAR算法虽然能满足目标检测需求, 但算法运行时间过长, 不利于信息的及时处理。传统的MPI(Message Passing Interface)并行化解决方案在分配检测任务给各进程时, 没有考虑因陆地掩膜, 几何校正等预处理所导致的图像中待检测点分布不均。针对这一问题, 本文提出改进的MPI并行化解决方案。与传统的MPI并行化解决方案相比, 该方案能较为均衡地为各个进程分配检测任务。在集群计算机上的实验结果表明, 改进后并行标准效率提高约43%。为应对机载SAR实时舰船目标检测的需求, 在多核PC机上进行实验, 结果表明, 本文算法在多核PC机上也能有效地缩短检测时间, 对实现机载SAR实时舰船目标检测有积极意义。

**关键词:** 舰船检测, 双参数恒虚警率CFAR, 合成孔径雷达SAR, MPI并行化, 检测窗口尺寸

**中图分类号:** TP751 **文献标志码:** A

**引用格式:** 张临杰, 张杰, 张晰, 郎海涛. 2016. 任务分配均衡的双参数CFAR舰船检测并行算法. 遥感学报, 20(2): 344-351

Zhang L J, Zhang J, Zhang X and Lang H T. 2016. Task distribution balancing for parallel two-parameter CFAR ship detection. Journal of Remote Sensing, 20(2): 344-351 [DOI: 10.11834/jrs.20165006]

## 1 引言

合成孔径雷达SAR具有全天候、全天时、大范围等诸多优点, 在舰船检测方面已被证明是一种有效的手段, 一直以来受到广泛的关注。

双参数恒虚警率CFAR(Constant False Alarm Rate)是一种自适应门限检测, 能够适应背景杂波的变化, 是常用的舰船目标检测方法之一(EI-Darymli 等, 2013)。近年来由于传感器技术的进步, SAR分辨率已能达到亚米级。随着分辨率的提高, SAR图像幅宽大幅增加, 长、宽均超过一万个像素点的情况已经比较常见; 舰船目标不再以点目标的形式出现, 而是以硬目标的形式出现, 目标轮廓较为清晰, 因此在检测的同时还要求尽可能地保持目标轮廓; 此外对小型舰船的检测也提出了要求。这些原因导致在使用双参数CFAR进行

检测时, 计算时间过长, 不利于信息的及时处理。利用并行处理技术是缩短检测时间的有效解决方法。

目前主流的并行处理技术有基于openMP、基于MPI和基于GPU共3种, 分别适用于多核计算机, 集群计算机和带GPU计算卡的计算机。由于SAR成像中处理的数据量大, 因此以上并行技术在SAR图像成像中的应用已引起广泛重视, 例如李景山等人(2014)利用MPI并行处理技术, 在集群计算机上实现了的环境一号C卫星SAR图像全分辨率快速实时处理系统; 赵敬亮等人(2010)和孟大地等人(2013)利用基于GPU的并行处理技术, 研究了SAR实时成像处理; Park等人(2012)利用openMP技术在配备了众核加速卡(Intel R Xeon Phi TM coprocessors)的单节点计算机上, 实现了秒以内 $3000 \times 3000$ 的图像成像。

收稿日期: 2015-01-27; 修订日期: 2015-09-14; 优先数字出版日期: 2015-09-21

基金项目: 国家自然科学基金(编号: 11371333); 山东省自然科学基金(编号: ZR2013FQ026); 中央高校基本科研业务费专项(编号: 201362033); 海洋公益性行业科研专项经费项目(编号: 201505002)

第一作者简介: 张临杰(1975—), 女, 博士研究生, 从事基于SAR图像的舰船目标识别的理论和应用研究。E-mail: zhanglinjie@ouc.edu.cn

通信作者简介: 张杰(1963—), 男, 教授, 从事于海洋遥感学研究。E-mail: zhangjie@fio.org.cn

由于SAR分辨率不断提高，对基于SAR图像的目标识别算法的并行化研究也逐渐引起重视：林旭等人(2013)利用openMP并行技术对双参数CFAR算法进行了并行化，在4核计算机上将检测时间缩短了约75%。目前，基于MPI和GPU的舰船目标检测算法的并行化研究在中国仍存在很大空白。

考虑到集群计算机目前已较为普遍，且具有不受单个节点内存和计算能力的限制，因此本文采用MPI并行技术对双参数CFAR进行并行化。传统的MPI并行方案按图像尺寸平均分割检测任务。然而我们在执行实际检测任务时，经常要对待检测的SAR图像先做陆地掩膜，几何校正等预处理。这些预处理会导致处理后的SAR图像中待检测点分布不均。众所周知，在未进行MPI并行时，这种待检测点的分布不均并不影响检测效率。然而在并行时则会导致各进程所负责的检测任务不均衡，有的进程完成地快，有的进程完成地慢。而并行检测时间是由最慢的那个进程所决定的。

针对这一问题，本文给出了一个改进的MPI并行化解决方案。与传统的解决方案相比，该方案考虑了预处理所导致的待检测点分布不均，按照待检测点个数将检测任务较为均衡地分配给各个进程，从而缩短整体检测时间。实验结果表明，该方案无论是在集群计算机，还是在多核PC机上的都能够有效地缩短检测时间，提高加速比和标准效率，是一个行之有效的并行解决方案。

## 2 双参数CFAR检测算法

双参数CFAR检测使用3个窗口：目标窗口T，保护窗口P和背景窗口B，分别对应图1中的斜纹区域、空白区域和横线区域。目标窗口内是待检测的点，海面背景信息从背景窗口获得，保护窗口是为了确保舰船目标部分不会被包含在背景窗口中。目标窗口内的点被检测为目标的判断标准是

$$\frac{\mu_t - \mu_b}{\sigma_b} > t \Leftrightarrow \text{目标}$$

式中， $\mu_t$ 为目标窗口内的均值， $\mu_b$ 为背景窗口均值， $\sigma_b$ 为背景窗口内的标准偏差， $t$ 控制虚警率，也称为标称化因子(一般取2或3)，式中不等式左边也被称为检测比率。在对整个SAR图像进行检测时，上面3个窗口按照一定的步长(一般与目标窗口尺寸取相同值)在图像中滑动，逐一判断目标窗口内的点是否为目标点(种劲松等，2006)。

由上面简述可知，在双参数CFAR检测中需要指定3个窗口的尺寸(图1中 $a, b, c$ )和标称化因子 $t$ 。一般来说，保护窗口的尺寸要足够大，以免舰船目标的一部分被包含在背景窗口中，从而增加背景均值和方差，使得检测比率减小；同样背景窗口的尺寸不宜过大，以免包含临近目标，减小检测比率；目标窗口尺寸越小越能够降低漏检的概率，但由于目标窗口要遍历整个图像，因此计算量也随之大大增加。

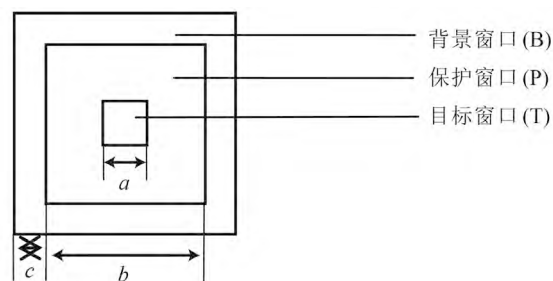


图1 双参数CFAR检测窗口

Fig. 1 Detection window of two-paramter CFAR (B: background window; P: guard window; T: target window)

在分辨率较低的SAR图像中，舰船目标表现为点目标(图2)，仅占若干个像素点，轮廓不清晰。当使用双参数CFAR检测时，一般取目标窗口尺寸(即图1中 $a$ 值)为小船目标船长的1—2倍，保护窗口尺寸(即图1中 $b$ 值)为大船目标的2倍，背景窗口尺寸(即图1中 $c$ 值)通常取3以上的整数(种劲松等，2006)。

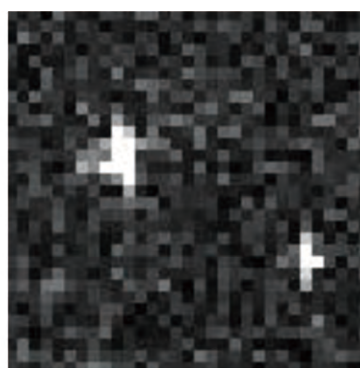


图2 分辨率为25 m时的舰船目标

Fig. 2 Ship targets when the image resolution is 25 m

当分辨率较高时，舰船目标在图像中以硬目标的形式出现，若仍采用上述选取方案，一是容易引起图3所示的漏检，二是无法很好地保留舰船轮廓，也是高分辨率舰船检测的实际需求之一。

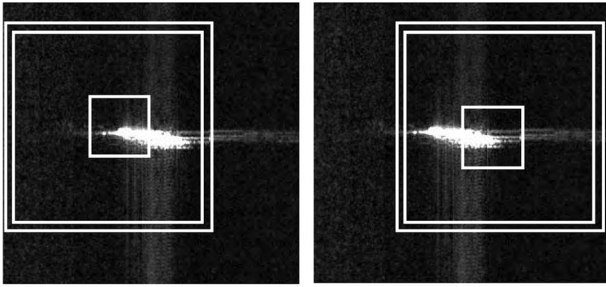


图3 舰船目标以面目标形式出现时的漏检例  
Fig. 3 Undetected targets in high resolution

舰船目标轮廓保持情况直接受目标窗口尺寸的影响(图4)。由此可知保持舰船轮廓的最直接的解决方案就是不再依据小船船长设定目标窗口尺寸,而是尽可能小地设置目标窗口尺寸(最小值为1)。而减小目标窗口尺寸又带来了计算量的大幅增加。

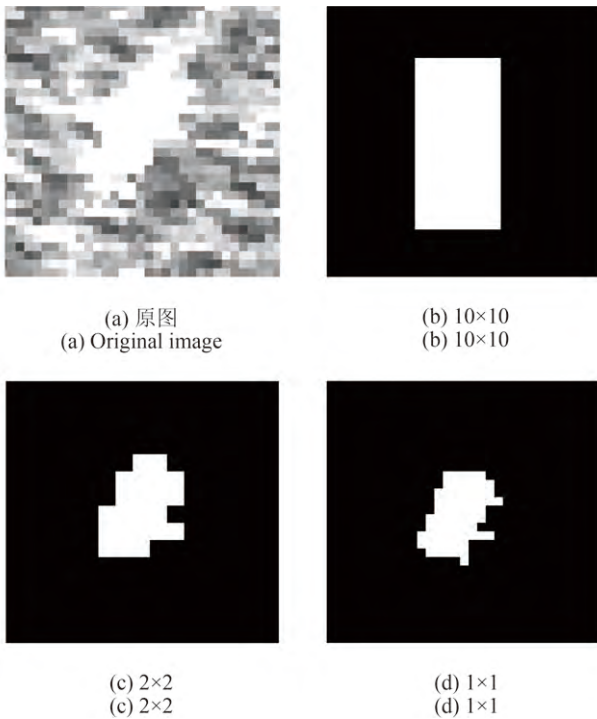


图4 不同目标窗口尺寸时的舰船轮廓比较  
Fig. 4 Ship contour in different sizes of target window

综上所述,将双参数CFAR检测算法应用于高分辨率的目标检测时,有两个原因导致计算量增大:一是图像本身尺寸增大直接导致计算量增大;二是为防止漏检,以及很好地保留舰船轮廓,而减小目标窗口尺寸所引起的计算量增大。

在对一幅大尺寸高分辨率图像进行检测时,检测时间长达几分钟或者更长,妨碍了舰船目标实时检测的实现。为缩短检测时间,考虑对双参数CFAR检测算法进行并行化处理。

### 3 传统的MPI并行化解决方案

对双参数CFAR检测算法进行MPI并行化时,通常将SAR图像平分为 $n$ 份,每个进程负责其中一份的检测,最后再将所有检测结果汇总。分割方案一般有水平、垂直和网格分割3种。由于双参数CFAR检测算法的目标判断标准仅依据图1所示的3个局部窗口内的信息,因此无论哪种分割都不影响最终检测结果。而本文所使用的SAR图像数据是按行存放,因此采用水平分割方式。另外,各进程在读取所负责的子图像数据时,应在子图的上下两个方向各多读取一个冗余区域的数据,以确保所负责子图的上下两端边界部分的检测能够正常进行(图5)。冗余区域的宽度应同图1中目标窗口左上角与背景窗口左上角的垂直距离相等。注意第1个进程只有下方的冗余区域,最后一个进程只有上方的冗余区域。并行步骤详述如表1。

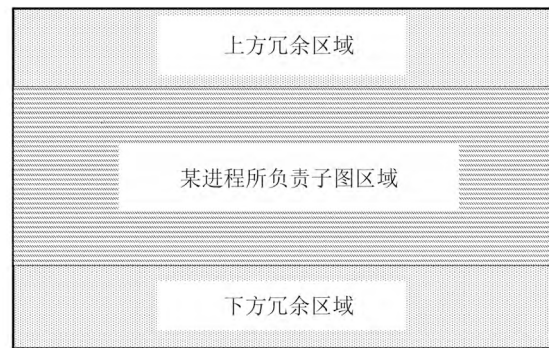


图5 某进程读取图像数据示意图  
Fig. 5 Part of image to be imported for a process

下面以一幅实际的SAR图像为例(图6),说明上面传统并行方案所存在的问题。该图像是一个经过了几何校正和陆地掩膜后的星载16位SAR图像,尺寸为 $12477 \times 14569$ ,分辨率约为11.83 m。检测窗口尺寸,即图1中的 $a$ 、 $b$ 、 $c$ 值分别设为2、30和3。实验在有3个节点的集群计算机上进行。每个节点配置如下:操作系统为SUSE Linux Enterprise Server 10,内存为4 GB,处理器为两个4核AMD Opteron(tm) Processor 6128 800 MHz处理器。

表1 传统的MPI并行步骤  
Table 1 Traditional MPI parallel steps

进程0	其他进程
步骤1 计算本进程所负责子图像范围区间	同进程0
步骤2 从文件读入本进程所需子图像数据	同进程0
步骤3 对本进程所负责子图进行双参数CFAR检测	同进程0
步骤4 收集其他进程的检测结果并汇总	将检测结果发送给进程0

Processor 0	0-1559 (1560)
Processor 1	1560-3119 (1560)
Processor 2	3120-4679 (1560)
Processor 3	4680-6239 (1560)
Processor 4	6240-7799 (1560)
Processor 5	7800-9358 (1559)
Processor 6	9359-10917 (1559)
Processor 7	10918-12476 (1559)

图6 水平平均分割图像结果

Fig. 6 Divide image horizontally in average

未进行MPI并行化时总检测时间为489.21 s, 其中CFAR算法运行时间为488.35 s, 其余时间为读写文件所用时间。将图像平分为8份, 每个进程(Processor)负责1份。各进程所负责的行范围区间见图6, 并行结果见表2与图7。

表2 改进前各进程计算时间  
Table 2 Detection time, waiting time and total time of each process before improvement

进程号	CFAR检测时间	通信等待时间	总时间
0	20.00	73.83	93.97
1	50.91	0.60	51.60
2	92.38	1.17	93.66
3	89.25	0.14	89.51
4	89.94	1.13	91.08
5	85.14	0.84	86.09
6	56.67	0.17	56.96
7	16.76	4.17	21.04

由并行结果可以看出, 经过几何校正和陆地掩膜后的SAR图像待检测点分布不均匀, 若将图像平均分配给各进程, 将导致各进程检测时间严重不均衡。例如进程0的CFAR检测时间仅20.00 s,

然而进程4的检测时间却长达89.94 s, 相差4倍以上。进程0在完成所负责子图的检测后, 必须等待所有进程完成检测, 才能收集汇总检测结果并输出, 等待时间高达73.83 s。加速比 $S_p = t_1/t_p$  ( $t_1$ 为非并行时的运行时间,  $t_p$ 是在有 $p$ 个进程时的运行时间)仅为5.20, 标准效率 $E_p = S_p/p = 0.65$ 。

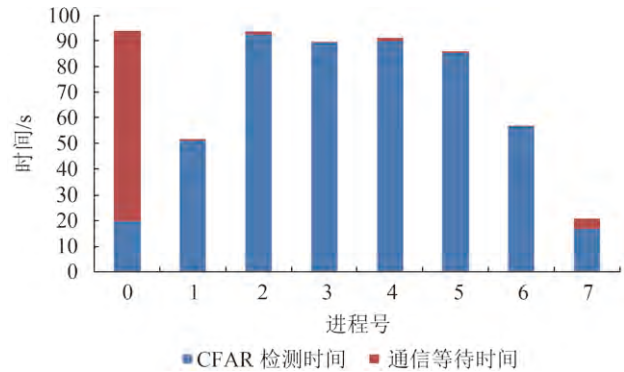


图7 改进前各进程检测时间与等待时间

Fig. 7 Detection time and waiting time of each process before improvement

## 4 改进的MPI并行化解决方案

为避免上述各进程负载不均衡问题, 本文所给出的并行化解决方案, 将根据待检测点个数分割图像。分割方案如下:

(1)统计整幅图像的待检测点总数 $n$ ;

(2)计算每个进程所负责的检测点个数 $m$ , ( $m = n/p$ , 其中 $p$ 为进程数);

(3)重新自上而下逐行遍历图像, 根据 $m$ 值逐一确定每个进程所负责的行区间范围。

为简化算法的实现, 进程间任务分割的最小单位为行, 因此每个进程所负责的检测点数并不严格等于 $m$ , 但与 $m$ 值最多相差不到1行像素点数, 即图像的列数; 各进程在读取所负责的子图像数据时, 仍和传统的解决方案一样, 在所负责子图的上下两个方向各读取一个冗余区域的数据, 以确保所负责子图的上下两端边界部分的检测能够正常进行; 本实验中的待检测点即为非零点, 在实际应用中也可以根据经验设定一个非零的全局阈值, 或是使用其他算法计算出一个全局阈值, 然后根据该阈值判断是否为待检测点; 与传统的解决方案相比, 本文仅改变了任务分配方式, 没有改变双参数CFAR的检测机制, 因此对检测结果没有影响, 这一点在随后的实验中也得到了证实。

最后给出并行方案的详细步骤流程如表3所示。

表3 改进的MPI并行步骤  
Table 3 Improved MPI parallel steps

进程0	其他进程
步骤1 读入整个图像, 统计待检测点总数, 确定各个进程所负责图像范围区间, 并将分割方案发送给其他进程	从进程0接收图像分割方案, 明确本进程所负责图像范围区间
步骤2 从文件读入本进程所需子图像数据	同进程0
步骤3 对本进程所负责子图进行双参数CFAR检测	同进程0
步骤4 收集其他进程的检测结果并汇总	将检测结果发送给进程0

在步骤1中, 让进程0负责统计待检测点总数, 也可以将统计任务分配给每个进程并发执行。但是考虑到多个进程同时读取一个文件会大大降低文件读取速度, 并且在内存满足需求的情况下, 读取文件一部分和读取整个文件的时间差别不明显, 因此本文对步骤1没有并发执行。

下面仍以图6所示图像为例进行试验, 进程数仍指定为8, 分割结果见图8, 并行结果见表4与图9。

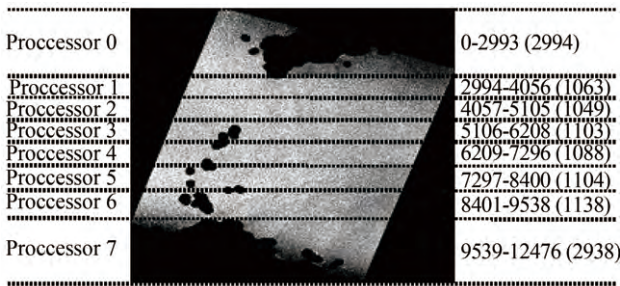


图8 根据待检测点个数分割图像结果

Fig. 8 Divide image in terms of number of points to be detected

表4 改进后各进程计算时间

Table 4 Details of each process after improvement /s

进程号	CFAR检测时间	通信等待时间	总时间
0	61.02	1.19	65.65
1	59.19	2.00	64.51
2	59.01	2.28	64.63
3	59.24	2.16	64.75
4	59.17	2.43	64.93
5	59.22	2.53	65.07
6	59.43	2.50	65.25
7	60.58	1.45	65.47

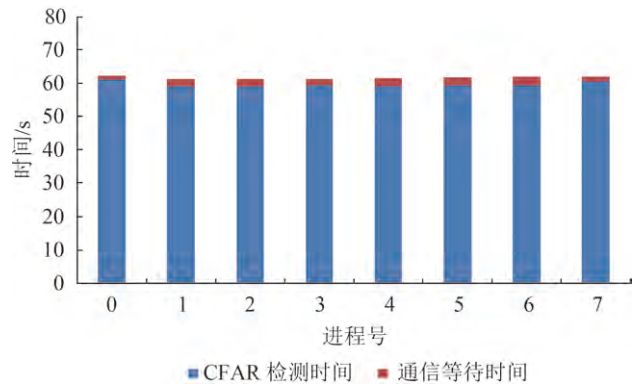


图9 改进后各进程的检测时间与等待时间

Fig. 9 Detection time and waiting time of each process after improvement

由检测结果可以看出, 与传统的并行方案相比, 各进程的CFAR检测时间接近均衡, 加速比7.45, 标准效率0.93, 都有了很大提高。虽然在第一步统计待检测点总数时, 需要读入整个SAR数据文件, 有一定的时间花费(约3 s), 但是从总计算时间的缩短情况来看, 这里的计算花销是值得的。

改进前后的并行效率比较见表5, 与传统的并行方案相比, 本文所给并行化解决方案, 有效地提高了并行效率。

表5 改进前后并行效率比较

Table 5 Comparison of the parallel efficiency before and after improvement

	总时间/S	加速比	标准效率
非并行时	489.21	---	---
改进前	93.97	5.20	0.65
改进后	65.65	7.45	0.93

检测结果与传统并行方案检测结果相同, 共检测出疑似目标583处, 检测结果见图10。由于本实验图像尺寸远超屏幕分辨率, 因此仅展示了整体检测结果的一部分。所展示局部左上角在整幅检测结果图中的坐标为(8860, 10715), 大小为300 × 300。图10还给出了其中一个疑似目标的放大图, 可以看出舰船轮廓被较好地保留了下来。

另外需要说明的是, 在执行舰船检测任务时, 还须在CFAR检测结果的基础上, 利用形态学的方法, 对疑似目标进行进一步地筛选, 剔除虚警。由于此部分超出本文所论范围, 因此略过。

图11给出不同进程数时的加速比。从图中可以观察到, 随着进程数的增加, 加速比呈增加趋

势。但是增加趋势有所变缓，即标准效率有所下降。原因有两个：(1)随着进程数的增加，并行方案中的步骤3的执行时间减少，在总计算时间中所占比例下降。而未被并发执行的步骤1的执行时间不因进程数的增加而改变，因而在总计算时间中所占比例增大，影响了标准效率的提高；(2)并行方案中的步骤2，即从图像数据文件读入本进程所需数据，当多个进程同时对同一个数据文件进行读取时，会在一定程度上影响各进程的读取速度。由以上分析可知，当进程数增加到一定程度，加速比将不再随进程数的增加而增大。

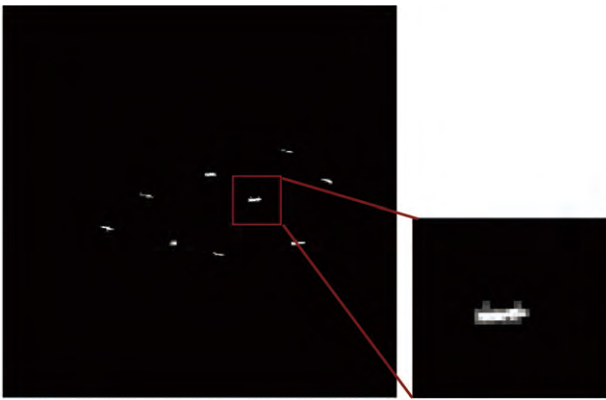


图 10 检测结果局部

Fig. 10 One part of detection result

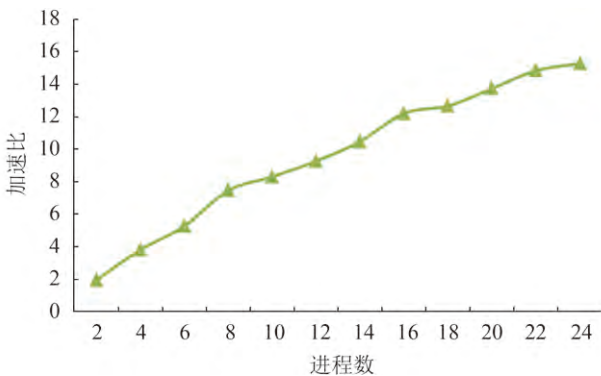


图 11 不同进程数时的加速比

Fig. 11 Speed-up ratio with different number of processes

## 5 多核PC机上的并行效果

MPI并行程序一般在集群计算机上执行。而有时在执行检测任务时无法使用集群计算机。例如在搭载机载SAR的飞行器上进行目标检测时，受飞行器空间限制无法搭载集群计算机，只能使用性

能较好的PC机。目前，绝大多数PC机都配备了多核处理器，甚至配备多个多核处理器。为考察上面MPI并行程序在多核PC机上的效果，在多核PC机上进行试验。试验用机为32位Windows XP操作系统，4 GB内存，2个4核Quad-core AMD opteron (tm) processor 2378 792 MHz处理器。

非并行时总检测时间为355.21 s。通过任务管理器观察到，在程序运行的不同阶段，各核使用情况大体分为3种，见图12(a)—(c)。无论是哪一种情况，CPU使用率都基本保持在13%左右，即在同一时刻仅1个核的资源被使用。执行上面MPI并行程序，并指定进程数为8时总检测时间缩短为50.22 s，加速比7.07，标准效率0.88。通过任务管理器，可以观察到8个核都被充分使用，CPU使用率达到100%，见图12(d)。

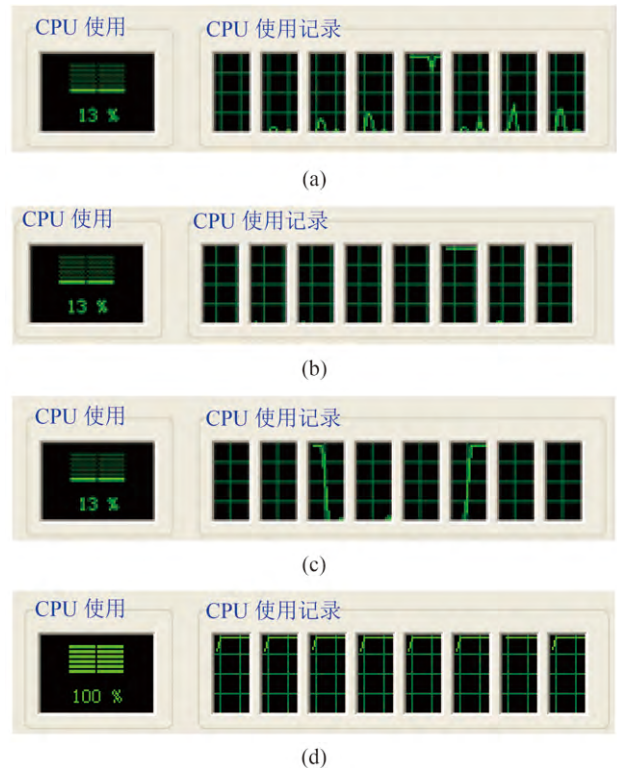


图 12 Windows任务管理器截图

Fig. 12 Screen shot of Windows task manager

## 6 结 论

传统的双参数CFAR的MPI的并行化解决方案，是将待检测图像按照图像的尺寸平均分配给各个进程。在实际执行目标检测任务时，往往会在检测前对SAR图像先进行陆地掩膜，几何校正等

预处理,使得处理后的SAR图像中待检测点分布不均,从而导致进程间任务分配不均衡,影响并行效率的发挥。

针对这一问题,本文所给解决方案依照待检测点个数为各个进程均衡分配检测任务。实验结果表明,该方案无论是在集群计算机上,还是多核PC机上都能有效地缩短检测时间,提高并行效率。尤其是在多核计算机上缩短检测时间对实现机载SAR舰船目标实时检测具有积极意义。

另外,本文所给出的并行方案不仅适用于对经过陆地掩膜或几何校正后的SAR图像进行检测,而且对一些复合检测算法(如Ji等人(2010)提出的先用Kfar算法计算出一个全局阈值,然后再对大于该阈值的点进行双参数CFAR检测)也同样适用。

在执行任务的飞机上搭载配备GPU或Intel众核加速卡的非集群计算机,利用GPU和openMP并行技术进一步缩短检测时间,实现机载SAR舰船目标实时检测,将是下一步的重要研究方向。

## 参考文献(References)

- El-Darymli K, McGuire P, Power D and Moloney C. 2013. Target detection in synthetic aperture radar imagery: a state-of-the-art survey. *Journal of Applied Remote Sensing*, 7(1): 071598 [DOI: 10.1117/1.JRS.7.071598]
- Ji Y G, Zhang J, Meng J M and Zhang X. 2010. A new CFAR ship target detection method in SAR imagery. *Acta Oceanologica Sinica*, 29(1): 12 – 16 [DOI: 10.1007/s13131-010-0002-6]
- Li J S, Wen S Y, Wang J, Zhong L H and Zhang W Y. 2014. Design and implementation of a real-time processing system of full resolution quick-look image of HJ-1 environmental satellite C SAR based on high performance cluster. *Journal of Radars*, 3(3): 332 – 338 (李景山, 温双燕, 王建, 仲利华, 张问一. 2014. 基于高性能机群的环境一号C卫星SAR图像全分辨率快视实时处理系统设计与实现. *雷达学报*, 3(3): 332 – 338) [DOI: 10.3724/SP.J.1300.2014.13095]
- Lin X, Hong J, Sun X and Yan Y. 2013. Fast ship detection method based on ScanSAR image. *Journal of University of Chinese Academy of Sciences*, 30(6): 793 – 799 (林旭, 洪峻, 孙显, 鄢懿. 2013. ScanSAR图像舰船目标快速检测方法. *中国科学院大学学报*, 2013, 30(6): 793 – 799) [DOI: 10.7523/j.issn.2095-6134.2013.06.012]
- Meng D D, Hu Y X, Shi T, Sun R and Li X B. 2013. Airborne SAR real-time imaging algorithm design and implementation with CUDA on NVIDIA GPU. *Journal of Radars*, 2(4): 481 – 491 (孟大地, 胡玉新, 石涛, 孙蕊, 李晓波. 2013. 基于NVIDIA GPU的机载SAR实时成像处理算法CUDA设计与实现. *雷达学报*, 2(4): 481 – 491) [DOI: 10.3724/SP.J.1300.2013.13056]
- Park J, Tang P T P, Smelyanskiy M, Kim D and Benson T. 2012. Efficient Backprojection-based synthetic aperture radar computation with many-core processors // *Proceedings of International Conference on High Performance Computing, Networking, Storage and Analysis*. Salt Lake City, Utah: IEEE: 1 – 11 [DOI: 10.1109/SC.2012.53]
- Zhao J L, Lei W M and Huang Y H. 2010. Real-time parallel imaging processing of SAR based on GPU. *Informatization Research*, 36(2): 28 – 30, 49 (赵敬亮, 雷万明, 黄银和. 2010. 基于GPU的SAR实时并行成像处理. *信息化研究*, 36(2): 28 – 30, 49) [DOI: 10.3969/j.issn.1674-4888.2010.02.008]
- Zhong J S, Ouyang Y and Zhu M H. 2006. *Ocean Target Detection in Synthetic Aperture Radar*. Beijing: Ocean Publishing Press: 47 – 49 (种劲松, 欧阳越, 朱敏慧. 2006. 合成孔径雷达图像海洋目标检测. 北京: 海洋出版社: 47 – 49)

# Task distribution balancing for parallel two-parameter CFAR ship detection

ZHANG Linjie<sup>1</sup>, ZHANG Jie<sup>2</sup>, ZHANG Xi<sup>2</sup>, LANG Haitao<sup>3</sup>

1. College of Mathematical Science, Ocean University of China, Qingdao 266100, China;

2. The First Institute of Oceanography, State Oceanic Administration, Qingdao 266061, China;

3. Physics and Electronics Department, Beijing University of Chemical Technology, Beijing 100029, China

**Abstract:** Ship detection is important in military and civilian applications. Synthetic Aperture Radar(SAR) with all-day, all-weather, and ultra-long-range characteristics has been extensively used. The two-parameter Constant False Alarm Rate(CFAR) method is one of the most well-known methods for target detection. CFAR is an adaptive threshold detection scheme that works efficiently when the background clutter is unevenly distributed. However, in recent years, the resolution of SAR images is significantly improved by the rapid development of the SAR sensor. With the improvement of the resolution, the size of SAR images significantly increased and the ship targets no longer appear as point targets. Instead, the ship targets appear as hard targets. The contour of the targets becomes clearer as well. When the two-parameter CFAR is used to detect ship targets with good contour, a longer computation time is needed. Message Passing Interface (MPI) parallelization is a workable solution used to shorten the computation time of two-parameter CFAR with MPI parallel technique.

The traditional MPI parallelization divides the SAR image horizontally/vertically on average. However, in practical applications, preprocessing methods, such as land masking and geometric correction, are conducted before detection. These preprocessing methods can cause the uneven distribution of the points to be detected. This uneven distribution leads to the unbalanced tasks between the parallel processes. Thus, the efficiency of MPI parallelization is highly influenced. The objective of this study is to eliminate the negative influence caused by the uneven distribution.

In this study, we propose an improved MPI parallel solution of the two-parameter CFAR ship detection method, in which the SAR image is divided in terms of the number of points to be detected. The partitioning strategy includes: First, the total number of points to be detected is calculated. Second, the approximate number of responsible points for each process is computed. Third, the responsible rows of image for each process are identified.

In this manner, the entire detection task is equally divided among the processes. The details of the improved parallel algorithm are listed as below: (1) The first process computes the partitioning strategy and transmits it to the other processes. (2) Each process imports its responsible part of the image. (3) Each process implements two-parameter CFAR detection on its responsible part of the image. (4) The first process obtains the detection results from the other processes.

The numerical experiment is conducted on a cluster computer. When the number of processes is 8, the speedup of the improved parallel algorithm is 7.45, which is better than that of the normal parallel algorithm. The efficiency of parallelization increases by approximately 43%. A similar experiment is conducted on a multicore computer, and a similar result is obtained.

The experimental results show that the improved parallel solution can shorten the detection time and improve the parallel efficiency of the cluster or multicore computer. This study is positively significant for real-time ship detection based on airborne SAR images. Further research is needed to shorten the detection time by using the GPU or Intel MIC architecture.

**Key words:** ship detection, Constant False Alarm Rate(CFAR), SAR images, Message Passing Interface (MPI) parallelization, size of detection window

**Supported by** National Natural Science Foundation of China (No. 11371333); Shandong Provincial Natural Science Foundation, China (No. ZR2013FQ026); Public Science and Technology Research Funds Projects of Ocean (No. 201505002)