



# 计算机科学

COMPUTER SCIENCE

## 高分辨率斜视聚束 SAR 回波仿真加速算法研究

郭拯危, 付泽文, 李宁, 白澜

### 引用本文

郭拯危, 付泽文, 李宁, 白澜. 高分辨率斜视聚束 SAR 回波仿真加速算法研究[J]. 计算机科学, 2022, 49(8): 178-183.

GUO Zheng-wei, FU Ze-wen, LI Ning, BAI Lan. Study on Acceleration Algorithm for Raw Data Simulation of High Resolution Squint Spotlight SAR[J]. Computer Science, 2022, 49(8): 178-183.

---

### 相似文章推荐 (请使用火狐或 IE 浏览器查看文章)

Similar articles recommended (Please use Firefox or IE to view the article)

#### [基于特征优化的 SAR 图像水华识别方法](#)

Algal Bloom Discrimination Method Using SAR Image Based on Feature Optimization Algorithm

计算机科学, 2021, 48(9): 194-199. <https://doi.org/10.11896/jsjcx.200800142>

#### [结合 MCycleGAN 与 RFCNN 实现太阳斑点图高分辨重建](#)

Combining MCycleGAN and RFCNN to Realize High Resolution Reconstruction of Solar Speckle Image

计算机科学, 2021, 48(6A): 38-42. <https://doi.org/10.11896/jsjcx.201000160>

#### [一种零分辨率 3D 网格模型的信息隐藏算法](#)

Zero-high-resolution Information Hiding Algorithm for 3D Mesh Model

计算机科学, 2020, 47(7): 328-334. <https://doi.org/10.11896/jsjcx.190800021>

#### [基于 Multi-Path RefineNet 的多特征高分辨率 SAR 图像道路提取算法](#)

Road Extraction Algorithm of Multi-feature High-resolution SAR Image Based on Multi-Path RefineNet

计算机科学, 2020, 47(3): 156-161. <https://doi.org/10.11896/jsjcx.190100124>

#### [高分辨率 SAR 图像道路提取综述](#)

Review of Road Extraction for High-resolution SAR Images

计算机科学, 2020, 47(1): 124-135. <https://doi.org/10.11896/jsjcx.190100033>

# 高分辨率斜视聚束 SAR 回波仿真加速算法研究

郭拯危 付泽文 李宁 白澜

河南大学计算机与信息工程学院 河南 开封 475004

河南大学河南省智能技术与应用工程技术研究中心 河南 开封 475004

河南大学河南省大数据分析与管理重点实验室 河南 开封 475004

(gzw@henu.edu.cn)

**摘要** 回波仿真是合成孔径雷达系统开发的前端工作,具有重要意义。针对高分辨率斜视聚束 SAR,一般采用时域回波仿真的方法,但是其仿真效率过慢。为了高效实现高分辨率斜视聚束 SAR 的回波仿真,提出了一种有效的加速算法。结合斜视聚束 SAR 时域回波模型及其信号特性,对回波仿真过程中存在的距离徙动进行补偿,以减少冗余计算量并节省内存空间;采用数据自适应分块的方法,在图形处理器中分别计算分块后的子数据,利用 GPU 强大的计算能力进行加速;将子数据块进行传输,在内存中拼接。该算法提高了时域回波仿真的计算效率,解决了数据量大、GPU 显存有限且显存与内存之间数据传输速度较慢的问题。点目标和面目标仿真的实验结果表明,该算法的加速比达到了 219.8,验证了所提方法的有效性。

**关键词:** 高分辨率;合成孔径雷达;时域回波仿真;距离徙动;图像处理

中图法分类号 TP702

## Study on Acceleration Algorithm for Raw Data Simulation of High Resolution Squint Spotlight SAR

GUO Zheng-wei, FU Ze-wen, LI Ning and BAI Lan

College of Computer and Information Engineering, Henan University, Kaifeng, Henan 475004, China

Henan Engineering Research Center of Intelligent Technology and Application, Henan University, Kaifeng, Henan 475004, China

Key Laboratory of Analysis and Processing on Big Data of Henan Province, Henan University, Kaifeng, Henan 475004, China

**Abstract** Raw data simulation is the front-end work of synthetic aperture radar(SAR) system development, which is of great significance. For high resolution squint spotlight SAR, time-domain raw data simulation is usually used, but its simulation efficiency is very low. In order to realize the raw data simulation of high resolution squint spotlight SAR efficiently, an effective acceleration algorithm is proposed. To reduce the redundant computation and save memory, this algorithm combines the time-domain raw data simulation model and its signal characteristics to compensate the range cell migration(RCM) in the raw data simulation process of squint spotlight SAR. An adaptive data partitioning algorithm is adopted to compute the partitioned data in graphic processing unit(GPU), and the powerful computing capabilities of GPU is used to improve efficiency. Then the sub data blocks are transmitted and spliced in memory. The proposed algorithm improves the computational efficiency of time-domain raw data simulation, and solves the problems of huge volume of raw data, limited GPU memory and data transmission between video memory and memory. Experimental results of point targets and distributed targets show that the speedup ratio of this algorithm reaches 219.8, which verifies the effectiveness of the proposed method.

**Keywords** High resolution, Synthetic aperture radar, Time-domain raw data simulation, Range cell migration, GPU

到稿日期:2021-06-04 返修日期:2021-09-06

基金项目:国家自然科学基金(61871175);河南省高等学校重点科研项目(19A420005,21A520004);河南省科技攻关计划项目(202102210175,212102210093,212102210101);自然资源部国土卫星遥感应用重点实验室经费资助项目(KLSMNR-202102)

This work was supported by the National Natural Science Foundation of China(61871175), College Key Research Project of Henan Province(19A420005,21A520004), Plan of Science and Technology of Henan Province(202102210175,212102210093,212102210101) and Key Laboratory of Land Satellite Remote Sensing Application, Ministry of Natural Resources of the People's Republic of China(KLSMNR-202102).

通信作者:李宁(hedalining@henu.edu.cn)

## 1 引言

合成孔径雷达(Synthetic Aperture Radar, SAR)是一种主动微波遥感技术,具有全天时、全天候的对地观测能力<sup>[1]</sup>。机载 SAR 系统在大斜视角下的工作能力是决定该系统具备灵活性和机动性的重要条件,高分辨率、大斜视角也是 SAR 技术不断追求的目标<sup>[2]</sup>。而回波仿真作为 SAR 系统开发的前端工作,已经成为一个热门方向,有助于验证 SAR 系统的性能,可以更好地确定系统参数<sup>[3]</sup>。因此,在大斜视角条件下的回波仿真意义重大。

现有的 SAR 回波仿真算法可分为两大类:频域回波仿真算法、时域回波仿真算法<sup>[4]</sup>。然而,距离向和方位向的耦合性、距离徙动过大等一系列因素造成的频谱混叠等特征使频域回波仿真算法无法适用于大斜视角条件下的回波仿真算法<sup>[5-6]</sup>。时域回波仿真算法可以在大斜视条件下精确计算出回波数据,但是其运算效率非常低,在应用于大数量下的面目标仿真时,运算速度过慢成为了一个亟需解决的问题。

目前,针对时域回波仿真运算量过大这一问题,国内外学者的解决思路大致可分为 3 个方向,第一种是基于多核中央处理器(Central Processing Unit, CPU)并行的方式,第二种是基于现场可编程门阵列(Field Programmable Gate Array, FPGA)的并行计算方式,第三种则是利用图形处理器(Graphic Processing Unit, GPU)进行高效运算的方式<sup>[7-9]</sup>。Chen 等于 2017 年提出了一种基于 GPU 加速的双基地 SAR 回波仿真方法<sup>[10]</sup>, Wijayasiri 等于 2018 年利用 GPU 并行进行 SAR 图像重建<sup>[11]</sup>。GPU 的可编程性、并行处理能力强大等特点,使其与传统算法相结合的方式已经成为目前最主流的加速策略之一。但是,上述 3 种方式都存在着一些弊端。例如:CPU 并行效果不强,性价比不高,运算能力较弱;FPGA 的成本高,资源昂贵,调试难度大;对于 GPU,在双精度浮点型计算能力较强的显卡中,主流的显存有限,容易出现显存溢出的问题。

本文对高分辨率大斜视角情况下的聚束 SAR 模式回波仿真进行研究,确定雷达系统的几何关系,建立回波信号模型。针对高分辨率聚束模式数据量较大的情况,提出了一种有效的高分辨率斜视聚束 SAR 回波仿真加速算法。首先对大斜视角产生的距离徙动进行补偿,消除其造成的冗余计算,然后根据 GPU 的显存大小,将数据进行自适应分块,在 GPU 中计算回波数据,完成后再传入内存进行数据拼接。在不损失目标质量的情况下,有效地提高了时域回波仿真的运算速度。

## 2 斜视聚束 SAR 回波信号模型

与传统正侧视条件下的聚束 SAR 回波仿真相比,大斜视角条件下存在较明显的距离徙动等问题,但正是这一系列问题的存在,使得时域回波仿真的效率有了进一步优化的空间<sup>[12]</sup>。

斜视聚束 SAR 模式的观测几何模型如图 1 所示。图 1 中, $H$  为机载 SAR 平台的高度,以场景中心点  $O$  建立坐标系, $X$  轴对应方位向, $Y$  轴对应距离向, $Z$  轴对应高度。 $R_{\text{start}}$ ,  $R_c$ ,  $R_{\text{end}}$  分别为成像开始时刻、中心时刻和结束时刻雷达到

场景中心点的距离; $r$  为零多普勒平面上雷达到场景中心的距离; $\theta$  为中心时刻雷达飞行方向和波束指向之间的夹角,即斜视角; $\alpha$  为零多普勒平面的下视角。

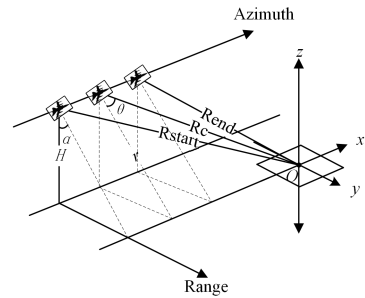


图 1 SAR 几何关系示意图

Fig. 1 Sketch chart of SAR geometric relation

根据图 1 所示的几何模型,雷达到场景中任意一点  $(x, y, z)$  的瞬时斜距  $R(t)$  为:

$$R(t) = \sqrt{(vt-x)^2 + (R_c-y)^2 - 2(R_c-y) \cdot (vt-x) \sin\theta} \quad (1)$$

其中, $v$  为平台的运动速度。

经过正交解调过程,去除载频后,得出  $t$  时刻点  $(x, y, z)$  的回波信号为:

$$S(t, \tau) = \text{rect}\left(\frac{\tau - 2R(t)/c}{T_r}\right) \cdot \exp\left(-j \frac{4\pi R(t)}{\lambda}\right) \cdot \exp\left(j\pi K_r \left(\tau - \frac{2R(t)}{c}\right)^2\right) \quad (2)$$

其中, $\lambda$  为波长, $K_r$  为距离向调频率, $T_r$  为发射信号脉宽, $c$  为电磁波传播速率。若有多个点目标,则将多点目标的  $S(t, \tau)$  累加即可。

将距离  $R(t)$  在中心时刻  $t_0$  处进行泰勒级数展开,如式(3)所示:

$$R(t) = R(t_0) - v \cdot \sin\theta \cdot (t-t_0) + \frac{1}{2} \frac{v^2 \cdot \cos^2\theta}{R(t_0)} (t-t_0)^2 + \frac{1}{2} \frac{v^3 \cdot \sin\theta \cos^2\theta}{R^2(t_0)} (t-t_0)^3 + \dots \quad (3)$$

式(3)由一阶项距离走动、二阶项距离曲率和高阶项组成,且距离走动远大于距离弯曲,距离弯曲远大于高阶项。

## 3 基于 GPU 加速的回波仿真实现

时域回波仿真是针对场景中每个目标点进行回波计算的过程,且计算过程相互独立、互不影响,因此可以通过并行的方式利用硬件进行加速。此外,还可以对回波仿真算法本身进行优化。本节基于大斜视角情况下的回波特性,结合算法优化以及硬件加速,提出了一种加速策略,使时域回波仿真算法在大数据量、大斜视角情况下的运算效率得到了有效提升。

### 3.1 GPU 与 CUDA 编程模型

GPU 的数据处理能力强大,与 CPU 相比,GPU 将更多的资源应用于核心部分的数据处理,而不是用于数据缓存等其他方面。因此,在成本相同的情况下,GPU 的并行计算能力要远远强于 CPU,而 CPU 更适合逻辑性更强的串行计算。NVIDIA 公司于 2007 年推出的统一计算架构(Compute Unified Device Architecture, CUDA),使 GPU 得到了充分的

利用,并且使得 GPU 的可编程性更加强大<sup>[13-14]</sup>。时域回波仿真算法中,每个点目标的回波处理是相互独立的,适合利用 GPU 多线程的特点进行加速,提高运算效率。

CUDA 的编程模型相对简单,其核心在于 kernel 函数。该函数被所有线程同时调用,其结构分为线程网格(Grid)、线程块(Block)和线程(Thread)。简单来说,它们之间是自上而下的控制关系:Grid 控制多个 Block,Block 又控制了多个 Thread。

采用 CUDA 编程模型,有效利用 GPU 的强大性能,可以使时域回波仿真算法的效率大幅提升。但是,由大斜视角下的回波特性可知,距离徙动将造成数据量和计算量的急剧增大,因此需要对其进行自适应优化。

### 3.2 计算量的优化

大斜视角下的回波距离徙动特性造成了大量的冗余计算。消除距离徙动,一方面可以减轻时域回波仿真算法的计算量,另一方面还可以节省运算时回波数据占用的显存空间。具体优化思路如下:首先在 CPU 中计算出回波的有效区域,再根据距离徙动计算出每一方位时刻距离向数据的偏移量,将其传入显存运算时可等效为无距离徙动影响的回波仿真计算。

设  $t_r$  为  $t$  时刻的距离时间轴,其计算式如下:

$$t_r = \frac{i - N_r/2}{F_r} + \frac{2 \cdot (R(t) - R_c)}{c}, t_r \in \left[-\frac{T_r}{2}, \frac{T_r}{2}\right] \quad (4)$$

其中,  $F_r$  为距离向采样率,  $N_r$  为距离向采样点数,  $i$  为距离向采样点的序号。根据式(4)可知,  $t_r$  的绝对值应当小于半个脉宽。可求出每一方位时刻  $t$  有效范围内的两个边界值  $i_{far}$ ,  $i_{near}$ 。

$$\begin{cases} i_{near} = \frac{N_r}{2} + \frac{2 \cdot F_r \cdot (R_{near}(t) - R_c)}{c} \\ i_{far} = \frac{N_r}{2} + \frac{2 \cdot F_r \cdot (R_{far}(t) - R_c)}{c} + F_r \cdot T_r \end{cases} \quad (5)$$

其中,  $R_{near}$  为  $t$  时刻雷达到场景的最近距离,  $R_{far}$  为  $t$  时刻雷达到场景的最远距离。

为了使时域回波仿真在 GPU 中的运算效率更高,对距离徙动进行补偿。由于距离徙动主要受一阶项距离走动和二阶项距离曲率的影响,因此根据式(3),以  $N_r/2$  为中心,对一次项的距离走动和二次项的距离曲率进行补偿。图 2 为距离徙动补偿的示意图。经过该补偿,既减少了计算量,也节省了 GPU 的存储空间。

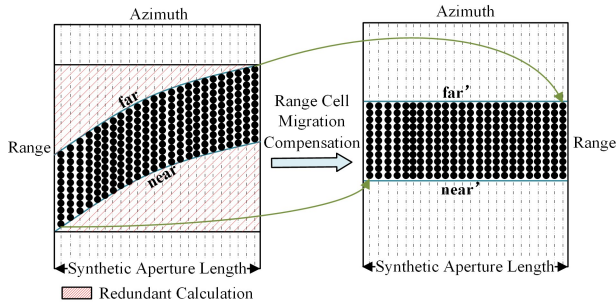


图 2 计算量优化示意图

Fig. 2 Sketch chart of calculation optimization

设  $N_a$  为方位向采样点数,根据每一方位时刻的时间,可以计算出需要补偿的距离  $Range$ ,即  $t$  时刻数据偏移的距离。

$$Range = v \cdot (t - t_0) \cdot \sin\theta - \frac{1}{2} \frac{v^2 \cos^2\theta}{R} (t - t_0)^2 \quad (6)$$

由式(6)可知,计算出  $t$  时刻像素偏移的大小  $P_{ix}$  为:

$$P_{ix} = Range / \Delta R \quad (7)$$

其中,  $\Delta R$  为像素间隔。将数据的近边界  $i_{near}$  和远边界  $i_{far}$  移位  $P_{ix}$  之后,即可求得移位后的边界  $i'_{near}$ ,  $i'_{far}$ 。

$$\begin{cases} i'_{near} = P_{ix} + i_{near} \\ i'_{far} = P_{ix} + i_{far} \end{cases} \quad (8)$$

如图 2 所示,阴影部分为冗余计算,经过距离徙动补偿将其消除后,计算效率将显著提升,并且对显存的占用也将减少。但是,当数据量过大时,仅仅消除冗余计算依然无法满足实际仿真的需求,GPU 显存可能会出现溢出等问题。

### 3.3 数据自适应分块

为了解决 GPU 在数据量过大情况下可能出现的问题,本文提出了适用于 GPU 的自适应分块算法,一方面能够使 GPU 的计算资源得到充分利用,另一方面防止了显存不够、显存溢出等情况的发生。

本文算法的思想如下:首先获取系统中可用的 GPU 数量以及所有可用的显存容量,然后根据显存的总量,计算一次所能存放数据块的大小,即得到需要分块的个数,分配内存、显存,依次进行回波模拟的计算,每完成一部分计算,就将该块数据从显存调入内存并释放显存。图 3 为数据自适应分块示意图。

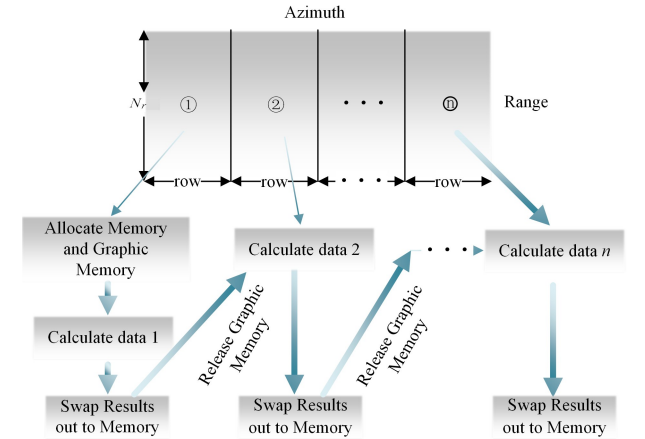


图 3 数据自适应分块示意图

Fig. 3 Sketch chart of adaptive data partition

设  $Gpumem$  为该系统可用的显存容量,根据显存大小可计算出单次能处理的数据块大小,设  $col$  为单次最多能处理的列数。

$$col = Gpumem / (N_r \cdot size) \quad (9)$$

其中,  $size$  为单个数据的大小,  $N_r \cdot size$  即每一列的数据大小,根据式(9)以及方位向采样点  $N_a$  即可计算出需要分块的数量  $N$ 。

$$N = \lceil N_a / col \rceil \quad (10)$$

其中,  $\lceil \cdot \rceil$  代表向上取整。分块完毕后,根据分块结果进行显存分配,然后依次进行 GPU 中时域回波仿真算法的计算。

在数据自适应分块算法中,由于回波仿真的各个子数据块互不影响,每一方位时刻距离向数据独立运算,目标点的回波数据计算完毕后全部累加即得最终所有目标的回波数据。

采用该分块方式,每块数据的大小都适合 GPU 的显存,使每一次的运算对 GPU 的利用率较为充分,避开了显存溢出问题,并且减少了数据在内存总线和显存总线的交换次数。将完整的数据分块之后,根据参数分配显存,依次在适用于任意显存大小的 GPU 中进行回波数据的运算。

此外,通过异步执行进一步提升数据传输过程中的效率。处于同一个流内的计算与数据拷贝依次进行,且一个流内的计算和另一个流的数据传输可同时进行,使 GPU 中的执行单元和存储器控制单元同时工作,以提高资源利用率并进一步提升算法的效率<sup>[15]</sup>。在实际的算法实现中,每个线程运算完成一个方位时刻的数据后即传输该数据至内存中,传输完毕后继续进行下一方位时刻数据的计算。每块回波数据的传输与计算过程无需等待,即可继续进行下一数据块的传输与计算。

### 3.4 数据恢复

当每块数据运算完毕后,需要将数据从显存调入内存中。由于在计算量优化的过程中补偿了距离徙动造成的影响,因此需要在 CPU 中将其还原,把数据再次补偿至存在斜视角影响的位置,并且将所有分块后的数据还原为一块完整的数据。

在内存中,每从显存调入一部分数据,就将其按照方位时间轴对应的时刻存放。存放完毕后,将数据补零,生成  $N_a \times N_r$  大小的数据块。此时,该数据块为距离徙动补偿后的回波数据。

根据式(7)计算出的补偿值  $P_{ix}$ ,将该矩阵的每一方位时刻中距离向有效区域还原到距离徙动补偿前的位置,即 3.2 节中距离徙动补偿的逆过程。这样就还原了在大斜视角下的回波仿真数据,即最终的回波数据。

最终的加速算法流程图如图 4 所示,其具体过程如算法 1 所示。

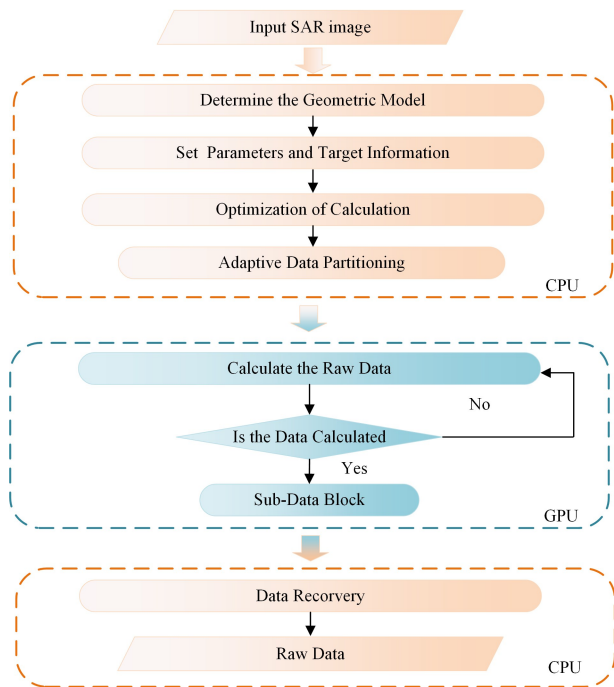


图 4 加速算法的流程图

Fig. 4 Flow chart of acceleration algorithm

### 算法 1 斜视聚束 SAR 回波仿真加速算法

输入: SAR 图像数据文件

输出: 回波数据文件 / \* 回波仿真加速方案 \* /

1. 初始化雷达系统参数
2. 计算上下边界以及偏移的距离
3. 获取 GPU 显存, 计算子数据块个数  $N$
4. For  $n \leftarrow 1$  to  $N$  do
5. For  $k \leftarrow 1$  to  $x$  do
6. /\*  $x$  为目标点个数 \*/;
7. For  $j \leftarrow 1$  to  $col$  do
8. /\*  $col$  为子数据块列数 \*/;
9. 计算每一方位时刻瞬时斜距及相位
10. 叠加计算目标点回波数据
11. End
12. End
13. 传入内存
14. End
15. 数据恢复

## 4 实验与分析

本实验采用的仿真平台参数如表 1 所列。

表 1 仿真平台及硬件信息

Table 1 Simulation platform and hardware information

Platform	Parameters
CPU	Inter Xeon Gold 6126 Number of Core: 48 Clock Speed: 2.6 GHz
Memory	128 GB
GPU	NVIDIA TITAN V
Graphics memory	12 GB

为了验证所提方法的有效性,在大斜视角聚束模式下进行时域回波仿真。该实验所采用的参数如表 2 所列。

表 2 仿真参数

Table 2 Simulation parameters

Parameter	Value
Signal pulse width/ $\mu$ s	40
Pulse repetition frequency/Hz	1 500
Signal bandwidth/MHz	500
Height/km	10
Velocity/m/s	200
Squint angle/ $^{\circ}$	60
Slant range of image center/km	60
Coherent processing interval/s	10.1

首先在场景中布置 5 个点,计算各项参数后确定最终回波矩阵的大小为  $49540 \times 46705$ ,然后对计算量进行优化,并根据优化后的各项参数以及 GPU 显存分为 5 块,每块大小为  $49540 \times 9341$ ,在 GPU 中分别计算 5 次,待数据从显存全部调入内存后进行拼接,将拼接完毕后的回波数据恢复至未进行距离徙动补偿前的位置。图 5(a)为距离徙动补偿前的回波数据幅度图,图 5(b)为经过距离徙动补偿后的回波数据幅度图。

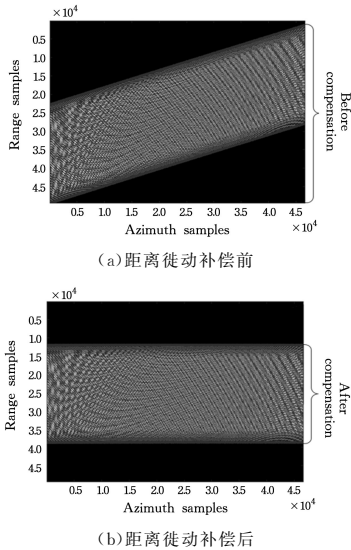


图 5 回波数据幅度图

Fig. 5 Amplitude chart of raw data

对比图 5(a)与图 5(b)可明显看出,经过距离徙动补偿后,有效数据的距离向单元明显减少,因此计算量和占用的显存也随之减少。

将时域回波仿真算法在传统串行方式下的运算时间、使用 MPI(Message-Passing Interface)消息传递并行编程模型的运算时间、使用传统 CUDA 编程模型的运算时间和本文算法加速后的运算时间进行对比。表 3 列出了使用不同算法生成矩阵大小为  $49\ 540 \times 46\ 705$  的回波数据所需要的时间对比结果。

根据表 3 可看出,与在 CPU 中使用传统串行计算方式<sup>[16]</sup>的运算相比,利用传统 MPI 和传统 CUDA 模型的运算效率有明显的提升,当数据量较小时,在 CUDA 模型中由于显存内存间的交互耗时较长,相比传统 MPI 模型运算效率较低。当数据量不断增大时,CUDA 模型的加速效果优于 MPI 模型,但是这两种模型的加速效果均有限<sup>[17-18]</sup>。采用本文

提出的加速算法对时域回波仿真算法进行加速,当点目标个数为 500 时,加速比可达 219.8,优于传统的加速算法。

表 3 仿真效率对比

Table 3 Simulation efficiency comparison

(单位:s)

Target number	Serial computing	Parallel computing		Proposed method
		CUDA	MPI	
5	1 092	49.5	45.9	6.4
25	5 286	208.6	206.4	27.0
100	20 532	739.5	787.6	97.8
500	98 572	2 816.4	3 769.5	448.4

为了验证回波仿真结果的正确性,使用反投影算法对使用回波仿真加速算法前后的 5 个点目标回波数据进行成像对比,如图 6(a)、图 6(b)所示。表 4 列出了使用回波仿真加速算法前后 5 个点目标的峰值旁瓣(PSLR)和积分旁瓣(ISLR)。表 4 所列的结果验证了加速前后的回波仿真的精度并无明显变化,且不损失精度。

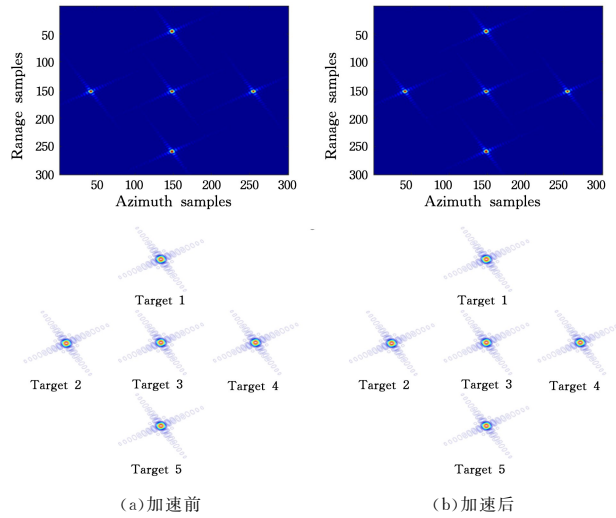


图 6 点目标成像结果对比

Fig. 6 Comparison of point targets imaging results

表 4 点目标仿真评估结果对比

Table 4 Comparison of evaluation results for simulated point targets

(单位: dB)

Target	Before acceleration		After acceleration	
	Azimuth	Range	Azimuth	Range
	PSLR/ISLR	PSLR/ISLR	PSLR/ISLR	PSLR/ISLR
1	-13.72/-10.08	-13.26/-10.12	-13.26/-10.06	-13.25/-10.16
2	-13.26/-10.09	-13.25/-10.13	-13.28/-10.05	-13.29/-10.14
3	-13.24/-10.08	13.26/-10.15	-13.28/-10.09	-13.23/-10.09
4	-13.26/-10.06	-13.26/-10.15	-13.27/-10.06	-13.27/-10.13
5	-13.27/-10.07	-13.27/-10.13	-13.25/-10.08	-13.28/-10.14

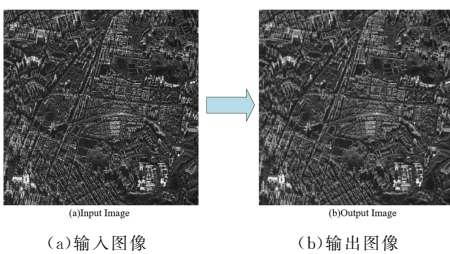


图 7 面目标成像结果

Fig. 7 Imaging result of distributed targets

采用方位向分辨率为 1 m、距离向分辨率为 1 m、像素个数为  $4\ 096 \times 4\ 096$  的 SAR 图像作为输入图像,如图 7(a)所示。将方位向、距离向分辨率均设为 1.5 m,脉冲重复频率设为 1 100 Hz,对该图像进行回波仿真,仿真后的成像结果如图 7(b)所示。可以看出,该算法在不损失图像精度的前提下显著提高了运算速度。

**结束语** 本文通过对高分辨率斜视聚束 SAR 模式下的几何模型、回波特性进行研究,针对该情况下的时域回波仿真设计了一种有效的加速算法。利用 CPU 与 GPU 各自的

优势,结合大斜视角下回波的特点,提出了一种优化计算量的策略,该策略既减少了计算量,又节省了存储空间;针对数据量过大而显存有限的问题,提出了一种自适应的分块算法。最后分别对点目标和面目标进行回波仿真,验证了所提加速算法的有效性。但是,本文仅对单 GPU 下的加速进行仿真实验,后续将继续在多 GPU 或集群 GPU 的条件下进行进一步优化<sup>[19-21]</sup>。

## 参 考 文 献

- [1] LI N, NIU S L. High-precision water segmentation from synthetic aperture radar images based on local super-resolution restoration technology[J]. *Journal of Radars*, 2020, 9(1): 174-184.
- [2] ZHOU Y Y, CHENG J H, LIU T, et al. Review of Road Extraction for High-resolution SAR Images[J]. *Computer Science*, 2020, 47(1): 124-135.
- [3] WANG B Q, QI X Y, ZHOU S B. An Improved Echo Simulation Method of GEO SAR[J]. *Radar Science and Technology*, 2019, 17(2): 198-207.
- [4] ZENG L T, YANG C H, LI Q, et al. Validation of Synthetic Aperture Radar (SAR) Image Algorithm Based on Simulation [J]. *Computer Science*, 2019, 46(S1): 287-290.
- [5] FAN W, ZHANG M, LI J, et al. Modified Range-Doppler Algorithm for High Squint SAR Echo Processing[J]. *IEEE Geoscience and Remote Sensing Letters (S1545-598X)*, 2019, 16(3): 422-426.
- [6] LI G, MA Y H, HOU J Q, et al. Sub-aperture Keystone Transform Based Echo Simulation Method for High-squint SAR with a Curve Trajectory[J]. *Journal of Electronics & Information Technology*, 2020, 42(9): 2261-2268.
- [7] ZUO X Y, ZHANG Z, SU Y H, et al. Extraction Algorithm of NDVI Based on GPU Multi-stream Parallel Model[J]. *Computer Science*, 2020, 47(4): 25-29.
- [8] MENG D D, HU Y X, SHI T, et al. Airborne SAR Real-time Imaging Algorithm Design and Implementation with CUDA on NVIDIA GPU[J]. *Journal of Radars*, 2013, 2(4): 481-491.
- [9] ZHOU F, LI J. Research on Multi-core DSP Based Parallel Access for SAR Data [J]. *Modern Electronics Technique*, 2018, 41(15): 26-30.
- [10] CHEN Q, ZHU M B, ZOU X H, et al. Fast Realization of Bistatic SAR Echo Simulation Based on GPU Acceleration[J]. *Computer Simulation*, 2017, 34(4): 1-4.
- [11] WIJAYASIRI A, BANERJEE T, RANKA S, et al. Dynamic Data-Driven SAR Image Reconstruction Using Multiple GPUs[J]. *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing (S1939-1404)*, 2018, 11(11): 4326-4338.
- [12] LIU Y, XING M D, SUN G C, et al. Echo Model Analyses and Imaging Algorithm for High-Resolution SAR on High-Speed Platform[J]. *IEEE Transactions on Geoscience and Remote Sensing (S0196-2892)*, 2012, 50(3): 933-950.
- [13] CHENG D, WANG W H. Application of OpenMP in SAR Image Processing[J]. *Computer Science*, 2017, 44(S1): 161-163, 187.
- [14] HU C, ZHANG F, LI G J, et al. Computation Reduction Oriented Circular Scanning SAR Raw Data Simulation on Multi-GPUs [J]. *Journal of Radars*, 2016, 5(4): 434-443.
- [15] LIU Y B, ZHOU Y C, ZHOU Y S, et al. Accelerating SAR Image Registration Using Swarm-Intelligent GPU Parallelization[J]. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2020, 13(1): 5694-5703.
- [16] YANG W D, WANG H T, ZHANG Y F, et al. Survey of Heterogeneous Hybrid Parallel Computing [J]. *Computer Science*, 2020, 47(5): 5-16, 3.
- [17] ZAMBRE R, SAHASRABUDHE D, ZHOU H, et al. Logically Parallel Communication for Fast MPI + Threads Applications [J]. *IEEE Transactions on Parallel and Distributed Systems*, 2021, 32(12): 3038-3052.
- [18] ZHANG Y R, CHEN L, AN X Z, et al. Study on Performance Optimization of Reduction Algorithm Targeting GPU Computing Platform[J]. *Computer Science*, 2019, 46(2): 306-314.
- [19] LIU S, ZOU B, ZHANG L M, et al. A Multi-GPU Accelerated Parallel Domain Decomposition One-Step Leapfrog ADI-FDTD [J]. *IEEE Antennas and Wireless Propagation Letters*, 2020, 19(5): 816-820.
- [20] KIM M, LIU L, CHOI W, et al. Multi-GPU Efficient Indexing for Maximizing Parallelism of High Dimensional Range Query Services[J/OL]. *IEEE Transactions on Services Computing*, 2021. <https://ieeexplore.ieee.org/document/9430517>.
- [21] FARIBORZ M, XIAO X, FOTOUHI P, et al. Silicon Photonic Flex-LIONS for Reconfigurable Multi-GPU Systems[J]. *Journal of Lightwave Technology*, 2021, 39(4): 1212-1220.



**GUO Zheng-wei**, born in 1963, bachelor, professor, master supervisor, is a member of China Computer Federation. Her main research interests include SAR image processing techniques, and SAR image application of ecological environment.



**LI Ning**, born in 1987, Ph.D, professor, Ph.D supervisor, is a member of China Computer Federation. His main research interests include multi-mode SAR imaging and SAR application.