

# 显微CT血管系统三维结构的骨架细化算法 并行化设计实现

谭海<sup>1,2</sup> 王大东<sup>3</sup> 薛艳玲<sup>1</sup> 王玉丹<sup>1</sup> 杨一鸣<sup>1,2</sup> 肖体乔<sup>1,2</sup>

<sup>1</sup>中国科学院上海应用物理研究所, 上海 201800

<sup>2</sup>中国科学院大学, 北京 100049

<sup>3</sup>澳大利亚联邦科学与工业研究组织, 悉尼 NSW 1670

**摘要** 提取骨架是计算机断层扫描(CT)三维(3D)血管图像定量分析中的关键步骤,通常耗费数小时,直接制约了图像分析的定量研究。分析串行骨架细化算法各步骤中包含的可并行化操作,对其进行并行化设计,提出的算法通过OpenMP多线程技术实现,并采用不同大小的三维CT血管图像进行分析和测试。根据测试结果,改进后的算法获取到的骨架准确可靠,对于1.95 GB大小的三维血管图像,使用16个线程进行并行运算时,可将运算时间由176 min缩短到13 min,时间消耗上降低了一个数量级。因此,提出的方法可实现大型血管骨架的准确、高效提取,解决了大型三维图像分析问题中运算效率低这一瓶颈问题。

**关键词** 医用光学; 三维骨架; 细化; 并行处理; OpenMP; 加速比

中图分类号 TP751.1

文献标识码 A

doi: 10.3788/AOS201535.1117003

## Parallelization of 3D Thinning Algorithm for Extracting Skeleton of Micro-CT Vasculature

Tan Hai<sup>1,2</sup> Wang Dadong<sup>3</sup> Xue Yanling<sup>1</sup> Wang Yudan<sup>1</sup>

Yang Yiming<sup>1,2</sup> Xiao Tiqiao<sup>1,2</sup>

<sup>1</sup>Shanghai Institute of Applied Physics, Chinese Academy of Sciences, Shanghai 201800, China

<sup>2</sup>University of Chinese Academy of Sciences, Beijing 100049, China

<sup>3</sup>Commonwealth Scientific and Industrial Research Organization, Sydney NSW 1670, Australia

**Abstract** The extraction of skeleton is often a crucial step in quantitative assessment of three-dimensional (3D) vasculature computed tomography (CT) image. The process always consumes several hours to get the skeleton, which confines the efficiency of quantitative analysis. By means of OpenMP, a parallel designing method based on sequential thinning is proposed to improve the computational time of the skeletonization. The implemented method is utilized for different sizes of real 3D vascular CT images in order to evaluate its performance and efficiency. The testing results show that the proposed method, which is implemented in 16-thread parallel, does not only extract precise skeleton, but also conspicuously reduces the processing time to an acceptable scale. The corresponding computational time is reduced from 176 min to 13 min. Therefore, the time efficiency of quantitative assessment is no longer an obstruction for the analysis of the large scale 3D vasculature CT images.

**Key words** medical optics; 3D skeleton; thinning; parallel processing; OpenMP; speedup

**OCIS codes** 170.7440; 100.2960; 100.6890; 100.6950; 200.4960

收稿日期: 2015-03-18; 收到修改稿日期: 2015-06-18

基金项目: CAS-CSIRO 国际合作项目(GJHZ1303)、国家自然科学基金(81430087, 11405260, 11475248)、国家973计划(2010CB834301)、国家自然科学基金联合重点项目(U1232205)

作者简介: 谭海(1985—),男,博士研究生,主要从事三维图像处理以及图像分析等方面的研究。E-mail: tanhai@sinap.ac.cn

导师简介: 肖体乔(1966—),男,研究员,博士生导师,主要从事成像光学和X射线光学等方面的研究。

E-mail: xiaotiqiao@sinap.ac.cn (通信联系人)

## 1 引 言

基于同步辐射的显微计算机断层扫描(CT)成像技术在生物医学以及材料学等学科中有着广泛的应用<sup>[1-4]</sup>。同时,采集到的绝大多数三维图像都有着迫切的图像定量分析需求。其中,在生物医学领域<sup>[5-6]</sup>,显微CT血管类三维(3D)图像定量分析中,三维骨架的提取是其中至关重要的一个步骤。通过骨架这种简化的图形表征,更有利于提取出三维图形中的结构以及长度等信息。例如,血管长度、血管弯曲度以及分叉点数目等定量信息都能通过分析血管的骨架提取出来。

近年来,随着成像技术的改进及探测器分辨率的提高<sup>[7-10]</sup>,获取高精度图像的同时也导致了图像容量的显著提升,从而导致了骨架提取这一关键步骤面临着严峻的运算效率问题。为了解决这一问题,本文对三维骨架细化算法的整个流程进行分析,提取出可进行并行运算的部分加以改进,使算法支持并行运算从而缩短计算时间,算法并行化通过OpenMP多线程编程技术实现;随后使用实际图像测试分析该算法在并行化后的可靠性和效率,以解决困扰血管定量分析的运算效率问题。

## 2 三维骨架细化算法及其基于OpenMP实现并行运算和优化加速

### 2.1 三维骨架细化算法

#### 2.1.1 细化算法介绍

传统的骨架提取算法主要分为4类<sup>[11]</sup>:拓扑细化算法<sup>[12]</sup>,基于距离变化的算法<sup>[13]</sup>,基于Voronoi图的算法<sup>[14]</sup>以及广义势场的方法<sup>[15]</sup>。基于距离变化的算法首先赋予图形中每个像素点一个基于边界的距离值,随后提取距离值的局部最大值实现图形骨架点的提取,该类算法的主要缺点在于不能保证骨架的连通性和单像素性;Voronoi图是计算机中的一个重要概念,物体边界点集的Voronoi多边形集合构成了整个物体的Voronoi图,而骨架点的集合则是Voronoi图的子集,求解靠近物体中心的Voronoi边的集合可得到物体的骨架点集。基于Voronoi图的方法存在其局限性,它只适合计算简单多面体,对于复杂的图形很难得到准确的骨架结果,而且其得到的骨架也不能保证是一维单像素宽度的,同时计算效率也十分低下;广义势场的方法假定物体的边缘上聚集了均匀分布的同种电荷,依照电场模型计算出内部每一个像素点的场强,场强矢量为零的点即为图形的骨架点,该类算法提取三维骨架的最大缺陷在于计算复杂度非常之高。相比于其他3类骨架提取算法,拓扑细化算法能够很好地克服上述3类算法的缺点,具有实现简单,准确性高(拓扑结构不变性,骨架中心性,单像素宽度和保持连通性等)以及容易实现的特点,更有利于提取出用于定量分析的血管骨架。

细化算法是应用范围最为广泛的图形骨架提取算法,其基本过程是从二值图像中物体的外表面边界向内逐层均匀地删除边界点,直到剩下最里层的不能再剥掉的部分(保持连通性的点)从而得到了图形的骨架。细化算法又可分为串行细化算法<sup>[12,16-17]</sup>和完全并行细化算法<sup>[18-20]</sup>。串行细化算法容易实现,提取的骨架准确率高,但对于处理大型的三维图像数据,其计算的时间效率较低。完全并行细化算法能够十分显著地提升运算效率,但提取的骨架准确率不及串行细化算法。例如,对各方向形状分布不均匀的物体同时由外向内删除边界点,删除的过程不能保证在各方向上都是均匀的,会造成细化得到的骨架在部分区域并不处于物体内部近似中心线的位置,同时也可能产生某些枝节的过度细化或者改变原图形拓扑结构等现象。Wang等<sup>[21]</sup>在文献中指出了完全并行细化算法<sup>[18]</sup>中存在的改变原图形连通性结构的问题。在三维血管图像定量分析中,提取骨架需要在保证获取准确骨架的同时兼顾运算的时间效率,因此本文的工作是在串行细化算法<sup>[12]</sup>的基础上加以改进使其部分支持并行运算,在保证精度的前提下提高算法的运算效率。

#### 2.1.2 三维串行细化算法

串行细化算法<sup>[12]</sup>最核心的要素是保持拓扑不变性,即细化得到的骨架和原始图形的拓扑结构要完全一致。在数字拓扑学<sup>[22]</sup>中,一个三维多面体图形的欧拉数用(1)式表示:

$$\chi(S) = O(S) - H(S) + C(S), \quad (1)$$

式中 $O(S)$ ,  $H(S)$ 和 $C(S)$ 分别为多面体图形连通成分、环洞和孔洞的数目。环洞可以认为是一个中间空心的环状结构,与背景相连,而孔洞则是图形内部的空心结构,不与背景相连。欧拉数相等是两个图形具有相同拓扑结构的必要不充分条件,若两个图形具有相同的欧拉数,只有在其 $O(S)$ ,  $H(S)$ 和 $C(S)$ 分别相等的条件下,

才能认为这两个图形是拓扑不变的。因此,细化提取骨架保持拓扑结构不变可以表述为:提取骨架是一个仅删除点的操作;删除每一个点的前后,欧拉数不变;图形中的连通成分,环洞和孔洞三者数目始终保持不变<sup>[23]</sup>。

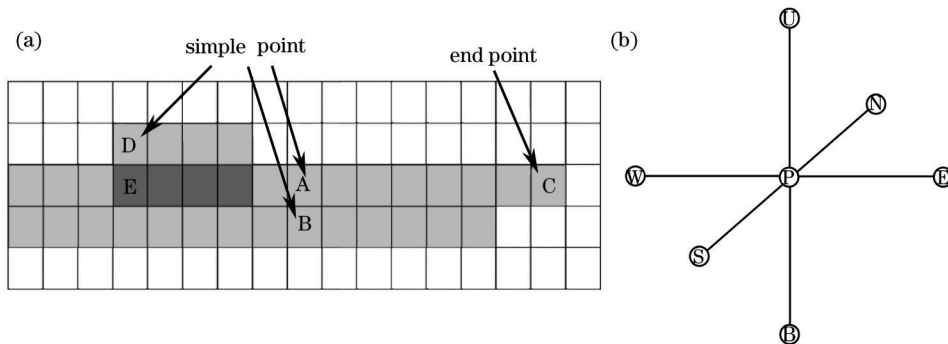


图1 细化算法中涉及到的各类点。(a)简单点和端节点;(b)基于6个方向的边界点

Fig.1 Different types of points in thinning algorithm. (a) Simple points and end points; (b) border points based on 6 directions  
在三维空间中,任意点的6邻域点和26邻域点分别定义为

$$N_6(m) = \{n(r,s,t) \in Z^3 | |i-r| + |j-s| + |k-t| = 1\}, \tag{2}$$

$$N_{26}(m) = \{n(r,s,t) \in Z^3 | \max(|i-r|, |j-s|, |k-t|) = 1\}. \tag{3}$$

在三维细化算法中,在图形边界上且删除后不会改变图像拓扑结构的点称为简单点,简单点删除前后,在其26邻域内欧拉数不变,各成分数目保持不变,具体定义在文献[12]中给出。如图1(a)所示,单独删除任何一个灰色的简单点都不影响图形的拓扑结构。另一类重要的点叫做端节点,如图1(a)中的点C。删除端节点不会改变图形的拓扑结构,但端节点是保持图形形状和长度信息的关键点,在细化过程中这类点是不可删除的。所以细化的过程实际上就是简单点和端节点的判断与删除的过程。为了保证由外向内在各方向上均匀地删除简单点, Lee等<sup>[12]</sup>将边界点基于6个方向分为6类,分别为北边界点 $P_N$ 、南边界点 $P_S$ 、东边界点 $P_E$ 、西边界点 $P_W$ 、上边界点 $P_U$ 和下边界点 $P_D$ 。如图1(b)所示,当P的6邻域点N的灰度值为0时,P为北边界点,以此类推。由于有的边界点的6邻域中存在多个灰度为零的点,所以同一个边界点有时会同时为多种类型的边界点。当图形外表面上的一个点被判断为简单点后并不能立即删除,否则会立即产生新的边界点[如图1(a)中点D立即删除后E立即变成边界点],从而产生过度细化的问题<sup>[12]</sup>。所以首先需要将外表面上的一类边界点中的简单点完全找出放入删除队列,再逐个顺序删除,每删除一个简单点时需要再判断一次是否改变图形的拓扑结构,因为有可能前一个删除的简单点使后删除的点变为非简单点。例如图1(a)中删除点A,使得点B已不是简单点,再删除点B必然会改变图形的拓扑结构。顺序删除6种边界点中的简单点作为一次循环迭代。如此反复迭代,从边界上一层层地搜索和删除简单点直到没有可删除的点,最终提取出骨架。整个串行细化算法的流程图如图2所示。

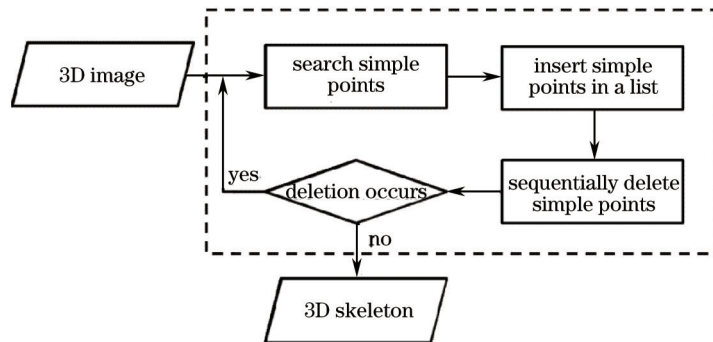


图2 串行细化算法流程图

Fig.2 Flow chart of 3D sequential thinning algorithm

## 2.2 细化算法并行运算的分析和设计

### 2.2.1 并行运算分析

在串行细化算法中,点的删除是一个顺序串行执行的过程,不具备并行化设计的条件。由于每一个简

单点的搜索都是相互独立的,故在边界上对简单点的搜索可以实现并行化设计。Lee等<sup>[12]</sup>将6类边界点 $P_N$ 、 $P_S$ 、 $P_E$ 、 $P_W$ 、 $P_U$ 和 $P_B$ 中搜索简单点的步骤并行化进行,使得一次迭代就能将边界上的所有简单点搜索出来。该方法的缺陷在于最多只支持6线程并行运算,而且实际图像中各类边界点有可能存在很大差别,如果某类边界点很多,处理该类点的线程会执行较长时间,并不能充分发挥出并行化运算的效率。而且对于某些边界点来说,可能被归为几类边界点,使得删除队列中出现部分重复的简单点,加大了后续串行删除点的运算开销,显然这样的并行化设计并不能明显改善串行细化算法的运算时间效率。

### 2.2.2 并行运算设计

基于串行细化算法的并行化改进设计针对的是简单点搜索的步骤。简单点的搜索是整个细化算法中反复进行的操作,其判断过程相互独立,所以简单点的搜索可实现并行运算。改进的算法同样将边界点分为6类 $P_N$ 、 $P_S$ 、 $P_E$ 、 $P_W$ 、 $P_U$ 和 $P_B$ ,依次遍历6次图像,每一次遍历都将一类边界点中的简单点找出并删除,遍历完6次完成一次图形边界细化的迭代,依此反复迭代直至没有简单点可删除。并行化出现在每一类边界点搜索简单点的步骤中,根据可用的线程数 $N$ ,将整个图像分成 $N$ 块区域搜索,每块区域在内存中都是连续存储的区域,这样使得简单点的搜索过程在 $N$ 个区域中互不干扰地并行化执行。为了保证并行搜索出的简单点与串行运算得到的简单点具有相同的删除顺序,在并行搜索结束后按划分区域的顺序将每一个区域搜索到的简单点删除列表合并到一起,再执行顺序删除点的操作。图3显示,用并行设计后的三维细化方法提取的血管骨架[图3(c)]与原串行细化算法提取的骨架[图3(b)]完全一致[图3(b)、(c)两图相减得到一幅全零图像]。

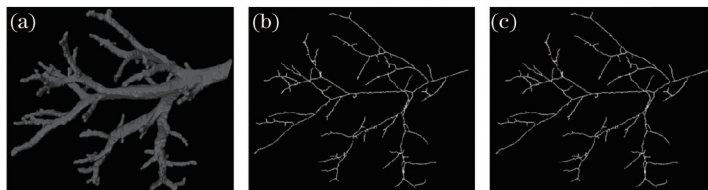


图3 三维血管及骨架。(a)二进制血管图像;(b)串行细化提取的骨架;(c)并行设计提取的骨架

Fig.3 Three-dimensional vasculature and its skeleton. (a) Binary vasculature image; (b) skeleton extracted by sequential thinning; (c) skeleton extracted by proposed parallel thinning

### 2.3 基于OpenMP实现并行运算及优化加速

OpenMP是一种针对共享内存的CPU多线程编程技术(<http://openmp.org/>),主要通过一套简单的编译器指令,较少的动态链接库函数和环境变量来实现程序的CPU多线程并行运算。OpenMP具有以下优点:1)简单易行,只需在程序中添加少量的编译指示性语句,不需要对代码进行较大改动;2)支持多种语言,例如Fortran语言和C/C++语言;3)支持多平台实现(类UNIX系统和Windows系统)。OpenMP标准由国际知名的软件和硬件厂商共同制定和认可,已成为共享存储系统并行编程的工业标准。对三维骨架细化算法的并行设计就是通过OpenMP多线程编程技术实现的。

OpenMP对程序实现并行运算十分简便,只需在支持并行运算的代码块前加入以`#pragma omp`为开头的编译指令,编译器会自动将并行运算的代码编译成多线程执行,并行运算时每个CPU核心负责执行单独的一个线程(提及线程,CPU以及CPU核心不作区别)。改进后支持并行运算的三维细化算法的伪代码如表1所示。

代码中除了每个线程独自拥有一个指针指向图像之外,图像数据、搜索得到的简单点删除列表都在内存中共享,而将图像和简单点删除队列进行分块处理,使得搜索简单点的每一个线程完全独立,实现并行。

## 3 实验结果与分析

实验平台的配置为:CPU为Intel(R)Xeon E5-2640 v2,最大支持16线程;内存为96 G;操作系统为Ubuntu 14.04。程序使用C++语言编程实现,代码编译器为g++ 4.8.2,支持OpenMP 3.1版本;为获得最大性能,并行化运算使用最大的16个线程,且程序在Release模式下编译。

采用不同大小的三维血管图像细化进行骨架提取,并行运算与串行运算的骨架结果一致,图3所示血管与骨架为Vessel\_4。细化算法并行化前后的运算时间如表2所示。并行设计后的细化算法,在运算时间上降低了一个数量级。对于实际的三维CT血管(如Vessel\_6)这类大型图像数据来说,可实现20 min内提取出血管

表 1 并行化设计的三维骨架细化算法的伪代码

Table 1 Pseudo-code of the parallel design method for 3D thinning

---

**Algorithm:** Parallel for 3D thinning skeletonization by OpenMP

---

**Initialization:** deleted\_point=true, region[num\_of\_threads],  
simple\_point\_list[num\_of\_threads]

**Iteration:**

- 1: **while** (deleted\_point=true) **do**
- 2:     **for** borderType ∈ [1,6] //6 kinds of border point
- 4: **#pragma omp parallel firstprivate**(pVoxel) //each thread has its own pointer
- 5: **#pragma omp for**
- 6:     for i ∈ [i, num\_of\_threads] //split image into num\_of\_threads regions
- 7:     pVoxel scan region[i]
- 8:     **if** pVoxel is simple point
- 9:         **then** add pVoxel to simple\_point\_list[i];
- 10:     **end scan** region[i]
- 11:     **end for** //end of parallel for loop
- 12:     deleted\_point=false //preparation for deletion of point
- 13:     sequentially delete simple point in simple\_point\_list
- 14:     **if** delete any point
- 15:         **then** deleted\_point=true
- 16:     **end for** //end delete simple point for one kind of border point
- 17: **end While** // when there is no point to be deleted

---

表 2 并行设计前后 6 组不同大小图像数据的处理时间

Table 2 Computational time of 6 image data groups by two kinds of thinning

	Vessel_1 (13 MB) 254×266×195	Vessel_2 (60 MB) 579×390×280	Vessel_3 (106 MB) 494×560×400	Vessel_4 (120 MB) 580×460×470	Vessel_5 (293 MB) 525×613×954	Vessel_6 (1.95 GB) 1150×1350×1350
Sequential thinning	16.714 s	1 m 30.64 s	3 m 17.000 s	3 m 43.234 s	13 m 30.714 s	176 m 13.019 s
Proposed partial parallel thinning	1.946 s	8.584 s	18.935 s	22.007 s	1 m 12.007 s	13 m 21.023 s
Speedup	8.589	10.559	10.404	9.962	11.259	13.199

骨架,相对于串行细化将近 3 h 的处理时间,加速效果尤为显著。细化算法经过并行化设计,解决了三维血管图像定量分析中骨架提取的运算效率问题。

在并行化加速运算中,用加速比和并行效率对并行化运算进行评价<sup>[24]</sup>。加速比表示运算时间缩短的比率,计算公式为

$$S = \frac{T_{\text{sequential}}}{T_{\text{parallel}}}, \quad (4)$$

式中  $T_{\text{sequential}}$  表示串行运算消耗的时间,  $T_{\text{parallel}}$  表示并行运算消耗的时间。

由表 2 中的运算时间计算出对应的加速比,得到图像大小与加速比之间的关系,如图 4 所示。三维骨架串行细化算法并行化设计后,基本都能获得 10 倍以上的加速比,而且随着图像尺寸的增大,加速比也会略有提高。图 4 中显示处理小图像时获得的加速比较小,这是由于图像较小时,总的计算时间较短,程序并行化产生的开销在总的运行时间中还占有相当的比重,例如本文中设计的图像分块以及简单点删除队列的合并等操作,都会产生额外的运算开销。

在并行化运算中,用并行化效率来衡量每个 CPU 核心(线程)处理算法本身所占据的开销的比例,其值为 0~1,计算公式为

$$E = \frac{T_{\text{sequential}}}{n \times T_{\text{parallel}}} = \frac{S}{n}, \quad (5)$$

式中  $n$  表示并行运算的线程数目。

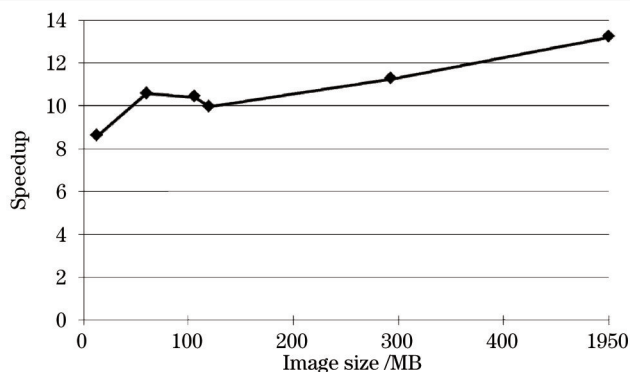


图4 并行设计细化算法对不同尺寸图像的加速比

Fig.4 Speedup for different sizes of images by the proposed parallel thinning method

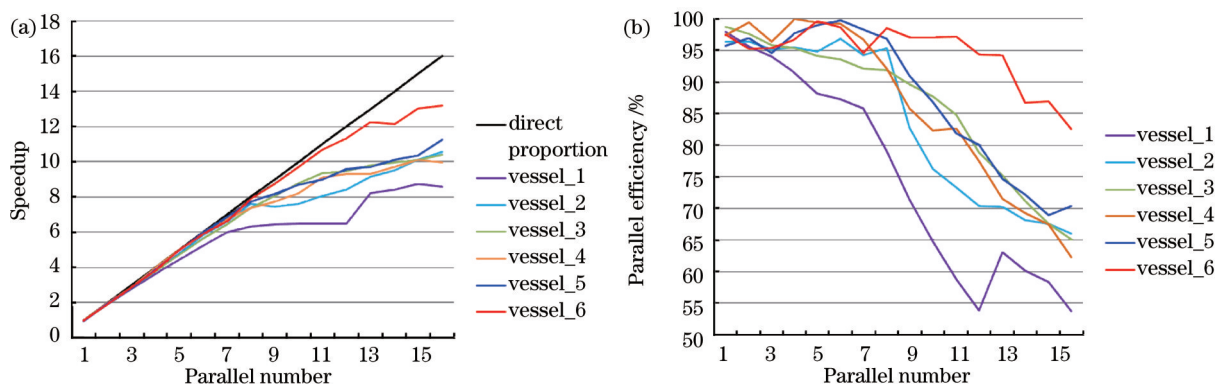


图5 6组血管在不同线程数下的加速比和并行化效率。(a)加速比;(b)并行化效率

Fig.5 Speedup and parallel efficiency under different number of threads for six vessel groups. (a) Speedup; (b) parallel efficiency

理想情况的并行运算,每个CPU的并行化效率为1,其开销完全用于算法本身,多个CPU平均分配算法串行执行时消耗的时间,其加速比是一个正比例函数,如图5(a)中黑线所示。并行运算时CPU除了运行算法本身,还需要额外开销实现程序并行化,如CPU多线程的调度和本文算法针对并行化的额外处理。所以实际的并行化效率总小于1,如图5(b)所示。当并行化的线程数小于8时,细化算法并行化后得到的加速比总体上与线程数成正比关系,而并行化效率也都在80%以上。随着线程数增加,其加速比还会有一定程度的提高,但是并行化设计中的图像分块,合并待删除点和多线程调度等操作造成运算开销进一步加大,使得CPU的并行化效率越来越低。当线程数为8时,大型血管图像数据Vessel\_6的处理时间为1273 s,其细化提取骨架的处理时间也在接受范围内。以上并行化评价测试可以看出,通常配备有4核心i7处理器的普通计算机也能够实现8线程并行化处理大型图像数据,并不需要用到高性能图形工作站,降低了实现该算法的硬件需求。所以该算法高效而简单地实现了三维血管图像中血管骨架的提取。

## 4 结 论

对三维骨架串行细化算法进行深入分析,在保证得到准确骨架的前提下,对其中可并行化的步骤进行并行化设计,并通过OpenMP多线程技术实现。实验结果表明,提出的方法得到的处理结果准确可靠,对于3D大型图像数据,使用16个线程进行并行化运算时,处理速度有10倍以上的提升,且加速比随图像数据量增大而递增。通过并行优化,可使三维血管骨架的提取时间缩短到合理可行的范围内,为三维血管定量分析等大型图像数据的处理与分析提供了一种有效的手段。

## 参 考 文 献

1 Peng Guanyun, Wang Yurong, Ren Haiqing, *et al.*. Investigation of characteristic microstructures of adhesive interface in wood/bamboo composite material by synchrotron radiation X-ray phase contrast microscopy[J]. Spectroscopy and Spectral Analysis, 2013, 33(3): 829-833.

- 彭冠云, 王玉荣, 任海青, 等. 基于同步辐射 X 射线相衬显微 CT 技术的竹木复合材料胶合界面特征研究[J]. 光谱学与光谱分析, 2013, 33(3): 829-833
- 2 Ye Linlin, Xue Yanling, Tan Hai, *et al.*. X-ray phase contrast micro-tomography and its application in quantitative 3D imaging study of wild ginseng characteristic microstructures[J]. Acta Optica Sinica, 2013, 33(12): 1234002.  
叶琳琳, 薛艳玲, 谭海, 等. X 射线相衬显微层析及其在野山参特征结构的定量三维成像研究[J]. 光学学报, 2013, 33(12): 1234002.
- 3 R Chen, P Liu, T Xiao, *et al.*. X-ray imaging for non-destructive microstructure analysis at SSRF[J]. Advanced Materials, 2014, 26(46): 7688-7691.
- 4 Xiao Tiqiao, Xie Honglan, Deng Biao, *et al.*. Progresses of X-ray imaging methodology and its applications at Shanghai Synchrotron Radiation Facility[J]. Acta Optica Sinica, 2014, 34(1): 0100001.  
肖体乔, 谢红兰, 邓彪, 等. 上海光源 X 射线成像及其应用研究进展[J]. 光学学报, 2014, 34(1): 0100001.
- 5 P Liu, J Sun, J Zhao, *et al.*. Microvascular imaging using synchrotron radiation[J]. Journal of Synchrotron Radiation, 2010, 17(4): 517-521.
- 6 M Shirai, D O Schwenke, H Tsuchimochi, *et al.*. Synchrotron radiation imaging for advancing our understanding of cardiovascular function [J]. Circulation Research, 2013, 112(1): 209-221.
- 7 B Deng, Y Ren, Y Wang, *et al.*. Full field X-ray nano-imaging at SSRF[C]. SPIE, 2013, 8851: 88511D.
- 8 B Dong, F Xu, X Hu, *et al.*. *In situ* investigation of the 3D mechanical microstructure at nanoscale: Nano-CT imaging method of local small region in large scale sample[J]. Scientific World Journal, 2014, 2014: 806371.
- 9 H Xie, B Deng, G Du, *et al.*. X-ray biomedical imaging beamline at SSRF[J]. Journal of Instrumentation, 2013, 8(8): C08003.
- 10 Qi Juncheng, Ren Yuqi, Du Guohao, *et al.*. Multiple contrast micro-computed tomography system based on X-ray grating imaging[J]. Acta Optica Sinica, 2013, 33(10): 1034001.  
戚俊成, 任玉琦, 杜国浩, 等. 基于 X 射线光栅成像的多衬度显微计算层析系统[J]. 光学学报, 2013, 33(10): 1034001.
- 11 N D Cornea, D Silver, P Min. Curve-skeleton properties, applications, and algorithms[J]. IEEE Transactions on Visualization and Computer Graphics, 2007, 13(3): 530-548.
- 12 T C Lee, R L Kashyap, C N Chu. Building skeleton models via 3-D medial surface axis thinning algorithms[J]. CVGIP: Graphical Models and Image Processing, 1994, 56(6): 462-478 .
- 13 G Borgefors. On digital distance transforms in three dimensions[J]. Computer Vision and Image Understanding, 1996, 64(3): 368-376.
- 14 J W Brandt, V R Algazi. Continuous skeleton computation by Voronoi diagram[J]. CVGIP: Image Understanding, 1992, 55(3): 329-338.
- 15 N D Cornea, D Silver, X Yuan, *et al.*. Computing hierarchical curve-skeletons of 3D objects[J]. Visual Computer, 2005, 21(11): 945-955.
- 16 T Saito. A sequential thinning algorithm for three dimensional digital pictures using the Euclidean distance transformation[C]. Proceedings of the 9<sup>th</sup> SCIA, 1995: 507-516.
- 17 K Palágyi, E Balogh, A Kuba, *et al.*. A sequential 3D thinning algorithm and its medical applications[M]// Information Processing in Medical Imaging. Berlin: Springer Berlin Heidelberg, 2001, 2082: 409-415.
- 18 C M Ma, M Sonka. A fully parallel 3D thinning algorithm and its applications[J]. Computer Vision and Image Understanding, 1996, 64(3): 420-433.
- 19 K Palagyi, A Kuba. A parallel 3D 12-subiteration thinning algorithm[J]. Graphical Models and Image Processing, 1999, 61(4): 199-221.
- 20 W Xie, R P Thompson, R Perucchio. A topology-preserving parallel 3D thinning algorithm for extracting the curve skeleton[J]. Pattern Recognition, 2003, 36(7): 1529-1544.
- 21 T Wang, A Basu. A note on 'a fully parallel 3D thinning algorithm and its applications' [J]. Pattern Recognition Letters, 2007, 28(4): 501-506 .
- 22 T Y Kong, A Rosenfeld. Digital-topology - introduction and survey[J]. Computer Vision Graphics and Image Processing, 1989, 48(3): 357-393.
- 23 K Palágyi, G Németh, P Kardos. Topology preserving parallel 3D thinning algorithms[M]// Digital Geometry Algorithms. Berlin: Springer, 2012: 165-188.
- 24 J L Hennessy, D A Patterson. Computer Architecture: A Quantitative Approach[M]. Waltham: Elsevier, 2011.

栏目编辑: 吴秀娟