

基于策略演化水平集的医学图像快速分割

董建园 郝重阳 齐敏

(西北工业大学生物医学工程研究所, 西安 710072)

摘要 医学图像分割在疾病诊断、手术规划和手术引导等实际应用中有着重重要的作用。提出了一种基于策略演化水平集算法的快速医学图像分割方法,其策略是通过转换外部轮廓曲线/曲面上的点为内部轮廓曲线/曲面上的点(或做相反操作时),检验能量函数是否减小来决策水平集演化;如此扫描内外轮廓曲线/曲面,使得分割曲线/曲面向目标边界移动。相对于传统水平集算法,该方法不需要解偏微分方程,可极大地减小计算量、提高图像分割的速度。同时,该算法克服了直接计算能量函数水平集方法中存在的问题(陷入局部能量最小和需要扫描整个图像)。最后通过2维和3维医学图像的分割实验,展示了该算法的快速性与精确性。

关键词 水平集 3维分割 策略演化

中图法分类号: TP391.41 文献标识码: A 文章编号: 1006-8961(2009)08-1689-07

Medical Image Segmentation Based on the Policy Evolution Level Sets

DONG Jian-yuan, HAO Chong-yang, QI min

(Department of Biomedical in NorthWestern Polytechnical University, xi'an 710072)

Abstract Medical image segmentation plays an important role in practical applications such as diseases diagnosis, surgical planning, and surgical guidance. In this article, we propose a fast medical image segmentation method based on the policy evolution level sets. Our evolution policy is to calculate the energy directly and check if the energy is decreased when we switch a point from the outer contour to the inner contour (or vice versa). By scan points of inner and outer contour, make the curve or surface move inward or outward to go to the boundary of object. This approach differs from the previous methods in that we do not need to solve PDEs, it can improves the computational speed dramatically. The problem (the local minimums and scan the whole image) of energy function calculate method is solved. At last some segmentation experiments is make on medical image in 2D image and 3D volume, and it demonstrated that our algorithm is fast and precision.

Keywords level set, 3D segmentation, policy evolution

1 引言

近年来,在图像分割、目标跟踪领域,水平集算法得到人们极大的关注^[1-3]。尽管水平集方法有一系列的优点,但是它通过求解偏微分方程的实现算法计算量过大,从而限制它在实时跟踪和3维分割领域的应用。本文提出了基于策略演化的水平集算

法,极大地减小了计算量,提高了分割的速度。

基于水平集的数值实现算法,需要对整个图像定义域的所有点的水平集函数进行更新。这种方式的计算量很大。其实,当人们只关心零水平集的演化过程,如果对整个水平集函数进行更新,显然有许多无效的计算。在该领域比较值得关注的相关研究有 Adalsteisson 等人提出的窄带法^[4]。其基本思想是只演化位于零水平附近很窄的一个带状区域水平

集函数的值。不过,窄带法存在的问题是经过几次迭代后,零水平集的位置可能超出窄带的范围,需要更新窄带的内外边界点,重新初始化水平集函数,使其保持为符号距离函数。此外是 Tsitsiklis 的全局优化算法^[5],它可以保证一次迭代的最优性,但是它的不足在于计算代价较高,而且易受初始位置和参数变化影响。再就是 Sethian 提出的快速行进法(Fast Marching)^[6]。即当曲线演化的速度总保持符号不变,曲线演化方程就是一个静态 Hamilton-Jacobi 方程。其基本思想是在传播边界的外围构造一个激活窄带,利用逆向机制将当前边界向外传播,就像水波扩散一样,凡是扩散到的点,就冻结其波前到达时间,然后根据当前的波前构造新的激活带,如此循环,就可以得到整个图像每个点的到达时间。但该方法需要构建速度场,算法的迭代停止条件也难以确定。

2 水平集的策略演化

水平集方法由 Osher 和 Sethian 提出^[2],现已广泛应用于计算物理、图像处理 and 计算机图形学。其思想是用高维空间零水平集函数来描述进化曲线或曲面。

考虑一个封闭可运动的曲线或曲面 $\Gamma(t)$, $\Omega(t)$ 为由 $\Gamma(t)$ 包围的区域或空间。水平集方法把 Γ 看成是函数 $\phi(x, t)$ 的零水平集。即

$$\begin{cases} \phi(x, t) < 0 & \text{for } x \in \Omega \\ \phi(x, t) = 0 & \text{for } x \in \Gamma \\ \phi(x, t) > 0 & \text{for } x \in \bar{\Omega} \end{cases} \quad (1)$$

其中, $x \in \mathbf{R}^n$, $\bar{\Omega}$ 为 Ω 外部区域,曲线或曲面 $\Gamma(t)$ 在速度场 V 的作用下演化,在经典的水平集方法中^[2],是通过数值计算方法求解下面的偏微分方程来得到 ϕ 。

$$\frac{d\phi}{dt} + V |\nabla \phi| = 0 \quad (2)$$

如果知道 ϕ ,就可以通过寻找 ϕ 函数的零水平集来确定目标曲线或曲面,即 $\Gamma(t) = \{x; \phi(x, t) = 0\}$ 。利用迭代法进行水平集函数演化时,时间间隔不能选择得太大,以保证偏微分方程数值计算的稳定性,即满足 CFL 条件。这会导致收敛速度比较慢。

然而,在很多情况下,不需求解 ϕ 的值,只需知道 ϕ 的正负号。比如图像分割,人们感兴趣的是目标物体的边界(零水平集),而不是所有的水平集

的值。基于这一考虑,可以得到一种新的直接算法^[7]。算法如下。

(1) 初始化。构造一个初始分割曲线或曲面,在其内部 $\phi < 0$,在其外部 $\phi > 0$,根据 ϕ 计算能量函数 F (在第 3 部分详述)。

(2) 扫描。对于图像中的每一个像素点 x ,如果改变 $\phi(x)$ 为 $-\phi(x)$ 时能量 F 减小,则更新此点的 $\phi(x) = -\phi(x)$,否则,保持 $\phi(x)$ 不变。如此扫描整个图像。

(3) 重复第 2 步直到能量 F 不改变为止。

3 基于 Chan-Vese 模型分割

Chan-Vese 图像分割模型^[8]的基本思想是寻找一个边界将图像分割为两个区域,一个区域为被检测的目标,另一个区域为背景。设 C_0 为目标的边界,能量函数可定义如下:

$$F = \alpha \times (\text{Length}(C)) + \nu \times \text{Area}(\text{inside}(C)) + \lambda_1 \int_{\text{inside}(C)} |\mu_0 - C_1|^2 dx dy + \lambda_2 \int_{\text{outside}(C)} |\mu_0 - C_2|^2 dx dy \quad (3)$$

C 是任何可变曲线, C_1 和 C_2 是图像 μ_0 在 C 内部和外部灰度平均值, $\alpha, \nu, \lambda_1, \lambda_2$ 是非负常数。调整项为曲线 C 的长度用来控制边界的光滑性。如果 $C = C_0$,这个分割将会使能量达到最小。如果用水水平集来描述 C ,即 C 是函数 ϕ 的零水平集,则能量函数 F 为

$$F = \alpha \int_{\Omega} |\nabla H(\phi)| dx dy + \nu \int_{\Omega} H(\phi) dx dy + \lambda_1 \int_{\Omega} |\mu_0 - C_1|^2 H(\phi) dx dy + \lambda_2 \int_{\Omega} |\mu_0 - C_2|^2 (1 - H(\phi)) dx dy \quad (4)$$

在这里 $\int |\nabla H(\phi)| dx dy$ 为 $\Gamma(t)$ 的长度, $\int H(\phi) dx dy$ 为 $\Omega(t)$ 的面积, $H(\phi)$ 为

$$H(\phi) = \begin{cases} 1 & \phi \leq 0 \\ 0 & \phi > 0 \end{cases} \quad (5)$$

为了求解方程(4)最小化问题,通常采用解 Euler-Lagrange 方程的方法。但该方法收敛速度慢,计算量大。利用前述思想,针对 C-V 模型的算法如下(为简化运算取 $\alpha = 0, \nu = 0$,即不考虑曲线的长度和面积):

(1) 对于一幅图像, 设定其初始分割曲线, 在其内部 $\phi = -1$, 在其外部 $\phi = 1$ 。

(2) 假定当前像素点 x 的灰度值为 I_x, C_1 和 C_2 为 $\phi = 1$ 和 $\phi = -1$ 区域的灰度平均值, m 和 n 为相应区域内像素的个数。

当 $\phi(x) = 1$, 计算新旧能量之差:

$$\Delta F_{12} = (I_x - C_2)^2 \frac{n}{n+1} - (I_x - C_1)^2 \frac{m}{m-1}$$

如果 $\Delta F_{12} < 0$, 那么改变 $\phi(x)$ 值从 $+1$ 到 -1 。

当 $\phi(x) = -1$, 计算新旧能量之差:

$$\Delta F_{21} = (I_x - C_1)^2 \frac{m}{m+1} - (I_x - C_2)^2 \frac{n}{n-1}$$

如果 $\Delta F_{21} < 0$, 那么改变 $\phi(x)$ 值从 -1 到 $+1$ 。

(3) 重复第 2 步直到能量 F 不改变为止。

公式 ΔF_{12} 和 ΔF_{21} 的推导见文献[7], 该算法虽然计算量较小, 速度快, 但它比较适用于目标和背景相对比较单一的图像, 因为该方法仅仅依靠初始轮廓曲线, 而没有其他信息如灰度范围、边缘特征、区域连接的支持。在能量函数的计算中, 有时会陷入局部能量最小, 从而使迭代运算停止达不到分割目标, 造成错分割。此外, 该分割算法需要扫描整个图像中的每一个像素, 会增大计算量, 考虑到以上因素, 引入了窄带方法的思想。

4 曲线和曲面进化分析

在水平集方法中, 曲线 C 被隐式描述为函数 ϕ 的零水平集。受窄带算法思想的启迪, 可以将这个窄带缩减到极致, 即为 1。引入曲线 C 的内邻域点集 L_{in} 和外邻域点集 L_{out} , 对于一个固定的曲线 C 来说, 由曲线 C 确定的内外邻域点集 L_{in} 和 L_{out} 是唯一的。如图 1 所示。

对于图像分割来说, 在水平集的演化过程中, 只关心零水平集即曲线 C 的移动, 而对 ϕ 的精确值是多少不感兴趣。也就是说只需知道 ϕ 的符号, 而不是它的数值。

把曲线在 C 内部但不属于 L_{in} 的点称为内部点, 而把那些位于曲线 C 外部但不属于 L_{out} 的点称为外部点。水平集符号距离函数 ϕ 定义如下:

$$\phi(x) = \begin{cases} 3 & x \text{ 是外部点} \\ 1 & x \text{ 位于 } L_{out} \text{ 上} \\ -1 & x \text{ 位于 } L_{in} \text{ 上} \\ -3 & x \text{ 是内部点} \end{cases} \quad (6)$$

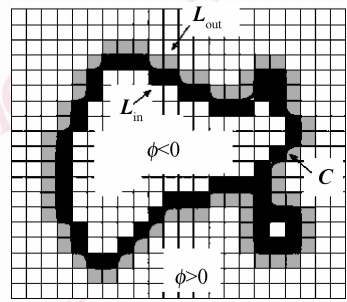


图 1 曲线 C 的邻域点集 L_{in} 和 L_{out} of curve C

在基于水平集的图像分割中, 由方程(2)可知, 如何构造演化速度场是关键, 它决定分割的最终效果。经典的解决办法是基于图像特征进行构建。一种为基于图像边缘特征, 如梯度, 另一种为基于图像的区域特征, 如灰度范围。而本文提出的策略演进算法, 只需扫描 L_{in} 和 L_{out} 中的点而不是整幅图像, 通过直接计算能量函数的增减, 而不需构建速度场, 来更新函数 ϕ 。

对于 L_{out} 中的一个点 A , 当曲线 C 向外扩展时, 点 A 的 ϕ 值由正变负; 而对于 L_{in} 中一点 B , 当曲线向内收缩, 点 B 的 ϕ 值由负变正。即将曲线扩展到点 A 以外, 只需将点 A 由 L_{out} 转换到 L_{in} 中; 同样只需将点 B 由 L_{in} 转换到 L_{out} , 曲线即可收缩。对于 L_{in} 和 L_{out} 中的所有点应用这样的处理, 就可将曲线 C 向内或外移动一格, 而且运算量极小^[9]。

曲线 C 在某 x 点向外扩展或向内收缩一格, 实际上是把 $\phi(x)$ 由 $+1$ 置为 -1 或由 -1 变为 $+1$, 这恰好是本文第 2 部分的策略演进思想。将两者相融合, 产生新的算法: 设 $I(x)$ 为像素点 x 的灰度值, $[I_1, I_2]$ 是欲分割目标的灰度范围。

(1) 设定初始分割曲线/曲面, ϕ, L_{out}, L_{in} 。

(2) 扫描 L_{in} 和 L_{out} 中的所有点, 更新 ϕ, L_{in}, L_{out} 。

① 向外扩展。扫描 L_{out} , 对于 L_{out} 中的每一个点 x , 如果 $\Delta F_{12} < 0$, 曲线 C 在该点向外扩展一格。

② 删除 L_{in} 中冗余的点。扫描 L_{in} , 如存在 L_{in} 中的点 x 的邻点 y , 所有 $\phi(y) < 0$, 则将 x 从 L_{in} 中删除, 并置 $\phi(x) = -3$ 。

③ 向内收缩: 扫描 L_{in} , 对于 L_{in} 中的每一点, 如果 $\Delta F_{21} < 0$, 则曲线 C 在该点向内收缩一格。

④ 删除 L_{out} 中冗余的点: 扫描 L_{out} , 如果存在

L_{out} 中的点 x 的邻点 y , 所有 $\phi(y) > 0$, 则将 x 从 L_{out} 中删除, 并置 $\phi(x) = 3$ 。

(3) 内外轮廓 L_{in} 和 L_{out} 上所有点的灰度值满足式(7)作为停止条件, 如满足则停止, 否则, 返回第(2)步。

$$\begin{cases} J_{L_{out}} & \notin [I_1, I_2] \\ J_{L_{in}} & \in [I_1, I_2] \end{cases} \quad (7)$$

5 医学图像分割实例

基于 Chan-Vese 图像分割模型, 利用本文提出的新算法, 即仅扫描 L_{in} 和 L_{out} 的策略水平集演化算法对一些图像进行分割。本文算法在 IDL6.2 平台实现, 其中如何对内外邻域曲面上的点进行扫描是影响速度的关键。

一幅 512×512 像素的胸部 CT 图像。预分割目标为肺部, 在图中下方的矩形为初始轮廓, 图 2 为迭代 100 步时的分割结果。图 3 为迭代 280 步时的最终分割曲线。



图 2 迭代 100 步时的分割曲线

Fig.2 Segmentation curve for 100 step



图 3 迭代 280 步时的分割曲线

Fig.3 Segmentation curve for 280 step

下面, 将该分割算法应用于 3 维医学图像的分割。对于 3 维图像, 由于其层间数据的相关性, 可以把上一切片的分割结果, 作为下一张切片的初始轮廓, 即间接的 3 维体数据分割。

某头部 CT 体数据, 体素为 $[256 \times 256 \times 124]$, 预分割目标为脑室。在图 4 中从左至右, 从上至下依次为体数据的体绘制图像、室状位、冠状位、和轴状位视图(对于体数据的描述以下雷同)。脑室位于切片 38 到切片 85 之间, 在第 38 切片设置初始轮廓为长度为 10 个像素的矩形, 其中心点位于点 $(165, 110)$, 目标灰度分布区间为 $[0, 60]$ 。在配置为 CPU1.6G, RAM 为 384M 的机器上, 分割时间为 11 s。图 5 为切片 60, 64, 68, 70 的分割结果。图 6 为分割出的脑室不同角度体绘制的图像。

脑室位于切片 38 到切片 85 之间, 在第 38 切片设置初始轮廓为长度为 10 个像素的矩形, 其中心点位于点 $(165, 110)$, 目标灰度分布区间为 $[0, 60]$ 。在配置为 CPU1.6G, RAM 为 384M 的机器上, 分割时间为 11 s。图 5 为切片 60, 64, 68, 70 的分割结果。图 6 为分割出的脑室不同角度体绘制的图像。

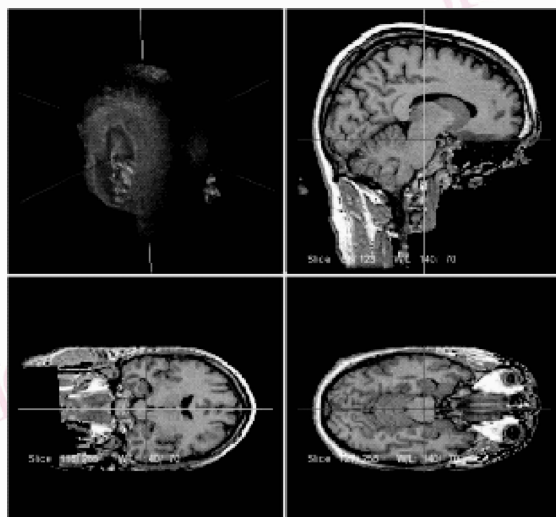


图 4 头部体数据

Fig.4 Volume data for heda

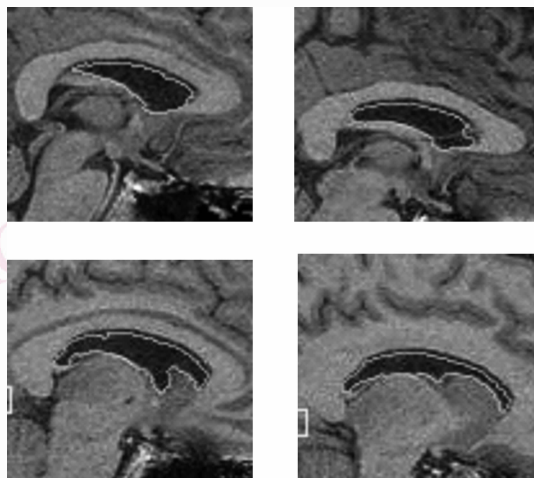


图 5 切片 60, 64, 68, 70 的分割结果

Fig.5 Segmentation result for 60, 64, 68, 70 slice

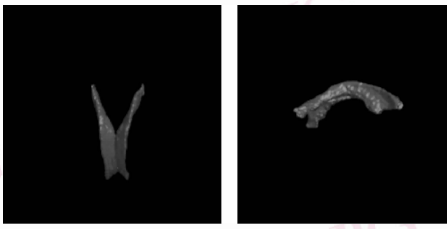
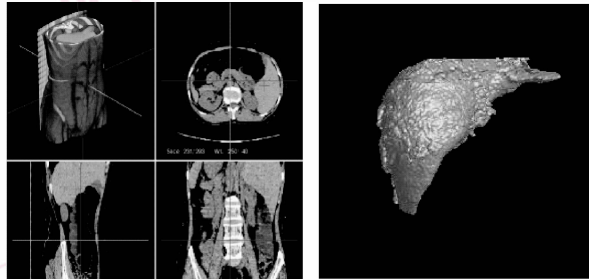


图 6 脑室体绘制图像

Fig. 6 Volume render image of brain ventricle



(a) 胸腹部CT体数据

(b) 肝脏体绘制图像

图 7 肝器官分割

Fig. 7 Segmentation for liver

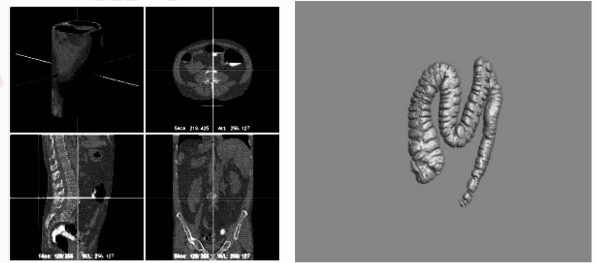
某人体胸腹部 CT 体数据,其体素为 $[256 \times 256 \times 294]$,如图 7(a)所示,预分割目标为肝脏。肝脏位于切片 0 ~ 113 之间,设置初始轮廓于 0 号切片,长度为 36 个像素的矩形,其中心点位于点 (236, 183),目标灰度分布区间为 $[333, 185]$,在上述配置的机器上,分割时间仅 45 s。图 7(b)为分割出的肝脏的体绘制图像。

该分割算法既适用于 2 维图像的分割,也可扩展到 3 维图像的分割。设 x 为内外轮廓 L_{in}, L_{out} 上的点,若设置初始轮廓为曲线,在更新 L_{in}, L_{out} 和 ϕ 时,取 x 的邻点为四邻点即为 2 维分割。若设置初始轮廓为曲面,在更新 L_{in}, L_{out} 和 ϕ 时,取 x 的邻点为空间六邻点即为 3 维分割。只是在 3 维时,公式 ΔF_{12} 和 ΔF_{21} 中的 C_1 和 C_2 的物理意义表示为相应空间的体积的平均值。

对于上述头部数据,采用直接 3 维分割算法,初始分割曲面是半径为 5,中心点位于点 (109, 87, 57) 的球面,目标灰度分布区间为 $[0, 60]$,在上述配置的机器上,分割脑室仅耗时 6 s。

图 8(a)是另一腹部 CT 体数据,其体素为 $[256 \times 256 \times 426]$,为了从 3 维体数据中分割出大肠(位于切片 90 至 400 之间,大小为 $256 \times 256 \times 301$),设置初始分割曲面为以点 (99, 53, 270) 为中心,半径为

30 的球面,在上述配置的机器上,分割时间为 101 s。图 8(b)是分割出的大肠的体绘制图。



(a) 腹部CT体数据

(b) 大肠的体绘制图

图 8 大肠的分割

Fig. 8 Segmentation for big intestine

对人体的大动脉血管,可以直接从 3 维 CT 图像中分割出血管,而不必注入对比剂进行血管增强。有一腹部和盆骨 CT 体数据如图 9 所示,体素为 $[256 \times 256 \times 273]$ 。设置初始分割曲面为以点 (139, 95, 130) 为中心,半径为 2 的球面,灰度范围为 $[124, 136]$ 。图 10 为本文算法迭代 60, 120, 180, 240 步时的分割结果。在上述配置机器上,最终分割时间仅 10 s。

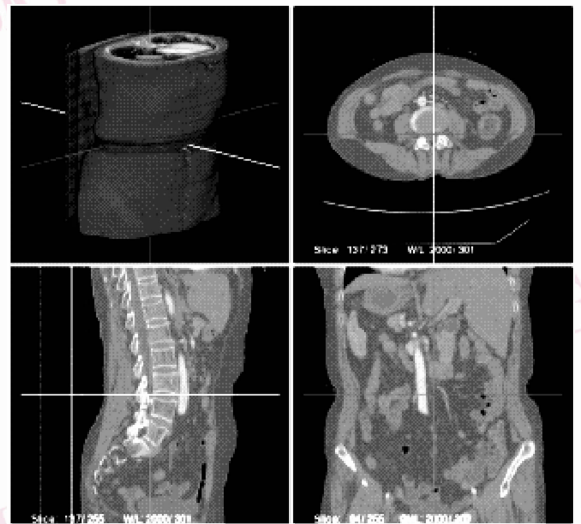


图 9 腹部和盆骨体数据

Fig. 9 Volume data of abdomen and pelvis

下面将本分割算法同快速行进法进行比较。Sethian 详细地分析了快速行进法的计算复杂度^[6](每一迭代步骤)为: $O(N \log N)$, N 为图像的点数;而窄带算法的计算复杂度为 $O(NK)$, K 为窄带宽度;水平集的直接数值计算法的计算复杂度为 $O(N^2)$ 。可见快速行进算法速度相对较快。

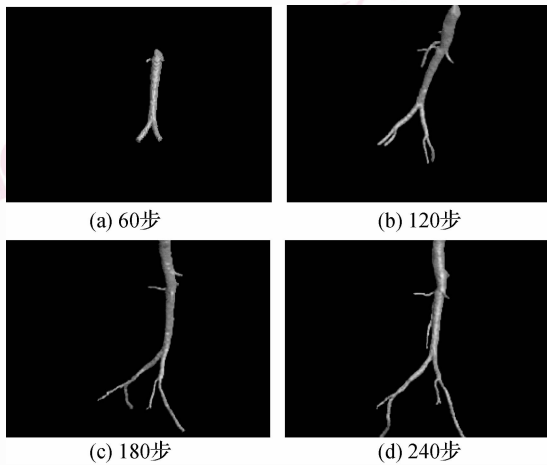


图 10 迭代步数不同时算法的分割结果

Fig. 10 Segmentation results with different iteration steps

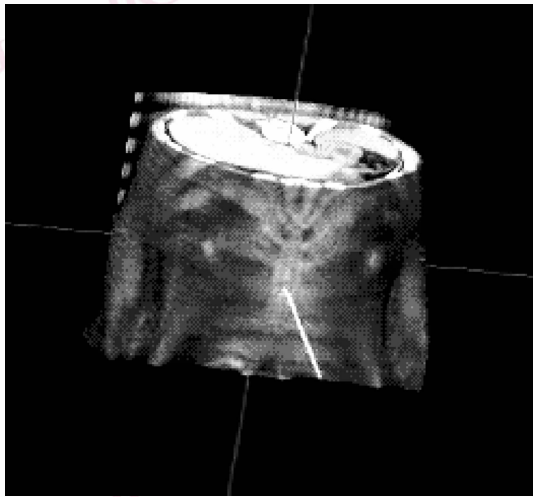


图 11 胸部 CT 体绘制图

Fig. 11 Volume render image for chest CT data

将本文算法和 ITK (insight toolkit)^[10] 的快速行进法做一比较。图 11 是一胸部 CT 体绘制图,体素为 $[256 \times 256 \times 170]$,待分割对象为肺器官。设置初始分割曲面为以点 $(127, 62, 110)$ 为中心,半径为 5 的球面,在上述配置的机器上,本算法共迭代 220 步,耗时 62 s,其分割结果如图 12(a) 所示;而快速行进算法耗时 160 s (设置 8 个种子点以减小计算时间,在计算图像的梯度场时,若需要进行高斯平滑滤波,总时间可能需要几分钟),其分割结果如图 12(b) 所示。可见本分割算法的速度和精度都有一定程度的提高。

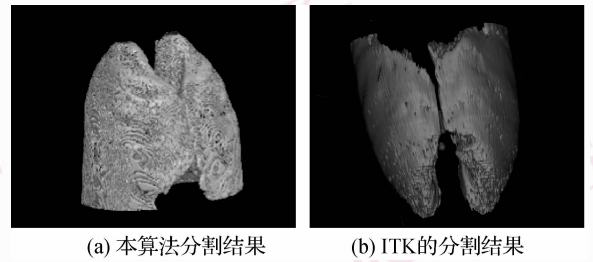


图 12 肺器官分割结果

Fig. 12 Segmentation result for lung

6 结 论

Adalsteisson 和 Sethian 提出的窄带法^[4]。其基本思想是只演化位于零水平集附近很窄的一个带状区域水平集函数的值,即只在该带状区域求解偏微分方程,比较合理的窄带宽度是 $K = 12 \sim 16$ 。基本流程是:构造窄带和迭代演化方程。不过,窄带法存在的问题是经过几次迭代后,零水平集的位置可能超出窄带的范围,需要更新窄带的内外边界点,重新初始化水平集函数,使其保持为符号距离函数。

而本文算法不需要计算符号距离函数,而是直接定义,形如文中式(6);不需要解偏微分方程;在水平集的演化过程也不需要重新初始化水平集函数;窄带宽度是 $K = 2$,同窄带算法相比,本文的算法将计算限制在内、外轮廓线之上。因而本算法分割速度较快。

能量函数的定义依赖于图像分割的模型,图像分割的精度也依赖于所采用的模型,本文算法是基于 Chan-Vese 模型,即将图像划分为两个不同的常数区域,实际上是一种简化的 Mumford-Shah 模型。

Mumford-Shah 模型可同时实现图像分割和滤波,它将图像划分为多个不同的平滑函数区域,该模型可克服 Chan-Vese 模型的对噪声敏感的缺陷,代价是算法更加复杂。本文算法也可推广至 Mumford-Shah 模型,基于 Mumford-Shah 模型能量函数的定义有兴趣的读者可参考文献[11]。

本文提出的基于策略演化水平集的快速分割方法。其特点是:

(1) 通过直接计算能量函数,考察当图像中一点的水平集函数符号改变是否使能量函数减小,来确定水平集的更新,而不是使用传统的解偏微分方程方法。

(2) 水平集的演化既不需要扫描整个图像,也

有别于一般的窄带方法,它是通过内外轮廓曲线上点的转换,使曲线或曲面进行扩展和收缩,这可大大减小计算量。

(3)该方法不需要构建速度场,从而在一定程度上绕开了水平集方法图像分割的瓶颈。而把问题的解决建立在新旧能量函数之差的计算上。

在实际的图像分割中,可不使用能量函数的改变作为算法的迭代停止条件。而把目标区域的灰度范围或边缘信息作为迭代停止条件,这样可以避免陷入局部最小问题。

上述算法仅能分割一个目标,下一步的研究目标是将其扩展到多目标分割。比如,用两个水平集去描述图像中的4个区域。

参考文献 (References)

- 1 Osher S, Fedkiw R. Level set methods: An overview and some recent result [J]. *Journal of Computational Physics*, 2001, **169** (2): 463-502.
- 2 Osher S, Sethian J A. Fronts propagation with curvature dependent speed; algorithms based on Hamilton-Jacobi formulation [J]. *Journal of Computational Physics*, 1998, **79** (1): 12-49.
- 3 Paragios N, Deriche R. Geodesic active contours and level sets for the detection and tracking of moving objects [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000, **22** (3): 266-280.
- 4 Adalsteinsson D, Sethian J A. A fast level set method for propagating interfaces [J]. *Journal of Computational Physics*, 1995, **118** (2): 269-277.
- 5 Tsitsiklis J N. Efficient algorithms for globally optimal trajectories [J]. *IEEE Transactions on Automatic Control*, 1995, **40** (9): 1528-1538.
- 6 Sethian J A. A fast marching level set method for monotonically advancing fronts [J]. *Proceedings of the National Academy of Sciences*, 1996, **93** (4): 1591-1595.
- 7 Song B, Chan T. A fast algorithm for level set based optimization [R]. CAM-VCLA, Los Angeles, CA, USA; University of California, 2002.
- 8 Chan T, Vese L. Active contour without edges [J]. *IEEE Transactions on Image Processing*, 2001, **10** (2): 226-277.
- 9 Shi Yong-gang, William C K. A fast level set method without solving PDEs [A]. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing [C]*, Philadelphia, Penn. USA: 2005: 97-100.
- 10 The Insight Toolkit [CP/OL], <http://www.itk.org>, 2006.
- 11 Yu Lu, Wang Qiao. Mumford-Shah mode with fast algorithm on lattice [A]. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing [C]*, Toulouse, France, 2006.