

# 基于多视图信息的CT序列图像颈椎间盘分割算法

代鹏程<sup>1</sup>, 任超<sup>1</sup>, 滕奇志<sup>1</sup>, 杨毅<sup>2</sup>

(1.四川大学电子信息学院图像信息研究所, 四川 成都 610065;

2.四川大学华西医院骨科, 四川 成都 610041)

✉2573896621@qq.com; chaoren@scu.edu.cn; qzteng@scu.edu.cn; hxyangyi@l63.com



**摘要:** 从颈椎CT(电子计算机断层扫描)序列图像中准确地分割出颈椎间盘是颈椎终板三维分型的基础,但在原始水平面数据中,颈椎间盘没有明显的边界,难以实现高精度的颈椎间盘分割。为此,在深入研究颈椎间盘本身的三维空间结构以后,提出了一种结合多视图信息的颈椎间盘分割算法,其能充分利用颈椎间盘上下表面的三维空间信息,解决了颈椎间盘边界在水平面难以界定的问题,使分割得到的颈椎间盘更加精确。从多组颈椎CT序列图像上的实验结果表明,文章提出的算法与双阈值法和天津法相比,实现了精度更高的颈椎间盘分割,更符合实际应用的需求。

**关键词:** 颈椎CT图像; 图像分割; 颈椎间盘

**中图分类号:** TP391.4 **文献标识码:** A

## Segmentation Algorithm of Cervical Intervertebral Disc in CT Sequence Image based on Multi-view Information

DAI Pengcheng<sup>1</sup>, REN Chao<sup>1</sup>, TENG Qizhi<sup>1</sup>, YANG Yi<sup>2</sup>

(1. Institute of Image Information, School of Electronic Information, Sichuan University, Chengdu 610065, China;

2. Department of Orthopedics, West China Hospital, Sichuan University, Chengdu 610041, China)

✉2573896621@qq.com; chaoren@scu.edu.cn; qzteng@scu.edu.cn; hxyangyi@l63.com

**Abstract:** Accurate segmentation of cervical intervertebral disc from cervical CT (Computed Tomography) sequence images is the basis of three-dimensional classification of cervical endplate. However, cervical intervertebral disc has no obvious boundary in the original horizontal data, so it is difficult to achieve high precision cervical intervertebral disc segmentation. Therefore, this paper proposes a cervical disc segmentation algorithm based on multi-view information after in-depth study of the three-dimensional spatial structure of the cervical disc itself. This algorithm makes full use of the three-dimensional spatial information of the upper and lower surfaces of the cervical disc, solves the problem that the boundary cervical disc is difficult to be defined in the horizontal plane, and makes the segmentation of the cervical intervertebral disc more accurate. The Results of the experiments on multiple sets of cervical vertebra CT sequence images show that the proposed algorithm achieves higher precision segmentation of cervical intervertebral disc, compared with double threshold method and OTSU method. It is more in line with the needs of practical applications.

**Keywords:** cervical vertebra CT image; image segmentation; cervical intervertebral disc

## 1 引言(Introduction)

医学上将颈椎骨质增生和椎间盘退行性病变引起的综合征命名为“颈椎病”<sup>[1]</sup>,近年来,颈椎病的发病率呈现出日益升高的趋势,并且从发病的群体来看,逐渐趋于年轻化,引起了医学界的高度关注<sup>[2]</sup>。颈椎人工椎间盘置换术是目前临床

常用的颈椎病治疗术之一,它具有安全、有效等优点<sup>[3]</sup>,与椎间盘置换密切相关的便是颈椎终板的三维分型,它对于指导颈椎植入物的3D打印有着重要意义。如何从颈椎CT序列图像中准确、无误地分割出颈椎间盘则是颈椎终板三维分型的关键和前提。在原始水平面CT图像中,存在如下问题:颈

椎间盘灰度值分布和颈椎中除椎骨以外的其他组织或器官的灰度值分布相近；原始的水平面图像中的颈椎间盘边界难以界定。这些情况增加了对颈椎CT序列图像中颈椎间盘的分割难度。

目前，针对医学CT图像分割的方法主要有基于深度学习和基于传统的方法两类。深度学习方法中，针对感兴趣目标的图像分割大都使用完全监督的方法训练神经网络，这个过程需要大量的标签图像。因为对CT图像中颈椎间盘的标注非常耗时，并且通常需要一定的专业知识，所以无法快速获得大量精准的颈椎间盘标签图像。而在没有标签图像或者仅有少量标签图像的情况下，深度学习的分割效果一般达不到预期，故本研究使用传统方法来解决CT图像中颈椎间盘分割问题。传统的医学图像分割方法主要包括阈值法、边界法、区域增长法及统计学方法等<sup>[4]</sup>。KIM等<sup>[5]</sup>通过CT图像的特定阈值及分水岭算法实现全自动地分割髋关节，精准地完成股骨的分割工作，但是最佳阈值需要根据不同CT图像进行变换，算法的泛化性差。ANTER等<sup>[6]</sup>提出了一种基于中子集、FCN及自适应性分水岭算法的CT肝脏肿瘤分割方法，此方法对于肝脏肿瘤的分割准确度高，但是算法的处理过程较为耗时。李孝斐<sup>[7]</sup>对肝脏肿瘤CT图像进行直方图均衡化、中值滤波等预处理，再运用混合滤波策略的区域生长算法实现对肝脏肿瘤的有效分割。

针对上述对颈椎CT序列图像中颈椎间盘分割存在的问题，本文提出了一种结合多视图信息的颈椎间盘分割方法，其能充分利用颈椎间盘上下表面的三维空间信息，解决了颈椎间盘在水平面“不可见”及颈椎间盘边界难以界定的问题，使分割得到的颈椎间盘更加精确。

## 2 算法原理(Principle of algorithm)

为了便于后文的描述，在此处先对本文使用的数据集、坐标系、符号表示及颈椎间盘分割整体思路做出说明。

本文用来示意的颈椎CT图像序列图为水平面序列图，图像宽度为  $w$ ，高度为  $h$ ，共  $k$  张，它们之间的对应关系如图1所示。

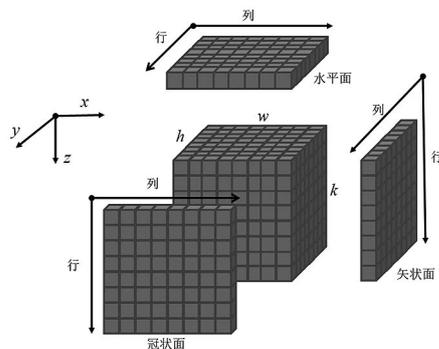


图1 坐标系及位面示意图

Fig.1 Coordinate system and plane diagram

为了描述方便，引入如下符号表示。

水平面序列图像数学表示：给定一个整数  $z_0$ ， $0 \leq z_0 < k$ ，令  $T_l(z_0) = \{(x, y, z_0) | 0 \leq x < w, 0 \leq y < h\}$ ， $x, y$  均为整数，表示一张水平面图像中所有的像素点。进一步令  $Z = \{z_0 | 0 \leq z_0 < k\}$ ，故  $T_l(Z)$  表示整个水平面序列图像。

冠状面序列图像数学表示：给定一个整数  $y_0$ ， $0 \leq y_0 < h$ ，令  $C_l(y_0) = \{(x, y_0, z) | 0 \leq x < w, 0 \leq z < k\}$ ， $x, z$  均为整数，表示一张冠状面图像中所有的像素点。进一步令  $Y = \{y_0 | 0 \leq y_0 < h\}$ ，故  $C_l(Y)$  表示整个冠状面序列图像。

矢状面序列图像数学表示：给定一个整数  $x_0$ ， $0 \leq x_0 < w$ ，令  $S_l(x_0) = \{(x_0, y, z) | 0 \leq y < h, 0 \leq z < k\}$ ， $y, z$  均为整数，表示一张矢状面图像中所有的像素点。进一步令  $X = \{x_0 | 0 \leq x_0 < w\}$ ，故  $S_l(X)$  表示整个矢状面序列图像。

三维体素数学表示：定义  $V_l(x, y, z)$ ，其中  $0 \leq x < w$ ， $0 \leq y < h$ ， $0 \leq z < k$ ，表示如图2所示三维空间中点  $(x, y, z)$  处的灰度值，其等价于  $T_l(z)$  这张水平面图像中  $y$  行  $x$  列位置处像素点的灰度值， $C_l(y)$  这张冠状面图像中  $z$  行  $x$  列位置处像素点的灰度值， $S_l(x)$  这张矢状面图像中  $z$  行  $y$  列位置处的灰度值。

分割序列图像数学表示：令  $T_g(Z)$  表示对整个水平面序列图像  $T_l(Z)$  的分割序列图像， $C_g(Y)$  表示对整个冠状面序列图像  $C_l(Y)$  的分割序列图像， $S_g(X)$  表示对整个矢状面序列图像  $S_l(X)$  的分割序列图像，在分割序列图像中使用1表示对颈椎椎骨的分割结果，颈椎椎骨以外的其他组织或器官使用0表示；并且假定  $T_g(Z)$ 、 $C_g(Y)$ 、 $S_g(X)$  三个方位的序列图中存储的总数据始终相等，因为它们只是从不同的方位进行切片得到的不同表示。 $V_g(x, y, z)$  的定义与上文  $V_l(x, y, z)$  类似，故不再赘述。

本文分割颈椎间盘的整体思路如图2所示。

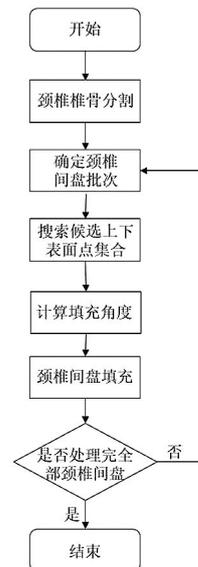


图2 算法流程框图

Fig.2 Algorithm flow block diagram

颈椎间盘分割步骤如下：(1)颈椎间盘是相邻两块颈椎椎骨之间的部分，因此颈椎间盘的边界应该由相邻的颈椎椎骨确定。颈椎椎骨的分割是本文所提方法的第一步。(2)确定一个颈椎间盘的批次有利于该颈椎间盘的定位。(3)在给定的颈椎间盘批次中搜索颈椎间盘候选的上表面点集和候选的下表面点集。(4)若一个候选上表面点沿着一定角度能够在对应的候选下表面点集中找到一个候选下表面点，那么这两点所成“线段”之间的部分就是颈椎间盘。(5)重复步骤(2)、步骤(3)和步骤(4)，即完成对人体颈椎CT序列图像中颈椎间盘的分割。

## 2.1 颈椎椎骨阈值分割

颈椎间盘是相邻两块颈椎椎骨之间的软组织，所以颈椎间盘的边界应该由相邻的颈椎椎骨确定。颈椎椎骨的分割是本文所提方法的第一步。如图3所示，在矢状面序列图像  $S_f(X)$  和冠状面序列图像  $C_f(Y)$  中，颈椎椎骨形态特征更明显，故可以选择在  $S_f(X)$  或  $C_f(Y)$  进行颈椎椎骨分割；同时，由于颈椎椎骨的灰度值与其他部分差异明显，因此本文选择手动调节分割阈值至最佳，对矢状面序列图像  $S_f(X)$  采用相同的阈值进行颈椎椎骨的分割得到  $S_G(X)$ 。

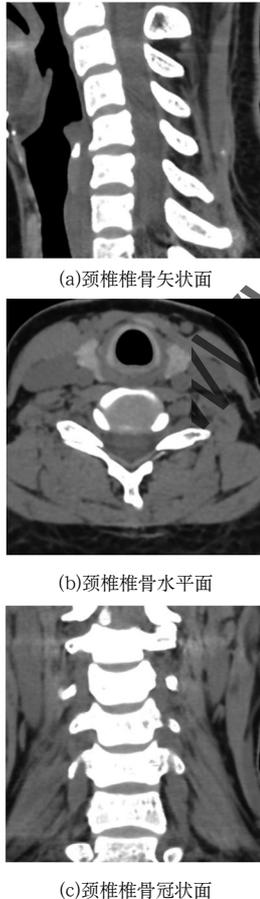


图3 颈椎椎骨示意图

Fig.3 Diagram of cervical vertebrae

为了后续能够得到更好的颈椎间盘分割效果，在分割颈椎椎骨时应尽可能地只分割包含颈椎间盘那一部分的颈椎椎骨，分割结果如图4(a)所示。由于颈椎椎骨上“孔洞”的存在，因此分割结果图上也存在一些“孔洞”，本文采用形态学中的孔洞填充算法<sup>[8]</sup>对其进行处理。对  $S_G(X)$  进行填充操作，消除像素点个数少于100的“孔洞”，填充结果如图4(b)所示。

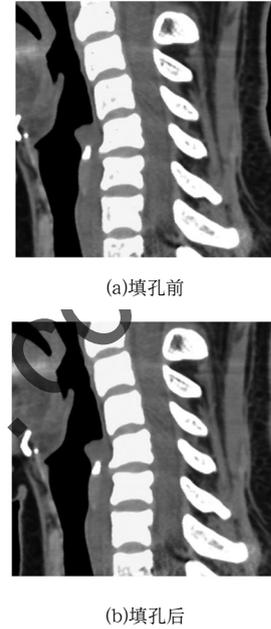


图4 颈椎椎骨分割结果示意图

Fig.4 Diagram of cervical vertebrae segmentation results

## 2.2 颈椎间盘批次的确定

对于每一个颈椎间盘而言，至少可以找到一个包含该颈椎间盘完整信息的  $T_G(Z)$  子集。不妨记包含颈椎间盘  $D_i (1 \leq i \leq 6)$  完整信息的一个  $T_G(Z)$  子集为  $T_G(Z_i)$ ，其中  $Z_{i1} \leq Z_i < Z_{i2}$ ，确定颈椎间盘批次即等价于确定与  $D_i$  对应的  $Z_{i1}, Z_{i2}$ 。在矢状面的分割序列图  $S_G(X)$  中选择形态良好的一张图像  $S_G(x_0)$ ，通常可以选择  $S_G(w/2)$ 。由图1可知， $S_G(x_0)$  的行数方向和  $z$  正轴方向一致，故我们可以通过  $S_G(x_0)$  的行数确定  $Z_{i1}, Z_{i2}$ 。

图5中每一节颈椎椎骨上的小点表示该节椎骨  $C_i$  点的平均位置  $\bar{P}_i$ ，其中  $N_i$  为属于  $C_i$  的像素点的数量。

$$\bar{P}_i = \frac{1}{N_i} \sum_{p \in C_i} p = (\overline{Col}_i, \overline{Row}_i) \quad (1)$$

一种直观的确定  $Z_{i1}, Z_{i2}$  的方式如下：

$$Z_{i1} = \overline{Row}_i, \quad Z_{i2} = \overline{Row}_{i+1} \quad (2)$$

即用与该颈椎间盘相邻的两块颈椎椎骨的平均位置的行数确定在  $T_G(Z)$  中与该颈椎间盘相关联的部分，这也是本文使用的方法。

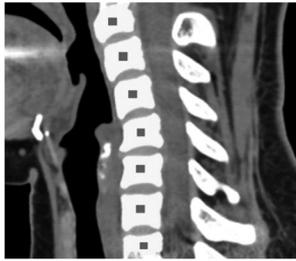


图5 颈椎椎骨中心位置示意图

Fig.5 Central position diagram of cervical vertebrae

### 2.3 候选上下表面点集合的搜索

在本文“2.2”部分中已经确定了与每一个颈椎间盘  $D_i (1 \leq i \leq 6)$  相关的水平面分割图像  $T_G(Z_i)$ ，其中  $Z_{i1} \leq Z_i < Z_{i2}$ ，接下来需要在  $T_G(Z_i)$  中搜索  $D_i$  候选的上表面点和下表面点。候选的上下表面点定义如下：

若  $p \in \{p(x, y, z) | 0 \leq x < w, 0 \leq y < h, Z_{i1} \leq z < Z_{i2}, V_G(x, y, z) = 1, V_G(x, y, z + 1) = 0\}$ ，则称  $p$  为  $D_i$  候选的上表面点。

若  $p \in \{p(x, y, z) | 0 \leq x < w, 0 \leq y < h, Z_{i1} \leq z < Z_{i2}, V_G(x, y, z) = 1, V_G(x, y, z - 1) = 0\}$ ，则称  $p$  为  $D_i$  候选的下表面点。

使用一个大小与水平面图像大小一致的矩阵  $US_i$  存储  $D_i$  候选的上表面点集合，其中  $US_i$  的每一个元素都是一个二元组，该二元组拥有两个字段，其中字段  $flag1$  取值为真或假。  $flag1$  为假，表示  $US_i(x_0, y_0)$  处不存在  $D_i$  候选的上表面点，  $flag1$  为真表示  $US_i(x_0, y_0)$  处存在  $D_i$  候选的上表面点，其具体位置由字段  $flag2$  指出；若字段  $flag1$  为真，则表明点  $p(x_0, y_0, flag2)$  为  $D_i$  的一个候选上表面点。在确定  $US_i(x_0, y_0)$  处是否存在  $D_i$  候选的上表面点时，本文采取的措施是由小到大依次遍历所有可能的  $z_0$ ，其中  $Z_{i1} \leq z_0 < Z_{i2}$ ，判断  $V_G(x_0, y_0, z_0)$  与  $V_G(x_0, y_0, z_0 + 1)$  是否满足上述候选上表面点的定义。获得  $US_i$  的伪代码如下。

```

Algorithm 1 Search candidate set of upper surface points algorithm
Input: Horizontal segmentation sequence batches  $T_G(Z_i)$ 
Output: Candidate set of upper surface points  $US_i$ 
1: Initialize  $w, h, Z_{i1}, Z_{i2}$  via  $T_G(Z_i)$ ; Initialize the matrix  $US_i$  with  $w, h$  and then initialize  $flag1$  of each element in  $US_i$  to false and  $flag2$  to -1.
2: for  $x = 0 \rightarrow w$  do
3:   for  $y = 0 \rightarrow h$  do
4:     for  $z = Z_{i1} \rightarrow Z_{i2}$  do
5:       Let  $value1 = V_G(x, y, z), value2 = V_G(x, y, z + 1)$ ;
6:       if  $value1 = 1 \ \& \ value2 = 0$  then
7:         Update  $US_i$ , set  $US_i(x, y).flag1 \leftarrow true$  and  $US_i(x, y).flag2 \leftarrow z$ ;
8:         break
9:       end if
10:    end for
11:  end for
12: end for
    
```

图6 获得  $US_i$  的伪代码

Fig.6 Pseudocode for  $US_i$

同理，使用  $LS_i$  存储  $D_i$  候选的下表面点集合，其大小与获得方式与  $US_i$  类似，唯一不同之处在于需要由大到小依次遍历所有可能的  $z_0$ ，其中  $Z_{i1} \leq z_0 < Z_{i2}$ ，故不再赘述。  $D_4$  的候选上下表面点集合在矢状面的结果如图7所示。此处为了示意，在描绘上下表面时有意将其厚度设置为5个像素点。从图7中可以看到，通过上述方式寻找的上下表面点不够精确，这也

是本文将其称作候选上下表面点的原因所在。

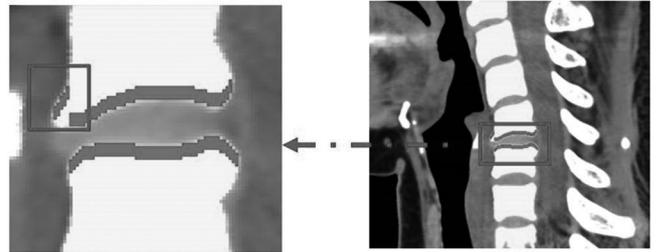


图7  $D_4$  候选上下表面矢状面示意图

Fig.7 Sagittal plane diagram of candidate upper and lower surfaces of  $D_4$

### 2.4 颈椎间盘填充角度的计算

在上文中已经得到  $D_i$  候选的上表面点集和  $US_i$  及候选的下表面点集和  $LS_i$ ，为了能够得到真正的上下表面点集和，本文采取的策略如下：假设  $US_i(x_0, y_0)$  处  $flag1$  为真，  $flag2$  值为  $z_0$ ，则表明点  $p_u(x_0, y_0, z_0)$  为一个候选的上表面点，并且称  $p_u(x_0, y_0, z_0)$  为  $US_i$  中的一个有效点。如果  $p_u(x_0, y_0, z_0)$  沿着某一个角度能够在  $LS_i$  中找到一个与之对应的候选的下表面点  $p_l(x'_0, y'_0, z'_0)$ ，则判定  $p_u(x_0, y_0, z_0)$  为真正的上表面点，  $p_l(x'_0, y'_0, z'_0)$  为真正的下表面点。对  $US_i$  中所有的有效点进行上述操作，以得到真正的上表面点集和真正的下表面点集。接下来探讨如何确定上述所说的角度。

根据  $x_0, y_0$  与  $x'_0, y'_0$  的关系可以分为如下四种情况进行讨论。

(1) 当  $x_0 = x'_0, y_0 = y'_0$  时，对应情况如图8所示，此种对应关系会导致后续颈椎间盘填充时矢状面和冠状面都是垂直向下填充，但在实际应用中颈椎间盘在矢状面一般都具有一定的倾斜角度。

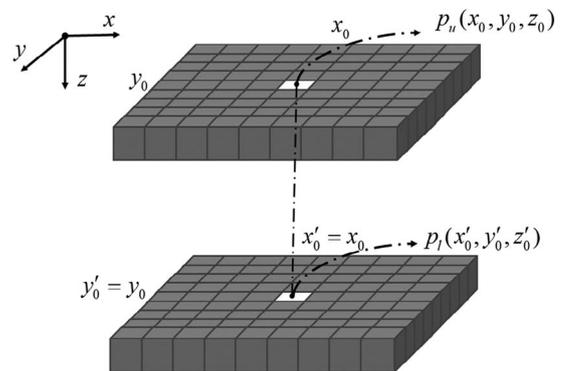


图8 当  $x_0 = x'_0, y_0 = y'_0$  时的对应情况示意图

Fig.8 Diagram of the corresponding situation when  $x_0 = x'_0, y_0 = y'_0$

(2) 当  $x_0 \neq x'_0, y_0 = y'_0$  时，对应情况如图9所示，此种对应关系会导致后续颈椎间盘填充时矢状面为垂直向下填充，冠状面为倾斜填充，与实际情况不吻合。

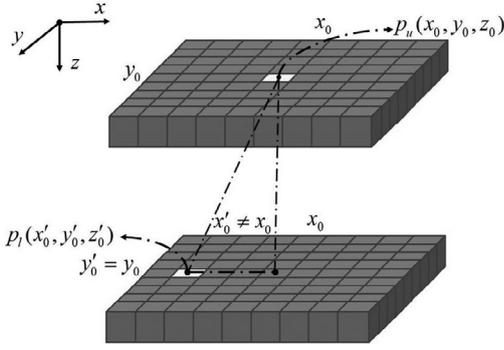


图9 当  $x_0 \neq x'_0, y_0 = y'_0$  时的对应情况示意图

Fig.9 Diagram of the corresponding situation when

$$x_0 \neq x'_0, y_0 = y'_0$$

(3)当  $x_0 = x'_0, y_0 \neq y'_0$  时, 对应情况如图10所示, 此种对应关系会导致后续颈椎间盘填充时矢状面为倾斜填充, 冠状面为垂直向下填充, 与颈椎间盘的真实情况较为一致。

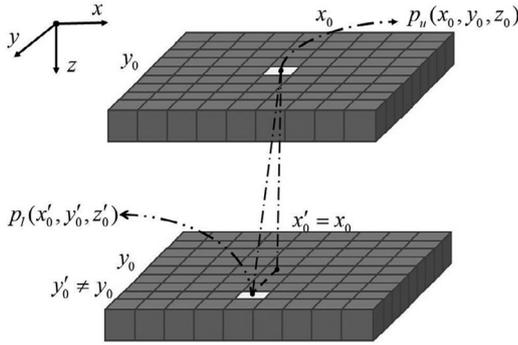


图10 当  $x_0 = x'_0, y_0 \neq y'_0$  时的对应情况示意图

Fig.10 Diagram of the corresponding situation when

$$x_0 = x'_0, y_0 \neq y'_0$$

(4)当  $x_0 \neq x'_0, y_0 \neq y'_0$  时, 对应关系如图11所示, 此种对应关系会导致后续颈椎间盘填充时矢状面和冠状面都为倾斜填充。

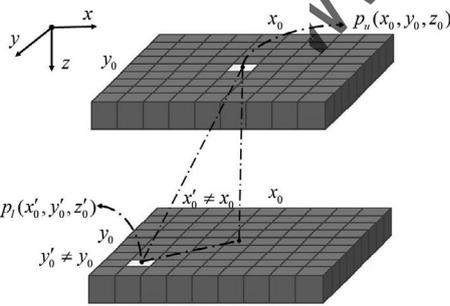


图11 当  $x_0 \neq x'_0, y_0 \neq y'_0$  时的对应关系示意图

Fig.11 Diagram of the corresponding situation when

$$x_0 \neq x'_0, y_0 \neq y'_0$$

综合对比以上四种对应关系可知: 第一和第二种对应关系与实际情况不太一致; 第四种情况所示的对应关系是最理想的对应关系, 但是不方便求解填充角度; 第三种情况所示的对应关系与实际情况较为吻合, 并且易于求解填充角度, 故本文选择第三种情况的对应关系, 即当  $x_0 = x'_0, y_0 \neq y'_0$  时。

对于第三种情况的对应关系, 可以直接在矢状面序列图  $S_G(X)$  中进行填充角度的求解。

在本文“2.2”部分中已经求解每一节颈椎椎骨  $C_i$  平均位置  $\bar{P}_i$ , 如公式(1)和图5所示。为了得到颈椎间盘  $D_i$  的填充角度  $\theta_i$ , 本文选择令  $\theta_i$  为  $\bar{P}_i$  与  $\bar{P}_{i+1}$  两点连成的线段与矢状面列方向所呈现的夹角, 如公式(3)和图12所示。

$$\theta_i = \arctan \left( \frac{\overline{Row_{i+1}} - \overline{Row_i}}{\overline{Col_{i+1}} - \overline{Col_i}} \right) \quad (3)$$

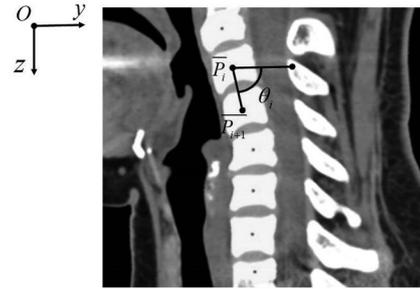


图12 填充角度示意图

Fig.12 Diagram of filling angle

事实上, 第三种情况的对应关系已经把与候选的上表面点  $p_u(x_0, y_0, z_0)$  对应的下表面点  $p_l(x'_0, y'_0, z'_0)$  的  $x'_0$  确定了且  $x'_0 = x_0$ 。因此, 对于  $US_i$  中候选上表面点  $p_u(x_0, y_0, z_0)$ , 可直接在  $LS_i$  的  $x_0$  列中寻找是否存在有效的候选下表面点  $p_l(x_0, y'_0, z'_0)$ , 使得两点连成的线段与矢状面列方向的夹角“等于”  $\theta_i$ ,  $p_l(x_0, y'_0, z'_0)$  若存在, 则判定  $p_u(x_0, y_0, z_0)$  为真正的上表面点,  $p_l(x_0, y'_0, z'_0)$  为真正的下表面点。在实际实现时, 并不需要形成的夹角严格等于  $\theta_i$ , 因为  $\theta_i$  值本身就是一个估计值。

### 2.5 基于DDA的颈椎间盘填充方法

假设  $p_u(x_0, y_0, z_0)$  为真正的上表面点,  $p_l(x_0, y'_0, z'_0)$  为真正的下表面点, 那么  $p_u(x_0, y_0, z_0)$  与  $p_l(x_0, y'_0, z'_0)$  两点之间非颈椎椎骨的体素点属于颈椎间盘  $D_i$ 。为了得到  $p_u(x_0, y_0, z_0)$  与  $p_l(x_0, y'_0, z'_0)$  两点连成“线段”上的体素点, 本文使用数值微分法(Digital Differential Analyzer, DDA), 其算法伪代码<sup>[9]</sup>如图13所示。

```

Algorithm 2 DDA algorithm
Input: Upper surface point  $p_u$ , Lower surface point  $p_l$ .
Output: All the cervical disc voxel  $V$  between  $p_u$  and  $p_l$ .
1: Initialize  $x_0 = p_u.x$ ; Initialize  $steps = \max(|p_u.y - p_l.y|, |p_u.z - p_l.z|)$ ; Initialize  $\Delta_y = \frac{p_l.y - p_u.y}{steps}$ ;
   Initialize  $\Delta_z = \frac{p_l.z - p_u.z}{steps}$ .
2: for  $i = 0 \rightarrow steps$  do
3:   Let  $val = V_G(x_0, [y], [z])$ ;
4:   if  $val = 0$  then
5:      $V.push(x_0, [y], [z])$ ;
6:      $y \leftarrow y + \Delta_y$ ;
7:      $z \leftarrow z + \Delta_z$ ;
8:   end if
9: end for
    
```

图13 DDA算法伪代码

Fig.13 DDA algorithm pseudocode

接下来只需要对  $US_i$  中每一个有效点  $p_u$  重复上述操作, 然后将得到的  $V$  中的每一个点标记为颈椎间盘, 即实现了颈椎间盘的分割。分割结果的矢状面、水平面及冠状面如图14所示。

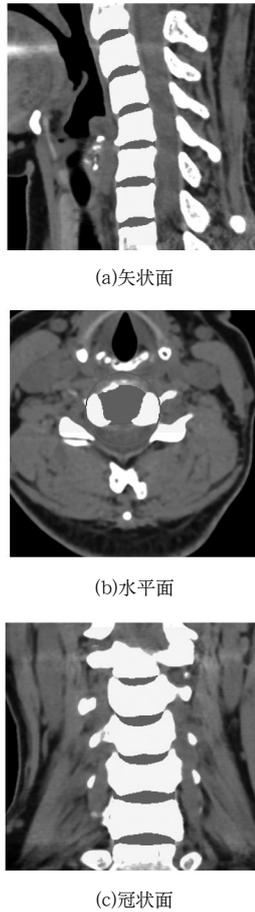


图14 分割结果示意图

Fig.14 Diagram of segmentation results

### 3 实验结果及对比分析(Experimental results and comparative analysis)

#### 3.1 评价指标

为了定量分析本文提出的算法对于CT图像中颈椎间盘的分割性能，采用像素级评价指标进行评估，包括像素准确率(Pixel Accuracy, PA)、精准率(Precision)和召回率(Recall)。按照图像分割结果和人工标注结果，可以将图像像素分为4个类别，如表1所示。

表1 图像像素类别

Tab.1 Image pixel category

分割	标注	
	颈椎间盘	非颈椎间盘
颈椎间盘	TP	FP
非颈椎间盘	FN	TN

依据图像像素分类，将评价指标定义如下。

像素准确率：

$$Acc_{PA} = (TP + TN) / (TP + FP + FN + TN)$$

精准率：

$$Pre = TP / (TP + FP)$$

召回率：

$$Rec = TP / (TP + FN)$$

其中， $Acc_{PA}$ 反映了像素级整体分割准确度； $Pre$ 表示既被标注为颈椎间盘又被分割为颈椎间盘像素点占有所有被分割为颈椎间盘像素点的比例，可反映目标误检率； $Rec$ 表示既被标注为颈椎间盘又被分割为颈椎间盘像素点占有所有被标注为颈椎间盘像素点的比例，可反映目标漏检率。由于评价的对象是多组序列图，故先计算每一组序列图准确率、精准率和召回率的均值，然后对多组序列图求均值。

#### 3.2 实验设计与结果分析

为了验证本文提出的算法在CT序列图中进行颈椎间盘分割的有效性，以下将针对双阈值分割、大津阈值分割及本文所提算法进行分割对比实验，并对实验结果进行分析。

双阈值分割是对某张图像进行人工交互，确定分割阈值范围，之后所有序列图像都根据这一固定的阈值范围进行分割。大津阈值分割又称最大类间方差法(OTSU)，是对每一张图像分别计算最佳分割阈值，使得按照灰度特性划分的背景相与目标相类间方差最大，从而使得二者错分概率最小。这两种方法都是在图像分割领域应用较成熟的方法。将这两种方法与本文提出的算法进行实验效果对比。此外，为了验证算法的普适性，随机挑选了10组病人的颈椎CT数据作为测试数据集，并将其中一组测试结果进行展示。该组颈椎CT序列图矢状面的分辨率为 $435 \times 349$ ，共435张。选择矢状面分割结果的第180张、第200张、第220张为代表图像，三种算法的分割结果如图15所示，其中图15第一列是矢状面原始图像，图15第二列是人工标记图，图15第三列是大津法分割结果图，图15第四列是双阈值法分割结果图，图15第五列是本文提出方法的分割结果图。

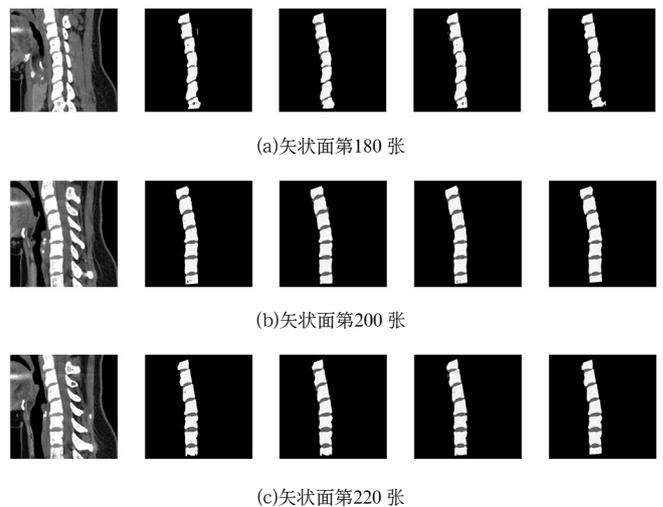


图15 分割结果对比

Fig.15 Comparison of segmentation results

(下转第51页)