

文章编号:1005-3085(2002)05-0022-07

管道切片的三维重建

廖武鹏, 邓俊晔, 王 丹

指导教师:数模教练组

(河海大学,南京 210098)

编者按:该论文根据问题以离散形式给出数据而所求轴心轨迹及切片轮廓实质是连续曲线的特点,并充分利用生成球的某个位置在上、下半径距离范围内的切片都有截点的规律,就确定边界点及求切片最大内切圆半径提出连续算法。在后一种算法中还讨论了某切片最大圆心实际计算中出现的不唯一的情况下如何筛选的问题。

摘 要:文中证明了所有切片含有过轴心的大圆,该大圆直径一定与切片边界相交。通过构造连续型模型和离散型模型,从 0 BMP 中定出轴心为(0,160)和半径为 30 的最大圆,并相继在其它切片中运用最大圆必包含在切片中的先决条件,找出相应切片中所有可能的轴心坐标,进一步对每一切片待选的轴心坐标,根据其球体必在上 29~下 29 层切片中存在相应半径的圆(在上下 29 层中存在半径为 7.68,在 24 层存在半径为 18 的该球体的相应的截面圆)的特征,筛选上述待选轴心坐标,比较准确地定出了 0 到 70 层的轴心坐标。对于 71 至 99 层由于上 29 层的信息不全,还存在不少待选点,再应用切片尖端特性(在 70 层左下角的点只能由半径较小的圆包络而成,由此定出 99 层的轴心坐标)确定其余切片的轴心坐标。绘制出的三维图形和各坐标面的投影图是光滑流畅的。最后文中用所得轴心坐标重新构造各切片,与原切片比较,相异像素点误差不足 3%,结果令人满意。

关键词:连续模型;离散模型;尖端特性

分类号:AMS(2000) 65D17

中图分类号:O242.1

文献标识码:A

1 问题的重述(略)

2 模型的假设(略)

3 问题的分析

问题第一部分需要求出管道的中轴线方程和半径,第二部分需要绘制中轴线在各个平面上的投影。

对于第一个问题,由于一张切片的厚度为 1 个单位,故切片间的中轴线可以做线性化处理,即用一条线段代替。基于 100 张切片所提供的信息,利用计算机搜索球体的半径,并找到每张切片上中轴线与切片交点的坐标,称交点坐标为轴心坐标。利用这些坐标,求出两张切片间中轴线方程。将 100 个轴心连接起来,形成一条完整的空间曲线。

对于第二个问题,由得出的中轴线方程,求出每段中轴线段在 XY、YZ、ZX 平面上的投影(一段线段),综合每一段投影即可得中轴线在三个坐标面上的投影(将每段投影连接起来)。

考虑到实际图象边界上的点是连续的,只有位置而没有大小,且点的位置以坐标(任何实

数对)来表示。在转换成BMP格式图象时,象素表示的图象边界是离散的,一般成锯齿状,图象范围与实际图象有误差,包括图象转换的系统误差,即点取舍引起的误差和计算的舍入误差。

针对BMP图象,我们试图就圆这一特定的图形,反演图象变换,消除系统误差,找出实际图形理论上的圆心和半径等。就一般平面图形可根据现有计算机图形生成技术,力求较好地再现原始图形的特征指标。由于所给问题数据是离散的,我们可以将所求中轴线看成是象素坐标上的点,但问题本身是连续的,进一步将连续点的处理与离散的象素图形综合考虑,求出相应的轴心坐标,并检查所求结果与所给图形的误差大小。

4 模型的建立

从几何特征,先明确下面重要的结论:

定理1 任一切片的边界必是滚动球在切片处相应的截面图所形成的包络图。

由定理1,我们可得下面定理2:

定理2 任意切片 Z (Z 表示切片的高度),一定包含球心在 $Z \pm 1, Z \pm 2, \dots, Z \pm 29$ 位置上球体在切片处半径分别为 $\sqrt{R^2 - i^2}$ ($i = 1, 2, \dots, 29$)的截面圆(这里 R 设为球体半径)。

题目中存在的不确定性分析:

注意到问题所给信息不能排除下述可能:直径一定的球体可以产生比球体直径大得多的切片。如管道是由球心沿竖直方向上升较慢的螺旋线的球滚动而成的。在此情形下,实际无法确定中轴线坐标与球半径。针对所求问题,不妨设中轴线在水平方向的螺旋绕动变化相对竖直方向变化较慢,使得任一切片最大内含圆的半径与滚动球半径相同。最大内含圆的圆心即为切片与中轴线的交点。

根据题目中欲求的管道半径,设管道横截面是一个圆,其半径与球体半径相同,则有:

定理3 若切片与中轴线有交点,且管道的横断面是圆,则该切片必含有半径与球体相同的最大圆,且圆心在交点处。

由上述假设可建立第一个模型。

模型一、连续模型

为了寻找球体的半径,需要对100张BMP格式的图象文件进行搜索,这里涉及到一个对BMP格式文件的处理问题。由于BMP格式文件在计算机中是以二进制数进行存储的。图象保存在一个二维的由0或1组成的矩阵中。0和1分别对应于图象中的黑白象素。一张BMP格式的文件包含了 512×512 个象素,每一个象素都有自己的一个确定的坐标。一个16进制的数代表4个二进制位,每个二进制位可以记录一个象素。为方便使用数据,我们将16进制的文件编程转化为由0和1组成的文本文件,然后利用该文本文件计算每张图片上管道与切片的交所形成的边界点坐标。

在说明计算边界点坐标方法之前,需引进图象处理技术中的四邻域概念:

四邻域:某个象素的左、右、上、下四个象素称为该象素的四邻域,如图1,象素 D, B, F, H 称为象素 E 的四邻域。

具体寻找边界点坐标的算法:由于我们将图象信息用0和1两个不同的灰度值表示黑白象素。对图象进行逐行搜索,当遇到灰度值为零的象素点时,再搜索其四邻域内的点。若在其四邻域中有一个象素的灰度值为1,则该点为边界点。将每张切片的边界点坐标保存在一个二维数组中,为求解半径所用。

寻找球体半径的算法:由于每张切片图象有一组确定的边界坐标。每个被截的球体在切片上的投影均是圆,在这些圆中过球心的圆的半径最大。图象的边界是由这些大大小小的圆包络而形成的。不妨以 50. BMP(具有一般性)为例说明问题。

第 i 张切片的边界点坐标保存在数组 $x[1], x[2], \dots, x[n]; y[1], y[2], \dots, y[n]$ 中。

取

$$\begin{aligned} \min x &= \min\{x[1], x[2], \dots, x[n]\}, \\ \max x &= \max\{x[1], x[2], \dots, x[n]\}, \\ \min y &= \min\{y[1], y[2], \dots, y[n]\}, \\ \max y &= \max\{y[1], y[2], \dots, y[n]\}. \end{aligned}$$

以 $\min x, \max x, \min y, \max y$ 为边界作矩形区域 D , 在 D 的内部逐行搜索如果遇到一个值为 0 的点, 再搜索其四邻域的点, 如果四邻域内所有像素点所对应的值为 0, 则该点一定是边界区域所包围内部图形的点, 称之为内点。设内点为 A , 以其为圆心作一个半径为 R 的圆 (R 可以从零开始递增), 如果边界上所有的点都不落在该圆内, 记录下圆心的坐标, 并继续增加半径 R 直到有一部分边界的点落入圆内为止。最后寻找半径最大的圆 (该圆必与边缘相切), 与边缘相切的最大圆即是过球心的最大圆。

用 Pascal 语言编程计算出每张切片上的中轴线与切线交点的坐标以及该切面上最大圆的半径的范围。此处给出前面几个切片的交点坐标和半径的范围。

表 1 轴心的坐标与半径

切 片	0	1	2	3	4	5
半径范围	29.8	(29.7, 29.9)	(29.7, 29.9)	(29.8, 29.9)	(29.8, 29.9)	(29.8, 29.9)
坐 标	(0.1, 160.5)	(0.1, 160.2)	(0.1, 159.8)	(0.1, 159.9)	(0.1, 160.1)	(0.1, 160.2)

模型二、离散方法

由前定理知, 所有切片中均含有半径为 R 的最大圆。对 0. BMP 所对应的切面, 沿 X 轴向在其内部取定坐标 (这里选用题目所给的像素坐标, 即几何坐标位置理解为像素坐标的中心)。用通用 Bresenham 算法计算出某一给定圆的边界坐标点, 判断其是否含于切片中。根据电脑画图的特点, 这里选用的半径、坐标都为整数。用这种方法处理像素文件可以克服由原实际图象转换到 BMP 数据图象的系统误差。求出所有最大圆的圆心坐标, 找出最大半径 R 。

利用计算机图形学中的 Bresenham 算法作半径为 R 的圆的 BMP 格式图象。

具体算法如下:

1) 初始像素点取 $x_0 = 0, y_0 = R$, 取 $f_0 = 1, g_0 = 2 \times (1 - R)$

2) 对 $i = 0$ 到 $\left[\frac{R}{\sqrt{2}} + \frac{1}{2}\right] - 1$, 有 $x_{i+1} = x_i + 1$, 且

若 $|f_i| \leq |g_i|$, 则 $y_{i+1} = y_i, f_{i+1} = f_i + 2x_i + 3, g_{i+1} = f_{i+1} - 2y_i + 1$,

若 $|f_i| > |g_i|$, 则 $y_{i+1} = y_i - 1, f_{i+1} = g_i + 2x_i + 3, g_{i+1} = f_{i+1} - 2y_i + 3$,
(x_{i+1}, y_{i+1}) 即为圆边界像素点坐标。

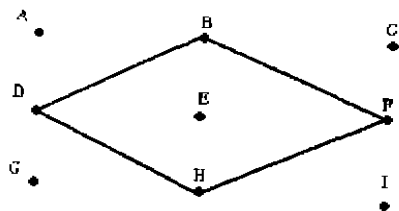


图 1 像素 E 的四邻域

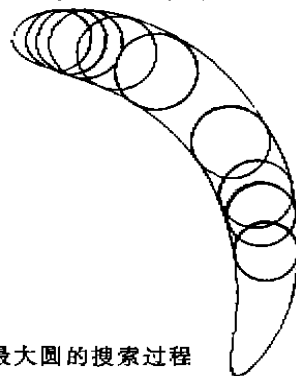


图 2 最大圆的搜索过程

利用 WINDOWS 画图软件 Paintbrush 生成半径为 30 的圆,并与由 Bresenham 算法所作半径为 30 的圆进行比较,发现两者边缘的象素点重合,见图 3。我们有理由相信,切片上的图象(一系列圆叠加而成)也是利用这种方法生成的。

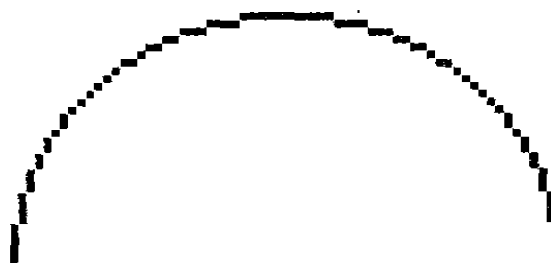


图 3a Paintbrush 生成的图

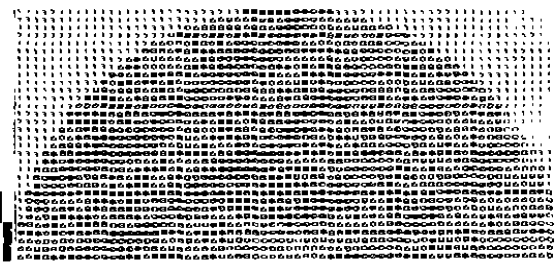


图 3b Bresenham 算法作的图

寻找最大圆算法:对 0.BMP 图像运用 Bresenham 算法计算出最大内含圆半径是 30,圆心为(0,160)。由于采用了离散型的模型,容许一个象素的误差,因此 30 是可以接受的。以此为基础去寻找 1.BMP 到 99.BMP 中的最大圆。为了加快搜索的速度,对图象进行逐行扫描时,记录扫描线与图象边缘的最左边与最右边交点。它们的横坐标分别为 $x[i]$, $x[j]$ 。这两个边缘点之间的距离 $d = |x[j] - x[i]|$ 。如果 $d < 60$ 则停止搜索,做下一行的扫描。如果 $d \geq 60$,则判断横坐标介于 $x[i]$ 与 $x[j]$ 之间的所有点是否为最大圆的圆心。判断方法是以这些点为圆心,以 30 为半径作一圆周,检测边缘象素点是否在圆周以内,若是则此点不是最大圆的圆心坐标,否则该点为最大圆的圆心坐标。这些圆心的坐标为待选的轴心坐标,需对其进行筛选,筛选出的轴心坐标所对应的最大圆为最优圆。

扫描线与边缘相交时可能会遇到如图 4 所示的情况。扫描线 1,2,3 与边缘的交点有 2 个、3 个和 4 个的情况。如扫描线 2 与边缘相交三次,则我们应该选择 B2 与 C2 之间的边缘点计算距离,扫描线 3 与边缘相交四次,应取 A3 与 B3、C3 与 D3 之间的点计算距离。

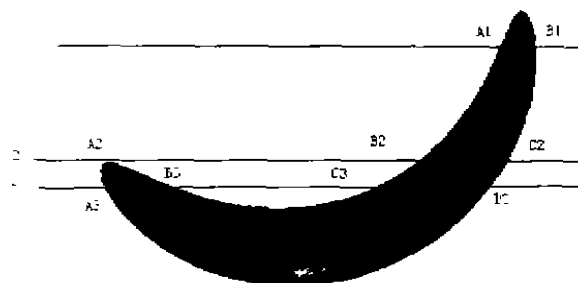


图 4 扫描线与切片边界相交情况

寻找最优圆算法:管道视为球体滚动包络而成,对于某一个确定的球体被下 29 ~ 上 29 张切片将其均匀分割。形成一系列的同轴圆。假设球心经过第 0 张切片,球顶部分被第 29 张切片所截,见图 5。最大圆的圆心与所截的第 29 个小圆圆心位于同一个轴上。最大圆的圆心坐标就是第 29 个小圆的圆心坐标。

如果找到第 29 个小圆在第 29 张切片上的位置,则最大圆圆心(即中轴线与切片的交点)便确定了。

下面阐述具体步骤:

对 0.BMP 到 99.BMP 均有多个待选轴心坐标,在第 Z 层中,以待选轴心坐标所作的球体被上下 i ($i = 1 \sim 29$) 张切片所截的小圆应在第 $Z \pm i$ 张切片中,若有任意一张切片所截的小圆不在第 $Z \pm i$ 张切片中,则该待选轴心坐标不可能是最优圆的轴心坐标。

0.BMP 到 70.BMP 图象用以上方法处理效果较好。对于 0 ~ 28 的几张切片一般只有三个待选轴心坐标需要进行筛选,虽然下方切片的信息不完整,但由于待选轴心坐标较少,完

全可以筛选出最优圆。但对于 71.BMP 到 99.BMP 的切片经筛选后仍有不少待选点,这是因为 71.BMP 到 99.BMP 的信息量不够,故被淘汰的最大圆不多。这时需应用尖端特性进行二次筛选。以 99.BMP 为例,注意到 70.BMP 切片左下角因不可能是轴心在第 98 层和第 42 层切片处的球体在 70 层截面小圆的包络,而是更小截面圆的包络,这个小圆只能来自第 99 和第 41 层中球体,经计算知第 41 层不存在这样的球体,故此小圆一定存在于轴心在第 99 层的球体中。

其他一些靠近第 99 层的切片也类似处理。表 2 为一部分切片的轴心坐标。

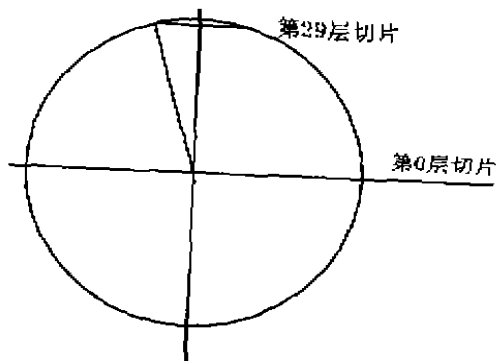


图5 截面圆与轴心的关系图

表2 部分轴心坐标

X	Y	Z
0	160	0
0	160	1
0	160	2
0	160	3
0	160	4
0	160	5
114	116	50
119	111	51
123	106	52

X	Y	Z
128	101	53
132	95	54
136	89	55
140	83	56
144	76	57
147	70	58
151	62	59
153	55	60
71	-170	90

X	Y	Z
65	-173	91
57	-176	92
51	-179	93
44	-181	94
37	-183	95
31	-185	96
24	-187	97
17	-188	98
12	-189	99

下面需要构造中轴线方程,在问题分析中我们提到过,由于每张切片的厚度很薄,考虑最简单的方法,在每段切片中的中轴线用直线段线性化表示。得到直线的点法式方程将这 99 个方程联立可得完整的中轴线的方程。

问题的第二部分要求绘出中轴线在 XY , YZ , ZX 三个坐标平面上投影,在每段中轴线的点法式方程中消去相应的变量,则得到该段中轴线在这个平面上的投影方程。分别联立每个坐标面上方程就构成了三个坐标面上的中轴线投影方程。下面是我们利用 EXCEL 所绘制的管道三维模型中轴线的投影图象。

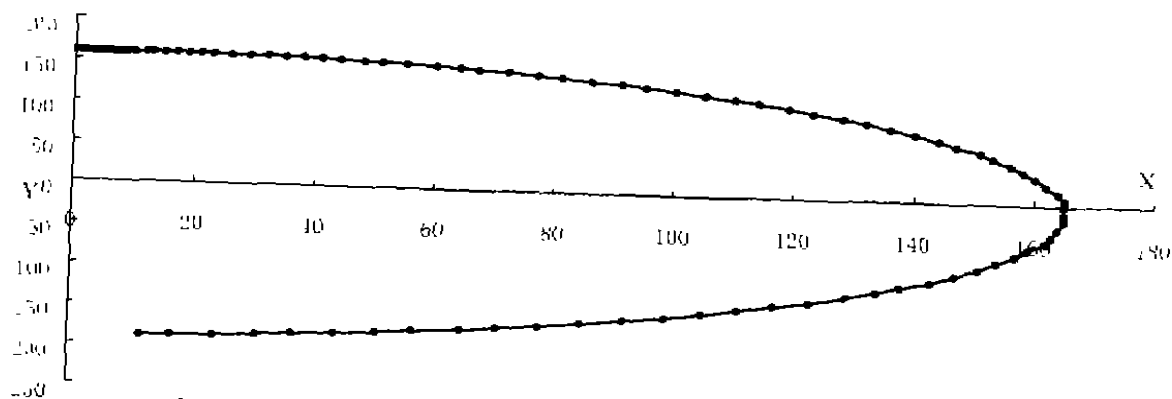


图6 轴线在 XY 平面上的投影图

5 模型的分析 and 检验

由所得中轴线坐标,可生成 30 ~ 70 层所有切片的图形,方法如下:

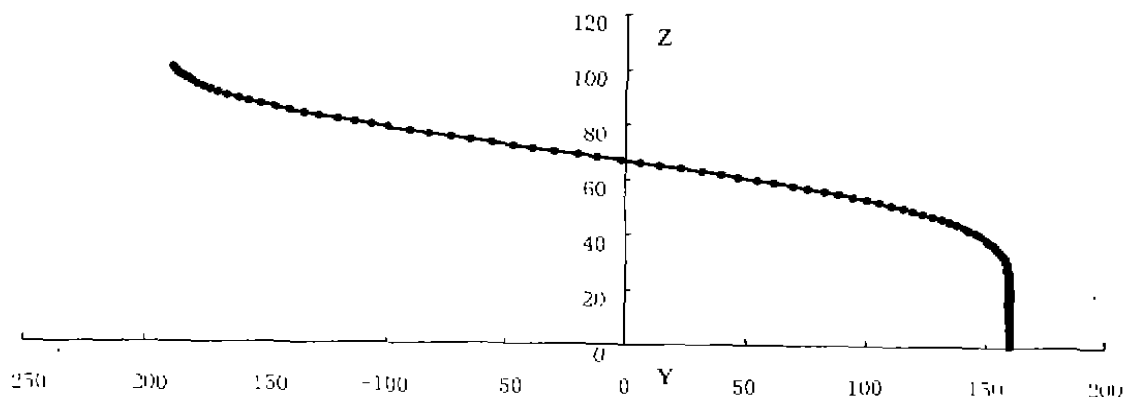


图7 轴线在 YZ 平面上的投影图

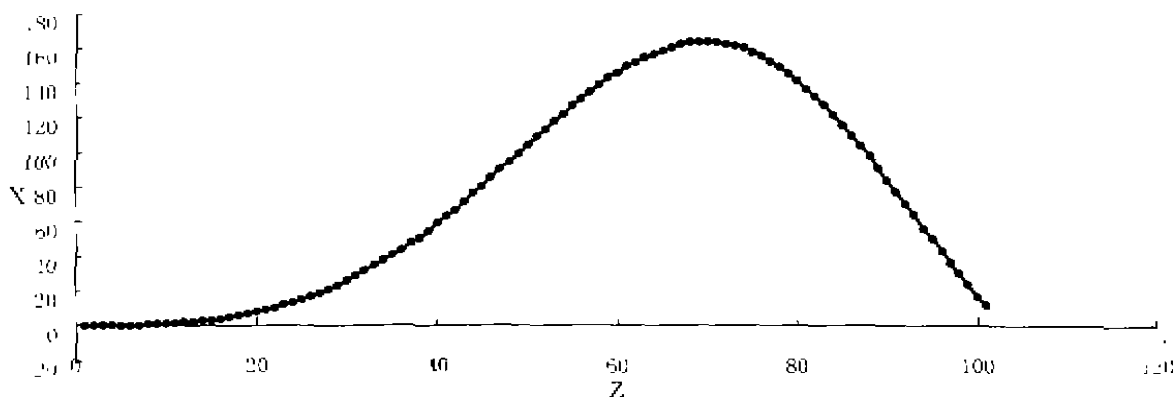


图8 轴线在 ZX 平面上的投影图

欲产生第 Z 层切片,由 $Z-29, Z-28, \dots, Z, Z+1, \dots, Z+29$ 相应轴心坐标计算出对应的所有圆的象素点,并将它们迭加便可形成第 Z 层切片。进一步,将所得图形与原切片作比较,计算相异点所占比例,经计算,第 40 层切片的相对误差为 3%。这说明我们的模型精度是较高的。

但是建立第一个模型是比较粗糙的。通过控制半径的步长大小,可以得到一系列半径非常接近的最大圆,以此我们能够给出对于一张切片上最大圆半径的一个范围,发现多数是靠近 30 这个理想半径的。出现这种问题的原因是由于切片上的图象是由离散的象素点表示的。不同方向的半径是不相同的,见图 9。由于问题中需要求解中轴线的方程,这条曲线在数学上是没有粗细且唯一确定的。因此对于中轴线与切片的交点应该是唯一的。但是用来表示交点的象素点经筛选后仍不唯一。可选这些象素点的几何中心的整数点为轴心坐标。

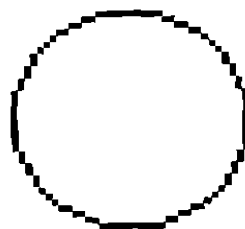


图 9a BMP 格式存储的圆

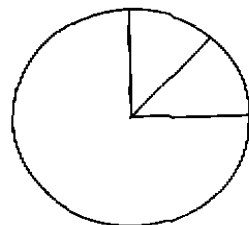


图 9b 理论圆

由于实际的图形转化为 BMP 格式图象文件时会造成图象的失真,正如前面提到过计算机是根据一定的算法去存储图象的,并且算法的对象是离散的点,这对于实际连续的实物来说误差不可避免,所以是一种系统误差。而模型二所采用的算法恰是对离散 BMP 象素点的处理,可以根本消除系统误差。

6 模型的推广

如果考虑中轴线与切片的交点不止一个,比如有两个交点,仍然可以利用我们的离散模型去求解,因为不含轴心的切片中一定不含有最大圆。

我们对于离散问题讨论较多,但对于轴心坐标及半径是实数的情况,没有过多的涉及。问题本身是连续的,用离散方法去模拟,不可避免地会出现误差。如果知道像素圆的生成算法,我们就可以运用本模型的思想,将模型推广到实数领域,从而使模型的精度提高。

参考文献:

- [1] 管伟光. 体视化技术及其应用[M]. 北京: 电子工业出版社, 1998
- [2] 孙家广, 扬长贵. 计算机图形学[M]. 北京: 清华大学出版社, 1995
- [3] Kenneth R. Castleman. 数字图象处理[M]. 北京: 电子工业出版社, 1981

3D Reconstruction of Pipeline Slice

LIAO Wu-peng, DENG Jun-ye, WANG Dan

Advisor: Mathematical Modeling Tutor Group

(Hohai University, Nanjing 210098)

Abstract: This article proves that there exists the biggest circle on each slice, whose center is on the pipeline center curve. By constructing the continuous model and the discrete one, we are able to determine the biggest circle on 0.bmp slice, of which radius is 30 and center coordinate is (0,160). Using the predetermined condition that the biggest circle should be contained in each slice, we find out all possible coordinates of the pipeline center curve on corresponding slices. For each slice, basing on the principle that a slice which lies on over 29~under 29 layers of the slice must contain a corresponding section of the sphere of radius 30, we filtrate the coordinates of the pipeline center curve and determine the more precise one. However, there are still some coordinates of the pipeline center curve left to be filtrated, such as slices from 71 to 99, due to the information is not enough. Further, we use the pointed end property to determine the coordinates of the pipeline center curve on other slices. The projecting figures on each coordinate plane are smooth and fluent. In the end, by using the coordinates to reconstruct all the slices and comparing the slices with the original ones, we found that the error of different pixels is less than 3%. The result is satisfied.

Key words: continuous model; discrete model; pointed end property