

CT 系统参数标定及成像问题的研究

摘要

CT 可以在不破坏样品的情况下获取样品内部的结构信息，CT 系统安装时往往存在误差，从而影响成像质量。因此探究对安装好的 CT 系统进行参数标定，并据此对未知结构的样品进行成像的问题具有重要意义。

本文针对 CT 系统参数标定及基于已标定参数的 CT 成像问题进行研究，主要解决了如何确定 CT 系统旋转中心在正方形托盘中的位置、CT 系统使用的 X 射线的所有方向以及在已知转轴中心的位置的情况下如何利用未知介质的接收信息复原介质图像的问题。

针对问题一，我们以 Beer-Lambert 定律为基础，根据几何关系，应用解析几何知识，建立了参数求解模型。计算可得探测器单元之间的距离 d 为 0.2768mm；以椭圆中心为原点，短轴为 X 轴，长轴为 Y 轴，向右和向上为正方向建系，计算可得旋转中心坐标为 (-9.6713, 6.3511)；以 X 轴为 0 度角，以逆时针旋转为正方向，建立极坐标系，计算可得 X 射线初始角度为 119.6774 度，之后的旋转过程为每次逆时针旋转几乎一度。

针对问题二，我们以 Radon 变换和中心切片定理为理论基础，采用滤波反投影法进行图像重建。并考虑误差（旋转中心不在图像几何中心），根据第一问求得的旋转中心坐标，采取平移法修正误差。而且由于滤波反投影法之后容易出现灰点误差，所以我们通过设定阈值过滤灰点。最终通过归一化把吸收率控制在 0—1 之间，得到的十个点坐标及其吸收率分别为：[(10,18): 0], [(34, 25): 0.6785], [(43.5,33): 0], [(45,75.5): 0.8067], [(48.5,55.5): 0.6984], [(50,75.5):0.8786], [(56,76.5):0.8703], [(65.5,37):0], [(79.5,18):0], [(98.5,43.5):0]。

针对问题三，基本原理与误差修正方法同问题二。但由于问题三种的图像基本覆盖了正方形托盘，旋转中心偏移导致复原图像经平移修正后图像数据部分缺失，所以我们采用旋转的方法在不同方向上获得全部图像，再集中拼接到初始角度图像上，取平均值，然后进行平移修正。就算出准确完整的复原图像。十个点坐标及其吸收率为：[(10,18): 0], [(34, 25): 0.0579], [(43.5,33): 0.4070], [(45,75.5): 0], [(48.5,55.5): 0.3020], [(50,75.5):0.1914], [(56,76.5):0.2058], [(65.5,37):0.2861], [(79.5,18):0.3364], [(98.5,43.5):0]。

针对问题四，我们通过第一问求解的误差分析，得出误差主要来源于椭圆与小圆接收信息重合的情况。我们采用四个小圆设计新模板，它们的圆心分别在正三角形的三个顶点和几何中心。并调整半径和正三角形尺寸，以保证各个投影方向总出现两组各自独立的接收信息。从而增加了参数标定的精度和稳定性。

关键字： 椭圆方程 Beer-Lambert 定律 matlab CT 成像 Radon 逆变换 滤波反投影重建算法

一、问题重述

1.1 引言

CT(Computed Tomography)可以在不破坏样品的情况下,利用样品对射线能量的吸收特性对生物组织和工程材料的样品进行断层成像,由此获取样品内部的结构信息。而CT系统安装时往往存在误差,从而影响成像质量,因此需要对安装好的CT系统进行参数标定,即借助于已知结构的样品(称为模板)标定CT系统的参数,并据此对未知结构的样品进行成像。

一种典型的二维CT系统如图1(a)所示平行入射的X射线垂直于探测器平面,每个探测器单元看成一个接收点,且等距排列。X射线的发射器和探测器相对位置固定不变,整个发射-接收系统绕某固定的旋转中心逆时针旋转180次。对每一个X射线方向,在具有512个等距单元的探测器上测量经位置固定不动的二维待检测介质吸收衰减后的射线能量,并经过增益等处理后得到180组接收信息。

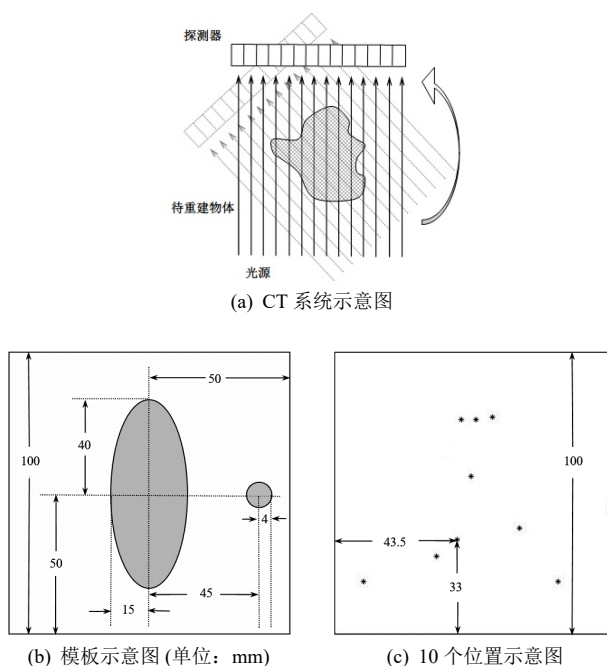


图1 问题中的图形

1.2 问题的提出

(1) 在正方形托盘上放置两个均匀固体介质组成的标定模板,模板的几何信息如图1(b)所示,相应的数据文件见附件1,其中每一点的数值反映了该点的吸收强度,这里

称为“吸收率”。对应于该模板的接收信息见附件 2。请根据这一模板及其接收信息，确定 CT 系统旋转中心在正方形托盘中的位置、探测器单元之间的距离以及该 CT 系统使用的 X 射线的 180 个方向。

(2) 附件 3 是利用上述 CT 系统得到的某未知介质的接收信息。利用 (1) 中得到的标定参数，确定该未知介质在正方形托盘中的位置、几何形状和吸收率等信息。另外，请具体给出图 3 所给的 10 个位置处的吸收率，相应的数据文件见附件 4。

(3) 附件 5 是利用上述 CT 系统得到的另一个未知介质的接收信息。利用 (1) 中得到的标定参数，给出该未知介质的相关信息。另外，请具体给出图 1(c) 所给的 10 个位置处的吸收率。

(4) 分析 (1) 中参数标定的精度和稳定性。在此基础上自行设计新模板、建立对应的标定模型，以改进标定精度和稳定性，并说明理由。

二、模型的假设

- 假设 CT 无故障，接收的信息是完全准确的数据
- 假设探测器单元足够小，可以看成是一个点并且在接收信号过程中正常工作
- 假设介质的吸收率只与介质材料相关
- 假设重建图像的像素值与介质的吸收率成正比
- 假设未知介质的吸收率在 0, 1 之间

三、符号说明

符号	意义
r	小圆半径
a	椭圆长半轴
b	椭圆短半轴
d	发射器单元的间距
V_m	椭圆接收信息图像峰值
x_1	椭圆接收信息图像峰值坐标
x_2	圆接收信息图像峰值坐标
u	介质中某个点的吸收率
l	x 射线穿过介质的长度
I_0	x 射线透射前光强
I	x 射线透射后光强
n	有接收信息的
θ	x 射线所在直线与 x 轴的夹角
l_i	直角坐标系中第 i 条直线

四、模型的建立与求解

4.1 问题一的分析和求解

I. 问题的分析

问题一分为三个子问题，确定 CT 系统旋转中心在正方形托盘中的位置，确定探测器单元之间的距离以及确定 CT 系统使用的 X 射线的 180 个方向。

对于确定探测器单元之间的距离的问题，首先确定探测器接收信息值 $\ln(\frac{I_0}{I})$ 与介质的吸收率 u 、x 射线穿过介质的距离 l 之间的关系,再由标定模板中圆模板的对称性得出 x 射线穿过圆模板后在探测其中留有接收信息的区域的长度恒定并且为圆模板的直径，由此确定探测器单元之间的距离。

对于确定 CT 系统使用的 x 射线的 180 个方向的问题，首先根据探测器的接收信息

计算出椭圆模板与圆模板中心所连线段在探测器直线上的投影在探测器直线上的位置，进而确定投影长度，根据投影长度与实际长度的比值确定射线的方向。

对于确定 CT 系统旋转中心在正方形托盘中位置的问题，此问题属于平面几何的问题，运用解析几何的方法求解，已椭圆模板的中心为坐标原点建立平面直角坐标系，再确定三个不同角度下过同一探测器的三条直线方程，根据几何关系得出旋转中心与这三条直线的距离相等，计算出坐标系下与这三条直线的距离相等的点的坐标，即得出旋转中心的位置。

II. 模型的建立

1. 求解探测器间距模型的建立

x 射线穿过圆模板在探测器上留下接收信息的过程如下图所示：由圆模板的对称性，

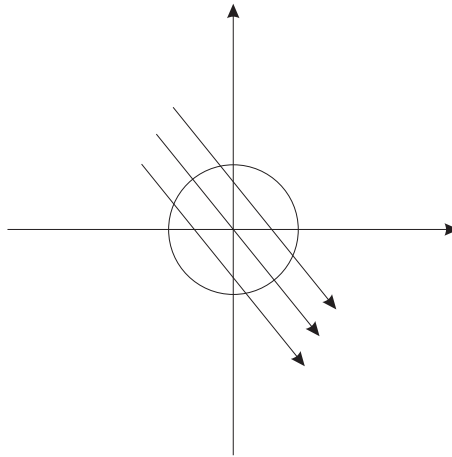


图 2 x 射线穿过圆模板过程示意图

x 射线由不同方向穿过源模板，在探测器上留有接收信息的范围相等为圆模板的直径 $2r$ ，此范围包含的探测器数目相等为 n ，发射器单元的间距 $d=2r/n$

2. 求解 CT 系统使用的 X 射线的角度模型的建立

Step 1: x 射线穿过模板强度衰减过程分析 x 射线在穿过均匀材料的物质时 [1] 其强度的衰减率与强度本身成正比，即

$$\frac{dI}{dl} = -kuI \quad (k \text{ 为系数}) \quad (1)$$

其中 I 为射线强度， l 为物质在射线方向的厚度， u 为吸收率，由此可得

$$I = I_0 e^{-kul} \quad (2)$$

当 x 射线穿过由不同衰减系数材料组成的非均匀物体, u 为某平面坐标 x,y 函数 $u(x,y)$, 当射线沿 xy 平面内直线 L 穿行时, (2) 式变为

$$I = I_0 e^{-k \int_L u(x,y) dl} \quad (3)$$

其中 $\int_L u(x,y) dl$ 是 $u(x,y)$ 沿 L 的线积分, 由 (3) 可得

$$k \int_L u(x,y) dl = \ln \frac{I_0}{I} \quad (4)$$

对于本问题中的情况, 有

$$kul = \ln \frac{I_0}{I} \quad (5)$$

Step 2: x 射线角度与探测器接收信息峰值 V_m 关系的建立 在某角度下, x 射线穿过椭圆模板在接收器上留下接收信息的过程示意如下: 以椭圆中心为原点建立平面直角

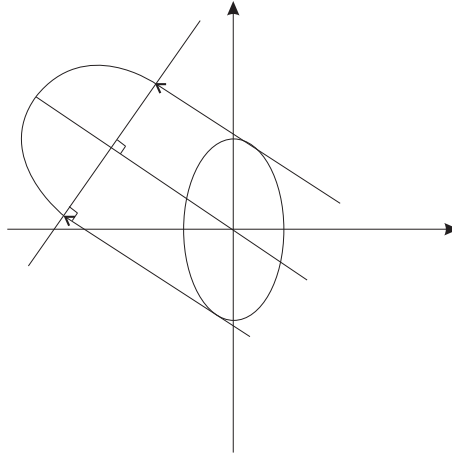


图 3 x 射线穿过椭圆模板过程及接收信息示意图

坐标系, 由平面几何的知识可知, 接收信息峰值 V_m 对应的射线所在直线过椭圆中心 O , 由峰值 V_m 可以给出弦长 l (x 射线穿过介质的长度), $l = V_m/k$, 再由峰值 V_m 可以确定直线斜率, 进而确定 x 射线所在直线与 x 轴的夹角 θ 。夹角 θ 与接收信息峰值 V_m 的关系由下式给出:

$$\theta = \arctan \sqrt{\frac{4 - V_m^2/k^2 b^2}{V_m^2/k^2 a^2 - 4}} \quad (6)$$

3. 求解 CT 系统旋转中心在正方形托盘中的位置的模型建立

示意图如下: 为求出旋转中心的位置, 只分析各个角度下过接收器中间位置点的 x 射线, 由几何关系可知, 给定三个不同角度下过接收器中间位置点的直线 $l_i (i=1,2,3)$, 旋转中心到三条直线的距离相等, 确定三个任意角度下的直线方程计算出到此三条直线距离相等的点集, 再由几何关系可计算出旋转中心。

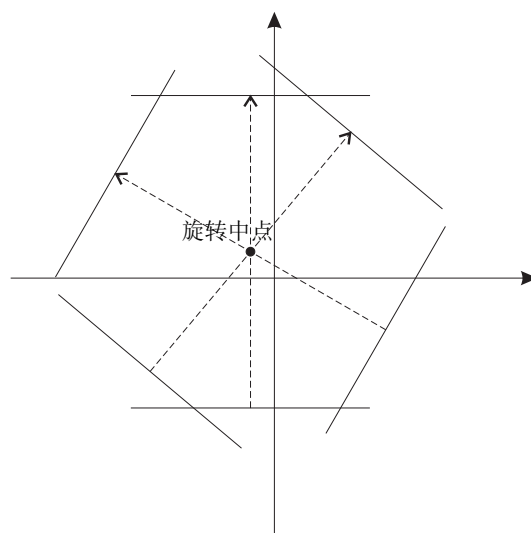


图 4 求解旋转中心模型示意图

III. 模型的求解

1. 求解探测器单元间距

分析附件二的数据可知，多数角度下 x 射线穿过源模板在探测器上留有接收信息的范围内包含的探测器数目 n 为 29，极少数的探测器数目 n 为 28，故取 n 值为 29，将数值代入 $d=2r/n$ 中得探测器单元间距：

$$d = 0.2768mm$$

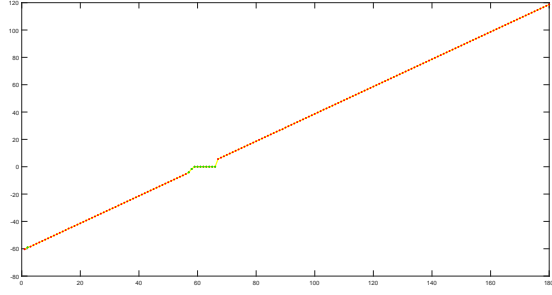
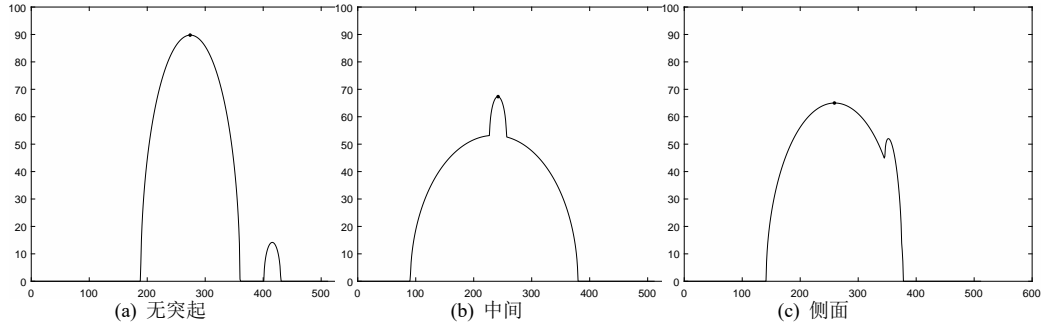
2. 求解 CT 系统使用的 X 射线的角度

根据公式 (6) 可知，计算出不同角度下 x 射线穿过椭圆模板得探测器接收信息峰值 V_m 带入式中即可计算出相应的角度 θ 。先将三种典型角度下接收信息峰值 V_m 与位置图像以及计算出的角度值结果图像给出如下：

求解的 180 个角度值见附件中 jiaodu.xls。由上图可知，在求不同角度下接收信息峰值 V_m 的过程中，图像中央由凸起的角度无法精确计算，其它角度下可以精确计算角度，经过对附件二数据的分析并求取 180 个峰值 V_m ，将 V_m 代入公式 (6) 可计算出角度值 θ 。(具体数值见附件)。求解的角度值与位置整体为线性关系，由于图 5(a) 中的凸起导致少数的角度值计算误差较大。

3. 求解旋转中心

由上述求解旋转中心模型可知，运用解析几何的方法计算旋转中心，椭圆模板的中心为坐标原点建立平面直角坐标系，为便于计算确定三个特殊角度下过接收器中间位置



(d) 求解的角度值图像

点的 x 射线所在直线的方程:

$$l_1 : y = 5.8128$$

$$l_2 : y = x + 15.2615$$

$$l_3 : x = -9.133$$

求解到这三条直线距离相等的点集，再根据几何关系确定旋转中心的坐标。求解的旋转中心的坐标为:

$$(-9.6713, 6.3511)$$

4.2 问题二的分析和求解

I. 问题的分析

对于问题二，首先我们认识到一个 CT 图像重建的问题，其次在进行图像重建的过程中，需要考虑到初始成像角度 α 以及探测器旋转中心偏离正方形托盘中心对于复原图象的影响。

II. 模型的建立

Step 1: 对 x 射线穿过介质的衰减过程模型的建立

设 $f(x,y)$ 为被测介质的线性衰减系数的分布函数, 当强度为 I_0 的 x 射线穿过介质时, 探测器检测到的射线强度 I 服从 Beer 定理, 即

$$I = I_0 e^{-\int f(x,y)\delta(x\cos\theta+y\sin\theta-R)dx dy} \quad (7)$$

式中 θ 为投影函数的坐标轴 R 与 x 轴的夹角, 它反映了投影的方向; R 是投影函数的一维变量。令 $g_{\theta}(R) = \ln(I_0/I)$, 得到

$$g_{\theta}(R) = \iint f(x,y)\delta(x\cos\theta+y\sin\theta-R)dx dy$$

$g_{\theta}(R)$ 为射线穿过物体的投影值。可见投影值 $g_{\theta}(R)$ 是被检介质的线性衰减系数分布函数 $f(x,y)$ 沿射线路径的线积分值, 即 Radon 变换。所谓图像重建就是由各个不同方向不同位置的投影值来得到线性衰减系数分布函数 $f(x,y)$ 。

Step 2: 直接反投影与滤波反投影模型的建立

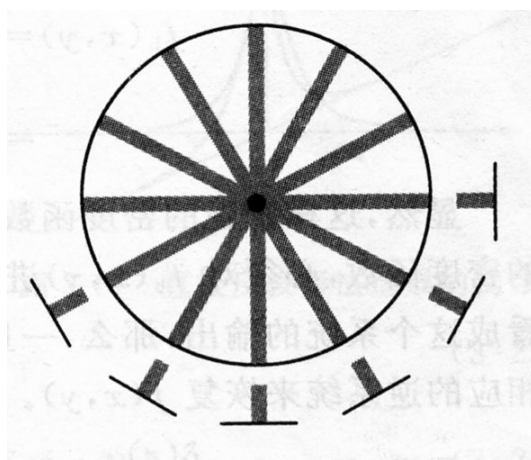


图 5 直接反投影模型示意图

[2] 直接反投影法把每次测得的投影数据“原路”反投影到投影线的各个像素上。即指定投影线上所有各点的值等于所测得的投影值。如上图所示的例子中, 被探查物体只是在原点位置上的一个致密点。在一个角度上测量到其投影值时, 就把这个值赋给投影线上的所有点。于是, 从不同角度进行反投影后的重建图像是由以原点为中心的一系列辐射线。显然, 原点位置上的分布密度最高; 愈往四周, 密度愈低。这当然可以说是粗略地把图像恢复出来了。但问题是除了密度最大的中心点, 四周出现了逐渐变浅的云晕状阴影。

直接反投影法重建的图像是模糊的。虽然这种模糊的图像可以经过修正后再再现原始的密度函数，但修正的过程是很费时的。这就是先反投影，后修正重建方法存在的问题。滤波反投影法采用先修正、后反投影的做法，同样可得到原始的密度函数。

III. 模型的求解

Step 1: 定量分析探测器旋转中心的偏移对于重建图像的影响

为探究探测器旋转中心的偏移对于重建图像的影响，我们首先采用 `radon` 变换及其逆变换并考虑到初始探测角度的因素对于附件一的标准图像进行了模拟实验，实验结果如下图：分析实验数据，发现旋转中心的偏移对于重建图像的影响仅为将图像分别向下

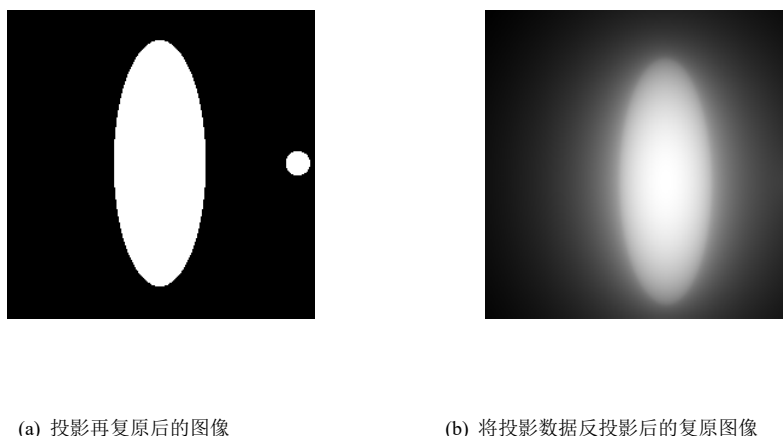


图 6 问题中的图形

平移 6.3511mm, 向右平移 9.6713mm, 即与旋转中心的偏移相对于正方形中心的偏移。

Step 2: 用滤波反投影法重建图像

根据上述模型的建立过程，采用滤波反投影法对 CT 图像进行重建。算法设计如下：

- (1) 对某一角度下的投影函数作一维傅立叶变换
- (2) 对 (1) 的变换结果乘上一维权重因子
- (3) 对 (2) 的加权结果作一维逆傅立叶变换
- (4) 用 (3) 中得出的修正过的投影函数做直接反投影
- (5) 改变投影角度，重复 (1) (4) 的过程，直到完成全部 180 度的反投影。我们采用 RL 滤波器，图像复原结果如下图：

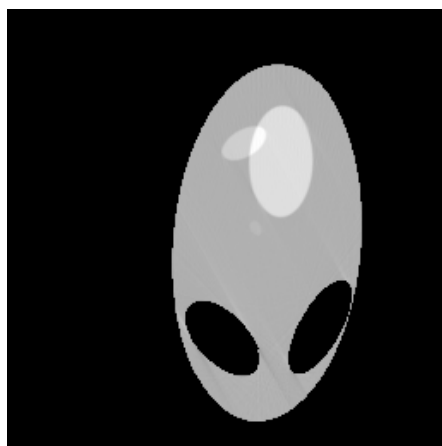


图 7 RL 滤波反投影重建后的图像

step3: 对图像进行修正

将第二步中的图像按照第一步中得到的修正方法进行修正后的图像如下：



图 8 修正后的图像

复原图像数据见附件中 problem2.xls 文件

点的位置	(10,18)	(34,25)	(43.5,33)	(45,75.7)	(48.5,55.5)	(50.75,5)	(56,76.5)	(65.5,37)	(79.5,18)	(98.5,43.5)
吸收率	0.0000	0.6785	0.0000	0.8067	0.6984	0.8786	0.8703	0.0000	0.0000	0.0000

表 1 10 个位置的吸收率

4.3 问题三的分析 and 求解

I. 问题的分析

对于问题三，问题中的图像复原问题存在着与问题二相同的问题，需要同时考虑到初始角度值和旋转中心偏离对于复原图像的影响，先采用滤波反投影算法得到此问题中未修正的原图像，观察发现此图像几乎完全覆盖正方形托盘，而问题二中的图像局限在正方形中的一部分区域，这导致将图像用滤波反投影重建算法复原并且经过平移修正后不能得到完整的原始图像. 通过在问题一中的模拟实验发现，初始角度值设置为不同的数值时，实现了图像相对于正方形托盘背景的旋转，因此可以通过图像旋转，再叠加取平均值的方法复原原始图像。模拟实验结果如下：

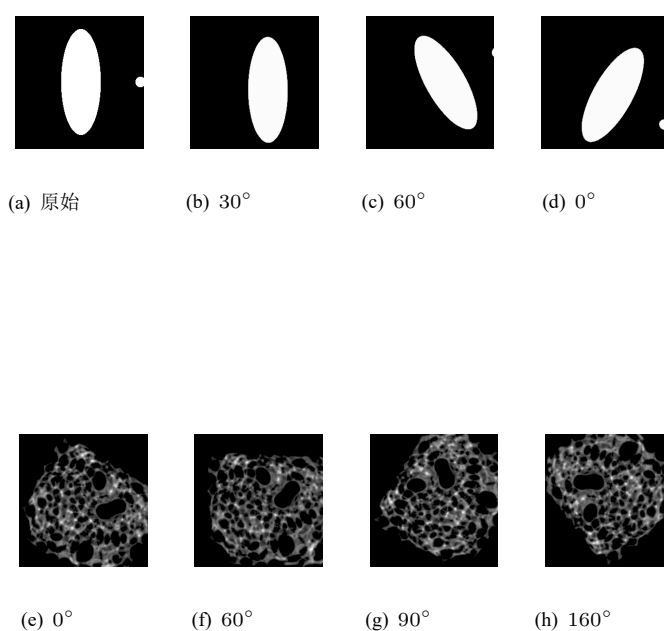


图 9 复原图像

对于问题三，问题的模型是问题二模型的延伸，由以上的问题分析知，需要改变不同的初始角度先对图像进行滤波反投影图像重建得到一组初始的复原图像，进一步将图像旋转至同一角度并叠加求取平均值得到完整的复原图像，最后对图像进行平移修正得到正确的复原图像。(上图为四个不同初始角度下的初始复原图像)

为此设计算法如下：

(1) 对某一角度下的投影函数作一维傅立叶变换

- (2) 对 (1) 的变换结果乘上一维权重因子
- (3) 对 (2) 的加权结果作一维逆傅立叶变换
- (4) 用 (3) 中得出的修正过的投影函数做直接反投影
- (5) 改变投影角度，重复 (1) (4) 的过程，直到完成全部 180 度的反投影。
- (6) 改变反投影中的初始角度，重复 (1) (5) 的过程，得到一组初始复原图像
- (7) 将这一组初始复原图像旋转至问题一确定的初始角度上，图像相加取平均值
- (8) 将 (7) 中的图像进行平移修正，得到正确的复原图像

III. 模型的求解

按照模型设计中的算法编写 matlab 程序，得到复原图像如图 9 复原图像数据见附

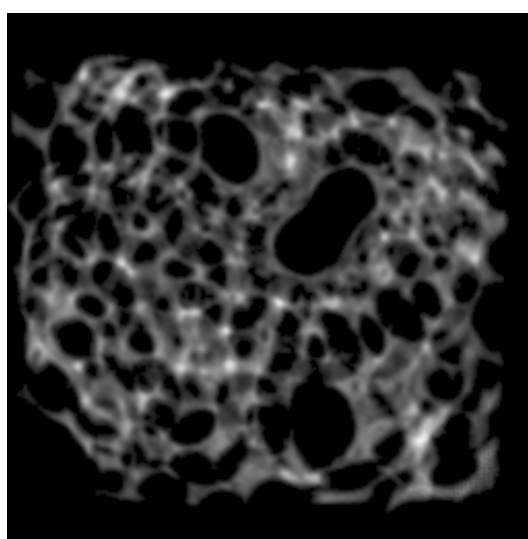


图 10 复原图像

点的位置	(10,18)	(34,25)	(43.5,33)	(45,75.7)	(48.5,55.5)	(50.75,5)	(56,76.5)	(65.5,37)	(79.5,18)	(98.5,43.5)
吸收率	0.0000	0.0579	0.4070	0.0000	0.3020	0.1914	0.2058	0.2861	0.3364	0.0000

表 2 10 个位置的吸收率

件中 problem3.xls 文件

4.4 问题四的分析和求解

通过第一问的计算，会发现对于角度的计算部分数据存在较大偏差的原因是椭圆与圆接收信息重合的区域，从而间接影响了对旋转中心的确定. 在此考虑模型需保证各投影方向总出现两组各自独立的接收信息。故建立如下图的模型 如图，四周的三个圆的圆心构成正三角形，中心圆圆心的位置恰好为正三角形的几何中心。对于旋转角度的确定，

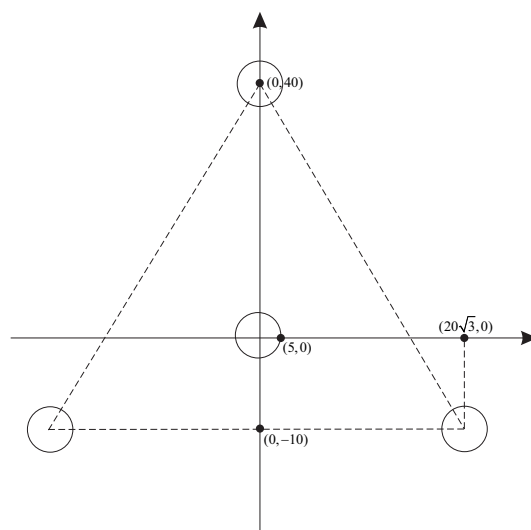


图 11 模型示意图

使用新的方法，示意图如下，如图，当圆 1 与圆 3 的接收信息发生重叠的时候，可根据连接圆 2 圆心与圆 4 圆心的线段的投影 1，确定此时的偏转角度为 $60 - \arctan \frac{l}{l_{\text{实}} * \sqrt{3}}$ ($l_{\text{实}}$ 为圆 4 圆心与圆 2 圆心在模板上的真实距离) 通过选取适当的圆半径和圆间距，可保证

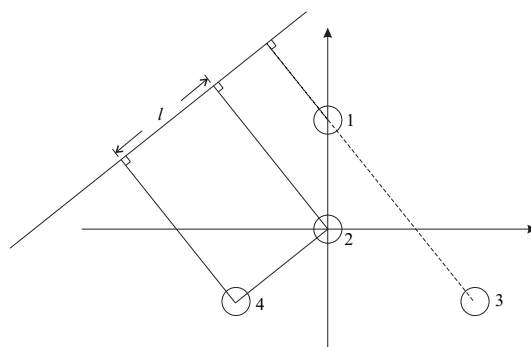


图 12 计算方法示意图

在任意视角之下，不会出现三个圆的接收信息重叠的情况。经过计算可得，选取圆半径为 5mm，中心圆的圆心坐标为 (0, 0)，正上方圆的圆心坐标为 (0, 40)，左下方圆的圆心坐标为 $(-20\sqrt{3}, -20)$ ，右下方圆的圆心坐标为 $(20\sqrt{3}, 20)$ 。

现验证此模型的合理性，如下图，设使得 2,3 圆接收信息相交的射线斜率临界值为 k ，设公切线的方程为 $y = kx + b$ 公切线方程与圆 1、圆 3 方程联立解得 $k = -1.5$ ，现计算 $(20\sqrt{3}, -20)$ 到 $y = -1.5x$ 的距离为 $17.7282 > 5 * 2 = 10$ ，故圆 2 与圆 3 的接收信息不相交，由对称性可知，在各方向不会出现三个圆模板接收信息重合的情况。所以此模型合理。

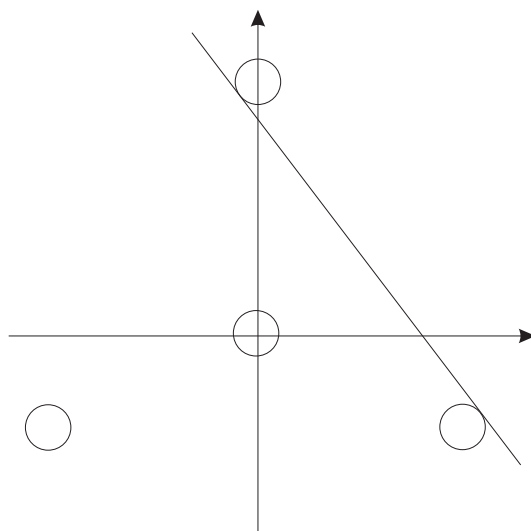


图 13 模型合理性验证示意图

五、模型的评价

5.1 模型优点

对于第一问的求解主要使用几何法，易于操作，直观易懂。

对于第二三问的求解使用反投影算法，而且使用滤波算法，还原图像较为精确，对于吸收率的计算通过图像矩阵归一化控制在 0-1 之间，而且采用最大类间方差方法设置阈值，达到清除误差的效果。而且考虑旋转中心与坐标原点的偏差，加以修正。

对于第四问的求解，反思第一问的模型，设计出了新的模板以保证总会出现两个不相交的圆的接收消息。

5.2 模型缺点

第一问关于旋转中心的求解未能考虑全部数据，容易出现误差。

对于第二三问的求解，平移移动容易出现误差。

对于第四问的求解，设计的新模板对称性过强，为判断角度增加一定的繁琐工作。

六、参考文献

- [1] 姜启源谢金星数学模型 [M], 第四版北京：高等教育出版社
- [2] 张顺利李卫斌唐高峰滤波反投影图像重建算法研究 [J] 咸阳师范学院学报 2008 23(4):116-120

附录 A 问题一程序

1.1 求增益系数

```
%求增益系数
%求圆的直径，等效为求单独圆的曲线的最高点
clear
clc
data=xlsread('A 题附件.xls','附件 2','A1:FX512');
%suppose left/right value=0
count=zeros(180,1);
maxlen=zeros(180,1);
for i=1:180
vis=false;
count(i)=0;
for j=1:512
%一条曲线的起始坐标
if data(j,i)>0 && vis==false
count(i)=count(i)+1;
vis=true;
pos(i,count(i)).l=j;
else
%一条曲线的终止坐标
if data(j,i)==0 && vis==true
vis=false;
pos(i,count(i)).r=j-1;
pos(i,count(i)).len=j-pos(i,count(i)).l;
%如果一条曲线的宽度小于 35，则这条曲线是单独圆的投影曲线
%找到单独圆的投影曲线的最高点
if (pos(i,count(i)).len<35)
maxlen(i)=max(data( pos(i,count(i)).l : pos(i,count(i)).r ,
i));
endend
end
end
end
maxlen=sort(maxlen,'descend');
%观察数据发现最大值 14.1796000000000 有 10 个，数目较大，可认为该值为圆的直径
fprintf('圆的直径为%f\n',maxlen(1));
```

1.2 求探测器单元之间距离

```
%求探测器单元之间距离(用逐差法：减少误差)
%验证：每个圆曲线几乎有 29 个点，探测器单元之间距离在 8(直径)/29~8/28 之间
clear
```



```

clc
data=xlsread('A 题附件.xls','附件 2','A1:FX512');
coe=8/14.1796;
r=4;
len=coe.*data./1;
len=len./2;
len=sqrt(r*r-len.*len);%线到圆心的距离
%suppose left/right value=0,即最左边/右边都没有被使用
dist=[];
count=zeros(180,1);
for i=1:180
vis=false;
count(i)=0;
for j=1:512
%一条曲线的起始坐标
if data(j,i)>0 && vis==false
count(i)=count(i)+1;
vis=true;
pos(i,count(i)).l=j;
else
%一条曲线的终止坐标
if data(j,i)==0 && vis==true
vis=false;
pos(i,count(i)).r=j-1;
pos(i,count(i)).len=j-pos(i,count(i)).l;
%如果一条曲线的宽度小于 35, 则这条曲线是单独圆的投影曲线
%找到单独圆的投影曲线的最高点if (pos(i,count(i)).len<35)
for k=pos(i,count(i)).l : pos(i,count(i)).r
if data(k,i) >= data(k+1,i)
break;
end
end
%这里的最高点不知道是在单独椭圆的投影曲线左边还是右边
%所以最高点不能用, 且把数据一分为二来求
%左边的数据: pos(i,count(i)).l~k-1
sum=0;
count_cal=floor((k-1-pos(i,count(i)).l) / 2);
for
index=pos(i,count(i)).l+count_cal:pos(i,count(i)).l+2*count_cal-1
sum=sum-len(index,i);
end
for index=pos(i,count(i)).l:pos(i,count(i)).l+count_cal-1
sum=sum+len(index,i);
end
sum=sum/count_cal/count_cal;
dist=cat(1,dist,sum);
%右边的数据: k+1~pos(i,count(i)).r

```

```

sum=0;
count_cal=floor((pos(i,count(i)).r-k-1) / 2);
for index=k+1+count_cal:k+1+2*count_cal-1
sum=sum+len(index,i);
end
for index=k+1:k+1+count_cal-1
sum=sum-len(index,i);
end
sum=sum/count_cal/count_cal;
dist=cat(1,dist,sum);
end
end
end
end
end
% dist=sort(dist);
avelen=mean(dist);
%[0.276759138603081]
fprintf('单个探测器单元之间距离为%f\n',avelen);

```

1.3 求 x 射线方向

```

%求射线方向(优化版)
%求椭圆曲线的最高点
%边界的点不太准，中间的点不准或求不出来，其它的点有些不太准，另外非常准
%误差大于 0.5: 58
%误差在 0.2~0.5: 2,57
%误差在 0.1~0.2: 3,16,32,56,89,150,180
%误差在 0.01~0.1: 5,152
%59~66:求不出来
clear
clc
data=xlsread('A 题附件.xls','附件 2','A1:FX512');
coe=8/14.1796;
data=coe.*data./1;
width=30;
height=80;%因为这个数值大，所以尽量使用这个数值，使误差比较小,而且曲线是椭圆，越
到顶部，增加越少
DeltaD=0.276759138603081;%探测器单元之间的距离
count=zeros(180,1);
maxh=zeros(180,1);
direction=zeros(180,1);
for i=1:180
vis=false;
count(i)=0;

```

```

for j=1:512
%一条曲线的起始坐标
if data(j,i)>0 && vis==false
count(i)=count(i)+1;
vis=true;
pos(i,count(i)).l=j;
else
%一条曲线的终止坐标
if data(j,i)==0 && vis==true
vis=false;
pos(i,count(i)).r=j-1;
pos(i,count(i)).len=j-pos(i,count(i)).l;end
end
end
% %测试某一特殊位置
% if i==66
% fprintf('Test\n');
% end
%如果有两条曲线，则椭圆投影曲线和圆投影曲线必然分开
%圆投影曲线对椭圆投影曲线没有影响
if count(i)==2
if pos(i,1).len>pos(i,2).len
maxh(i)=max(data(pos(i,1).l : pos(i,1).r,i));
part=data(pos(i,1).l : pos(i,1).r,i);
maxnum=find(part==max(part));
hpos(i)=maxnum(1)+pos(i,1).l-1;%
else
maxh(i)=max(data(pos(i,2).l : pos(i,2).r,i));
part=data(pos(i,2).l : pos(i,2).r,i);
maxnum=find(part==max(part));
hpos(i)=maxnum(1)+pos(i,2).l-1;%
end
else
%椭圆曲线方程: $y=\sqrt{a*x^2+b}$ ，相邻点的斜率是逐渐变小的
j=pos(i,1).l;
while j<=pos(i,1).r-1
j=j+1;
%目标是求单独椭圆曲线的最高点，由于圆曲线的影响导致有些椭圆曲线上面
有凸起，曲线最高值有可能不是原来单独椭圆曲线的最高值
%如果圆的凸起在椭圆曲线中间附近，则椭圆曲线中间附近变换很缓慢，一瞬
间数值会增大，可以执行删除圆的凸起
%如果圆的凸起不在椭圆曲线中间附近，距离椭圆曲线中间差很大，即使圆凸
不会导致一瞬间数值会增大，圆的凸起的最高点不是当前椭圆曲线的最高点， 保留圆的凸起
不会对求椭圆曲线的最高点无影响
if data(j,i)-data(j-1,i) < data(j+1,i)-data(j,i)
%圆的凸起的位置:
%左边: 则只要在下行下降的时候截止下降趋势(圆投影曲线和椭圆投影曲线重

```

合结束，斜率由负变正)，则停止，删除的是圆的凸起%右边：椭圆投影曲线的右边数据是逐渐递减的，所以不会在中途停止，
只会在椭圆投影曲线的最右边停止，

%删除的是从圆的凸起的左边到椭圆投影曲线的最右边的部分，这里

删除椭圆投影曲线的右边一部分对求椭圆投影曲线的最高点无影响，因为最高点不在此处

%如果圆的凸起在椭圆投影曲线中间附近，删去的圆的凸起在椭圆曲线的

中心位置，即把单独椭圆曲线的最高点也删去了

%那么剩下的点中高度最大值所在的点与椭圆曲线中心的距离比较大，

存在误差，且误差较大

%曲线上升(直到下降才停止)： k 为最高点

```
for k=j : pos(i,1).r
```

```
if data(k,i)>data(k+1,i)
```

```
break;
```

```
end
```

```
end
```

%曲线下降(直到上升才停止)： l 为转折点

```
for l=k+1 : pos(i,1).r
```

```
if data(l,i)<data(l+1,i)
```

```
break;
```

```
end
```

```
end
```

%删除圆的凸起部分(有时候还同时删除椭圆投影曲线的右边一部分)

```
data(j:l,i)=0;
```

```
j=l;
```

```
end
```

```
end
```

```
maxh(i)=max(data(pos(i,1).l : pos(i,1).r,i));
```

```
part=data(pos(i,1).l : pos(i,1).r,i);
```

```
maxnum=find(part==max(part));
```

```
hpos(i)=maxnum(1)+pos(i,1).l-1;%
```

```
end
```

%数值为 80 时，分母为 0，会发生错误

```
if maxh(i)~=80
```

```
direction(i)=atan(sqrt(1375/((1200/maxh(i))*(1200/maxh(i))-225) -1))
```

```
/pi*180;
```

```
else
```

```
error('error');
```

```
end
```

```
end
```

%角度作调整：可以不弄

```
for i=1:58direction(i)=-direction(i);
```

```
end
```

```
for i=152:180
```

```
direction(i)=180-direction(i);
```

```
end
```

%对曲线 和 错误的点在图上进行展示

```
x_wrong=[2,57:66];
```

```
direction_wrong=direction(x_wrong(:));
```

```

x=[1:180]';
plot(x,direction,'y-','Linewidth',2);
hold on
plot(x,direction,'r.','markersize',10);
plot(x_wrong,direction_wrong,'g.','markersize',10);
%平均误差
Diff=0;
Diff_count=180-(67-58+1);
count_good=0;
for i=1:58
Diff=Diff + abs( direction(i) - (i-61.3533) );
if abs( direction(i) - (i-61.3533) )<0.01
count_good=count_good+1;
end
end
for i=67:180
Diff=Diff + abs( direction(i) - (i-61.3533) );
if abs( direction(i) - (i-61.3533) )<0.01
count_good=count_good+1;
end
end
Diff=Diff/Diff_count;
fprintf('角度误差在 0.01 以内的数目有%d 个\n',count_good);
fprintf('角度平均误差为%f\n',Diff);
%某些单独椭圆曲线的宽度 hpos(i)不对,使用的是 1~30 93~180 的数据

```

附录 B 问题二、三程序

2.1 图像重建的程序

%生成探测器检测到的图，未知介质在正方形托盘中的位置的图（归一化/未归一化）等图
%和它们对应的数据，其中吸收率保存在 problem2/3.xls 中

```

clc clear
close all
%获得原始数据
num=0;
beg=30;
step=1;
photo=[];
changephoto=[];
maxlen=364;
finalphoto=zeros(maxlen,maxlen);
timephoto=zeros(maxlen,maxlen);
for delta=beg:step:beg+360-1
num=num+1;

```

```

fprintf('%d\n',delta);
theta = delta:1:delta+180-1; %投影角度
P = xlsread('A 题附件.xls','附件 5','A1:FX512'); %投影数据 3/5
rc = iradon(P,theta,'linear','None');%直接反投影重建
rec_RL = iradon(P,theta,'linear','Ram-Lak');%默认滤波器
rec_SL = iradon(P,theta,'linear','Shepp-Logan');
rc1=imresize(rc,[256 256],'bilinear');rc1=mat2gray(rc1);
rec_RL1=imresize(rec_RL,[256 256],'bilinear');rec_RL1=mat2gray(rec_RL1);
rec_SL1=imresize(rec_SL,[256 256],'bilinear');rec_SL1=mat2gray(rec_SL1);
if num==1
xlswrite('R-L 函数_探测器检测到的图
_Angle.xlsx',rec_SL1,'Sheet1','A1');%R/S
imwrite(rec_SL1,'R-L 函数_探测器检测到的图_Angle.png');%R/S
imshow(rec_SL1);
saveas(gcf,'R-L 函数_探测器检测到的图_Angle','eps');
a=rec_SL1;%R/S
a(a<0.08)=0;
xlswrite('R-L 函数_探测器检测到的图_Angle_归一
化.xlsx',a,'Sheet1','A1');%R/S
imwrite(a,'R-L 函数_探测器检测到的图_Angle_归一化.png');%R/S
imshow(a);
saveas(gcf,'R-L 函数_探测器检测到的图_Angle_归一化','eps');
xlswrite('problem3.xls',a,'Sheet1','A1');%2/3
end%对图片进行旋转,使图片处于原来的角度
newphoto=imrotate(rec_RL1,360-(delta-beg),'bilinear','loose');%R/S
%把旋转后的图片放入大正方形的中心位置(中心位置不变)
%对所有的旋转后的图片进行叠加
len=size(newphoto,1);
add=floor( (maxlen-len)/2 );
for i=1:len
for j=1:len
if newphoto(i,j)~=0 %0.5
x=i+add;
y=j+add;
finalphoto(x,y)=finalphoto(x,y)+newphoto(i,j);
timephoto(x,y)=timephoto(x,y)+1;
end
end
end
end
%对图片的所有点取平均值
for i=1:maxlen
for j=1:maxlen
%注意: 如果该点从未有数值,即 timephoto(i,j)=0, 则不能相除
if timephoto(i,j)>0
finalphoto(i,j)=finalphoto(i,j)/timephoto(i,j);
end

```

```

end
end
imshow(finalphoto);
xlswrite('R-L 函数_未经修改.xlsx',finalphoto,'Sheet1','A1');%R/S
imwrite(finalphoto,'R-L 函数_未经修改.png');%R/S
imshow(finalphoto);
saveas(gcf,'R-L 函数_未经修改','eps');
a=finalphoto;
a(a<0.08)=0;%0.1
imshow(a);
%经过观察图片，得出用 graythresh 求阈值的方法不好
% a=finalphoto;
% bw=graythresh(a);
% a(find(a<bw))=0;
% imshow(a);%(364-256)/2=54
%55~310
b=a(55:310,55:310);
xlswrite('R-L 函数_探测器检测到的图_归一化.xlsx',b,'Sheet1','A1');%R/S
imwrite(b,'R-L 函数_探测器检测到的图_归一化.png');%R/S
imshow(b);
saveas(gcf,'R-L 函数_探测器检测到的图_归一化','eps');
%未知介质在正方形托盘中的位置：在原有基础上进行向量的平移
%向量： -16,-25
c=a(55+16:310+16,55+25:310+25);
xlswrite('R-L 函数_未知介质在正方形托盘中的位置_归一化.xlsx',c,'Sheet1','A1');%R/S
imwrite(c,'R-L 函数_未知介质在正方形托盘中的位置_归一化.png');%R/S
imshow(c);
saveas(gcf,'R-L 函数_未知介质在正方形托盘中的位置_归一化','eps');
%原来图片-未归一化
p=finalphoto(55:310,55:310);
xlswrite('R-L 函数_探测器检测到的图.xlsx',p,'Sheet1','A1');%R/S
imwrite(p,'R-L 函数_探测器检测到的图.png');%R/S
imshow(p);
saveas(gcf,'R-L 函数_探测器检测到的图','eps');
q=finalphoto(55+16:310+16,55+25:310+25);
xlswrite('R-L 函数_未知介质在正方形托盘中的位置.xlsx',q,'Sheet1','A1');%R/S
imwrite(q,'R-L 函数_未知介质在正方形托盘中的位置.png');%R/S
imshow(q);
saveas(gcf,'R-L 函数_未知介质在正方形托盘中的位置','eps');

```

2.2 求十个样本点吸收率的程序

```

%求 10 个点的吸收率
%输入数据： problem2/3.xls，输出数据： txt 文件

```

```

clear
clc
close all
%获得点所在矩阵位置
matrix_pos=xlsread('A 题附件.xls','附件 4','A1:B10');
pos=zeros(10,2);for i=1:10
%matrix_pos:(43.5,33) pos:(100-33,43.5)
pos(i,1)=100-matrix_pos(i,2);
pos(i,2)=matrix_pos(i,1);
end
% pos(length) / 100(total_length) * 256(grid)
pos=pos./100.*256;
% xlswrite('点所在矩阵位置.xlsx',pos,'Sheet1','A1');
data=xlsread('problem3.xls','Sheet1','A1:IV256');%2/3
pos_new=ceil(pos);
value=zeros(10,1);
for i=1:10
if pos(i,1)-floor(pos(i,1))==0
value(i)=( data(pos_new(i,1),pos_new(i,2)) +
data(pos_new(i,1)+1,pos_new(i,2)) )/2;
else
if pos(i,2)-floor(pos(i,2))==0
value(i)=( data(pos_new(i,1),pos_new(i,2)) +
data(pos_new(i,1),pos_new(i,2)+1) )/2;
else
value(i)=data(pos_new(i,1),pos_new(i,2));
end
end
end
fid=fopen('jieguo3_2_求平均.txt','w');%2/3
for i=1:10
fprintf(fid,'%4f \n',value(i));
end
fclose(fid);

```