

# 基于接触式轮廓仪测量数据的工件形状自动标注方法

## 摘要

接触式轮廓仪在测量工件或产品的轮廓时，由于存在探针沾污、扫描位置不准等问题，会对工件形状的准确标注带来影响，本文基于接触式轮廓仪的测量数据，研究工件形状的自动标志方法。

针对问题一，本文首先采用一种基于一阶二阶导数的模式识别方法，以此作为理论基础区分水平线、斜线和圆。针对测试数据中的噪声问题，采用一种基于滑动平均的去噪方法。并比较了有重叠滑动窗口和无重叠滑动窗口的去噪效果。通过以上方法可以确定不同曲线段的形状，然后采用一种基于直线和圆的方程的拟合方法，对不同曲线段数据进行拟合，求得各个分段函数方程。最后计算得出直线夹角、半径等参数。与原始数据图形对比，基本吻合。

针对问题二，本文采用一种基于直线旋转量的图形校正方法。水平平移不改变直线的斜率，工件的倾斜旋转的角度就是直线夹角的旋转量。首先采用问题一的方法识别出直线的图形区域，再计算倾斜后的直线旋转量，并基于此进行图形校正。校正后的数据与问题一原始数据对比误差较小。

针对问题三，本文采用一种基于多组测量数据复原完整轮廓的方法。首先采用问题一方法识别出每组数据不同曲线段的形状。然后基于问题二水平直线的旋转量对各组数据进行校正，最后综合各组数据的公共部分和左右两侧的最大冗余数据进行拼接，复原完整轮廓。

针对问题四，本文采用一种基于多组局部数据修正全局图像的方法。首先采用问题一方法识别出每组局部数据不同曲线段的形状。然后通过拟合计算不同曲线段形状的方程和参数，最后等效替代全局图形中对应部分的方程和参数，实现基于局部数据对全局图形的修正。

模型的评价。以上工件自动标注方法，可以较好的替代人工计算，很大程度的解决了接触式轮廓仪工作中存在的问题，标注误差较小。

**关键词：**数据处理 拟合分析 去噪 线性回归 最小二乘法



## 1 问题重述

### 1.1 问题背景：

接触式轮廓仪的工作原理是，探针接触到被测工件表面并匀速滑行，传感器感受到被测表面的几何变化，在 X 和 Z 方向分别采样，并转换成电信号。

在理想状况下，轮廓曲线是光滑的，但由于接触式轮廓仪存在探针玷污、探针缺陷、扫描位置不准等问题，检测到的轮廓曲线呈现出粗糙不平的情况。

### 1.2 具体问题：

假设被测工件的轮廓线是由直线和圆弧构成的平面曲线，建立数学模型，根据附件所提供的轮廓仪测量数据，研究下列问题：

#### 1.2.1 问题一：

根据工件 1 在水平状态下的测量数据，标注出轮廓线的槽口宽度（如 $x_1, x_3$ 等）、圆弧半径（如 $R_1, R_2$ 等）、圆心之间的距离（如 $c_1, c_2$ 等）、圆弧的长度、水平线段的长度（如 $x_2, x_4$ 等）、斜线线段的长度、斜线与水平线之间的夹角（如 $\angle 1, \angle 2$  等）和人字形线的高度（ $z_1$ ）。

#### 1.2.2 问题二：

计算工件 1 在测量时的倾斜角度，并作水平校正，比较理想下的水平测量与倾斜测量状态数据之间的差异。

#### 1.2.3 问题三：

根据工件 2 的 10 次测量数据完成：(1) 每次测量工件 2 的倾斜角度；(2) 标注出工件 2 轮廓线的各项参数值；(3) 画出工件 2 的完整轮廓线。

#### 1.2.4 问题四：

利用工件 2 关于圆和角的 9 次局部测量数据，修正问题 3 的结论，并对工件的完整轮廓线作进一步修正。

## 2 问题分析

针对问题一，根据水平放置的工件 1 的测量数据，计算工件自身轮廓线的参数，已知轮廓线是由直线和圆构成，那么问题的关键是我们用什么指标和区分圆和直线，注意到直线的二阶导数为零，而圆的二阶导数不为零，用二阶导数区分



圆和直线，而斜线和水平线可以用一阶导数来区分。利用区分好的数据进行圆和直线的拟合从而得到圆和直线的函数表示，再得到整个轮廓线的分段函数表达，进而可以得到各种参数。

针对问题二，已知倾斜放置工件 1 的测量数据和水平放置工件 1 的测量数据，计算倾斜角度和平移量。首先利用前面第一问的方法区分出来直线和圆，把水平测量和倾斜测量的多个圆的位置做对比，得到多个圆的平移量，取平均得到整体轮廓线的平移量；注意到工件 1 的水平测量数据中有很多的水平线，倾斜之后变成斜线，计算多条斜线的倾斜角取平均得到工件 1 的倾斜角度；最后利用倾斜角度和平移量通过矩阵相乘把倾斜放置的工件 2 的测量数据计算得到水平放置的测量数据，然后利用问题 1 的计算参数方法计算得到工件 1 水平校正后轮廓线的参数，最后和工件 1 真实水平测量的标注参数进行对比。

针对问题三，题目中给出对工件 2 的 10 次不同的测量数据，首先计算每次测量工件 2 的倾斜角度，我们假设工件 2 中本身有很多的水平线，仍然使用问题一的方法区分出直线和圆，那么斜率差不多相等的直线段比较多，然后倾斜角的平均，就为此次测量的倾斜角度；其次，因为多次测量位置不同，每次测量的整体工件 2 的部分，所以我们把每次的测量数据乘以旋转矩阵水平校正后，通过对比，得到整体工件 2 水平校正后的数据，然后利用问题 1 的方法标注参数；最后，通过水平校正后的数据画出工件 2 的完整的轮廓线。

针对问题四，附件中给出了 9 次角和圆的 9 次局部测量数据，修正工件 2 的参数，并给出修正后的完整轮廓线。

### 3 模型假设

- 1) 假设被测工件的轮廓线是由直线和圆弧构成的平面曲线；
- 2) 假设接触式轮廓仪检测准确没有误差；
- 3) 假设工件检测的轮廓线无外界干扰。



## 4 符号说明

符号	说明
$x_i$	第 $i$ 个测量点在 $x$ 轴上的坐标
$z_i$	第 $i$ 个测量点距离水平线的高度
$k_i$	第 $i$ 条直线的斜率
$k'_i$	旋转后的第 $i$ 条直线的斜率
$b$	直线在 $z$ 轴上的截距
$R$	圆弧的半径
$f_1(x)$	工件 1 轮廓线函数
$f_2(x)$	工件 2 轮廓线函数

## 5 模型建立与求解

### 5.1 问题一模型建立和求解

根据问题限定条件，工件只包含水平直线、斜线、圆三种图形。我们要识别这三种模式，需要讨论三种图形数学上的本质区别是什么，为此本文首先提出以下模型：

#### 5.1.1 基于一阶二阶导数的模式识别方法：

建立  $x, z$  直角坐标系，设  $x$  轴的方向为水平方向， $z$  轴的方向为垂直方向。设测量水平样本数据为  $(x_i, z_i), i=1 \dots n, n$  为数据个数，设工件 1 轮廓线的函数为

$f_1(x)$ ，其中  $f_1(x_i) = z_i$ ；

若轮廓线为水平直线，则满足下列条件：

$$f_1'(x_i) = 0 \quad (1-1)$$

若轮廓线为斜线：则满足以下条件：

$$f_1'(x_i) = C \quad (1-2)$$



若轮廓线为圆，则满足以下条件

$$f_1''(x_i) \neq 0 \quad (1-3)$$

其中， $f_1'(x_i) = \frac{z_{i+1} - z_i}{x_{i+1} - x_i}$ ， $f_1''(x_i) = \frac{f'(x_{i+1}) - f'(x_i)}{x_{i+1} - x_i}$

综合以上分析，本文提出可以基于一阶导数和二阶导数是否为 0 来综合判断该曲线的图形模式。如下图所示，为问题一数据的一阶导数和二阶导数图形：

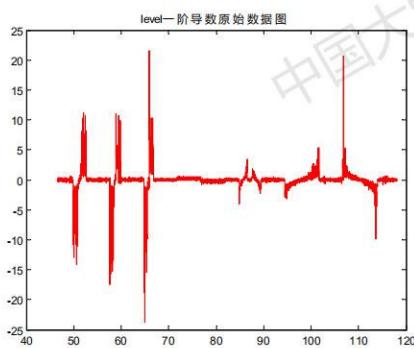


图 1 level 一阶导数原始数据图

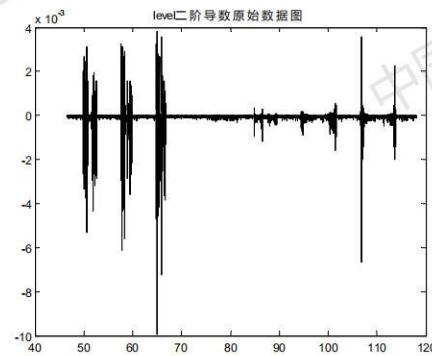


图 2 二阶导数原始数据图

从以上图形可以看出，原图形直线部分对应的一阶导数并不为 0，这是因为测量数据中包含了很多噪声。实际应用上理论方法还需要解决去噪问题。为此本文提出以下模型：

### 5.1.2 基于滑动平均的去噪方法

针对离散数据的一阶导数为

$$f_1'(x_i) = \frac{z_{i+1} - z_i}{x_{i+1} - x_i} \quad (1-4)$$

离散数据的噪声会导致以上公式的一阶导数剧烈波动，为消除噪声影响，本文提出基于滑动窗口内求平均的去噪方法，根据滑动窗口之间是否重叠，本文对比研究了有重叠区域的滑动窗口法和无重叠区域的滑动窗口法。

#### (1) 有重叠区域的滑动窗口法

选择滑动窗口长度为 N，起始横坐标  $x_i$ ，终止横坐标为  $x_{i+N}$ 。

有重叠区域的滑动窗口法，第一个窗口覆盖  $x_1$  到  $x_N$ ，第二个窗口覆盖  $x_2$  到  $x_{N+1}$ ，窗口内求平均，以此类推，则滑动窗口内的平均一阶导数记为  $\overline{f_1'(x_i)}$ ，



$$\overline{f'_1(x_i)} = \frac{1}{N} \sum_i^{i+N} f'_1(x_i) \quad (1-6)$$

同理，滑动窗口内的平均二阶导数记为  $\overline{f''_1(x_i)}$ ，

$$\overline{f''_1(x_i)} = \frac{1}{N} \sum_i^{i+N} f''_1(x_i) \quad (1-7)$$

## (2) 无重叠区域的滑动窗口法

无重叠区域的滑动窗口法，第一个窗口覆盖  $x_1$  到  $x_N$ ，第二个窗口覆盖  $x_{N+1}$  到  $x_{2N}$ ，窗口内求平均，以此类推，则滑动窗口内的平均一阶导数记为  $\overline{f'_1(x_i)}$ ，

$$\overline{f'_1(x_i)} = \frac{1}{N} \sum_{(i-1)N+1}^{iN} f'_1(x_i) \quad (1-6)$$

同理，滑动窗口内的平均二阶导数记为  $\overline{f''_1(x_i)}$ ，

$$\overline{f''_1(x_i)} = \frac{1}{N} \sum_{(i-1)N+1}^{iN} f''_1(x_i) \quad (1-7)$$

考虑到滑窗法并不能将实际误差完全降低到 0，选择阈值  $\lambda_1$  和  $\lambda_2$  分别表示一阶导数阈值和二阶导数阈值，并即滑窗内区域的图形模式为 Y，平行直线、斜线、圆分别使用 1、2、3 表示。则 Y 满足以下条件：

$$Y = \begin{cases} 1 & \text{if } (\overline{f'_1(x_i)} \leq \lambda_1) \\ 2 & \text{if } (\overline{f'_1(x_i)} > \lambda_1 \text{ 且 } \overline{f''_1(x_i)} \leq \lambda_2) \\ 3 & \text{if } (\overline{f''_1(x_i)} > \lambda_2) \end{cases} \quad (1-8)$$

应用以上模式可以求得图形中哪些数据属于水平线，哪些属于斜线，哪些属于圆。接下来由于分段的带噪声的数据分别求直线、斜线、圆的方程还需要进行拟合，即以下模型：

### 5.1.3 基于圆和直线方程的拟合方法<sup>[1]</sup>

#### (1) 圆的拟合：

设拟合曲线为  $(x - A)^2 + (z - B)^2 = R^2$ ,  $A, B, R$  是拟合参数

变形可得到圆的拟合曲线的另外形式

$$\text{所以 } x^2 + z^2 + ax + bz + c = 0$$

设截取的关于圆的测量数据为  $(x_i, z_i), i = 1 \dots N$



达到  $\min(x_i^2 + z_i^2 + ax_i + bz_i + c)^2$  求得到  $a, b, c$

得到圆心  $A = \frac{a}{-2}$ ,  $B = \frac{b}{-2}$ , 半径  $R = \frac{1}{2}\sqrt{a^2 + b^2 - 4c}$

## (2) 直线的拟合:

设拟合曲线为  $\hat{z} = kx + b$  其中  $k, b$  是拟合的参数

优化目标:  $\min(kx_i + b - z_i)^2$

$$\text{得到 } k = \frac{n \sum_{i=1}^n x_i z_i - \sum_{i=1}^n x_i \sum_{i=1}^n z_i}{n \sum_{i=1}^n x_i^2 - \sum_{i=1}^n x_i \sum_{i=1}^n x_i} \quad b = \frac{n \sum_{i=1}^n x_i^2 \sum_{i=1}^n z_i - \sum_{i=1}^n x_i z_i \sum_{i=1}^n x_i}{n \sum_{i=1}^n x_i^2 - \sum_{i=1}^n x_i \sum_{i=1}^n x_i}$$

根据该模型, 带入实测数据, 可以求得各个分段图像的方程表达式, 进而求得夹角等参数。

### 5.1.4 问题一模型的求解:

#### (1) 去噪:

分别使用有重叠滑窗和无重叠滑窗方法对一阶导数和二阶导数进行去噪, 如下图所示。

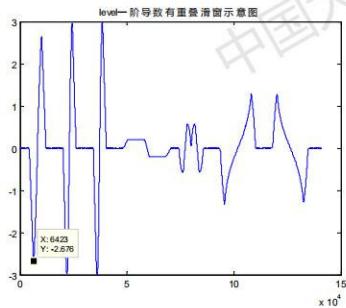


图 3 有重叠滑窗一阶导数

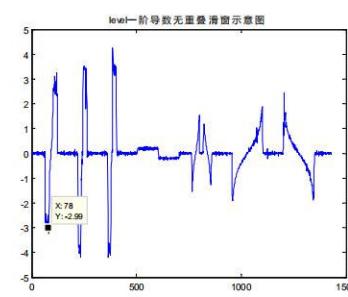
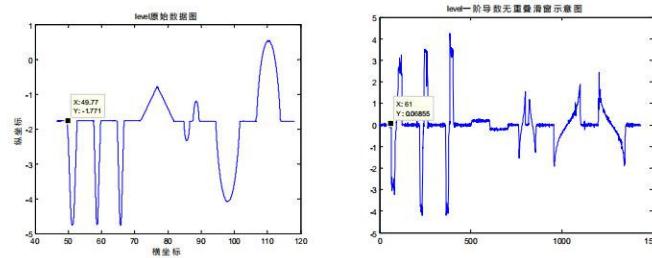


图 4 无重叠滑窗一阶导数

通过上图对比, 原始工件的第 2 个图形竖直向下斜线部分, 在有重叠滑窗条件下, 几乎不能识别出来。而图 4 中无重叠滑窗部分存在一阶导数基本不变的短暂区域, 可以大体识别出来。因此下文主要采用无滑窗的去噪方法。

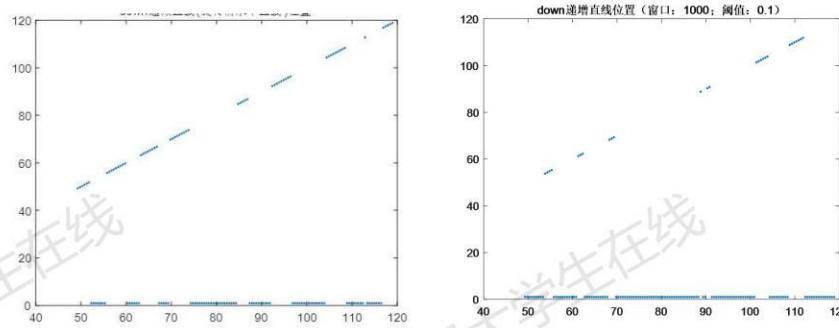
原始数据第一个曲线交点为下图标注点, 查原始数据该点为第 6052 个数据。





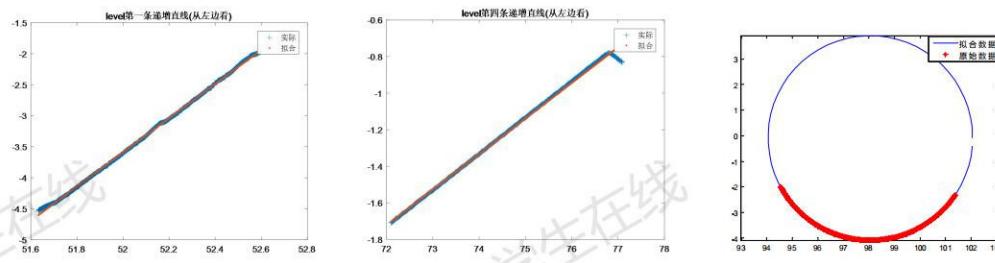
如上图是采用滑动平均后的去噪一阶导数数据，该图的第一个拐点为 61，考虑到滑窗长度设置为了 100，该点对应第 6100 个数据。

以上对比表明，该方法可以识别出不同曲线段的转折点。下图左是抓取的水平测量数据递减直线的位置，下图右是抓取的倾斜位置测量数据递减直线的位置，效果很好。（其中，一个点表示 1000 个数据）



## (2) 拟合结果：

下面是水平放置的轮廓线的第一条和第四条和第一个圆的拟合情况，其他的见附录。



## (3) 工件 1 的参数标注：

轮廓线的函数表达：



$$f(x) = \begin{cases} 1.77 & 46.595892371385 < x \leq x_1 \\ -2.723867741639066x + 133.8002234310488 & x_1 < x \leq x_2 \\ -\sqrt{0.4917^2 - (x - 51.2173)^2} - 4.2610 & x_2 < x \leq x_3 \\ 2.754359108225884x - 146.8171658730294 & x_3 < x \leq x_4 \\ 1.77 & x_4 < x \leq x_5 \\ -3.724845609376202x + 213.0069757236736 & x_5 < x \leq x_6 \\ -\sqrt{0.5259^2 - (x - 58.6019)^2} - 4.1492 & x_6 < x \leq x_7 \\ 2.980598439534949x - 179.8507093519987 & x_7 < x \leq x_8 \\ 1.77 & x_8 < x \leq x_9 \\ -3.723602901285295x + 239.2514907860618 & x_9 < x \leq x_{10} \\ -\sqrt{0.3802^2 - (x - 65.7338)^2} - 4.3746 & x_{10} < x \leq x_{11} \\ 3.511651703755757x - 236.2555247229892 & x_{11} < x \leq x_{12} \\ 1.77 & x_{12} < x \leq x_{13} \\ 0.194764774178566x - 15.745625965295059 & x_{13} < x \leq x_{14} \\ -0.200744158613707x + 14.645116449010247 & x_{14} < x \leq x_{15} \\ 1.77 & x_{15} < x \leq x_{16} \\ -\sqrt{0.9112^2 - (x - 85.7238)^2} - 1.4363 & x_{16} < x \leq x_{17} \\ 1.77 & x_{17} < x \leq x_{18} \\ \sqrt{0.5299^2 - (x - 89.7685)^2} - 2.3050 & x_{18} < x \leq x_{19} \\ 1.77 & x_{19} < x \leq x_{20} \\ -\sqrt{3.9923^2 - (x - 98.0405)^2} - 0.0869 & x_{20} < x \leq x_{21} \\ 1.77 & x_{21} < x \leq x_{22} \\ -\sqrt{1.1702^2 - (x - 114.4627)^2} - 0.6154 & x_{22} < x \leq x_{23} \\ 1.77 & x_{23} < x \leq 118.123985877831 \end{cases}$$

把拟合得到的函数与 x 轴得到相交求交点，即对函数求零点得到所有槽口宽度  $x_2, x_4$ ，水平线段的长度  $x_1, x_3$ ，通过圆心的参数得到圆心之间的距离  $c_1, c_3$ ，最后两条斜线的交点求得人字形的高度  $z_1$ 。

表 1 轮廓线的槽口宽度及圆弧的长度、水平线段的长度

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$
2.8914075 13568545	4.9982367 01987779	2.0873795 02859584	4.9800147 21154198	2.0468106 48264795	4.9940161 36801619	9.9916311 60546746
$x_8$	$x_9$	$x_{10}$	$x_{11}$	$x_{12}$	$x_{13}$	
2.6672212 42061245	0.9356328 84535536	1.1740463 57141904	12.268778 447088195	4.6218374 51824376	10.153688 233493440	

表 2 轮廓线的斜线与水平线之间的夹角

$\angle 1$	$\angle 2$	$\angle 3$	$\angle 4$
110.1594348872009	109.9539638548886	105.0276884989337	108.5467473758782
$\angle 5$	$\angle 6$	$\angle 7$	$\angle 8$
105.0324768433333	105.8951506831928	168.9787730174710	168.6490737801013



表 3 轮廓线的圆弧半径

$R_1$	$R_2$	$R_3$	$R_4$	$R_5$	$R_6$	$R_7$
0.4917	0.5259	0.3802	0.9112	0.5299	3.9923	1.1702

表 4 轮廓线的圆心之间的距离

$c_1$	$c_2$	$c_3$	$c_4$	$c_5$	$c_6$
7.44739	7.10274	20.17928	2.94711	9.73397	12.67828

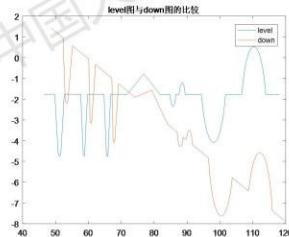
表 5 轮廓线的斜线线段的长度

$l_1$	$l_2$	$l_3$	$l_4$	$l_5$	$l_6$	$l_7$	$l_8$
2.852009	2.905920	3.485828	2.841833	2.851870	2.866131	4.828060	4.641530

### 5.3 问题二的模型建立与求解

#### 5.3.1 直线旋转量的图形校正方法<sup>[3]</sup>

首先利用前面第一问的二阶导数是否为零识别圆和直线，把所有的直线数据提取出来，分别通过拟合计算其斜率，斜率相等的直线所在的直线的斜率即是工件 2 的倾斜斜率，并基于此进行图形校正。



#### 1. 旋转角的确定：

设  $k_i, i=1 \dots l$  为第  $i$  条直线的斜率， $k'_i, i=1 \dots m$  其中差不多相等那些直线中的第  $i$  条，所以所在直线斜率

$$k = \frac{1}{m} \sum k'_i = -0.131$$

所以倾斜角为

$$\theta = \arctan k$$

利用前面的直线拟合算法，用 matlab 求解得到  $\theta = -0.13 = -7.4^\circ$ 。

#### 2. 水平校正：

水平校正分两步：



### (1) 倾斜测量数据的旋转

把倾斜的数据旋转：

$$\begin{pmatrix} \mathbf{x}_i^0 \\ \mathbf{z}_i^0 \end{pmatrix} = \begin{pmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \mathbf{x}_i' \\ \mathbf{z}_i' \end{pmatrix}, i=1 \dots n$$

其中  $\begin{pmatrix} \mathbf{x}_i^0 \\ \mathbf{z}_i^0 \end{pmatrix}$ ,  $i=1 \dots n$  为工件 1 倾斜的数据水平旋转之后得到的数据,

$\begin{pmatrix} \mathbf{x}_i' \\ \mathbf{z}_i' \end{pmatrix}$ ,  $i=1 \dots n$  为倾斜放置的工件 1 测量的原始数据。

### (2) 比较平移量

计算每一个点的平移量： $\mathbf{x}_i^0 - \mathbf{x}_i, i=1 \dots n$

取平均得到整个轮廓的水平平移量  $a = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i^0 - \mathbf{x}_i)$

通过几个特别的点，比如圆心验证。

### (3) 检验：

根据上面求得旋转角度为  $\theta$ ，水平偏移量为  $a$ ，

$$\begin{pmatrix} mx_i \\ mz_i \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta & a \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x}_i \\ \mathbf{z}_i \\ 1 \end{pmatrix}, i=1 \dots n$$

$\begin{pmatrix} \mathbf{x}_i \\ \mathbf{z}_i \end{pmatrix}$ ,  $i=1 \dots n$  为水平放置的工件 1 测量的原始数据,

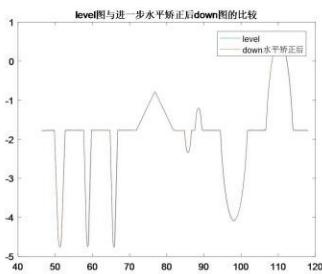
$\begin{pmatrix} mx_i \\ mz_i \end{pmatrix}$ ,  $i=1 \dots n$  为水平放置的旋转角度得到的数据。

通过误差平方和的平均检验上面得到的倾斜角度和水平偏移量的准确性，即

$$\frac{1}{n} \sum_{i=1}^n (x_i' - mx_i)^2 + (z_i' - mz_i)^2$$

从旋转后的和水平放置的原始数据对比图看出效果很好：





### 3. 对校正后的数据进行标注参数：

利用问题一模型里的三步：第一，识别圆和直线，第二，拟合直线和圆，第三，得到路廓线函数表达，计算参数。

$$f_2(x) = \begin{cases} -1.843 \\ -2.503x + 122.9 \\ -\sqrt{0.4895^2 - (51.2761 - x)^2} + 4.2784 \\ 2.548x - 136.2 \\ -1.8 \\ -3.376x + 193 \\ -\sqrt{0.2943^2 - (58.7638 - x)^2} + 4.4582 \\ 3.534x - 213.1 \\ -1.746 \\ -3.392x + 217.9 \\ -\sqrt{0.2249^2 - (65.8258 - x)^2} + 4.5422 \\ 3.498x - 235.7 \\ -1.811 \\ 0.2003x - 16.17 \\ -0.1996x + 14.57 \\ -1.701 \\ -\sqrt{1.0065^2 - (85.7614 - x)^2} + 1.3321 \\ -2.126 \\ -\sqrt{1.0833^2 - (88.5825 - x)^2} + 2.2870 \\ -1.749 \\ -\sqrt{4.0101^2 - (98.1053 - x)^2} + 0.0590 \\ -1.845 \\ -\sqrt{4.0361^2 - (110.3620 - x)^2} + 3.4915 \\ -1.705 \end{cases}$$

## 5.4 问题三的模型建立和求解：

本问题考虑的是在对工件作多次检测过程中，工件每次放置的角度、测量的起点和终点都会有偏差，此时如何根据已测得的多组数据复原工件完整轮廓的问题。

### 5.4.1 基于多组测量数据复原完整轮廓的方法

#### 1. 每次测量的倾斜角度：

我们假设工件 2 中水平放置时本身有很多的水平线，仍然使用问题一的方法区分出直线和圆，那么斜率差不多相等的直线段比较多，那么这倾斜角的平均为此次测量的倾斜角度，所以类似问题二中求解倾斜角的模型得到每次测量的倾斜角。



## 2. 对比得到工件 2 轮廓的全部水平测量数据

(1) 先把每一次的数据通过进行旋转

$$\begin{pmatrix} x_i^{j0} \\ z_i^{j0} \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_i^j \\ z_i^j \end{pmatrix}, i=1 \cdots n, j=1, 2 \cdots 10$$

设为第  $j$  次的测量数据  $(x_i^j, z_i^j), i=1 \cdots n, j=1, 2 \cdots 10$ ,

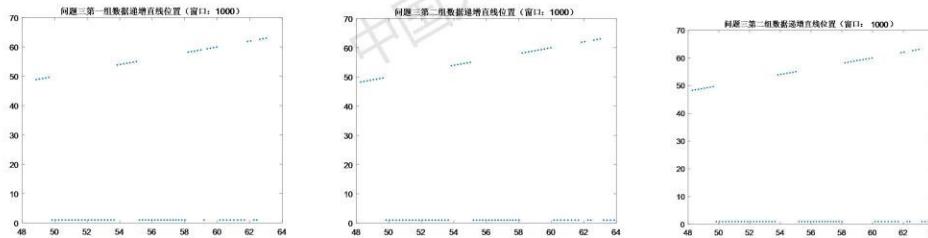
第  $j$  次旋转后的测量数据  $(x_i^{j0}, z_i^{j0}), i=1 \cdots n, j=1, 2 \cdots 10$ .

(2) 整合为一组数据

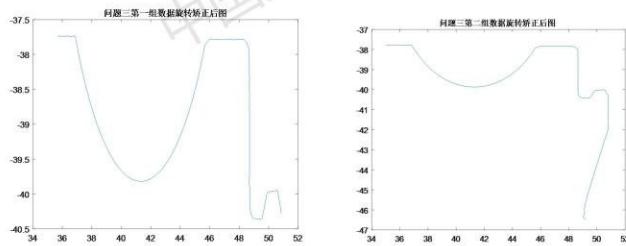
因为多次测量位置不同，每次测量的整个工件 2 的部分，对比第  $j$  次旋转后的测量数据  $(x_i^{j0}, z_i^{j0}), i=1 \cdots n, j=1, 2 \cdots 10$ . 得到工件 2 的水平校正后的整体数据  $(x_i^0, z_i^0), i=1 \cdots n$ ,

### 5.4.2 基于多组测量数据复原完整轮廓的方法具体求解步骤

**步骤 1：** 基于在问题 1 中建立起来的基于一阶二阶导数的模式识别方法和基于滑动平均的去噪方法对附件 2 中给出的 10 组测量数据进行直线或圆的图形类型的识别，并抓取每个图形所在的位置信息。



**步骤 2：** 基于步骤 1 获得的图形类型及其位置信息，挖掘每组数据的图形结构信息。具体地，首先确定每组中的直线有几条，圆有几个；其次，根据位置信息将这些直线和圆的前后位置关系搞清楚，从而对所测工件的结构有了框架性的认识。



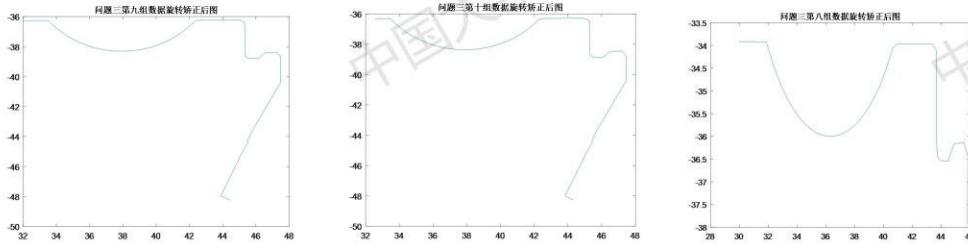
**步骤 3：** 基于每条直线的位置信息，抓取附件 2 中直线的原始数据。然后通过拟



合的方法，得到每条直线的方程。

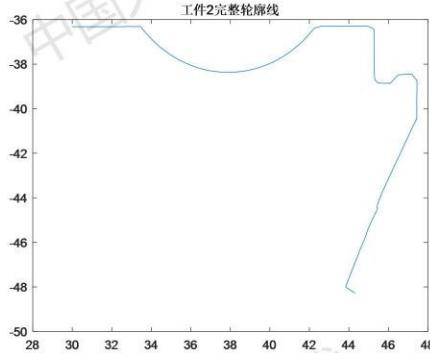
**步骤4：**根据直线的斜率相等与否，对每组数据中的直线进行分类。把每组中直线数量最多的类作为目标类。将目标类中直线的斜率作为本组数据的旋转角度 $\theta$ 。

**步骤5：**基于问题2中建立的直线旋转量的图像校正方法，以旋转角度 $\theta$ 对每组数据进行图像矫正。



**步骤6：**对每组数据矫正后的图像进行比较，选用每组数据中从左边起第一个圆的位置作为基准，对每组数据进行水平方法的平移矫正。

**步骤7：**对步骤6得到的矫正后的数据以横坐标范围的大小对数据进行同一整合，复原工件的轮廓。



依据问题1建立的方法，对复原之后的工件2进行测量，数据如下：

圆的圆心坐标：(37.8866, -32.5784) 半径：5.7886

水平直线（从左边看起）的长度：

第一条：3.8256； 第二条：3.3156； 第三条：0.8752； 第四条：1.1463

垂直直线（从上边看起）的长度：

第一条：2.7381； 第二条：1.9542

斜线（从上边看起）的长度：



第一条：9.0421；第二条：0.3862

## 5. 5 问题四的模型建立和求解

该问提供了多组局部的测试数据，包含了圆和角的多次测量。局部的测试数据相比第三问的全局测试数据更加精细，目的是为了解决全局测试时个别局部区域标注偏差较大的问题。基于此，本文采用一种基于多组局部数据修正全局图像的方法。

### 5. 5. 1 基于多组局部数据修正全局图像的方法

该方法分为以下几步：

步骤一：使用问题一的方法识别出每组局部数据的图形模式，即它是圆的一部分还是直线。

步骤二：使用问题二的方法，对该组局部数据进行校正。

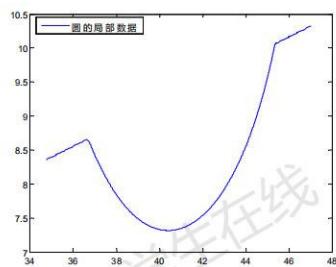
步骤三：使用校正后的局部数据与第三问全局数据进行等效匹配，从而确认校正后的局部数据是全局数据中的某一区域的放大。

步骤四：根据校正后的数据计算各个参数，并修正全局图形

步骤五：对其他各组局部数据重复以上步骤，得到修改后的完整工件图形。

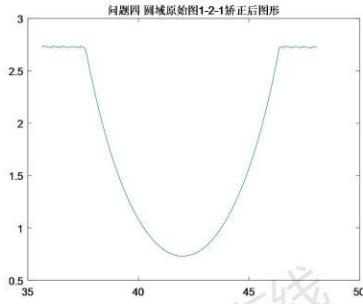
按照以上步骤，以第一组圆形数据为例进行处理，其余数据的处理结果见附录。

如下图所示，是第一组圆的原始局部数据图形：



然后使用问题一和问题二中的算法进行校正，得到下图：





与问题三中得到的全局图形的各个参数进行匹配，得到以上局部数据为全局图形中第一处圆弧的数据。

对以上数据进行拟合计算，求得圆的各个参数，如下表

**本问第一组局部数据圆弧参数**

横坐标	纵坐标	半径
0.4937	0.5229	0.9172

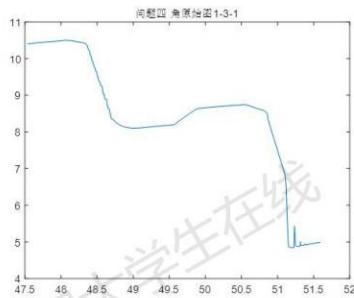
如下表是第三问全局数据的第一处圆弧参数

**第三问全局数据第一处圆弧参数**

横坐标	纵坐标	半径
0.4917	0.5259	0.9112

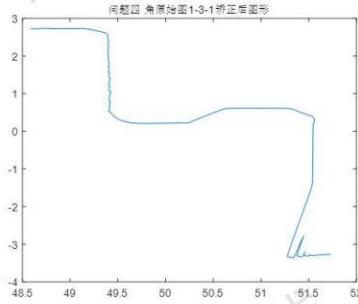
从以上数据可以看出，圆的局部数据与全局数据第一处圆弧的参数相差不大。可以使用该出局部数据的参数对全局数据的图形进行修正。

对第一处局部角度数据作相同处理，如下是第一处角的原始数据：

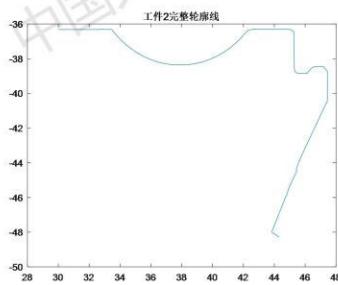


校正的图形如下：





使用以上方法，对剩余的各处圆弧和角度的局部数据进行处理，并对全局数据进行修改，得到如下修正后的工件轮廓图：



## 6 模型的评价

1. 整个问题的求解中，我们建立的基于一阶二阶导数的提取圆和直线的识别模型进行去噪处理可以重复使用，效果很好，可以很快的识别数据是圆还是直线。
2. 基于旋转量的图像矫正方法做了逆旋转进行比较检验效果好。
3. 所有模型的求解过程无需人工参与，实现的自动标注。
4. 如果想做到自动标注，需要做一个算法，实现如果是圆那么我们得到圆的参数  $R_1, R_2$ ，如果是直线得到直线的倾斜角，所有的轮廓部分都识别拟合完成后对函数求零点得到所有槽口宽度  $x_2, x_4$ ，水平线段的长度  $x_1, x_3$ ，通过圆心的参数得到圆心之间的距离  $c_1, c_3$ ，最后只需要最后求人字形的高度  $z_1$ 。

## 7 参考文献

- [1] 最小二乘法圆拟合 <https://wenku.baidu.com/>
- [2] 接触式三坐标测量自由曲面轮廓的数据处理模型 [J], 仇谷烽; 余景池; 黄启泰;
- [3] 李顺, 巩岩. 长程轮廓仪用于筒状超光滑表面测量的误差分析及校正[J].



光学学报, 2011(11):125–131.

## 附录:

```
clc, clear;
close all;
%导入数据
gj1=xlsread('附件 1_工件 1 的测量数据');
x_gj1=gj1(:, 1); z_gj1=gj1(:, 2);
plot(x_gj1, z_gj1)
```

%设置窗口大小及存放一阶导数、二阶导数、均值、方差等的数组

```
size_chuangkou=1000;
p=round(length(x_gj1)/size_chuangkou)-2;
m=ones(p, 1);
n=ones(p, 1);
x_new=ones(p, 1);
z_new=ones(p, 1);
```

```
zhi_m=ones(p, 1);
zhi_n=ones(p, 1);
zhi_x_new=ones(p, 1);
```

%求一阶导数 k1、二阶导数 k2 以及根据窗口去噪的方法求窗口均值和方差

```
for d=1:p
v=(d-1)*size_chuangkou;
k1=ones(size_chuangkou);
for i=1:size_chuangkou
k1(i)=(z_gj1(i+v+1)-z_gj1(i+v))/(x_gj1(i+v+1)-x_gj1(i+v));
```



```
end
```

```
k2=ones(size_chuangkou);  
for i=1:size_chuangkou  
    k2(i)=(k1(i+1)-k1(i))/(x_gj1(i+v+1)-x_gj1(i+v));  
end
```

```
m(d)=mean(k2(1:size_chuangkou));  
n(d)=var(k2(1:size_chuangkou));  
x_new(d)=x_gj1(d*size_chuangkou+1);  
z_new(d)=z_gj1(d*size_chuangkou+1);
```

```
zhi_m(d)=mean(k1(1:size_chuangkou));  
zhi_n(d)=var(k1(1:size_chuangkou));  
zhi_x_new(d)=x_gj1(d*size_chuangkou+1);  
end  
%plot(zhi_x_new(1:p), zhi_m(1:p), '.')  
%hold on
```

%设置二阶导数的窗口均值的阈值，抓取圆域所在的位置

```
x_circle=ones(p, 1);  
for i=1:p  
    if m(i)>3.5  
        x_circle(i)=x_new(i);  
    end  
end  
%plot(x_new(1:p), x_circle(1:p), '.')  
%hold on
```

%通过抓取到的圆域的位置，抓取圆域的原始数据以备拟合圆的方程之用



```

j=0;
x_nihe=ones(20000, 1);
z_nihe=ones(20000, 1);
for i=1:p
    if x_circle(i)>2
        for k=1:size_chuangkou
            x_nihe(j+k)=x_gj1(i*size_chuangkou+k);
            z_nihe(j+k)=z_gj1(i*size_chuangkou+k);
        end
        j=j+size_chuangkou;
    end
end

```

%圆的拟合计算圆心，半径

```

%xx=xlsread('新建 Microsoft Excel 工作表');
x=x_gj1(10001:13000);y=z_gj1(10001:13000);
N=length(x);
C=N*sum(x.^2)-sum(x)*sum(x);
D=N*sum(x.*y)-sum(x)*sum(y);
E=N*sum(x.^3)+N*sum(x.*y.^2)-sum(x.^2+y.^2)*sum(x);
G=N*sum(y.^2)-sum(y)*sum(y);
H=N*sum(x.^2.*y)+N*sum(y.^3)-sum(x.^2+y.^2)*sum(y);
a=(H*D-E*G)/(C*G-D.^2);
b=(H*C-E*D)/(D.^2-G*C);
c=-(sum(x.^2+y.^2)+a*sum(x)+b*sum(y))/N;
A=a/(-2);
B=b/(-2);
R=1/2*sqrt(a.^2+b.^2-4*c);

```

%设置一阶导数窗口均值的阈值，抓取直线所在的位置



```

x_zhi=ones(p);
for i=1:p
    if zhi_m(i)>0.15
        x_zhi(i)=zhi_x_new(i);
    end
    if x_circle(i)>2
        x_zhi(i)=1;
    end
end
plot(zhi_x_new(1:p), x_zhi(1:p), '.')
hold on
title('level 水平直线位置 (窗口: 1000; )')

%通过抓取到的直线的位置，抓取直线的原始数据以备拟合直线的方程之用
s=0;
x_nihe_zhi=ones(20000,1);
z_nihe_zhi=ones(20000,1);
for i=1:p
    if x_zhi(i)>2
        for k=1:size_chuangkou
            x_nihe_zhi(s+k)=x_gj1(i*size_chuangkou+k);
            z_nihe_zhi(s+k)=z_gj1(i*size_chuangkou+k);
        end
        s=s+size_chuangkou;
    end
end
%直线的拟合求出斜率和截距
x=x_gj1(22301:23500)'; y=z_gj1(22301:23500)';

```



```
%x=x_nihe_zhi(17101:19700)';y=z_nihe_zhi(17101:19700)';
```

```
a = x*x' ;
```

```
b = sum(x) ;
```

```
c = x*y' ;
```

```
d = sum(y) ;
```

```
N=length(x) ;
```

```
% 求解斜率 k
```

```
k = (N.*c-b*d)./(N.*a-b*b);
```

```
% 求解截距 t
```

```
t = (a.*d-c.*b)/(a*N-b.*b);
```

```
plot(x,y,'+')
```

```
hold on
```

```
plot(x,k*x+t,'.')
```

```
title('level 第二条递减直线(从左边看)')
```

```
legend('实际','拟合')
```

```
%图形的旋转矫正和水平偏移矫正
```

```
xuanZhuanJiao=0.130361871364958;
```

```
x_jiaoZheng=cos(xuanZhuanJiao)*x_gj1_down-sin(xuanZhuanJiao)*z_gj1_do
```

```
wn;
```

```
z_jiaoZheng=sin(xuanZhuanJiao)*x_gj1_down+cos(xuanZhuanJiao)*z_gj1_do
```

```
wn;
```

```
z_weitiao=mean(z_jiaoZheng)-mean(z_gj1_level);
```

```
x_weitiao=mean(x_jiaoZheng)-mean(x_gj1_level);
```

```
x_pianyi=x_weitiao+0.33;
```

```
z_jiaoZheng_chuizhi=z_jiaoZheng-z_weitiao;
```

```
x_jiaoZheng_shuiping=x_jiaoZheng-x_pianyi;
```

```
plot(x_gj1_level,z_gj1_level)
```

```
hold on
```



```

plot(x_jiaoZheng_shuiping, z_jiaoZheng_chuizhi)
title(' level 图与进一步水平矫正后 down 图的比较')
legend(' level', ' down 水平矫正后')

%问题中参数的计算

y0=mean(z_gj1(14690:21690));
kd1=-2.723867741639066; bd1=133.8002234310488;
kd2=-3.724845609376202; bd2=213.0069757236736;
kd3=-3.723602901285295; bd3=239.2514907860618;
kd4=-0.200744158613707; bd4=14.645116449010247;
kd5=-0.565025739534814; bd5=45.935780974909875;
kd6=-1.123311555044076; bd6=98.725881910432650;
kd7=-0.343117911107307; bd7=29.331114981311476;
kd8=-0.425604356360074; bd8=47.817846154298124;

ku1=2.754359108225884; bu1=-146.8171658730294;
ku2=2.980598439534949; bu2=-179.8507093519987;
ku3=3.511651703755757; bu3=-236.2555247229892;
ku4=0.194764774178566; bu4=-15.745625965295059;
ku5=1.312714609538673; bu5=-115.4903009149167;
ku6=1.119240436916190; bu6=-99.776517012025040;
ku7=1.472580398274092; bu7=-151.5742987893202;
ku8=1.001404926554808; bu8=-108.2696306825218;

k2u1=1.873469412912976; b2u1=-102.8003499165892;
k2u2=2.441766357967081; b2u2=-152.1327739769671;
k2u3=2.414152499053679; b2u3=-168.2737582070639;

k2d1=-0.129926436840446; b2d1=7.740846276449202;

```



```
k2d2=-0.130937034696584; b2d2=7.799039243107876;  
k2d3=-0.131141148034880; b2d3=7.813060406404597;  
k2d4=-0.131105391993604; b2d4=7.811663609602934;
```

%下降曲线的折现点横坐标

```
td1=(y0-bd1)/kd1;  
td2=(y0-bd2)/kd2;  
td3=(y0-bd3)/kd3;  
td4=(y0-bd4)/kd4;  
td5=(y0-bd5)/kd5;  
td6=(y0-bd6)/kd6;  
td7=(y0-bd7)/kd7;  
td8=(y0-bd8)/kd8;
```

%上升曲线的折现点横坐标

```
tu1=(y0-bu1)/ku1;  
tu2=(y0-bu2)/ku2;  
tu3=(y0-bu3)/ku3;  
tu4=(y0-bu4)/ku4;  
tu5=(y0-bu5)/ku5;  
tu6=(y0-bu6)/ku6;  
tu7=(y0-bu7)/ku7;  
tu8=(y0-bu8)/ku8;
```

```
x1=tu1-td1;  
x2=td2-tu1;  
x3=tu2-td2;  
x4=td3-tu2;  
x5=tu3-td3;
```



```

x6=tu4-tu3;
x7=td4-tu4;
x8=td5-td4;
x9=tu6-tu5;
x10=td7-td6;
x11=tu7-td6;
x12=tu8-tu7;
x13=td8-tu8;

j1=atan(kd1)*180/3.141592+180;
j2=180-atan(ku1)*180/3.141592;
j3=atan(kd2)*180/3.141592+180;
j4=180-atan(ku2)*180/3.141592;
j5=atan(kd3)*180/3.141592+180;
j6=180-atan(ku3)*180/3.141592;
j7=180-atan(ku4)*180/3.141592;
j8=atan(kd4)*180/3.141592+180;

xuanzhuanjiao1=atan(ku1)*180/3.141592-atan(k2u1)*180/3.141592;
xuanzhuanjiao2=atan(ku2)*180/3.141592-atan(k2u2)*180/3.141592;
xuanzhuanjiao3=atan(ku3)*180/3.141592-atan(k2u3)*180/3.141592;

xuanzhuanjiao4=atan(k2d1)*180/3.141592;
xuanzhuanjiao5=atan(k2d2)*180/3.141592;
xuanzhuanjiao6=atan(k2d3)*180/3.141592;
xuanzhuanjiao7=atan(k2d4);

y_jian=bu4+ku4*(bd4-bu4)/(ku4-kd4)-y0;

x_jian=(bd4-bu4)/(ku4-kd4);

```



支撑材料的文件列表：

1. down 矫正之后的数据\_new.xlsx
2. 程序整理.txt
3. 第二问 相关图片.docx
4. 第三问轮廓重构后数据.xlsx
5. 第四问相关图片圆.docx
6. 问题三的结果.docx

