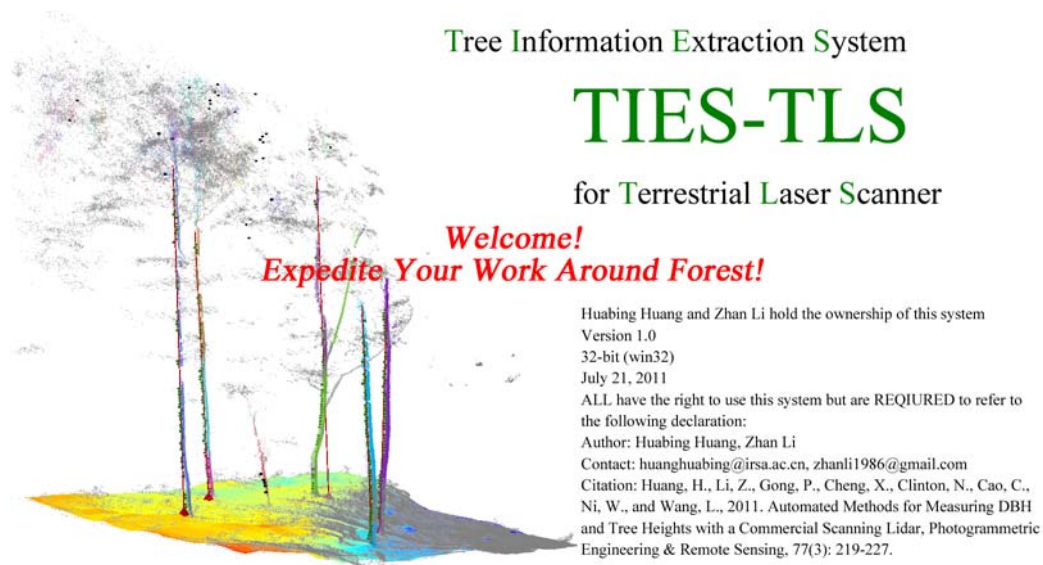


基于地面激光雷达的树木信息提取系统

使用手册

1 欢迎使用



2 版权说明

Version 1.0

32-bit (win32)

2011 年 7 月 21 日

本系统的所有权属于黄华兵、李展。

任何人均拥有本系统的使用权，但是被要求引用如下信息：

作者：黄华兵，李展

联系方式：huanghuabing@irsa.ac.cn，zhanli1986@gmail.com

引用文献：Huang, H., Li, Z., Gong, P., Cheng, X., Clinton, N., Cao, C., Ni, W., and Wang, L., 2011. Automated Methods for Measuring DBH and Tree Heights with a Commercial Scanning Lidar, Photogrammetric Engineering & Remote Sensing, 77(3): 219-227.

任何人均拥有本系统的修改权，作者欢迎任何用户的任何修改意见和修改工作，并和您联系讨论修改和完善工作。

3 系统介绍

基于地面激光雷达的树木信息提取系统，用于处理森林地区的地面激光雷达扫描数据，并从中提取树木，特别是单木的结构信息，包括树木的胸径、树高、地表位置以及树干不同高度的中心位置和半径。

本系统是一套数据处理系统，具有以下主要功能：激光点云滤波、栅格化点云、单株树分离、树干结构信息提取、多站扫描数据的自动配准。本系统不具备数据显示功能。

4 安装说明

4.1 系统要求

操作系统

Windows XP SP2。

运行平台

MATLAB7.0 或以上版本。

内存要求

最小 512M，根据处理数据的大小，内存要求不断上升，最大和 MATLAB 的内存设置一致。

4.2 安装过程

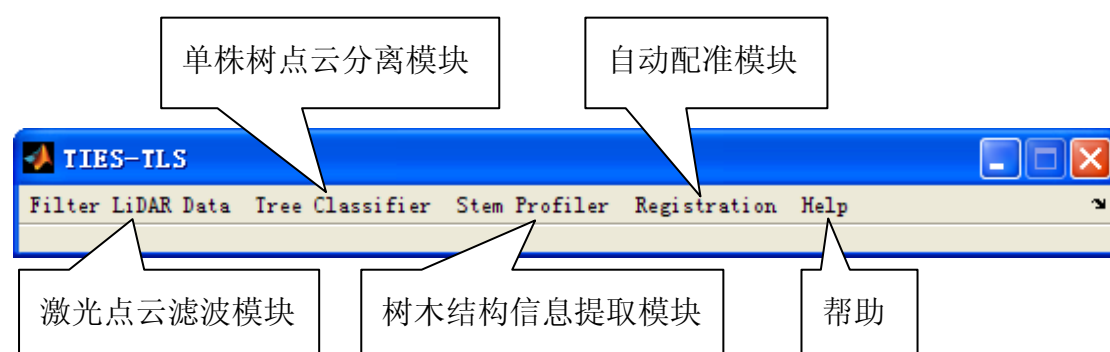
1. 安装 MATLAB7.0 或以上版本。
2. 将软件压缩包解压，双击文件“TIES_TLS.fig”即可运行系统。

5 入门

数据文件类型

本系统运行过程中，使用的文件类型均为：**文本文件 (*.txt)**。请根据各个功能模块的要求，将激光点云数据在其他相关软件中转换为相应的文本文件后，输入本系统进行处理。

主界面窗口



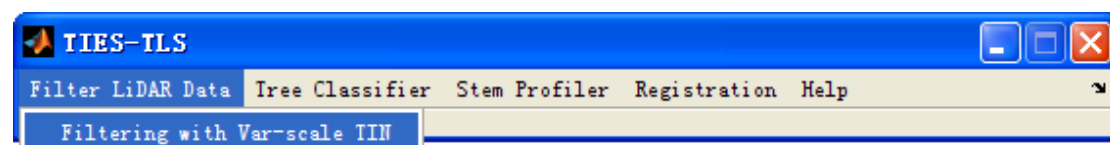
激光点云的坐标系

本系统要求激光点云的三维坐标值是如下坐标系中的取值：该坐标系统的 z 轴平行或者近似平行大部分的树干方向，即 $x-o-y$ 平面垂直于大部分的树干，或者说 $x-o-y$ 平面与地面近似平行，并且 z 轴正方向指向树顶。

6 激光点云滤波

6.1 变尺度不规则三角网滤波

菜单选择



界面窗口

Filter Points with Var-TIN

Input Files
 ASCII file of point clouds ...

Var-TIN Parameters
 Number of Scales
 Input the cell size and distance thresholds in descending order into the below table:

Scale NO.	Cell Size	Distance Threshold
1	4	3
2	2	1.5000
3	1	0.5000
4	0.5000	0.2000

Cut Height
☐ Specify Cut Height with a Given Value:
 The Given Cut Height
☒ Calculate Cut Height with the Parameters:
 Slice Thickness
 Coefficient
 The Calculated Cut Height:

Data Extent to Be Filtered
☒ Full Extent of Data
☐ Specify Extent
 Minimum X Maximum Y
 Minimum Y Maximum X

Output Files
 ASCII file of ground points ...
☐ Output non-ground points into the ASCII file
 ...
☐ Output the remaining points in all iterations of different scales besides the least one
 The remaining points in the iteration of the least scale are ground points
 Output Directory ...

文件格式

输入文件:

1. 激光雷达扫描的点云数据，文本文件，每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x、y、z 坐标值。

格式示例:

```
13.178963  0.022621  9.144001
13.186680  0.022706  9.115335
13.180342  0.022757  9.077018
.....
```

输出文件:

1. 地面的点云数据，文本文件，每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x 、 y 、 z 坐标值。

格式示例：

```
13.462655  0.057078  -0.893276
13.458093  0.057156  -0.916534
13.471447  0.057324  -0.941036
.....
```

2. 非地面的点云数据，可选，文本文件，每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x 、 y 、 z 坐标值。

格式示例：

```
0.642219  0.019371  11.427914
15.556318  0.023140  12.987809
13.665884  0.019556  11.366869
.....
```

3. 滤波过程中，每一网格大小下获得的地面点云数据（地面点云数据即最小的网格下获得的点云数据），可选，文本文件，每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x 、 y 、 z 坐标值。

格式示例：

```
13.449423  0.047151  1.458556
13.419143  0.047107  1.431609
13.452458  0.047335  1.411462
.....
```

参数说明

1. **Number of Scales:** 尺度个数。即滤波采用的网格尺寸及其对应的距离阈值的个数。
2. **Cell Size:** 网格尺寸。采用给定大小的网格对点云在 x - o - y 平面上的投影进行栅格化操作，从每个栅格中寻找最低点。
3. **Distance Threshold:** 距离阈值。如果一个点到对应的三角面的距离小于该阈值，则这个点被归为地面点，否则将其排除。

4. **Cut Height:** 截断高度。在最小网格下（最后一次迭代操作中），如果一个点的 z 值大于截断高度，则该点一定不是地面点，强制排除，否则，将其作为候选地面点。该参数可以人为给定（Specify Cut Height with a Given Value），也可以让算法自动确定，算法自动确定的方法是：将点云按照给定的切片厚度（Slice Thickness）分为许多个平行于 $x-o-y$ 平面的层，统计每层的点云个数，找出峰值位置，峰值高度和最低点高度之间的距离乘以给定系数（Coefficient）之后，加上峰值高度，作为截断高度。
5. **Data Extent to Be Filtered:** 滤波范围。可以使用整个点云数据（Full Extent of Data），也可以指定范围（Specify Extent），请输入 x 、 y 的最小值和最大值。

算法说明

变尺度不规则三角网滤波算法引入变尺度栅格和变距离阈值，先以较大的栅格和距离阈值滤除高度较大的地物，逐步减小栅格和距离阈值时，滤除高度较小的地物，并同时保留地表的微小起伏，从而生成高精度的数字高程模型。步骤如下：

- (a) 给定初始的栅格大小和距离阈值。
- (b) 将激光点云在 $x-o-y$ 平面上的投影转化为栅格图像，寻找每个像元中的最低点，由这些最低点构成地面点。
- (c) 利用地面点生成不规则三角网。
- (d) 计算任一激光点到不规则三角网表面的距离，如果该距离小于给定的距离阈值，则记录该点为地面点。
- (e) 利用找到的地面点生成新的激光点云，减小栅格大小和距离阈值，重复步骤（b）~（d），直到栅格大小和距离阈值减小到给定的最小值，生成最终的地面点。

关于生成DEM的说明

本系统不提供生成数字高程模型（DEM）的功能，很多软件都能根据输入的地面点生成 DEM。

以 ArcGIS 为例，将生成的地面点云数据导入 ArcGIS 中生成 shp 文件，然后使用“3D Analyst Tools -> TIN Creation”生成不规则三角网 TIN，利用“3D Analyst Tools -> Conversion”将 TIN 转换为栅格数据(Raster)，最后使用“Conversion Tools”

将栅格数据转换为文本文件（ASCII）。

在本系统的其他部分模块中，需要使用扫描数据范围内的 DEM 数据，DEM 采用文本文件输入，DEM 文本文件的格式和 ArcGIS 的“Conversion Tools”转换得到的文本文件格式一致，如下：

NCOLS xxx

NROWS xxx

XLLCENTER xxx | XLLCORNER xxx

YLLCENTER xxx | YLLCORNER xxx

CELLSIZE xxx

NODATA_VALUE xxx

row 1

row 2

.

.

.

row n

格式示例：

NCOLS 480

NROWS 450

XLLCORNER 378922

YLLCORNER 4072345

CELLSIZE 30

NODATA_VALUE -32768

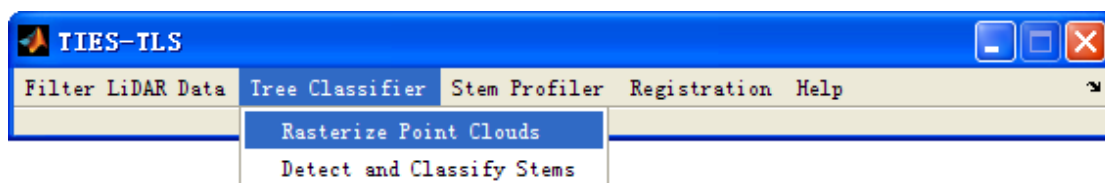
43 2 45 7 3 56 2 5 23 65 34 6 32 54 57 34 2 2 54 6

35 45 65 34 2 6 78 4 2 6 89 3 2 7 45 23 5 8 4 1 62

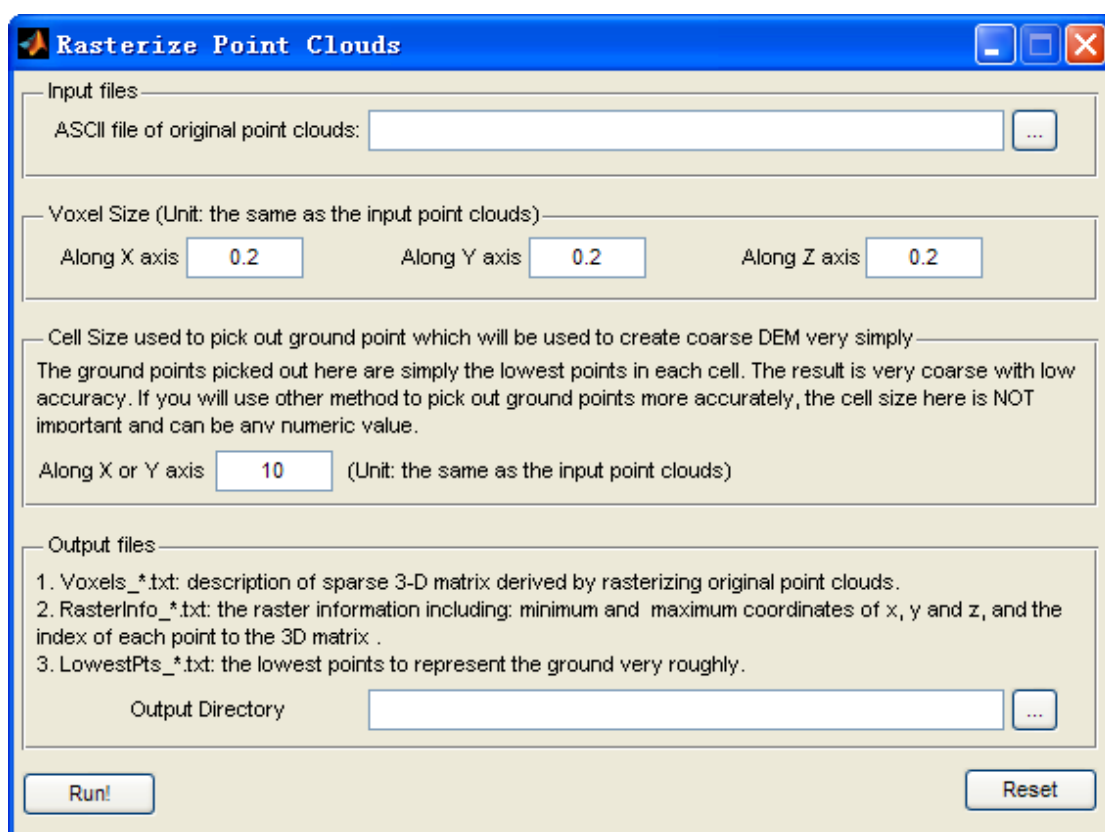
7 单株树点云分离模块

7.1 栅格化点云

菜单选择



界面窗口



文件格式

输入文件:

1. 激光雷达扫描的点云数据，文本文件，每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x、y、z 坐标值。

格式示例:


```
13.178963  0.022621  9.144001
13.186680  0.022706  9.115335
13.180342  0.022757  9.077018
.....
```

输出文件:

1. 体素数据，文本文件，文件名前缀“Voxels_”。

第一行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最小 x、y、z 值；

第二行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最大 x、y、z 值；

第三行存储由空格符（空格或者制表符）分隔的三个数值，代表体素的尺寸，x、y、z 方向上的边长；

第四行存储由空格符（空格或者制表符）分隔的三个数值，代表点云栅格化之后，y、x、z 方向上的栅格个数（行数、列数和层数）；

第五行及其之后每一行存储由空格符（空格或者制表符）分隔的四个数值，记录每个包含点云的体素的位置，依次是行（y 轴位置）、列（x 轴位置）、层（z 轴位置），以及该体素包含的激光点的个数。

格式示例：

```
-96.676453 -80.120293 -4.236923
99.439079  84.000381  24.501331
0.200000   0.200000   0.200000
821 981 144
279 1   58  1
283 3   57  1
286 3   55  1
.....
```

2. 栅格信息，文本文件，文件名前缀“RasterInfo_”。

第一行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最小 x、y、z 值；

第二行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最大 x 、 y 、 z 值；

第三行存储由空格符（空格或者制表符）分隔的三个数值，代表体素的尺寸， x 、 y 、 z 方向上的边长；

第四行及其之后每一行存储一个数值，记录每个激光点所在体素的索引位置，1 行 1 列 1 层的索引为 1，沿列计数，然后从第一层向上数，获取体素索引。

格式示例：

```
-96.676453 -80.120293 -4.236923
99.439079 84.000381 24.501331
0.200000 0.200000 0.200000
63274050
69725468
63274050
.....
```

3. 最低点，文本文件，文件名前缀 “LowestPts_”。记录每个栅格中的最低点文件，用于生成粗精度的 DEM。

第一行是字段名 “ x , y , z ”；

第二行及其之后每一行存储由逗号分隔的三个数值，记录地面点的 x 、 y 、 z 坐标值。

格式示例：

```
x, y, z
-71.676453, -75.120293, 11.262774
-61.676453, -75.120293, 6.767901
-51.676453, -75.120293, 6.449819
.....
```

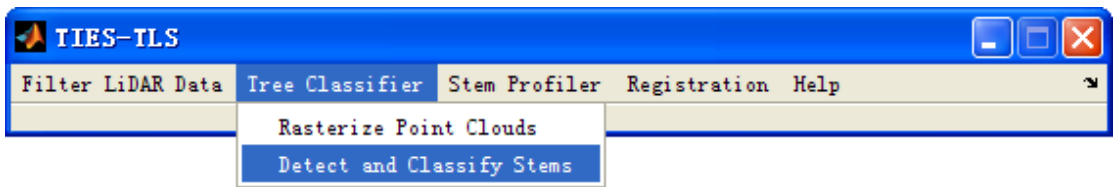
参数说明

1. Voxel Size: 体素的尺寸。
2. Cell Size used to pick out ground point which will be used to create coarse DEM

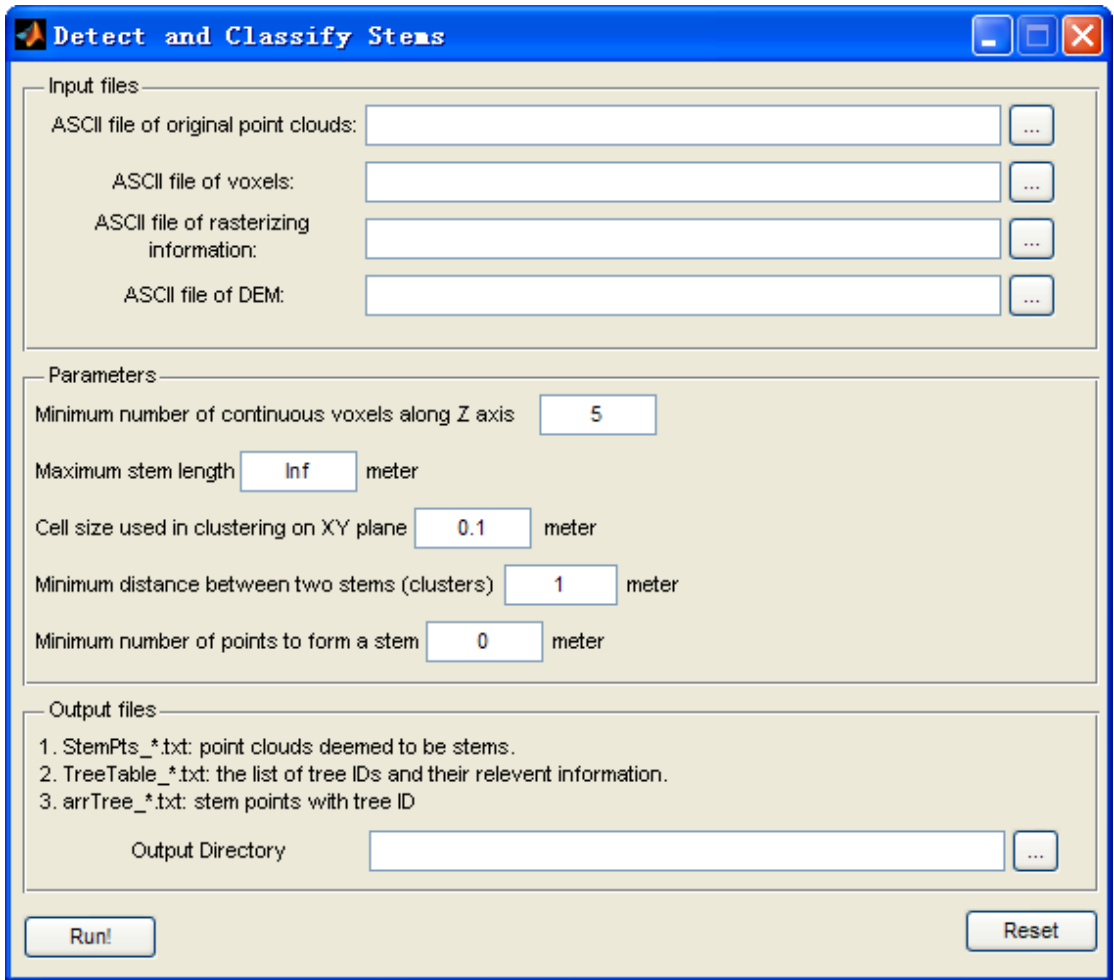
very simply: 在该尺寸下栅格化点云，寻找每个栅格中的最低点，作为地面点，用于生成粗精度的 DEM。

7.2 检测和分离单株树点云

菜单选择



界面窗口



文件格式

输入文件:

1. 激光雷达扫描的点云数据，文本文件，每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x 、 y 、 z 坐标值。

格式示例：

```
13.178963  0.022621  9.144001
13.186680  0.022706  9.115335
13.180342  0.022757  9.077018
.....
```

2. 体素数据，文本文件。

第一行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最小 x 、 y 、 z 值；

第二行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最大 x 、 y 、 z 值；

第三行存储由空格符（空格或者制表符）分隔的三个数值，代表体素的尺寸， x 、 y 、 z 方向上的边长；

第四行存储由空格符（空格或者制表符）分隔的三个数值，代表点云栅格化之后， y 、 x 、 z 方向上的栅格个数（行数、列数和层数）；

第五行及其之后每一行存储由空格符（空格或者制表符）分隔的四个数值，记录每个包含点云的体素的位置，依次是行（ y 轴位置）、列（ x 轴位置）、层（ z 轴位置），以及该体素包含的激光点的个数。

格式示例：

```
-96.676453 -80.120293 -4.236923
99.439079  84.000381  24.501331
0.200000   0.200000   0.200000
821 981 144
279 1  58 1
283 3  57 1
286 3  55 1
.....
```

3. 栅格信息，文本文件。

第一行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最小 x、y、z 值；

第二行存储由空格符（空格或者制表符）分隔的三个数值，代表输入的点云数据的最大 x、y、z 值；

第三行存储由空格符（空格或者制表符）分隔的三个数值，代表体素的尺寸，x、y、z 方向上的边长；

第四行及其之后每一行存储一个数值，记录每个激光点所在体素的索引位置，1 行 1 列 1 层的索引为 1，沿列计数，然后从第一层向上数，获取体素索引。

格式示例：

```
-96.676453 -80.120293 -4.236923
99.439079 84.000381 24.501331
0.200000 0.200000 0.200000
63274050
69725468
63274050
.....
```

4. DEM 数据，文本文件，地表位置信息。

格式说明：

```
NCOLS xxx
NROWS xxx
XLLCENTER xxx | XLLCORNER xxx
YLLCENTER xxx | YLLCORNER xxx
CELLSIZE xxx
NODATA_VALUE xxx
row 1
row 2
.
.
.
```

row n

格式示例：

NCOLS 480

NROWS 450

XLLCORNER 378922

YLLCORNER 4072345

CELLSIZE 30

NODATA_VALUE -32768

43 2 45 7 3 56 2 5 23 65 34 6 32 54 57 34 2 2 54 6

35 45 65 34 2 6 78 4 2 6 89 3 2 7 45 23 5 8 4 1 62

输出文件：

1. 树干点云，文本文件，文件名前缀“StemPts_”。每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x、y、z 坐标值。

格式示例：

13.178963 0.022621 9.144001

13.186680 0.022706 9.115335

13.180342 0.022757 9.077018

.....

2. 分离出来的树干列表，文本文件，文件名前缀“TreeTable_”。每一行存储由空格符（空格或者制表符）分隔的四个数值：一棵树的点云聚类中心（依次是中心的 x、y 坐标），该棵树包含的激光点个数，树木的编号。

格式示例：

-80.815018 -56.673392 12 1

-78.848351 -52.273392 8 2

-75.968351 -55.673392 13 3

.....

3. 单株树分离后的树干点云，文本文件，文件名前缀“arrTree_”。每一行存储由空格符（空格或者制表符）分隔的四个数值，记录一个点的 x、y、z 坐标值，以及该点所属的树木的编号。

参数说明

1. Minimum number of continuous voxels along Z axis: 当含有点云的体素连续分布超过这个数值, 这些体素内的激光点被记录为树干点云。
2. Maximum stem length: 只有从最低点向上位于这个长度之内的激光点被归为树干点云。
3. Cell size used in clustering on XY plane: 分离单株树时, 为了提高效率, 采用栅格聚类的方式, 这是给定的栅格尺寸。
4. Minimum distance between two stems (clusters): 两棵树之间的最小距离, 当两个像元集群中心之间的距离小于该数值, 合并为一棵树, 大于该数值, 标记为两棵树。
5. Minimum number of points to form a stem: 构成一棵树的激光点的最小个数。

算法说明

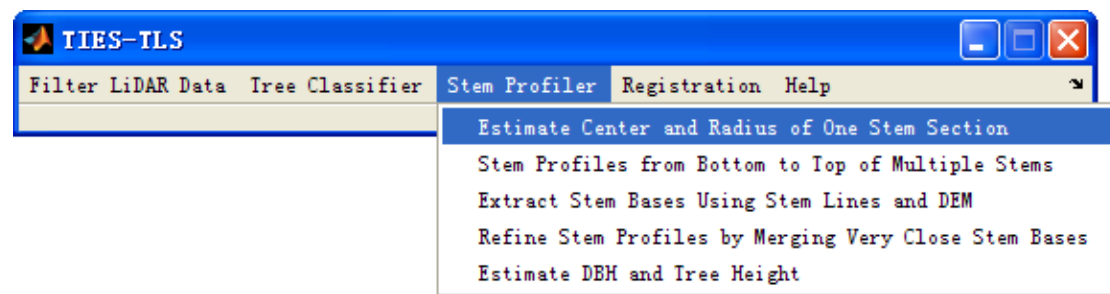
1. 提取树干点云的三维体素直方图算法。根据给定的三维立方体(体素)大小, 将激光点云栅格化, 对 $x-o-y$ 平面上的每个栅格位置, 沿 z 轴搜寻含有点云的连续体素, 当连续体素的个数大于给定阈值时, 将这些体素内的点云记录为树干点云。

2. 分离单株树的聚类算法。该算法假设树干为直线, 并近似垂直于 $x-o-y$ 平面, 那么同一个树干的点云在 $x-o-y$ 平面上的投影构成一个集群, 不同树干的点云构成不同的集群。为了提高算法效率, 给定栅格大小, 将点云在 $x-o-y$ 平面上的投影转换为栅格图像, 对含有点云投影的像元进行聚类处理, 如果两个像元集群中心的距离小于给定阈值, 则被合并为一个集群, 这些像元中的点云被标记为来自同一个树干; 否则, 这两个集群中的像元包含的点云被标记为来自不同的树干。

8 树木结构信息提取模块

8.1 从一个树干点云切片中估计树干中心和半径

菜单选择



界面窗口

Input files

ASCII file of points in a stem section

Parameters

Cell size used in Hough transform meter

Maximum possible radius pixels Minimum number of points to give a circle

Coefficient of standard deviation of radius to define ROI

Minimum change of center position and radius meter Maximum count of iterations

☐ Process Input Points as Single Scan Data
 ☒ Process Input Points as Complete Scan Data

If processing the input point clouds as single scan data, please provide the position of the scanner in the coordinate system of the input point clouds to remove points outside the range of a single scan, represented by the half space with the scanner.

X	Y	Z
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

Estimation Outputs for This Stem Section

Estimate Circle By Circle Fitting (Unit: the same as the input points)

Center_X	Center_Y	Center_Z	Radius
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Estimate Arc Angle of Stem Section (Unit: the same as the input points)

#1Endpoint of Arc (Radian)	#2Endpoint of Arc (Radian)	Angle of Stem Section (Radian)
<input type="text"/>	<input type="text"/>	<input type="text"/>

Circle By Hough Transform (Unit: the same as the input points)

Center_X	Center_Y	Radius	Std of Radius
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Information about Iterations (Unit: the same as the input points)

Iteration Count	Final Change of Center	Final Change of Radius	Point # in Circle Fitting
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Initial Point # Remark for the Estimation Result:

文件格式

输入文件:

1. 一个树干点云切片，一个树干的具有一定厚度的点云切片，文本文件。每一行存储由空格符（空格或者制表符）分隔的三个数值，代表一个点的 x、y、

z 坐标值。

格式示例：

```
13.178963  0.022621  9.144001
13.186680  0.022706  9.115335
13.180342  0.022757  9.077018
.....
```

参数说明

1. **Cell size used in Hough transform:** 使用该栅格尺寸栅格化点云在 x-o-y 平面上的投影，该栅格图像用于哈夫变换。
2. **Maximum possible radius:** 半径的取值范围，单位是像元个数，请使用最大可能的半径取值除以栅格尺寸确定该值。
3. **Minimum number of points to give a circle:** 输入检测到的有效激光点（落在感兴趣区 ROI 中的点）少于该数值，不足以得到中心和半径的可靠估计，点云没有构成圆或者圆弧。
4. **Coefficient of standard deviation of radius to define ROI:** 确定ROI的系数 $Coef_{\sigma}$ ，详见算法说明。
5. **Minimum change of center position and radius:** 当中心位置和半径的估计值的变化小于该阈值时，停止迭代算法。
6. **Maximum count of iterations:** 迭代的最大次数。
7. **The position of the scanner (X, Y, Z):** 扫描仪在点云本地坐标系下的坐标位置。
8. **Estimate Circle By Circle Fitting:** 最后一步迭代中圆拟合得到的中心和半径，是我们所需要的树干中心和半径。
9. **Estimate Arc Angle of Stem Section:** 该树干切片构成的圆弧的大小。
10. **Circle By Hough Transform:** 最后一步迭代中哈夫变换估计的中心和半径，中间结果，供观察和分析。
11. **Information about Iterations:** 包括：迭代次数、中心位置的最终变化值、半径的最终变化值、最终圆拟合使用的激光点个数
12. **Initial Point #:** 切片中点云的原始个数。

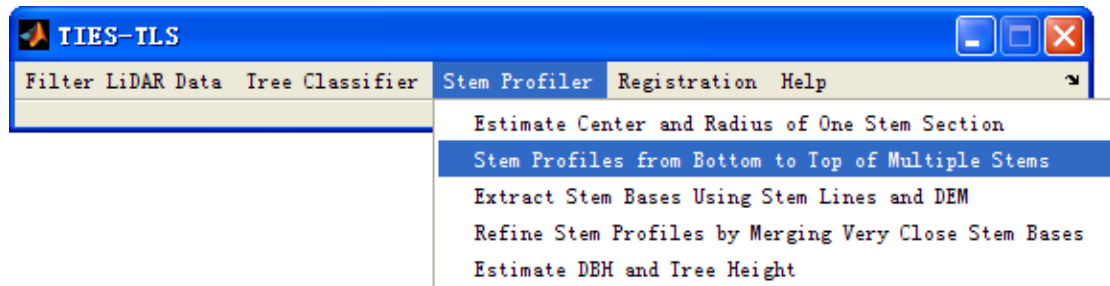
算法说明

估计树干中心和半径的哈夫变换和圆拟合相结合的迭代算法。该算法将具有给定厚度的树干点云切片近似为不完整的圆，将 $x-o-y$ 平面上的点云投影转换为栅格图像，采用哈夫变换识别是否存在圆并去除噪声点，利用剩下的点，采用圆拟合估计树干中心和半径。步骤如下：

- (a) 迭代次数记为 k ，设定初值为 1。对某个树干点云切片，利用切片内的点在 $x-o-y$ 平面上的投影点，采用圆拟合（最小二乘法）估计树干中心 $O_{ls,k-1}$ 和半径 $R_{ls,k-1}$ 。
- (b) 将点云在 $x-o-y$ 平面上的投影转换为栅格图像，含有点云的像元为前景像元。
- (c) 采用哈夫变换检测栅格图像中的前景像元是否构成一个圆或者圆弧，如果没有检测到圆或者圆弧，则算法结束，该点云切片无效；如果检测到圆或者圆弧，得到中心 $O_{HT,k-1}$ 和半径 $R_{HT,k-1}$ 。
- (d) 计算 $O_{HT,k-1}$ 到每个前景像元的距离，计算这些距离的标准差 $\sigma_{D,k-1}$ 。
- (e) 计算 $O_{HT,k-1}$ 到每个激光点的距离，将距离大于 $R_{HT,k-1} - Coef_{\sigma} \cdot \sigma_{D,k-1}$ 、小于 $R_{HT,k-1} + Coef_{\sigma} \cdot \sigma_{D,k-1}$ 的点作为有效点，之外的点作为噪声点，采用圆拟合利用有效点估计新的树干中心 $O_{ls,k}$ 和半径 $R_{ls,k}$ 。
- (f) 计算中心的变化值： $O_{ls,k-1}$ 和 $O_{ls,k}$ 之间的距离，半径的变化值： $R_{ls,k-1}$ 和 $R_{ls,k}$ 之间的差值，如果中心和半径的变化均小于给定阈值 ε ，输出树干中心 $O_{ls,k}$ 和半径 $R_{ls,k}$ ；否则，去除离树干中心 $O_{HT,k-1}$ 的距离大于 $R_{HT,k-1} + Coef_{\sigma} \cdot \sigma_{D,k-1}$ 的前景像元，更新栅格图像；然后更新 k 值为 $k+1$ ，利用更新后的栅格图像，重复步骤（c）~（f）直到中心和半径的变化小于给定阈值，或者迭代次数达到给定的最大值 $MaxCount$ 。当达到最大迭代次数时，中心和半径的变化仍然大于给定阈值，则表明该点云切片无效。

8.2 获取多个树干的剖面（从树干最低点到最高点的树干中心和半径）

菜单选择



界面窗口

Extract Stem Profiles from Bottom to Top

Input files—
 ASCII file of point clouds with TREE ID ...

Parameters—

Cell size used in Hough transform meter Slice thickness along Z axis meter

Maximum possible radius pixels Minimum number of points to give a circle

Coefficient of standard deviation of radius to define ROI

Minimum change of center position and radius meter Maximum count of iterations

Angle resolution of theta (rotate about Z axis) degree

Angle resolution of phi (rotate about X axis) degree

Ratio of minimum number to theoretical number of points in a stem arc

Maximum distance between stem centers to fitted line meter

Minimum number of points to give a line of stem

☐ Process Input Point Clouds as Single Scan Data ☒ Process Input Point Clouds as Complete Scan Data

If processing the input point clouds as single scan data,
 please provide the position of the scanner in the coordinate system of the input point clouds
 to remove points outside the range of a single scan, represented by the half space with the scanner.

X	Y	Z
<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>

Output files—

1. StemCenter_*.txt: the stem centers by estimating stem profiles from bottom to up.
2. ReliableStemCenterByPtsNum_*.txt: the reliable stem centers picked out by minimum point number in each stem section.
3. ReliableStemCenterByPtsNum&FitLine_*.txt: the reliable stem centers pick out by minimum point number in each stem section and fitting lines.
4. FittedStemLine_*.txt: the parameters of fitted line of stems.

Output Directory ...

文件格式

输入文件:

1. 单株树分离后的树干点云，文本文件。每一行存储由空格符（空格或者制表符）分隔的四个数值，记录一个点的 x、y、z 坐标值，以及该点所属的树木的编号。

输出文件:

1. 初始树干中心和半径，文本文件，文件名前缀“StemCenter_”。每一行存储由空格符（空格或者制表符）分隔的七个数，依次记录各个树干各个高度上的：树干中心和半径（y、x、半径、z），树木编号，最终圆拟合使用的激光点个数，树干切片构成的圆弧大小（单位：弧度）。

格式示例：

```
10.065695 -38.322525 0.419517 3.181771 75 9 1.651437
-5.969211 -35.383311 0.210647 1.645929 97 9 1.486139
-8.125523 -25.828318 0.144873 -0.119948 175 12 2.394584
.....
```

2. 可靠树干中心和半径（激光点最小个数），文本文件，文件名前缀“ReliableStemCenterByPtsNum_”。格式同上。
3. 可靠树干中心和半径（激光点最小个数和三维直线拟合），文本文件，文件名前缀“ReliableStemCenterByPtsNum&FitLine_”。格式同上。
4. 模拟树干的三维直线参数，文本文件，文件名前缀“FittedStemLine_”。每一行存储由空格符（空格或者制表符）分隔的八个数，依次记录代表一棵树干的直线的方向向量（m、n、p，分别表示方向在 x、y、z 上的分量），直线上一点（x0、y0、z0），树木编号，直线拟合使用的中心个数。

格式示例：

```
-0.534342 0.225095 9.74032 -21.7062 1.94663 2.54463 208 12
-0.0203583 -0.037052 0.993142 -20.5136 11.5978 0.09419 224 6
0.0120045 0.0347607 1.08977 -20.1675 5.9183 1.90278 240 30
.....
```

参数说明

1. Cell size used in Hough transform: 使用该栅格尺寸栅格化点云在 x-o-y 平面上的投影，该栅格图像用于哈夫变换。
2. Slice thickness along Z axis: 每个高度上的切片的厚度。
3. Maximum possible radius: 半径的取值范围，单位是像元个数，请使用最大可能的半径取值除以栅格尺寸确定该值。
4. Minimum number of points to give a circle: 输入检测到的有效激光点（落在感

兴趣区 ROI 中的点) 少于该数值, 不足以得到中心和半径的可靠估计, 点云没有构成圆或者圆弧。

5. Coefficient of standard deviation of radius to define ROI: 确定ROI的系数 $Coef_{\sigma}$, 详见算法说明 (从一个树干点云切片中估计树干中心和半径)。
6. Minimum change of center position and radius: 当中心位置和半径的估计值的变化小于该阈值时, 停止迭代算法。
7. Maximum count of iterations: 迭代的最大次数。
8. The position of the scanner (X, Y, Z): 扫描仪在点云本地坐标系下的坐标位置。
9. Angle resolution of theta (rotate about Z axis): 扫描的角度分辨率, 绕 z 轴的旋转角度。
10. Angle resolution of phi (rotate about X axis): 扫描的角度分辨率, 绕 x 轴的旋转角度。
11. Extinction coefficient to calculate minimum number of stem points in a slice: 衰减系数, 用于计算该树干切片中的应该含有的激光点的最小个数。
12. Maximum distance between stem centers to fitted line: 利用树干中心拟合三维直线时, 距离直线大于该数值的中心将被去除。
13. Minimum number of points to give a line of stem: 利用树干中心拟合三维直线所需要的最小中心个数。

算法说明

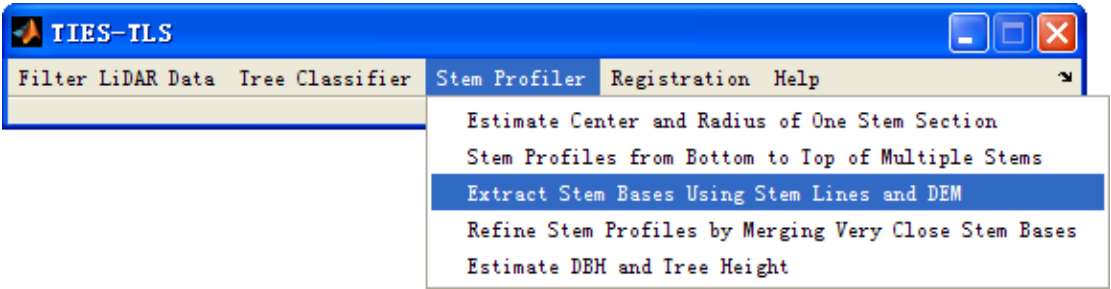
1. 哈夫变换和圆拟合相结合的迭代算法, 根据某个高度的树干切片估计中心和半径, 详见算法说明 (从一个树干点云切片中估计树干中心和半径)。同时利用扫描分辨率、衰减系数和扫描仪位置计算该树干切片中的应该含有的激光点的最小个数, 小于该最小个数的点云切片将被去除, 留下的是可靠的树干中心 (文件名前缀 “ReliableStemCenterByPtsNum_”)。

2. 三维直线拟合算法。利用给定的三维空间中的点, 采用最小二乘法拟合直线, 计算每个点与拟合直线间的距离, 去除距离大于给定阈值的中心, 然后利用剩下的点重新拟合直线, 重复该过程, 直到所有中心到直线的距离都在阈值以内, 或者剩下的中心数量小于可以拟合直线的最小个数时, 停止

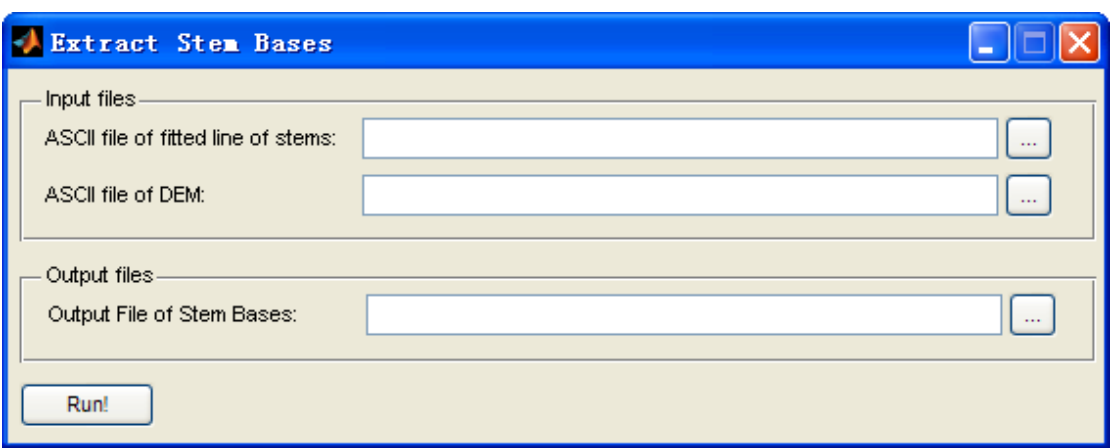
迭代，说明这些中心点没有组成有效的三维直线。剩下的中心就是进一步挑选出来的可靠中心点（文件名前缀“ReliableStemCenterByPtsNum&FitLine_”）。

8.3 提取树干地表位置

菜单选择



界面窗口



文件格式

输入文件:

1. 模拟树干的三维直线参数，文本文件，文件名前缀“FittedStemLine_”。每一行存储由空格符（空格或者制表符）分隔的八个数值，依次记录代表一棵树干的直线的方向向量（m、n、p，分别表示方向在 x、y、z 上的分量），直线上一点（x0、y0、z0），树木编号，直线拟合使用的中心个数。

格式示例：

-0.534342 0.225095 9.74032 -21.7062 1.94663 2.54463 208 12

-0.0203583 -0.037052 0.993142 -20.5136 11.5978 0.09419 224 6
0.0120045 0.0347607 1.08977 -20.1675 5.9183 1.90278 240 30
.....

2. DEM 数据，文本文件，地表位置信息。

格式说明：

NCOLS xxx

NROWS xxx

XLLCENTER xxx | XLLCORNER xxx

YLLCENTER xxx | YLLCORNER xxx

CELLSIZE xxx

NODATA_VALUE xxx

row 1

row 2

.

.

.

row n

格式示例：

NCOLS 480

NROWS 450

XLLCORNER 378922

YLLCORNER 4072345

CELLSIZE 30

NODATA_VALUE -32768

43 2 45 7 3 56 2 5 23 65 34 6 32 54 57 34 2 2 54 6

35 45 65 34 2 6 78 4 2 6 89 3 2 7 45 23 5 8 4 1 62

输出文件：

1. 树干的地表位置，文本文件，文件名前缀 “StemBases_”。

第一行是字段名 “x, y, z, TreeNO”;

第二行及其之后每一行存储由逗号分隔的四个数值，记录地面点的 x 、 y 、 z 坐标值，以及树木编号。

格式示例：

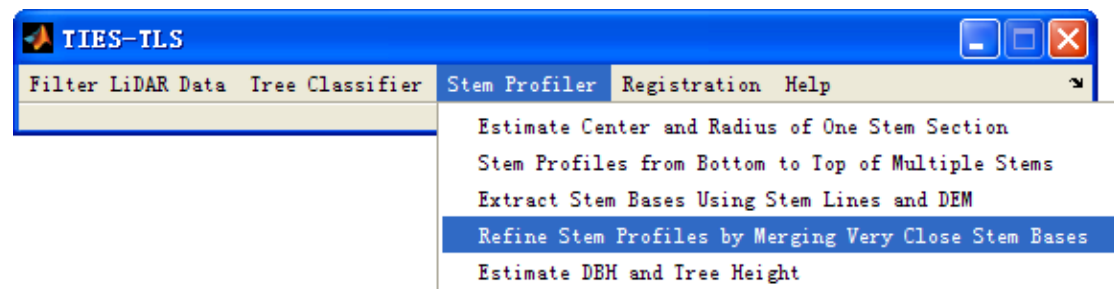
```
x, y, z, TreeNO
-21.525,1.87029,-0.758789,208
-20.4708,11.6757,-1.99434,224
-20.1999,5.82462,-1.03421,240
.....
```

算法说明

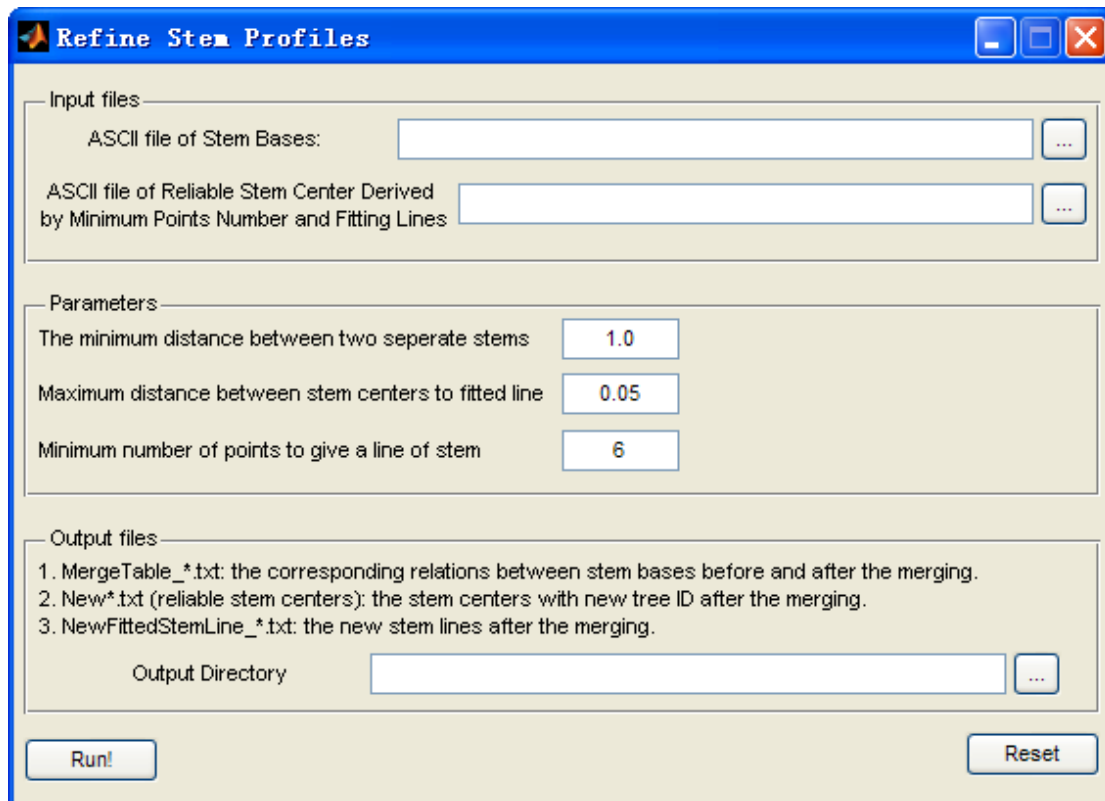
求解代表树干的直线和 DEM 的交点，即为树干的地表位置。

8.4 改进树干地表位置

菜单选择



界面窗口



文件格式

输入文件:

1. 树干的地表位置，文本文件，文件名前缀“StemBases_”。

第一行是字段名“x, y, z, TreeNO”;

第二行及其之后每一行存储由逗号分隔的四个数值，记录地面点的 x、y、z 坐标值，以及树木编号。

格式示例:

x, y, z, TreeNO

-21.525,1.87029,-0.758789,208

-20.4708,11.6757,-1.99434,224

-20.1999,5.82462,-1.03421,240

.....

2. 可靠树干中心和半径（激光点最小个数和三维直线拟合），文本文件，文件名前缀“ReliableStemCenterByPtsNum&FitLine_”。每一行存储由空格符（空格或者制表符）分隔的七个数值，依次记录各个树干各个高度上的：树干中

心和半径 (y、x、半径、z)，树木编号，最终圆拟合使用的激光点个数，树干切片构成的圆弧大小（单位：弧度）。

格式示例：

```
10.065695  -38.322525  0.419517   3.181771   75  9   1.651437
-5.969211  -35.383311  0.210647   1.645929   97  9   1.486139
-8.125523  -25.828318  0.144873   -0.119948  175 12  2.394584
.....
```

输出文件：

1. 改进前后树木编号对应列表，文本文件，文件名前缀 “MergeTable_”。

第一行是字段名 “old_TreeNO new_TreeNO”；

第二行及其之后每一行存储由空格符（空格或制表符）分隔的两个数值，记录改进前的树木编号，改进后的树木编号（通常，改进之后的树木编号为负值）。

格式示例：

```
old_TreeNO new_TreeNO
208 -1
224 -2
240 -3
.....
```

2. 改进之后，标记了新的树木编号的树干中心和半径，文本文件，文件名前缀 “New”。每一行存储由空格符（空格或者制表符）分隔的七个数值，依次记录各个树干各个高度上的：树干中心和半径 (y、x、半径、z)，树木编号，最终圆拟合使用的激光点个数，树干切片构成的圆弧大小（单位：弧度）。

格式示例：

```
10.065695  -38.322525  0.419517   3.181771   75  9   1.651437
-5.969211  -35.383311  0.210647   1.645929   97  9   1.486139
-8.125523  -25.828318  0.144873   -0.119948  175 12  2.394584
.....
```

3. 改进之后，标记了新的树木编号的树干直线参数，，文本文件，文件名前缀

“NewFittedStemLine_”。每一行存储由空格符（空格或者制表符）分隔的八个数值，依次记录代表一棵树干的直线的方向向量（ m 、 n 、 p ，分别表示方向在 x 、 y 、 z 上的分量），直线上一点（ x_0 、 y_0 、 z_0 ），树木编号，直线拟合使用的中心个数。

格式示例：

```
-0.534342  0.225095   9.74032 -21.7062    1.94663 2.54463 208 12
-0.0203583 -0.037052   0.993142   -20.5136    11.5978 0.09419 224 6
0.0120045  0.0347607   1.08977 -20.1675    5.9183  1.90278 240 30
.....
```

参数说明

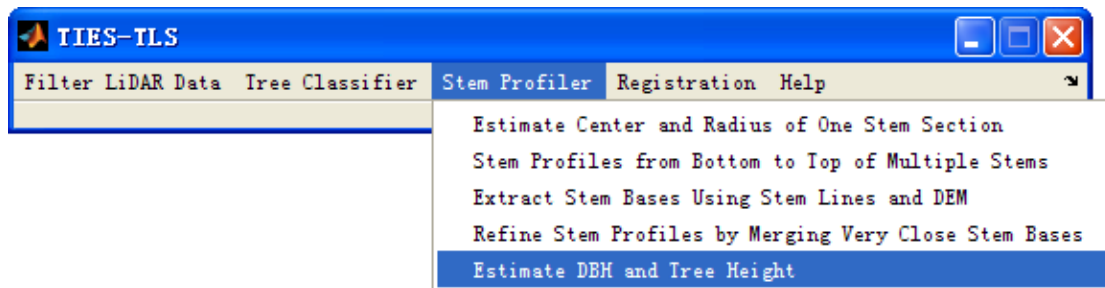
1. The minimum distance between two separate stems: 树干地表位置之间的最小距离，如果两个地表位置间的距离小于该数值，改进算法将这两棵树所对应的树干中心标记为同一个树木编号，并且重新拟合直线。
2. Maximum distance between stem centers to fitted line: 利用树干中心拟合三维直线时，距离直线大于该数值的中心将被去除。
3. Minimum number of points to give a line of stem: 利用树干中心拟合三维直线所需要的最小中心个数。

算法说明

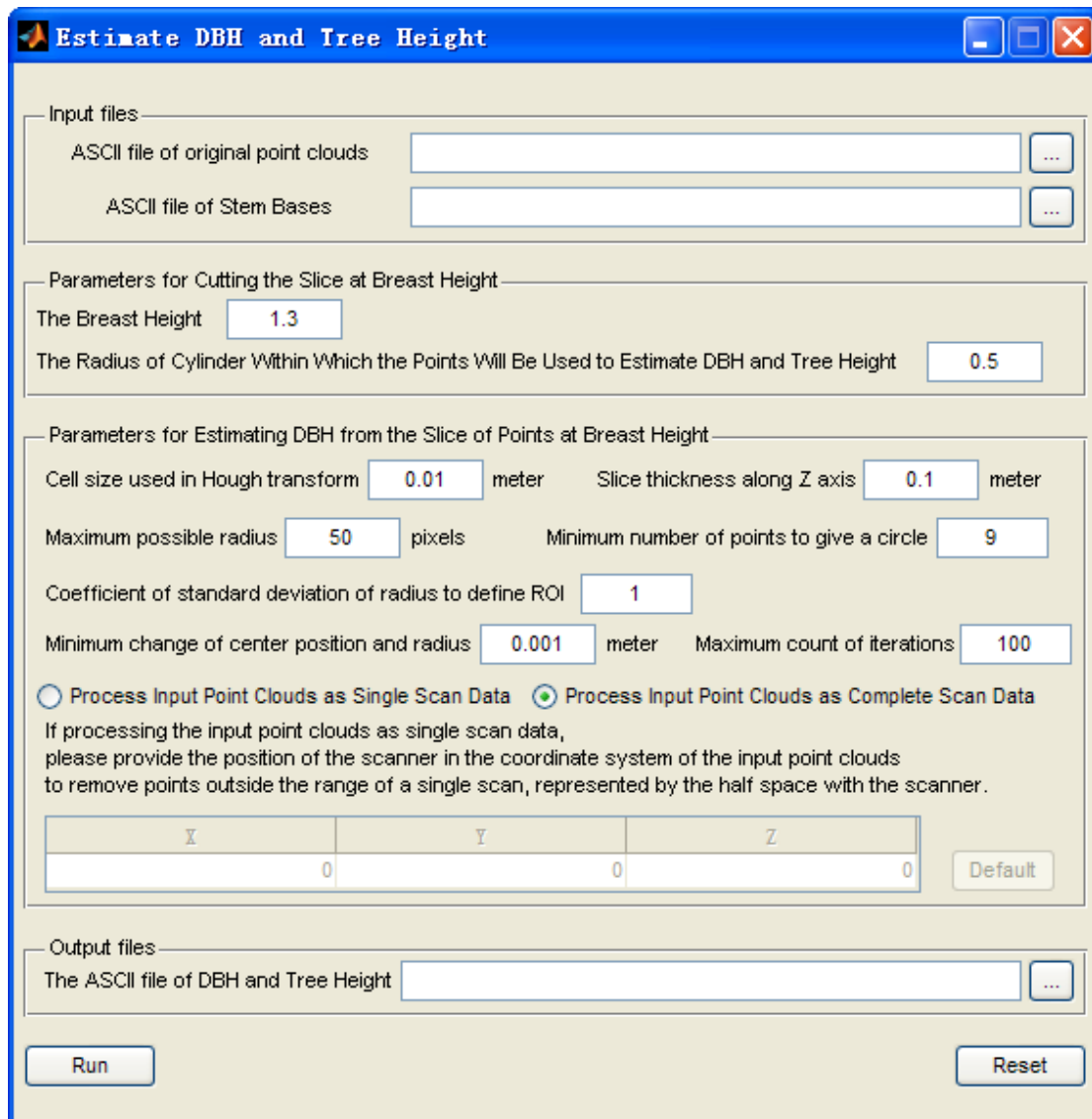
该模块通过合并距离较近的树干表面位置，并将其相关的树干中心重新标记为同一棵树的编号，并重新拟合直线。但是没有给出改进后的树干地表位置，重新调用模块“提取树干地表位置”，利用新的树干直线即可得到改进后的树干地表位置。

8.5 估计胸径和树高

菜单选择



界面窗口



文件格式

输入文件:

1. 激光雷达扫描的点云数据，文本文件，每一行存储由空格符（空格或者制表

符) 分隔的三个数值, 代表一个点的 x、y、z 坐标值。

格式示例:

```
13.178963  0.022621  9.144001
13.186680  0.022706  9.115335
13.180342  0.022757  9.077018
.....
```

2. 树干的地表位置, 文本文件。

第一行是字段名 “x, y, z, TreeNO”;

第二行及其之后每一行存储由逗号分隔的四个数值, 记录地面点的 x、y、z 坐标值, 以及树木编号。

格式示例:

```
x, y, z, TreeNO
-21.525,1.87029,-0.758789,208
-20.4708,11.6757,-1.99434,224
-20.1999,5.82462,-1.03421,240
.....
```

输出文件:

1. 胸径和树高, 文本文件。

第一行是字段名 “TreeNO, DBH, TreeHeight, CenterX_BH, CenterY_BH, StemBaseX, StemBaseY, StemBaseZ”

第二行及其之后每一行存储有逗号分隔的八个数值, 依次记录树木编号, 胸径, 树高, 胸高处树干中心 (x、y), 树干地表位置 (x、y、z)。

格式示例:

```
TreeNO, DBH, TreeHeight, CenterX_BH, CenterY_BH, StemBaseX, StemBaseY, StemBaseZ
-50, 0.322587, 15.833931, 17.899813, -4.291951, 17.519100, -4.207300, -0.287352
-49, NaN, NaN, NaN, NaN, 18.060500, 10.168200, -1.568740
-48, 0.217097, 13.311381, 16.459278, 6.113325, 16.497200, 6.105290, -1.435610
.....
```

参数说明

1. The Breast Height: 胸高，一般都取 1.3 米。
2. The Radius of Cylinder Within Which the Points Will Be Used to Estimate DBH and Tree Height: 根据该半径和胸高处树干中心确定一个圆柱，利用该圆柱内的最高点来估计树高。详见算法说明。
3. Cell size used in Hough transform: 使用该栅格尺寸栅格化点云在 x-o-y 平面上的投影，该栅格图像用于哈夫变换。
4. Slice thickness along Z axis: 每个高度上的切片的厚度。
5. Maximum possible radius: 半径的取值范围，单位是像元个数，请使用最大可能的半径取值除以栅格尺寸确定该值。
6. Minimum number of points to give a circle: 输入检测到的有效激光点（落在感兴趣区 ROI 中的点）少于该数值，不足以得到中心和半径的可靠估计，点云没有构成圆或者圆弧。
7. Coefficient of standard deviation of radius to define ROI: 确定ROI的系数 $Coef_{\sigma}$ ，详见算法说明（从一个树干点云切片中估计树干中心和半径）。
8. Minimum change of center position and radius: 当中心位置和半径的估计值的变化小于该阈值时，停止迭代算法。
9. Maximum count of iterations: 迭代的最大次数。

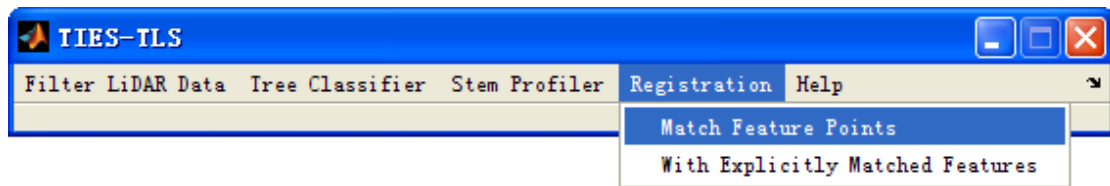
算法说明

从地表位置以上 1.3m 处，截取一定厚度的树干点云，采用哈夫变换和圆拟合相结合的迭代算法估计胸高处的中心和半径，即得到树木的胸径。最后从胸高处中心周围一定范围内的点云中找出最高点（z 值最大的点），树高即为最高点和地表位置之间的高差。

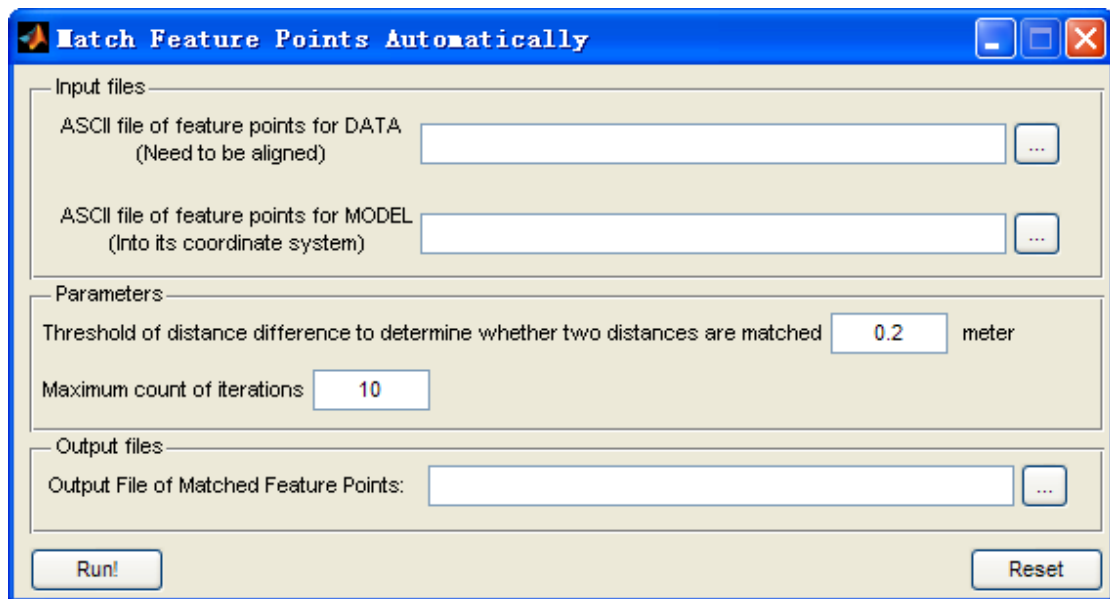
9 自动配准模块

9.1 匹配特征点

菜单选择



界面窗口



文件格式

输入文件:

1. 待配准的特征点，文本文件。

第一行是字段名 “x, y, z, FeaturePointNO”，其他名字亦可，第一行将被跳过不予读取。

第二行及其之后每一行存储由逗号分隔的四个数值，记录地面点的 x、y、z 坐标值，以及特征点编号。

格式示例：

x, y, z, TreeNO

-21.525,1.87029,-0.758789,208

-20.4708,11.6757,-1.99434,224

-20.1999,5.82462,-1.03421,240

.....

2. 坐标系作为基准的特征点，文本文件。

输出文件:

1. 匹配的特征点点对，文本文件。每一行存储由逗号分隔的十个数值，依次是待配准特征点的坐标 (x、y、z) 及其编号，与之匹配的基准特征点的坐标 (x、y、z) 及其编号，匹配距离个数和匹配误差均值（详见算法说明）。如果匹配距离个数或匹配误差均值是 NaN，则该点对无效。

格式示例：

```
13.0195 -11.4951    -2.38152   -49 6.566   5.920   -1.703  -38 3   0.00969959
7.94843 0.513456    -1.8726  -39 -6.214   3.328   -1.597  -17 3   0.0085985
5.99162 -13.8951    -2.01246   -29 7.508   -1.419   -1.086  -40 2   0.010283
7.63303 14.7127 -1.02497    -36 -20.1999   5.8246  -1.034  -3  2   0.0128349
.....
```

参数说明

算法说明

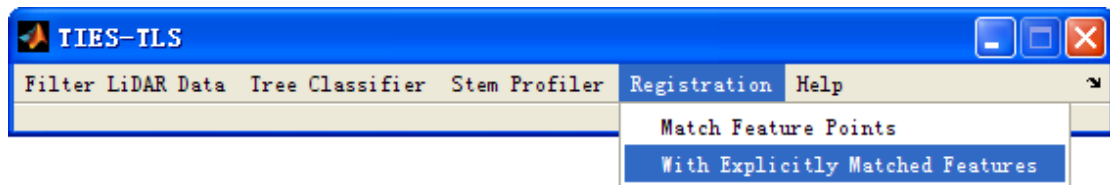
基于距离不变的迭代配准算法。该算法利用两点之间的距离在旋转、平移后保持不变的特性，采用变化的距离阈值和迭代过程，寻找两个点集之间的对应点，并利用对应点计算配准参数。步骤如下：

- (a) 给定距离阈值 Δ_{dis} ，如果两个距离之差小于 Δ_{dis} ，则将它们记作匹配距离，距离之差记作匹配误差。对两个点集 S 和 D ，将 S 中某个点 $p_{S,i}$ 到 S 中其他点的距离，和 D 中某个点 $p_{D,j}$ 到 D 中其他点的距离进行比较，得到 $p_{S,i}$ 和 $p_{D,j}$ 之间的匹配距离个数 $C_{i,j}$ 和匹配误差均值 $MDD_{i,j}$ 。计算 S 中的点 $p_{S,i}$ 和 D 中每个点之间的匹配距离个数和匹配误差均值，将匹配距离个数最大、匹配误差均值尽量小的点作为 $p_{S,i}$ 在 D 中的匹配点。对 S 中的每个点，找出 D 中的对应点，构成对应点对集合。
- (b) 去除对应点对集合中的重复点及其对应点，然后将对应点对按照匹配距离个数降序（主要顺序）、匹配误差均值升序（次要顺序）排列，去除后面 50% 的点对及其相应的点，并更新点集 S 和 D 。

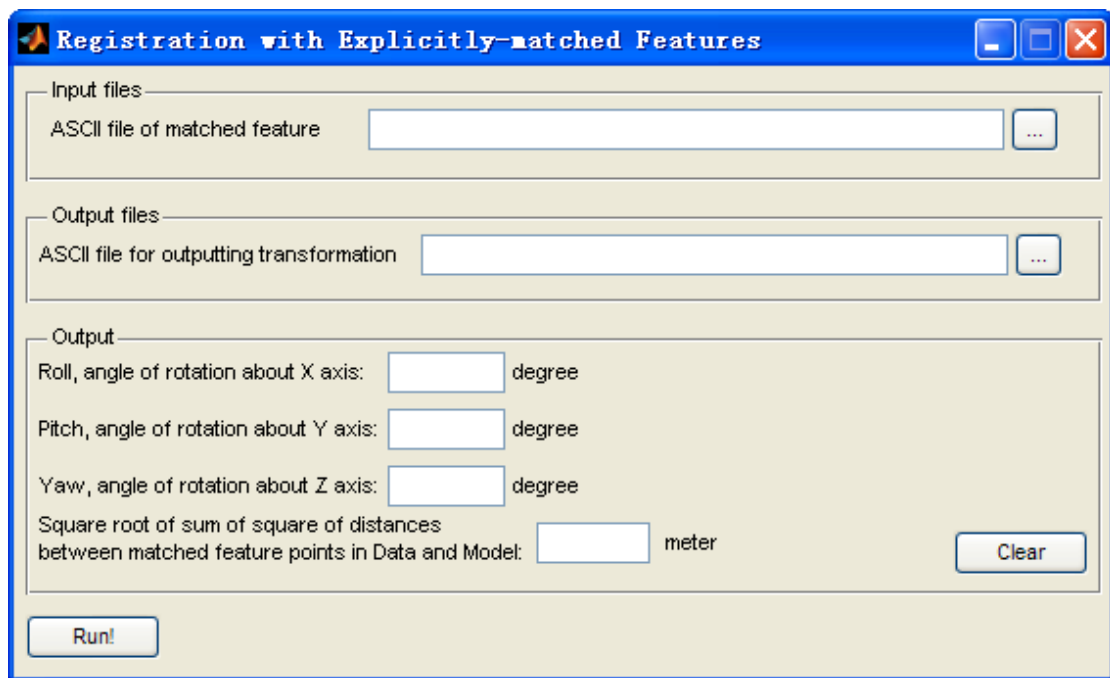
- (c) 缩小距离阈值 Δ_{dis} ，利用更新后的点集 S 和 D ，重复步骤 (a) 和 (b)，直到对应点对个数小于给定的最大阈值 M 为止。如果最终的对应点对个数大于给定的最小阈值 N ，则使用这些对应点对计算配准参数；如果小于 N ，则表明两个点集没有足够多的可靠点对，无法计算配准参数。

9.2 利用匹配的对特征点计算配准参数

菜单选择



界面窗口



文件格式

输入文件:

1. 匹配的特征点点对，文本文件。每一行存储由逗号分隔的十个数值，依次是待配准特征点的坐标 (x、y、z) 及其编号，与之匹配的基准特征点的坐标

(x、y、z) 及其编号，匹配距离个数和匹配误差均值（详见算法说明）。如果匹配距离个数或匹配误差均值是 NaN，则该点对无效。

格式示例：

```
13.0195 -11.4951    -2.38152    -49 6.566    5.920    -1.703    -38 3    0.00969959
7.94843 0.513456    -1.8726 -39 -6.214    3.328    -1.597    -17 3    0.0085985
5.99162 -13.8951    -2.01246    -29 7.508    -1.419    -1.086    -40 2    0.010283
7.63303 14.7127 -1.02497    -36 -20.1999    5.8246    -1.034    -3  2    0.0128349
.....
```

输出文件：

1. 配准矩阵，文本文件。存储一个 4*4 的矩阵，由旋转矩阵 R (3*3) 和平移向量 T (3*1) 构成。

$$\begin{bmatrix} R & T \\ \bar{0} & 1 \end{bmatrix}$$

算法说明

匹配特征点对两点之间的距离平方的均值为误差函数，最小化该函数求解。