电 子 科 技 大 学 实 验 报 告

课程名称： 数学实验

实验地点：

指导教师：

评 分：

完成实验学生信息：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **姓名** | **学号** | **选课**  **序号** | **贡献百分比**  **(%)** | **备注**  **（主要工作**） |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

1. 学生人数按照任课教师要求限定；
2. 对于“评价、改进、总结和体会”都要认真填写，和其他内容是评价实验成绩的重要参考。

图片拼接的变换矩阵

摘要

在生活中常常会遇到需要将数张有重叠部分的图像拼成一幅无缝高分辨率图像的问题，而解决这个问题的其中一种较为良好的方式是采用图像拼接技术。图像拼接技术主要分为两个部分，其一是图像配准，其二是图像融合。本文主要针对后一项问题进行了分析和解答。

关键词：回归分析、噪声消除，图像变换、图像拼接

目 录

[摘要 2](#_Toc105753166)

[**1.** **问题重述** 4](#_Toc105753167)

[**2.** **问题分析** 4](#_Toc105753168)

[2.1. 图像拼接的分析 4](#_Toc105753169)

[2.2. 求解变换矩阵的分析 4](#_Toc105753170)

[2.3. 变换矩阵的计算效果的分析 4](#_Toc105753171)

[**3.** **模型假设** 5](#_Toc105753172)

[**4.** **定义与符号说明** 5](#_Toc105753173)

[4.1. 名词解释 5](#_Toc105753174)

[4.2. 符号说明 5](#_Toc105753175)

[**5.** **模型的建立与求解** 6](#_Toc105753176)

[5.1. 参考回归平面的建模与噪声数据的去除 6](#_Toc105753177)

[5.1.1. 建模前准备 6](#_Toc105753178)

[5.1.2. 模型建立 6](#_Toc105753179)

[5.1.3. 模型求解及检验 7](#_Toc105753180)

[5.2. 图像拼接的建模与求解 9](#_Toc105753181)

[5.2.1. 建模前准备 10](#_Toc105753182)

[5.2.2. 模型建立 10](#_Toc105753183)

[5.2.3. 模型求解及检验 10](#_Toc105753184)

[**6.** **模型的评价与推广** 10](#_Toc105753185)

[6.1. 模型优点 10](#_Toc105753186)

[6.2. 模型缺点 11](#_Toc105753187)

[6.3. 模型改进方向 11](#_Toc105753188)

[6.4. 模型推广 11](#_Toc105753189)

[**7.** **心得体会与总结** 11](#_Toc105753190)

[**8.** **对本实验问题的设计提出改进意见** 12](#_Toc105753191)

[**参考文献** 12](#_Toc105753192)

[附件 12](#_Toc105753193)

[附件 13](#_Toc105753194)

[附件一：排序后的相对噪声值 13](#_Toc105753195)

[附件二：实验代码 17](#_Toc105753196)

[附件三：最终拼接后的图片 19](#_Toc105753197)

# **问题重述**

1. 已知两张图片以及配对的SURF特征点，需要设计方法计算出第2张图片变换到第1张图片上的“变换矩阵”。
2. 通过图像的拼接来观察变换矩阵的计算效果

# **问题分析**

每一个特征点通过相对应的横坐标和纵坐标描述，

## 图像拼接的分析

要实现图像的拼接，关键在于将一张图片按照一定的旋转，平移，伸缩等操作变换后与另一种图片拼接起来。本题背景已给出若干对SURF特征点，通过对局部若干点的变换从而推出全局的变换矩阵。考虑一阶线性变换，我们可以利用矩阵来描述转换，问题转化为求解变换矩阵的系数。

## 求解变换矩阵的分析

由于特征点提取与匹配算法的限制，部分特征点对可能不符合要求，需要去除这些噪声。剩余的多组特征点对作为自变量需要考虑其对于因变量变换矩阵系数的影响，一种统计方法是使用回归分析。每一对特征点作为一个数据用三维坐标描述，通过matlab的多元线性拟合函数regress拟合出相应的平面。平面的参数即为我们所求的变换矩阵参数。

## 变换矩阵的计算效果的分析

得到了变换矩阵后，可以将第2张图片上的点变换到第1张图片上，再利用matlab中的绘图函数将图片显示出来，观察图片的拼接效果，即可评估变换矩阵的计算效果。

# **模型假设**

1. 特征点对中噪声数据占少数。
2. 将srow-col-2.jpg当作待拼接图片。
3. 图片变换是一阶线性操作。

# **定义与符号说明**

## 名词解释

噪声数据**：**不符合匹配要求的SURF特征点。

参考回归平面：选定3个非噪声数据构造的线性回归平面。

目标回归平面：对非噪声数据构造的线性回归平面。

相对噪声值：消除噪声数据过程中，每对数据相对于参考回归平面的残差。

噪声阈值：对于相对噪声值超过噪声阈值的点可以视作噪声数据。

## 符号说明

|  |  |  |
| --- | --- | --- |
| 符号 | 说明 | 单位 |
| x1，y1 | 第1张图片变换前的横，纵坐标 | 像素 |
| x2，y2 | 第2张图片变换前的横，纵坐标 | 像素 |
| x’ | 第2张图片变换后的横坐标 | 像素 |
| y’ | 第2张图片变换后的纵坐标 | 像素 |
| t11 t12 | 横坐标变换系数 | / |
| t21 t22 | 纵坐标变换系数 | / |

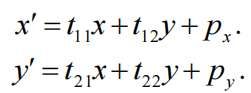
# **模型的建立与求解**

## 参考回归平面的建模与噪声数据的去除

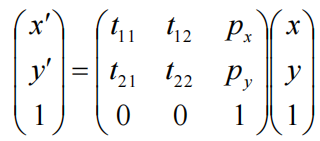
### 建模前准备

假设第2张图片变换前的坐标为（x,y），变换后的坐标为（x’,y’）

而变换前后的坐标满足下列方程组：

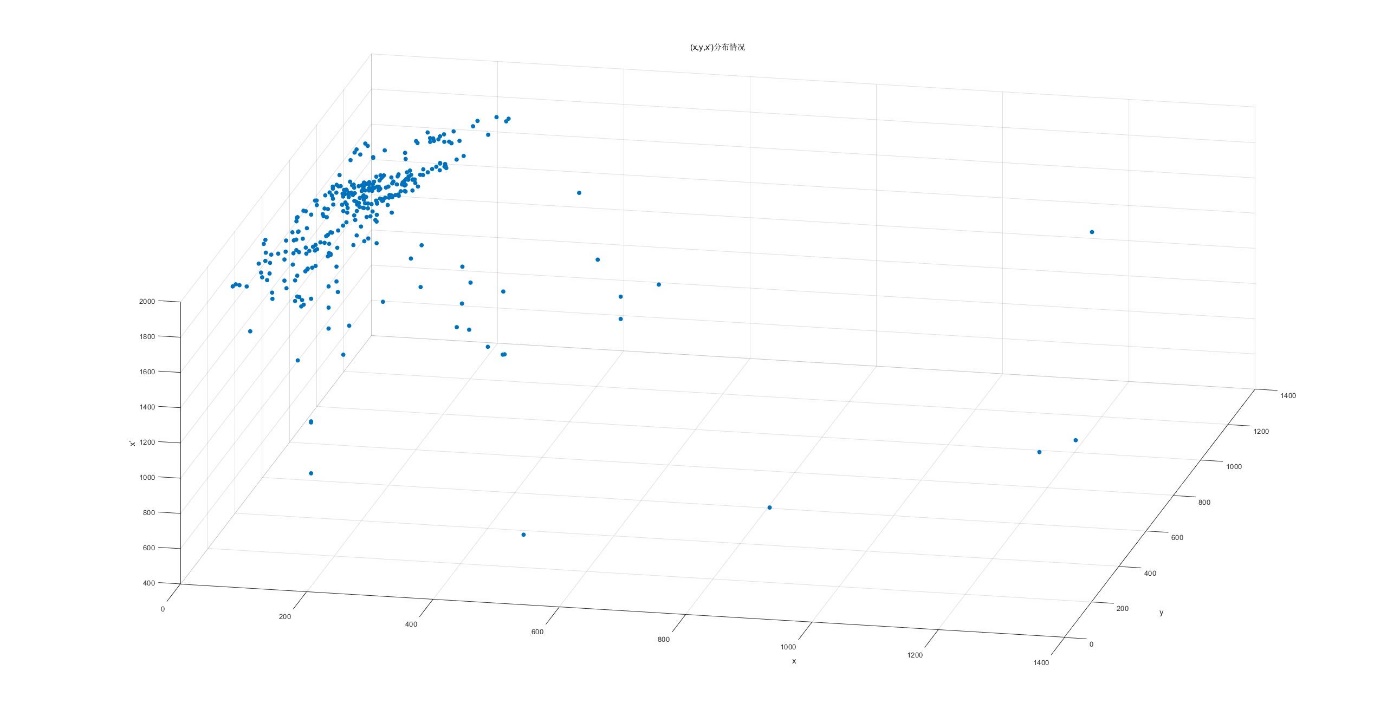


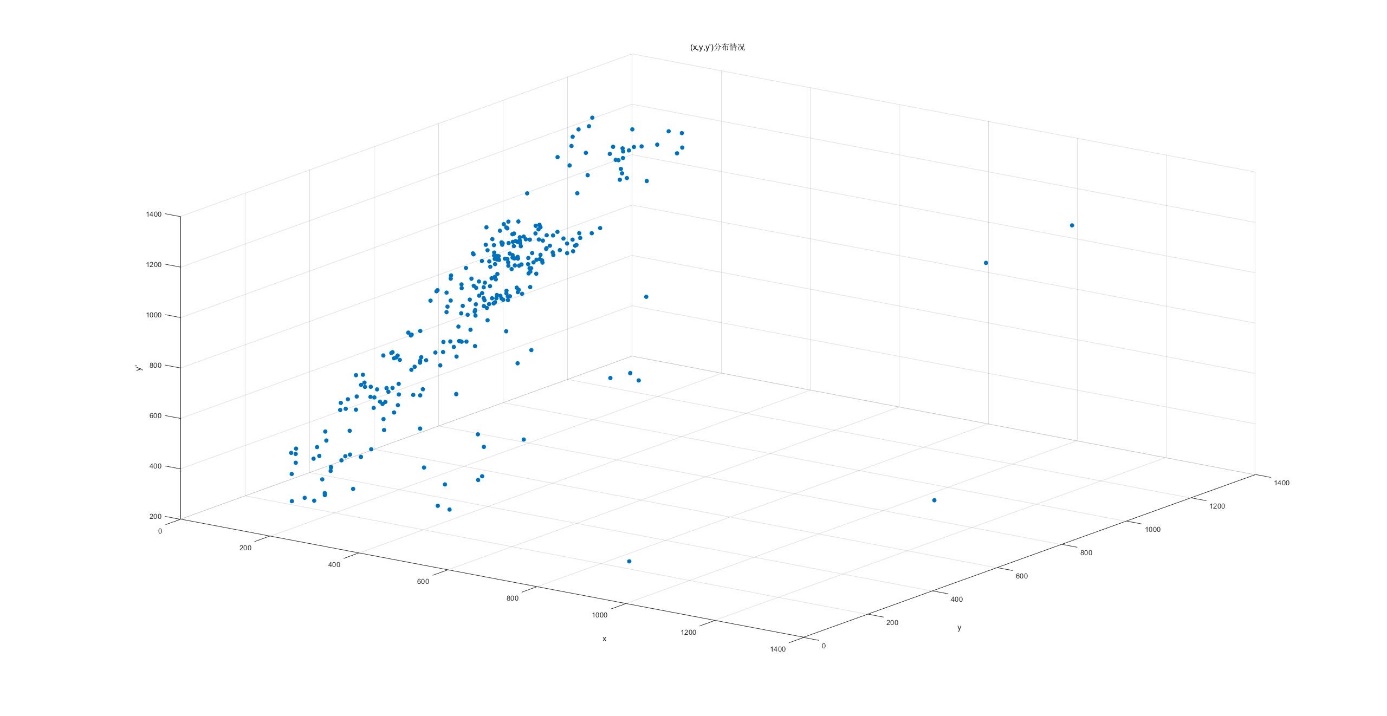
可以将上述方程组转化为矩阵乘法的形式



### 模型建立

首先对待处理数据进行分析：

横轴为x2,纵轴为y2，竖轴为x1



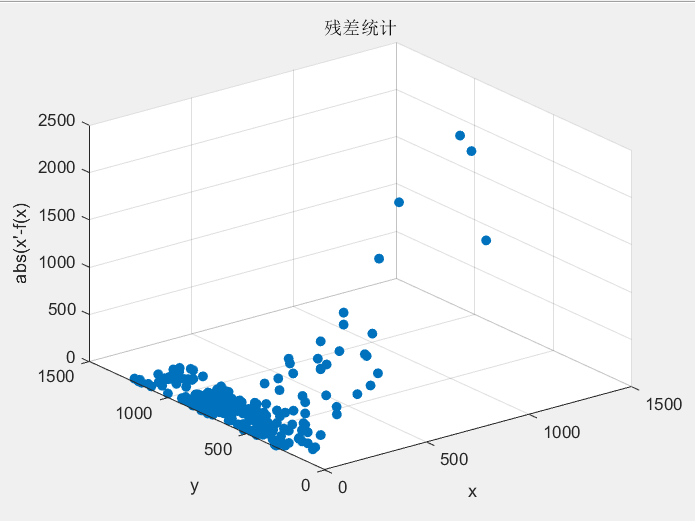
横轴为x2,纵轴为y2，竖轴为y1

将每组数据用三维坐标（x2,y2,x1）（x2,y2,y1）表示出来,我们发现，绝大部分坐标都集中于一个特定的平面，我们可以将这个平面视作我们需要求的回归平面，而将其他点是为噪声数据。

我们可以先初步预估一个参考回归平面，将离参考回归平面较远的点去除。

### 模型求解及检验

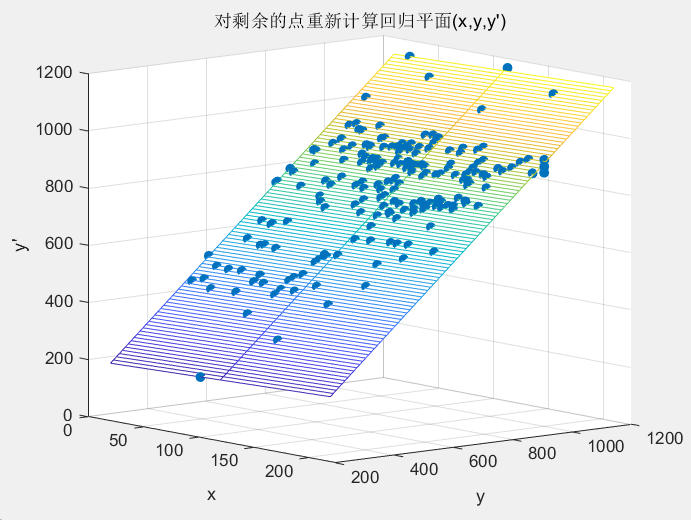
首先由上述分析以及（x2,y2,x1）（x2,y2,y1）坐标图，我们可以得到3组非噪声数据点，分别是原始数据的第154，215，237组。调用线性回归函数得到参考回归平面，设相对噪声值为数据点与回归平面的残差，求解出残差分布的图像如下。



残差分布图（以（x2,y2,x1）为例）

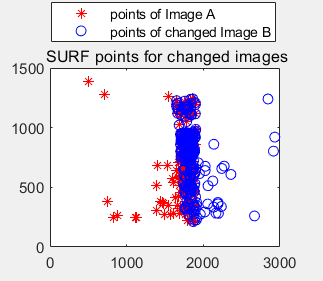
将相对噪声值排序，得到以下数据（见附件一）：

实验中发现将噪声阈值设为2可以实现较好的图像拼接效果，去掉相对噪声值大于噪声阈值的点，再对剩余的点进行一次回归平面的计算，计算后得出的回归平面如下图所示。



目标回归平面与剩余点的拟合程度（以（x2,y2,y1）为例）

由图可见，非噪声点基本分布在回归平面上，也就意味着剩余的点基本上是两幅图像可以一一配对的点，因此完成了对变换矩阵的求解。



将特征点按照变换矩阵变换后的分布情况

## 图像拼接的建模与求解

### 建模前准备

有了计算效果较好得变换矩阵后，还需进一步观察图像的拼接效果，以此来最终评价变换矩阵的计算效果。首先创建一张1050\*3000的uint8数组储存新图片的像素。

### 模型建立

参照图片srow-col-1中的像素直接写入新图片，待拼接srow-col-2的像素经过变换矩阵变换后写入新图片。利用matlab中的imshow函数观察图像的拼接效果。

### 模型求解及检验

使用for循环遍历图片的每一个像素按照上述步骤写入到新图，并观察点分布的位置情况，



图像的拼接效果如图所示，可见，最终图像能基本呈现出无缝拼接效果，也就从侧面体现出变换矩阵的计算效果较好。

# **模型的评价与推广**

## 模型优点

1. 拼接效果良好，符合预期。
2. 去除噪声数据的过程简单高效。
3. 一阶线性变换可以处理大部分图片操作。
4. 求解参考回归平面的过程保留了大部分原始数据，且拟合效果良好。
5. 利用图像直观分析数据特点。

## 模型缺点

1. 选取参照点的过程中依赖人工容易出错。
2. 图片变换过程出现了像素损失。
3. 噪声阈值的选取缺乏理论依据。
4. 待拼接图片变换过程较慢。
5. 一阶线性变换对一些复杂操作不适用，需要更高级算法。

## 模型改进方向

1. 改用随机选取参考点，并判断该组参考点是否可行。
2. 改用更高效的算法转换拼接图片。
3. 设计模型量化拼接效果。

## 模型推广

本模型通过研究特征点从而得出整体图片的变换矩阵，该基于线性回归方法的模型同样适用一类通过研究部分性质从而得出整体性质的问题，如天气，市场分析等问题。

# **心得体会与总结**

通过本次实验，我们学会如何通过已有数据和问题来分析其特征，初次遇到此类问题时对庞大数据的特点一无所知，所幸通过已有的数学知识才得以分析出问题本质，这让我们深深体会到了数理基础的重要性：基础的大学数学甚至中学数学涉及的内容却能在实际生活中有广泛的应用。另外，充分利用matlab强大的功能也是本次实验成功元素之一：通过绘图函数直观展现数据特点、通过回归分析函数快速拟合平面、通过图像处理功能转换拼接图片。最后，本次实验结束也给我们留下更进一步的思考：是否存在更高效快速的方法来处理图片，对于非线性变换的问题又该如何处理。这些都是值得我们下一步研究的问题。

本次实验我们总结了几组方法：

1. 利用数据特点去除噪声点。
2. 利用回归分析处理一类函数多元线性关系问题。
3. 利用像素数组处理新图片。

# **对本实验问题的设计提出改进意见**

本次实验题目深入浅出，涉及知识面广，任务量足。希望在题目中增加SURF特征点相关背景，了解其特点及可能存在的问题，方便后续实验进行。

# **参考文献**

[1]范钦慧，工程力学，北京：清华大学出版社，2005.8，73-74

[2]叶其孝，沈永欢，数学手册，北京：科学出版社，2006.1 516-518

[3]陈杰，MATLAB宝典，北京：电子工业出版社，2010.3，340-370

[4]姜启源，谢金星，叶俊，数学模型，北京：高等教育出版社，2011.1,85-115

[5]隋秀凛，高安邦，实用机床设计手册，北京：机械工业出版社，2010.4，附录

[6]黄雍检，陶冶，钱祖平，最优化方法—MATLAB应用，北京：人民邮电出版社，2010.11

# 附件

附件清单

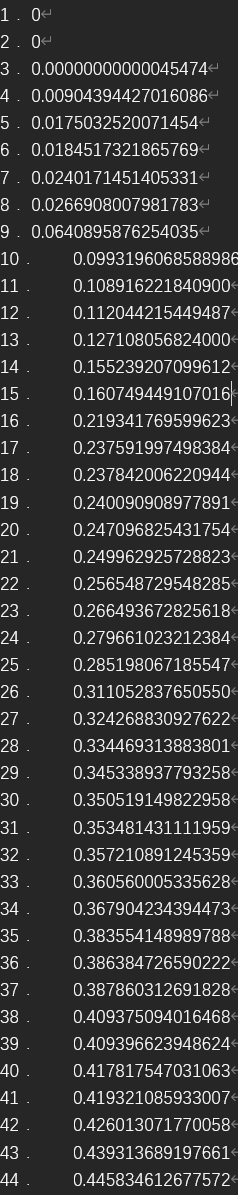
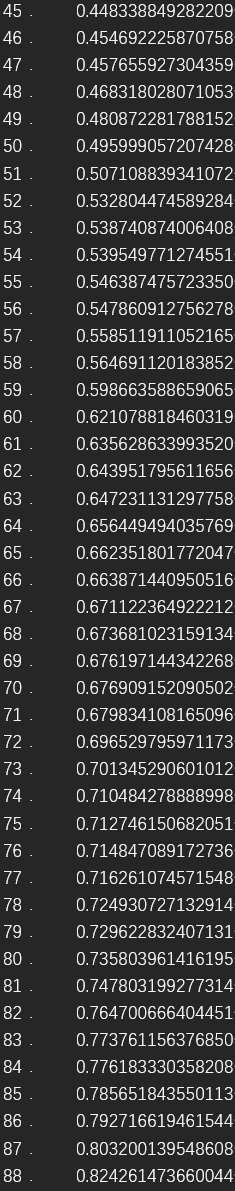
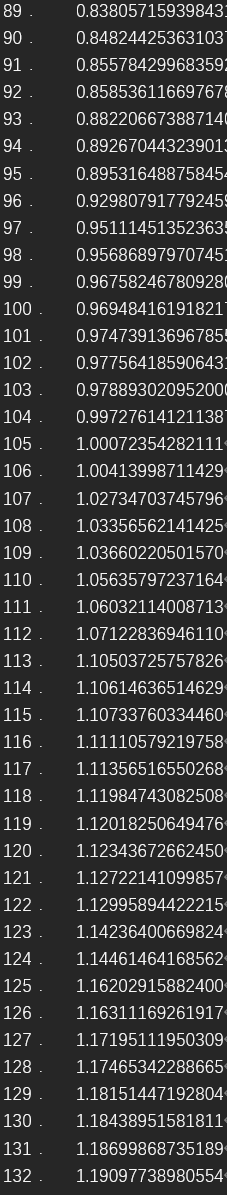
附件一：排序后的相对噪声值

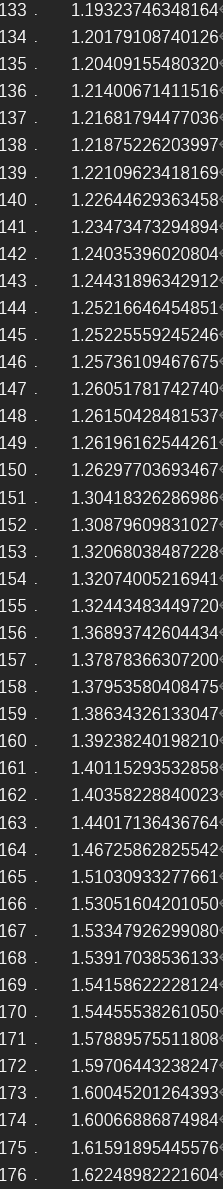
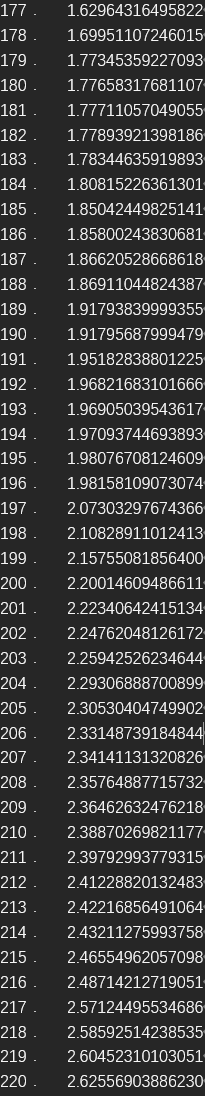
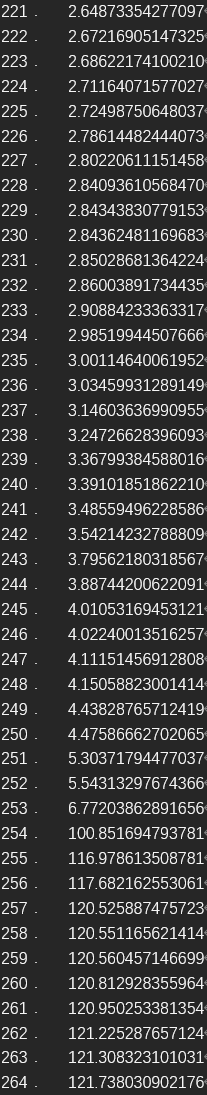
附件二：实验代码

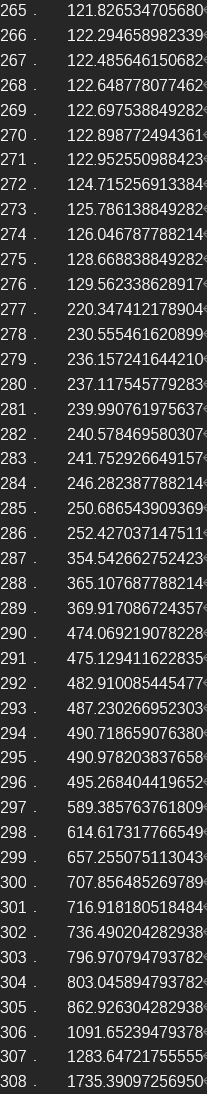
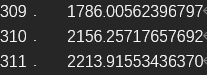
附件三：最终拼接后的图片

# 附件

## 附件一：排序后的相对噪声值

## 附件二：实验代码

function []=splic()

%拼接两张图片

[reg2 reg3] = mregress(); %获取变换矩阵

Fx=@(x,y)reg2(1) + reg2(2)\*x + reg2(3)\*y;

Fy=@(x,y)reg3(1) + reg3(2)\*x + reg3(3)\*y;

filename1 = 'srow-col-1.jpg'; %载入图片

filename2 = 'srow-col-2.jpg';

imgA =imread(filename1);

imgB =imread(filename2);

imgC=uint8(zeros(1050,3080,3)); %创建新图

for i=1:1263

for j=1:1451

xx=round(Fx(j,i));

yy=round(Fy(j,i));

if(xx>0&&yy>0)

imgC(yy,xx,1:3)=imgB((i),(j),1:3); %将'srow-col-2.jpg'中的坐标变换后写入

end

end

end

for i=1:1429

for j=1:1920

imgC(i,j,1:3)=imgA((i),(j),1:3); %写入参照照片'srow-col-1.jpg'

end

end

imshow(imgC)

end

function [reg2 reg3]=mregress()

%对数据求出相应回归平面

load locA.txt % 导入第1个图片文件的SURf特征点

load locB.txt % 导入第2个图片文件的SURf特征点

y=locA([154 215 237],1); %经观察这三个点在所求回归平面上

x1=locB([154 215 237],1);

x2=locB([154 215 237],2);

X=[ones(3,1),x1,x2];

reg1=regress(y,X); %调用线性回归函数初步估计回归平面

F1=@(x,y)reg1(1) + reg1(2)\*x + reg1(3)\*y;

y=locA(:,1);

x1=locB(:,1);

x2=locB(:,2);

ax1=x1;

ax2=x2;

for i=1:311

ay(i)=abs(F1(ax1(i),ax2(i))-y(i)); %计算残差

end

[ay,ind]=sort(ay); %对残差排序

ax1=ax1(ind);

ax2=ax2(ind);

y=y(ind);

%ans=[ax1,ax2,ay']

theta=2;

m=max(find(ay<theta)); %去除残差大于阈值的点

%m为剩余点个数，共196个

y=y(1:m,1);

ax1=ax1(1:m,1);

ax2=ax2(1:m,1);

X=[ones(m,1),ax1,ax2];

reg2=regress(y,X); %对剩余的点重新计算回归平面（求出x'的变换矩阵系数）

%-------------------------------------

y=locA(ind,2);

y=y(1:m,1);

reg3=regress(y,X); %对剩余的点重新计算回归平面（求出y'的变换矩阵系数）

%-------------------------------

end

## 附件三：最终拼接后的图片

