

项目名称：图像配准

摘要：图像配准在目标检测、模型重建、运动估计、特征匹配，肿瘤检测、病变定位、血管造影、地质勘探、航空侦察等领域都有广泛的应用。每一种配准方法通常都针对某个具体问题而设计的，众多方法中，唯一的共性就是每个配准问题最终都要在变换空间中寻找一种最优的变换，这种变换能够使两幅图像之间在某种意义上达到匹配，但对于不同的应用领域，对图像类型的要求不同，就需要具体问题具体分析。

关键字：图像配准

姓名：金瑞达

班级：自动化 61

学号：2160504013

提交日期：2019.3.3

图像配准题目

一. 题目要求:

要求根据已给的两幅图像，在各幅图像中随机找出 7 个点，计算出两幅图像之间的转换矩阵 H，并且输出转换之后的图像。

注：已给图像分别为 Image A 和 Image B。

二. 图像配准简介

图像配准的方式

图像配准的方式可以概括为相对配准和绝对配准两种：相对配准是指选择多图像中的一张图像作为参考图像，将其它的相关图像与之配准，其坐标系统是任意的。绝对配准是指先定义一个控制网格，所有的图像相对于这个网格来进行配准，也就是分别完成各分量图像的几何校正来实现坐标系的统一。本文主要研究大幅面多图像的相对配准，因此如何确定多图像之间的配准函数映射关系是图像配准的关键。通常通过一个适当的多项式来拟合两图像之间的平移、旋转和仿射变换，由此将图像配准函数映射关系转化为如何确定多项式的系数，最终转化为如何确定配准控制点（RCP）。^[1]

图像配准是数字图像处理的一种重要应用，用于对齐两幅或多幅相同场景的图片。

图像配准需要输入图像与参考图像。输入图像是我们希望变换的图像，参考图像是想要配准输入图像的图像。图像配准的主要方法是使用约束点，输入图像产生输出图形的特定变换通常是不知道，所以需要估计变换函数，而估计变换函数问题是建模问题之一。

基于双线性近似的简单模型：

$$x = c1v + c2w + c3vw + c4;$$

$$y = c5v + c6w + c7vw + c8;$$

(v, w) 和 (x, y) 分别是输入图像和参考图像中约束点的坐标。如果在两幅图像中有四对相应的约束点，则可以解出 8 个未知参数。这样构成简单模型，从而可以把一幅图像的像素变换为另一幅图像的像素位置。

当然可以选择更复杂的模型，如最小均方算法等。

标准基准点对，将其保存至工作空间。

利用 matlab 提供的 cpselect 函数可以交互式地选择基准点。在命令行中按照以下方式调用 cpselect 启动交互工具

```
>> cpselect(I_in(:, :, 1), I_base(:, :, 1));
```

在该 control point select 窗口中，点击 tool 里面的 add point 一项进行手工标注点（交互式标点），然后在 file 菜单里 export points to workspace 即可输出表定点的坐标位置。

注意：调用 cpselect（）函数要将需要配准的图像作为第一参数，将基准图像作为后一个参数。并且 cpselect 只接受灰度图像，如果需要处理 RGB 彩色图像，可以只给 CPselect 函数传递一个图像的层。

根据之前得到的控制点对坐标，利用 cp2tform 函数可以计算变换的参数。将基准点对作为输入传递给 cp2tform，选择一种适当的变换类型，cp2tform 函数就能确定该类型变换所需的参数，实际上相当于一种数据拟合。cp2tform 函数寻找能够拟合控制点对的变换参数，返回一个 TFORM 的结构体几何变换结构，其中包括几何变换的类型和参数^[2]

三. 主要函数简介

1. cpselect 函数：用于在两张图像上进行选点
2. Cp2tform 函数：生成映射变换矩阵
3. Imtransform 函数：根据映射变换矩阵对原图像进行变换

四. 实验结果

MovingPoints 的位置矩阵如图所示

969.204663212435	1741.80569948187
817.992227979275	2318.30310880829
808.541450777202	929.038860103627
2452.97668393782	787.277202072539
2745.95077720207	1760.70725388601
2197.80569948187	2072.58290155440
1942.63471502591	484.852331606218

fixedPoints 的位置如下所示

665.451724137931 1257.50172413793

370.110344827586 1761.74310344828

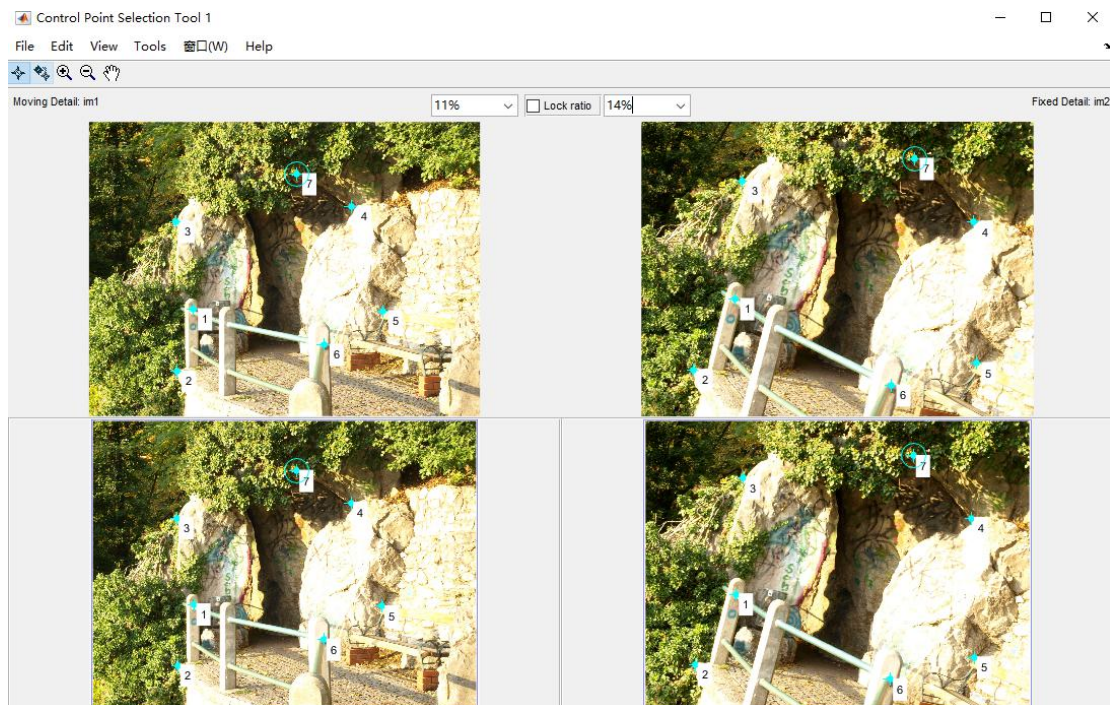
715.875862068966 421.901724137931

2358.26206896552 710.039655172414

2379.87241379310 1711.31896551724

1774.78275862069 1869.79482758621

1940.46206896552 263.425862068965



五. 变换矩阵 T

$$\begin{bmatrix} 0.971371949415010 & 0.250973255568899 & 0 \\ -0.260640058090819 & 0.966457431453963 & 0 \\ 177.491210108821 & -679.376605843981 & 1 \end{bmatrix}$$



六. 代码

```
close all;

im1 = imread('Image A.jpg');
im2 = imread('Image B.jpg');

% cpselect(im1,im2) %用于在两张图片中进行取点进行配准
tform = cp2tform(movingPoints,fixedPoints,'affine');
lout = imtransform(im1,tform);
figure
subplot(1,2,1),imshow(lout);
subplot(1,2,2),imshow(im2);
```

七. 结果讨论心得体会

在选取 7 个变换点时，我的选取是尽可能的覆盖整幅图像，来达到一个比较好的结果，书上的用于变换的方程有 8 个未知数，因此理论上只需要 4 组点，但是这里用了 7 组，可以提高配准的准确度，使结果更好。

有所得的结果可以看出，最终的结果就是将一幅图片进行一个适当的映射，也就是将原图像乘以一个映射矩阵，来使得到的图像与需要配准的图像尽可能的想重叠，以便进行后续的图像处理。

参考文献：

【1】百度百科 图像配准

<https://baike.baidu.com/item/%E5%9B%BE%E5%83%8F%E9%85%8D%E5%87%86>

【2】博客园 数字图像处理--图像配准

<https://www.cnblogs.com/MrZheng9511/p/register.html>