

数字图像处理图像配准作业

赵子瑞

自动化钱61班

2019年3月5日

摘要

本文是数字图像与视频处理的第二次作业，通过理论学习和代码实践，完成了以下任务：

通过手动标点，得到图像中 7 个位置的坐标，进行输出并保存，计算仿射变换矩阵，通过仿射变换矩阵计算得到变换后的图像，输出图像。

本次实验较为简单，建立在第一次实验的基础之上，提高了动手能力和对图像处理的理论理解。

关键词： 图像配准 图像处理 仿射变换 矩阵计算

1 项目任务

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 $ImageA$ 和 $ImageB$ 。

2 手动标点

本次实验中标定的点如图1所示



(a) 原图像标点选择



(b) 变换后图像选择

图 1: 标点选择

3 输出两幅图中对应点的坐标

得到的坐标输出如下。

Image A 的特征点如下。

(977.693542, 1919.11292)
(1286.59680, 1624.91931)
(1374.85486, 1867.62903)
(1749.95166, 1316.01611)
(1786.72583, 1448.40320)
(2139.75806, 1249.82263)
(2581.04834, 2022.08069)

Image B选取的特征点坐标如下。

(639.600830, 1397.73987)
(1015.80042, 1201.21777)
(1032.64514, 1453.88916)
(1537.98792, 1010.31049)
(1537.98792, 1161.91333)
(1931.03223, 1066.45972)
(2150.01416, 1925.54236)

4 计算转换矩阵

转换矩阵的计算可以由公式1得到：

$$H = Q \cdot P^T \cdot (P \cdot P^T)^{-1} \quad (1)$$

，其中， P 为原图像的特征点组成的矩阵， Q 为变换后对应的特征点组成的矩阵， H 为变换矩阵。

经过计算，变换矩阵为：

$$\begin{pmatrix} 0.958704864 & -0.263552481 & 208.122426 \\ 0.267151126 & 0.967908948 & -720.247315 \\ 0 & 0 & 1 \end{pmatrix}$$

5 输出在转换后的图像

变换后的图像如图2所示。



图 2: 变换后的图像

6 代码示例

```
1 import cv2
2 import matplotlib.pyplot as plt
3 import base64
4 import struct
5 import numpy as np
6 from scipy import interpolate
7 from pylab import *
8 from PIL import Image
9
10 class basic_cv_tool:
11
```

```
12     def __init__(self, ImageName):
13         self.ImageName = ImageName
14
15     def ImageRead(self, ImageName):
16         img = cv2.imread(ImageName)
17         return img
18
19     def interest_point_choosing(self, ImageName):
20         img = array(Image.open(ImageName))
21         imshow(img)
22         fea_point = ginput(7)
23         fea_point = np.float32(fea_point)
24         fea_point = np.column_stack((fea_point, \
25         array([1,1,1,1,1,1,1])))
26         return fea_point
27
28     def Getting_H_Matrix(self, img_points_1, img_points_2):
29         H_matrix = ((img_points_2.transpose()) \
30         .dot(img_points_1)).dot(np.linalg.inv( \
31         (img_points_1.transpose()).dot(img_points_1)))
32
33         print(H_matrix)
34         return H_matrix[:2]
35
36 if __name__ == '__main__':
37     image_1_name = '../.. /homework2/Image_A.jpg'
38     image_2_name = '../.. /homework2/Image_B.jpg'
39
40     tool = basic_cv_tool(image_1_name)
41     image = tool.ImageRead(image_1_name)
42     print(image.shape)
43     img1 = tool.interest_point_choosing(image_1_name)
44     print("interest_point_of_image_A is",img1)
45     img2 = tool.interest_point_choosing(image_2_name)
46     print("interest_point_of_image_B is",img2)
47     M = tool.Getting_H_Matrix(img1, img2)
48     img = cv2.warpAffine(image, M, (image.shape[1], image.shape[0]))
49     print(img.shape)
50     cv2.imwrite('test.jpg', img)
```

7 心得体会

本次作业，通过实践，我深入了解了图像配准的相关基础知识，同时也提高了动

手能力，加深了我对仿射变换的理解和掌握程度，也让我拥有了解决实际问题的基本功。同时，代码能力也得到了有效锻炼，我的python能力得到了有效锻炼，同时也学习了opencv的很多基本功能。我自行组织的`cvtoolbox`在我的`github`上维护，欢迎访问 <https://github.com/1989Ryan/Digital-Image-Processing-Project> 进行了解。