

# Medical Image Analysis Assignment 3

## Registration of Medical Images

Jingmo Bai

### Introduction

In this assignment we are going to perform the landmark-based image registration.

For the first task, rigid registration using Euclidean transformations is needed. For the second task, Procrustes alignment using similarity transformations is needed.

The landmarks are generated using SIFT (scale invariant feature transform) features descriptor, then we generate preliminary matching points from the key points. Among those preliminary matches we find real pairs of matching points using RANSAC, to get a optimal set of matches with a maximal number of inliers. Finally, we choose this set of matching points as our landmarks. And we perform the registration alignment with those landmarks.

### Description of theory

#### Transformations

In task 1, the data consists of 29 pairs of microscopy images with different staining methods. So they were acquired with the same magnification, which means they are in the same scale, our assumption is that they may be aligned using a Euclidean transformation, only translation  $t$  and rotation  $R$ . The degree of freedom of Euclidean transformation is 3 in 2D cases.

$$y(x; R, t) = Rx + t$$

In task 2, the data consists of 12 pairs of images with different staining methods but with different magnification, therefore the TRF images show only a detail of the HE images. So, we use a similarity transformation with translation  $t$ , rotation  $T$  and scale factor  $s$ . The degree of freedom of similarity transformation is 4 in 2D cases.

$$y(x; s, R, t) = sRx + t$$

#### Pre-processing

The first step is pre-processing, we downscale all the images to half the size (a quarter of the number of pixels) to reduce the runtime with MATLAB command `imresize`. We also convert them to grayscale to make them suitable for the following process. For the TRF images in task 2, I also perform the inversion and histogram equalization. Because the TRF images have inverse intensity as the HE images, the background is black and the tissue is white. The histogram equalization transforms the grayscale image so that the histogram of the output grayscale image has 64 bins and is approximately flat.

## SIFT features detection

For the landmarks extraction, we use the SIFT features descriptor. It is a feature detection algorithm invented by David Lowe in 1999. In this algorithm, SIFT key points of objects are first extracted from a set of reference images. An object is recognized in a new image by individually comparing each feature from the new image to this database and finding candidate matching features based on Euclidean distance of their feature vectors. From the full set of matches, subsets of key points that agree on the object and its location, scale, and orientation in the new image are identified to filter out good matches.

It is revealed by the name that this algorithm is invariant to image scale, that is why it is suitable for our task 2.

We use MATLAB command `matchFeatures` to find preliminary matchings of pairs of keypoints using the descriptors. The function `matchFeatures` has two parameters “MatchThreshold” and “MaxRatio” need to be tuned. For task 1, I use 8 for “MatchThreshold” and 0.6 for “MaxRatio”; for task 2, I use 8 for “MatchThreshold” and 0.7 for “MaxRatio”. Those parameters are chosen by trying out, not too little then there are not enough key points pairs generated for RANSAC to use, not too much then there are many incorrect pairs.

## Finding transformations from landmarks

To obtain the parameters of transformations from some pairs of landmarks, we are actually solving an optimization problem:

$$\min \sum_{i=1}^N \|y_i - t - sRx_i\|^2.$$

We can implement an algorithm given by the lecture to solve this problem. It is based on singular value decomposition of a correlation matrix for the data.

## RANSAC

After finding the preliminary matches by SIFT, we perform RANSAC to choose a set of optimal key points with a maximal number of inliers.

First, we randomly choose a number of key points depending on the degree of freedom of the

model which is 3 for rigid transformation and 4 for similarity transformation. Then we calculate a transformation model from this set of key points, and perform an evaluation on the other key points. The evaluation method can be the distance from the points to the model representing error. If the error is low then it can be counted as inlier, vice versa. After randomly selecting the points and iterating many times, we can get an optimal model with parameters recorded, this optimal model gives the maximal number of inliers. Finally, we use this model to perform our transformation.

There are two parameters need to be chosen for RANSAC, one is the number of iterations, the other is the evaluation threshold. For the number of iterations, it can be calculated by a statistic model,  $L$  is the number of how many iterations it need for a certain success rate, given the probability of inliers.

$$L = \frac{\log(1 - P_{success})}{\log(1 - P_{inlier}^N)}$$

In our case,  $N$  is 3 and 4 respectively. If the success rate is 0.9, the probability of inliers is 0.5. The  $L$  are around 18 and 36 based on calculation. I increased the number of iterations to 100 because we are not sure about the probability of inliers.

For the evaluation threshold, I set the distance as 5. If the distance between the transformed point and its corresponding point is lower than 5, it will be considered as an inlier.

## Results

### Task 1:

Table 1: Transformation parameters for data collection 1

Pair	Rotation angle in degree	Magnitude of translation	Assessment
1	3.7753799	15.315951	successful
2	7.1715145	28.380377	successful
3	4.4841914	27.406898	successful
4	5.1213193	28.085562	successful
5	4.8899240	31.111631	successful
6	6.1967249	29.873571	successful
7	4.7565813	38.498768	successful
8	5.7723088	37.482277	successful
9	4.3841863	22.720339	successful
10	1.8029828	25.072701	successful
11	3.2405717	23.378321	successful
12	5.0099282	42.302979	successful
13	3.8678656	42.643871	successful
14	116.97438	355.00977	unsuccessful

15	2.5689504	26.877899	successful
16	2.8977971	14.546814	successful
17	2.7773490	32.868435	successful
18	23.596786	538.63153	unsuccessful
19	4.6996546	29.511806	successful
20	3.4601738	47.248077	successful
21	3.1207325	19.530659	successful
22	135.94609	489.18723	unsuccessful
23	33.033791	252.30566	unsuccessful
24	2.9518092	51.466843	successful
25	3.9829040	30.611416	successful
26	2.0430601	68.873558	successful
27	4.0239930	40.941711	successful
28	4.1869426	30.683752	successful
29	4.0636554	39.839897	successful

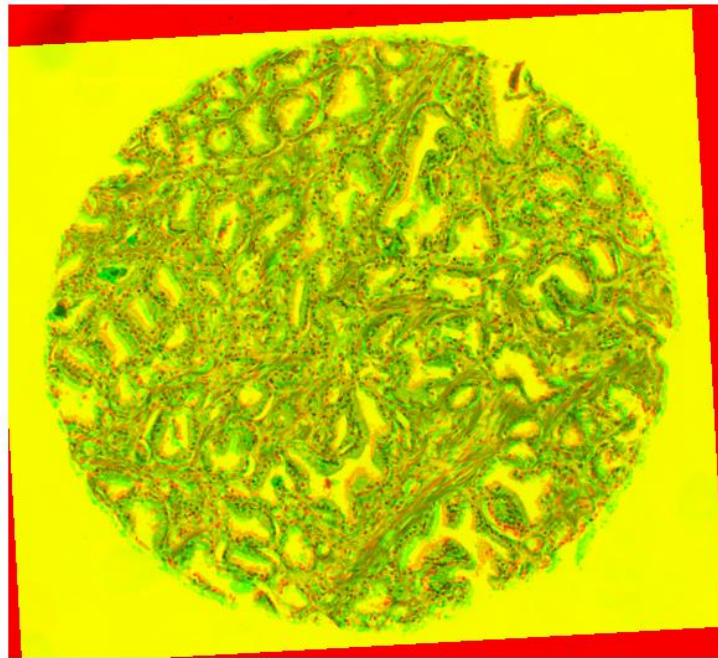


Figure 1: Successful alignment example image for task 1

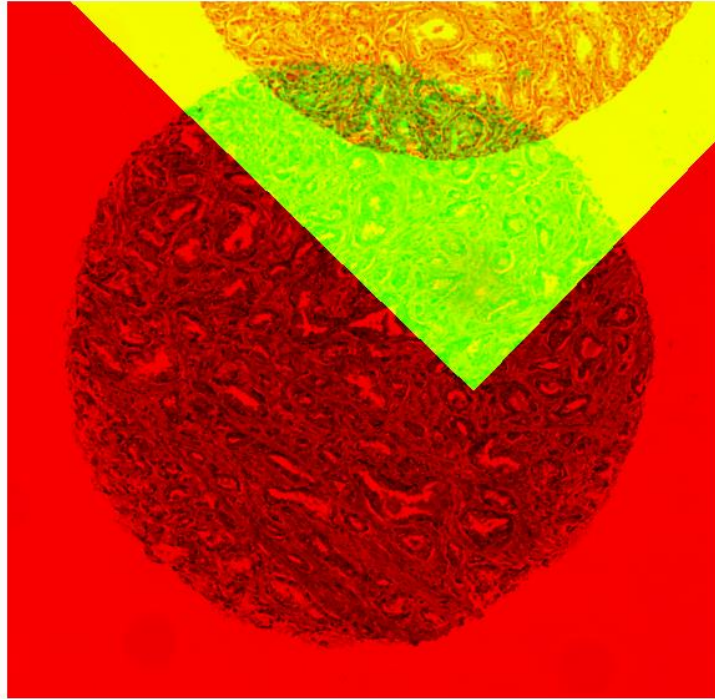


Figure 2: Unsuccessful alignment example image for task 1

As seen in the table there are a few pairs of images which were not aligned successfully. One of the unsuccessful examples is shown above. I think the reason it fails is mainly about the SIFT features extracted. Because I checked the percentage of inliers after performing RANSAC, it is very low for those failed pairs. It means none of the set of extracted features are all correct. So in those cases, the SIFT can not obtain useful matching points.

For the rest of the images, they look overall good, even if some of them are not perfectly aligned. But I think it is due to the images are not identical.

## Task 2:

Table 2: Transformation parameters for data collection 2

Pair	Rotation angle in degree	Magnitude of translation	Scale factor	Assessment
1	94.718491	655.74750	0.43374878	Successful
2	95.320953	353.49103	0.44229433	Successful
3	133.44952	523.28583	0.19799694	Unsuccessful
4	94.764221	795.83618	0.46910381	Successful
5	94.459557	466.80750	0.37217885	Successful
6	94.459557	655.13293	0.39328644	Successful
7	94.939194	656.87805	0.40732324	Successful
8	95.238144	549.41571	0.43226114	Successful
9	95.162735	678.60626	0.44062242	Successful
10	53.566433	505.80975	0.48088738	Successful

11	95.036209	418.47244	0.47807989	Successful
12	94.724800	313.44568	0.46685398	Successful

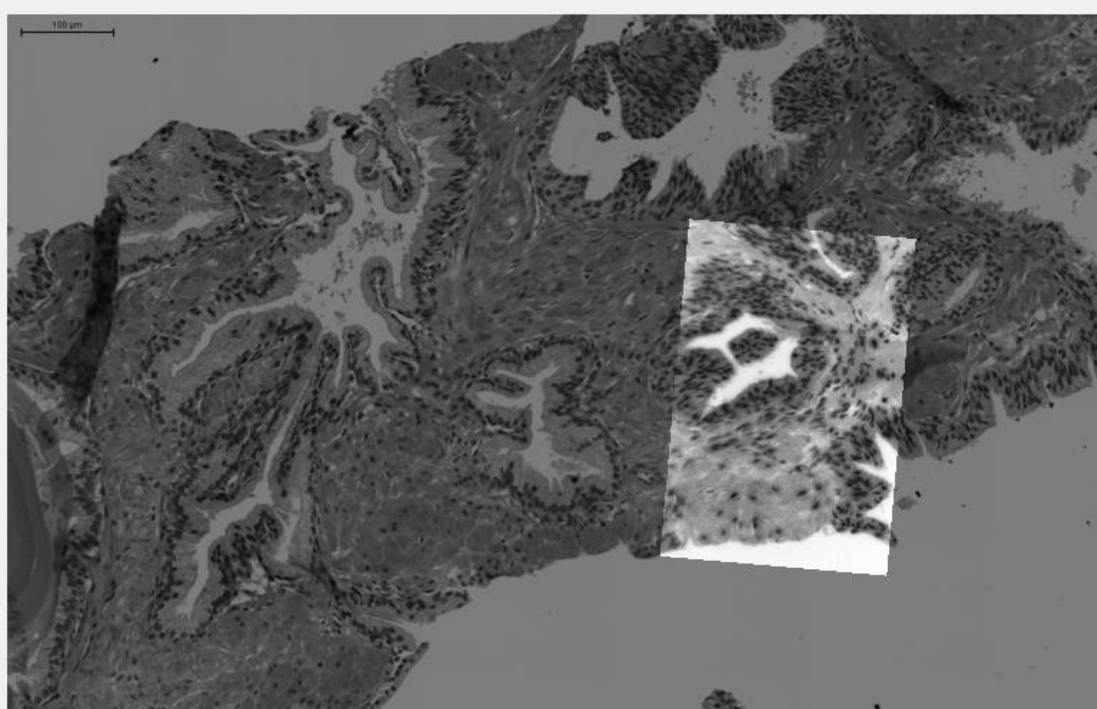


Figure 3: Successful alignment example image for task 2

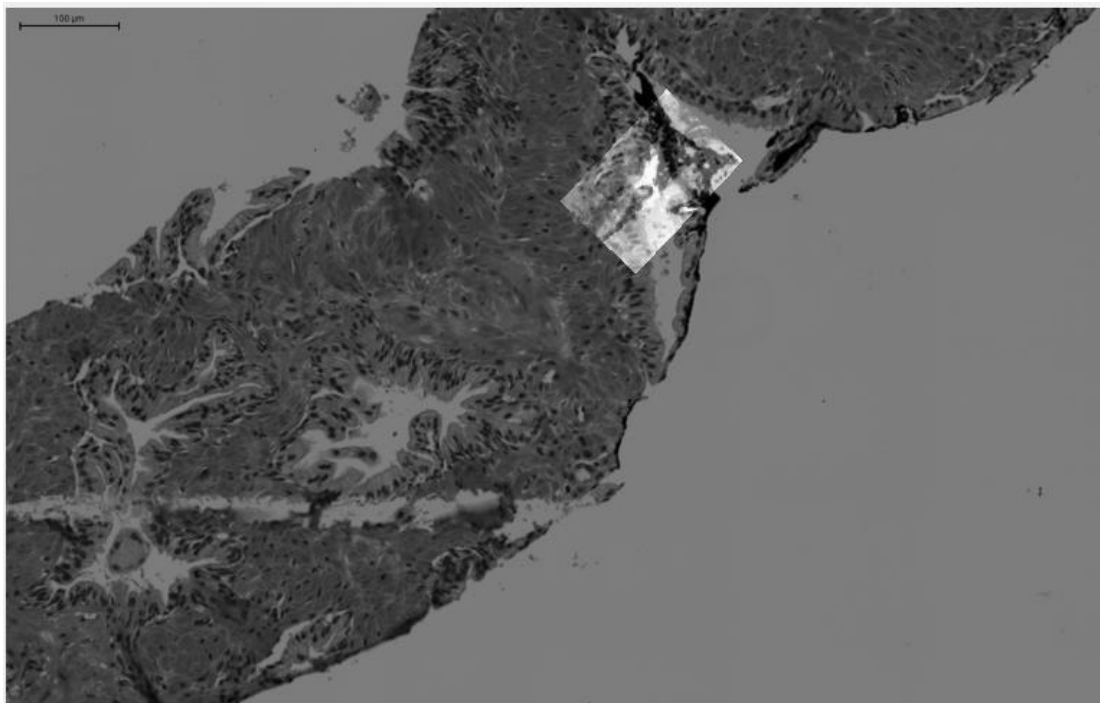


Figure 4: Unsuccessful alignment example image for task 2

For this task, I find there is only one pair of images which are not aligned successfully as shown above. For all the other pairs, my implementation seems to work well and align them successfully.

### Task 3:

I performed the quantitative evaluation method provided in the paper by Lippolis et al... The outline is, first, manually select 4-5 pairs of matching points. We perform our registration algorithm and get a manual transformation  $T_{\text{manual}}$ . Because we choose the corresponding points manually, we believe this transformation is optimal.

Next, the intrinsic uncertainty of  $T_{\text{manual}}$  is estimated by the residuals error  $\epsilon$

$$\epsilon^i = \mathbf{y}^i - T_{\text{manual}}(\mathbf{x}^i)$$

The intrinsic uncertainty is defined as the standard deviation of the length of residuals.

$$\epsilon_{\text{manual}}^2 = \frac{1}{N-1} \sum_{i=1}^N |\epsilon^i|^2.$$

The next step is to perform the same process with the transformation we obtained by our algorithm. We get intrinsic uncertainty of the models we get as in task 1.

$$\epsilon_{\text{auto}}^2 = \frac{1}{N-1} \sum_{i=1}^N |\epsilon_{\text{auto}}^i|^2.$$

Finally, we define our registration models to be quantitatively correct if

$$\epsilon_{\text{auto}} \leq \epsilon_{\text{manual}} + 5 \text{ pixels}$$

I perform this evaluation method on four image pairs: 1, 6 which I visually evaluate as correct. And 18,22 which I visually evaluate as incorrect.

The result is the same as visually evaluation. For the 1 and 6, they are quantitatively correct, for 18 and 22 they are incorrect.

**Table 3: Results of objective assessment**

Pair	$\epsilon_{\text{manual}}$	$\epsilon_{\text{auto}}$	Assessment
1	2.77741549504469	3.6136234	Correct
6	3.98634126176353	5.5980945	Correct
18	18.3342303952738	148.92554	Incorrect
23	2.48901494379682	18.855843	Incorrect

## Conclusion:

For task 1, some of the image pairs are not aligned correctly, I think the reason it fails is mainly about the SIFT features extracted. I checked the percentage of inliers after performing RANSAC, it is very low for those failed pairs. It means none of the set of extracted features are all correct. So in those cases, the SIFT fail to obtain useful matching points.

For task 2, most of the image pairs are aligned successfully, according to visual assessment. That is because SIFT features descriptor is able to extract and match features and keypoints from images in different scales. That is a very powerful tool.

For task 3, I learnt that it is not enough for a scientific paper to assess success in subjective way. Thanks to the method provided in the paper by Lippolis et al., I performed an objective assessment to the registration task, and it worked well.

Overall, this assignment allows me to obtain some insights into image registration and how useful it is in medical image analysis field. However, my basic implement is still not perfect when dealing with some difficult cases.