MIRA FINAL PROJECT

Image registration of chest CT volumes

January 10, 2018

Yeman Brhane Hagos
Vu Hoang Minh

1 INTRODUCTION

Image registration is the process of aligning and transforming a 2D or 3D image or multiple images into a reference image. The sources of difference could be from numerous viewpoints, different sensors, times or depths [1]. In the field of medical imaging, where images are acquired from variety of modalities, for example, X-ray scanners, Magnetic Resonance Imaging (MRI) scanners, Computed Tomography (CT) scanners, and Ultrasound scanners, image registration, is an essential tool; because it helps to combine patient data to yield additional anatomy information by finding the related spatial information [2].

Deformable Image Registration (DIR) has many exciting potential applications in diagnostic medical imaging and radiation oncology. Automated propagation of physician-drawn contours to multiple image volumes, functional imaging, and 4D dose accumulation in thoracic radiotherapy are just a few examples. However, before such applications can be successfully and safely implemented, we require that the DIR spatial accuracy performance is rigorously and objectively assessed.

In this project, we are provided four cases including intensity volumes (inhale and exhale) plus landmark points (see figure 1) to 'train' that is to search for the optimal parameters in registration from the moving chest to the fixed one. The goals of this project are to build a robust registration method to predict 300 fixed landmarks based on provided inputs: inhale, exhale volumes and 300 moving landmarks coordinates and evaluate by 3D Euclidean distance between transformed landmarks (TRE).

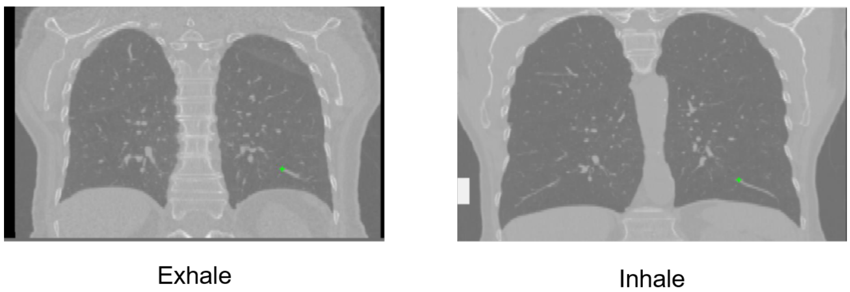


Figure 1: Image registration of chest CT: exhale to inhale volumes

2 SOFTWARE AND TOOLS USED

In this section, software and tools used in the project are discussed:

First, Insight Segmentation and Registration Toolkit (ITK) is an open-source, cross-platform system that provides developers with an extensive suite of software tools for image analysis. Developed through extreme programming methodologies, ITK employs leading-edge algorithms for registering and segmenting multidimensional data [3]. In this project, we use ITK to visualize the difference between unregistered and registered images compared to the reference one, evaluate qualitatively and convert .raw to .nii format with the proper orientation.

Second, Elastix is a toolbox for rigid and nonrigid registration of images which is based on the well-known ITK. This software includes a group of algorithms for medical image registration [4]. In this project, we adopt Elastix to perform the registration using a non-rigid method and predict fixed landmarks by applying found transformation in the registration step on moving landmarks.

Third, Matlab Utility Pack, which is provided by the group organizing this registration challenge, includes basic utility functions for manipulating 3D raw image volumes in Matlab. The included features require the Matlab Image Processing Toolbox and should be added to the Matlab path prior to use. In this project, we use this package to visualize the landmarks.

Advanced Normalization Tools (ANTs) is the state-of-the-art medical image registration and segmentation toolkit which is built from ITK. In this project, we tried to use ANTs for its Diffeomorphic Transformation with the support of adding landmark guidance. However, in the end due to time constraints, we could not figure out how to

Finally, Matlab is adopted to preprocess CT chest volume, call Elastix executables, and compute TRE.

3 Registration Algorithm Analysis

Image registration is the process of aligning and combining data coming from more than one image source into a unique coordinate system. This problem has become one of the pillars of computer vision and medical imaging.

Suppose that we have two images: moving, I_M, and fixed, I_F. The registration process of I_M into I_F is equivalent to finding the transformation matrix T that makes I_M spatially aligned to I_F. In theory, the registration problem equals to the optimization of T with regards to the cost function C [4] as follows:

$$\hat{T} = \underset{T}{\operatorname{argmin}} C(T; I_F, I_M), \tag{1}$$

$$C(T; I_F, I_M) = -S(T; I_F, I_M) + \gamma P(T), \tag{2}$$

where S denotes the spatial difference between I_F and I_M, γ weighs similarity against regularity and P is regularization term. The basic registration components of a registration problem can be seen in figure 2.

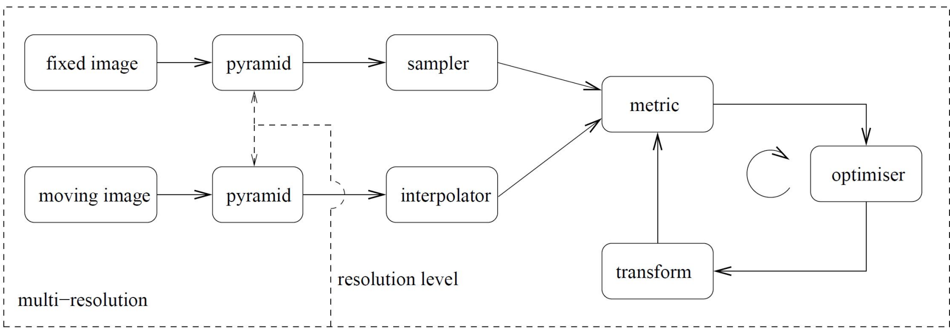


Figure 2: The basic registration components. This figure is adopted from the manual of Elastix [4].

Later, in section 4.4, we will discuss in detail how to select registration parameters and how different parameters affect the quantitative result. Now, let's examine the purpose and options in each component [4] briefly.

First, **metric** computes the similarity between the fixed and moving image. There are several choices in

Elastix: Mean Squared Difference (MSD), Normalised Correlation Coefficient (NCC), Mutual Information (MI) and Normalized Mutual Information (NMI). Second, **interpolator** estimates pixel intensities given its current position and neighborhood pixel. There are three options in Elastix: nearest neighbor, which is the simplest method and requires the least resources, linear and B-spline. Third, **transform** is the movement of the moving image to become spatially aligned with the fixed image. There are some methods, for example, translation, rigid, similarity, affine, B-spline and thin-plate splines. Fourth, **optimizer** is used to find the optimal transformation parameters in an iterative optimization strategy [4]. Fifth, **sampler** component samples pixels in the fixed image. Here, we could sample all pixels, but it is unnecessary. Finally, **pyramid** is one level of multi-resolution for the registration problem. It is advised that the number of resolution should be set to 1 if the expected deformations are small, to 3 or 4 for the general case, and to 5 or 6 for large images and large deformations.

4 IMPLEMENTATIONS

In this section, we will discuss step by step on how we improve the quantitative result that is the mean TRE of four cases in the training set. The result before and after each step is recorded to summarize registration accuracy.

4.1 Dataset

We are given four cases. Each case includes:

- CaseID_300_iBH_xyz_r1.txt
- CaseID_300_eBH_xyz_r1.txt
- CaseID_iBHCT.img
- CaseID_eBHCT.img

The CaseID_300_iBH_xyz_r1.txt and CaseID_300_eBH_xyz_r1.txt text files contain a list of 300 landmark locations of superior/inferior (SI) coordinate locations corresponding to the respective iBHCT and eBHCT component phase images from a 4DCT set. The iBHCT label represents the maximum inhale phase of the 4DCT, while the eBHCT label corresponds to the maximum exhale phase.

In the challenge, we will be given CaseID_300_eBH_xyz_r1.txt, CaseID_iBHCT.img and CaseID_eBHCT.img. Our task is to predict CaseID_300_iBH_xyz_r1.txt.

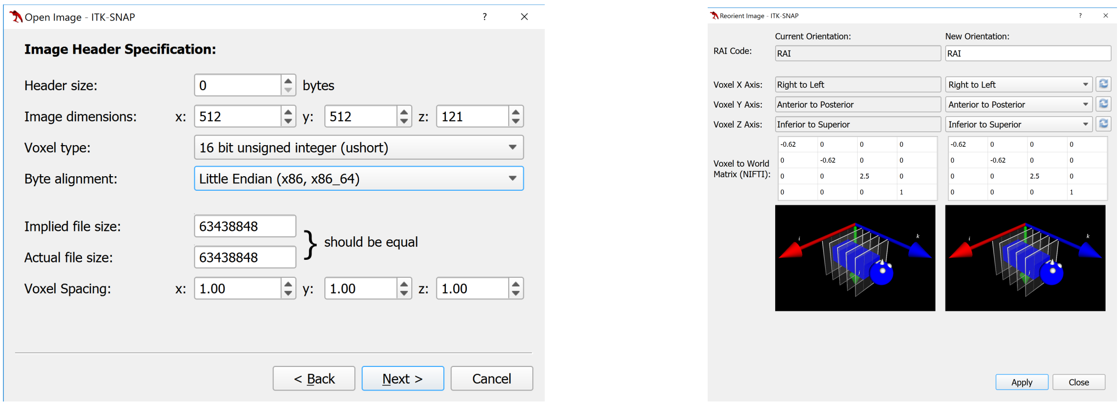
4.2 Data Preparation

We start off by reading volume in *.raw* format into ITK. One issue, which can occur when reading *.raw* data, is wrong declaration of Image Dimensions and Voxel Type. For example, the correct dimensions of Case 1 is (512, 512, 121) dimensions and 16 bit unsigned integer voxel type (see figure 3(a)).

After loading *.raw* volume, we can correct the orientation and save the data back in *.nii* format. As can be seen from figure 3(b), the correct orientation is 'Anterior to Posterior' for Y-Axis and 'Posterior to Anterior' for Z-Axis.

4.3 Preprocessing

We start with a 'naive' method that is registering exhale to inhale volume and predicting fixed landmarks without using any preprocessing techniques. As expected, the mean TRE of four cases is **23.12**



(a) (b)

Figure 3: (a) Image header specification and (b) Orientation correction of *.raw* format before saving to *.nii* for further processing

which is even greater than **22.72** (without registration). It means that preprocessing techniques play an important role in this project.

The first preprocessing method we have tried is finding the masks of both exhale and inhale volumes. The purpose of the masks is to filter out unwanted information and keep only the essential one. Eight steps in figure 4 demonstrate how the masks are found:

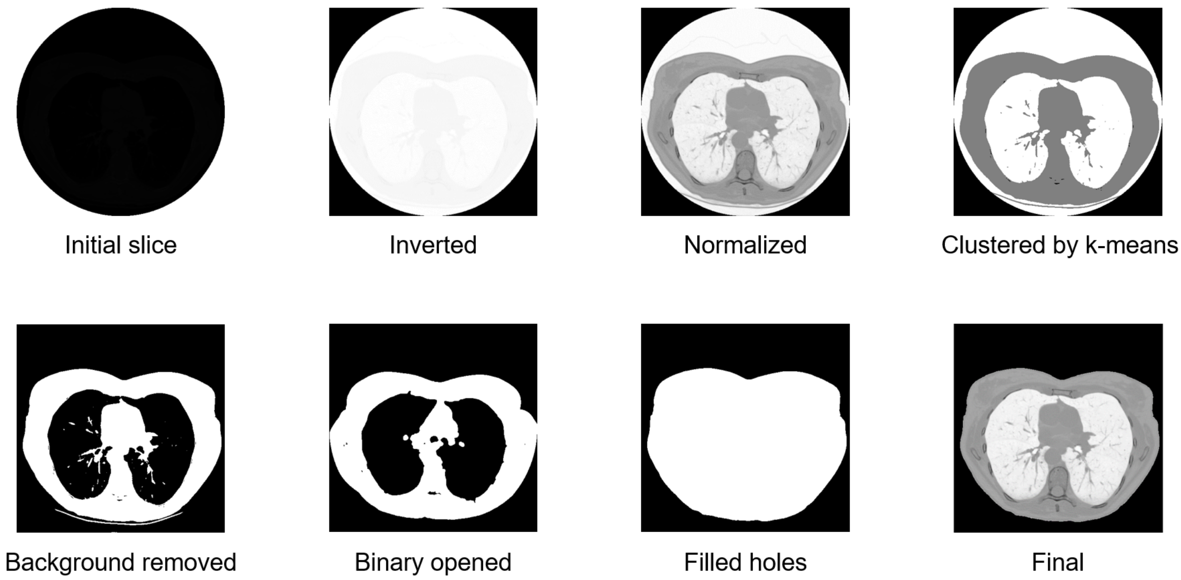


Figure 4: Eight steps to find mask and keep only essential voxels

- Step 1: we extract slice by slice in a volume and process on it.
- Step 2: slice is inverted to obtain better visualization of the data.
- Step 3: normalize the inverted slice from 0 to 1.
- Step 4: cluster normalized slice by k-means when k=3.
- Step 5: remove background which contains the brightest cluster and the darkest cluster, we keep

only the mid one.
- Step 6: remove unwanted artifact which is a thin line below the back using binary opening with a 4-pixel-radius disk.
- Step 7: fill holes to find the mask.
- Step 8: extract data only inside the mask.

After extracting foreground voxels from each slice, the masked volume containing only foreground voxels is reconstructed and saved for registration.

We use Elastix's functions: *elastix* to register exhale to inhale volume and find the transformation matrix, T, and *transformix* to predict fixed landmarks from moving landmarks after applying T. Eventually, we design our two batch files as simple as follows:

regmask.bat
```
elastix -f temp/ImageFixed -m temp/ImageMoving -fMask temp/FixedMask -mMask temp/-
MovingMask -out temp -p parameters/NameParameter
```

trans.bat
```
transformix -def DirLandmark/NameLandmark -out temp -tp temp/TransformParameters.0.txt
```

where:

- *DirImageFixed* is the directory of the fixed image,
- *ImageFixed* is the filename of the fixed image.
- *DirImageMoving* is the directory of the moving image.
- *ImageMoving* is the filename of the moving image.
- *FixedMask* is the fixed mask of exhale volume.
- *MovingMask* is the moving mask of inhale volume.
- *parameters* is the name of the folder containing a number of Parameter files. Here, we design a loop to run registration on all parameter files to select the best.
- *NameParameter* is the name of the Parameter files.
- *DirLandmark* is the directory of the landmark text file.
- *NameLandmark* is the filename of the landmark text file.
- *TransformParameters.0.txt* is the parameters of B-spline transformation T_B that relates the fixed and the moving volume.
- *temp* is the temporary folder containing files that are only needed temporally in one iteration. At the end of one iteration, the essential outputs will be moved to the specific folders.

After applying the mask, the mean TRE improves as it decreases to **18.12**. Here, we observe that besides preprocessing method, Elastix's registration parameter is also critical. In next section, we will discuss what the purpose of each parameter is and how it affects the overall result.

4.4 Elastix's Registration Parameters and the First Improvement: Parameters Tuning

In fact, there are plenty of parameters to tune in a parameter file. However, there are a few essential parameters which play key roles. They are listed below in table 1. Table 1 includes four columns. The first column represents the parameters. The second, third and fourth columns demonstrate the

Table 1: Elastix's essential parameters

Parameter	**Default**	**First Tuned**	**Final tuned**
FinalGridSpacingInPhysicalUnits	16	10	10
NumberOfResolutions	4	5	6
GridSpacingSchedule	4 4 2 1	8 8 4 2 1	16 8 8 4 2 1
ImagePyramidSchedule	$8^3 4^3 2^3 1^3$	$16^3 8^3 4^3 2^3 1^3$	$16^3 8^3 4^3 2^3 1^3 1^3$
MaximumNumberOfIterations	500	2000	10000
MaximumStepLength	1	1	5
ImageSampler	Random	RandomCoordinate	RandomCoordinate
Metric	AMMI	ANC	ANC

selections of parameters in the default, first tuned and final tuned, respectively.

Now, let's examine what the purpose of each parameter is:

FinalGridSpacingInPhysicalUnits is the parameter controlling point spacing of the B-spline transformation in the finest resolution level. The unit of its is *mm*. The lower this value, the more flexible the deformation. In fact, low values may improve the accuracy; however, they may also cause unrealistic deformations that make the registration hence prediction worse. For this parameter, we have tried with a few values: 4, 6, 8, 10. The best result is at 10. There is one parameter , *FinalGridSpacingInVoxels*, which is interchangeable for this parameter. The difference is *FinalGridSpacingInVoxels* is in voxel unit while *FinalGridSpacingInPhysicalUnits* is in physical unit.

NumberOfResolutions is the number of resolutions (pyramids). Since we expect a large deformation in this project, the larger the value, the better the result. However, processing time should be taken into account as it increases dramatically when the number of resolutions is set too high. For example at the value of 8, it could take twice as much as processing time as 4 if the *GridSpacingSchedule* in every resolution is the same. Furthermore, the value of 7 or 8 does not outperform the value of 6. It means that there is a negligible improvement. Thus, our final selection for this parameter is 5 in the first tunning and 6 in the final tunning.

GridSpacingSchedule defines the grid spacing in each resolution. By default, the value is halved after every resolution, such that the final grid spacing is obtained in the last resolution level. If there are 6 resolutions, we need to declare 6 values accordingly for this parameter. After some trials, we finalize with (16 8 8 4 2 1) for this parameter.

ImagePyramidSchedule controls the downsampling/blurring factors for the image pyramids. Similar to *GridSpacingSchedule*, by default, the images are downsampled by a factor of 2 compared to the next resolution. Note that, the number of elements of this parameter equals the number of resolutions times the image dimension. For example, if we have 4 resolutions and 3D volume, the number of elements for this parameter is 12.

MaximumNumberOfIterations is one of the most important parameters in Elastix's registration. It defines the maximum number of iterations in each resolution level. This value is proportional to processing time. For example, with CPU i7-6700HQ, it takes 10 minutes for registration with the final-tuned parameters in table 1. When *MaximumNumberOfIterations=40000*, the processing time is 32 minutes. Here, we need to find a proper combination of parameters to balance between computation time and accuracy.

MaximumStepLength is the step size of the optimizer, in mm. By default, the voxel size, which equals to 1, is used which usually works well. The greater the value, the more flexible the deformation. If we set this parameter too high, for instance, 10, the deformation, in this case, would be too 'flexible' which could lead to a weird transformation and somehow become unstable. We can relate this parameter to gain K in control theory or learning rate in deep learning.

ImageSampler is the parameter controlling how to sample voxel to register. For this project, *RandomCoordinate* give us the best result. In addition to *Random* and *RandomCoordinate*, we could set this parameter to *Full* which means that Elastix will sample all voxel. However, it is not a good choice because of two reasons. First, it takes a lingering time to process. For example with the same configurations in the final tunning in table 1, it took me 3 hours. Second, the combination of *Random* sampler and *AdaptiveStochasticGradientDescent* of optimizer could do the job.

Metric is the 'loss' function of Elastix. In the default parameter file, it is set to *AdvancedMattesMutualInformation*. In this project, *AdvancedNormalizedCorrelation* gives us the best result.

Next, we will take a look at some parameters which is similar in three cases:

NumberOfSpatialSamples is the number of spatial samples used to compute the mutual information (and its derivative) in each iteration. We set it to 2048.

FinalBSplineInterpolationOrder is the order of B-spline interpolation to apply in the final deformation. We set it to 3 since it gives good accuracy in most cases.

After applying the first tuned parameters in table 1, the mean TRE improves as it decreases to **12.81**.

4.5 Second Improvement: Mask's size Reduction

After applying the first improvement plus the mask in the preprocessing step, we can obtain a result in figure 5.

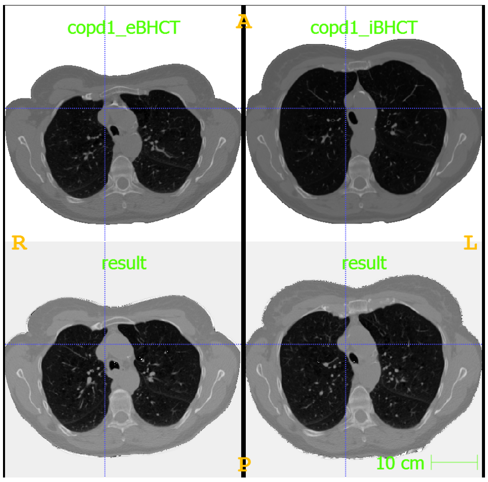


Figure 5: Registration of exhale to inhale volume when applying first tuned parameters

The registration is proper as the shape of the registered, and the reference volume is similar. However, we realize that the outer part including the boundary and bones (very bright and easy to distinguish) outperforms the inner portion including landmarks (see figure 6(a)) which should be paid more attention. Thus, a reduction in mask size excluding outer part mentioned above should be taken into account.

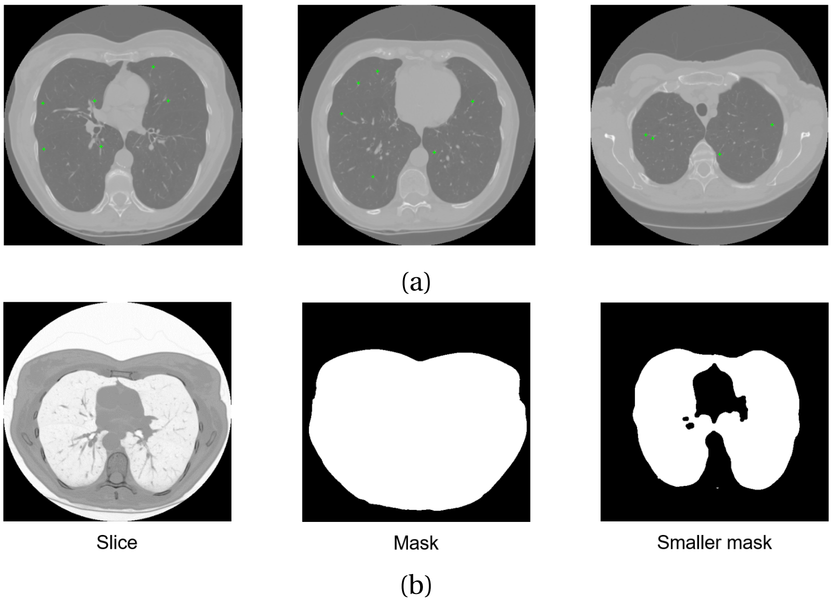


Figure 6: (a) Observation: all landmarks lie inside the inner part and (b) Extracting smaller mask from the original mask

To find the reduced mask is very straightforward (see figure 6(b)): there are two distinct regions which can be seen in the final result of figure 4. The idea is to take the brighter areas after clustering by k-means ($k = 2$).

After applying the fine-tuned mask in the registration, the result improves to **8.47**.

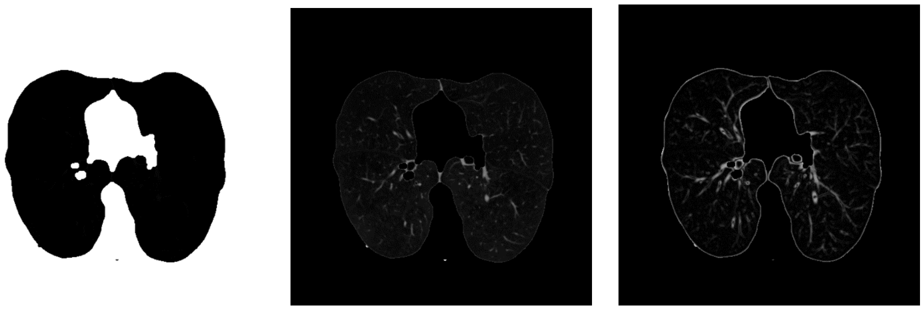
4.6 Third Improvement: Volume Normalization and Gradient

Instead of using original volume, we propose to adopt normalized and gradient volume. As a result, with the final tuned parameters in table 1, the mean TRE has improved tremendously from **8.47** from the last step to **5.3** and **2.95**, respectively.

It can be seen that the intensities are enhanced from the original to the normalized slice (see figure 7(a-b)). In addition to the intensity enhancement, there is another reason why the normalized image in specific and normalized data in general works better than the original one: normalized data is brought into a range that similar or normal to the distribution of the whole dataset, while uncleaned data often produce the noise, thus lead to a deterioration in result.

Compared to the normalized image, there is a significant boost in the 'quality' of gradient image (refer to figure 7(c)). The image becomes sharper and brighter, thus help Elastix identify key features more easily.

To the best of our knowledge, there are two reasons for such improvement. First, most of the landmarks



(a) Original (b) Normalized (c) Gradient

Figure 7: (a) Original, (b) Normalized and (c) Gradient

lie on the edges of the artifacts. Since image gradient measures the change in intensity of the same point (in other words, to find edges); it makes the features points, which might include landmarks, become more obvious to be matched between moving and fixed. Second, image gradient works well in this case because of the small size of the fined-tuned mask which was mentioned in section 4.5. In specific, there are fewer features to detect and match. Thus, matching key points between two volumes become easier, hence more accurate.

4.7 Failure

We also tried two enhancement techniques (non-local means denoising and histogram matching), but they make the final TRE deteriorate.

For non-local means, it is reasonable because our best approach adopts image gradient which aims to amplify edges. Here, denoising techniques destroy the structures of artifacts, hence worsen the landmark predictions. Moreover, given four training cases are not noisy.

In fact, we expect that histogram matching technique would enhance the result, but finally, it did not. The reason may be due to our implementation. We leave this issue for the future task.

5 Result and Discussion

The deformable registration, which was performed on a duo-core processor achieves very fast runtime of on average **7 minutes** per case when *MaximumNumberOfIterations* and *NumerOfResolutions* are set to 10000 and 6, respectively.

Figure 8(a) shows the first ten landmarks of exhale, predicted inhale and inhale groundtruth of case 1, respectively. As can be seen, we obtain a decent result in spite of using only an existing registration tool (Elastix) and a set of comprehensive preprocessing techniques.

In addition, registration accuracy results of DIR evaluation are published and updated online, which allows for a broad comparison of methods. Figure 8(b) shows the mean TRE for four 4D-CT cases, comparing to the best performing published methods. The results on this database indicate that our approach is acceptable given a short-time project. In specific, our approach is merely 1.8mm worse than the state-of-the-art methods (isoPTV and DIC-CO).

Point number	Exhale landmarks	Predicted inhale landmarks	Inhale landmarks
1	194 257 7	188 257 4	188 260 4
2	186 283 9	179 281 7	179 281 7
3	194 255 11	187 250 9	187 253 8
4	169 319 10	160 314 9	161 315 9
5	181 250 14	173 238 11	173 241 9
6	313 256 14	317 241 10	317 242 10
7	308 321 11	310 318 10	310 319 10
8	281 276 13	282 271 11	282 273 11
9	204 240 15	200 231 13	198 234 11
10	220 259 14	218 255 12	215 256 12

(a)

Algorithm	First volume	Second volume	Third volume	Fourth volume
NLR	1.33 (1.55)	2.34 (2.88)	1.12 (1.07)	1.54 (1.61)
LMP	1.21 (1.46)	1.97 (2.38)	1.06 (0.96)	1.64 (1.75)
MILO	0.93 (0.92)	1.77 (1.92)	0.99 (0.91)	1.14 (1.04)
SGM3D	1.22 (2.73)	2.48 (3.79)	1.01 (0.93)	2.42 (3.56)
MRF	1.00 (0.93)	1.62 (1.78)	1.00 (1.06)	1.08 (1.05)
meLDDMM	0.90 (0.93)	1.56 (1.67)	1.03 (0.99)	0.94 (0.98)
isoPTV	0.77 (0.75)	2.22 (2.94)	0.82 (0.80)	0.85 (0.86)
DIS-CO	0.79 (0.85)	1.46 (2.28)	0.84 (0.82)	0.74 (0.86)
No DIR	25.90 (11.57)	21.77 (6.46)	12.29 (6.39)	30.90 (13.49)
Our method	2.44	4.01	1.73	3.36

(b)

Figure 8: (a) First ten landmarks of exhale, predicted inhale and inhale groundtruth of case 1, and (b) Comparison of the TRE for four 4D-CT cases, comparing to the best performing published methods

6 Project Management

As the requirement of this lab is to implement *elastix*, *transformix* batch files and build an image registration of chest CT volumes; we scheduled a proper plan to meet the deadline and finish the work within the given time. The schedule of our meetings and an internal progress report is given in the following Table 2. All the deadlines were decided by mutually discussing and selecting the best approach scenario. An internal progress report was also maintained which was updated daily based on the work done or problem faced and that needs to be solved. Table 2 shows the internal progress report.

The following tools were used for completing and managing the coursework and keeping track of the proposed deadlines:

- Shared Google Drive for updating and writing the report together

Table 2: Internal progress report

Name	Work Assigned	Problems Faced	Next step
		Week 1	
Minh	Design batch files	No	Fix bugs. Update report
Yeman	Find mask	No	Update report
		Week 2	
Minh	Reduce mask size	No	Prepare slides
Yeman	Test histogram matching	No	Prepare report
		Week 3	
Minh	Predict on multiple param files	No	Finalize codes. Update report
Yeman	Test non-local means	No	Update report
		Week 4	
Minh	Test ANTs	No	Finalize report
Yeman	Test multiple paramfiles	No	Finalize report

- GoogleDocs for keeping track of individual team member progress and problems in hand
- Facebook for communication
- Skype for fixing bugs together

7 Conclusion

In this project, we (i) understand properly how registration works and how to perform a single registration of two 3D volumes using nonrigid, (ii) develop a script automating registering and transforming a whole training set (volumes and landmarks) to a reference volume, (iii) develop thorough preprocessing techniques to obtain a robust registration in a large deformation scenario (exhale to inhale).

By the end of this project, we successfully build a robust registration methodology and obtain a low target registration errors. As a result, we have won the first price for non-rigid 3D Lung CT registration challenge organized by MAIA.

References

[1] Lisa Gottesfeld Brown. A survey of image registration techniques. *ACM computing surveys (CSUR)*, 24(4):325–376, 1992.

[2] Terry S Yoo, Michael J Ackerman, William E Lorensen, Will Schroeder, Vikram Chalana, Stephen Aylward, Dimitris Metaxas, and Ross Whitaker. Engineering and algorithm design for an image processing api: a technical report on itk-the insight toolkit. *Studies in health technology and informatics*, pages 586–592, 2002.

[3] ITK. Itk, 2017.

[4] Stefan Klein, Marius Staring, Keelin Murphy, Max A Viergever, and Josien PW Pluim. Elastix: a toolbox for intensity-based medical image registration. *IEEE transactions on medical imaging*, 29(1):196–205, 2010.