Universitat De Girona

Medical Image Registration and Applications

Intensity Based Image Registration

Students:
Ali Berrada
Ama Katseena Yawson

Supervisor:
Robert Martí

October 28, 2018

Contents

List of Figures

List of Tables

1 Introduction

1.1 Overview

Image registration is the challenge of finding the optimal geometric transformation that aligns a moving image to a fixed image. It plays an important role in several domains including medical imaging and computer vision. The most typical and simple procedure of registering two images is summarized in the diagram below (Figure 1).

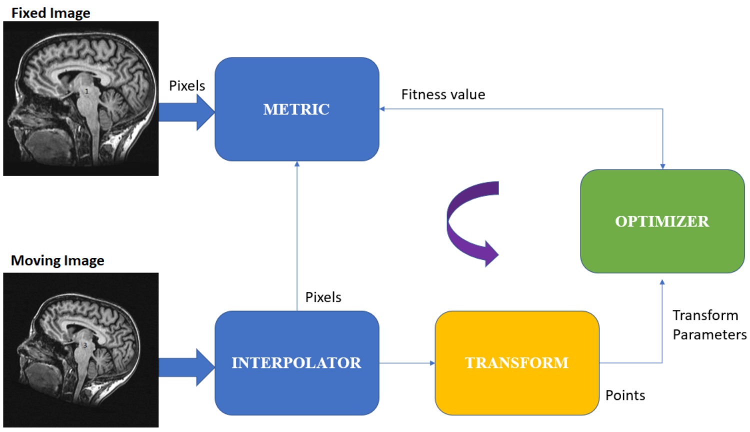

Figure 1: A simplified image registration framework

- **Fixed image**: it is the reference image that a second image will align to.
- **Moving image**: it represents the target image to which geometric transforms must be applied so that it aligns with the fixed image.
- **Metric**: it is the numerical value reflecting the degree to which a transformed image correctly aligns to a fixed image. It is an important element of the registration framework as it is used by the optimizer function when searching for the best parameters. Sum of squared difference and normalized cross-correlation are 2 commonly used similarity metrics.
- **Transform**: it defines how every point of one image maps to a corresponding point in another image. It is the geometric transformation which can be applied to the moving image to align it to the fixed image. The most generic transform is the affine transform which encompasses more specific transforms such as the rigid (translation and rotation), shearing and scale transforms.

- **Optimizer**: the optimizer adopts an optimization procedure which makes the metric effective and efficient to reach the optimal value. Given an initial configuration of parameters, the optimizer iteratively looks for the optimal solution that maximizes the similarity measure.
- **Interpolator**: it finds the intensity value for a pixel whose mapping from one image to another results in coordinate values that are non-integers by using information of the position of neighbouring pixels. Examples of interpolation methods are the nearest-neighbor interpolation and linear interpolation.

2 Registration Framework

2.1 Line Identification

After thorough reading and analysis of the given Matlab code, the components of the registration framework were found in the following line numbers and files as shown in Table 1.

Component	**Type**	**File name**	**Line number**
Metric	SSD	affine_registration_function	48
Interpolator	Bilinear	image_interpolation	-
Optimizer	fminsearch (Nelder−Mead minimizer)	affineReg2D	31
Transform	Rigid	affineReg2D	45

Table 1: Localization of the components in the Matlab code

2.2 Scale Vector

The scale vector can be termed as a weighting vector. In the optimization process, the weighting vector is used to control the parameters of the transforms to minimize errors with the cost function. In the case of the rigid transform given in the Matlab code, the scale vector was [1 1 0.1]. This scale vector implies that more weight was given to translation in the x and y direction than to rotation. This will help faster convergence for the optimizer as it restrains the rotation parameter's search space to a smaller range. If considering a more general scenario, the scaling factors help in compensating for the discrepancies between the numeric ranges of parameters. The scaling of parameters will then prevent parameters with large numeric values from monopolizing over parameters with small numerical values [2].

2.3 Center of Rotation

The transformation applies a rotation using geometric functions such as the cosine and sine. These functions from the concept of geometry works on the Cartesian frame whose center point is the (0,0) coordinates located at the middle of frame. However, the image coordinate system starts from the top left of the image. Hence, there is a need to convert from the image coordinate system to the Cartesian system before applying the rotation transformation. This is usually done by subtracting the coordinates (i.e. the x and y) of the mean pixel from the coordinates of all pixels. After this operation, the pixel at the center of the image will become the origin of the frame (the point (0,0) in Cartesian coordinate system). Therefore, the center of rotation of the transformation is basically the center of the image being rotated.

3 Implementation

The implementation was made using Matlab R2018b. The 4 brain images provided were used in our experiments to investigate the pros and cons related to each metric and transform type. The implementation details are explained in the following sections.

3.1 Normalized Cross Correlation

Normalized cross correlation (NCC) is a simple but effective method used to measure the similarity between 2 images. It is invariant to linear brightness and contrast variations as a result of the intrinsic normalization factor [3]. This metric was implemented using the relation given in Equation 1. In our implementation, we observed that correlation was maximum in the opposite direction and hence the output was negated to obtain the alignment.

$$R(A,B) = \frac{\sum_{x=0}^{X}\sum_{y=0}^{Y}(A(x,y)-\bar{I_A})(B(x,y)-\bar{I_B})}{\sqrt{\sum_{x=0}^{X}\sum_{y=0}^{Y}(A(x,y)-\bar{I_A})^2\sum_{x=0}^{X}\sum_{y=0}^{Y}(B(x,y)-\bar{I_B})^2}} \quad (1)$$

where $R(A,B)$ is the normalized cross correlation, $A(x,y)$ is image $A's$ intensity value at location (x, y), $B(x, y)$ is image $B's$ intensity value at location (x, y), $\bar{I_A}$ is the mean intensity value of image A, and $\bar{I_B}$ is the mean intensity value of image B.

3.2 Normalized Gradient Cross Correlation

The normalized gradient cross correlation (NGCC) varies slightly from the normalized cross correlation. This method uses the partial derivatives of an intensity image to obtain variations in illumination [4]. The equation used in the implementation of this metric is given by Equation 2 below.

$$GCC(A,B) = \frac{\sum_{x,y}(\frac{\partial f_1(x,y)}{\partial x}\frac{\partial f_2(x,y)}{\partial x} + \frac{\partial f_1(x,y)}{\partial y}\frac{\partial f_2(x,y)}{\partial y})}{\sqrt{\sum_{x,y}(\frac{\partial f_1(x,y)}{\partial x}^2 + \frac{\partial f_1(x,y)}{\partial y}^2)\sum_{x,y}(\frac{\partial f_2(x,y)}{\partial x}^2 + \frac{\partial f_2(x,y)}{\partial y}^2)}} \tag{2}$$

where $GCC(A, B)$ is the normalized gradient cross correlation, $\frac{\partial f_1(x,y)}{\partial x}$ and $\frac{\partial f_1(x,y)}{\partial y}$ are the partial derivatives w.r.t x and y of the first image, $\frac{\partial f_2(x,y)}{\partial x}$ and $\frac{\partial f_2(x,y)}{\partial y}$ are the partial derivatives w.r.t x and y of the second image.

3.3 Affine Transformation

Affine transformation is the transform that combines the rotation, translation, scaling and shearing transform operations. For our experiments, we considered both the 6-parameters and 7-parameters initialization technique. The transformation matrix used in implementing the 6-parameters and 7-parameters initialization are shown Equation (3) and Equation (4) respectively. Our tests showed more robustness for the second approach when not using a multi-resolution registration framework as there is more room for fine tuning the initialization of parameters.

$$A1 = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

$$A2 = \begin{bmatrix} cos(a) & sin(a) & 0 \\ -sin(a) & cos(a) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & b \\ 0 & 1 & c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} d & 0 & 0 \\ 0 & e & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & f & 0 \\ g & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

where a, b, c, d , e, f, g are parameters for the transforms.

3.3.1 Parameters Initialization

If there is no clear intuition about the initial parameters to select for the transformation or if a generic approach is to be taken considering images of different transformations might be injected to the algorithm, it is best to build the initial transformation as an identity transformation. Specifically, for the case of the 6-parameters affine transform in Equation (3), the variables a and e can be set to 1 and the others to 0. On the other hand if considering the 7-parameters affine transform in Equation (4), the variables d, e can be set to 1 while the others to 0 to obtain a series of identity matrices.

3.4 Multi-resolution

The main idea behind this approach is that 2 images are systematically registered at different scales (Figure 2). The registration happens progressively from the lowest resolution (coarsest) of the pyramid to the highest (finest), passing by the intermediate layers. At each resolution, the initial transform parameters are obtained from the computed parameters resulting from the registration of the images at the previous level. We validated this method with the various metrics and transforms and it proved to be very effective in computation time and quality of results. It succeeds in avoiding local minima more often than normal, single resolution registration.

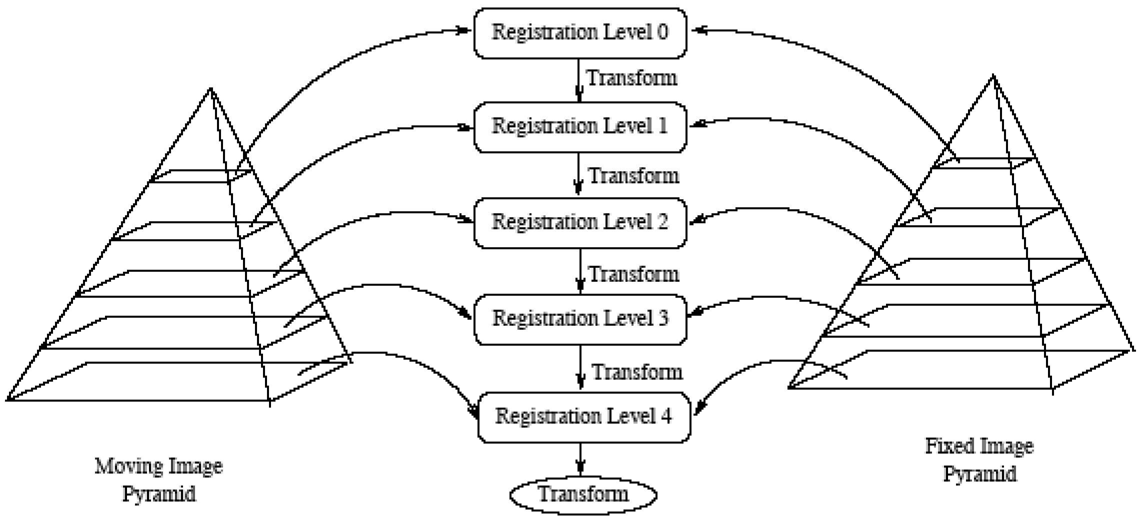

Figure 2: A simplified multi-resolution image registration algorithm[1]

4 Results and Discussions

After performing the image registration on all the given brain images, the results obtained for each case are shown and discussed in this section.

4.1 Brain 1 and Brain 2 registration

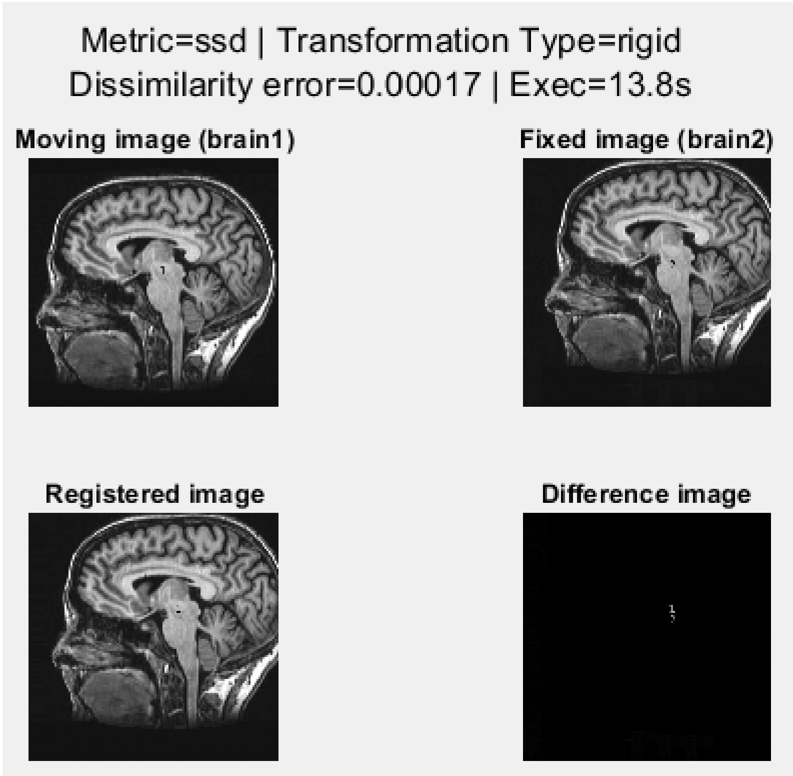

Figure 3: Brain 1 and Brain 2 registration using rigid and ssd.

Brain 1 and Brain 2 images do not suffer from deformations. The target image is basically the translation of the other and hence the rigid transform was enough to successfully register the 2 images using either the SSD or NCC as a similarity metric (Figure 3). When using an affine transformation, the complexity of minimizing the objective function increases, thus it required more time for the process to finish. With the SSD metric, the registration was completed in 69 sec. However, we noticed that the affine transformation along with the NCC metric did not yield the optimal result as the error measured was 0.03 (Figure 4). Using a multi-resolution approach with 3 levels resolved the problem in 24 sec (or even in shorter time if using more levels). The NGCC metric was also tested but fails to correctly converge with a pyramid of 3 levels; however, it worked correctly when the pyramid levels were increased to 5 as shown in Figure 5. We can demonstrate the advantage of the multi-resolution approach by comparing the convergence trend when using multiresolution and not using it. In Figure 6, Brain 1 and Brain 2 images were registered at the original resolution and the optimizer took more than 400 iterations to find the parameters that minimized the objective function. On the other hand, in Figure 7 the optimizer required less steps to find the optimal solution for the last layer of the multi-resolution pyramids.

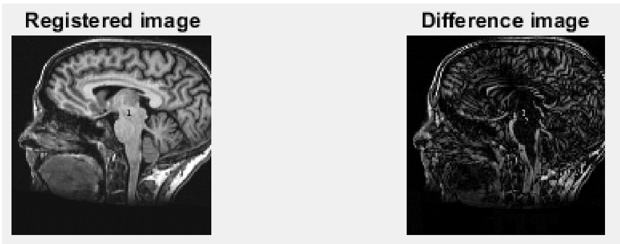

Figure 4: Brain 1 and Brain 2 registration with affine and NCC.

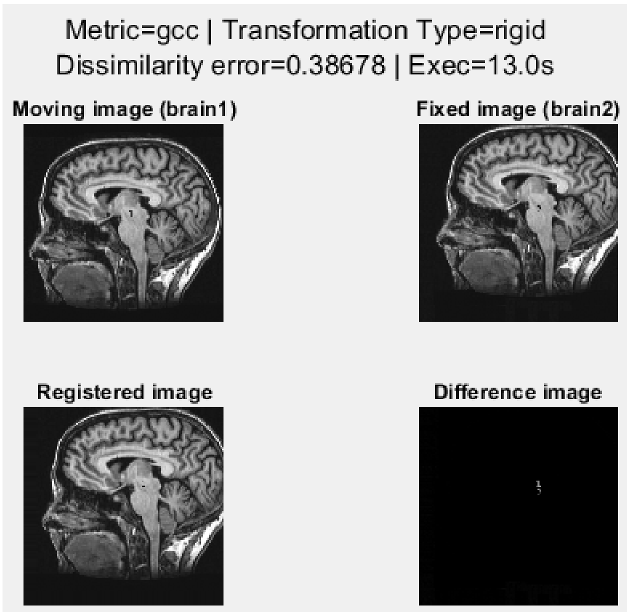

Figure 5: Brain 1 and Brain 2 registration with multi-resolution and NGCC.

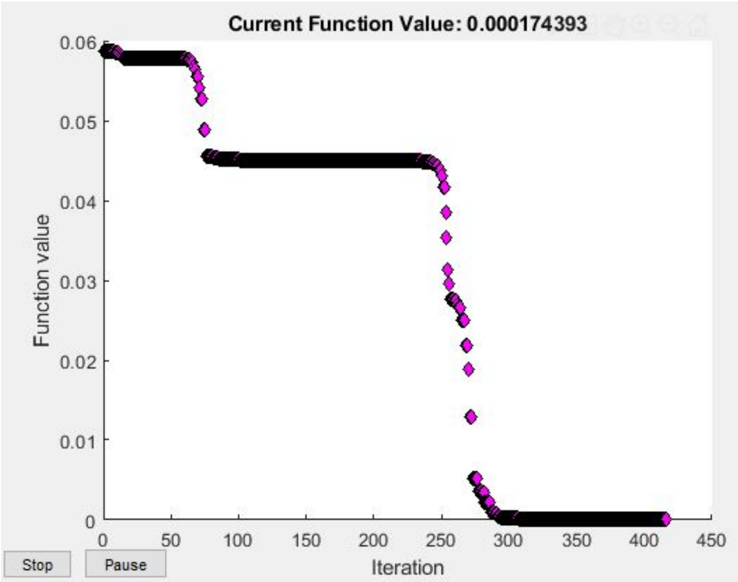

Figure 6: Plot of function value at each iteration when registering Brain 1 and Brain 2 at original scale without multi-resolution approach.

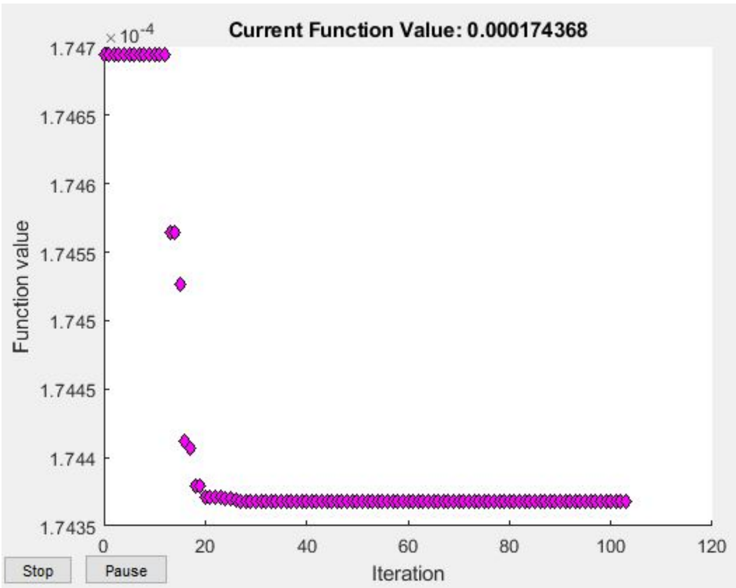

Figure 7: Plot of function value at each iteration when registering Brain 1 and Brain 2 at original scale with multi-resolution approach (i.e. at last layer of the pyramid).

4.2 Brain 1 and Brain 3 registration

Brain 1 image is larger than Brain 3 image, thus, it is necessary to use an affine transform to cater for the scaling factor. Another alternative was to increase the weight of the rotation parameter in the transformation matrix by modifying the scale vector. Figure 8 displays the result obtained with the NCC metric which gave a slightly better output than using the SSD metric. The optimal registration result was obtained when using the multi-resolution paradigm (a 5 levels pyramid was used in our experiment) which significantly reduced the total computational time (Figure 9). The registration with the NGCC metric yielded the same result but at a much longer time (36 sec in comparison to about 20 sec when using NCC).

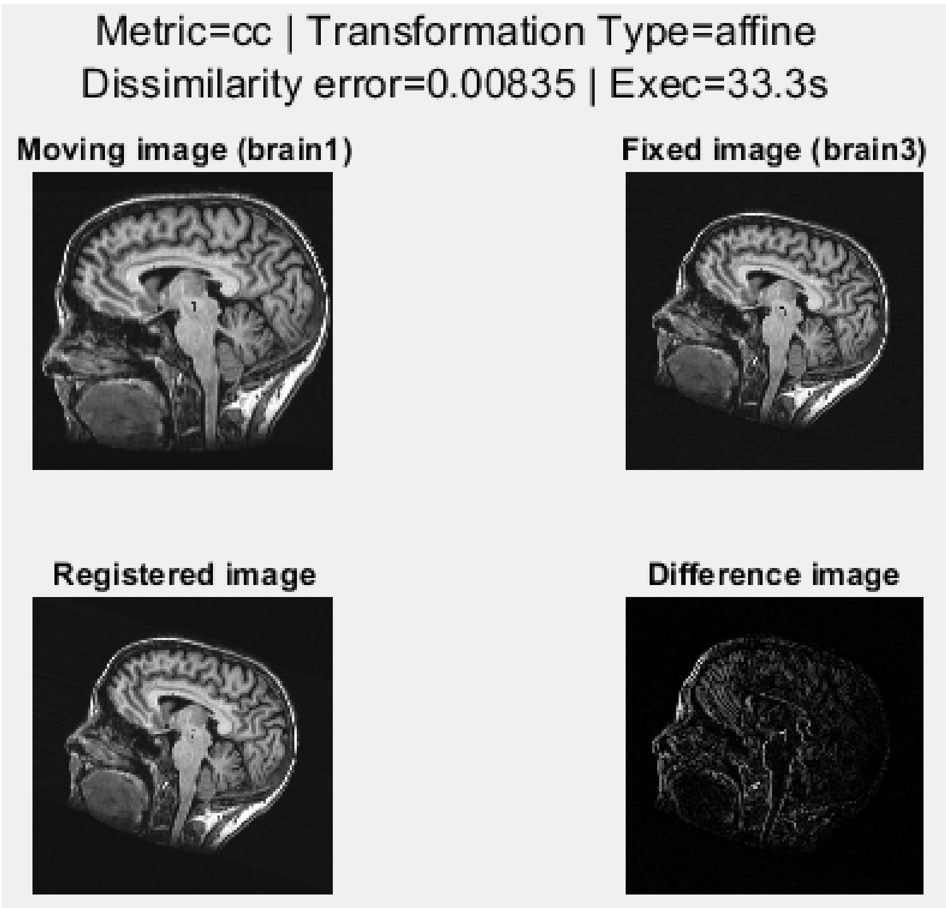

Figure 8: Brain 1 and Brain 3 registration with affine and CC

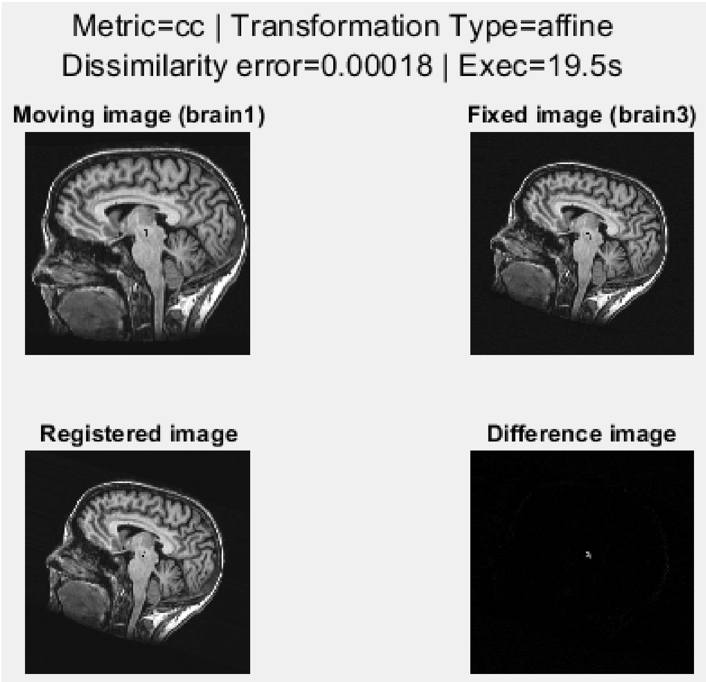

Figure 9: Brain 1 and Brain 3 registration with affine and NGCC (multi-resolution)

4.3 Brain 1 and Brain 4 registration

Brain 4 image is approximately Brain3 image with intensities inverted. Since the 2 images to be registered, i.e. Brain 1 and Brain 4, had different intensities, the SSD metric does not help in achieving the registration. The NCC and NGCC worked in this situation and gave similar results as when registering Brain 1 with Brain 3 provided a multi-resolution approach is used (at least 5 levels is needed if using the NGCC metric). However, the registration was much faster with NCC (Figure 10) than with NGCC (Figure 11).

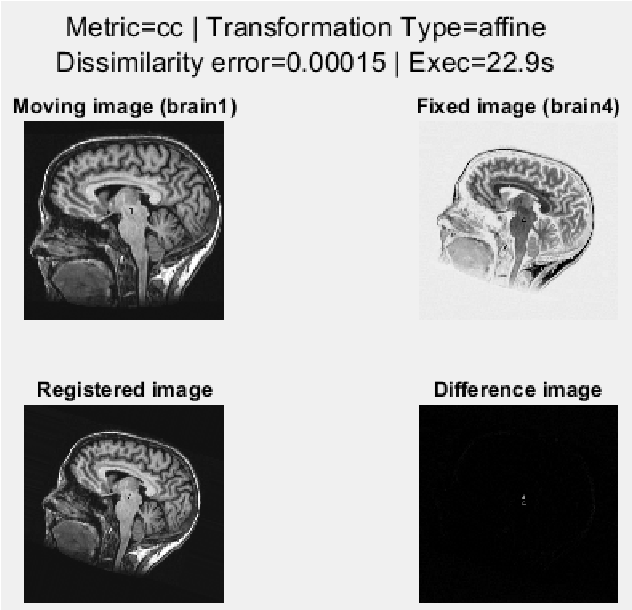

Figure 10: Brain 1 and Brain 4 registration with multi-resolution and NCC

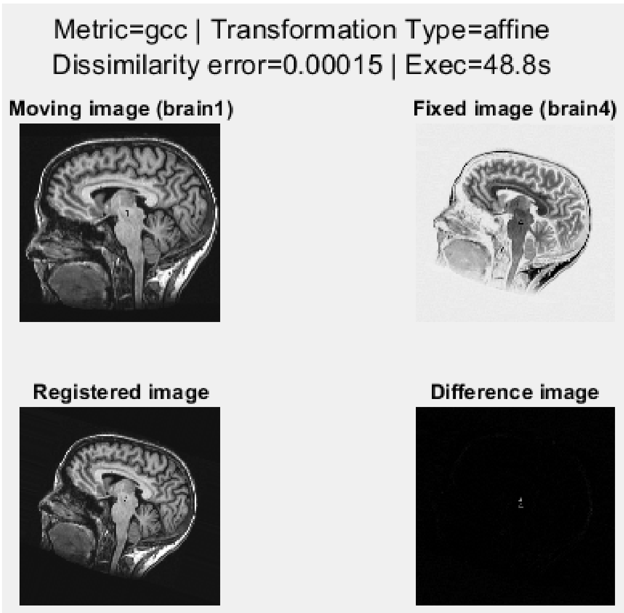

Figure 11: Brain 1 and Brain 4 registration with multi-resolution and NGCC

4.4 Brain 2 and Brain 3 registration

To align Brain 2 image to Brain 3 image, we can be sure by now that an affine transform and a multi-resolution approach will give optimal results. Also, since the moving image (Brain 2) content touches the border, we used the bilinear interpolation method with boundary pixels set to zero to avoid non-zero pixels from creating shadows. Figure 12 shows the registration result with 5-layer multi-resolution with NCC similarity metric. Since Brain 2 has some part of the head missing, it is possible to use Brain 3 image to add the missing part after Brain 2 image has been aligned to it.

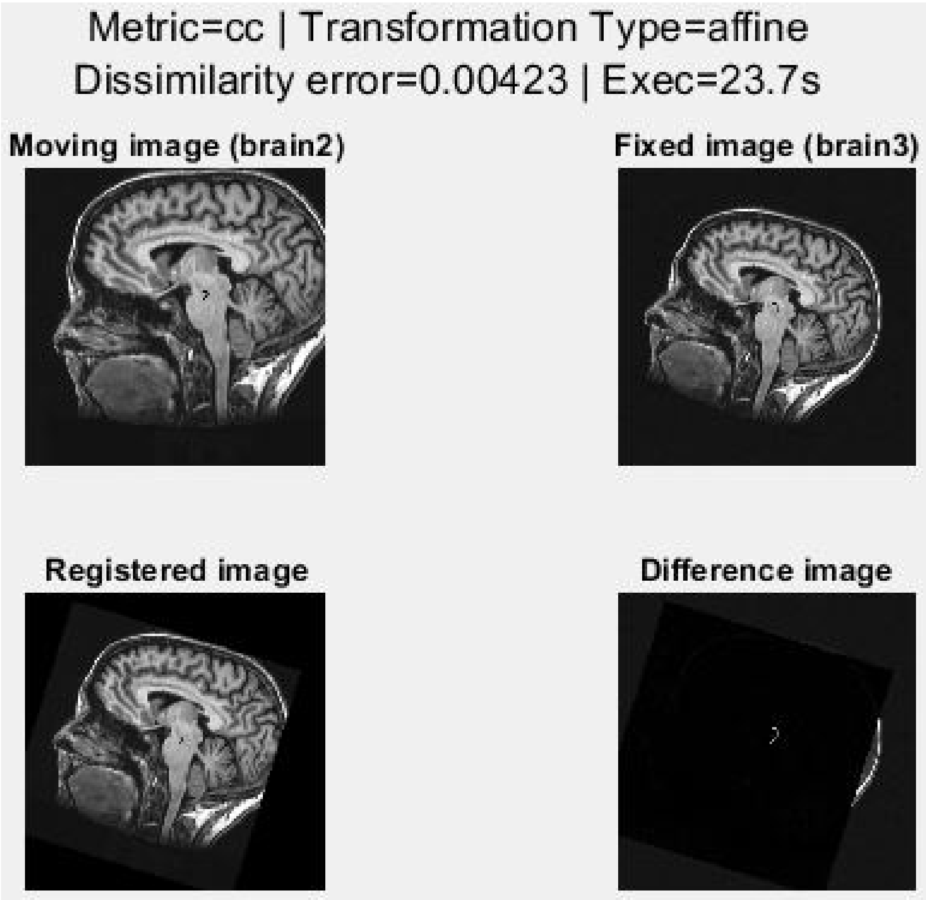

Figure 12: Brain 2 and Brain 3 registration with multi-resolution and NCC

5 Conclusion

5.1 Personal Reflections

This lab assignment helped in building a clear understanding of the basics of image registration. We can see the role of each component of the registration framework and its impact on the end result. The decomposition into distinct parts with each addressing a separate concern gives more insight and an easier grasp of the process flow. The complexity of the work was fair as we were provided with initial coding materials and instructions from the supervisor in addition to our constant follow-ups in matters regarding: the initialization of parameters, the scaling vector, the adaptation of the cross-correlation metric to work with a minimum-seeking optimizer (adding the minus sign), etc. We also realized the robustness of the multi-resolution approach to solve complex problems by reducing it to simple problems and iteratively building up to solve the problem. Finally, having to use Matlab environment made understanding and modification of the original code quite easy.

5.2 Final Remark

Image registration is an important procedure to compare and integrate images obtained at different times. In this exercise, we have studied the various effects of some registration framework. We investigated the impact of similarity metrics such as the SSD, NCC and NGCC on the various brain images given. These similarities metrics coupled with different transforms such as the rigid and affine produced different outputs. For some combinations, registration was well aligned but the computational time were longer and vice versa. To deal with the long computational time especially in the case with affine transforms, we introduced the multi-resolution registration concept in our implementation which improved the registration results at faster computational time.

References

[1] Luis Ibáñez, Will Schroeder, Lydia Ng, and Josh Cates. The ITK software guide. http://www.itk.org. Accessed: 2018-10-20.

[2] Terry S. Yoo. *Insight into images: principles and practice for segmentation, registration, and image analysis*. A K Peters, 2004.

[3] Feng Zhao, Qingming Huang, and Wen Gao. Image matching by normalized cross-correlation. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 2, pages II–II. IEEE, 2006.

[4] O Robert and Tetsuji Haga. Robust change-detection by normalised gradient-correlation. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.