BRAINSDemonWarp: An Applicaton to Perform Demons Registration

Release 1.00

Hans Johnson, Yongqiang Zhao

January 20, 2009

University of Iowa

Abstract

In this paper, a software application is introduced for image registration. BrainDemonsWarp is a command line program using Thirion and diffeomorphic demons algorithms. The program takes in a template image and a target image along with other optional parameters and registers the template image onto the target image. The resultant deformation fields and metric values can be written to a file. The program uses the Insight Toolkit (www.ITK.org) for all the computations, and can operate on any of the image types supported by that library.

Contents

1 Introduction

Thirion developed a diffusing model of registration based on the concept of demons that was introduced in the 19th century by Maxwell to illustrate a paradox of thermodynamics [1, 3]. In ITK implementation of demons registration [2], a whole image grid and a free form transformation constrained by a Gaussian filter of the displacement vector field were used to control the deformability of the model. The expression of the demons force is given by Optical flow equation [4]. By plugging the Newton method tools for Lie groups within the alternate optimization framework of the demons, T. Vercauteren *et al.* developed an efficient non-parametric diffeomorphic image registration algorithm based on an extension of the demons algorithm [5]. The second chapter in this document gives a short description of the program. The user's guide is given in the chapter 3. Examples and test cases for parameters selection were described at Chapter 5.

2 Description

The inputs to the BrainDemonsWarp program are the target image, the template image and the optional parameters. These parameters define the arguments for histogram matching and multi resolution registration. The outputs are the deformation field, output image, checkerboard image of the output and the fixed image and the x,y,z displacement vectors. If we specify debug option we can get the outputs at different stages . The filter is templated over the input image type, real image type and the output image types. We implement the algorithm by parsing the input, preprocessing them and registering the processed images.

(1) **Parsing -** The images are initialized by the ValidationInputParser. This function reads in the arguments from the parameter file. It sets the histogram bins, match points, number of levels in the multi resolution registration, shrink factors and number of iterations at each levels. If the orientations of the images are different it sets the orientation of the moving image to that of the fixed image.

(2) **PreProcessing -** In the next step the DemonsPreProcessor preprocesses the images by resampling the template image to target image space. The intensity mismatch problem is solved by histogram matching the images. Histogram matching is done only if the command line option -e is set. ItkHistogramMatchingImageFilter is used to perform this function. Another important step in preprocessing is skull stripping. Skull stripping is done only if the command line option -bobf is set. We have written an itk filter, named itkBOBFFilter for this purpose. This filter takes in an input image and a whole brain mask and outputs a Brain Only Background Filled(BOBF) image. The non-brain parts in the image are filled with the user specified background value. All computations are performed in the precision of float data.

(3) **Registration -** The resulting moving Image and the fixed image are given as inputs to the demons registrator.It uses the MultiResolutionPDEDeformableRegistration2(Using NN extrapolation as interpolator) filter and implements the demons deformable algorithm by computing the deformation field which will map a moving image onto a fixed image. It is assumed that the vector elements behave like floating point scalars. Each vector in the deformation field represent the distance between a geometric point in the input space and a point in the output space. The output image is generated by warping the input image with the deformation field using the ItkWarpImageFilter. WarpImageFilter warps an existing image with respect to a given deformation field. Typically the mapped position does not correspond to an integer pixel position in the input image. Interpolation via an image function is used to compute values at non-integer positions. We have used the LinearInterpolateImageFunction for our application. To write the output image we cast the image to the user specified output pixel type.

3 User's Guide

This section outlines usage of BrainDemonsWarpCLP command-line tool. The goal of this application is to perform an intra-modality deformable registration with a chosen variant of the demons. If a target image is specified then the input image is registered onto the target image to produce an output image. Along with the target image other parameters need to be given by the user giving information for registration. This program can be run using the command
BrainDemonsWarpCLP <Arguments>
Arguments:
Required arguments are

- **movingVolume-** The input image file name is given with **-m or –movingVolume** option.
- **outputVolume-** The output image file name is set by **-o or –outputVolume** option.
- **fixedVolume-** The target image file name is given with **-f or –fixedVolume** option.

The optional arguments are

- **registrationFilterType-** The registration filter type(Demons—FastSymmetricForces—Diffeomorphic) is given with **–registrationFilterType** option. Default Demons
- **medianFilterSize-** X Neighborhood to be included for BOBF. Default 0,0,0
- **maxStepLenght-** Maximum length of an update vector(0: no restriction) with **-l** option. Default 2
- **numberOfLevels-** Number of multiresolution levels with **-n** option. Default 3
- **numberOfIterations-** Number of demons iterations per level with **-i** option. Default 10,10,10
- **smoothDefomationFieldSigma-** Smoothing sigma for the deformation field at each iteration with **-s** option. Default 0.0001
- **smoothingUp-** Smoothing sigma for the update field at each iteration with **-g** option. Default 0
- **normalization-** Use histogram matching with **-e** option.
- **numberOf HistogramLevels-** Number of histogram bins to use in the histogram equalization. Default 256
- **numberOfMatchPoints-** Number of match points to use in the histogram matching algorithm. Default 2
- **minimumFixedPyramid-** The shrink factor for the first level of the fixed image pyramid. Default 4,4,4
- **minimumMovingPyramid-** The shrink factor for the first level of the moving image pyramid. Default 4,4,4
- **inputPixelType-**The input image pixel type is given with **–inputPixelType** option. Th input image is read using this option. Valid Input data types are: UCHAR SHORT USHORT INT UINT FLOAT DOUBLE. Default float

- **outputPixelType-** Output image type. Image is written to the file in this type. Default is float.
- **outputCheckerboardVolume-** If this option is set the checker board image comparing the deformed image with the target image is written
- **checkerboardPatternSubdivisions-** This option sets the checker board image pattern. Enter a non zero number for each dimension. Default is 4 in all dimensions.
- **outputDisplacementFieldPrefix-** You can write the displacement fields by giving the prefix of the displacement fields, (suffix of xdisp.hdr, ydisp.hdr, zdisp.hdr will be added).
- **outputDeformationFieldVolume-** Write the deformation fields with **-O** option
- **outputDebug-** This option is to write all the intermediate images, debug information and metric value file for debugging purposes with **-v** option.
- **makeBOBF-** This option is to make Brain Only Background Filled(BOBF) images.
- **movingBinaryVolume-** Target Mask name to perform BOBF
- **fixedBinaryVolume-** Template Mask name to perform BOBF
- **lowerThreasholdForBOBF-** Lower Threshold for performing BOBF. Default 0
- **upperThresholdForBOBF-** Upper Threshold for performing BOBF. Default 70
- **backgroundFilleValue-** Backgond fill with this value Default 70
- **seedForBOBF-** Seed (X,Y,Z) for BOBF. Default 0,0,0
- **neighborhoodForBOBF-** Neighborhood in all 3 directions to be included when performing BOBF. Default 1,1,1
- **movingLandmarks-** Landmarks filename for moving image to initialize deformation field.
- **fixedLandmarks-** Landmarks filename for fixed image to initialize deformation field.
- **initialDeformationFieldVolume-** Initial deformation field vector image file name.
- **initializeWithFourier-** Initial Coefficient filename.
- **initializeWithTransform-** Initial Transform filename
- **forceCoronalZeroOrigin-** Flag to indicate that all images being read in must have their spatial centers matched.
- **outputNormalized-** Flag to warp and write the normalized images to output. In normalized images, the image values are fit-scaled to be between 0 and 1.

4 Evaluation

To evaluate the performance of this registration results, the proper metric is necessary. Different medical image types or different application results may require different parameters of registration. These parameters are very important to achieving the corresponding requirements for any given application. Evaluation of the registration metrics can help to choose an optimal set of parameters. This application can write the file that provides three cost functions for each iteration:

- **MSE**: compute the mean square difference in intensity between the fixed image and transforming moving image.
- **Harmonic energy**: compute the harmonic energy of the deformation field.
- **Jacobians**: compute a scalar image from the deformation field, where each output scalar at each pixel is the Jacobian determinant of the vector field at the location.

Harmonic energy and the minimum and maximum values of determinant of the Jacobian of the transformations show how smooth spatial transformation is.

5 Parameter selection

The selection of the parameters is crucial in deformable registration. The criteria for parameter selection depends on the applications. Evaluation of the registration metrics can help to choose an optimal set of parameters. Here, we give two cases for different parameters selection as examples.

5.1 Iteration

Case 1: Fixed Image: 256*256*256 MRI image (Philips)

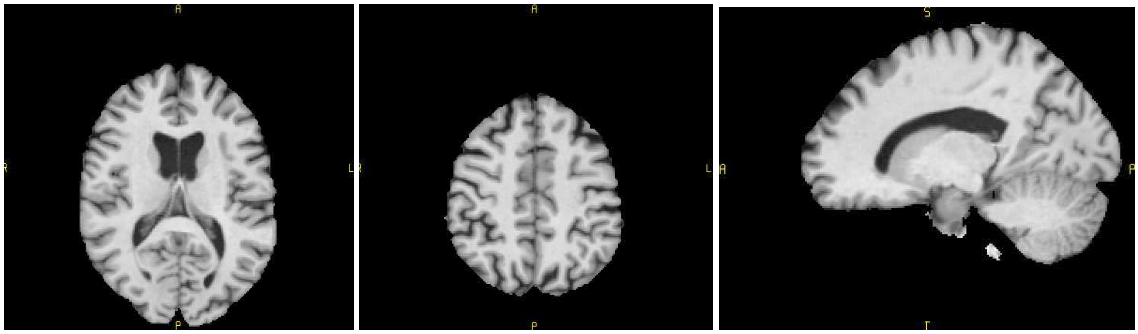
Moving Image: 256*256*256 MRI image (Siemens)

BRAINSDemonWarpCLP -f fixedimage.nii.gz -m movingimage.nii.gz -o outputimage.nii.gz -O outputdeformationfield.nii.gz -n 5 -s 3.0 -e –numberOfHistogramBins 1024 –numberOfMatchPoints 7 -i 1000, 400,300,200,100 -v –registrationFilterType Diffeomorphic

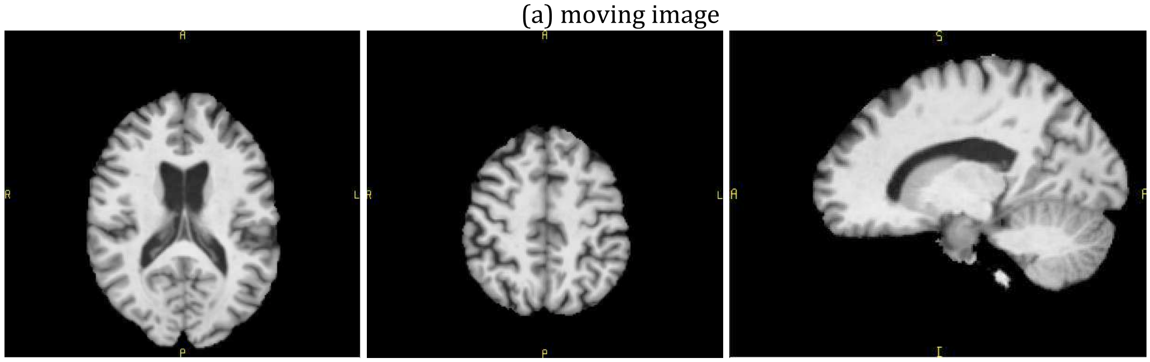
The following images show how well the registrations are doing. Here, three typical layers were selected to compare the registration process at each level(See Fig. 1). For better visualization, subtractions of fixed image from the results at each level are shown at Fig. 2. Fig. 3 showed the plots of these cost functions vs iteration case 1. From Table 1, case 1 spent more than 3 hours for registration. Especially, the fifth level has the lest iterations but with most computation cost.

This high computational requirements would make this method not much attractive for practical applications. More computational cost provides better registration results. In contrast, low cost will bring the poor deformation. We wanted o find a tradeoff that provided the best efficiency and the best quality of regitration results.

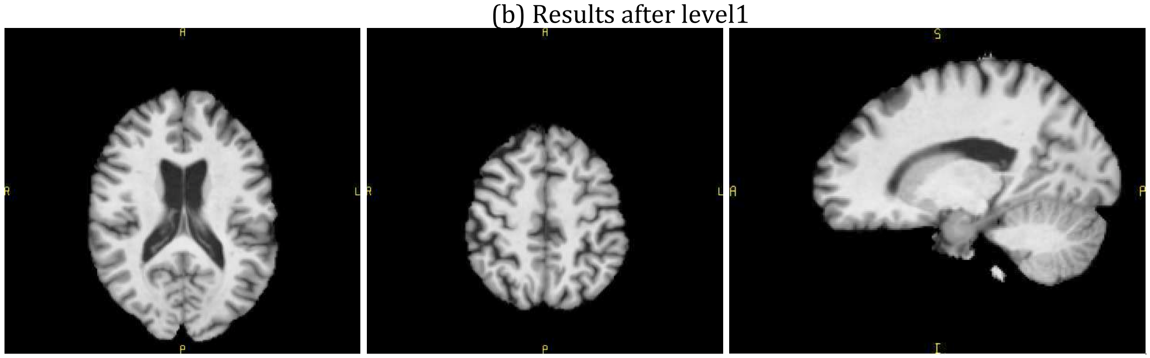
MSE can be used to quantify the parameters while deformation was measured using Harmonic energy and Jacobin. For example, Fig. 4 showed the derivative of MSE vs iterations for each level of the Case1.



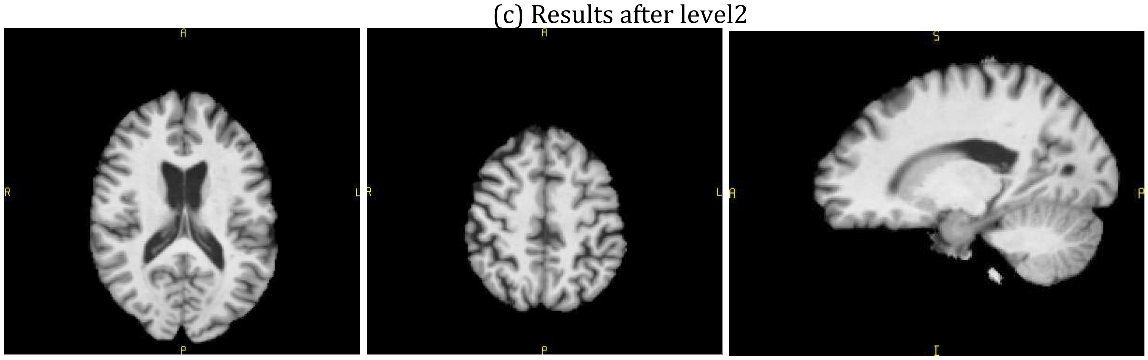
(a) moving image



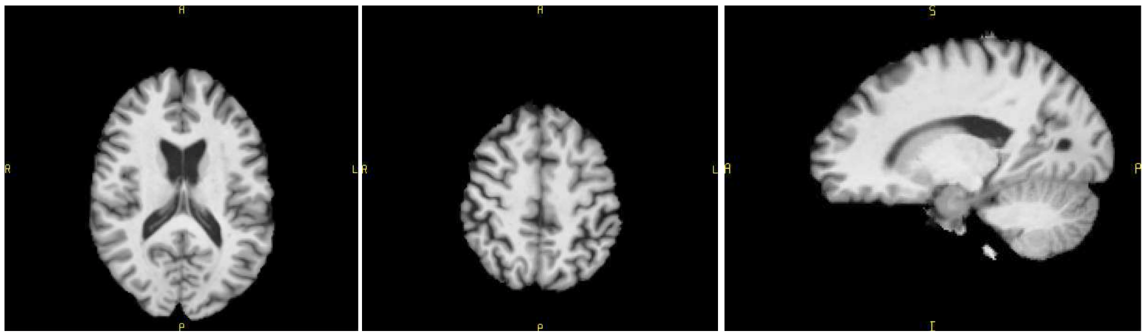
(b) Results after level1



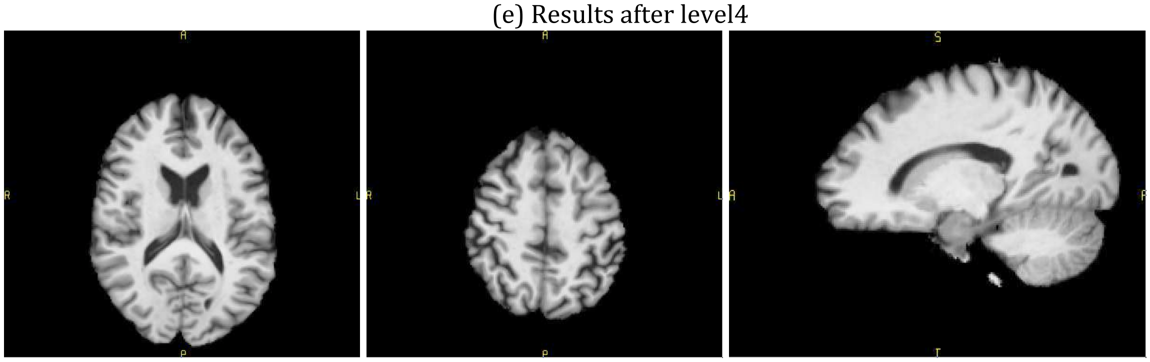
(c) Results after level2



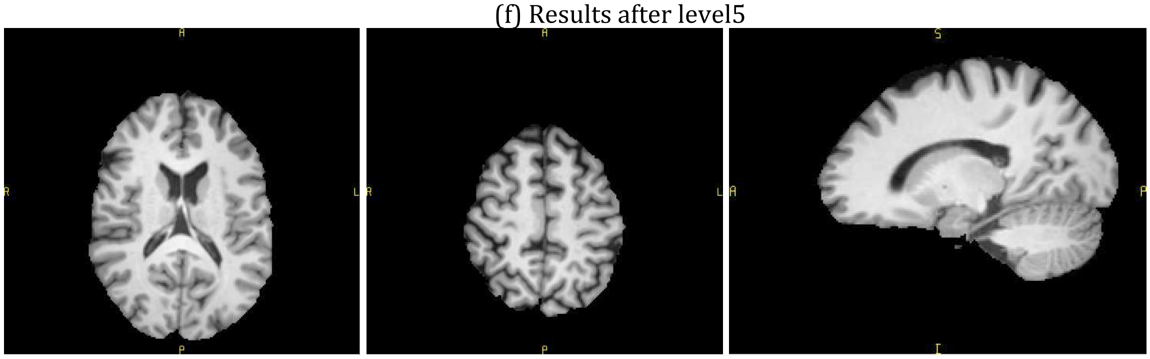
(d) Results after level3



(e) Results after level4

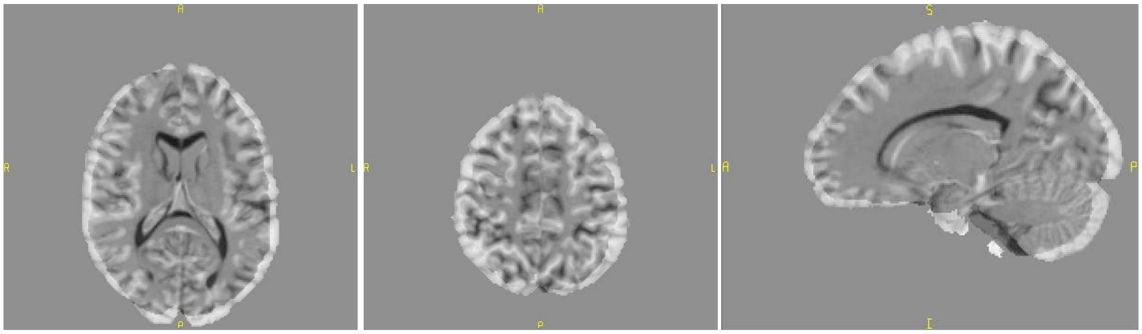


(f) Results after level5

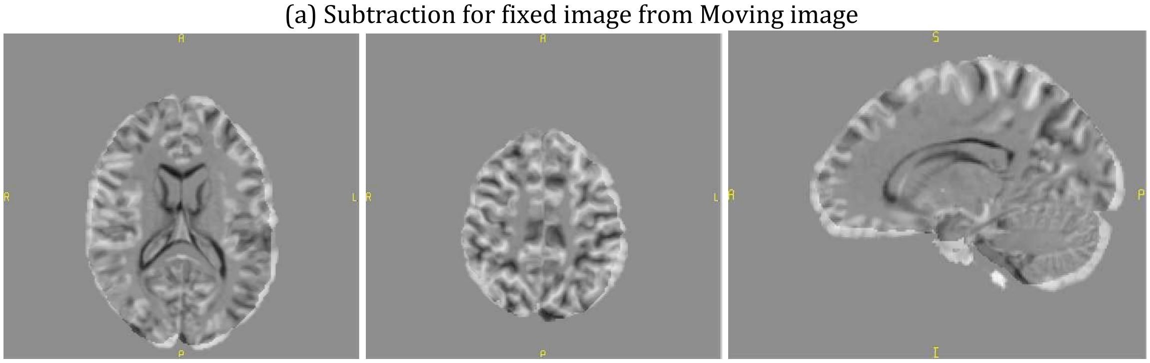


(g) Fixed image

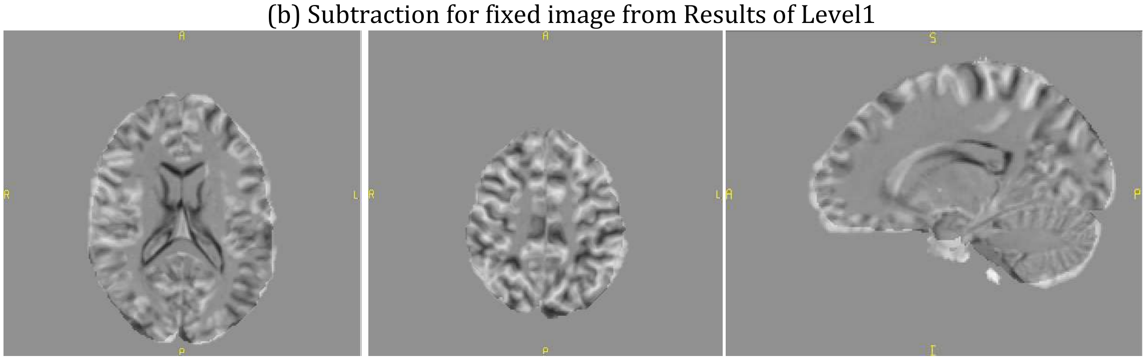
Figure 1: From left to right, the 143rd(Axial), 173rd (Axial) and 142nd (Sagittal) layers are selected to display



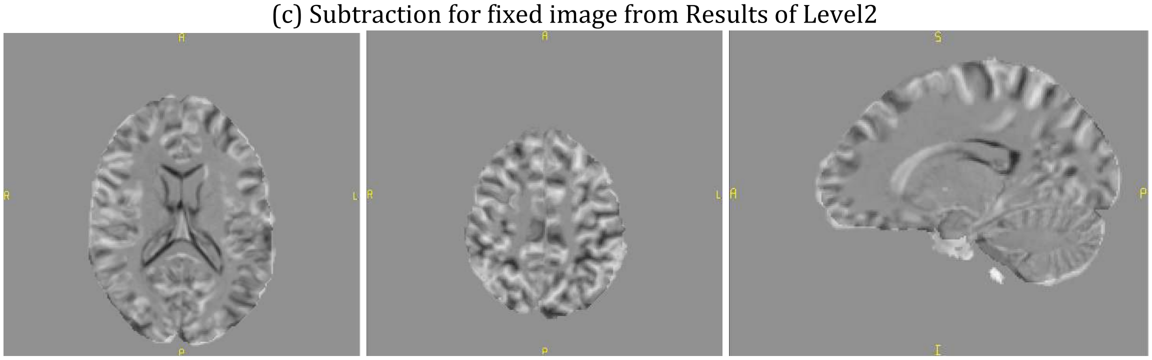
(a) Subtraction for fixed image from Moving image



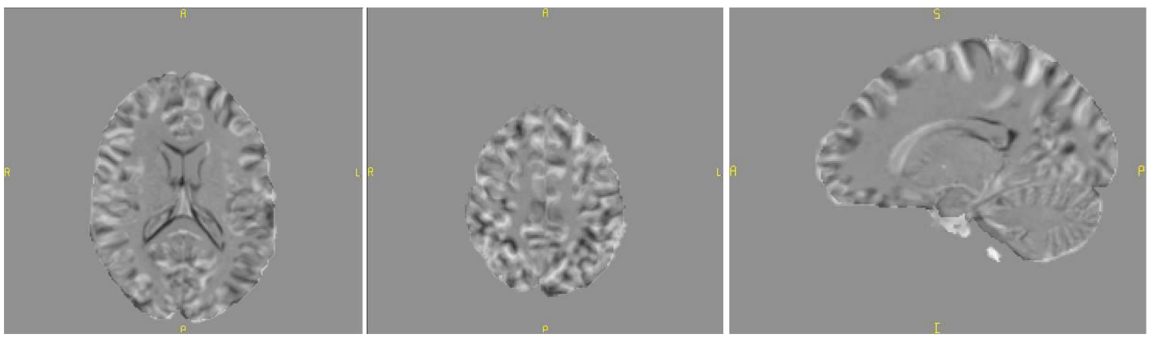
(b) Subtraction for fixed image from Results of Level1



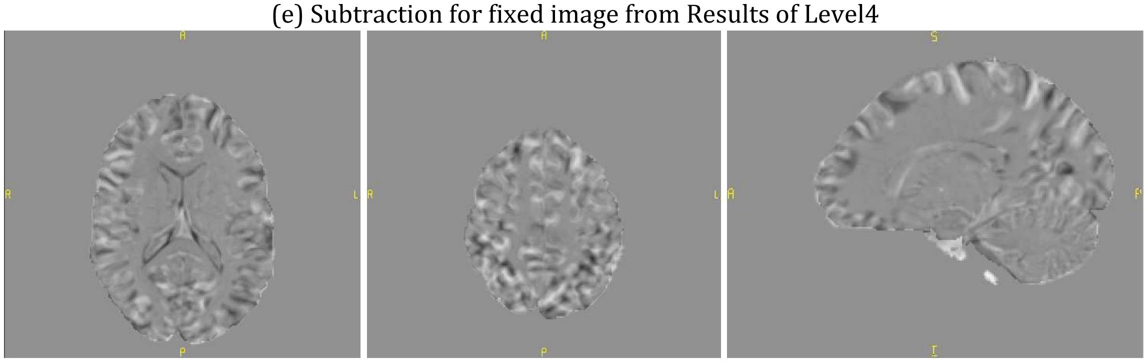
(c) Subtraction for fixed image from Results of Level2



(d) Subtraction for fixed image from Results of Level3



(e) Subtraction for fixed image from Results of Level4



(f) Subtraction for fixed image from Results of Level5

Figure 2: Subtraction results of fixed image from Results of each level From left to right, the 143rd(Axial), 173rd (Axial) and 142nd (Sagittal) layers are selected to display

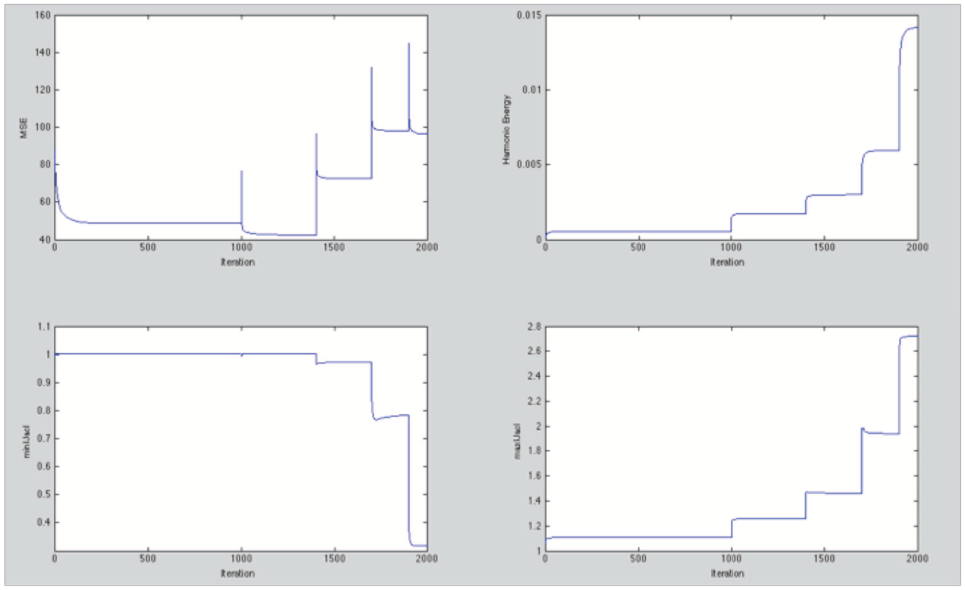


Figure 3: **different cost functions vs iterations**

The derivative is said to be convergent if its value approaches 0. It is obviously that 400, 100, 30, 20, 15 are optimal iterations for each level for this registration. We re-did the registration for Case 1 using these optimal iteration. It also spent 17 minutes. The result was very close but the computation cost was only one tenth.Fig. 5 showed the comparison of the different cost functions at each level of the previous results and optimal iterations. There are very small difference of the cost functions between these two registration.

Table 1: Running Time

	Iterations	Case 1
Level 1	1000	Less then 25s
Level 2	400	49s
Level 3	300	6m20s
Level 4	200	34m
Level 5	100	140m(2h20m)
Total	2000	187min(3h17m)

5.2 Smoothing sigma

Case 2: Fixed Image: 256*256*256 MRI image

Moving Image: 256*256*256 MRI image

BRAINSDemonWarpCLP -f fixedimage.nii.gz -m movingimage.nii.gz -o outputimage.nii.gz -O outputdeformationfield.nii.gz -n 5 -s (0.5 3.0) -e –numberOfHistogramBins 1024 –numberOfMatchPoints 7 -i 200,100,30,20,15 -v –registrationFilterType Diffeomorphic

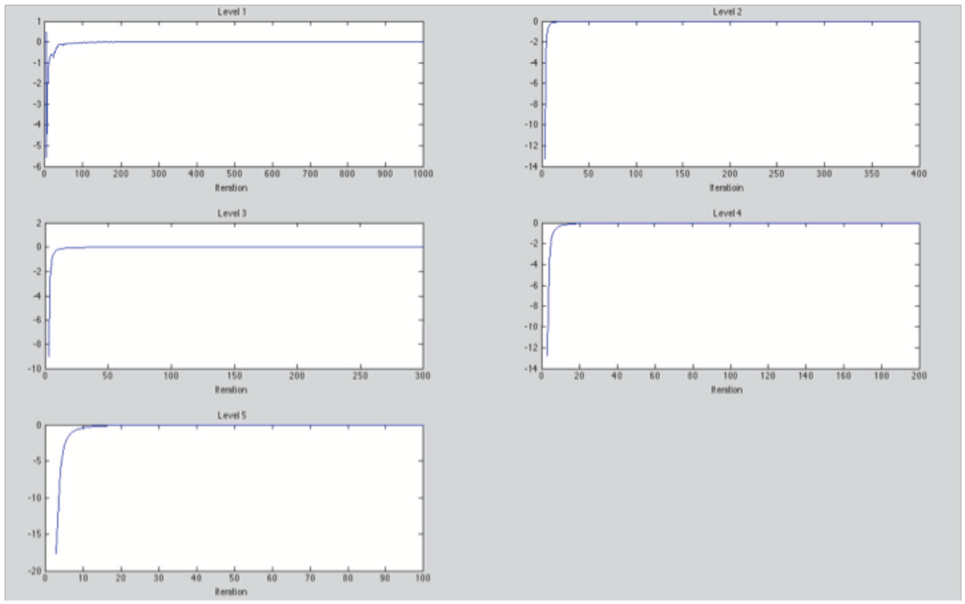


Figure 4: **the derivative of MSE vs iterations**

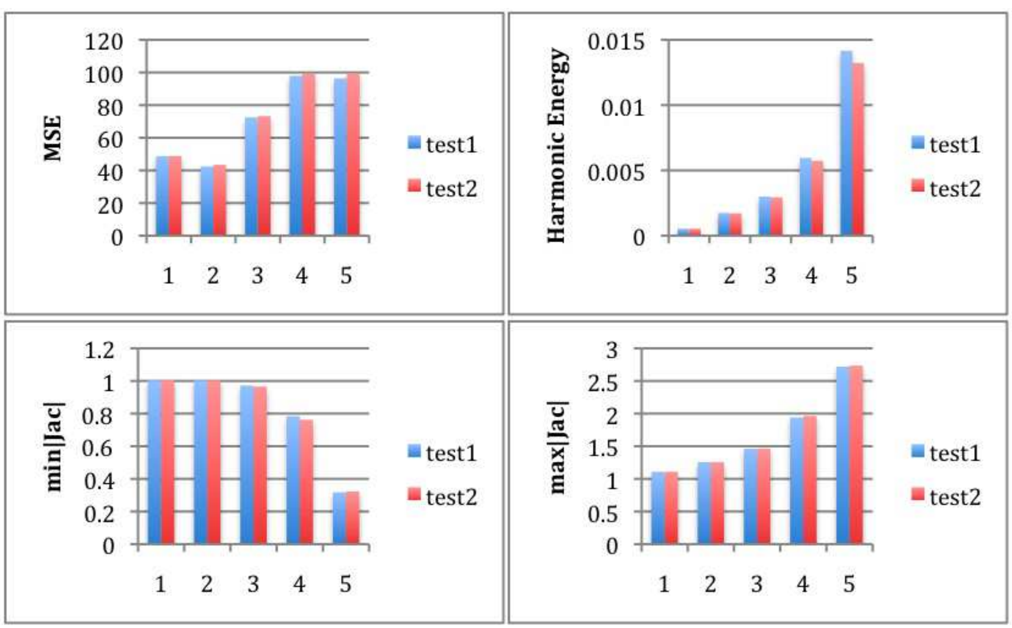


Figure 5: **Charts for cost functions comparison Test1: 1000,400,300,200,100 Test2: 300,50,30,20,15**

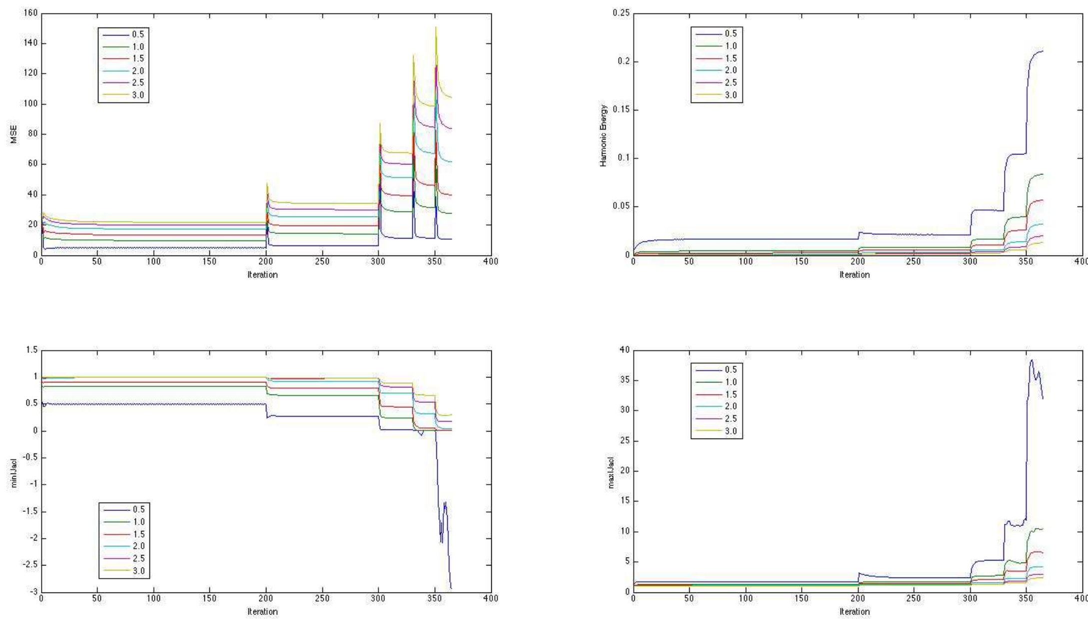


Figure 6: **cost functions obtained from different sigma vs iterations**

This is another important parameters for image registration. Strong smoothness for deformation field hinders large deformation. In contrast, lower values of sigma, the higher frequency components are conserved on deformation field. In this example, we still wanted to find a tradeoff between the intensity matching and deformation. Here, cost function and registration results are both to use for choosing the proper smoothing sigma. Fig. 6 showed the cost functions vs iterations according to different six sigma(from 0.5 to 3.0). From this image, when s is smaller, MSE is smaller.

As the harmonic energy of the deformation field is the measure inversely related to the smoothness of the deformation field, when s is bigger, energy is smaller. As for the determinant of Jacobian, smaller s, smaller minimum $|Jac|$, bigger maximum $|Jac|$. These three metrics showed that small sigma provided smoother spatial transformations. Furthermore, considering the registration results(Fig. 7 and 8), the difference were highlighted with the red circle. This may help one to see in the registration quality. When sigma is smaller, ventricles structures are close to the fixed image. However, gyrus is not satisfied. s=1.5 2.0 can be the optimal choice based on your requirement. In addition, more computation time was used when sigma is bigger.

6 Verification

Once these optimal parameters were obtained, large data set can be used to verify the effectiveness and reliability of this application. Here, 361 pair registrations(fixed image remained same)were executed with the optimal parameters. Fig. 9 shows three views of the fixed MRI brain image, average moving image and average registration results(both Thirion demons and diffeomorphic demons). It is obvious that the results of diffeomorphic demons showed more detail and clear structures compared to that of Thirions. However,

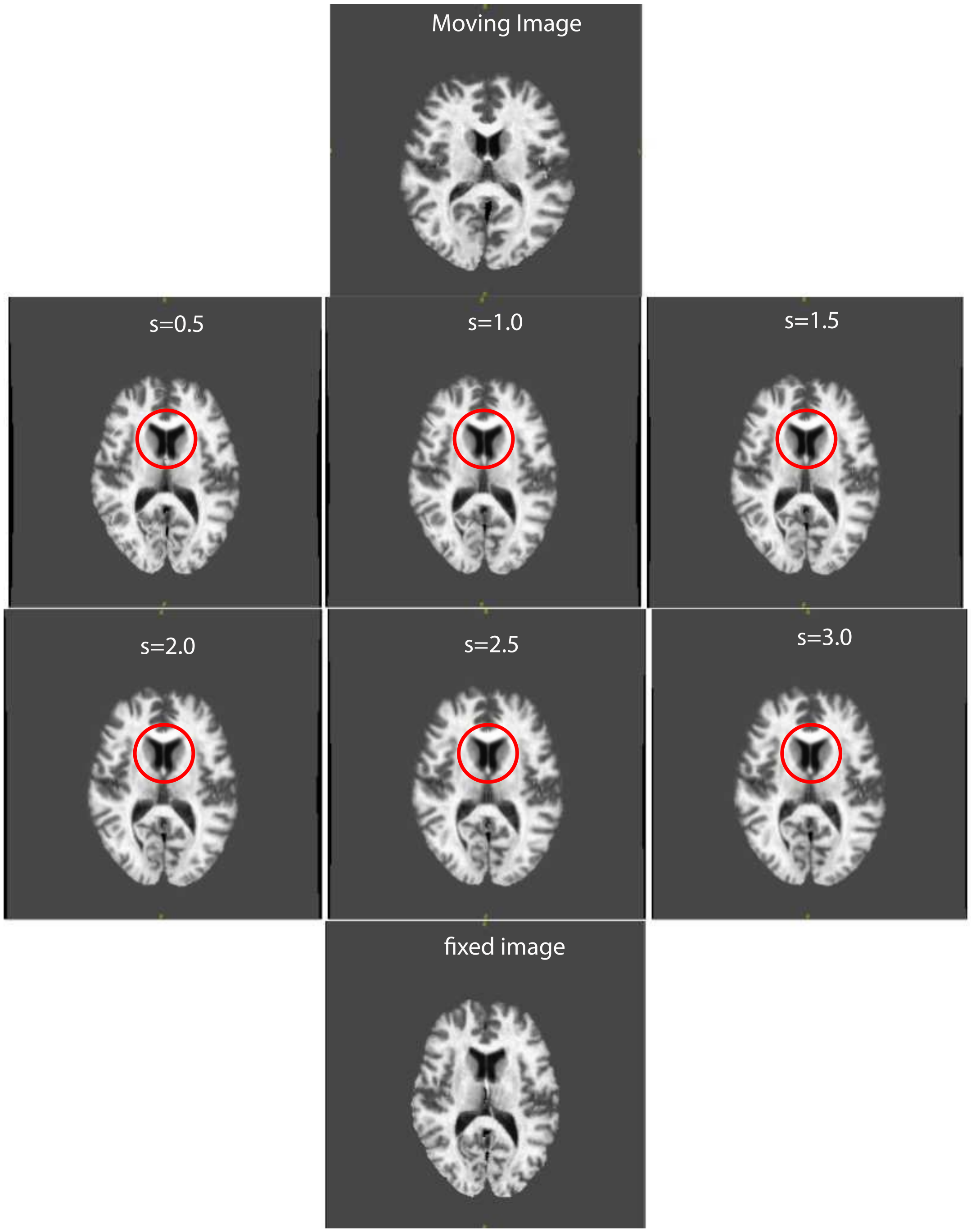


Figure 7: **Registration results of different smoothing sigma: the 143rd(Axial)**

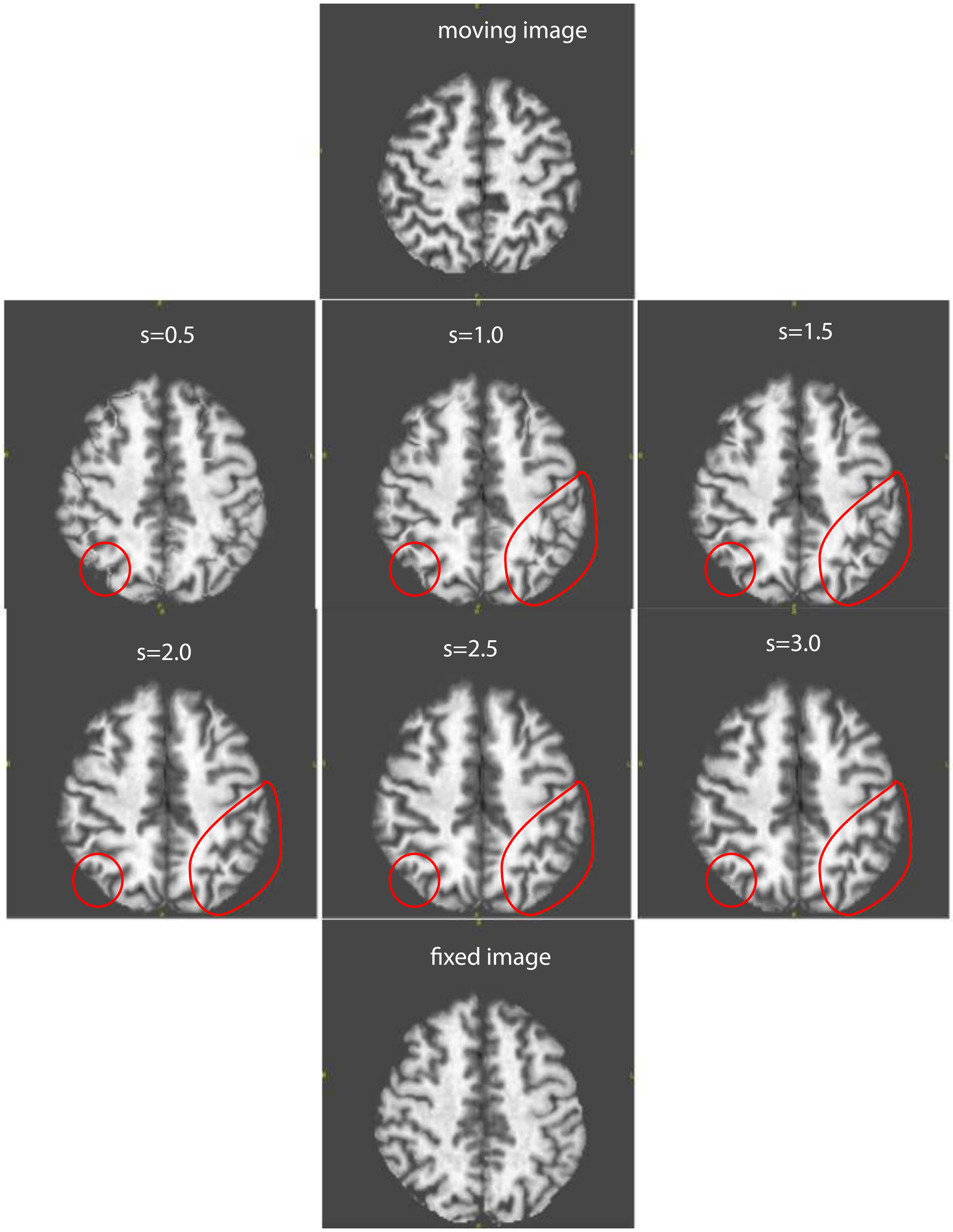


Figure 8: **Registration results of different smoothing sigma: the 173rd(Axial)**

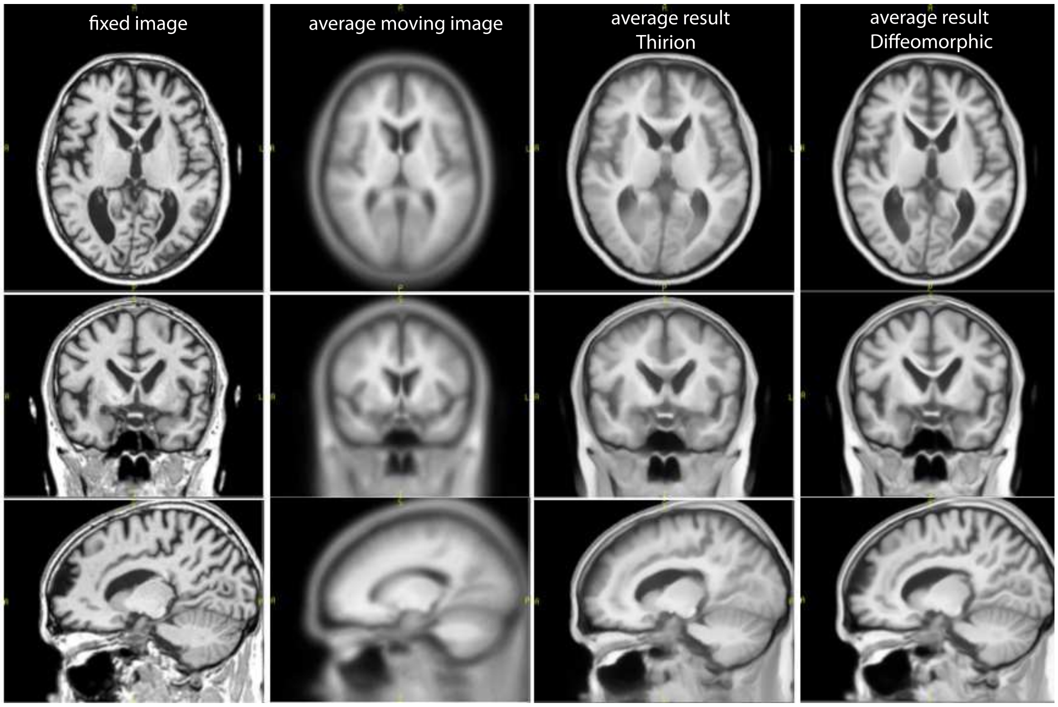


Figure 9: **Average source images and registration results of 361 pair registrations**

each registration spend 12 minutes that meant the computation cost increased 50% .

Acknowledgement

This work was supported by PREDICT, CHDI, and BRAINS Grant (R01 NS050568).

Appendix

Download site: http://www.nitrc.org/projects/brainsdemonwarp/

Dashboard: http://testing.psychiatry.uiowa.edu/CDash/index.php?project=BRAINSDemonWarp

References

[1] J.-P. Thirion, *Image matching as a diffusion process: an analogy with Maxwell's demons*. Medical Image Analysis 2(3):243-260, 1998. 1

[2] Lydia Ng, *Validation study: Atlas-based brain volume segmentation from MRI images*. Technical report, Insightful Corporation, 2002. 1

[3] J.-P. Thirion, *Fast non-rigid matching of 3D medical images*. INRIA, Tech. Rep. 2547, May 1995. 1

[4] B. K. P. Horn and B. G. Schunck, *Determining optical flow*. Artificial Intelligence, vol. 17, pp 185-203, 1981. 1

[5] T. Vercauteren, X. Pennec, A. Perchant and N. Ayache, *Non-parametric diffeomorphic image registration with the demons algorithm*. MICCAI'07. 1