Image Registration Techniques
Homework 5 - Ibáñez
Due: Monday February 17 at 9:00pm EST
Submission: via email

This homework exercises the concepts presented in Lectures 8 and 9.

You are required to write code using the Insight Toolkit and run the corresponding executable using specific images as input. The material that you should return is composed of your source code and the output images generated from the execution of your program. Include your CMakeLists.txt files along with the source code.

This material should be sent electronically to the email address:

luis.ibanez@ieee.org

1. (**10 points**) Image Registration Exercise.

   You will register the two images:

   - Insight/Examples/Data/BrainProtonDensitySlice.png
   - Insight/Examples/Data/BrainProtonDensitySliceShifted13x17y.png

   Using the program:

   Insight/Examples/Registration/ImageRegistration3.cxx

   This program is described in Section 8.3 of the *ItkSoftwareGuide.pdf* document available in your ITK-CD. Copy the program in a new empty directory. Create a CMakeLists.txt file for it. Configure the project and build it. The program is designed to be executed in the command line from an MS-DOS windows, a Cygwin shell, or a UNIX shell.

   Modify the program in order to use different step lengths for the optimizer. Try steps : { 1.0, 2.0, 5.0, 10.0 and 20.0 } millimeters.

   Hint: look for line 384 in ImageRegistration3.cxx and modify the value 4.00 in:

   ```
   optimizer->SetMaximumStepLength( 4.00 );
   ```

   you may find convenient to take the value from the command line so you don't need to recompile the code for each different step length. Something

like:

```
optimizer->SetMaximumStepLength( atof( argv[4] ) );
```

Return with you homework the version of the program that registered the images correctly, along with the resampled moving image generated by this program.

2. (**10 points**) Metric Smoothing Exercise.

   Using the smoothing program that you wrote for homework 4 you will smooth both images in the previous exercise using a sigma value of 4.0. Then repeat the execution of the registration process of the previous exercise using the same set of step lengths for the optimizer and compare how many iterations it takes now to converge to the solution.

3. (**10 points**) Rigid Registration Exercise.

   You will register the two images:

   - Insight/Examples/Data/BrainProtonDensitySliceBorder20.png
   - Insight/Examples/Data/BrainProtonDensitySliceRotated10.png

   Using the program:

   Insight/Examples/Registration/ImageRegistration5.cxx

   This program is described in Section 8.5 of the *ItkSoftwareGuide.pdf* document available in your ITK-CD. Copy the program in a new empty directory. Create a CMakeLists.txt file for it. Configure the project and build it. The program is designed to be executed in the command line from an MS-DOS windows, a Cygwin shell, or a UNIX shell.

   You may have to tune the values of the optimizer Step Length and the translationScale.

   Hint: look for lines 310 and 333 in ImageRegistration5.cxx

4. (**20 points**) Metric Plotting Exercise.

   You will compute the value of an image metric for a set of different translations. As fixed and moving image use the following two images:

   - Insight/Examples/Data/BrainT1Slice.png
   - Insight/Examples/Data/BrainProtonDensitySlice.png

   Use a NearestNeighbor interpolator and the Mattes Mutual Information metric.

(a) Use a translation transform and plot the values of the metric for translations along X going from -20.0mm to +20.0mm every 1.0mm. Set the translation along Y = 0.0mm in all the cases. Report your results as a graph of Metric versus translation in X.

(b) Testing the effect of changing the interpolator. Plot the same metric now for translations along X from -5.0mm to 5.0mm every 0.1mm. (**NOTE that in this case you are performing sub-pixel displacements**. Generate the plots for the following three interpolators: Neares Neighbor, Linear, BSpline. Comment on your observations on the effect of the interpolator over the smoothess of the metric function.