3 Homework 3 (Image Registration, sections 9-10)

Inexact landmark matching: Image matching with translation

a.) We consider aligning a 2D template image *I* to a 2D target image *I′* using translation only. We want to find the vector **b** that minimizes the cost

$$C(b) = \frac{1}{2}\int |I(x-b) - I'(x)|^2 dx$$

Consider the perturbation $b \longrightarrow b + \epsilon h$ and write

$$J(\epsilon) = C(b + \epsilon h)$$

Work out

$$\frac{d}{d\epsilon} J(\epsilon)\Big|_{\epsilon=0}$$

b.) Write down the gradient of the cost with respect to *b*. Recall that $\nabla C(b) \cdot h = \frac{d}{d\epsilon} J(\epsilon)|_{\epsilon=0}$.

c.) Write down a gradient descent algorithm that can be used to minimize this cost for a given *I* and *I′*. It should take the form of

$$b_{new} = b_{old} - \epsilon(\text{gradient term})$$

for ϵ for a small step.

d.) Write a program in matlab that will implement this algorithm. The inputs should be the template image *I*, the target image *I′* (both 2D arrays), the gradient descent step size ϵ, and the number of iterations of gradient descent.

The program should output the optimal vector *b*, and the transformed image *I(x-b)*.
Note that you can use MATLAB's built in function `gradient` to compute the gradient of the image.
One method that you can use to apply the translation to *I* is:

```
[X, Y] = meshgrid(1:size(I,2),1:size(I,1));
I_translated_by_b = interp2(I, X-b(1),Y-b(2), 'linear',0);
```

You should print the cost *C* at each iteration of gradient descent to make sure it is decreasing. If it is not decreasing, choose a smaller step size.

You may use the given code `splineImage.m` as a model for your program. You will notice a lot of similarities, but your program will be simpler. For example, you will not need the function `applyPowerOfA` or input arguments `alpha` or `sigma`. Remember that your gradient will be a vector with 2 components. It will not be a function at each point in space like in `splineImage.m` e.) Use your code to match the corpus callosum 0001_CC_Con.png to 0003_CC_Alz.png with translation. You can view the whole MRI these structures are segmented from the files 0001_MRI_Con.png and 0003_CC_Alz.png if you wish.

You can load them like this:

```
I = double(imread('0001_CC_Con.png') > 0);
IPrime = double(imread('0001_CC_Alz.png') > 0);
```

This image should now take two values only: 0 and 1, and be represented in double precision, not with an 8-bit integer. You can see what these images look like in Fig. 1.

Make a figure showing the template image, the translated image, and the target image
Make a figure showing the template image minus the target image, and the translated template image minus the target image. Include a colorbar.

f.) **Image matching with splines**. Use the given code splineimage.m to match your translated corpus callosum to your target. If you were unable to complete the previous part of the homework then just use the original (untranslated) corpus callosum.

- Use the value $\sigma = 0.01$. This parameter controls how important matching accuracy is relative to the size of the deformation.

- Use the value $\alpha = 20$. This parameter controls how smooth the deformation is.

Report the step size you used and the number of iterations. Report the initial cost, and the final cost after your algorithm is finished. Again, if the cost is not decreasing, choose a smaller gradient descent step size.

Make a figure showing the deformed image. Make a figure showing the deformed image minus the target image. Include a colorbar.

g.) **Jacobian calculations**. Calculate the determinant of the Jacobian of the transformation $\phi(x) = x + v(x)$ at each point in space. Show this in a figure with a colorbar.

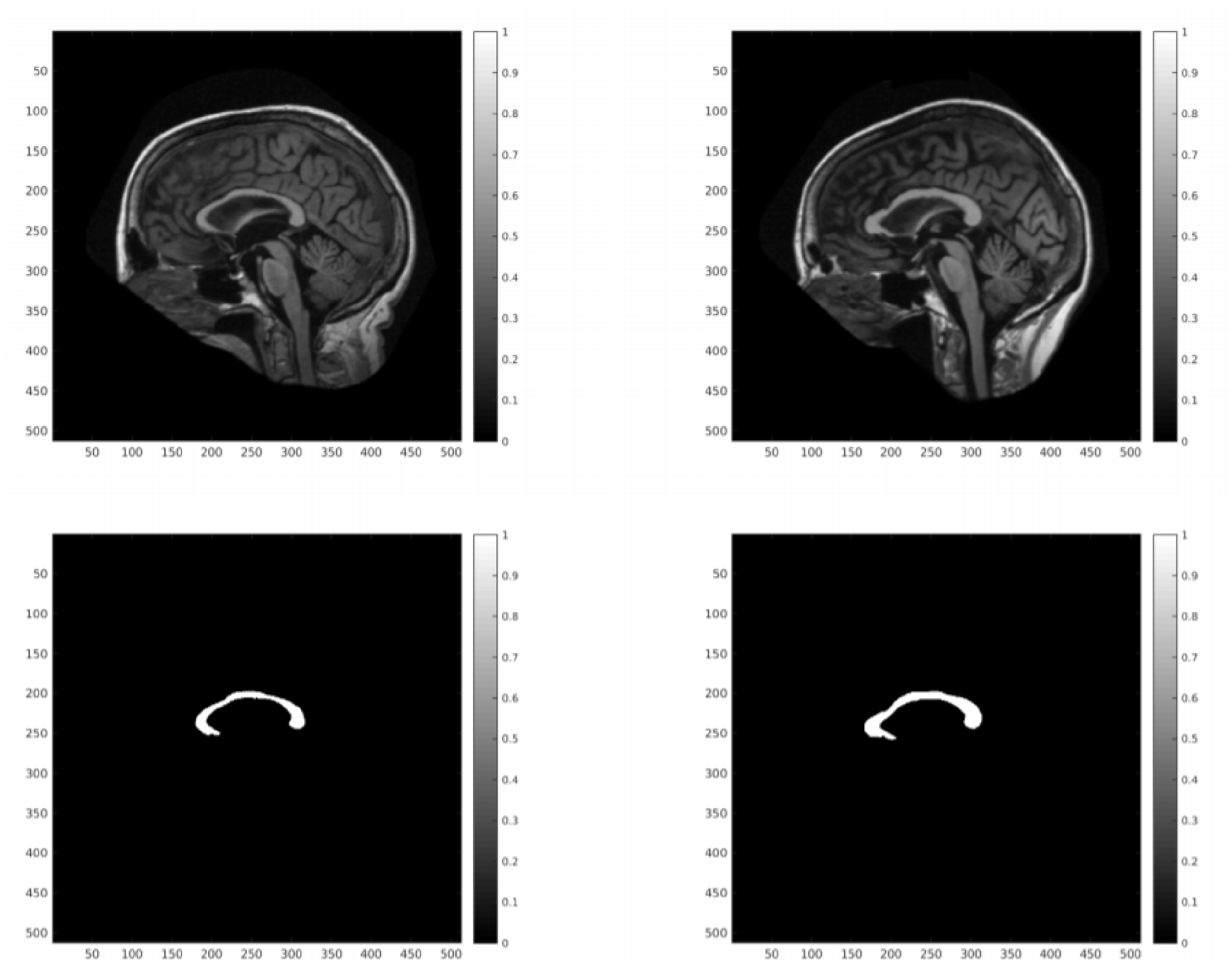

Figure 1: Corpus callosum for control (left) and Alzheimer's (right) subjects used for this exercise, shown with colorbar.