

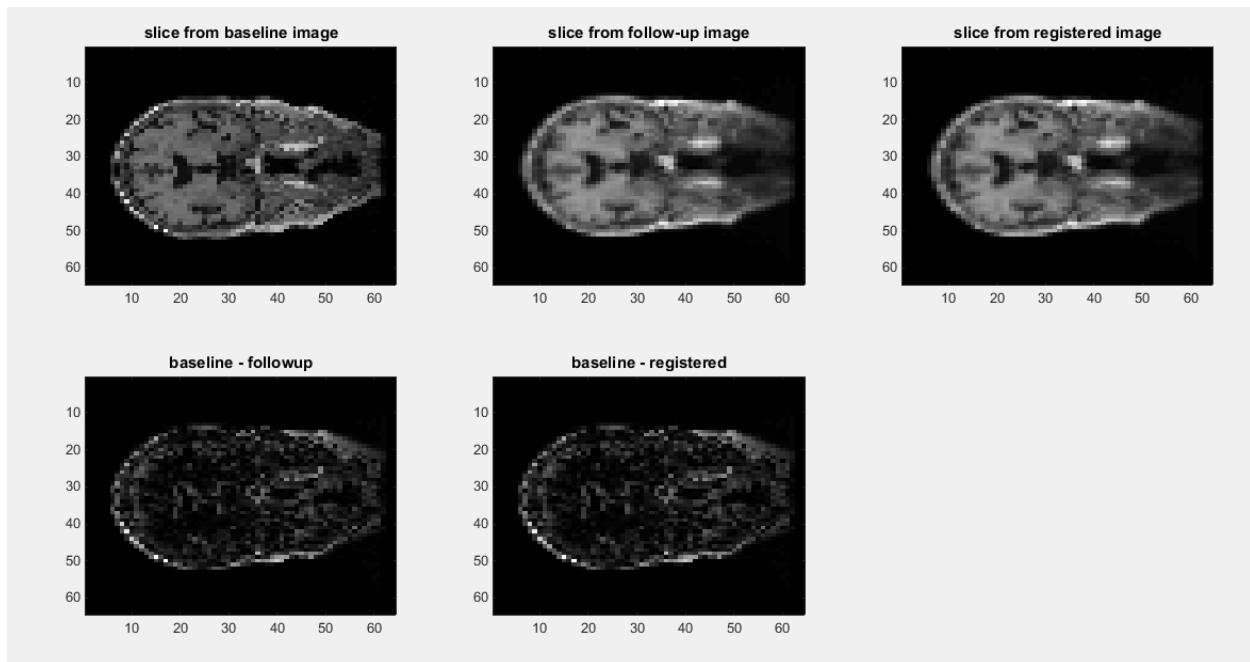
# Medical Image Analysis

## Fifth hand-in

## Improvements at registration

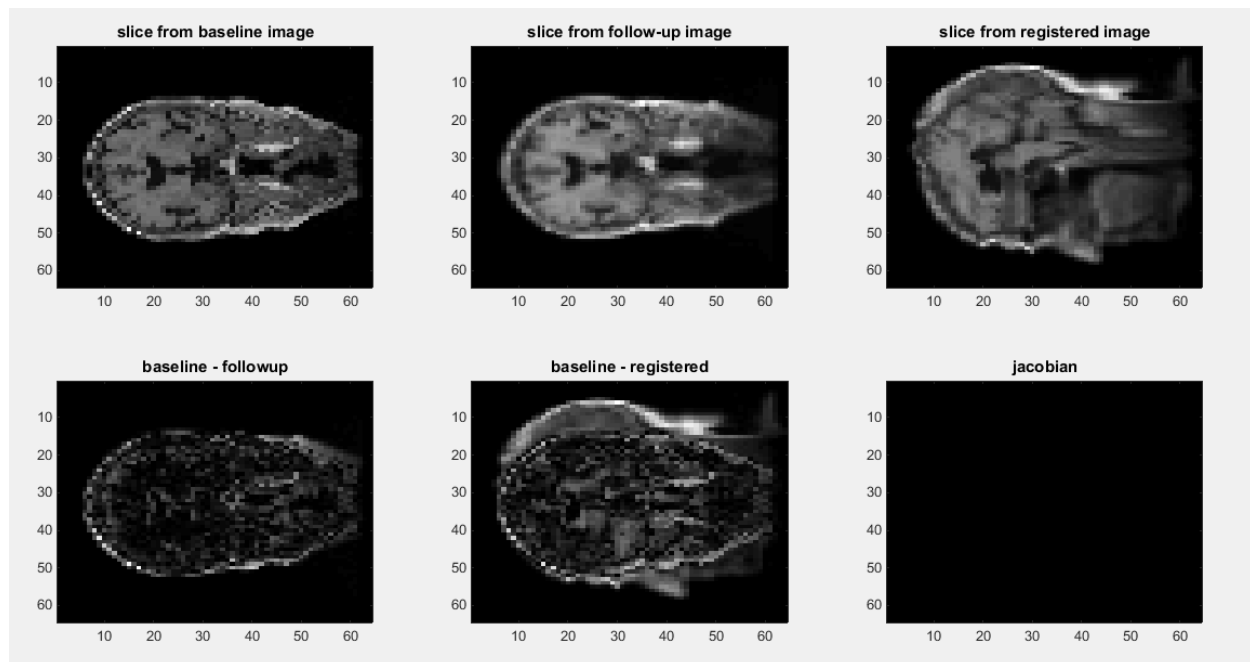
Before making the Jacobian, I did some more attempts to fix my registration and make them work so I can get proper results since my non-rigid registration was not working properly.

First of all, I used the Akshay's functions for calculating the cost of rigid transformation and used that one to optimize the parameters for translation and rotations. With much struggle, I made it work properly and obtained the following result:



As it can be seen in the images above, there is a slight difference between the difference image between the baseline and unregistered follow-up and the difference between the baseline and my rigid registered image. The sum-squared difference before the registration is 661.5 and after is 431.9 so this is another proof that I obtained an improvement.

Now with this registration I was expecting to obtain better results from non-rigid registration but, unfortunately, the results are same, the non-rigid registered image is still looking very bad and the sum-squared difference is really big.



I really do not understand why the registered image looks so bad since I implemented the algorithm as explained during lectures and as I told you, I worked together with Lou last week at the non-rigid registration and she got nice results with almost same code.

However, it appears to be a slight improvement from what I had at previous submission so I think there is a small mistake I make somewhere. The main difference I think is because I made the rigid registration properly and also I used `interp` instead of `interp3` function from Matlab that caused me issues at the rigid registration also.

## Jacobian determinant

Now, for the Jacobian determinant I have done readings on several materials on the Internet to understand how to implement the algorithm and I have done as follows:

```
%compute Jacobian
dx = deltaPoints(:,1);
dx = interp3(I2,
reshape(dx,size(I2)),reshape(dx,size(I2)),reshape(dx,size(I2)));

dy = deltaPoints(:,2);
dy = interp3(I2,
reshape(dy,size(I2)),reshape(dy,size(I2)),reshape(dy,size(I2)));

dz = deltaPoints(:,3);
```

```
dz = interp3(I2,
reshape(dz,size(I2)),reshape(z,size(I2)),reshape(dz,size(I2)));

J = my_jacobian(dx,dy,dz);
```

In the code above, I am using the delta points obtained by the SplineInterpolation function from Akshay to calculate the movement that has to be applied to make the registration. I split into matrices dx, dy and dz to have 3 matrices with the same of the image with the difference on each of the dimensions.

Now, inside the my\_jacobian function, I am doing the following calculation:

$$\nabla \vec{T}(\vec{r}) = \nabla \left( \vec{r} + \vec{d}(\vec{r}) \right) = I_3 + \begin{bmatrix} \frac{\partial dx}{\partial x} & \frac{\partial dx}{\partial y} & \frac{\partial dx}{\partial z} \\ \frac{\partial dy}{\partial x} & \frac{\partial dy}{\partial y} & \frac{\partial dy}{\partial z} \\ \frac{\partial dz}{\partial x} & \frac{\partial dz}{\partial y} & \frac{\partial dz}{\partial z} \end{bmatrix}$$

And the implementation is quite straightforward, and it looks like this:

```
function J = my_jacobian(dx,dy,dz)

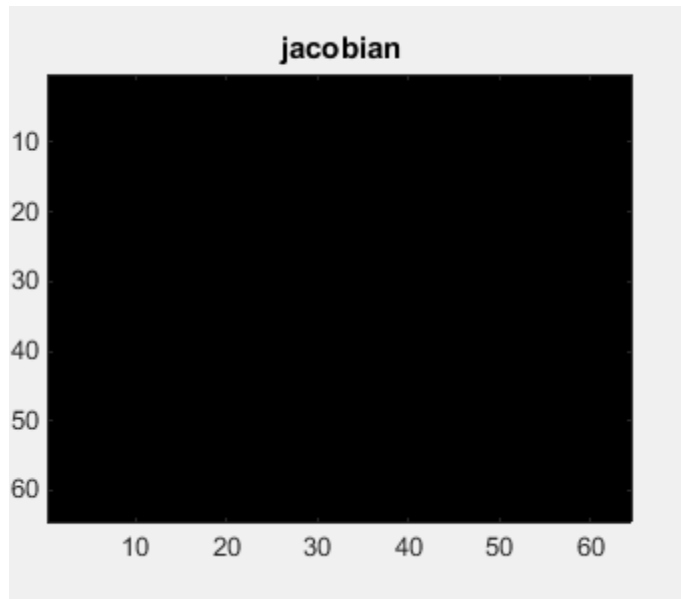
%calculate gradients
[gx_x,gx_y,gx_z] = gradient(dx);
[gy_x,gy_y,gy_z] = gradient(dy);
[gz_x,gz_y,gz_z] = gradient(dz);

%determinant calculation
J = gx_x.*gy_y.*gz_z + ...
    gy_x.*gz_y.*gx_z + ...
    gz_x.*gx_y.*gy_z - ...
    gz_x.*gy_y.*gx_z - ...
    gx_x.*gx_y.*gx_z - ...
    gy_x.*gy_y.*gy_z;

%add identity matrix
I3 = zeros(size(dx));
I3(1:1+size(dx,1)+size(dx,2)*size(dx,3):end) = 1;
J = I3 + J;

end
```

I first calculated the gradients on each direction for the difference matrices and then calculated the determinant from equation above. At the end I generated the 3d identity matrix and added it to the determinant result as it appears in the transformation theory.



Unfortunately, the result from calculating the Jacobian of my transformation is completely blank and once again, I have no clue why it happens. I know from the class that I should also use the regularization parameter but I am not expecting to get blank matrix even if it is without the regularization.

For me it appears that the implementation of Jacobian is correctly, as I read on Wikipedia and other online resources but I think I get wrong result because I am not having a correct input, since my registration is not working properly. I am pretty sure that somewhere on the pipeline, before Jacobian calculation something is wrong.