Assignment 2: IT 530 (Image Representation and Analysis)

Due Date: 3rd October

In this assignment, you will implement an information theoretic method for image alignment. You will work with the following image:

http://intranet.daiict.ac.in/~ajit_r/IT530/ims_retino.jpg

Let us call this image as I1. Firstly, you need to create a distorted version of this image by taking its negative, then applying a translation of 2 pixels in the X direction and -3 pixels in the Y direction. Let us call this image as I2.

Now you will crop out a 241 x 241 from around the center of I1, add noise of standard deviation 10 to it, and call the noisy cropped image as J1. Likewise obtain J2 from I2.

1. Your first task is to find the translation parameters that will best align J2 with I2. For this you will run a brute force search from -15 to +15 pixels (in both X and Y directions, in steps of 1 pixel) in order to maximize the mutual information. Compare your estimates with the true estimates. Repeat this for minimizing the joint entropy. Estimate both mutual information and joint entropy using histograms with different bin counts: 5, 20, 50 and 150. Plot out the landscape of joint entropy (with all 4 bin counts) w.r.t. both translation parameters. Repeat for mutual information. Use the 'surf' command in MATLAB for this.
2. Repeat the above experiment with only bin count of 20, with the brute force search from -200 to +200 (steps of 5, in both directions). Comment on the results.
3. Repeat the above experiment (using all 4 bin counts) for the case of zero translation and only 30 degree rotation to create J2 from I2. Plot out graphs of joint entropy (with all 4 bin counts) w.r.t. angle of rotation, all on the same scale. Repeat for mutual information. Comment on the results.
4. Read the paper by Maes and Collignon on the website to understand the concept of partial volume interpolation (PVI). What are the advantages of PVI in image registration? (You do not need to implement PVI).

Note:

One of the images (I2) is kept fixed and J2 is moved during the search. For each set of values of the search parameters, compute the joint histogram p(I2 = a1,J2 = b1) where a1 and b1 are two histogram bin values. Compute the marginal histogram p(I2=a1) and p(J2=b1) by appropriately summing across the row/column of the joint histogram. There is a subtle reason (apart from computational cost) why this is better than computing the marginal histograms afresh. What is it?