Project 2: Super-resolution Imaging

Introduction. This project will teach you how to perform super-resolution imaging on a simple simulated dataset. While the steps to localize the individual points are relatively straightforward, creating an algorithm that is robust to noise can prove difficult. ***This project is due Friday, October 6***.

Part I: Localize a point source. (*Useful Matlab functions:* ***lsqcurvefit, cat, imregionalmax, optimoptions, find, ind2sub, @ (function handle), imfilter****)* First, you will write a function that can localize a single, blurred point source in an image.

Use the PointSource.m file provided. Begin with a noise-free single-point image. Write a function that takes in the image and returns the x- and y-coordinates of the point. I would suggest taking the following steps:

1. Low-pass filter the image (to suppress noise)
2. Threshold (force all small pixel values to 0. This further suppresses the noise).
3. Find the local maxima in the image
4. Use lsqcurvefit to fit a 2-D Gaussian function to the image. Use your local maximum location and the known standard deviations of the point spread function to set initial parameters.
5. (optional) Determine the error between the 2-D Gaussian fit with the actual image. This could be used to remove bad fits.
6. Return the x- and y-coordinates (i.e., the mean of the fitted Gaussian function).

Answer Question 1.

Now add a few different levels of noise to the image. Use SNR values of 0 dB, 10 dB, 20 dB, 30 dB, and 40 dB. For each level of noise, localize the point 100 times and determine the spread of the localized x- and y-locations (i.e., find the standard deviation of the estimated coordinates). ***Answer Question 2.***

Part II: Super-Resolution Image Reconstruction. Once you are confident your function works on the single point source image, you should use it to reconstruct a more realistic image. I have provided some simulated PALM data (PALMData.mat). The data has 1000 frames. Each frame has between 1 and 8 randomly located blurred points. You can see how the data was generated by looking at the attached PALMDataGeneration.m file. This file also has the precise form of the PSF which you should use in your code. You can get an estimate of the "conventional" fluorescence image by averaging over the 3rd dimension of the 1000 frames. Use this image for future comparison.

Use your code from Part I to localize the points in all 1000 frames. Special care must be taken in frames with multiple local maxima. I would suggest making a new image for each local maximum then extracting a small region containing the point. Also, start with individual images containing 1, 2, 3, … 8 points and make sure your code works on these images before processing the entire 1000 frames. ***Answer Question 3.***

Finally, add Gaussian noise to the 1000 images to result in a signal-to-noise ratio of 20 dB. Reconstruct the images to form a single super-resolution image. Again, start small with individual frames to ensure your code works before scaling it up. ***Answer Question 4.***

Part III: Challenge (optional, not for a grade). I have included an additional dataset in the “Challenge.mat” file. The data includes a set of 5000 32x32 pixel acquisitions with a significant amount of noise. The goal is to reconstruct a 256x256 pixel image from the data. If you want to complete the challenge, then please submit a .mat file containing your image and the .m file with the code that reconstructed the image. I will show the submissions in class and the image with the highest structural similarity to the original high-resolution image will be the winner.

Part IV: Questions.

Question 1. How close are your x-y coordinates to the ground truth in the noise-free case? Is the result consistent if you change the location of the point source? Comment on any sources of error.

Question 2. Plot a scatter plot of the estimated x-y locations for the 20-dB case. Estimate the resolution of the technique for the 20-dB case. How much of an improvement is this over the original image? Finally plot the standard deviation of the x-y locations versus the SNR. What trends do you notice?

Question 3. Plot the reconstructed super-resolution image. Compare it to the original Thayer.jpg image. Have you accurately reconstructed it? Are there any incorrectly localized points? Comment on why this might be the case.

Question 4. Plot the reconstructed image from the noisy data. Compare it to the result from Question 3. Comment on any differences.