Prof. Raghuveer Parthasarathy
University of Oregon; Fall 2024

Physics 410/510 Image Analysis: Homework 4

Due date: Wednesday October 23 by 11:59 pm, submitted through Canvas. You'll see in "Assignments" a place to submit a **PDF**.

Note 1: Please see the "Additional Notes" document – I think you'll find it helpful!

Note 2: It is likely that I will extend the due date for this assignment, making #1-5 due Oct. 23 as usual and the rest due later ***if*** you describe your progress or intended approach on #6, etc.

Note 3: I haven't been emphasizing class participation, mainly because it has been great. However, participation isn't uniform, and it is important (and useful!) for the course. (See also the syllabus.) I'll try to call on *everyone* next Thursday, and I will specifically assign points for this.

Reading: All PDFs are on Canvas in \Readings

- [optional] This describes how to make a simulated image of a point source, similar to what we discussed in class: The first few sections of Materials and Methods, page 2 through "noise model" on page 3, of: M. K. Cheezum, W. F. Walker, W. H. Guilford, "Quantitative comparison of algorithms for tracking single fluorescent particles," *Biophys J* **81**, 2378-2388 (2001). https://www.cell.com/biophysj/fulltext/S0006-3495(01)75884-5 PDF: cheezum guilford tracking algorithms 2001.pdf
- [Very Optional, for those who want more a detailed treatment of the optics of the point spread function.] A few pages on point-spread-functions: Chapter 3, through "circular lenses." M. Gu, *Advanced Optical Imaging Theory*, Springer, 1999. A PDF is posted: "Gu Chapter 3 through 3p2p1 PointSpreadFunction.pdf". **Note** that λ here is not the free-space wavelength, but the wavelength in the medium: λ_{free}/n.

Upcoming, in case you want to get an early start:

- Sections 1-3 of Parthasarathy and Small, "Superresolution Localization Methods," *Annu. Rev. Phys. Chem.* **65:**107-125 (2013). (*http://www.annualreviews.org/doi/abs/10.1146/annurev-physchem-040513-103735*)
- R. J. Ober, S. Ram, E. S. Ward, "Localization Accuracy in Single-Molecule Microscopy," *Biophys. J.* **86**, 1185–1200 (2004). *https://www.cell.com/biophysj/fulltext/S0006-3495(04)74193-4*
- [Optional] Regarding super-resolution and the challenge of finding objects, you might like this video, *http://www.ibiology.org/ibioseminars/cell-biology/xiaowei-zhuang-part-1.html*, which gets at the methods described in the first section or two of the Parthasarathy and Small paper.

1 Frequency modulation. (4 pts.) Download "Buster_Keaton_General_Train_512.png" from Canvas, a still from Buster Keaton's wonderful film "The General" (1926), which you all should watch, especially since it was filmed in Oregon! I'm sure it's on Kanopy. Also download "Buster_Keaton_General_Train_512_sineMod.png", which is the same image multiplied by a sine wave ($\sin(2.0\ \pi\ x\ /\ P_x)$, with period P_x = 8 pixels, scaled to [0, 255]). Fourier Transform each image

and show the amplitude arrays. Take the log of the amplitude so that we can see its range easily. You can copy the Fourier Transform code I provided for last week's homework. Look carefully and describe what's different between the two Fourier Transforms. Explain why they look like they do; this doesn't have to be a mathematical proof, but it should be convincing and plausible. Be ready to share your explanation in class. *Hint:* Recall trigonometric identities, especially for multiplication! (If you email me by Tuesday with your thoughts on this, I may provide more of a hint.) (Don't spend a lot of time on this.)

2 Quantized aggregates. (8 pts.) Download "emitters_33px_100ph.png," a simulated image of 100 point-sources arranged on a grid. Each emitter is located at a multiple of (33, 33) pixels from the top left. Imagine that these are images of single molecules, a protein linked to green fluorescent protein for example. You want to know: do these proteins exist as monomers (one unit), dimers (two units), trimers (three units), ..., or a combination of these? In other words, is the brightness of the dots quantized, and if so, how many quanta are there per dot? Is this the same for all of them? Your task is to figure this out from the image.

- **(a)** Clearly explain each step of your assessment in addition to stating the answer. (Pasting code without explanations will get zero points.) Except for the placement on a grid, this is a very "real-world" problem – in fact, it was inspired by a graduate student talk I attended!
- **(b)** Show a histogram of the intensities of the molecules. (Why? Think about this, or it may become obvious after making it.)
- **(c)** Of the 100 molecules, how many do you think are monomers, dimers, trimers, ...?

Notes:

(*1*) You're only interested in brightness for now, so this problem doesn't require any of the localization concepts we're about to learn – in fact, I could have assigned it last week.

(*2*) There are, of course, background intensities of various sorts that you'll have to figure out how to deal with. Take the spacings, multiples of (33, 33) pixels as a given, though.

(*3*) The same sort of image but without any background or noise except photon noise, and with brighter molecules, is at "emitters_33px_1000ph_noNoise.png.". ***Suggestion:*** Try your analysis on this first! You should find 69 monomers, 20 dimers, 11 trimers, and nothing higher. *Note:* I'm not asking you to apply some rigorous clustering algorithm; determining bins "by eye" based on your histogram is fine.

(*4*) I am deliberately not spelling out how to solve this – just like in real life, you'll have to figure it out! You have all the tools in hand...

(*5*) If you're confused, work backwards: Suppose you had an image with 50 monomers and 50 dimers. What would it look like?

3 A high-resolution PSF. (4 pts.) Write a function that calculates the point spread function, outputting it as an N x N array as a function of input parameters: N, the wavelength of light, the numerical aperture, and the pixel scale (i.e. the distance in the focal plane that each pixel corresponds to.) You'll use this function in various exercises in this and future problem sets, so make it a good one! Note that for a circular lens or aperture, the point spread function

$$PSF = 4\left(\frac{J_1(v)}{v}\right)^2$$

where:

- J_1 is a Bessel function of the first kind of order 1 (see note below)
- $v = (2\pi/\lambda)\,\mathrm{NA}\,r$
- λ = free-space wavelength of light
- NA = the numerical aperture
- r = position in the focal plane (in the same units as λ, of course)

Submit three images (i.e. three PSF arrays), each spanning approximately 1 μm x 1 μm: **(i)** Use λ = 0.5 μm (green light), NA = 0.9, N = 101; **(ii)** the same as (i), but λ = 0.4 μm (blue light); **(iii)** the same as (ii), but NA = 0.5.

Also submit the code for your function.

Notes:

- J_1 in Python can be calculated with the scipy package: `scipy.special.j1` . The whole `scipy` package is too large to import, so use `import scipy.special as scipy_special` and `scipy_special.j1(`v`)` . In MATLAB, $J_1(v)$ is `besselj(1,`v`)`.
- For $v \rightarrow 0$, $J_1(v) / v \rightarrow 0.5$, so PSF($v$ == 0) = 1.

Suggestion: First make an N x N array of distances from the center (r), then make an array of v values, and then calculate the PSF array.

Please see the "Additional Notes" document!

4 A worse worm image. (4 pts.) Download from Canvas "fetter_Celegans_cellfig10.jpg", shown below, a transmission electron micrograph of a slice of a *C. elegans* embryo, from Rick Fetter and Cori Bargmann (source: http://www.wormbook.org/chapters/www_intromethodscellbiology/intromethodscellbiology.html). Note the scale bar. Create an image that shows what the slice would look like if imaged with visible light (e.g. wavelength 0.53 μm) rather than electrons. Use your PSF function from the previous problem, and NA = 0.7. Don't simulate noise or pixelation, just consider the consequences of diffraction (i.e. the wave-nature of light). Submit the simulated optical image. (Note: the calculation may take several minutes!)

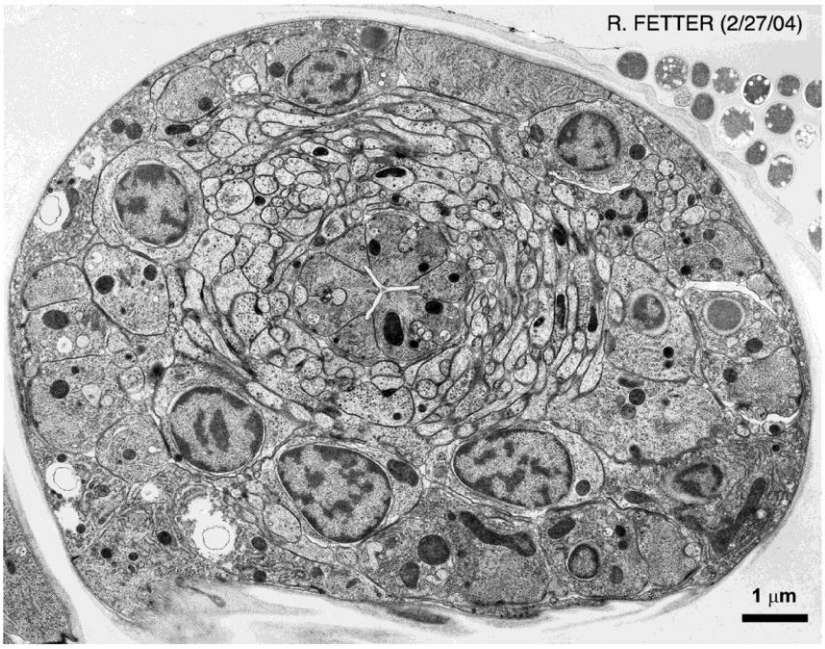


***Figure**. A TEM image of a slice of a C. elegans embryo. Source: Rick Fetter and Cori Bargmann.*

5 SNR and Poisson Noise (1 pt.) For a Poisson-distributed random variable like the total number of photons (N_{photon}), how are N_{photon} and the signal-to-noise ratio (SNR) related? Remember: "noise" is the standard deviation of whatever we're measuring. Yes, this is a very short question.

6 Simulated point sources, Part 1. (9 pts.) Now, let's make a more realistic image of a point source, incorporating pixelation and noise. Write a new function, building on what you wrote in Problem #2.

- **(a)** (2 pts.) *Pixelation:* In #2, you made a "high resolution" PSF on a fine grid. Now create a grid corresponding to actual, typical camera pixel scales by summing the intensities in each fine-grid element corresponding to each camera grid element. For example, if your camera scale is 0.1 um/px and your fine grid is 0.01 um/px, each camera pixel will sum 10 x 10 elements of the fine grid. (Feel free to make things easier by having the camera grid scale be an integer multiple of the fine grid scale.) Submit a NxN = 15x15 simulated camera image calculated for λ = 0.5 μm, NA = 0.9, camera scale 0.1 μm/px. The number of camera pixels (N), λ, NA, the camera scale, and the "fine grid" scale should be the inputs to your function.
- **(b)** (3 pts.) Scale the output of your array in (a) so that the sum of all the pixel intensities is (N_{photon}), where N_{photon} is another input to your function. *Noise:* The PSF tells us the *probability* of getting photons at various positions. The value of the scaled PSF from part (c) is the *average* number of photons arriving at each pixel, but the *actual* number of photons at each pixel is a Poisson-distributed random number. How do we go from "proportional to" to the number of photons? Modify your function to implement Poisson noise statistics. *Hint:* it will require one line of code. Submit two examples with N_{photon} = 50, and two with N_{photon} = 500.
- **(c)** (4 pts.) Submit your code: a function that makes a simulated image of a point source in the center of an NxN grid, with the inputs described above. It should be clear and comprehensible!

Please see the "Additional Notes" document!

7 Simulated point sources, Part 2. (3 pts.) Modify your earlier function to allow the point source to be located somewhere other than the center of the image; the position (xc, yc) measured from the center, in physical units like μm, should be an input to the function. This should involve changing one or two lines of code; describe or show these lines.

8 Simulated point sources, Part 3. (3 pts.) We're almost done! Modify your function to add a noisy background intensity – make it Poisson-distributed with some mean value B **at each pixel**. (Note: this is just like what you did in #8 of HW2.)

(i) Show the outputs for λ = 0.5 μm, NA = 0.9, camera scale 0.1 μm/px, N_{photon} = 50, B = 2 photons, and (x_0,y_0) = (0,0)

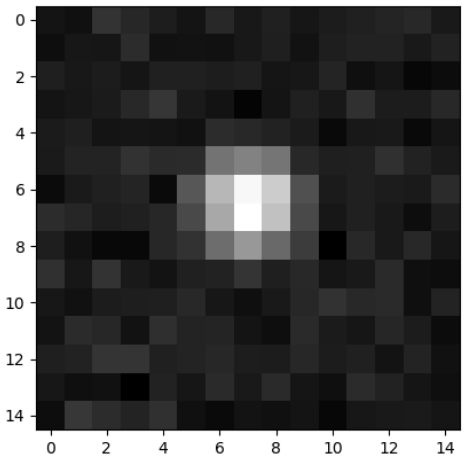
(ii) Show the outputs for the same parameters except (x_0,y_0) = (0.03, 0.03) **microns** (i.e. 0.3, 0.3 pixels), and then …

(iii) both of the above, but for N_{photon} = 500.

Don't worry if one of these differences is not very evident! (*Suggestion:* make N_{photon} = something very large, to make sure the output looks like it should.)

Note that you've made a function that makes a very realistic simulated image of a point source (e.g. a single molecule or a star)! We'll use this function next week!

Example, y-offset = -0.5 pixels; Nphoton = 5000; B = 50 photons:



9 Simulating a ring. (4 pts.) Describe in words the steps involved in making a function to simulate the image of a ring. Comment on what steps are similar / different to the point source simulation. You **don't** have to write this function!

10. Participation points, so I make sure to record this somewhere. (10 pts.)