UCF Physics: AST 5765/4762: (Advanced) Astronomical Data Analysis

Fall 2019 Homework 10

Due Tuesday 12 November 2019

Work:
Become sufficiently familiar with aperture photometry to:

1. Measure the relative fluxes of stars in an image.

Resources:

1. ALL READINGS ON PRIOR ASSIGNMENTS!
2. **AST5765 only:** Horne, K. 1986. An Optimal Extraction Algorithm for CCD Spectroscopy. *PASP* **98**, 609–617. Think about how you would apply this to photometry. You wouldn't need the elaborate bad-pixel finding routine in that case; you'd just use the routines you already know for images, which makes the algorithm much simpler, except how would you calculate and center the PSF?

Hand in:

This assignment continues the correction of the data in `WebCourses/Files/data/hw7`. Be careful to follow the instructions above the questions in HW7.

Follow the convention that the center of a pixel has the integer position of the pixel and that the lower-left corner of the array has position (-0.5, -0.5).

1. (10 points) Copy the routines and homework files from the previous homework (including any inclusions of prior assignments) into the directory for this assignment. Correct any errors you may have had, making a comment that says "`# FIXED:` " and giving the date. You may refer to the posted solution, but if you do so you must state what you used from it. The only non-comment for this problem should be the prior homework file run as a batch job (e.g., `from hw9_sol import *`). This will run all the homework files as batch files, each calling the prior one as its first thing, back to HW7. Of course, make sure they all still run without errors. It is a good idea to compare your results to those in the solutions.
2. (10 points) Make a table for your photometric results. Do this by explicitly initializing a 3D floating-point array with the table data. The axes should be star number, frame number, and parameters. Start with a header comment for the parameters and use `np.nan` to indicate that you haven't set a value yet. You will do photometry on three stars in each of three frames. Fill in the position of the first star in each frame, and the positions of the other two stars in the first frame. Each row should contain:
   (a) the approximate y location,
   (b) the approximate x location,

(c) the width of the star,

(d) the fitted y location,

(e) the fitted x location,

(f) the star's flux level, and

(g) the average sky level.

You may copy-and-paste this table:

```
# Stellar photometry:
photometry = np.array(
  [
  #  yguess, xguess,     width,     cy,     cx,     star,     sky
  # star 0
   [[   698,    512,    np.nan, np.nan, np.nan,   np.nan,  np.nan],  # frame 0
    [fillin, fillin,    np.nan, np.nan, np.nan,   np.nan,  np.nan],  # frame 1
    [fillin, fillin,    np.nan, np.nan, np.nan,   np.nan,  np.nan]], # frame 2
  # star 1
   [[   668,    520,    np.nan, np.nan, np.nan,   np.nan,  np.nan],  # frame 0
    [np.nan, np.nan,    np.nan, np.nan, np.nan,   np.nan,  np.nan],  # frame 1
    [np.nan, np.nan,    np.nan, np.nan, np.nan,   np.nan,  np.nan]], # frame 2
  # star 2
   [[   568,    283,    np.nan, np.nan, np.nan,   np.nan,  np.nan],  # frame 0
    [np.nan, np.nan,    np.nan, np.nan, np.nan,   np.nan,  np.nan],  # frame 1
    [np.nan, np.nan,    np.nan, np.nan, np.nan,   np.nan,  np.nan]]  # frame 2
  ], dtype=float)
```

Calculate the offset for star 0 between frames 0 and 1 and frames 0 and 2. Add the appropriate offset to the positions of stars 1 and 2 in frame 0 to get their positions in frames 1 and 2, and update the table. Use a loop. Find the numbers of stars and frames by asking the table.

For small datasets like this one, this offset procedure seems like extra work since guessing the positions of a few stars isn't hard. For big datasets, this is the only way to go, and "big" here is only a few dozen stars. For big datasets, you would allocate the table with `np.zeros()`, fill with `np.nan`, and set the few known values in assignments rather than quoting the whole table as done here.

3. (10 points) Use the N-dimensional Gaussian fitting function in `Webcourses/Files/python/gaussian.py` to estimate the 1σ widths and location of each star from your table. Record the average of the two width values for each star in the table, along with the fitted y and x. To avoid NaNs, other stars, and bad pixels, copy a subarray around each star, subtract the subarray median value to get a $\sim$zero background, and fit in that image. Do this in a COPY of the subimage, rather than a view into the full frame (why?). Be sure to adjust the guess y and x values to be relative to the subarray, and to adjust the positions you get from the subarray fits back to the coordinates of the full array. Print the table (or its filled-in portion) and copy-and-paste it to a comment in your homework file. Check whether your final positions are consistent with your guesses.

4. (10 points) Take the average of all widths to get a single width that applies to the whole dataset. Set variables for the photometry aperture radius, two sky annulus radii, and sub-image size as multiples of your fitted width. The sub-image size is the size of the sub-image that will be copied from each image for photometry. **Justify your choices** of multipliers for each new variable in a few commented sentences.

5. (a) (4 points) You observe for 10 min and get S/N = 10. If you observe for 30 minutes, what is the best S/N you can hope for, and why? What might reduce the S/N?
   (b) (3 points) A star gives you S/N = 10 in 10 min. If you observe a star that is five times as bright, what is the best S/N you can hope for in a similar observation, and why?
   (c) (3 points) If you observe the same star for 10 min with the same instrument in a telescope that has four times the diameter, what is the best S/N you can hope for, and why?

6. (10 points) Look up the planet GJ 436b in the Exoplanet Encyclopedia (exoplanet.eu). Find the STAR's effective (equivalent blackbody) temperature, radius, and distance from us. What is the best signal-to-noise ratio one could hope to get on the stellar flux by observing this star for one minute with an 85 cm space telescope, a perfect instrument, and a filter that passes 5% of the energy in the spectrum, centered at 3.6 μm (so, you can pretend that all the photons are at 3.6 μm, and there is no sky background)? You should get a pretty big number, compared to S/N that we have computed in the past. It's a bright star and a wide filter.

7. (10 points) Include a copy of your class log file in your handin. Print the Git log for your main homework file.