Introduction to Astronomical Data Methods

The work below is to be completed by the end of the course. Please contact me (marla.geha@yale.edu) with any questions.

Searching for Asteroids and Neptune's Moons

We will reduce images taken last year with Yale's La Silla-QUEST survey telescope The survey utilizes the 40" ESO Schmidt Telescope at the La Silla Observatory in Chile. The QUEST camera is a mosaic of 112 CCD's covering the full field of view of the telescope (approximately 4 x 4 square degrees). For more information on the camera and survey see:

http://adsabs.harvard.edu/abs/2012AJ....144..140R

1. The pixel scale of the QUEST camera is 0.88" per pixel. Each CCD in the QUEST camera is 600 x 2400 pixels. What is the field of view in arcseconds? How does this compare to the full moon (which is roughly 0.5 degrees in diameter)?

2. The data below are images of the planet Neptune and its moons (plus a few Main Belt asteroids). Neptune orbits the Sun every 165 years at a distance of 30 A.U. Over the time period of one night (8 hours), how many arcseconds does Neptune appear to move in the sky? How many pixels does this correspond to on the QUEST images? What about a main-belt asteroid at a distance of 3 A.U? Will these motions be observable in the QUEST data during one night of observing?

1. *Estimate parallax motion:* Determine the apparent motion (in arseconds) of the objects above due only to the parallax motion of the Earth over 8 hours. Is it reasonable to determine distances using the parallax only and neglecting the motion of the object itself?

2. *Neptune's Moons:* There are 13 moons of Neptune, we should be able to see the few brightest of these. Given the orbital distance from Neptune, what is the maximum distance we might find the brightest moons (Triton, Proteus and Nereid) in arcseconds? In QUEST pixels? Data on these moons can be found at:
   http://solarsystem.nasa.gov/planets/profile.cfm?Object=Neptune&Display=Sats

Data Reduction II -- Basic Image Data reduction

Download the data files from the class website. These are fits files and can be read into python using the (now familiar) `pyfits.getdata`. Headers can be read using `pyfits.getheader`. As you begin reducing the data, we suggest writing out new files using `pyfits.writeto` and viewing the processed images in ds9.

Please explain each step of your data reduction, including screen grabs of intermediate steps.

3. *Examine the science data*: Load a raw science image into ds9. Zoom out/in of the data. Load in two science images and blink between them. List at least four features of the raw science data that are associated with the telescope/CCD, rather than the science objects themselves.

4. *Examine the dark frames:* We will correct for the bias and dark current simultaneously using the dark frame. Briefly explain why can we combined these two reduction steps and state how we will use this single image to correct the science data. Load the dark frames into ds9 and python. Take a look a header and note the exposure times. Do these match the science and flat frames?

5. *Examine the Flat Field:* The flat field images were taken during twilight when the sky is fairly bright. By taking a series of sky images and media-combining we will create a flat field image. Examine a few flat field images and describe what you see.

6. *Median Combine the Flat Field:* Read the individual flat fields into python. Prepare for median combining by first subtracting the appropriate dark frame and then normalizing each flat field so that the mean value is equal to unity. Combine the flat field images (we suggest using numpy.median). Plot a histogram of the final combine image to demonstrating that the mean of this image is one.

7. *Dark Subtraction and Flat Fielding:* Apply the dark and flat field correction to the science data. Verify that these corrections are working by (1) showing an example of a hot pixel which has been corrected, and (2) showing that a region where a dust donut was seen has been removed.

8. *Trim the Overscan region:* Trim your images from so that the files include only science pixels (e.g., trim out the overscan region).

9. *Sky Subtract the Science Images:* Determine the sky brightness in each image by making a histogram of the image and determining the mode of the distribution. Subtract the sky value from each image.

10. *Shifting the science data:* Load three science images into ds9 and blink between images, noting that there is spatial shift between these images. In order to search for moving objects, we need to shift this image. Chose a few stars (min 4) and determine their (x,y) positions across the three images. Calculate the (x.y) shifts needed to align the second and third image to the same positions as the first. We suggest using the interpolation scipy.ndimage.interpolate.

11. *Visually Find Moving Targets:* To visually find moving objects, we will blink between the three reduced images in DS9. Load each file into a different frame in DS9 (frame-> new; file-> open). For one image, set the scale so that you can see faint objects and match the other frames (under the pull down menu frame -> match -> frame -> image to match the same zoom, frame-> match-> scale and colorbar to match display). Now use the buttons to blink between images (frame-> blink). Search for objects which appear to move between the three images. You should find at least one (Neptune itself doesn't count), but there are several others. Try search at different images stretches, particularly to look for close-in Neptune moons. (Bonus: Create a gif image animation for one of your moving objects).

12. *Measure Positions:* For each moving object found above, measure its position on each image. Determine the time each image was taken from information in the image header and calculate the rate this object is moving in arcseconds per day. You can estimate positions in each image by eye or use your own algorithm as discussed in the Extra Credit below.

13. *Estimate Parallax Distances:* Determine distances for each moving object, assuming motion is from parallax only. Given these distances, discuss the nature of each, noting whether it is a moon, asteroid or something else.

Extra Credit #1: To determine the names of the objects you found above and/or whether or not these objects have been found before, we need to determine the Ra/Dec of each object. To do this, we will determine the Ra/Dec of Neptune and bootstrap the Ra/Dec of our unknown objects. For one image, estimate the x/y position of Neptune (it's saturated, but try to find the center). Using the time of the image from the header, determine the Ra/Dec of Neptune. Click on the blue links to change the name of the planet and the time of observation and the time step (chose hours):

http://ssd.jpl.nasa.gov/horizons.cgi

In our images, the Ra increases with increasing y-pixels and the Declination decreases (increasing negative values) with increasing x-pixels. Calculate the Ra/Dec of your moving objects in the one of your images. If you think your object is a Neptune satellite, see if it matches the Ra/Dec of objects here:

http://scully.cfa.harvard.edu/cgi-bin/mpcheck.cgi

If you think a given object is an asteroid, try inputing the Ra/Dec here (the observatory code is 809). Note that the Ra/Dec need to be input in the format "22 22 00.1":

http://scully.cfa.harvard.edu/cgi-bin/mpcheck.cgi

Extra Credit #2: Reduce second night of data. Report and document any additional moving targets and estimate distances.

Extra Credit #3: Write a script to quantitatively search for all moving objects in the images. Describe your algorithm and demonstrate it is working by recovering at least the objects which you found visually. Use this algorithm (or a variation of this) to determine the position of your moving objects more accurately than the by-eye calculation in #6