



ESOpY 3.0

15-16-17 April 2019

ESO Vitacura

PSF of photometric images

Project coordinated by Alessandro Razza

Abstract

Python packages for astronomers are rapidly evolving to adjust the community necessity to quickly analyze and plot astronomical data within the same framework. More than packages of a programming language, they work as authentic astronomy software providing for instance tools to detect astronomical sources and perform aperture or PSF photometry on images, as in the case of `photutils`. By implementing some features of `photutils`, `astropy` and possibly `astroquery` packages, we write a script capable to read fits files, detect stars on the images by using a model PSF and build an effective PSF (ePSF) out of the selected stars. Some optional exercise is also proposed to perform aperture photometry, to print out the list of sources with their parameters in a table and to use the stellar centroid positions, refitted with the PSF photometry routine, to evaluate the images astrometric solution. With a basic knowledge of python syntax and object-oriented programming (object instances, classes attributes and methods), working on this project simply results in finding and writing the series of instructions to perform the requested tasks.

Required background and skills

Being familiar with FITS file headers and the World Coordinate System (WCS) is required at a basic extent. Having worked with photometric data is not necessary, although it is required to have a basic knowledge of aperture and PSF photometry. No particular programming skills are required. However, understanding from ESOpY 3.0 lectures how a python object/class is instantiated and how arguments are passed to a function/class method can be beneficial.

Python packages

- python 3.6+ and other packages used for ESOPy 3.0 (e.g. numpy, matplotlib)
- [photutils 0.6](#)
Install with
`conda install -c astropy photutils`
from within your environment.
- [astropy 3.1](#)
Install with
`conda install -c anaconda astropy`
from within your environment.
- [astroquery 0.3](#) (not strictly necessary if you do not want to query GAIA stars for the astrometry or to match the stars)
Install with
`conda install -c astropy astroquery`
from within your environment.

Main project tasks

A list of the main task with a suggested sub-package to use is presented (tutoring is provided to avoid getting stuck)

1. Read a given set of photometric images (FITS files) ([astropy.io](#)).
2. Find stars in the images with a proper find-star algorithm ([photutils.detection](#)).
3. Implement a way to cut out non-stellar object (a first order solution could be matching the coordinates with GAIA stars with [astropy.coordinates](#) and [astroquery.vizier](#))
4. build an effective PSF (ePSF) out of the selected stars following (look [here](#) to avoid wasting a month doing it!)

Optional tasks

Although recommended to complete all of the proposed optional tasks, you can pick up only one or more from the following list:

5. Compute the aperture photometry (few passages more) and PSF photometry (it is an automatic outcome of the process for finding the ePSF) of the list of stars
6. Print on browser/save in a text file the list of stars with their parameters (pixel positions, coordinates, aperture photometry, PSF photometry)
7. Save the FWHM from the ePSF in the image header
8. Evaluate the WCS solution by comparing the PSF centroid positions of the stars (it is an automatic outcome of the process for finding the ePSF) with GAIA catalog