# Zè's Google Summer of Code Application

Zè Vinícius (`@mirca`)

February 28, 2016


## 1 Background

I am a senior year undegraduate student in Eletrical Engineering with plans to apply to a PhD program in Optics.

During the Summer of 2015, I worked as an undergraduate guest researcher with the Nanofabrication Research Group (NRG) at the National Institute of Standards and Techonology (NIST), USA. In that opportunity, I developed MATLAB code for single molecule localization microscopy (beyond diffraction limit).

Basically, I wrote a toolbox which provides to users easy access to maximum likelihood estimators for the position, number of photon counts, width, and number of photons from background of dye molecules, and the uncertainties associated with each of the mentioned parameters as well. In other words, the code fits the point spread functions, which was assumed to have a symmetric Gaussian shape, to, in general, Poisson data (source plus background).

In addition, I wrote fitting routines which takes into account specific instrument information, e.g., electron multiplying CCD cameras (EMCCD) and scientific CMOS cameras. This was important in order to fully characterize the counts' statistics, so that the fitting procedure would give meaningful results even in low counts scenarios. For example, counts from EMCCD images are not exactly Poisson distributed.

All that said, the problem of single molecule localization is, surprisingly, quite similar with the project that I am applying for, namely, "Implement PSF photometry for fitting several overlapping objects at once", with Brigitta Sipocz and Moritz Güenther as mentors.

I am quite sure the Google Summer of Code will be a very nice opportunity not only to become more involved and work closer with `photuilts` (`astropy`), but also to learn a lot, and to write readable, maintainable, and high quality code, of course.

Summary of relevant skills:

**Research oriented and problem solver student** I have 3+ years of experience as undergraduate research assistant (I have published about ten articles in international conferences and three papers in journals). Currently, I hold an undergraduate teaching assistant scholarship for the course of Probability and Statistics for Electrical Engineering and Computer Science.

**Programming and Open Source Development** Roughly 1+ year of experience with Python, 3+ years of experience with C/C++, and 3+ months with Github (I can follow the Astropy development workflow :D).

**Coursework** During an one year period, I was a visiting student at the University of Maryland, College Park, USA, in which I took graduate level coursework in Data Analysis, Stochastic Processes, and Estimation and Detection Theory.

**Github**

- Relevant pull requests to `astropy`:
  - Jackknife resampling: `https://github.com/astropy/astropy/pull/4439`
  - Circular statistics (WIP): `https://github.com/astropy/astropy/pull/4472`

# 2 Project Proposal

## 2.1 Project Title

Implement PSF photometry for fitting several overlapping objects at once

## 2.2 Proposal Title

PSF photometry for fitting multiple overlapping objects simultaneously

## 2.3 Proposal Description

### 2.3.1 Project Overview

Aperture photometry assumes that the background varies in a linear fashion in the aperture's vicinity. However, in a dense star cluster the background is usually not linear. Therefore, one may use point spread function (PSF) photometry in order to meaningfully measure the brightnesses of the sources. In the latter approach, an analytical function (usually Gaussian shaped) is fitted to each object in order to determine its position ($x, y$ center), amplitude (flux), and width. Nonetheless, localize and fit several overlaped objects simultaneouly is not a straightforward task. In fact, the applied algorithm must be able to distinguish (with high probability) how many objects there exist in a given subregion of a given image.

Shortly, the primary problem proposed by this project is to localize several overlaped sources (e.g. a dense star cluster) simultaneously. To do so, one can not "just fit a model with hundreds parameters". In fact, there exist several problems with this "brute force" approach, and the most critical one might be that the parameter space will have many dimensions (as many as the number of parameters, precisely), which almost certainly will make optmization algorithms to diverge or to get stuck in a local minima.

One possibly resonable approach would be to assume the existence of $n$ sources in each subregion and then proceed sequentially, i.e., fitting $n$ point spread functions (from $n = 1$ to $n = n_{max}$). Finally, some goodness-of-fit (e.g., bayesian information criterion, likelihood ratio) would be used to evaluate and decide which model would be selected.