Astronomy and Computing 16 (2016) 79–87

Contents lists available at ScienceDirect

Astronomy and Computing

journal homepage: www.elsevier.com/locate/ascom

Full length article

ComEst: A completeness estimator of source extraction on astronomical imaging

I. Chiu[a,b,*], S. Desai[a,b], J. Liu[c]

[a] Faculty of Physics, Ludwig-Maximilians University, Scheinerstr. 1, D-81679 München, Germany
[b] Excellence Cluster Universe, Boltzmannstr. 2, 85748 Garching, Germany
[c] Bosch Research and Technology Center North America, 4005 Miranda Ave #200, Palo Alto, CA 94304, United States

ARTICLE INFO

Article history:
Received 28 November 2015
Accepted 25 April 2016
Available online 9 May 2016

Keywords:
Survey
Photometry
Depth
Completeness

ABSTRACT

The completeness of source detection is critical for analyzing the photometric and spatial properties of the population of interest observed by astronomical imaging. We present a software package ComEst, which calculates the completeness of source detection on charge-coupled device (CCD) images of astronomical observations, especially for the optical and near-infrared (NIR) imaging of galaxies and point sources. The completeness estimator ComEst is designed for the source finder SExtractor used on the CCD images saved in the Flexible Image Transport System (FITS) format. Specifically, ComEst estimates the completeness of the source detection by deriving the detection rate of synthetic point sources and galaxies simulated on the observed CCD images. In order to capture any observational artifacts or noise properties while deriving the completeness, ComEst directly carries out the detection of simulated sources on the observed images. Given an observed CCD image saved in FITS format, ComEst derives the completeness of the source detection from end to end as a function of source flux (or magnitude) and CCD position. In addition, ComEst can also estimate the purity of the source detection by comparing the catalog of the detected sources to the input catalogs of the simulated sources. We run ComEst on the images from Blanco Cosmology Survey (BCS) and compare the derived completeness as a function of magnitude to the limiting magnitudes derived by using the Signal-to-Noise ratio (SNR) and number count histogram of the detected sources. ComEst is released as a Python package with an easy-to-use syntax and is publicly available at https://github.com/inonchiu/ComEst.

1. Introduction

The optical and Near-Infrared (NIR) imaging provides one of the backbones of astronomy. Since the early twentieth century, countless milestones have been made in various fields of astronomy based upon the data taken by the optical sky surveys, e.g., SDSS (York et al., 2000). Nowadays wide field optical/NIR surveys – such as the Pan-Starrs (Morgan et al., 2014), DES (Flaugher, 2005), KiDS (de Jong et al., 2013), HSC (Miyazaki et al., 2012), 2MASS (Skrutskie et al., 2006), WISE (Wright et al., 2010) and ATLAS (Shanks et al., 2015) – and dedicated deep imaging (e.g., COSMOS (Koekemoer et al., 2007) or HUDF (Beckwith et al., 2006)) have become the frontier of astronomical studies in various topics. In the next decade, the upcoming surveys – for instance, LSST (Ivezic et al., 2008) or *Euclid* (Laureijs et al., 2011) – with the unprecedented deep imaging of large portion of the sky will revolutionize our understanding of the Universe.

Nowadays, modern astronomical observations are imaged by charge-coupled devices (CCD) saved in the Flexible Image Transport System (FITS) format, which various source finders, e.g., Bertin and Arnouts (1996) are run on for detecting sources. To analyze the observed images taken by various telescopes, the astronomical objects are identified by the source finder, and then information about the photometric and spatial properties is extracted from the image. It is therefore critical to verify the source detection on the observed image for the analysis of, for instance, modeling the luminosity function or spatial clustering of galaxies. One of the most important factors for such studies is to quantify the completeness of the source detection on the observed images.

The completeness of the source detection indicates the fraction of objects present in the image, which can be detected by the source finder above a certain detection threshold against the

* Corresponding author at: Faculty of Physics, Ludwig-Maximilians University, Scheinerstr. 1, D-81679 München, Germany.
E-mail addresses: inonchiu@usm.lmu.de (I. Chiu), shantanu@usm.lmu.de (S. Desai), jiayiliu@usm.lmu.de (J. Liu).

http://dx.doi.org/10.1016/j.ascom.2016.04.005
2213-1337/

noise. The completeness is expected to decrease with increasing detection threshold because the required signal-to-noise ratio (SNR) of the source is higher. At the same time, the purity of the source detection – the fraction of the detected sources which are not spuriously detected by the source finder – becomes higher due to the high SNR as well. The lack of understanding the completeness (or purity) of the source detection could lead to biased scientific results. For example, the modeling of luminosity function at the faint end would be biased if one does not pay special care to the completeness near the detection threshold.

There are several ways to quantify the completeness of the source detection in terms of limiting depth (or limiting magnitude). In general, one can compare the source catalog, which is extracted from the observed image by the source finder, to a reference catalog built from the image with deeper depth. By comparing the source counts normalized by the observed area between two catalogs, one can calculate the completeness as the ratio of source counts given a limiting magnitude. For example, the source count–magnitude relation assuming a power law, with the power law index estimated from the reference catalog can be fitted to the source catalog to determine the completeness, which is defined as the ratio of the number counts of the source catalog to the best-fit power law model at given magnitude (e.g., Zenteno et al., 2011; Chiu et al., 2016). However, this approach ignores the cosmic variance of the source properties on the sky and is frequently affected by the systematics between the catalogs, such as the difference in the observed filter systems, detection algorithms or the observational conditions. Another example is to link the photometric uncertainties of the detected sources to the completeness as a function of the survey depth (Rykoff et al., 2015). Nevertheless, this method requires intensive modeling of the sky noise and photometry measurement; in addition, the large number of extracted sources in the overlapping region between the source and reference catalogs is needed in order to obtain the precise completeness estimates.

On the other hand, the most direct way to estimate the completeness of the source detection is to quantify the performance of the source finder by running the same detection algorithm on the objects simulated on the observed image. In this way, the completeness (and purity) of the source detection can be derived by comparing the extracted catalog to the catalog of simulated sources used as the input of the simulation. Simulating the sources on the observed image preserves any observational artifacts and sky noise while quantifying the performance of the source detection. Moreover, carrying out exactly the same algorithm to detect the simulated sources on the observed image provides an end-to-end verification and prevents any systematics arising from the catalog comparison. Several packages exist which can estimate the completeness using a similar methodology mentioned above, for example DAOPHOT (Stetson, 1987) or 2DPHOT (La Barbera et al., 2008). However, DAOPHOT is specifically designed for stellar-like objects and is not optimized for the extended sources such as galaxies. For the latter example, 2DPHOT requires intensive pre-modeling of the source properties (e.g., the morphology). In addition, the packages mentioned above are less user-friendly compared to the other image simulation software, such as `GalSim` (Rowe et al., 2015). A more straightforward way for source simulation is to simulate the sources based on the models with various properties, so that we have full control and are independent from the observed sources.

In this paper, we present the user-friendly software package `ComEst` – which is designed for estimating the completeness of the source detection on the observed image saved in the FITS format, by deriving the detection rate of the sources simulated with various properties assigned by the users. In addition, `ComEst` is designed for the source finder `SExtractor` (Bertin and Arnouts, 1996), which is widely used in the astronomy community for detecting the sources observed by the optical/NIR imaging. This paper is organized as follows. The methodology of `ComEst` is described in Section 2, while we demonstrate the usage of `ComEst` with an example and present the results in Section 3. We conclude in Section 4.

2. Methods

`ComEst` is developed as a completeness estimator for the `SExtractor` cataloging program for any given imaging of point sources or galaxies with well-calibrated photometric zeropoint (ZP). To capture any observational effects which are already present in the image, `ComEst` directly simulates sources with various properties (e.g., the sizes or fluxes) and runs `SExtractor` to carry out the source detection on the observed image. Therefore, by design, `ComEst` heavily relies on the source finder `SExtractor` and the image simulation toolkit `GalSim` (Rowe et al., 2015), which is used as the engine for simulating sources.

The workflow of `ComEst` is described as follows. For a given CCD image, `ComEst` first runs `SExtractor` to detect the observed sources and returns a set of standard outputs, especially the check-images. Then, `ComEst` removes the detected sources from the observed image and replaces them – in the same positions – by the background values estimated by `SExtractor`. In this way, `ComEst` creates the source-free image (SFI) with the observed noise properties only. Next, `ComEst` puts various sources simulated by `GalSim` (Rowe et al., 2015) on the SFI to create a set of simulated images. After creating a set of simulated images, `ComEst` re-runs `SExtractor` again to detect the simulated sources. Finally, `ComEst` derives the completeness as a function of source flux and image position, by comparing the `SExtractor` catalogs of the simulated sources and the true catalog used as the input to the simulation. The workflow is shown in Fig. 1. We describe the details of `ComEst` in this section.

2.1. Source detection and the source-free image (SFI)

The Source Extractor `SExtractor` (Bertin and Arnouts, 1996) is used as the source finder in `ComEst`. `SExtractor` is a program which detects the sources on the images and performs the photometric measurements of them. Since the first version in 1996, `SExtractor` has been recognized as the standard tool for detecting sources and estimating the photometry on the images in modern astronomy. For a given image and the `SExtractor` configuration, `SExtractor` builds the catalog of the sources detected on the images and returns a set of diagnostic outputs, such as the check-images. It is worth mentioning that the performance of the `SExtractor` depends on the configuration file provided by the user. We refer readers to the `SExtractor` manual[1] or (Holwerda, 2005) for more details about the optimized usage of `SExtractor`.

We stress that the observed image saved in the FITS format is the only input required by `ComEst`. The input image must contain the World Coordinate System (WCS) information in the header, which `ComEst` extracts to get information about the astrometry. `ComEst` first runs `SExtractor` on the input image to detect all the sources with fluxes above the threshold assigned in the configuration file, then `SExtractor` returns the output catalog and a set of check-images. The goal of the first run of `SExtractor` is to identify all the observed sources and estimate the background values on the image. Once the sources are

[1] http://www.astromatic.net/software/sextractor.