

The Blending ToolKit

Ismael Mendoza (@**imendoza**)
imendoza@umich.edu

Aug 11, 2022

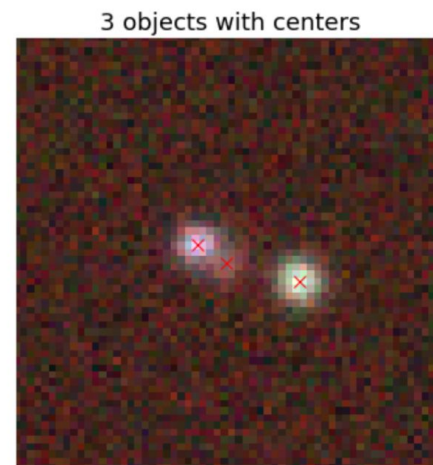
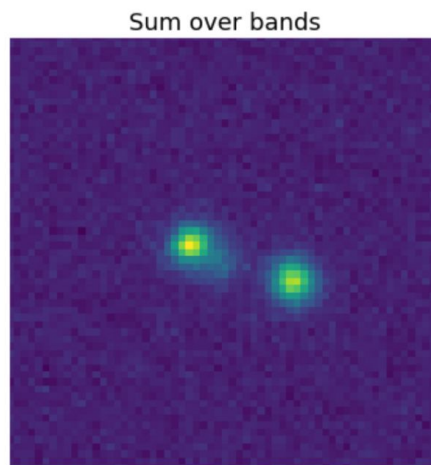
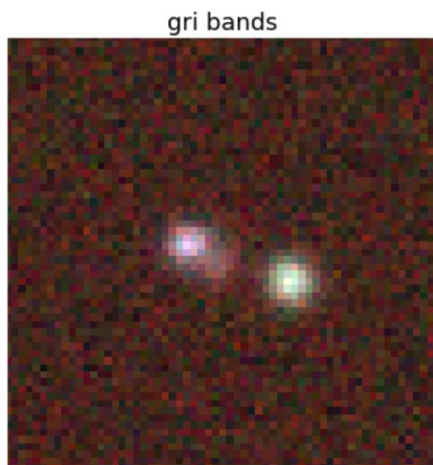
Outline

- What is BTK ?
- BTK feature spotlight
- BTK v1.0 and how to install
- BTK in PCW
- Other projects using BTK
- Summary

What is the BlendingToolKit (BTK)?



- **Goal:** Evaluating performance metrics for detection, deblending, and measurement algorithms when applied to images of blended objects.
- **Why?** Computation of performance metrics on identical datasets will provide a **standard benchmark** for algorithms and allow for their development.

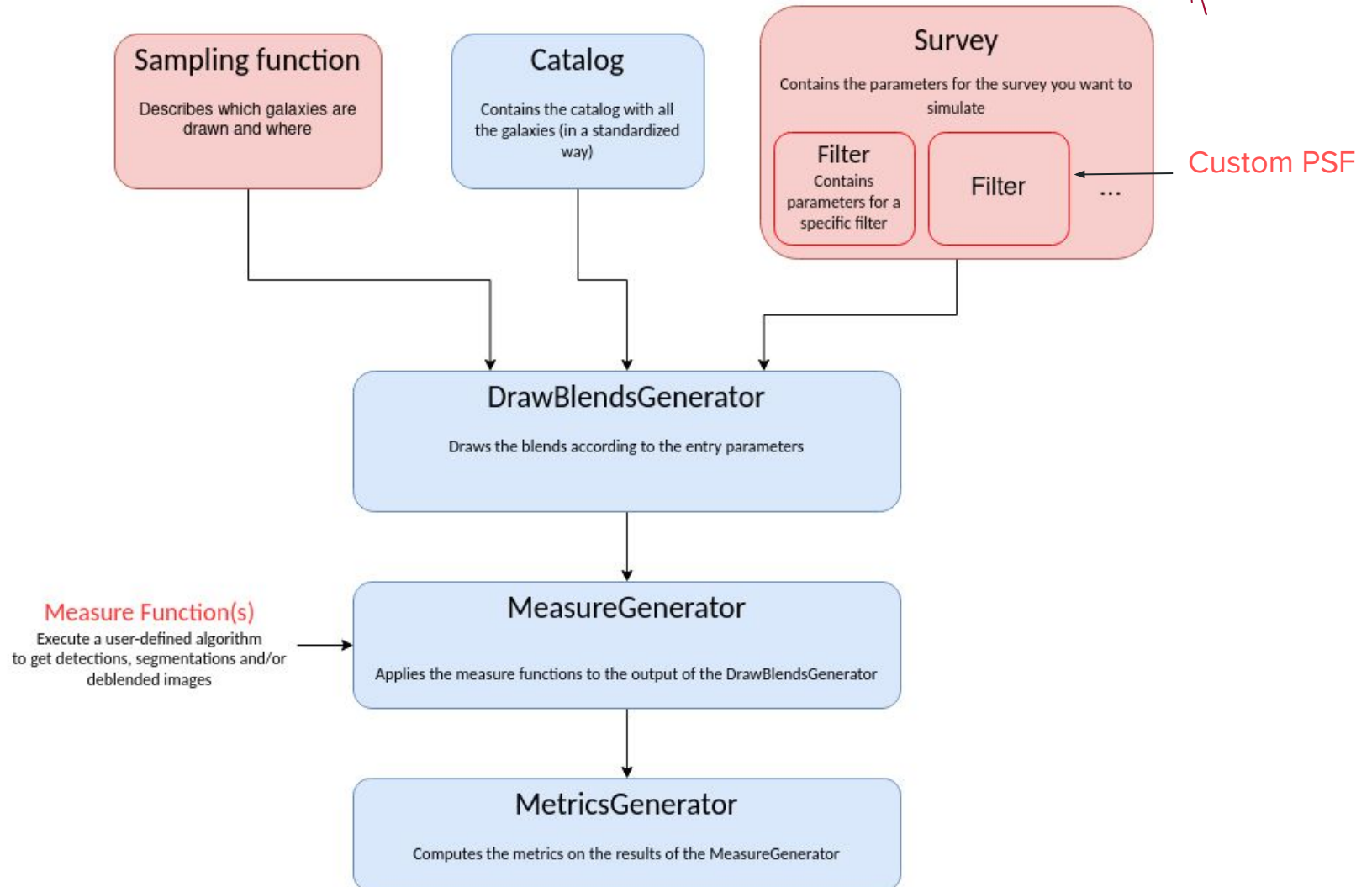


BTK Topical Team

BTK today wouldn't be possible without the contribution of many wonderful collaborators!

- Ismael Mendoza (co-lead)
- Thomas Sainrat (co-lead)
- Alexandre Boucaud
- Maxime Paillassa
- Hironao Miyatake
- Remy Joseph
- Biswajit Biswas
- Axel Guinot
- Andrii Torchilo

Red = User
customizable



Feature spotlight

- Can ingest both CATSIM and COSMOS **catalogs**.
- Two **galaxy models** to build blends: Bulge+Disk and Galsim COSMOS “Real” Galaxies
- Uses [galcheat](#), the standardized survey reference library, for observing conditions in postage stamps.
- Provides a **measurement interface** for running detection and deblending algorithms on the simulated datasets produced. Currently: SEP, Scarlet, and a basic peak finding algorithm is implemented.
- Contains **extensive library** of metrics divided in three groups:
 - *Detection*: Precision, recall, efficiency matrices, etc.
 - *Reconstruction*: Mean-squared residual, ellipticities, etc.
 - *Segmentation*: IoU

Announcement on latest version

- We have made the first official pre-release of v1.0 of BTK! 🎉
- The BTK Topical team is focusing on rough edges based on feedback from the community.
 - Let us know if you have any!

v1.0.0 pre-release !

Pre-release



ismael-mendoza released this 23 days ago

1.0.0a1

8624d08



Install and run v1.0

- BTK v1.0 is pip installable!
- You will be able to run all [tutorial notebooks](#), just need to download one of the sample catalogs from “data” folder of the repo.
- Or you can use your own catalog, see [documentation](#) for details.
- Binder [link](#) where you can run all notebooks and tutorials with no installation.



```
pip install blending-toolkit==1.0.0a2
```

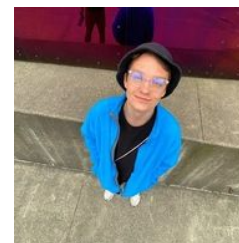

Getting started

- See previous [slides](#) for a deeper dive into BTK.
- Interested in using BTK for your project?
 - Take a look at our documentation:
<https://lsstdesc.org/BlendingToolKit/index.html>
and our tutorials:
<https://lsstdesc.org/BlendingToolKit/tutorials.html>
- Idea or feature you would like to see in BTK?
 - => [Open an issue](#) in github, we can brainstorm on how to integrate it!
- Have questions on BTK or how to use it? Ping us at [#blending_toolkit](#) in the LSSTC slack

PCW Posters using BTK



Andrii Torchylo



- Using BTK as dataset for NN galaxy detection in blends
- Reach out to him at **@Andrii Torchylo** if you have any questions!



Galaxy Detection with Neural Networks

Andrii Torchylo, undergraduate, Stanford University, KIPAC

torchylo@stanford.edu



Blending problems

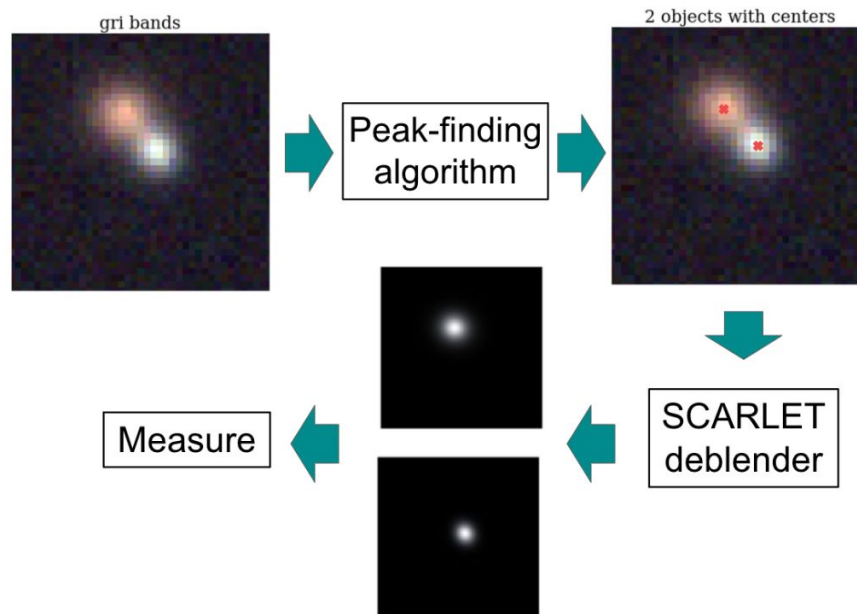
- Because of the increased depth of observations in the LSST, ~63% of galaxies will appear to be “blended” — their fluxes will partially overlap in an image (Sanchez et al. arXiv:2103.02078).
- Effects of galaxy blending can bias measurements of **position**, **flux**, **shear**, and **photometric redshifts** of galaxies, introducing significant uncertainty to our science results.

Current solutions

- **Detection:** typically a peak-finding algorithm (like **SourceExtractor**) is used to find the centroids of stars and galaxies.
- **Deblending:** current state-of-the-art approach is the **SCARLET** deblender (Melchior, Moolekamp et al. arXiv:1802.10157), which separates the flux using the input image and centroid detections.

Limitations

- Deblender performance will degrade significantly if incorrect number of detections is provided.
- Most detection algorithms use a single-band image for detecting sources, or compress multi-band information into one band.




Biswajit Biswas



- Using BTK galaxies as dataset for a Bayesian approach to deblending
- Reach out at **@Biswajit Biswas** if you have any questions! Also in-person at the PCW.
- Link to [poster](#)







MADNESS

MAP estimate with
Deep Neural Networks for Source separation

Biswajit Biswas, CNRS-IN2P3
Laboratoire APC, Université PSL



biswas@apc.in2p3.fr

Deblending:

In LSST, more than 63% objects are expected to be blended. Deblending is the inverse problem of separating these overlapping sources.

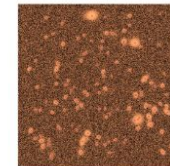


VAE for galaxy simulation:

We train a Variational AutoEncoder using Galsim COSMOS galaxies.

Preliminary Results:

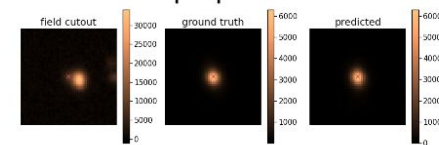
sinh(Input field)



Residuals



Example prediction:



Shuang Liang



- Using BTK for characterizing blending bias in a variety of settings.
- Reach out at **@Shuang Liang** if you have any questions! Also in-person at the PCW.



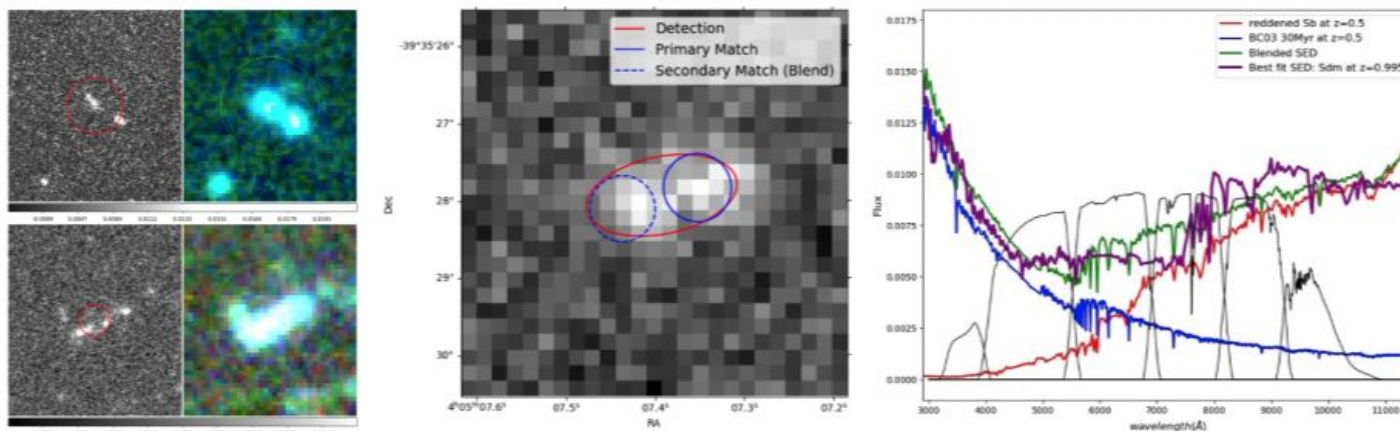
BLENDING IS SCARY

SHUANG LIANG, POSTDOC
STANFORD UNIVERSITY/KIPAC



BLENDING IMPACTS BOTH SHAPE AND PHOTO-Z MEASUREMENT

Figure 1: Left: Recognized and un-recognized blends. Middle: Blending affects shape measurement. Right: Blending impacts photo-z estimate (S. Schmidt, DESC meeting).



Two or more objects are “blended” when they are close to each other in projection. [1] find that about 58% of galaxies are blended at $i \sim 26$ in the Hyper Suprime-Cam (HSC) survey. [2] show that 14% of objects are “unrecognized blends” at the depth of LSST of $i \sim 27$, where multiple objects overlap so much as to be detected as one source. If not treated properly, they would contribute to 14% increase in shear noise for LSST. [3] also show that the impact of blends on photometric redshift (photo-z) is especially difficult due to different selections between the training sample and the target sample.

[1] Bosch et al. 2018 [2] Dawson et al. 2016. [3] R. Mandelbaum et al. 2017. [4] Laigle et al. 2015 [5] Melchior et al. 2018 [6] W Dong et al. 2020

Other ongoing projects



PZxBL: Developing blending metrics in the context of photo-z measurements



Alex Malz (@[aimalz](#)), Ismael Mendoza, others across PZ and BL working groups

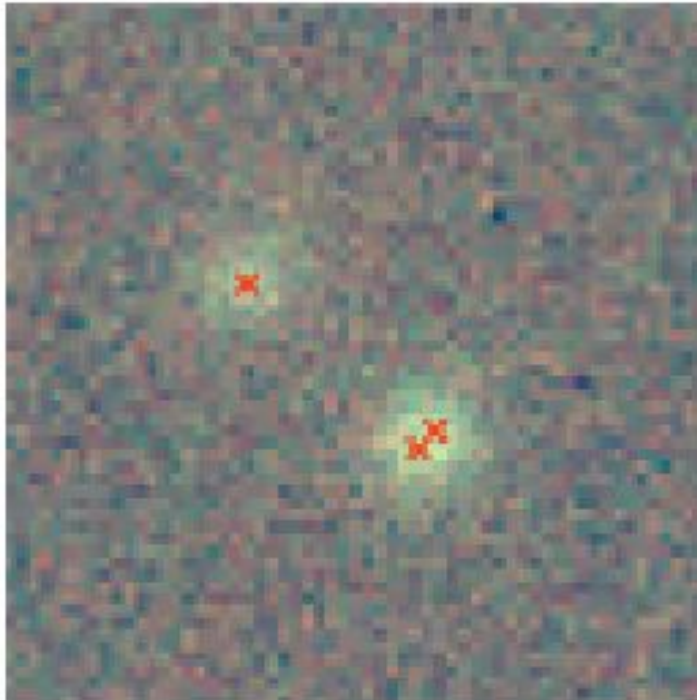
- **Goal:** Brainstorm and develop interpretable metrics on catalog membership mismatch.
 - E.g. unidentified blend fraction, recovered flux ratio between input and output catalog, estimated photo-z quality vs. input catalog true z
 - Either in BTK, [RAIL](#), or [DESCQA](#)
- Connect with RAIL
- Very preliminary [github package](#) started to collate PZ and BL efforts

Stress-testing Scarlet on dense galaxy cluster fields

Prakruth Adari (@Prakruth Adari)



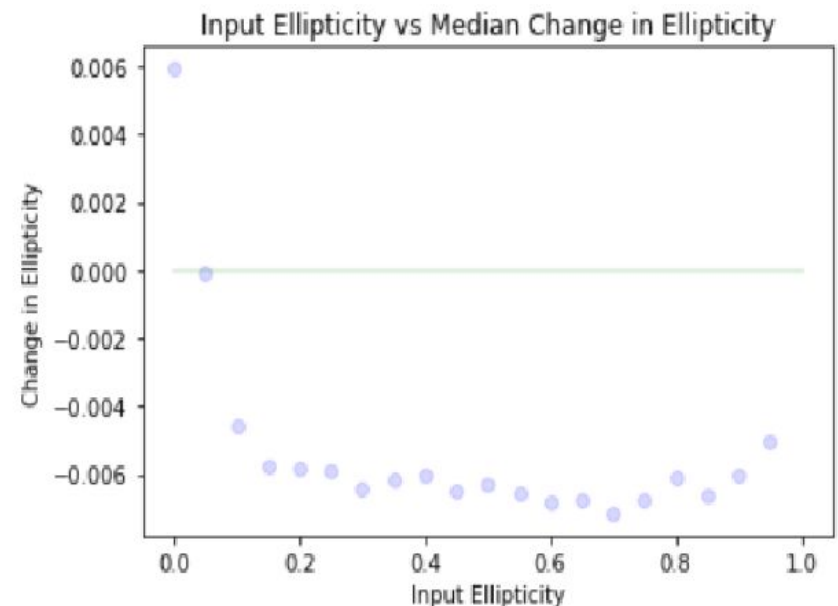
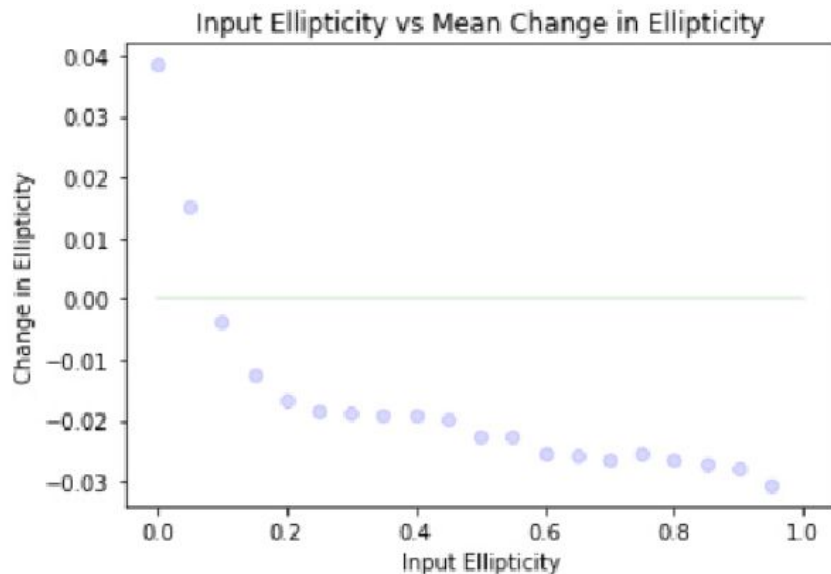
- Focused on impact of imperfect deblending on photo-z measurements
- (Future) Using BTK for galaxy cluster simulations



BTK to study blending shear

Nathan Craig, Ian Dell'Antonio (@idellant), Shenming Fu (@sfu)

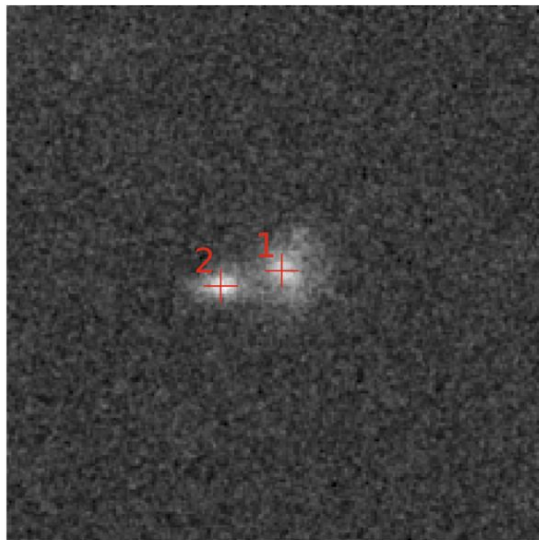
- **Goal:** Understanding the effect of blending on cosmic shear measurement
- Develop new BTK functionality: add option for constant shear, ingest DC2 catalogs, ingest properly distorted images to add non-linear shear.



HSC Unrecognized Blends

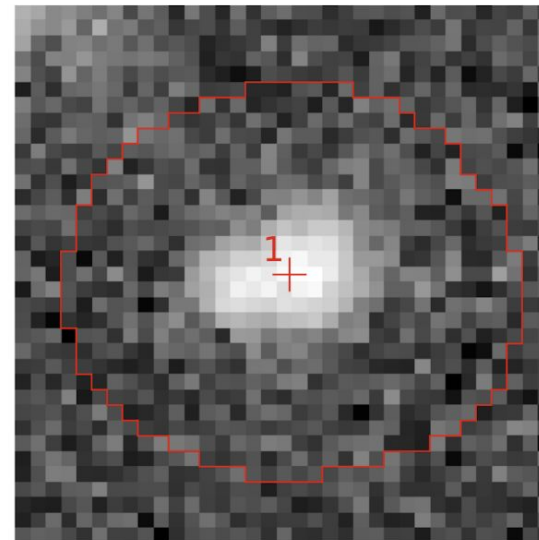
Maxime Paillassa (@mpaillasa), Hironao Miyatake, Atsushi Nishizawa, Suchetha Cooray

- Maxime implemented HSC-like simulations in BTK
- Maxime is developing a ambiguous blend classifier in HSC
- Source injecting BTK sims into HSC images



object 1
sep_min=0.705" (obj 2)
SN=30.83
MAG_AUTO=23.1857319

object 2
sep_min=0.705" (obj 1)
SN=22.98
MAG_AUTO=23.9714737



object 1
object_id=43158597349154500
SN=75.71
i_cmodel_mag=22.873243
i_blendedness_abs=0.0

Bayesian Light Source Separator (BLISS)

Ismael Mendoza (@imendoza), Derek Hansen, Jeffrey Regier, Camille Avestruz ++



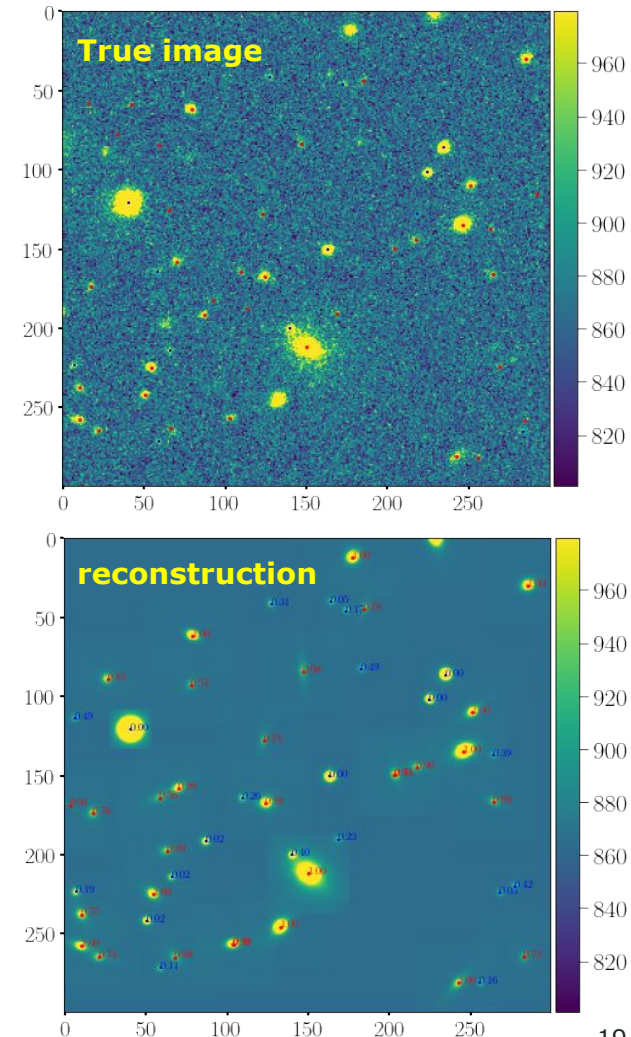
Summary: Fully probabilistic detection+deblending

Main Features:

- Output **posterior samples** of source's properties given an observed image (**probabilistic catalog**).
- Enables **probabilistic deblending** of stars + galaxies.
- Does **not require centroids** as additional input.
- Captures **uncertainties** on predicted source properties due to **noise+blending**.
- Performs **fast inference** on **large survey scenes**.

BTK:

- Will incorporate BLISS in BTK and compare with other deblenders like Scarlet.



Summary



Summary



- Blending ToolKit (BTK) can be used to **simulate galaxy blends** and **study the impact** of blending on various measurements in a reproducible and systematic way.
- BTK is quickly becoming a full-fledge python package and rapidly converging to a **stable v1.0 release**. You can install and try it today via pip!
- Anyone interested in blending is welcome to join us, feel free to join the conversation at **#blending_toolkit** in the LSSTC slack
- **Several ongoing projects** further developing and using BTK to better understand blending science.
- Find the BTK github [here](#)