VERA C. RUBIN OBSERVATORY

Update on the LSST Deblender

scarlet lite and more

Project and Community Workshop
August 10 2022

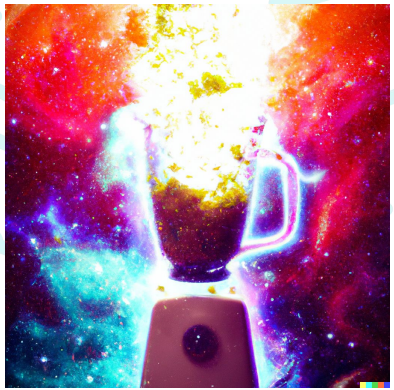
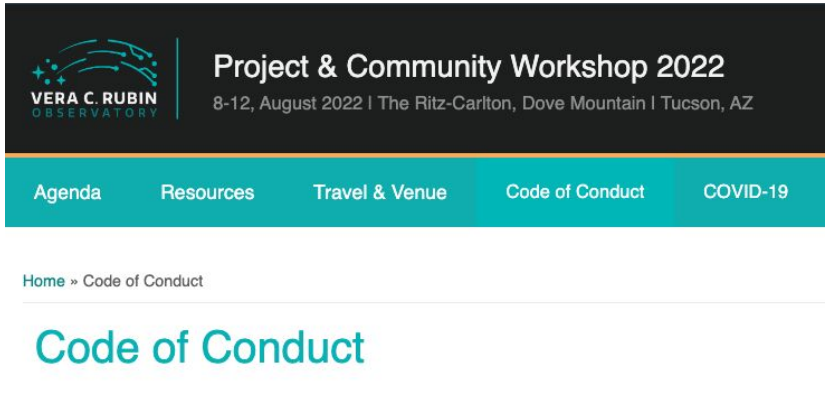


Image credit:
generated with AI using DALL-E

NSF AURA U.S. DEPARTMENT OF ENERGY SLAC CHARLES AND LISA SIMONYI FUND FOR ARTS AND SCIENCES LSST CORPORATION

Friendly reminders - CoC & Covid



Harassment and unprofessional conduct (including the use of offensive language) of any kind is not permitted at any time and should be reported.

Rubin Observatory adheres to the principles of kindness, trust, respect, diversity, and inclusiveness in order to provide a learning environment that produces rigor and excellence.

Check name-tags for these contact comfort level stickers.

Thank you for masking indoors!

Use the confidential email rubin2022-covid@lists.lsst.org to request a test, report your test results, or ask questions.

Reporting bullying, harassment, or aggression.

The Rubin 2022 Organizing Committee has appointed designated contacts:

- Ranpal Gill (rgill@lsst.org)
- Andrew Connolly (ajc@astro.washington.edu)
- Melissa Graham (mlg3k@uw.edu)

Contact via email, Slack, or the Community Forum.

Friendly reminders - virtual participation

Virtual participants should be muted when they're not speaking.

In-person participants should speak into the room microphone(s), or the chair should repeat all questions into the microphone, so that the virtual participants can hear what is said.

In the Rubin2022_PCW Slack Space, all participants can use the session's channel for Q&A and discussion.

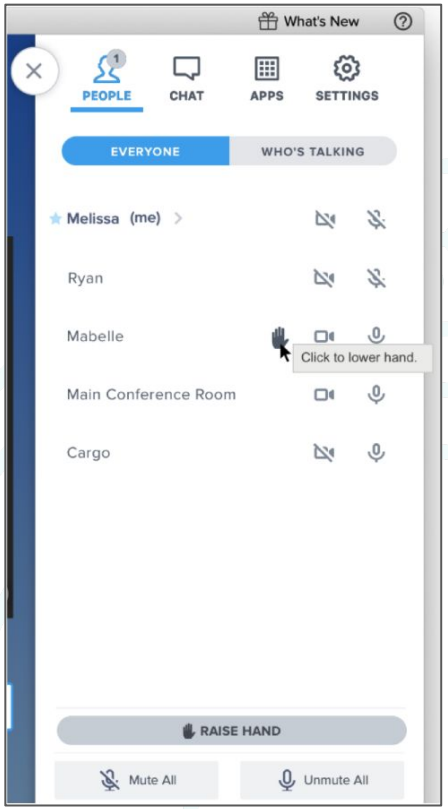
The channel name convention is, e.g.: #day1-mon-slot3a-intro-to-rubin

In BlueJeans, virtual participants should:

use the BlueJeans "raise hand" feature and wait for the moderator to call on you before speaking

or

use the BlueJeans chat functionality to ask questions or make comments.



The problem

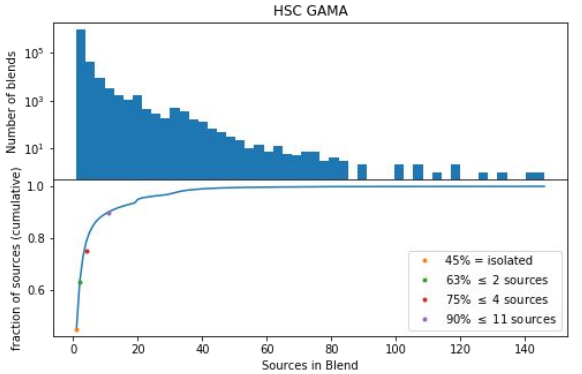
HSC COSMOS
UDEEP image

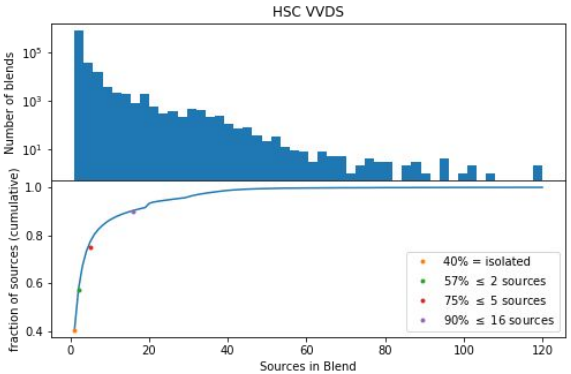
The problem

Image credit: AI generated with DALL-E

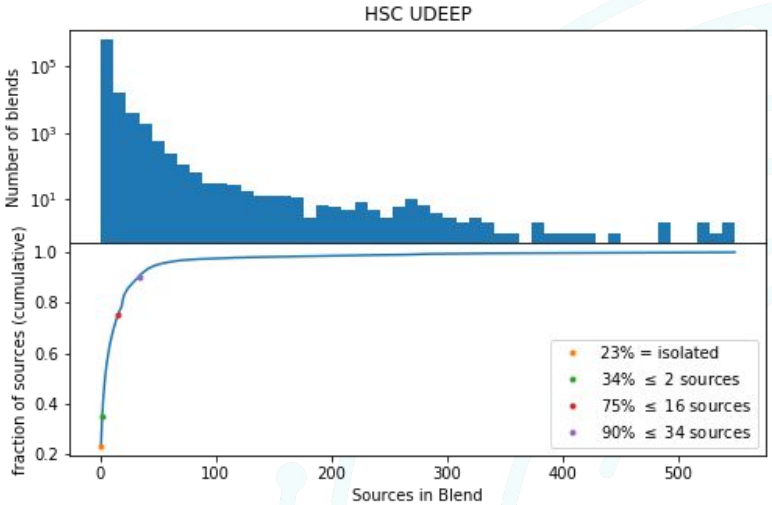
How worried should we be?

WIDE



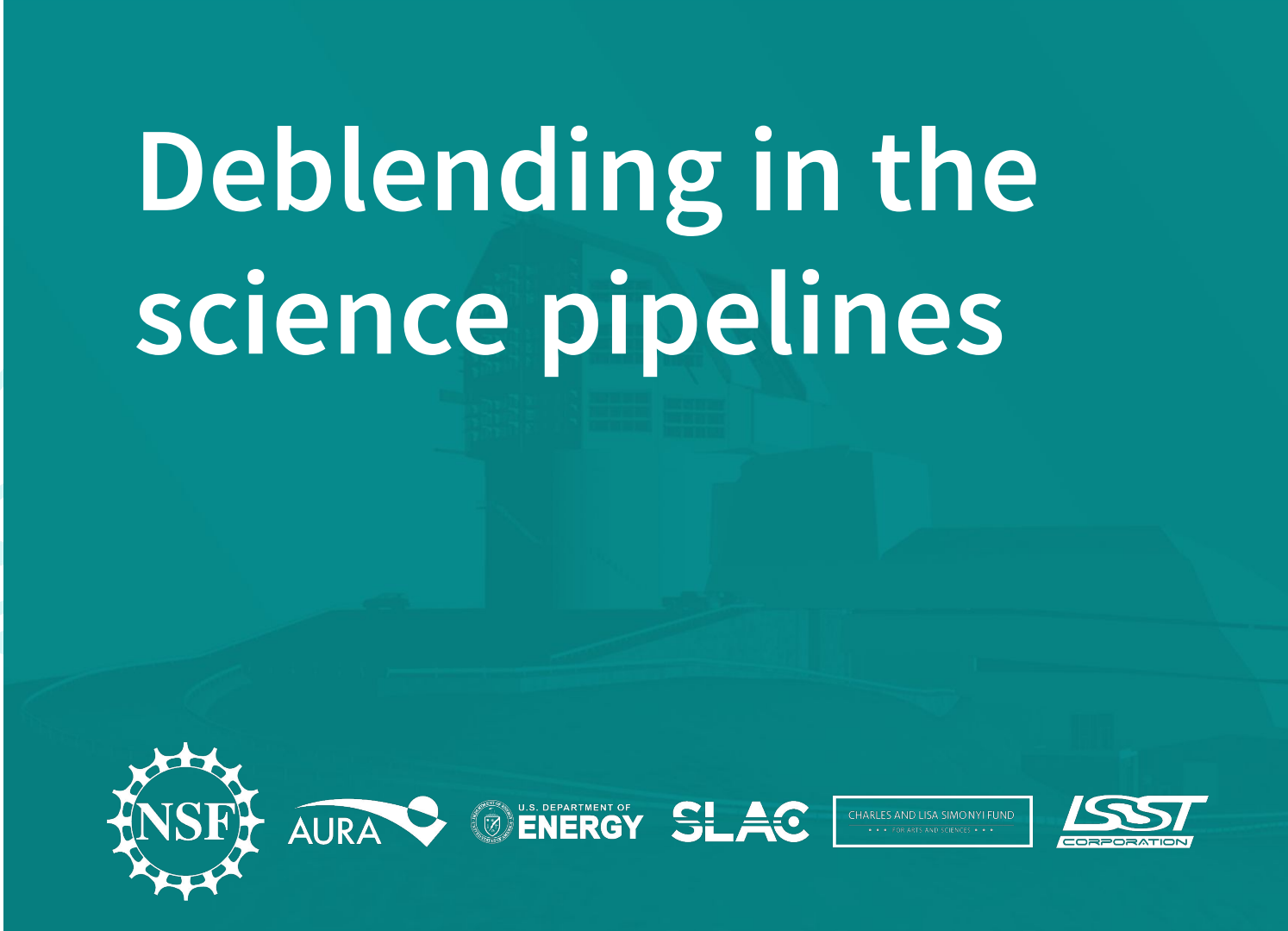


DEEP



VERA C. RUBIN OBSERVATORY

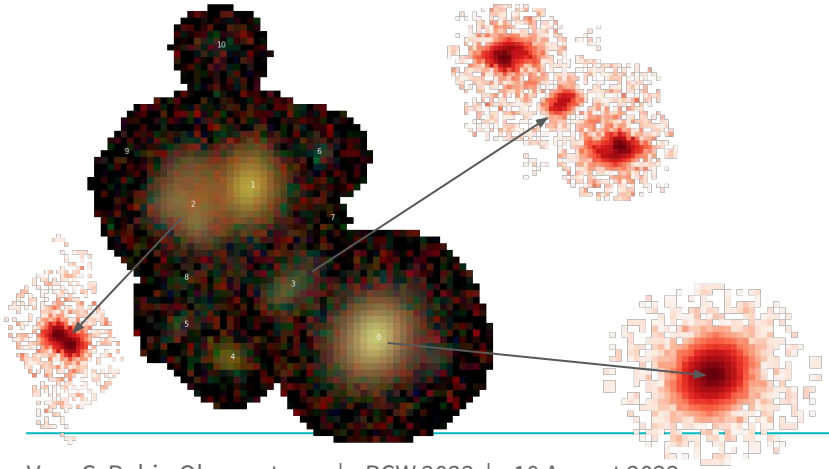
Deblending in the science pipelines



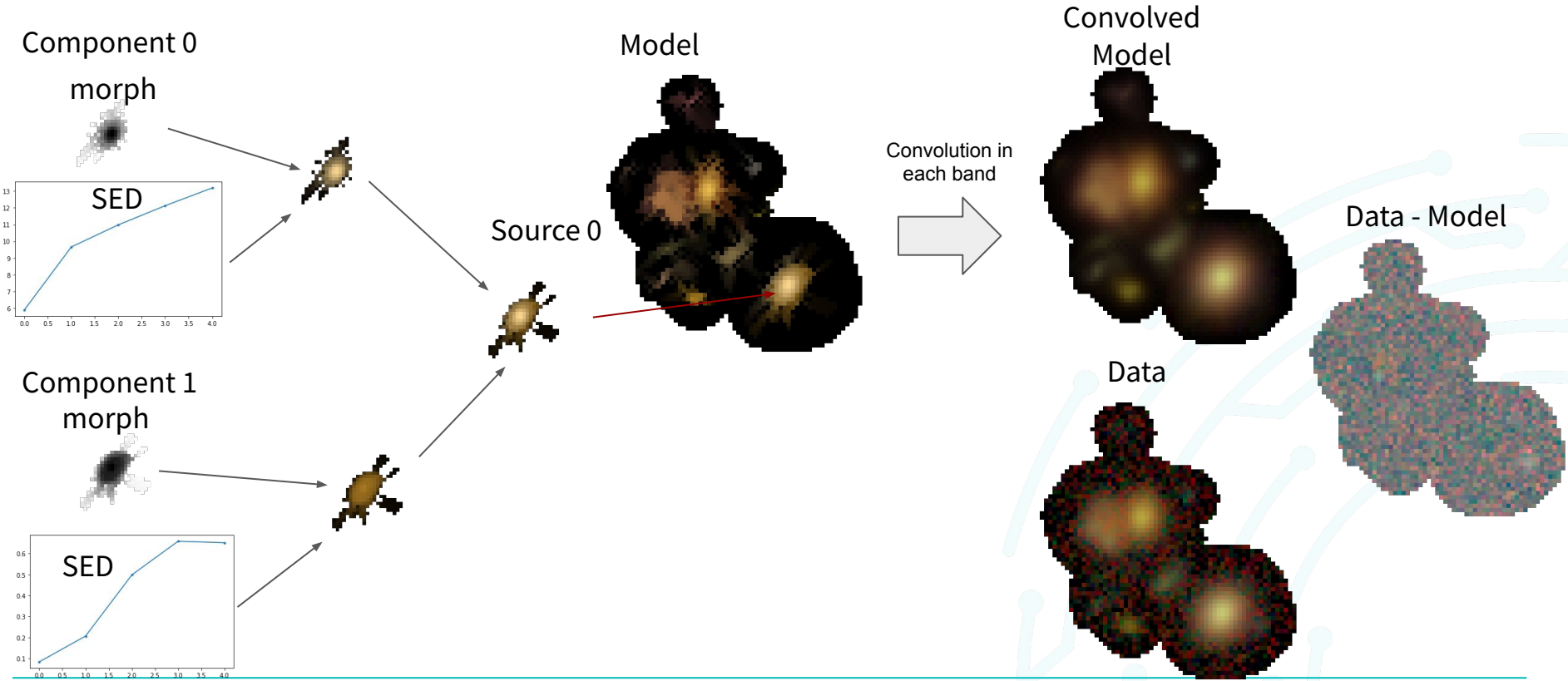
NSF AURA U.S. DEPARTMENT OF ENERGY SLAC CHARLES AND LISA SIMONYI FUND FOR ARTS AND SCIENCES LSST CORPORATION

Single-band deblender (meas_deblender)

- Runs on single visit catalogs (but not coadds)
- Based on the SDSS deblender
- Runtime ~ 5ms / source/band
- Creates symmetric template centered at each peak
- Fits faint sources to the PSF
- Re-apportions flux based on the symmetric templates

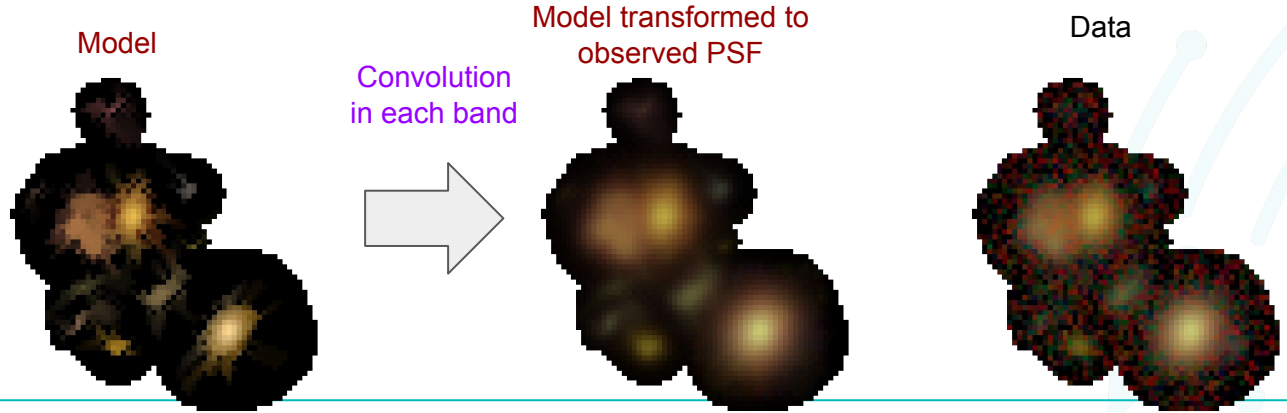


scarlet (and scarlet lite) models

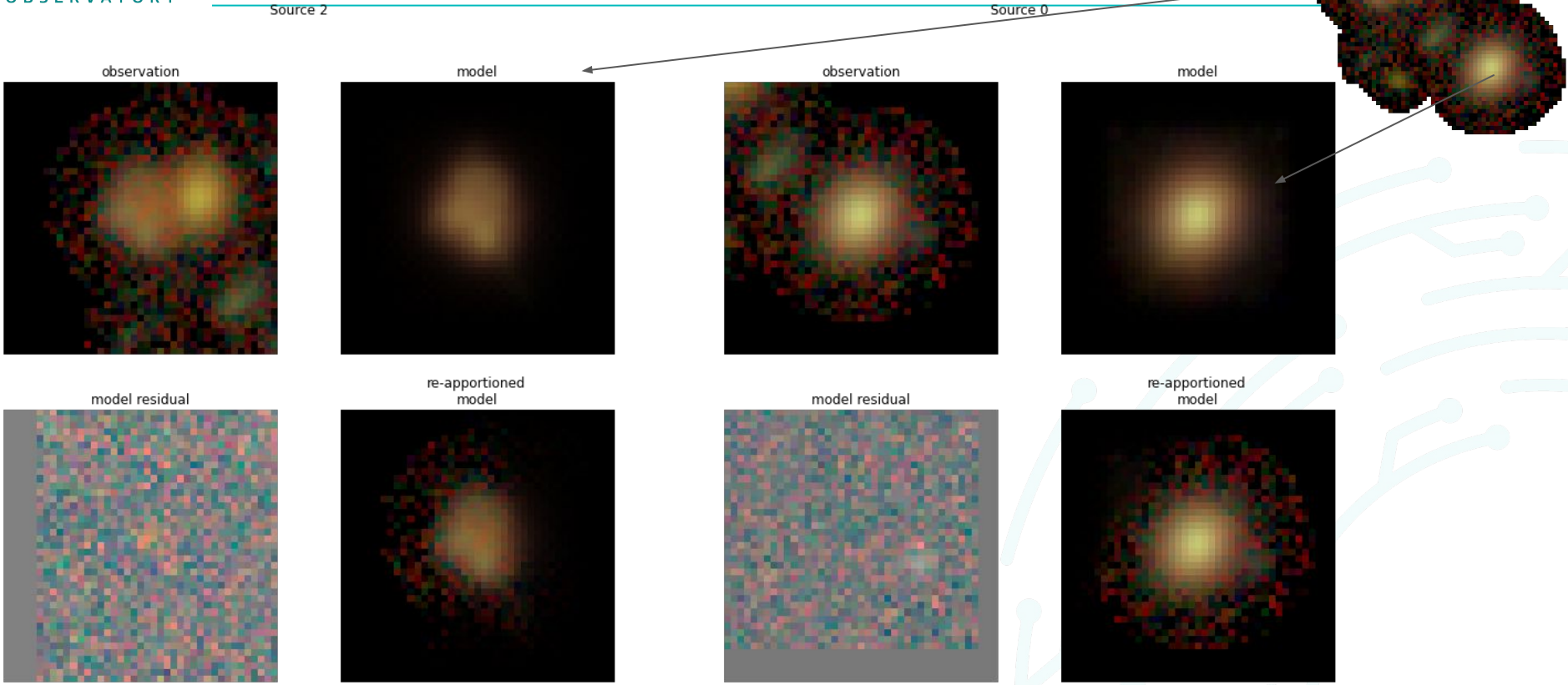


scarlet (and scarlet lite) models

- Basic algorithm
  1. The user defines an initial multiband model (basically the single-band template with monotonicity)
  2. The blend model exists in a frame with a narrow (but nyquist sampled) PSF
  3. The blend model is convolved to the observed PSF in each band
  4. AdaProx (Melchior et al. 2019) implementation of ADAM is used to apply constraints and priors to the models and calculate the gradient step
  5. The gradients are back-propagated to update the model
  6. Steps 2-5 are repeated until convergence

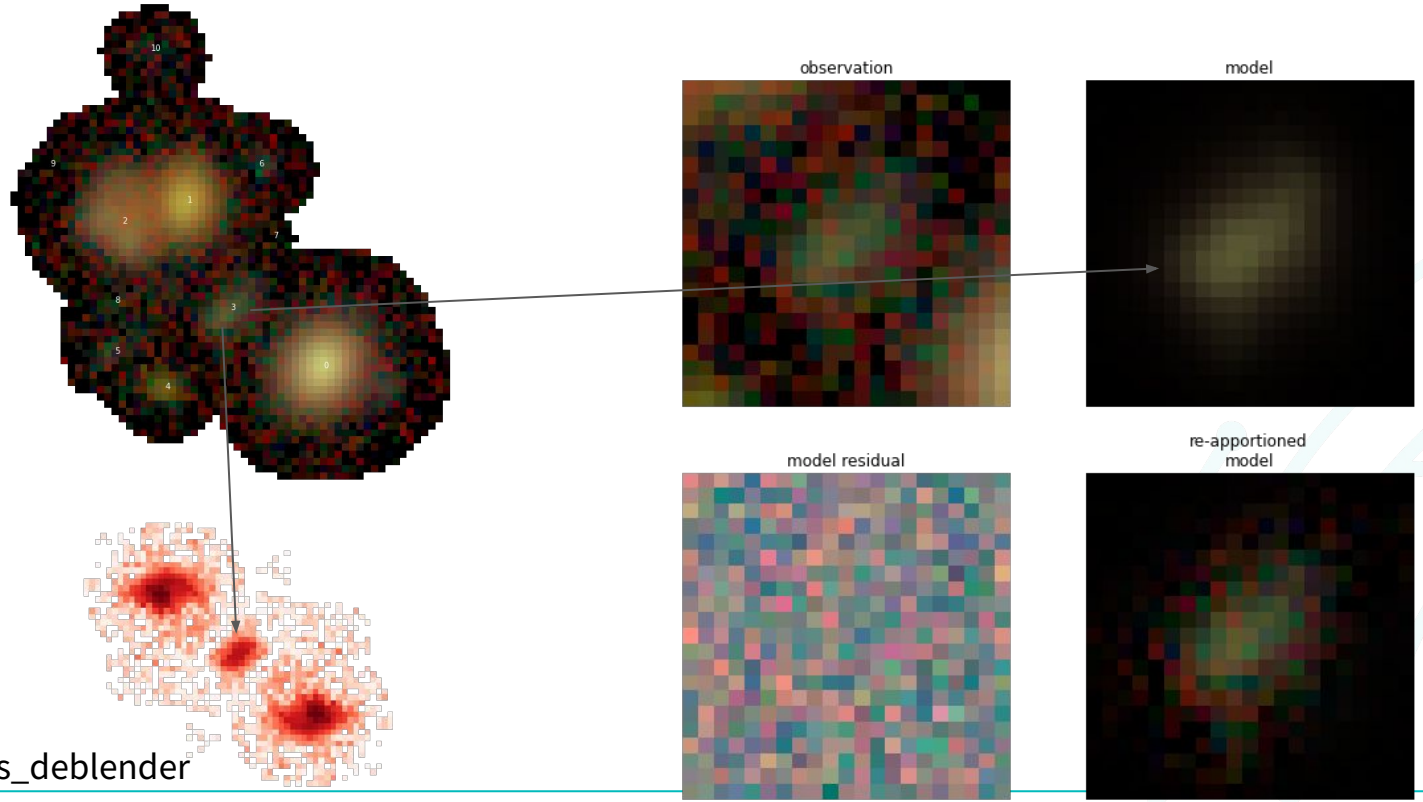


scarlet (and scarlet lite) models



scarlet (and scarlet lite) models

scarlet (lite) model



scarlet vs scarlet lite

scarlet

scarlet-lite

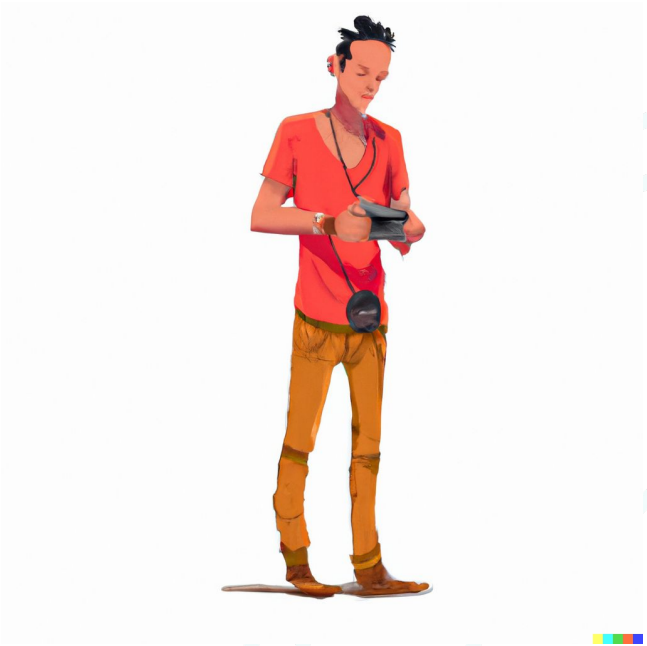
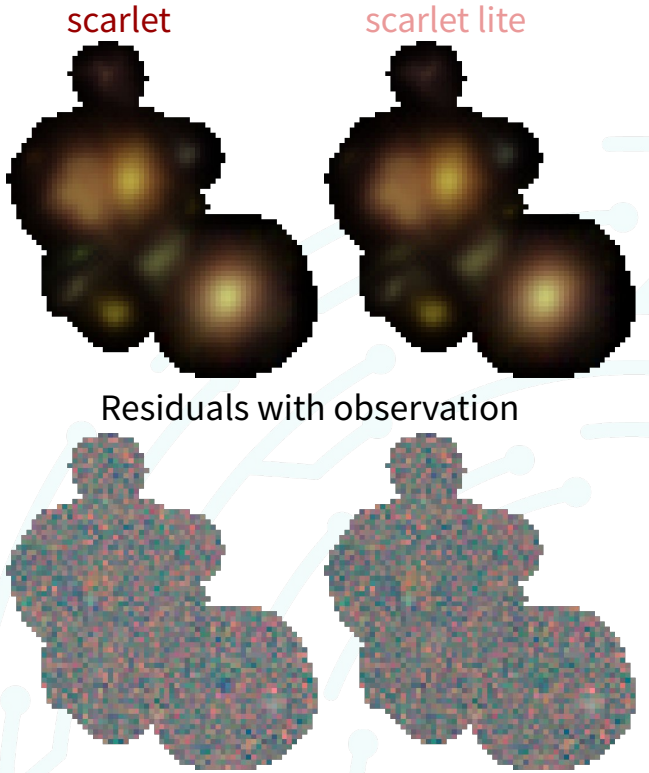


Image credits: AI generated with DALL-E

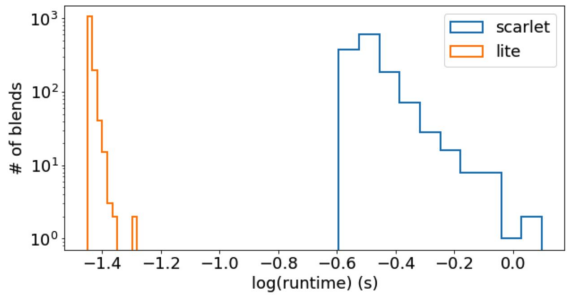
From scarlet to scarlet lite

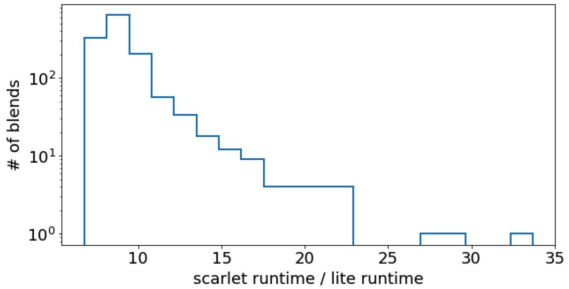
- What is scarlet lite?
  - An *algorithmically identical* version of scarlet *optimized for performance with Rubin data*
  - Changes from scarlet to scarlet-lite
    - **Analytic Gradients** (***removal of autograd*** for calculating gradients)
    - Significant changes in the class hierarchy for easier development
- Why a change was necessary?
  - scarlet was the runtime bottleneck for measurement algorithms
  - scarlet used a large memory footprint due to unnecessary copying operations in autograd
  - Removing autograd improved scarlet runtime significantly



scarlet lite

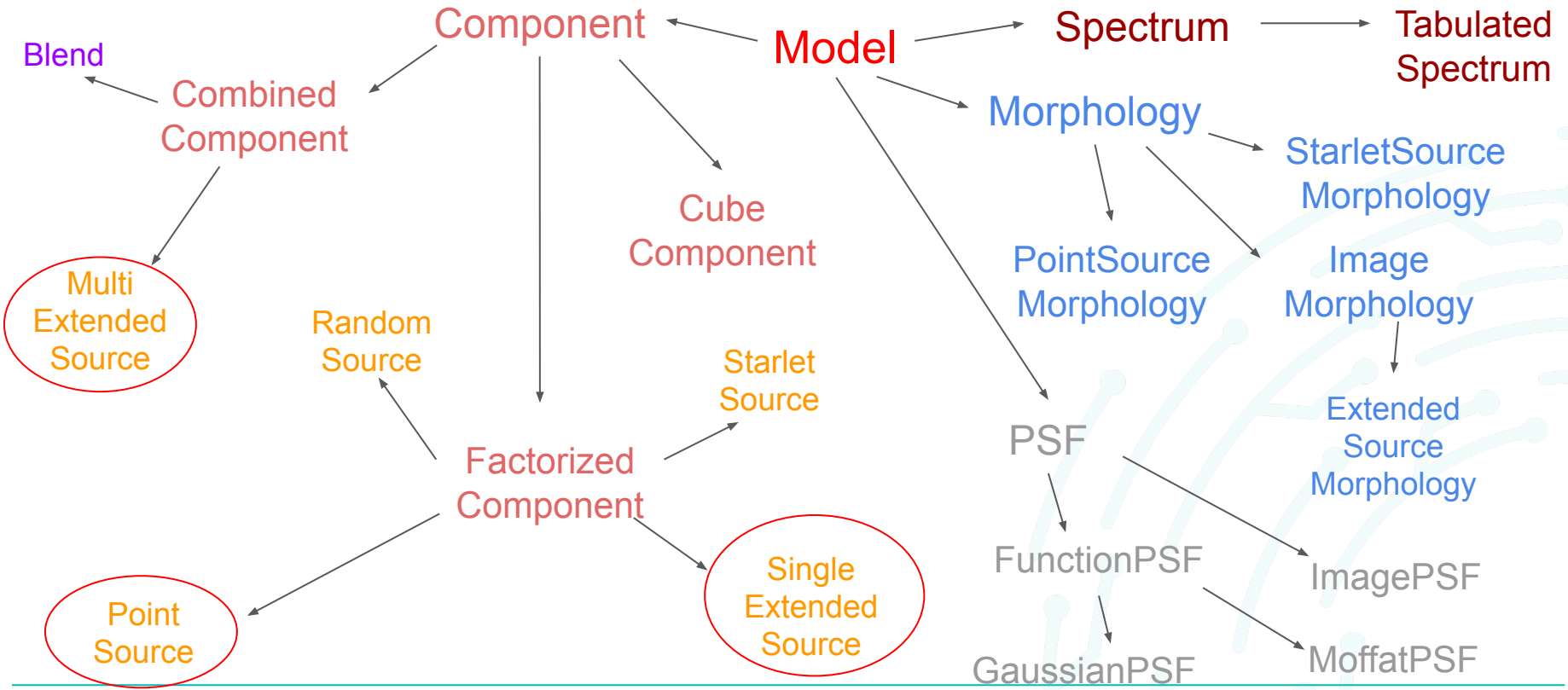
- Implements the scarlet optimization algorithm provided:
  - All of the images are WCS aligned to the same pixel grid
  - *Source components can be factorized into a 1D SED array and 2D morphology array
  - Source components contain a method to calculate the gradient
- scarlet vs scarlet lite reprocessing HSC RC2 nightly patch (tract 9813, patch 40) on blends with between 3-10 children:
  - max scarlet memory used: 526.5 MB
  - max scarlet lite memory used: 5.7 MB!





*This is not a strict requirement, and scarlet lite can be run with non-separable component models. But they aren't used in LSST so have not been implemented

scarlet class hierarchy



scarlet lite class hierarchy

LiteBlend
- sources
- observation

→ LiteSource
- components

→ LiteFactorizedComponent
- SED
- Morphology
- center

scarlet lite outputs

- Compressed scarlet model outputs persisted as of w_2022_26
  - Note: implemented ***after*** DP 0.2 freeze 😔
- Models can be converted into footprints, used to re-distribute observed flux (default), or used as scarlet ***LiteBlend*** models
- Docs in peer review now

Output (patch, rc2_subset)	**Previous disk space**	**Compressed disk space**
Deblender catalog	1.5G	112M
Measurement catalog	1.9G	787M
Forced phot catalog	2.0G	345M

Loading catalogs with footprints

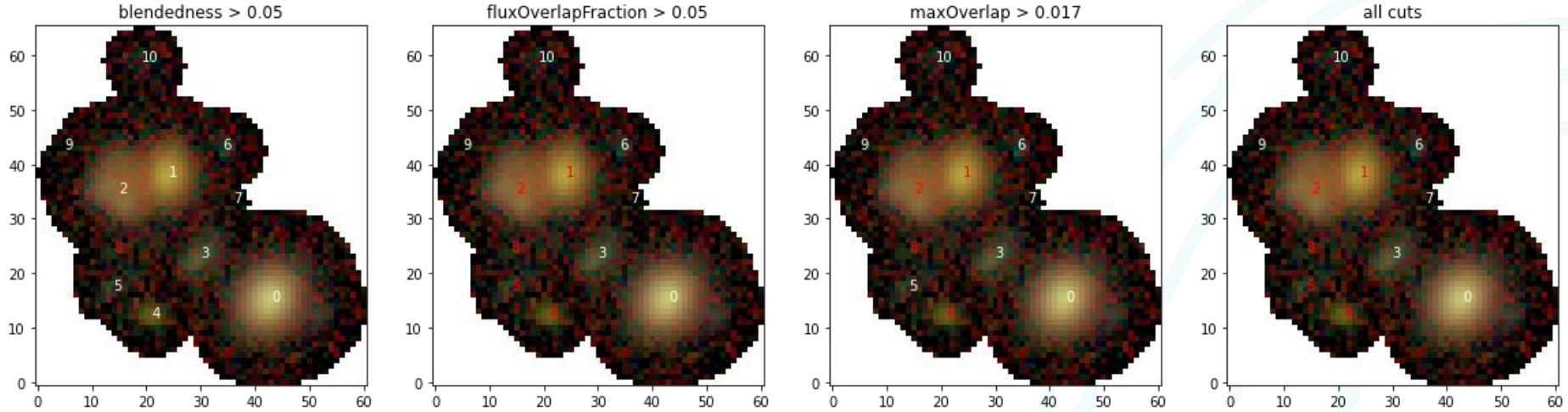
```
from lsst.daf.butler import Butler
# Initialize the butler
butler = Butler("/repo/main", skymap="hsc_rings_v1", collections=collections)
# Load the deblender output catalog
catalog = butler.get("deepCoadd_deblendedCatalog", tract=tract, patch=patch)

# Load the scarlet models for the catalog
modelData = butler.get("deepCoadd_scarletModelData", tract=tract, patch=patch)
# Load the PSF model
psfModel = butler.get("deepCoadd_calexp.psf", tract=tract, patch=patch,
band=band)
# Update the footprints for all of the deblended sources.
modelData.updateCatalogFootprints(catalog, band=band, psfModel=psfModel,
removeScarletData=True)
```

Loading scarlet models for a blend

```
from lsst.daf.butler import Butler
import lsst.meas.extensions.scarlet as mes

# Initialize the butler
butler = Butler("/repo/main", skymap=skymap, collections=collections)
# Load the deblender output catalog
catalog = butler.get("deepCoadd_deblendedCatalog", tract=tract, patch=patch)
# Load the scarlet models for the catalog
modelData = butler.get("deepCoadd_scarletModelData", tract=tract, patch=patch)

# Select the first record with exactly 5 deblended children
parent = catalog[catalog["deblend_nChild"]== 5][0]
# Load the PSF for the observation in each band
observedPsfs = []
for band in modelData.bands:
    observedPsfs.append(butler.get("deepCoadd_calexp.psf", tract=tract, patch=patch, band=band))

# Extract the scarlet LiteBlend from the ScarletModelData
blend = mes.io.multibandDataToScarlet(
    modelData=modelData,
    blendId=parent.getId(),
    observedPsfs=observedPsfs
)
```

Blending metrics

- maxOverlap: Maximum value of a pixel that overlaps with a neighbor
- fluxOverlapFraction: neighbor flux / source flux over the sources footprint
- blendedness/purity: $\beta = 1 - \frac{\mathbf{S}_k \cdot \mathbf{S}_k}{\mathbf{S}_{\text{all}} \cdot \mathbf{S}_k}$
- Others can be added



Blending metrics

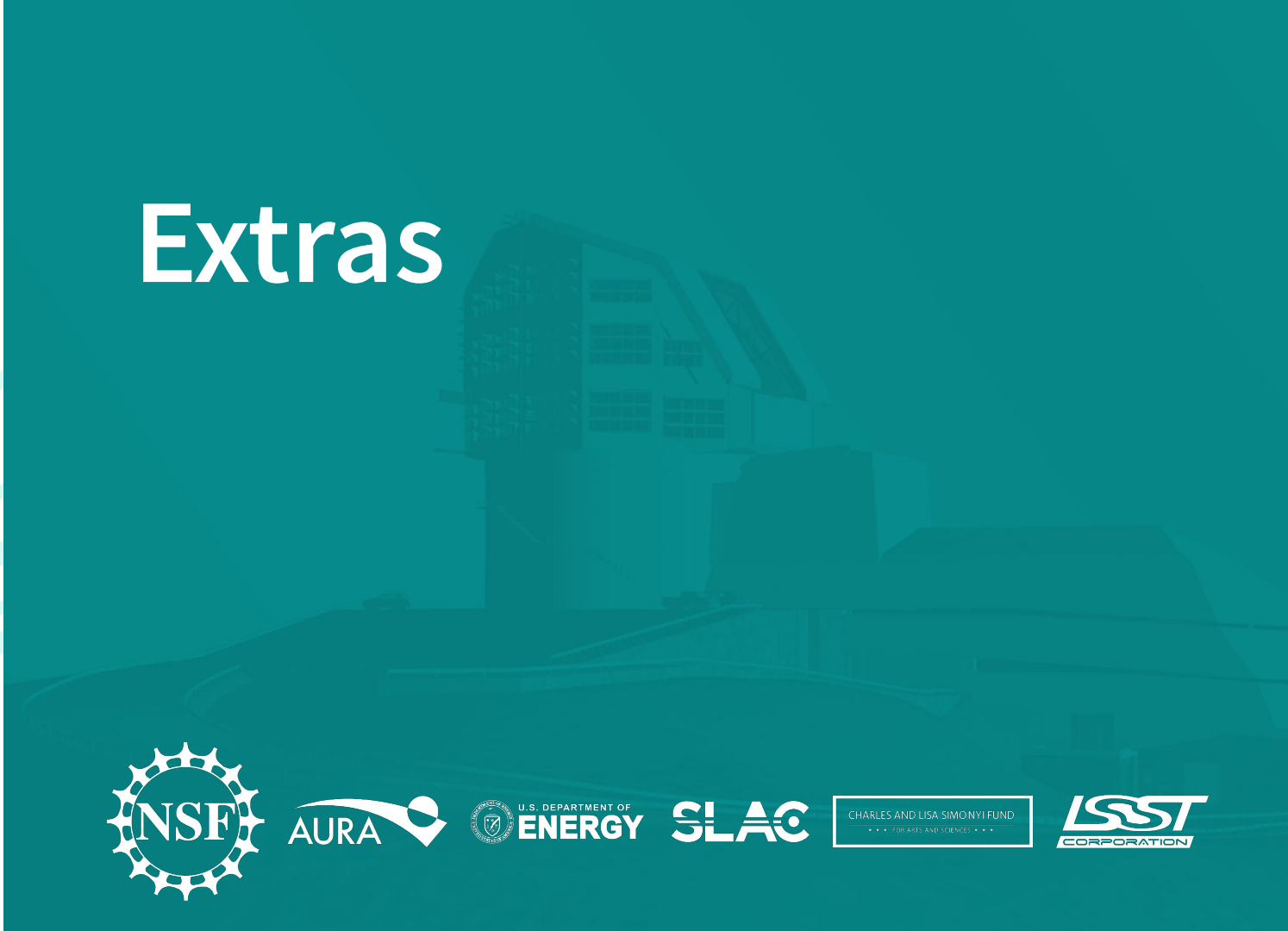
- maxOverlap: Maximum value of a pixel that overlaps with a neighbor
- fluxOverlapFraction: neighbor flux / source flux over the sources footprint
- blendedness/purity: $\beta = 1 - \frac{S_k \cdot S_k}{S_{all} \cdot S_k}$
- Others can be added
- Example: HSC RC2 DM nightly patch:
  - 1,297,130 total sources
  - 13.8% overlap above the noise in r-band (maxOverlap)
  - 7.8% overlap with at least 5% of the total flux (fluxOverlapRatio)
  - 4.5% have blendedness > 0.05 (blendedness)
  - 13.8% meet at least one of the above criteria

Future work

- Test subtracting scarlet models instead of using the noise replacer for measurement (similar to DES)
- Test fitting only the high SNR regions with scarlet and fitting the wings of stars and galaxies with something else (Gaussian mixtures?)
- Create a new stand alone scarlet lite package
- Improve initialization around bright objects and crowded fields

VERA C. RUBIN OBSERVATORY

Extras



NSF AURA U.S. DEPARTMENT OF ENERGY SLAC CHARLES AND LISA SIMONYI FUND FOR ARTS AND SCIENCES LSST CORPORATION

Crowded field initialization problem

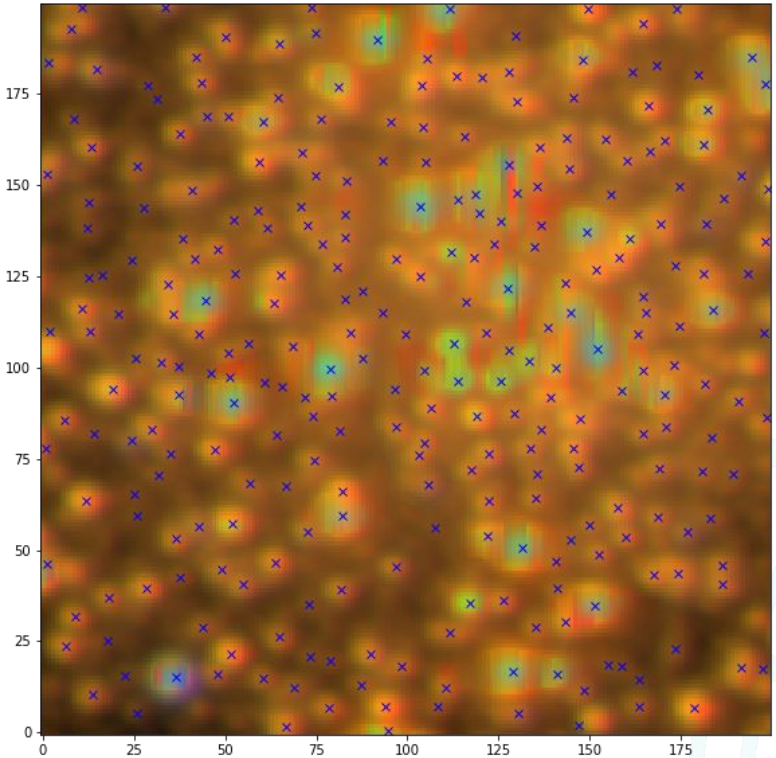
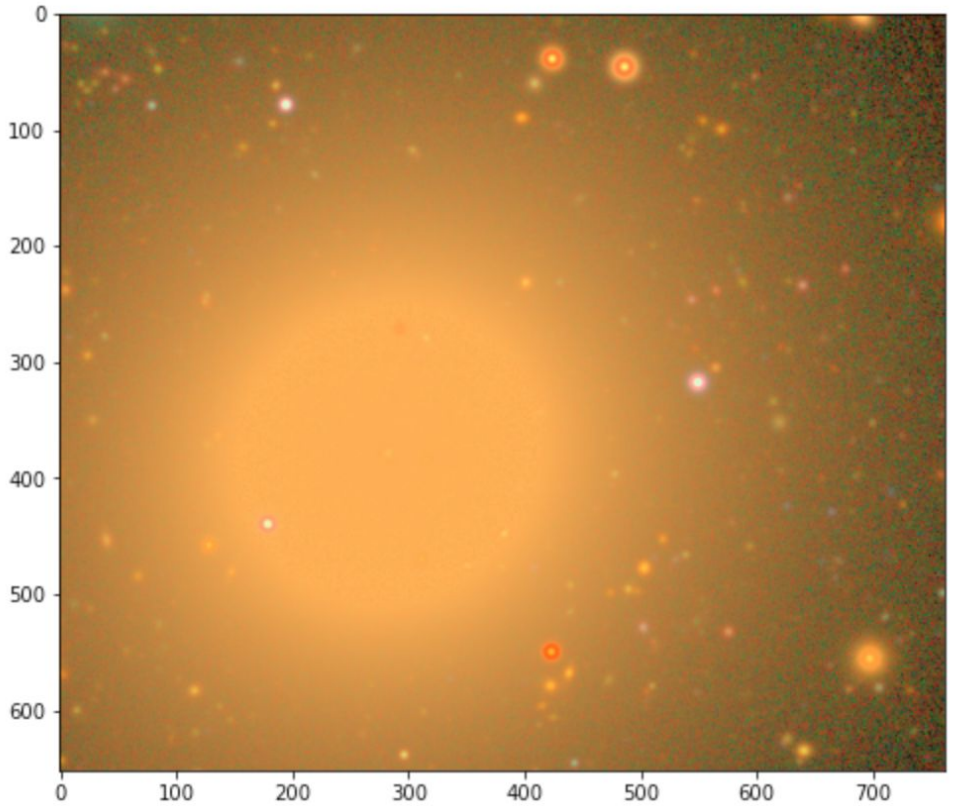


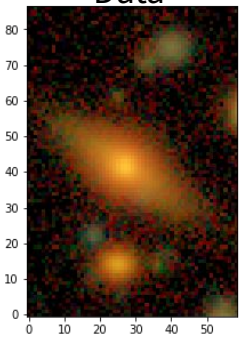
image: A small region in NGC 6569

Bright sources

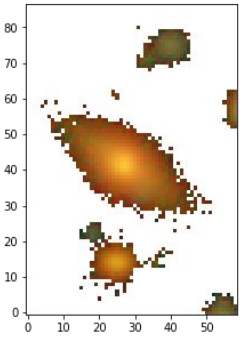


Fitting only the above noise regions

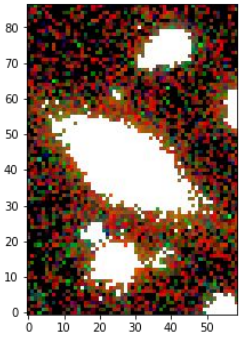
Data



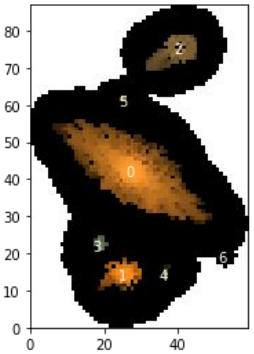
Data above noise



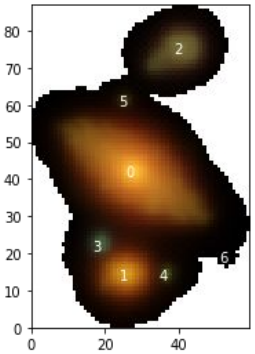
Flux below noise



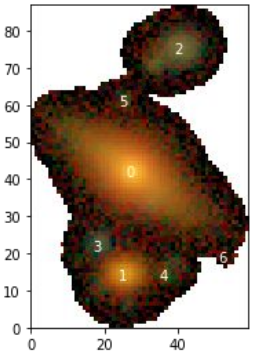
Model



Model Rendered



Observation



Residual