# **SSTCam pointing correction software**

### Overview and current status

Evert Rol - February 2024





## Overview

- Live pointing corrections using background stars in the field-of-view
  - Use slow-mode / variance signal
  - Also calculate an overall correction (e.g., once per observation)
- Generic setup; potentially applicable to other cameras and telescopes
  - Test with / use various data sources



# Implementation

- Python package
  - CTAPipe as main dependency
- Source detection & fitting home-made

  - For example, existing source detection s/w wouldn't handle module gaps Corrections (fitting) done in camera frame
- Modular
  - Change / add algorithms, functionality, instruments Select / configure through configuration object / file

https://gitlab.cta-observatory.org/cta-array-elements/sst/camera/server/sstcam-pointing



# **Caveats & to-dos**

- The pointing should already be known relatively accurately. Roughly within one pixel.
- Handle close / overlapping stars
- Handle bad PSF
- Interfaces lacksquare
  - Control / adjust settings
  - Data acquisition  $\bullet$
  - Results output



#### **Proof of concept / example**



Two example interfaces. Left SSTCam simulations, right HESS data. For the HESS data, variance data was used, and simulated to be transferred at 10 images per second. The examples present a web-ui, showing continuous updates of the camera frame and fit results. Fits to positional offsets are calculated over a rolling average of 1-second stacked frames in a 10-second window. Additional fits are performed over longer time frames. Only two parameters (x, y offset) are fit for. Additional parameters would be rotation or (x, y) offsets dependent on altitude.

Detections are shown with open circles, while the green filled circles indicate positions of catalogue stars. This work is very preliminary, and bugs (such as the low chi-squared value) show, as well as the lack of detections for relevant sources. This interface was mainly used to test whether the fitting procedure can handle a rate of 10 images / sec, and send the result onwards, without lagging behind.





0.00828 ° -0.05427 °

0.058408; reduced = 0.002920

6.29 mm
£ 16.29 mm
0.00 mm
0.00 mm
0.00 mm
0.00 mm
0.00 mm
0.00 mm
0.00 mm
0.00 mm

0.582073; reduced = 0.002670

-34.56 ± 4.93 mm

