# Aperture Calculation

Spitzer Space Telescope: Star Imaging Data

*Hannah Diamond-Lowe, Zakir Gowani*

# Project Dataset and Goals

- *Spitzer Space Telescope* data of a star-plus-planet system
- Problem: background noise disrupts star analysis, and finding optimal aperture/mask by hand is tedious
- Solution: automate process using image processing and signal-to-noise ratio calculations
- Overall goal: find intelligent time-dependent apertures that conserves star signal and rejects noise

# Prototype: Computational Methods

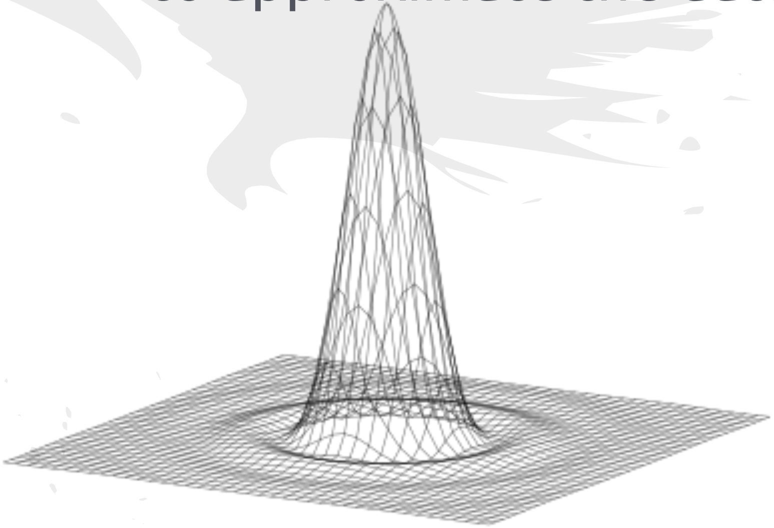
- Images processed into 32x32 numpy arrays
- Using midpoint circle algorithm, draw circular masks for different radii
- Calculate weighted ratios of signal flux to noise flux and compare to ratios of labeled data
- Output optimal mask radius and save image to file (currently png, in the future will opt for the original FITS format)

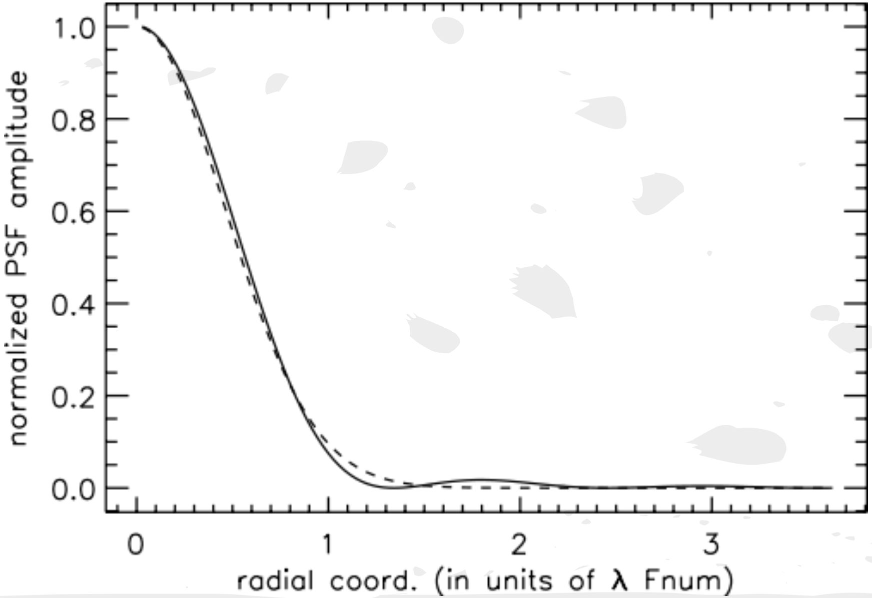
21.0 22.0 5.0 29.0 22.0 15.0 -0.0 14.0 1.0 13.0 -8.0 17.0 26.0 5.0 22.0 2.0 15.0 16.0 17.0 13.0 0.0 -5.0 25.0 27.0 12.0 14.0 13.0 -13.0 -6.0 24.
16.0 12.0 5.0 13.0 8.0 5.0 15.0 17.0 14.0 8.0 13.0 14.0 21.0 23.0 10.0 9.0 10.0 11.0 32.0 28.0 17.0 20.0 17.0 6.0 -2.0 17.0 10.0 15.0 -2.0 2.0
38.0 -11.0 20.0 23.0 11.0 17.0 -12.0 17.0 4.0 20.0 15.0 31.0 -9.0 8.0 18.0 12.0 6.0 -3.0 19.0 6.0 18.0 5.0 11.0 18.0 19.0 19.0 27.0 10.0 16.0 1
12.0 18.0 2.0 6.0 8.0 5.0 22.0 13.0 1.0 20.0 28.0 6.0 19.0 10.0 23.0 15.0 22.0 16.0 12.0 31.0 12.0 19.0 -6.0 -1399.0 12.0 10.0 25.0 23.0 20.0 1
26.0 14.0 18.0 33.0 6.0 3.0 24.0 17.0 29.0 21.0 30.0 25.0 18.0 17.0 23.0 17.0 15.0 6.0 30.0 29.0 26.0 11.0 9.0 7.0 27.0 8.0 24.0 23.0 17.0 17.0
28.0 9.0 -11.0 28.0 6.0 -3.0 19.0 9.0 26.0 28.0 13.0 17.0 11.0 13.0 26.0 20.0 20.0 8.0 33.0 10.0 31.0 16.0 11.0 17.0 -4.0 24.0 18.0 14.0 20.0 -
20.0 16.0 5.0 24.0 24.0 13.0 17.0 12.0 36.0 25.0 -7.0 35.0 38.0 20.0 38.0 34.0 16.0 27.0 31.0 33.0 14.0 37.0 16.0 17.0 25.0 18.0 16.0 7.0 9.0 3
19.0 17.0 55.0 27.0 27.0 16.0 16.0 19.0 24.0 68.0 21.0 25.0 18.0 44.0 49.0 33.0 31.0 14.0 33.0 26.0 21.0 28.0 19.0 16.0 14.0 24.0 11.0 26.0 20.0
27.0 22.0 26.0 13.0 15.0 20.0 17.0 11.0 25.0 26.0 15.0 29.0 41.0 73.0 79.0 54.0 35.0 8.0 37.0 17.0 25.0 18.0 13.0 10.0 26.0 17.0 16.0 12.0 4.0
16.0 13.0 13.0 28.0 15.0 11.0 24.0 38.0 46.0 33.0 34.0 31.0 43.0 70.0 96.0 53.0 36.0 32.0 25.0 11.0 40.0 23.0 37.0 44.0 12.0 11.0 21.0 12.0 43.
12.0 9.0 -4.0 15.0 28.0 40.0 15.0 14.0 14.0 15.0 23.0 53.0 69.0 454.0 756.0 136.0 51.0 42.0 34.0 30.0 18.0 2.0 9.0 6.0 18.0 27.0 23.0 6.0 26.0
12.0 9.0 11.0 37.0 8.0 10.0 22.0 23.0 19.0 27.0 32.0 67.0 135.0 1404.0 2788.0 550.0 108.0 60.0 23.0 37.0 34.0 12.0 17.0 20.0 22.0 12.0 15.0 4.0
14.0 -2.0 5.0 17.0 -19.0 17.0 19.0 27.0 25.0 34.0 31.0 49.0 530.0 4357.0 5854.0 1855.0 273.0 48.0 60.0 23.0 17.0 23.0 16.0 24.0 22.0 -2.0 10.0
8.0 0.0 10.0 4.0 31.0 7.0 9.0 6.0 3.0 15.0 76.0 71.0 279.0 709.0 807.0 604.0 238.0 79.0 30.0 31.0 19.0 39.0 -2.0 9.0 2.0 15.0 12.0 1.0 28.0 28.
18.0 25.0 72.0 6.0 13.0 16.0 28.0 16.0 27.0 0.0 19.0 87.0 88.0 66.0 89.0 78.0 45.0 40.0 45.0 15.0 27.0 18.0 27.0 14.0 17.0 7.0 25.0 9.0 32.0 15.
22.0 2.0 9.0 3.0 12.0 21.0 -2.0 16.0 13.0 36.0 8.0 34.0 57.0 91.0 14.0 40.0 27.0 32.0 19.0 9.0 14.0 24.0 16.0 26.0 7.0 11.0 10.0 9.0 16.0 6.0 8.
14.0 13.0 12.0 27.0 6.0 23.0 10.0 16.0 13.0 35.0 25.0 26.0 44.0 58.0 34.0 26.0 26.0 10.0 19.0 -1.0 22.0 0.0 15.0 -7.0 16.0 19.0 16.0 9.0 11.0 8.
34.0 2.0 9.0 12.0 -6.0 9.0 17.0 22.0 16.0 25.0 38.0 8.0 28.0 40.0 15.0 12.0 37.0 26.0 29.0 12.0 -7.0 15.0 14.0 22.0 9.0 27.0 23.0 6.0 11.0 15.0
30.0 32.0 18.0 12.0 17.0 21.0 14.0 14.0 -1.0 24.0 22.0 24.0 26.0 29.0 29.0 28.0 9.0 15.0 26.0 25.0 -15.0 15.0 18.0 -4.0 14.0 28.0 26.0 -6.0 12.
35.0 29.0 20.0 -1.0 6.0 12.0 -2.0 18.0 11.0 37.0 28.0 27.0 22.0 13.0 16.0 22.0 16.0 2.0 27.0 22.0 20.0 13.0 24.0 21.0 20.0 -0.0 16.0 14.0 12.0
14.0 -4.0 9.0 12.0 21.0 30.0 36.0 8.0 27.0 15.0 11.0 22.0 13.0 -3.0 29.0 29.0 12.0 30.0 27.0 19.0 14.0 28.0 25.0 14.0 6.0 -3.0 15.0 11.0 9.0 28.
15.0 2.0 18.0 19.0 29.0 28.0 15.0 13.0 20.0 12.0 24.0 16.0 26.0 16.0 18.0 23.0 11.0 26.0 30.0 2.0 21.0 7.0 16.0 26.0 6.0 25.0 15.0 14.0 18.0 16.
17.0 8.0 25.0 1.0 13.0 5.0 5.0 -7.0 11.0 47.0 17.0 16.0 21.0 15.0 2.0 11.0 33.0 18.0 10.0 29.0 24.0 12.0 18.0 7.0 17.0 5.0 22.0 1.0 3.0 3.0 3.0
24.0 15.0 7.0 21.0 8.0 9.0 35.0 20.0 4.0 18.0 17.0 11.0 4.0 34.0 10.0 16.0 4.0 -8.0 -7.0 -2.0 16.0 25.0 16.0 9.0 -8.0 38.0 5.0 16.0 21.0 -2.0 1

| | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 70.0 | 96.0 | 53.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 69.0 | 454.0 | 756.0 | 136.0 | 51.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 67.0 | 135.0 | 1404.0 | 2788.0 | 550.0 | 108.0 | 59.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 49.0 | 530.0 | 4357.0 | 5854.0 | 1855.0 | 273.0 | 48.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 71.0 | 279.0 | 709.0 | 807.0 | 604.0 | 238.0 | 79.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 88.0 | 66.0 | 89.0 | 78.0 | 45.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 91.0 | 14.0 | 39.0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

# Developing an Algorithm

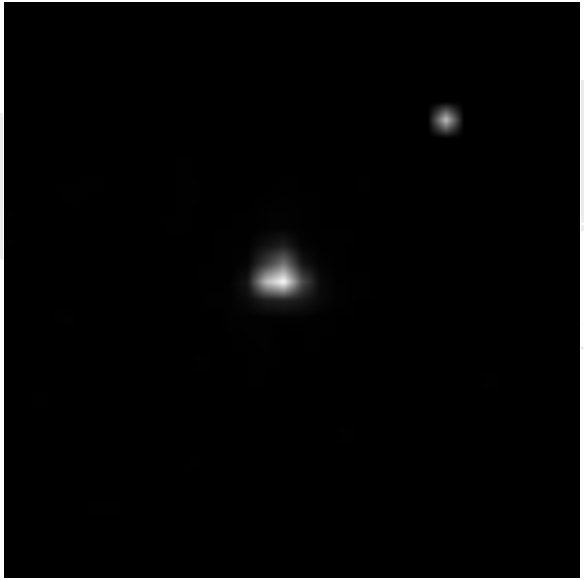
- Subject of the data is a star
- Have to understand how light from a star behaves and how to approximate the data



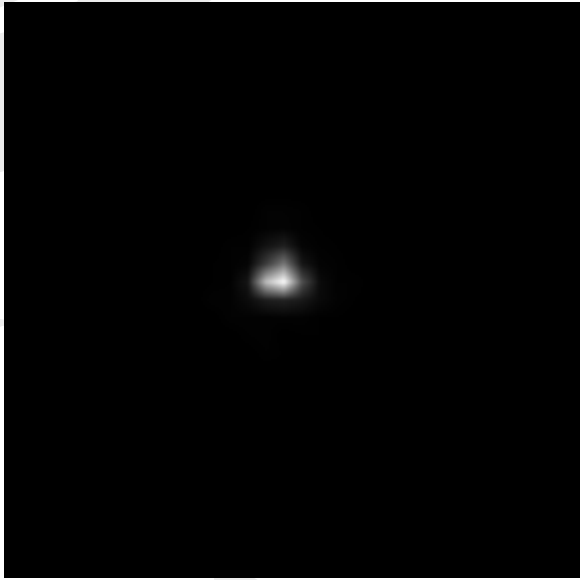



Images on this slide from: Wikipedia user Sakurambo

# Progress & Results



Spitzer image of HD 209458



Output of prototype: aperture radius = 2 pixels; noise spike is masked

# What's Next

- ➢ Revise algorithm to use and learn from pre-masked images (labeled data)
- ➢ Extend algorithm to operate on ~1000 FITS files; each FITS file contains 64 frames
- ➢ Parallelize with MPI library for Python

# Timeline

| | |
|---|---|
| May 12 | Complete algorithm revisions, numpy optimizations, use of labeled data |
| May 17 | Complete small scale implementation of MPI |
| May 25 | Complete parallelized implementation on RCC cluster |
| May 29 | Test on new datasets - different star systems |
| June 3 | Complete debugging, speed improvements and accuracy of output |