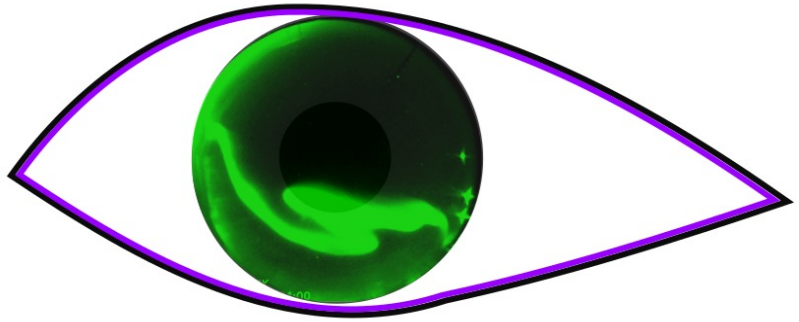
aurora-asi-lib

Aurora All-Sky Imager Library



Easily download, plot, animate, and analyze auroral all sky imager (ASI) data

Mykhaylo (Mike) Shumko, Bea Gallardo-Lacourt, Isaac Thompson, Alexa Halford, and Kyle Murphy

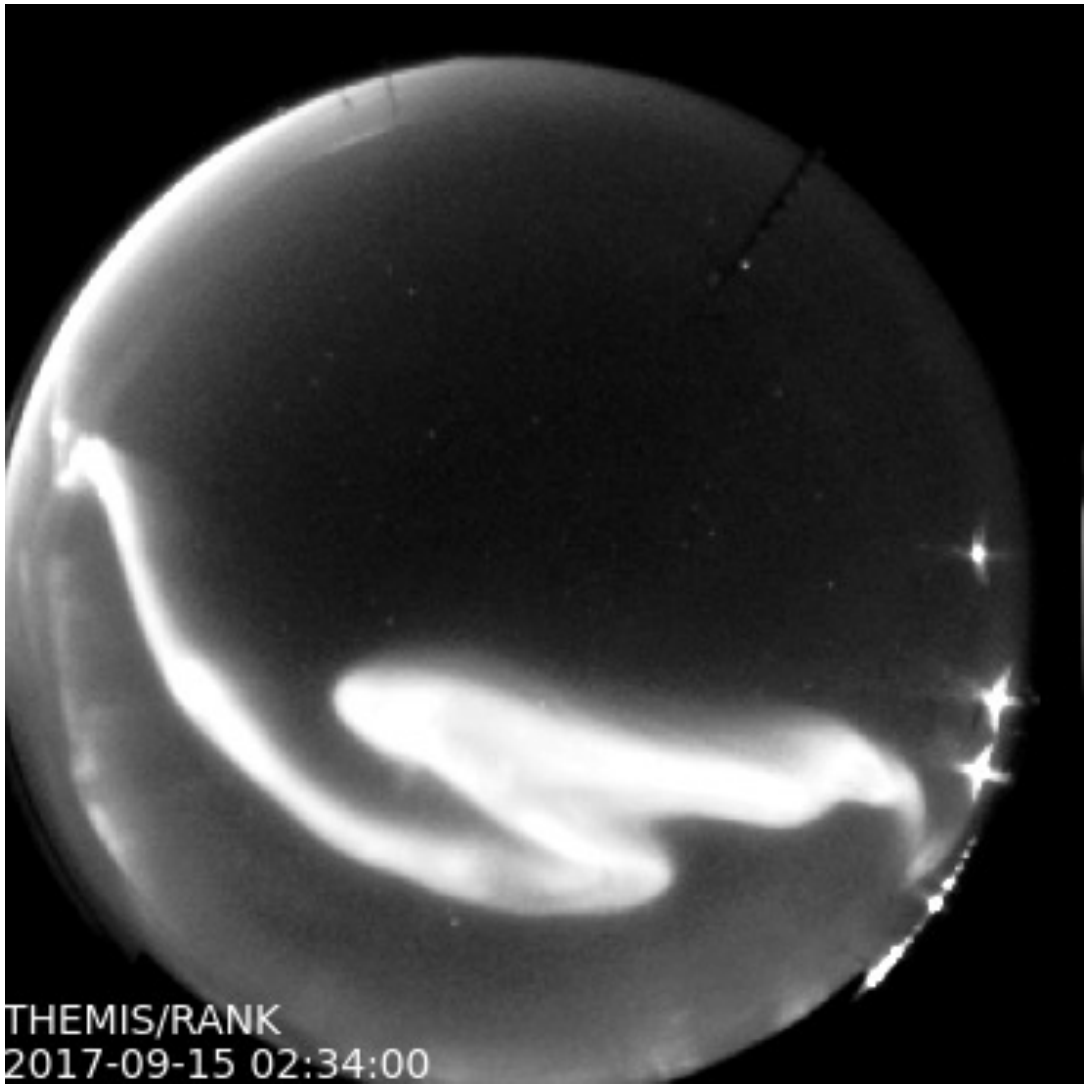
aurora-asi-lib overview

What?

A python package that enables seamless and painless handling and analysis of auroral images

Why?

Who has the time to write programs to process aurora data? **All auroral researchers do similar analysis steps---asilib helps by focusing their time and energy on what matters: studying the aurora!**

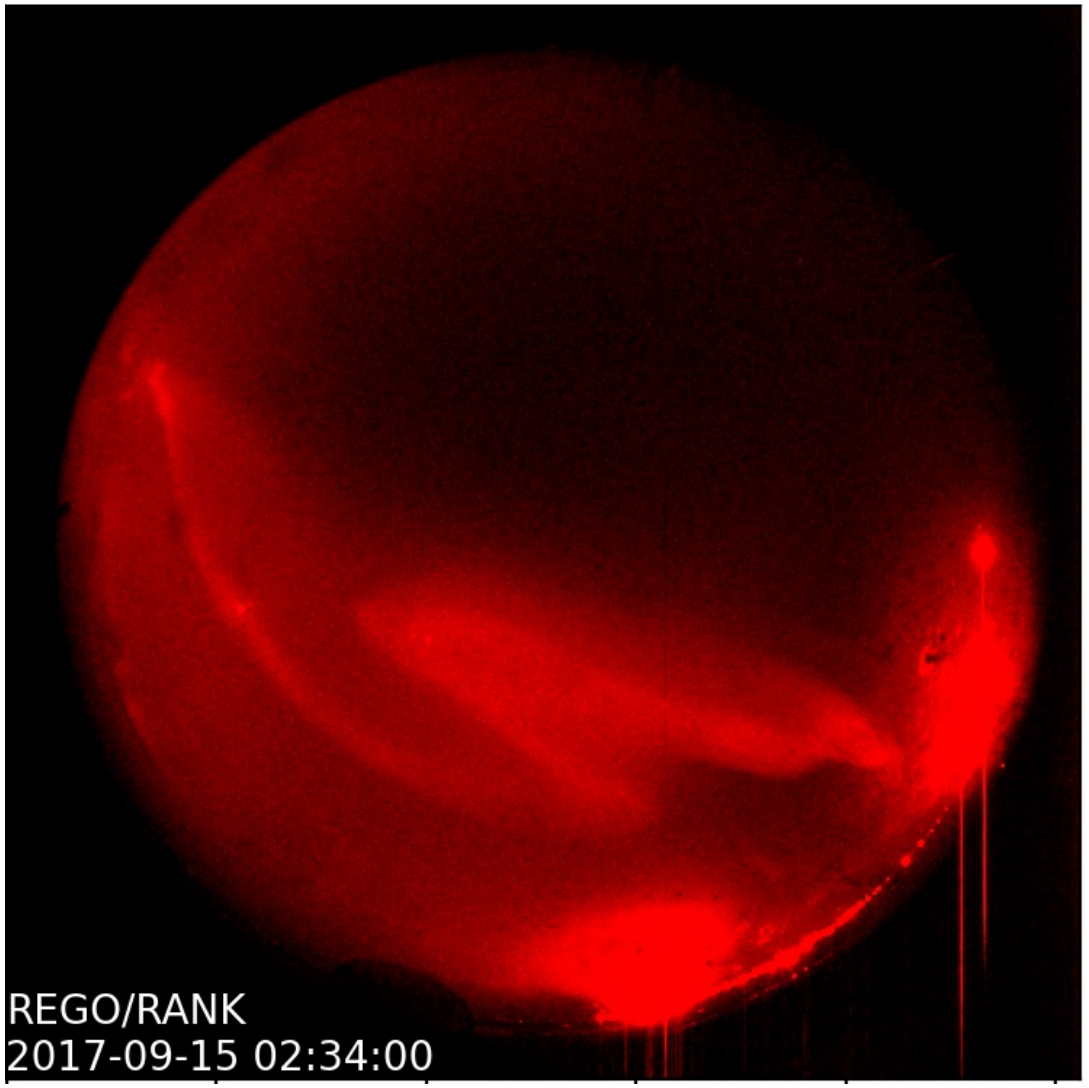


aurora-asi-lib overview

Supported camera arrays:

- **THEMIS**
- **REGO**

Once these two missions are fully supported, we plan to add other camera arrays to asilib.



What can it do?

Plot one fisheye lens frame:

asilib.plot_frame()

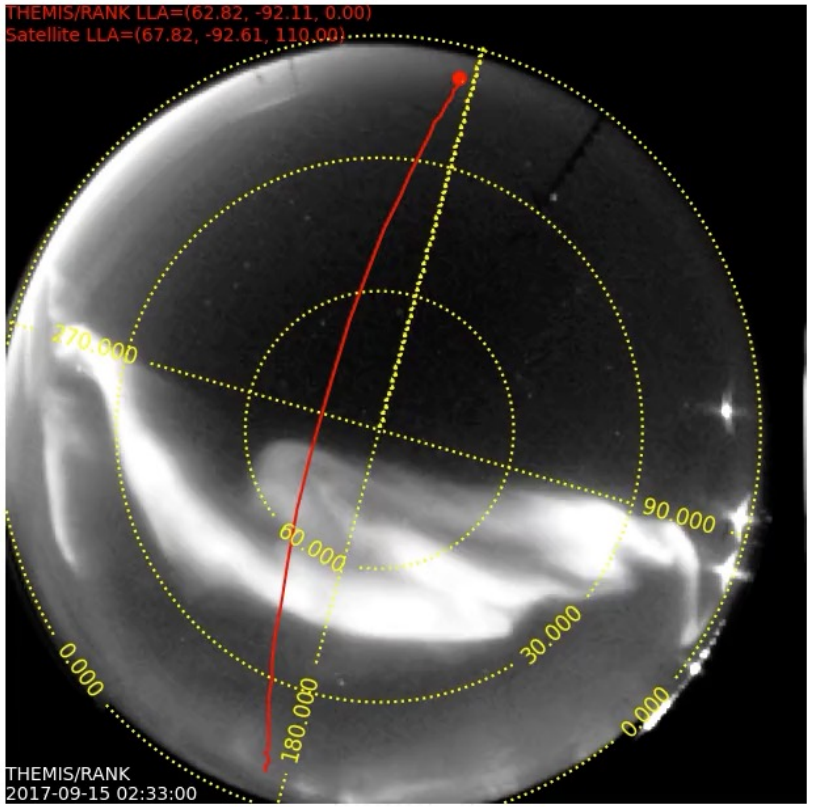
Make a movie:

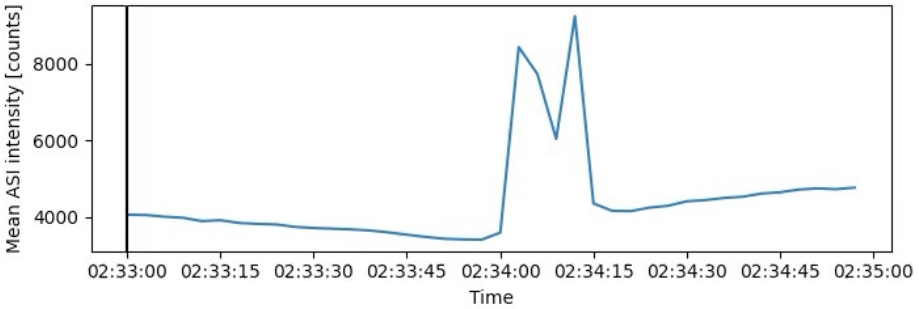
asilib.plot_movie()*

asilib.plot_movie_generator()*

Plot a keogram:

asilib.plot_keogram()





* Requires ffmpeg

What can it do?

Map a satellite's location:

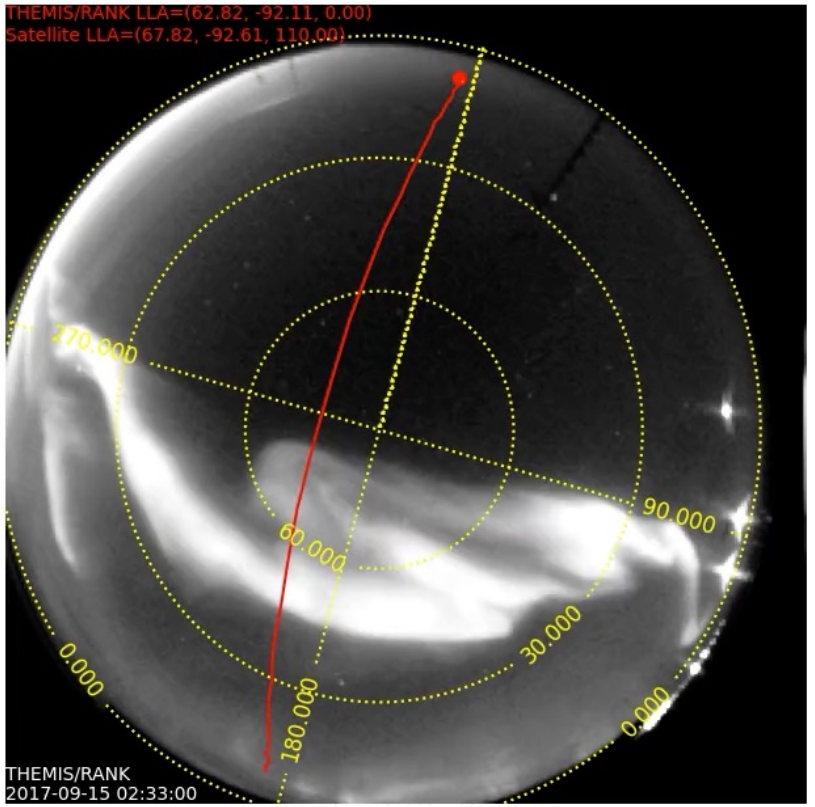
asilib.lla2azel()

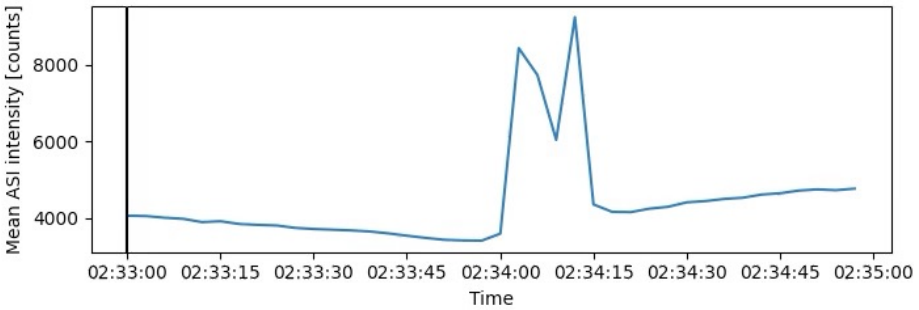
asilib.lla2footprint()*

Calculate equal areas in the image:

asilib.equal_area()

* Requires IRBEM





What can it do?

Load data

asilib.load_img()

asilib.load_cal()

If a file is not found, one will be automatically downloaded!

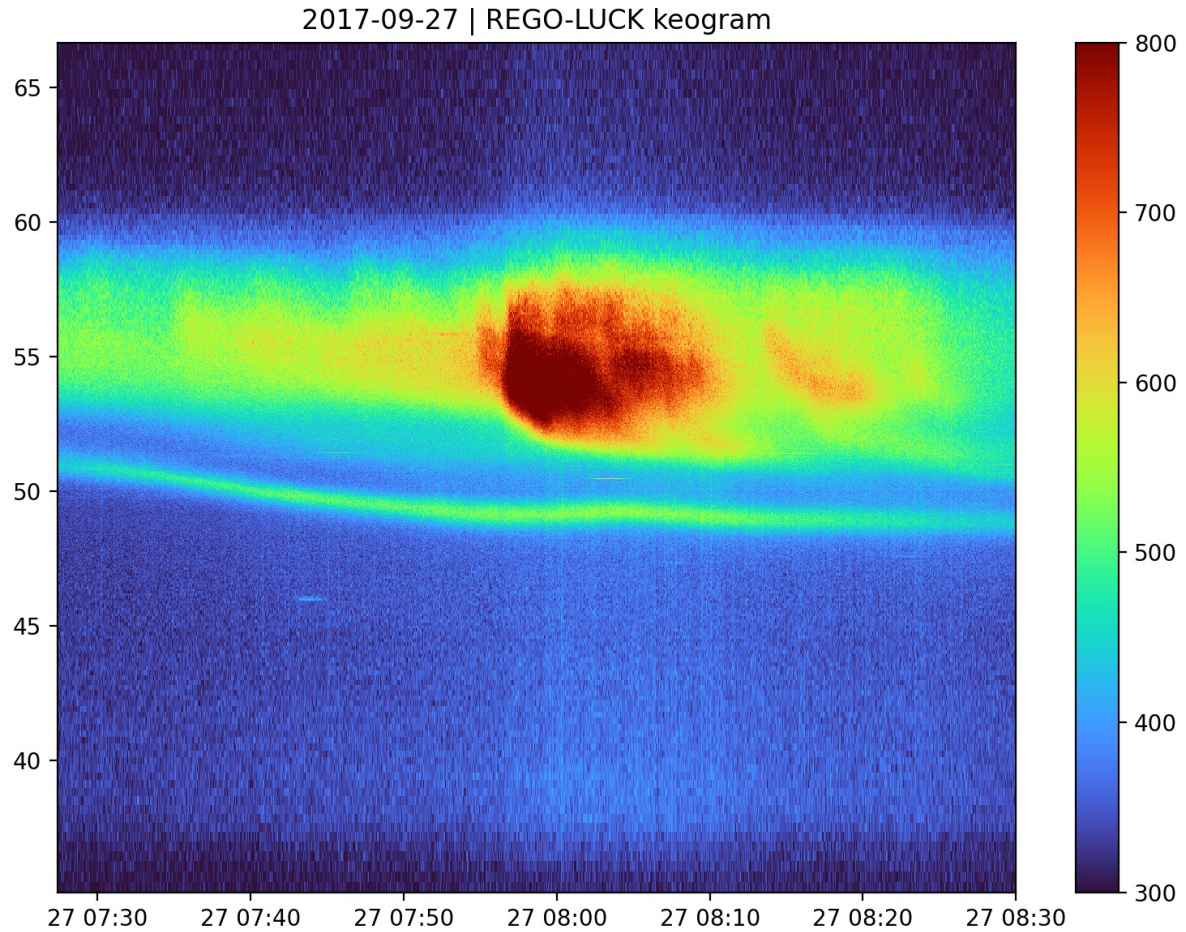
Bulk download data

asilib.download_themis_cal()

asilib.download_themis_img()

asilib.download_rego_cal()

asilib.download_rego_img()



One class to rule them all

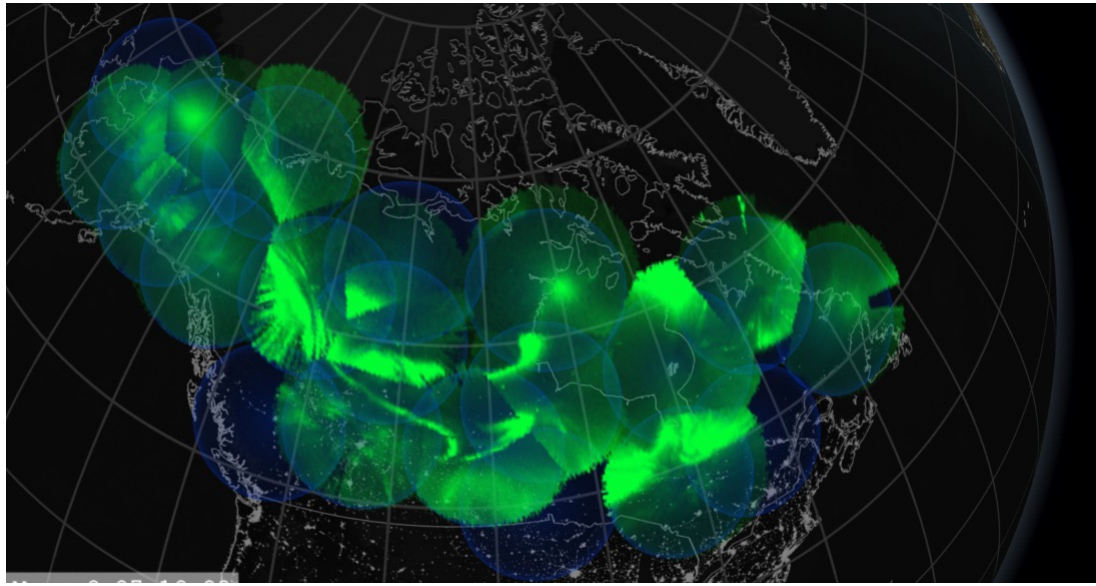
- The most usable (and fun!?) python libraries have a central class:
  - numpy.array
  - pandas.DataFrame
  - xarray.DataArray
  - pysat.Instrument
  - bs4.BeautifulSoup
  - …

One class to rule them all

- The most usable (and fun!?) python libraries have a central class:
  - numpy.array
  - pandas.DataFrame
  - xarray.DataArray
  - pysat.Instrument
  - bs4.BeautifulSoup
  - …
- And now:
  - aurora-asi-lib -> Imager

Ongoing Development Topics

- Handle computer resources effectively
- Project the fisheye images to maps (e.g. plot on the right)
- Unify the asilib functionality into an asilib.Imager() class
- Integrate with Aurora X
- Update the documentation with more examples
- And add other imager arrays as plugins



We need your help! Please contact me, mykhaylo.shumko@nasa.gov if you'd like to contribute or have ideas (I am always interested in ways to improve asilib)

How to get started

python3 –m pip install aurora-asi-lib (import as asilib)

Documentation: https://aurora-asi-lib.readthedocs.io

Code: https://github.com/mshumko/aurora-asi-lib

Thank you for listening!