shumko_asilib_figures

July 15, 2022

1 Figure Notebook for "AuroraX, aurorax-api and aurorax-asilib: a user-friendly auroral all-sky imager analysis framework"

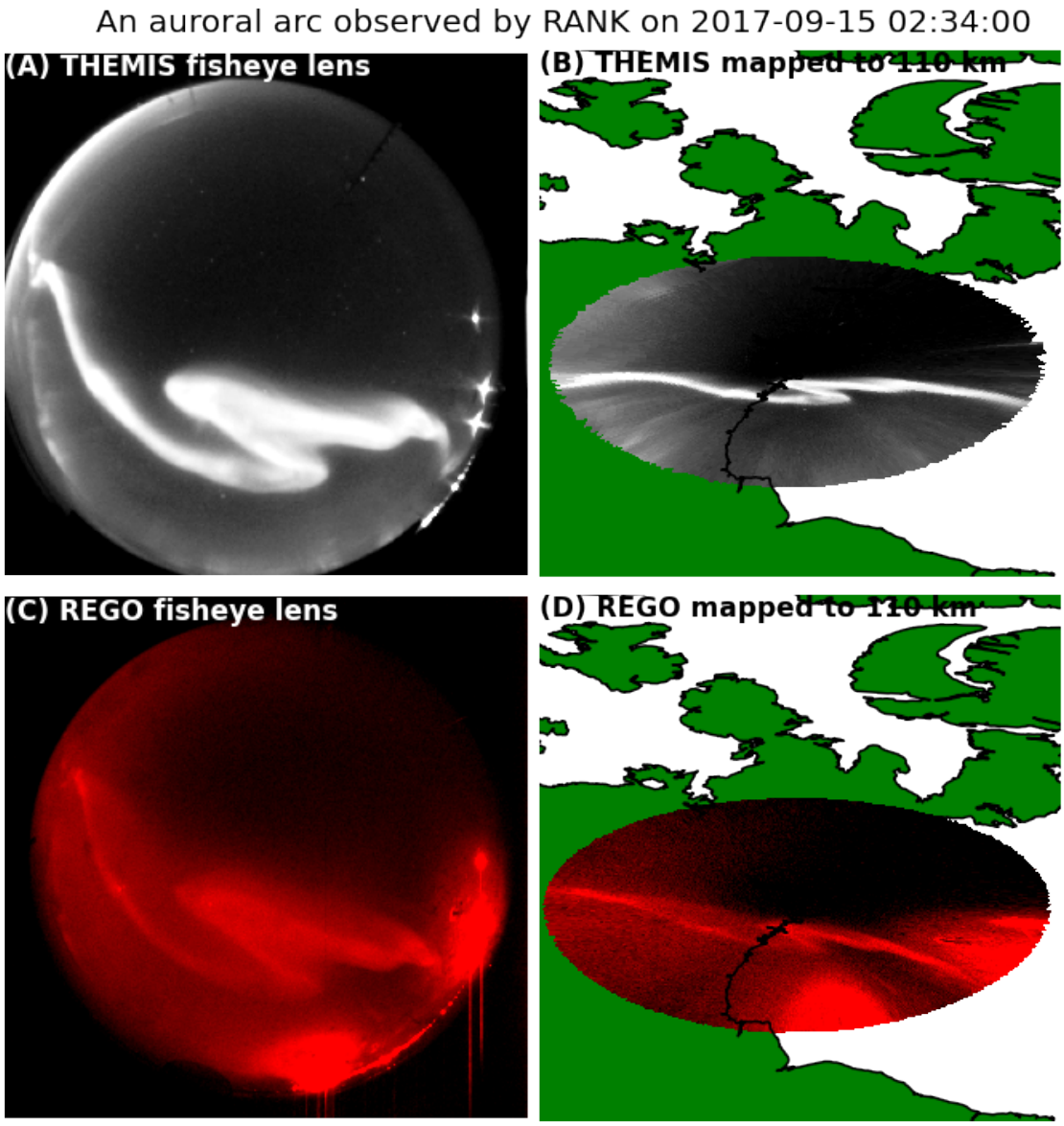
[ ]:
```
from datetime import datetime, timedelta
import string

import matplotlib.pyplot as plt
from matplotlib import dates
from matplotlib.gridspec import GridSpec
import numpy as np
import asilib

print(f'asilib version: {asilib.__version__}')
```

```
asilib version: 0.11.0
```

2 Figure 2

[ ]:
```
location_code = 'RANK'
time = datetime(2017, 9, 15, 2, 34, 0)
map_alt_km = 110
fontsize=17

lon_bounds = (-102, -82)
lat_bounds = (55, 75)

fig, ax = plt.subplots(2, 2, figsize=(10, 10))
asilib.make_map(ax=ax[0, 1], lon_bounds=lon_bounds, lat_bounds=lat_bounds)
asilib.make_map(ax=ax[1, 1], lon_bounds=lon_bounds, lat_bounds=lat_bounds)

ax[0, 0].axis('off')
ax[1, 0].axis('off')
ax[0, 1].axis('off')
ax[1, 1].axis('off')

asilib.plot_fisheye('THEMIS', location_code, time, ax=ax[0, 0], label=False)
asilib.plot_fisheye('REGO', location_code, time, ax=ax[1, 0], label=False)
```

```
asilib.plot_map('THEMIS', location_code, time, map_alt_km, ax=ax[0, 1],␣
 ↪asi_label=False)
asilib.plot_map('REGO', location_code, time, map_alt_km, ax=ax[1, 1],␣
 ↪asi_label=False)

ax[0, 0].text(0, 1, f'(A) THEMIS fisheye lens', va='top', transform=ax[0,0].
 ↪transAxes,
    color='white', fontsize=fontsize, weight='bold')
ax[0, 1].text(0, 1, f'(B) THEMIS mapped to {map_alt_km} km', va='top',␣
 ↪transform=ax[0,1].transAxes,
    color='k', fontsize=fontsize, weight='bold')
ax[1, 0].text(0, 1, f'(C) REGO fisheye lens', va='top', transform=ax[1,0].
 ↪transAxes,
    color='white', fontsize=fontsize, weight='bold')
ax[1, 1].text(0, 1, f'(D) REGO mapped to {map_alt_km} km', va='top',␣
 ↪transform=ax[1,1].transAxes,
    color='k', fontsize=fontsize, weight='bold')

plt.suptitle(f'An auroral arc observed by {location_code} on {time}',␣
 ↪fontsize=20)
plt.tight_layout()
plt.savefig('figures/fig2.jpg', dpi=300)
```



[ ]:
```
themis_skymap = asilib.load_skymap('THEMIS', location_code, time)
rego_skymap = asilib.load_skymap('REGO', location_code, time)
```

[ ]:
```
themis_skymap['SKYMAP_PATH']
```

[ ]:
```
PosixPath('/media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/themis/skymap/rank/themis_skymap_rank_20150825_vXX.sav')
```

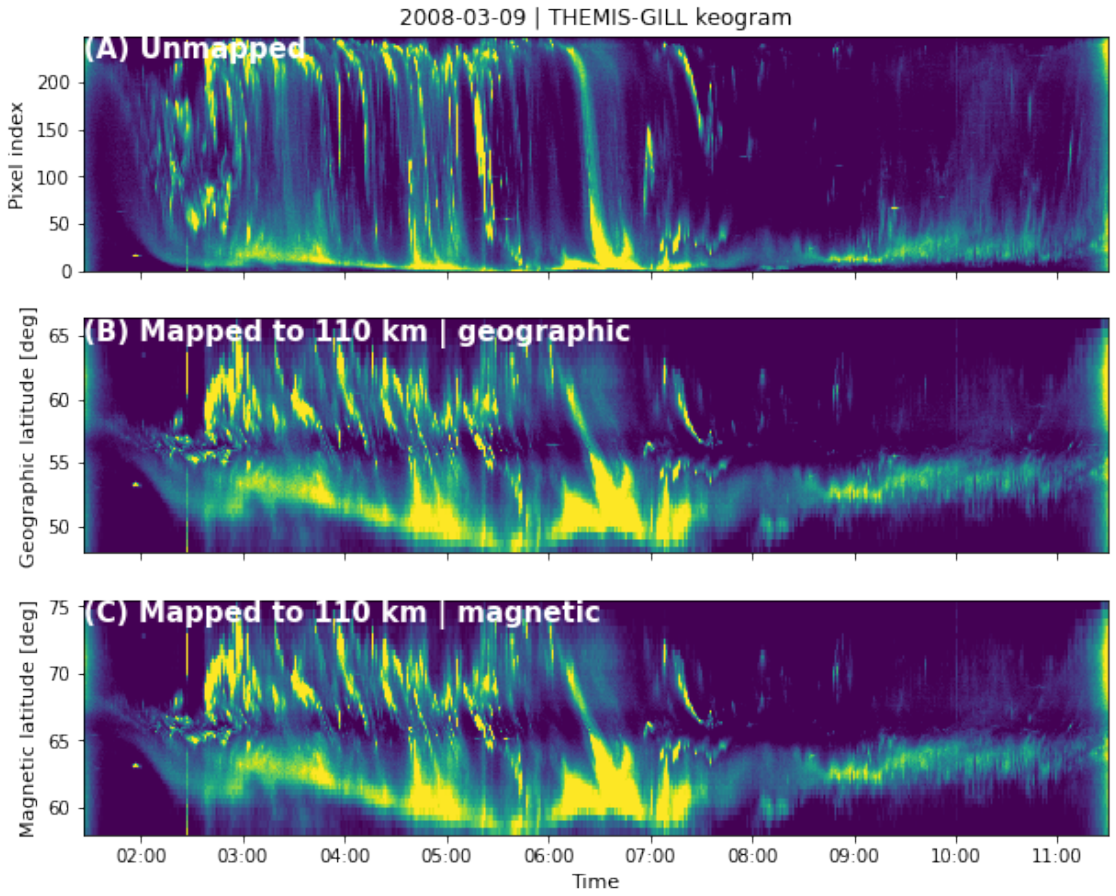
[ ]:
```
rego_skymap['SKYMAP_PATH']
```

[ ]:
```
PosixPath('/media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/rego/skymap/rank/rego_skymap_rank_20170317_vXX.sav')
```

3 Figure 3

[ ]:
```
fontsize=11
fig, ax = plt.subplots(3, 1, figsize=(10, 8), sharex=True)
time_range = (datetime(2008, 3, 9), datetime(2008, 3, 10))
asilib.plot_keogram('THEMIS', 'GILL', time_range, ax=ax[0])
asilib.plot_keogram('THEMIS', 'GILL', time_range, ax=ax[1], map_alt=map_alt_km)
asilib.plot_keogram('THEMIS', 'GILL', time_range, ax=ax[2], map_alt=map_alt_km,␣
↪aacgm=True)
ax[-1].set_xlabel('Time', fontsize=fontsize)
ax[0].set_ylabel('Pixel index', fontsize=fontsize)
ax[1].set_ylabel('Geographic latitude [deg]', fontsize=fontsize)
ax[2].set_ylabel('Magnetic latitude [deg]', fontsize=fontsize)

fmtr = dates.DateFormatter("%H:%M")
ax[-1].xaxis.set_major_formatter(fmtr)
ax[1].set_title('')
ax[2].set_title('')
ax[0].text(0, 1, f'(A) Unmapped', va='top', transform=ax[0].transAxes,
    color='white', fontsize=fontsize+4, weight='bold')
ax[1].text(0, 1, f'(B) Mapped to {map_alt_km} km | geographic', va='top',␣
↪transform=ax[1].transAxes,
    color='white', fontsize=fontsize+4, weight='bold')
ax[2].text(0, 1, f'(C) Mapped to {map_alt_km} km | magnetic', va='top',␣
↪transform=ax[2].transAxes,
    color='white', fontsize=fontsize+4, weight='bold')
plt.savefig('figures/fig3.jpg', dpi=300)
```

```
/home/mike/research/aurora-asi-lib/env/lib/python3.9/site-
packages/scipy/io/idl.py:281: UserWarning: Not able to verify number of bytes
from header
  warnings.warn("Not able to verify number of bytes from header")
```

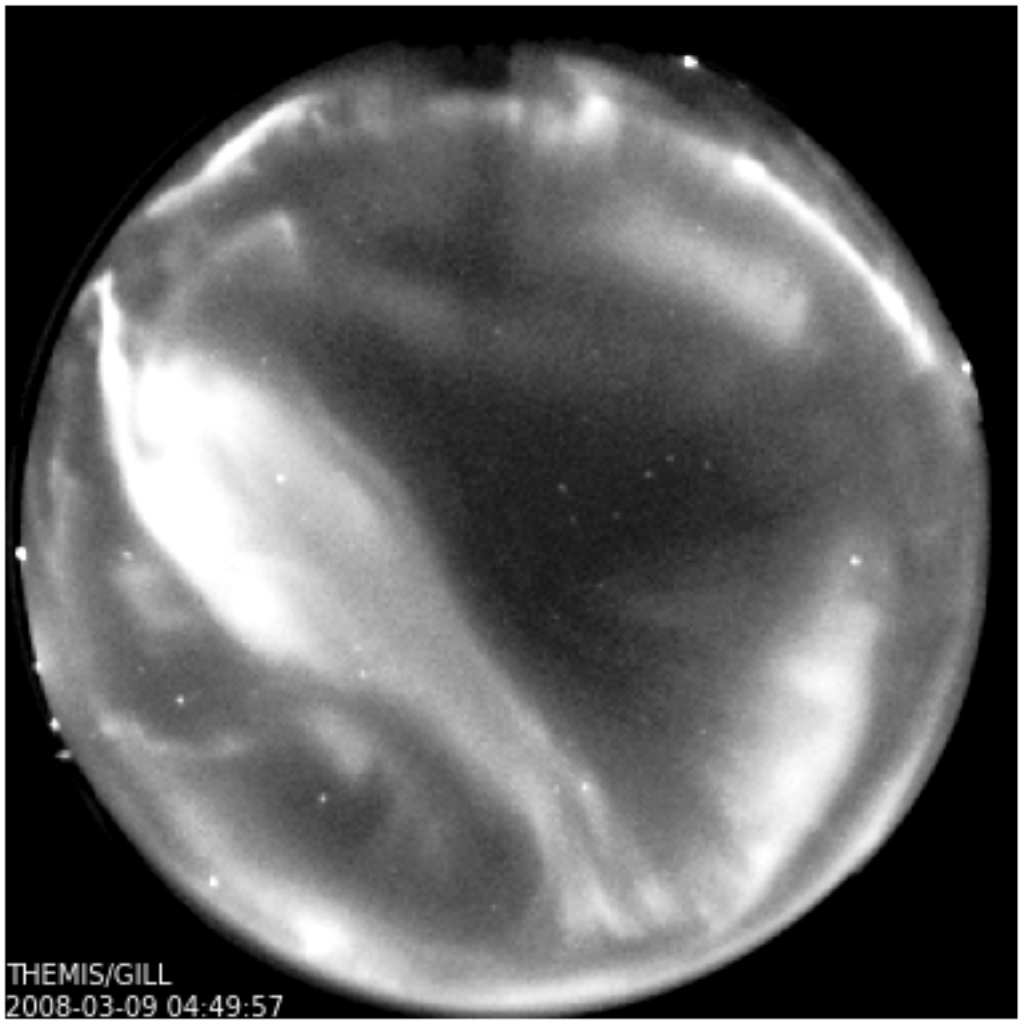


[ ]:
```
time_range = (datetime(2008, 3, 9, 4, 35), datetime(2008, 3, 9, 4, 50))
asilib.animate_fisheye('THEMIS', 'GILL', time_range, overwrite=True)
```

```
Created a /media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/animations/images/20080309_043500_themis_gill_fisheye directory

ffmpeg version 4.2.7-0ubuntu0.1 Copyright (c) 2000-2022 the FFmpeg developers
  built with gcc 9 (Ubuntu 9.4.0-1ubuntu1~20.04.1)
  configuration: --prefix=/usr --extra-version=0ubuntu0.1 --toolchain=hardened
--libdir=/usr/lib/x86_64-linux-gnu --incdir=/usr/include/x86_64-linux-gnu
--arch=amd64 --enable-gpl --disable-stripping --enable-avresample --disable-
filter=resample --enable-avisynth --enable-gnutls --enable-ladspa --enable-
libaom --enable-libass --enable-libbluray --enable-libbs2b --enable-libcaca
--enable-libcdio --enable-libcodec2 --enable-libflite --enable-libfontconfig
--enable-libfreetype --enable-libfribidi --enable-libgme --enable-libgsm
--enable-libjack --enable-libmp3lame --enable-libmysofa --enable-libopenjpeg
--enable-libopenmpt --enable-libopus --enable-libpulse --enable-librsvg
--enable-librubberband --enable-libshine --enable-libsnappy --enable-libsoxr
--enable-libspeex --enable-libssh --enable-libtheora --enable-libtwolame
--enable-libvidstab --enable-libvorbis --enable-libvpx --enable-libwavpack
```

```
--enable-libwebp --enable-libx265 --enable-libxml2 --enable-libxvid --enable-
libzmq --enable-libzvbi --enable-lv2 --enable-omx --enable-openal --enable-
opencl --enable-opengl --enable-sdl2 --enable-libdc1394 --enable-libdrm
--enable-libiec61883 --enable-nvenc --enable-chromaprint --enable-frei0r
--enable-libx264 --enable-shared
  WARNING: library configuration mismatch
  avcodec     configuration: --prefix=/usr --extra-version=0ubuntu0.1
--toolchain=hardened --libdir=/usr/lib/x86_64-linux-gnu
--incdir=/usr/include/x86_64-linux-gnu --arch=amd64 --enable-gpl --disable-
stripping --enable-avresample --disable-filter=resample --enable-avisynth
--enable-gnutls --enable-ladspa --enable-libaom --enable-libass --enable-
libbluray --enable-libbs2b --enable-libcaca --enable-libcdio --enable-libcodec2
--enable-libflite --enable-libfontconfig --enable-libfreetype --enable-
libfribidi --enable-libgme --enable-libgsm --enable-libjack --enable-libmp3lame
--enable-libmysofa --enable-libopenjpeg --enable-libopenmpt --enable-libopus
--enable-libpulse --enable-librsvg --enable-librubberband --enable-libshine
--enable-libsnappy --enable-libsoxr --enable-libspeex --enable-libssh --enable-
libtheora --enable-libtwolame --enable-libvidstab --enable-libvorbis --enable-
libvpx --enable-libwavpack --enable-libwebp --enable-libx265 --enable-libxml2
--enable-libxvid --enable-libzmq --enable-libzvbi --enable-lv2 --enable-omx
--enable-openal --enable-opencl --enable-opengl --enable-sdl2 --enable-libdc1394
--enable-libdrm --enable-libiec61883 --enable-nvenc --enable-chromaprint
--enable-frei0r --enable-libx264 --enable-shared --enable-version3 --disable-doc
--disable-programs --enable-libaribb24 --enable-liblensfun --enable-
libopencore_amrnb --enable-libopencore_amrwb --enable-libtesseract --enable-
libvo_amrwbenc
  libavutil      56. 31.100 / 56. 31.100
  libavcodec     58. 54.100 / 58. 54.100
  libavformat    58. 29.100 / 58. 29.100
  libavdevice    58.  8.100 / 58.  8.100
  libavfilter     7. 57.100 /  7. 57.100
  libavresample   4.  0.  0 /  4.  0.  0
  libswscale      5.  5.100 /  5.  5.100
  libswresample   3.  5.100 /  3.  5.100
  libpostproc    55.  5.100 / 55.  5.100
Input #0, image2, from '/media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/animations/images/20080309_043500_themis_gill_fisheye/%05d.png':
  Duration: 00:00:30.00, start: 0.000000, bitrate: N/A
    Stream #0:0: Video: png, rgba(pc), 432x432 [SAR 2835:2835 DAR 1:1], 10 fps,
10 tbr, 10 tbn, 10 tbc
Stream mapping:
  Stream #0:0 -> #0:0 (png (native) -> h264 (libx264))
Press [q] to stop, [?] for help
[libx264 @ 0x5616422c0140] using SAR=1/1
[libx264 @ 0x5616422c0140] using cpu capabilities: MMX2 SSE2Fast SSSE3 SSE4.2
AVX FMA3 BMI2 AVX2
[libx264 @ 0x5616422c0140] profile High, level 2.2
[libx264 @ 0x5616422c0140] 264 - core 155 r2917 0a84d98 - H.264/MPEG-4 AVC codec
```

```
- Copyleft 2003-2018 - http://www.videolan.org/x264.html - options: cabac=1
ref=8 deblock=1:0:0 analyse=0x3:0x133 me=umh subme=9 psy=1 psy_rd=1.00:0.00
mixed_ref=1 me_range=16 chroma_me=1 trellis=2 8x8dct=1 cqm=0 deadzone=21,11
fast_pskip=1 chroma_qp_offset=-2 threads=12 lookahead_threads=1 sliced_threads=0
nr=0 decimate=1 interlaced=0 bluray_compat=0 constrained_intra=0 bframes=3
b_pyramid=2 b_adapt=2 b_bias=0 direct=3 weightb=1 open_gop=0 weightp=2
keyint=250 keyint_min=10 scenecut=40 intra_refresh=0 rc_lookahead=60 rc=crf
mbtree=1 crf=25.0 qcomp=0.60 qpmin=0 qpmax=69 qpstep=4 ip_ratio=1.40 aq=1:1.00
Output #0, mp4, to '/media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/animations/20080309_043500_044957_themis_gill_fisheye.mp4':
  Metadata:
    encoder         : Lavf58.29.100
    Stream #0:0: Video: h264 (libx264) (avc1 / 0x31637661), yuv420p, 432x432
[SAR 1:1 DAR 1:1], q=-1--1, 10 fps, 10240 tbn, 10 tbc
    Metadata:
      encoder         : Lavc58.54.100 libx264
    Side data:
      cpb: bitrate max/min/avg: 0/0/0 buffer size: 0 vbv_delay: -1
frame=  300 fps=222 q=-1.0 Lsize=    1130kB time=00:00:29.70 bitrate=
311.6kbits/s speed=  22x
video:1125kB audio:0kB subtitle:0kB other streams:0kB global headers:0kB muxing
overhead: 0.385254%
[libx264 @ 0x5616422c0140] frame I:2     Avg QP:20.50  size: 11730
[libx264 @ 0x5616422c0140] frame P:77    Avg QP:24.13  size:  6469
[libx264 @ 0x5616422c0140] frame B:221   Avg QP:26.31  size:  2852
[libx264 @ 0x5616422c0140] consecutive B-frames:  0.7%  2.0%  4.0% 93.3%
[libx264 @ 0x5616422c0140] mb I  I16..4: 27.3% 58.7% 14.0%
[libx264 @ 0x5616422c0140] mb P  I16..4:  0.4% 15.4%  1.2%  P16..4: 22.4% 17.7%
6.8%  0.3%  0.1%    skip:35.8%
[libx264 @ 0x5616422c0140] mb B  I16..4:  0.0%  1.8%  0.1%  B16..8: 22.0%  7.6%
2.6%  direct: 9.0%  skip:57.0%  L0:46.9% L1:44.2% BI: 8.9%
[libx264 @ 0x5616422c0140] 8x8 transform intra:88.5% inter:76.6%
[libx264 @ 0x5616422c0140] direct mvs  spatial:97.3% temporal:2.7%
[libx264 @ 0x5616422c0140] coded y,uvDC,uvAC intra: 84.5% 0.0% 0.0% inter: 30.1%
0.0% 0.0%
[libx264 @ 0x5616422c0140] i16 v,h,dc,p: 65% 23% 11%  1%
[libx264 @ 0x5616422c0140] i8 v,h,dc,ddl,ddr,vr,hd,vl,hu:  6%  3%  9%  7% 23%
19% 16%  8%  7%
[libx264 @ 0x5616422c0140] i4 v,h,dc,ddl,ddr,vr,hd,vl,hu: 10%  5%  5%  6% 23%
18% 15%  9%  9%
[libx264 @ 0x5616422c0140] i8c dc,h,v,p: 100%  0%  0%  0%
[libx264 @ 0x5616422c0140] Weighted P-Frames: Y:3.9% UV:0.0%
[libx264 @ 0x5616422c0140] ref P L0: 43.6% 14.3% 19.8%  5.8%  5.9%  3.5%  4.4%
2.6%  0.1%
[libx264 @ 0x5616422c0140] ref B L0: 75.5% 12.6%  5.4%  2.6%  2.0%  1.3%  0.6%
[libx264 @ 0x5616422c0140] ref B L1: 96.5%  3.5%
[libx264 @ 0x5616422c0140] kb/s:307.15
```



4 Figure 4

A conjunction montage. Lets take this one step at a time. First we define the ASI info and load the skymap file (to make the fictional satellite path overhead).

[ ]:
```
asi_array_code = 'THEMIS'
location_code = 'RANK'
area_box_km = (20, 20)
time_range = (datetime(2017, 9, 15, 2, 32, 0), datetime(2017, 9, 15, 2, 35, 0))
```

[ ]:
```
skymap_dict = asilib.load_skymap(asi_array_code, location_code, time_range[0])
```

Create the satellite path (time, latitude, longitude, altitude) at a 500 km altitude. It is a north-south satellite track oriented to the east of the imager.

```
[ ]: n = int((time_range[1] - time_range[0]).total_seconds() / 3)  #  3 second␣
     ↪cadence.
     time = np.array([time_range[0] + timedelta(seconds=i*3) for i in range(n)])
     lats = np.linspace(skymap_dict["SITE_MAP_LATITUDE"] + 2,␣
     ↪skymap_dict["SITE_MAP_LATITUDE"] - 3, n)
     lons = (skymap_dict["SITE_MAP_LONGITUDE"] - 0.25) * np.ones(n)
     alts = 500 * np.ones(n)
     time_lla_500km = np.array([time, lats, lons, alts]).T
```

Map the satellite's altitude from 500 km to the 110 km footprint. Time is necessary to correctly evaluate the magnetic field model.

NOTE You will need to install IRBEM for the following line to run.

```
[ ]: lla_110km = asilib.lla2footprint(time_lla_500km, 110)
```

Next, map the satellite's footprint to the imager's (Azimuth, Elevation), i.e. AzEl coordinates.

```
[ ]: sat_azel, sat_azel_pixels = asilib.lla2azel(asi_array_code, location_code,␣
     ↪time_range[0], lla_110km)
```

The last step before we make the movie is to calculate what pixels are in a box_km around the satellite, to convolve it with the images to pick out the ASI intensity in that box.

```
[ ]: area_box_mask = asilib.equal_area(
         asi_array_code, location_code, time_range[0], lla_110km, box_km=area_box_km
     )
```

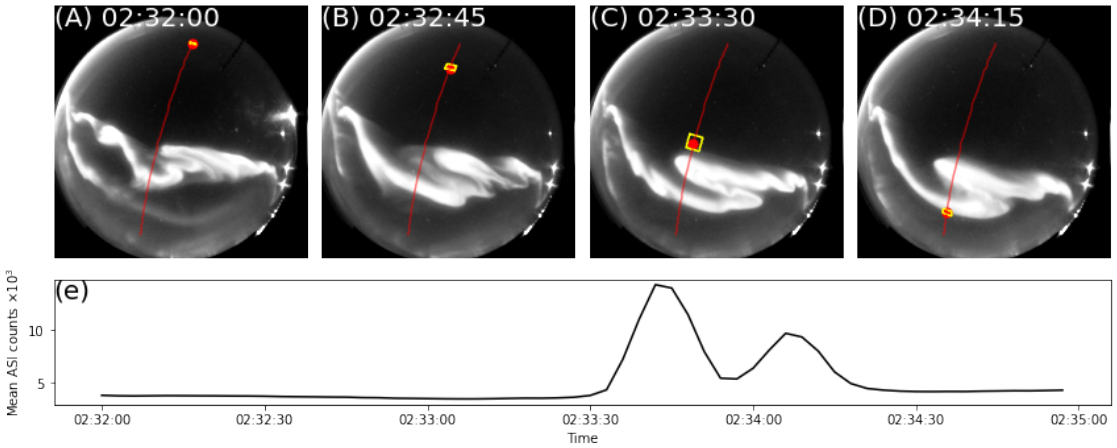
Calculate the mean ASI intensity in the area_box_km

```
[ ]: times, images = asilib.load_image(asi_array_code, location_code,␣
     ↪time_range=time_range)
     asi_brightness = np.nanmean(images * area_box_mask, axis=(1, 2))
     area_box_mask[np.isnan(area_box_mask)] = 0  # To play nice with plt.contour()
```

Plot preparation

```
[ ]: num_images = 4

     delta_time_s = int((time_range[1]-time_range[0]).total_seconds()/num_images)
     montage_times = [time_range[0]+timedelta(seconds=i*delta_time_s) for i in␣
     ↪range(num_images)]
     formatted_times = [t.strftime('%H:%M:%S') for t in montage_times]
     subplot_labels = [f'({l}) {t}' for l, t in zip(string.ascii_uppercase[:
     ↪num_images], formatted_times)]

     downsampled_satellite_indices = np.zeros(num_images)
     for i, montage_time in enumerate(montage_times):
         downsampled_satellite_indices[i] = np.where(time == montage_time)[0][0]
```

[ ]:
```
fig = plt.figure(figsize=(12, 5))

gs = GridSpec(3, num_images, figure=fig)
ax = [fig.add_subplot(gs[:2, n]) for n in range(num_images)]
bx = fig.add_subplot(gs[-1, :])

for i, (montage_time, ax_i, subplot_label) in enumerate(zip(montage_times, ax,␣
 ↪subplot_labels)):
    asilib.plot_fisheye(asi_array_code, location_code, montage_time, ax=ax_i,␣
 ↪label=False)
    ax_i.axis('off')

    index = int(downsampled_satellite_indices[i])
    ax_i.plot(sat_azel_pixels[:, 0], sat_azel_pixels[:, 1], 'red', alpha=0.5)
    ax_i.scatter(sat_azel_pixels[index, 0], sat_azel_pixels[index, 1], c='red',␣
 ↪marker='o', s=50)
    ax_i.contour(area_box_mask[index, :, :], levels=[0.99], colors=['yellow'])

    # # Plot the time series of the mean ASI intensity along the satellite path
    # ax[1].plot(image_data.time, asi_brightness)
    # ax[1].axvline(time, c='b')
    ax_i.text(0, 1, subplot_label, va='top', transform=ax_i.transAxes,␣
 ↪color='white', fontsize=20)

bx.plot(time, asi_brightness/1000, c='k')
bx.text(0, 1, f'({string.ascii_lowercase[num_images]})', va='top', transform=bx.
 ↪transAxes, fontsize=20)
bx.set_ylabel(r'Mean ASI counts $\times 10^3$')
bx.set_xlabel('Time')

plt.tight_layout()
plt.savefig('figures/fig4.jpg', dpi=300)
```



5 Movie S2

Now to make the conjunction movie.

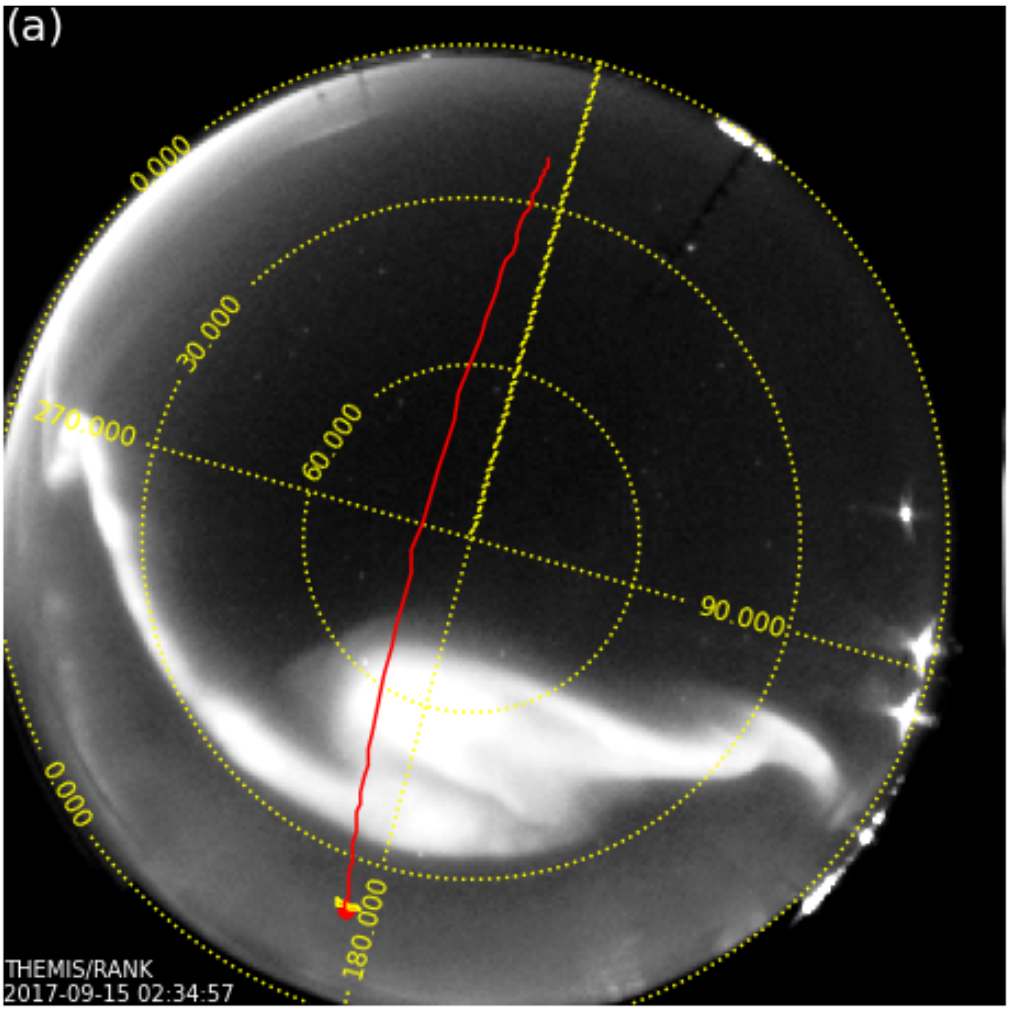
[ ]:
```
fig, ax = plt.subplots(
    2, 1, figsize=(7, 10), gridspec_kw={'height_ratios': [4, 1]},␣
 ↪constrained_layout=True
)

# Initiate the movie generator function. Any errors with the data will be␣
 ↪raised here.
movie_generator = asilib.animate_fisheye_generator(
    asi_array_code, location_code, time_range, azel_contours=True,␣
 ↪overwrite=True, ax=ax[0]
)

# Use the generator to get the images and time stamps to estimate mean the ASI
# brightness along the satellite path and in a (20x20 km) box.
image_data = movie_generator.send('data')

asi_brightness = np.nanmean(image_data.images * area_box_mask, axis=(1, 2))
area_box_mask[np.isnan(area_box_mask)] = 0  # To play nice with plt.contour()

for i, (time, image, _, im) in enumerate(movie_generator):
    # Note that because we are drawing different data in each frame (a unique␣
 ↪ASI
    # image in ax[0] and the ASI time series + a guide in ax[1], we need
    # to redraw everything at every iteration.

    ax[1].clear()  # ax[0] cleared by asilib.animate_fisheye_generator()
    # Plot the entire satellite track, its current location, and a 20x20 km box
    # around its location.
    ax[0].plot(sat_azel_pixels[:, 0], sat_azel_pixels[:, 1], 'red')
    ax[0].scatter(sat_azel_pixels[i, 0], sat_azel_pixels[i, 1], c='red',␣
 ↪marker='o', s=50)
    ax[0].contour(area_box_mask[i, :, :], levels=[0.99], colors=['yellow'])

    # Plot the time series of the mean ASI intensity along the satellite path
    ax[1].plot(image_data.time, asi_brightness)
    ax[1].axvline(time, c='b')

    # Annotate the location_code and satellite info in the top-left corner.
    location_code_str = (
        f'{asi_array_code}/{location_code} '
        f'LLA=({skymap_dict["SITE_MAP_LATITUDE"]:.2f}, '
```

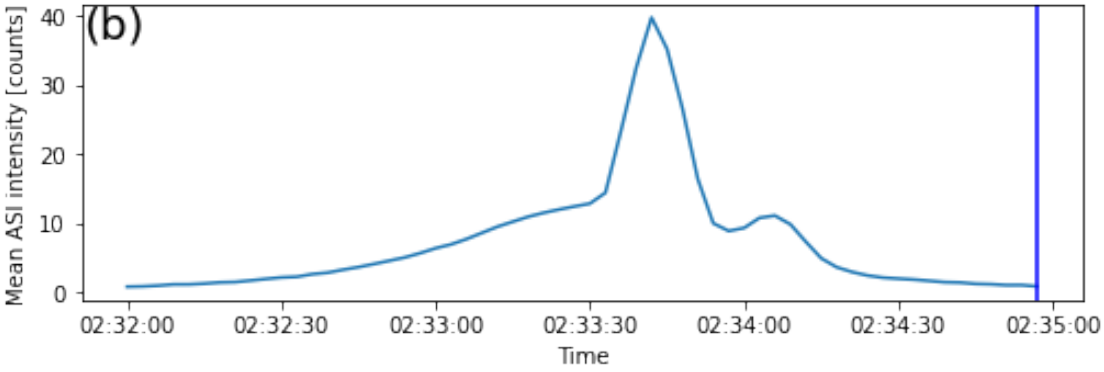
```
        f'{skymap_dict["SITE_MAP_LONGITUDE"]:.2f},␣
↪{skymap_dict["SITE_MAP_ALTITUDE"]:.2f})'
    )
    satellite_str = f'Satellite LLA=({lla_110km[i, 0]:.2f}, {lla_110km[i, 1]:.
↪2f}, {lla_110km[i, 2]:.2f})'
    ax[0].text(0, 1, '(a)', va='top', transform=ax[0].transAxes, color='white',␣
↪fontsize=20)
    ax[1].set(xlabel='Time', ylabel='Mean ASI intensity [counts]')
    ax[1].text(0, 1, '(b)', va='top', transform=ax[1].transAxes, color='black',␣
↪fontsize=20)

print(f'Movie saved in {asilib.config["ASI_DATA_DIR"] / "movies"}')
```

```
Created a /media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/animations/images/20170915_023200_themis_rank_fisheye directory

/usr/local/lib/python3.9/subprocess.py:1048: ResourceWarning: subprocess 76500
is still running
  _warn("subprocess %s is still running" % self.pid,
ffmpeg version 4.2.7-0ubuntu0.1 Copyright (c) 2000-2022 the FFmpeg developers
  built with gcc 9 (Ubuntu 9.4.0-1ubuntu1~20.04.1)
  configuration: --prefix=/usr --extra-version=0ubuntu0.1 --toolchain=hardened
--libdir=/usr/lib/x86_64-linux-gnu --incdir=/usr/include/x86_64-linux-gnu
--arch=amd64 --enable-gpl --disable-stripping --enable-avresample --disable-
filter=resample --enable-avisynth --enable-gnutls --enable-ladspa --enable-
libaom --enable-libass --enable-libbluray --enable-libbs2b --enable-libcaca
--enable-libcdio --enable-libcodec2 --enable-libflite --enable-libfontconfig
--enable-libfreetype --enable-libfribidi --enable-libgme --enable-libgsm
--enable-libjack --enable-libmp3lame --enable-libmysofa --enable-libopenjpeg
--enable-libopenmpt --enable-libopus --enable-libpulse --enable-librsvg
--enable-librubberband --enable-libshine --enable-libsnappy --enable-libsoxr
--enable-libspeex --enable-libssh --enable-libtheora --enable-libtwolame
--enable-libvidstab --enable-libvorbis --enable-libvpx --enable-libwavpack
--enable-libwebp --enable-libx265 --enable-libxml2 --enable-libxvid --enable-
libzmq --enable-libzvbi --enable-lv2 --enable-omx --enable-openal --enable-
opencl --enable-opengl --enable-sdl2 --enable-libdc1394 --enable-libdrm
--enable-libiec61883 --enable-nvenc --enable-chromaprint --enable-frei0r
--enable-libx264 --enable-shared
  WARNING: library configuration mismatch
  avcodec     configuration: --prefix=/usr --extra-version=0ubuntu0.1
--toolchain=hardened --libdir=/usr/lib/x86_64-linux-gnu
--incdir=/usr/include/x86_64-linux-gnu --arch=amd64 --enable-gpl --disable-
stripping --enable-avresample --disable-filter=resample --enable-avisynth
--enable-gnutls --enable-ladspa --enable-libaom --enable-libass --enable-
libbluray --enable-libbs2b --enable-libcaca --enable-libcdio --enable-libcodec2
--enable-libflite --enable-libfontconfig --enable-libfreetype --enable-
libfribidi --enable-libgme --enable-libgsm --enable-libjack --enable-libmp3lame
--enable-libmysofa --enable-libopenjpeg --enable-libopenmpt --enable-libopus
```

```
--enable-libpulse --enable-librsvg --enable-librubberband --enable-libshine
--enable-libsnappy --enable-libsoxr --enable-libspeex --enable-libssh --enable-
libtheora --enable-libtwolame --enable-libvidstab --enable-libvorbis --enable-
libvpx --enable-libwavpack --enable-libwebp --enable-libx265 --enable-libxml2
--enable-libxvid --enable-libzmq --enable-libzvbi --enable-lv2 --enable-omx
--enable-openal --enable-opencl --enable-opengl --enable-sdl2 --enable-libdc1394
--enable-libdrm --enable-libiec61883 --enable-nvenc --enable-chromaprint
--enable-frei0r --enable-libx264 --enable-shared --enable-version3 --disable-doc
--disable-programs --enable-libaribb24 --enable-liblensfun --enable-
libopencore_amrnb --enable-libopencore_amrwb --enable-libtesseract --enable-
libvo_amrwbenc
  libavutil      56. 31.100 / 56. 31.100
  libavcodec     58. 54.100 / 58. 54.100
  libavformat    58. 29.100 / 58. 29.100
  libavdevice    58.  8.100 / 58.  8.100
  libavfilter     7. 57.100 /  7. 57.100
  libavresample   4.  0.  0 /  4.  0.  0
  libswscale      5.  5.100 /  5.  5.100
  libswresample   3.  5.100 /  3.  5.100
  libpostproc    55.  5.100 / 55.  5.100
Input #0, image2, from '/media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/animations/images/20170915_023200_themis_rank_fisheye/%05d.png':
  Duration: 00:00:06.00, start: 0.000000, bitrate: N/A
    Stream #0:0: Video: png, rgba(pc), 504x720 [SAR 2835:2835 DAR 7:10], 10 fps,
10 tbr, 10 tbn, 10 tbc
Stream mapping:
  Stream #0:0 -> #0:0 (png (native) -> h264 (libx264))
Press [q] to stop, [?] for help
[libx264 @ 0x55deac50f580] using SAR=1/1
[libx264 @ 0x55deac50f580] using cpu capabilities: MMX2 SSE2Fast SSSE3 SSE4.2
AVX FMA3 BMI2 AVX2
[libx264 @ 0x55deac50f580] profile High, level 3.1
[libx264 @ 0x55deac50f580] 264 - core 155 r2917 0a84d98 - H.264/MPEG-4 AVC codec
- Copyleft 2003-2018 - http://www.videolan.org/x264.html - options: cabac=1
ref=8 deblock=1:0:0 analyse=0x3:0x133 me=umh subme=9 psy=1 psy_rd=1.00:0.00
mixed_ref=1 me_range=16 chroma_me=1 trellis=2 8x8dct=1 cqm=0 deadzone=21,11
fast_pskip=1 chroma_qp_offset=-2 threads=12 lookahead_threads=1 sliced_threads=0
nr=0 decimate=1 interlaced=0 bluray_compat=0 constrained_intra=0 bframes=3
b_pyramid=2 b_adapt=2 b_bias=0 direct=3 weightb=1 open_gop=0 weightp=2
keyint=250 keyint_min=10 scenecut=40 intra_refresh=0 rc_lookahead=60 rc=crf
mbtree=1 crf=25.0 qcomp=0.60 qpmin=0 qpmax=69 qpstep=4 ip_ratio=1.40 aq=1:1.00
Output #0, mp4, to '/media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/animations/20170915_023200_023457_themis_rank_fisheye.mp4':
  Metadata:
    encoder         : Lavf58.29.100
    Stream #0:0: Video: h264 (libx264) (avc1 / 0x31637661), yuv420p, 504x720
[SAR 1:1 DAR 7:10], q=-1--1, 10 fps, 10240 tbn, 10 tbc
    Metadata:
```

```
      encoder         : Lavc58.54.100 libx264
    Side data:
      cpb: bitrate max/min/avg: 0/0/0 buffer size: 0 vbv_delay: -1
frame=   60 fps=0.0 q=-1.0 Lsize=     204kB time=00:00:05.70 bitrate=
292.6kbits/s speed=12.7x
video:202kB audio:0kB subtitle:0kB other streams:0kB global headers:0kB muxing
overhead: 0.741242%
[libx264 @ 0x55deac50f580] frame I:1     Avg QP:18.77  size: 28768
[libx264 @ 0x55deac50f580] frame P:17    Avg QP:24.15  size:  5043
[libx264 @ 0x55deac50f580] frame B:42    Avg QP:26.92  size:  2185
[libx264 @ 0x55deac50f580] consecutive B-frames:  1.7%  6.7% 25.0% 66.7%
[libx264 @ 0x55deac50f580] mb I  I16..4: 40.7% 29.2% 30.1%
[libx264 @ 0x55deac50f580] mb P  I16..4:  0.8%  4.4%  0.8%  P16..4: 23.9%  6.3%
6.0%  0.2%  0.1%    skip:57.4%
[libx264 @ 0x55deac50f580] mb B  I16..4:  0.2%  0.8%  0.1%  B16..8: 27.2%  5.6%
1.9%  direct: 1.9%  skip:62.2%  L0:55.5% L1:38.6% BI: 5.9%
[libx264 @ 0x55deac50f580] 8x8 transform intra:56.0% inter:73.2%
[libx264 @ 0x55deac50f580] direct mvs  spatial:83.3% temporal:16.7%
[libx264 @ 0x55deac50f580] coded y,uvDC,uvAC intra: 53.6% 20.1% 16.7% inter:
10.2% 1.7% 1.5%
[libx264 @ 0x55deac50f580] i16 v,h,dc,p: 71% 22%  5%  2%
[libx264 @ 0x55deac50f580] i8 v,h,dc,ddl,ddr,vr,hd,vl,hu:  8%  8% 16%  5% 17%
12% 22%  5%  7%
[libx264 @ 0x55deac50f580] i4 v,h,dc,ddl,ddr,vr,hd,vl,hu: 13% 13% 10%  7% 13%
9% 14%  8% 13%
[libx264 @ 0x55deac50f580] i8c dc,h,v,p: 74% 10% 15%  1%
[libx264 @ 0x55deac50f580] Weighted P-Frames: Y:0.0% UV:0.0%
[libx264 @ 0x55deac50f580] ref P L0: 40.6% 16.5% 15.9%  8.5%  6.0%  5.2%  4.1%
3.1%
[libx264 @ 0x55deac50f580] ref B L0: 55.3% 18.5% 10.8%  7.7%  4.9%  2.1%  0.7%
[libx264 @ 0x55deac50f580] ref B L1: 95.8%  4.2%
[libx264 @ 0x55deac50f580] kb/s:275.01

Movie saved in /media/mike/692d5b55-e101-4c9f-a338-50bfdc97761e/asilib-
data/movies
```


(a)
0.000
30.000
60.000
270.000
90.000
0.000
180.000
THEMIS/RANK
2017-09-15 02:34:57


(b)
Mean ASI intensity [counts]
0
10
20
30
40
02:32:00
02:32:30
02:33:00
02:33:30
02:34:00
02:34:30
02:35:00
Time