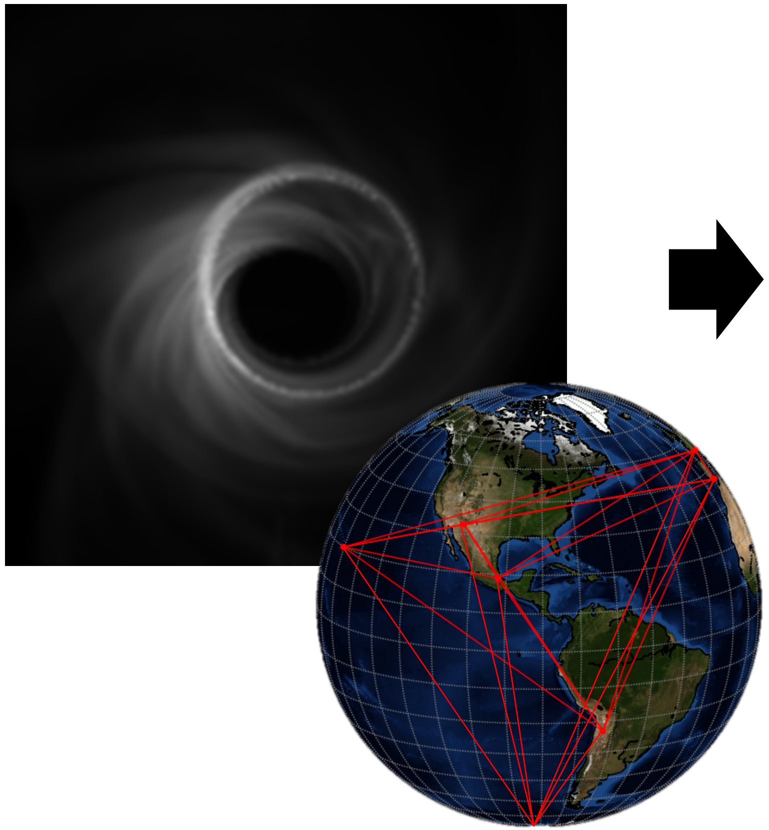
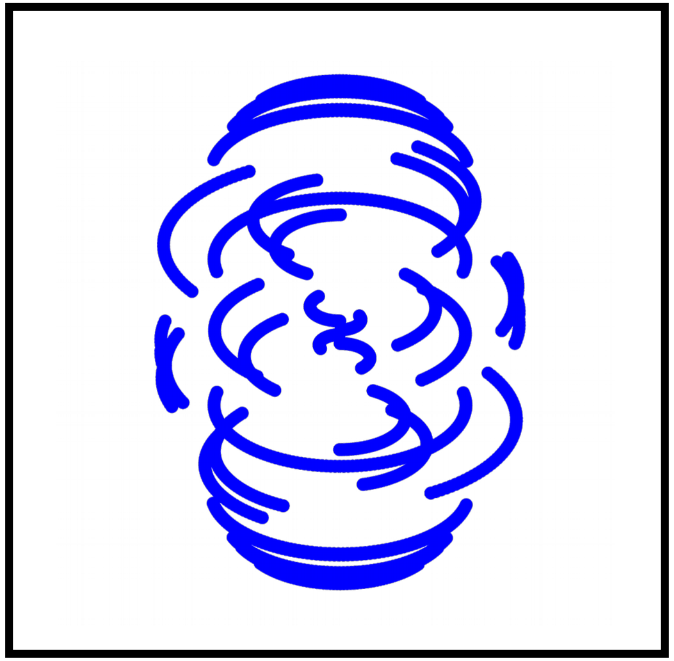
EHT Imaging Tutorial

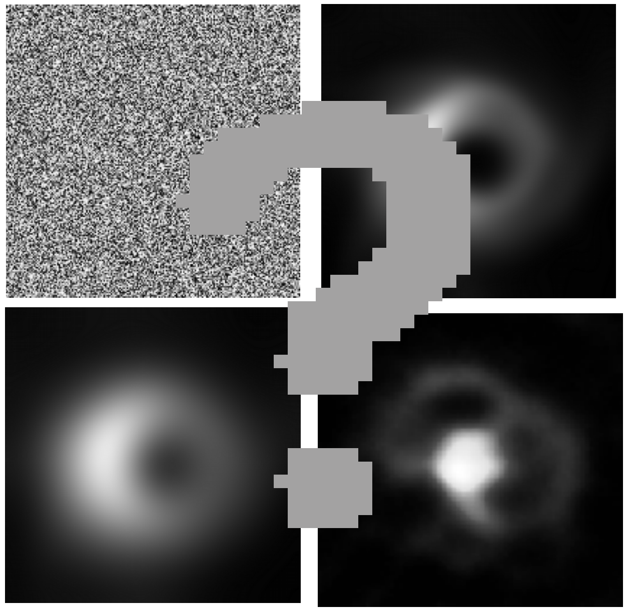
Andrew Chael & Katie Bouman

Imaging with the Event Horizon Telescope



Measurements

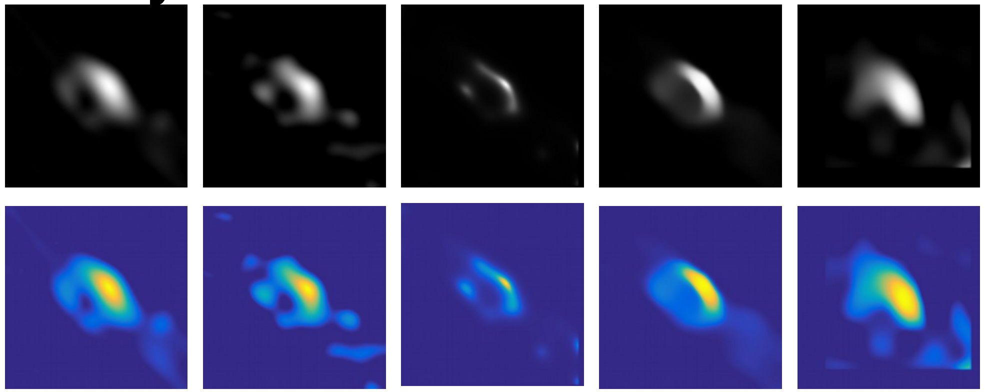


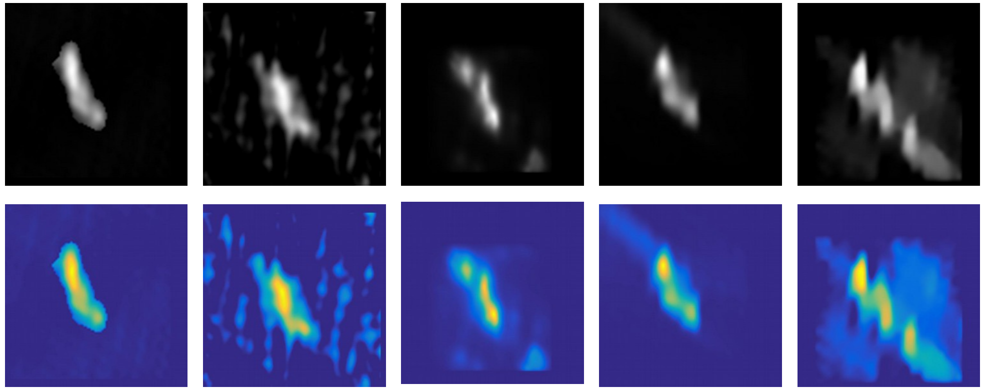


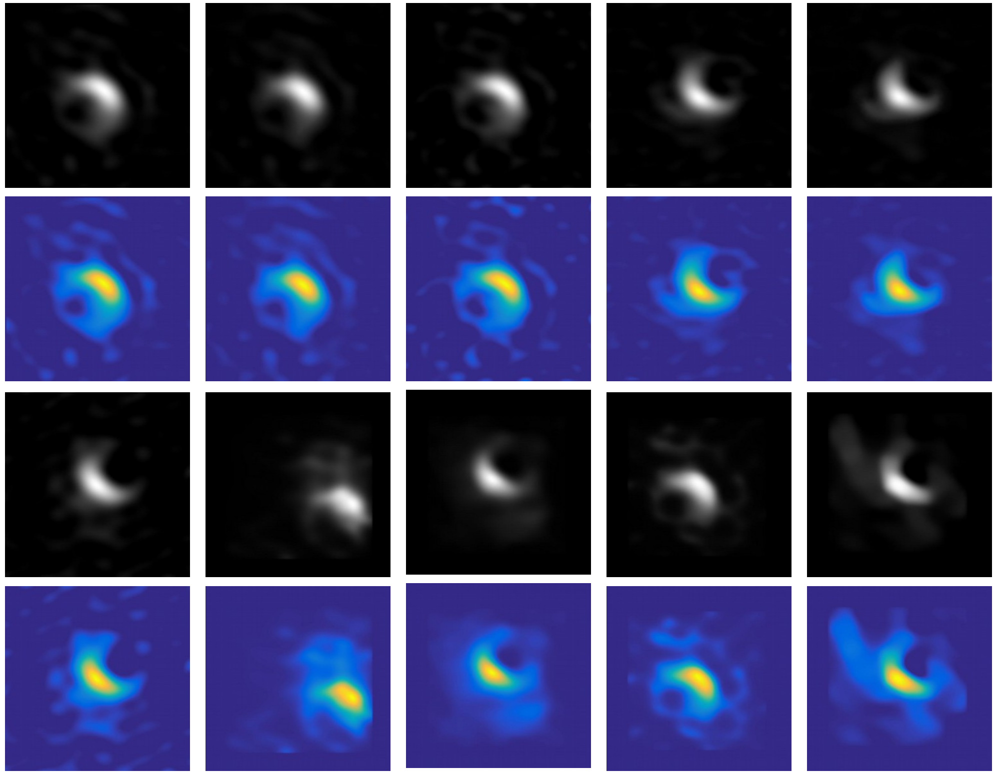
Infinite Number of Possibilities

Imaging Methods

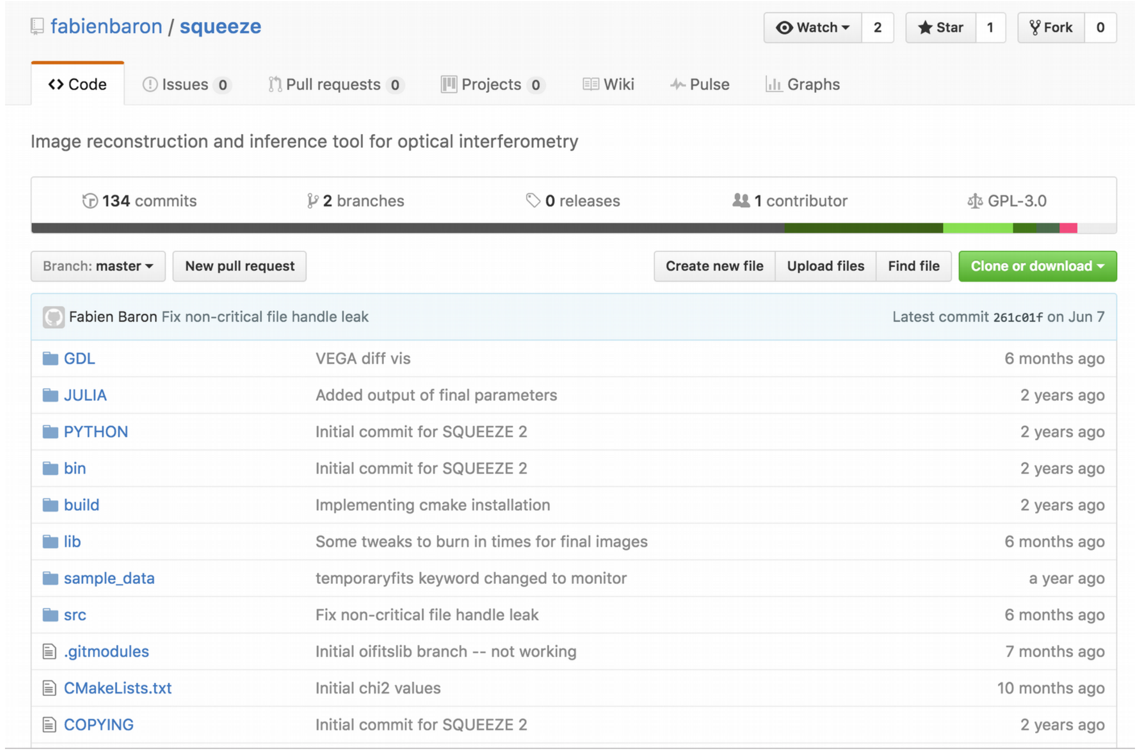
SQUEEZE, BSMEM, MEMHorizon, Sparse Imaging, CHIRP, Closur
only







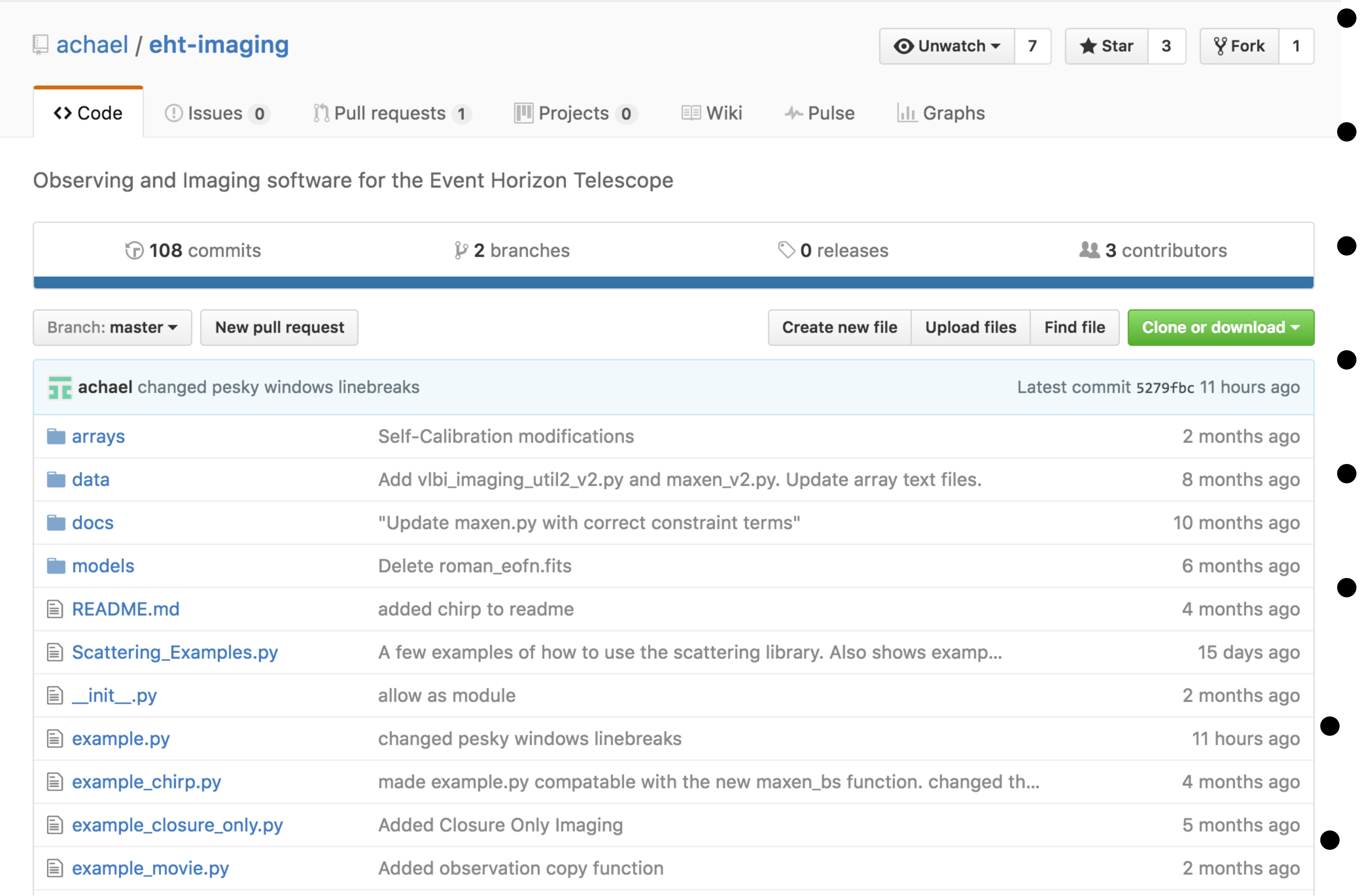
SQUEEZE – Fabien Baron



Sparse Imaging Grou

COMING SOON!

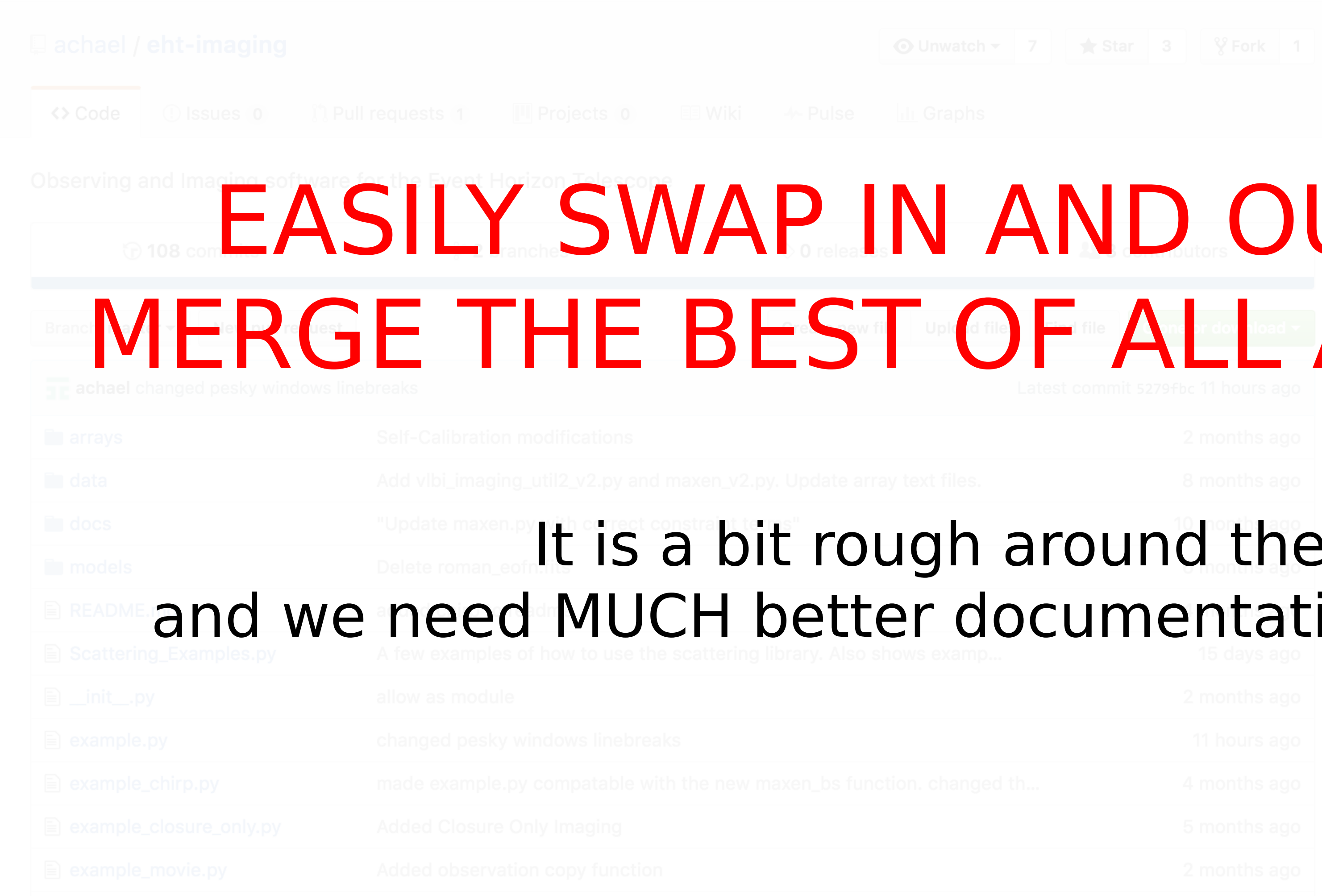
eht-imaging Python Library



- MEMHorizon
- Sparse Imaging Ideas
- CHIRP
- Closure-only Imaging
- Time-Variable Imaging
- Scattering Mitigation

- **Generate Data**
- **Plot Data/Results**

eht-imaging Python Library

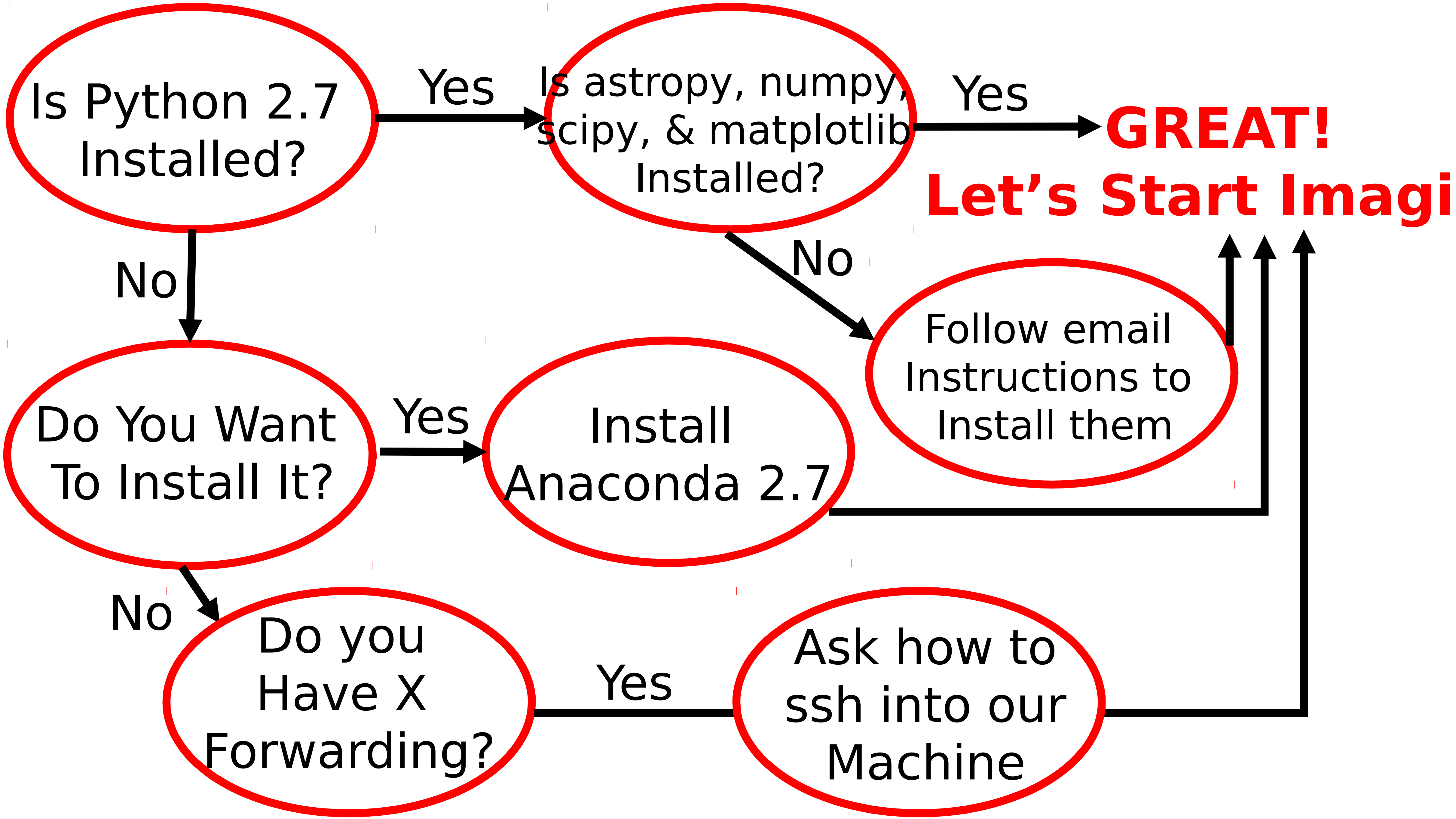


- MEMHorizon
- Sparse Imaging Ideas
- CHIRP
- Closure-only Imaging
- Time-Variable Imaging
- Scattering Mitigation
- **Generate Data**
- **Plot Data**

EASILY SWAP IN AND OUT IDEAS & MERGE THE BEST OF ALL ALGORITHMS

It is a bit rough around the edges…
and we need MUCH better documentation than we have now

Step 0: Installing Python 2.7 and Necessary Dependencies


Is Python 2.7 Installed?
Yes
Is astropy, numpy, scipy, & matplotlib Installed?
Yes
GREAT!
Let's Start Imagi
No
No
Follow email Instructions to Install them
Do You Want To Install It?
Yes
Install Anaconda 2.7
No
Do you Have X Forwarding?
Yes
Ask how to ssh into our Machine

Step 1: Exploring the VLBI Imaging Website

VLBI Imaging Website vlbiimaging.csail.mit.edu

VLBI Reconstruction Dataset

A Dataset Designed to Train and Test Very Long Baseline Interferometry Image Reconstruction Algorithms

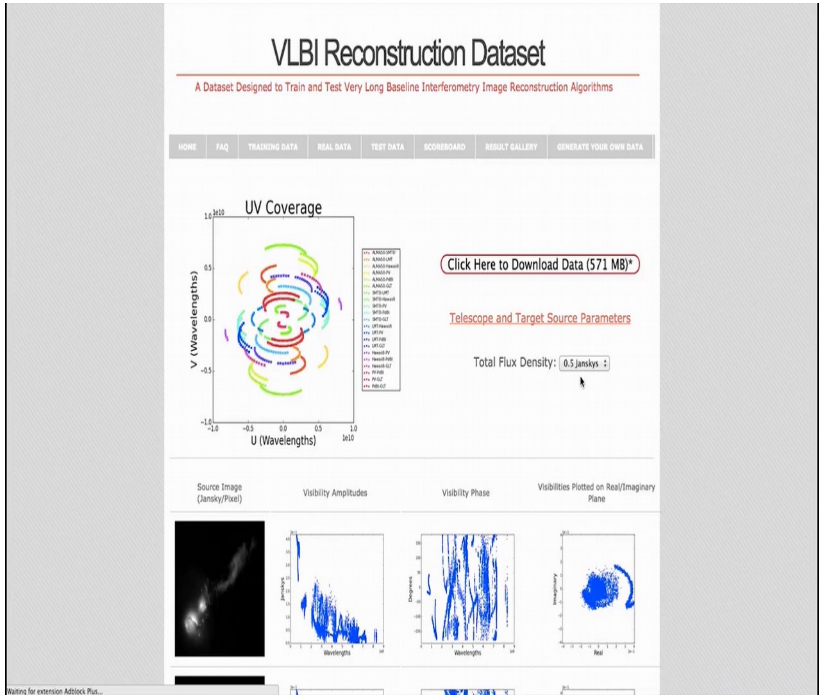
HOME | FAQ | TRAINING DATA | REAL DATA | TEST DATA | SCOREBOARD | RESULT GALLERY | GENERATE YOUR DATA | EHT IMAGING CHALLENGE

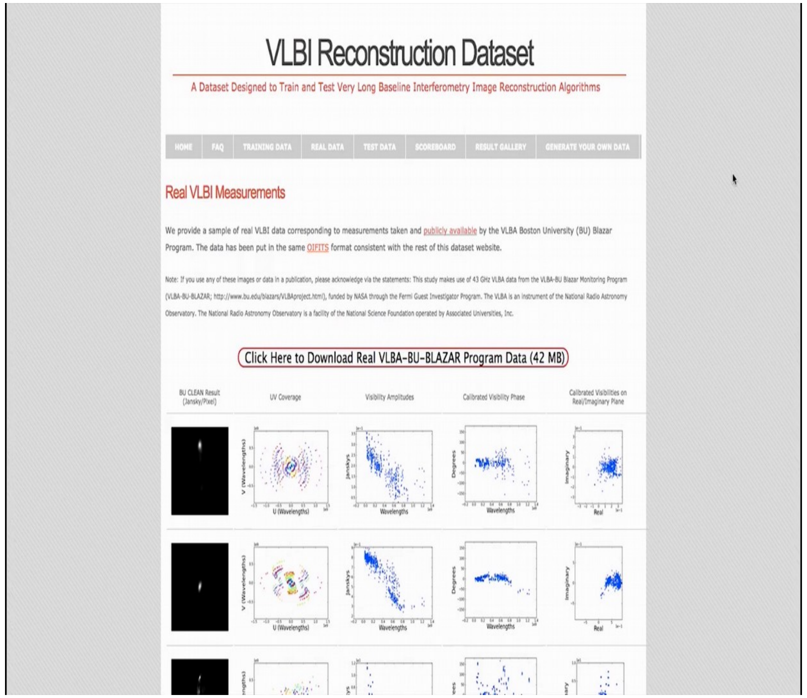
Welcome to the VLBI Reconstruction Dataset!

The goal of this website is to provide a testbed for developing new VLBI reconstruction algorithms. By supplying a large set of easy to understand training and testing data, we hope to make the problem more accessible to those less familiar with the VLBI field. Specifically, this website contains a:

- Large set of synthetic training data for many different VLBI arrays and targets
- Set of real data measurements provided in the same standard format
- Standardized data set for testing VLBI Image Reconstruction Algorithms
- Online quantitative evaluation of algorithm performance on simulated testing data
- Qualitative comparison of algorithm performance on the reconstruction of real data
- Online form to easily simulate realistic data using your own image and telescope parameters

VLBI Imaging Website vlbiimaging.csail.mit.edu

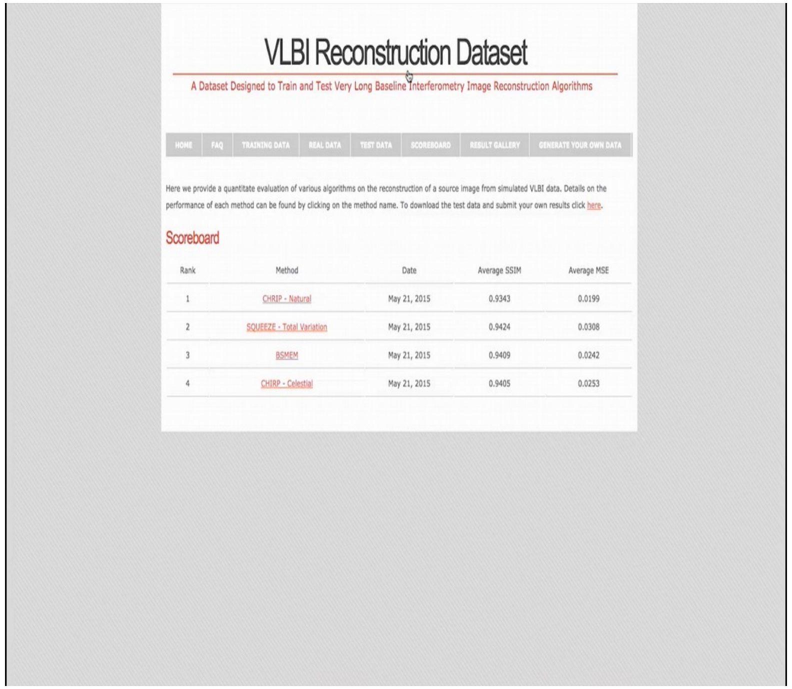




Standardized dataset of real & synthetic data

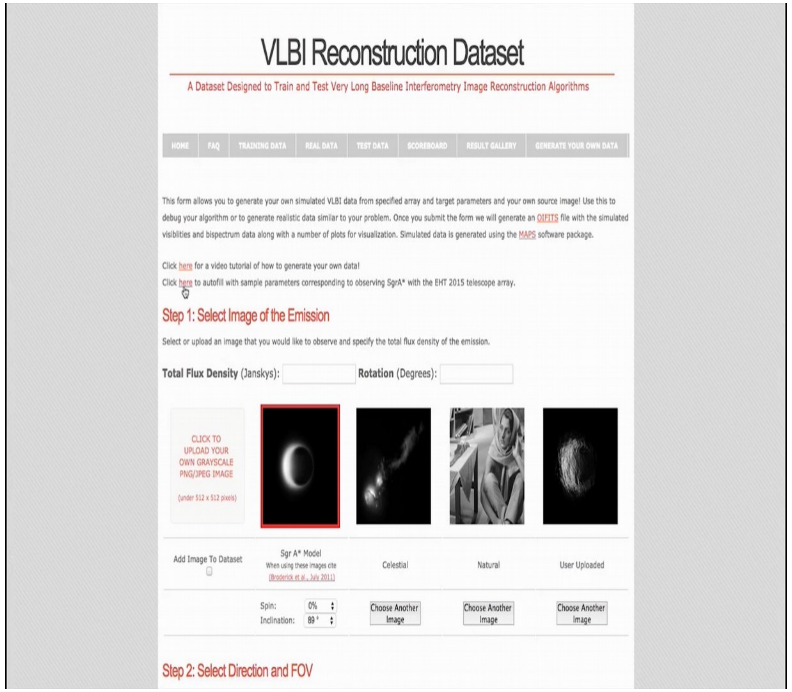
Over 5000 synthetic measurement sets:
14 Array Configurations, 96 Source Images, 4 Noise Levels

VLBI Imaging Website vlbiimaging.csail.mit.edu



Automatic Quantitative and Qualitative Evaluation

VLBI Imaging Website vlbiimaging.csail.mit.edu

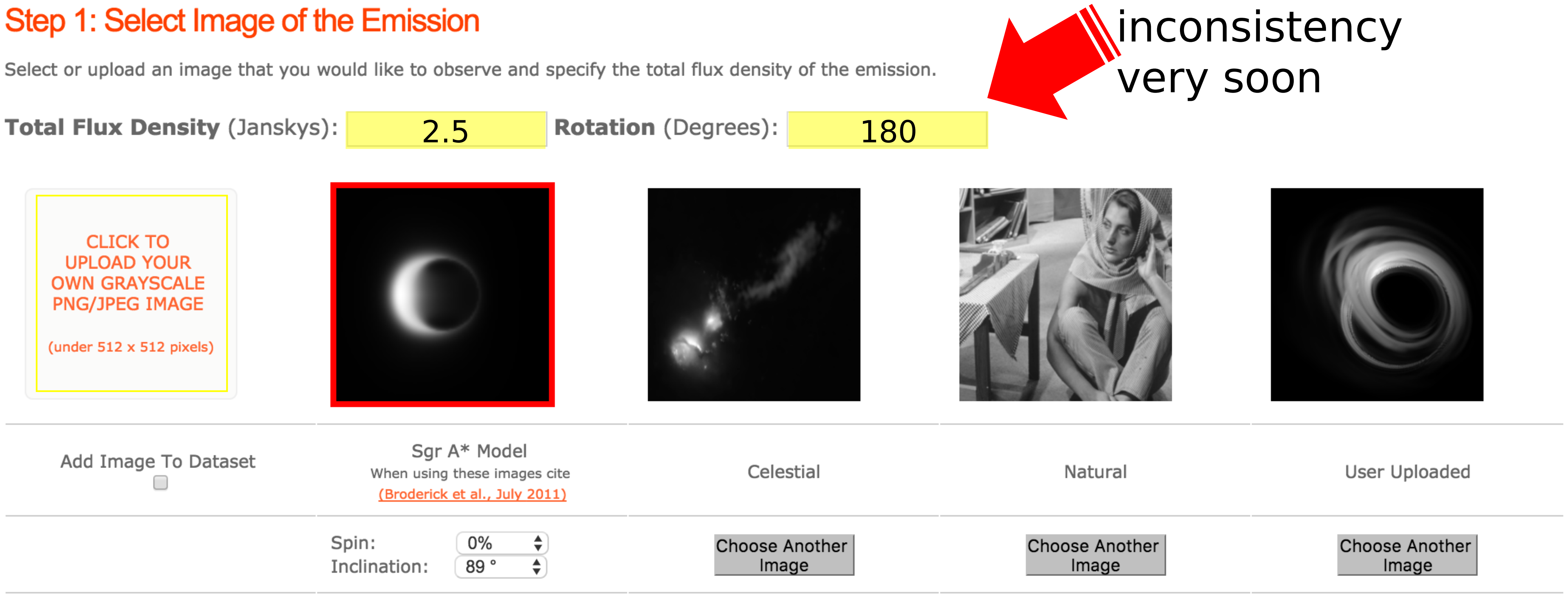


Online form to easily simulate realistic data using user-specified parameters

Step 1: Generating Data on the VLBI Imaging Website

Selecting/Uploading an Image

Sorry about this!
We will fix the
inconsistency
very soon



Selecting Target Location and Field of View

Step 2: Select Direction and FOV

Identify the direction to the target source. Right ascension should be in the form HH:MM:SS.SS for hours, minutes, and seconds and declination should be in the form DD:MM:SS.SS for degrees, arcminutes, and arcseconds. Field of view is specified in arcseconds. Warning: You must choose coordinates such that your region will be observable from your observatory site (the first telescope you specify below) at the start time that you specify, otherwise the resulting output will be incorrect.

Field Of View Center: Right Ascension (HH:MM:SS.SS) 17:45:40.041 Declination(DD:MM:SS.SS) -29:00:28.118

Field Of View Size: Right Ascension (arcseconds) 0.00016 Declination (arcseconds) 0.00016

Selecting Telescopes

Step 3: Specify Telescope Array

Add the telescope locations and intrinsic parameters that you would like to use to simulate data

Initilization: Select a pre-loaded telescope
Name: Unique name for each telescope station (up to 12 characters)
East Longitude/Latitude: East longitude and latitude of the array center. For locations less than 180 degrees west of Greenwich a minus sign should precede the longitude entry.
X/Y/Z Position: Absolute X, Y, Z coordinates of each station (in meters) relative to the center of the Earth
Lower/Upper Elevation: Lower and upper elevation limits of the of the antenna in degrees
SEFD: System equivalent flux denisty of the antenna
Diameter: Antenna diameter in meters

	Initilization	Name	East Longitude	Latitude	X-Position	Y-Position	Z-Position
☐	ALMA	ALMA	-67:45:11.4	-23:01:09.4	2225037.1851	-5441199.162	-2479303.4629
☐	SMT	SMT	-109:52:19	32:42:06	-1828796.2	-5054406.8	3427865.2
☐	LMT	LMT	-97:18:53	18:59:06	-768713.9637	-5988541.7982	2063275.9472
☐	SMA	SMA	-155:28:40.7	19:49:27.4	-5464523.4	-2493147.08	2150611.75
☐	PV	PV	-3:23:33.8	37:03:58.2	5088967.9	-301681.6	3825015.8
☐	PDB	PDB	05:54:28.5	44:38:02.0	4523998.4	468045.24	4460309.76
☐	SPT	SPT	-000:00:00.0	-90:00:00	0	0	-6359587.3

ADD TELESCOPE DELETE SELECTED

Data Collection Settings

Step 4: Specify Date and Time Data is Collected

Specify the time of when you would like measurments to be taken, and the time interval between measurements.

Start Time: Specify the time of your first observation in Universal Time (UT). The required format is "YYYY:ddd:hh:mm:ss" where YYYY is the year, ddd is the day number (e.g., December 31 is day 365); hh is the UT hour, mm is the UT minute, and ss is the UT second.

Scan Duration: The length of a continuous scan in seconds

Interval Length: The time in seconds between successive scans

Number of Samples: The number of successive scans of this type

	Start Time (UT)	Scan Duration (seconds)	Interval Length (seconds)	Number of Samples
☐	2017:95:00:00:00	12	600	100

ADD DATA COLLECTION DELETE SELECTED

Data Collection Settings

Step 5: Specify Collection Parameters

Specify the center frequency and width of the observing channel in MHz.

Center Frequency (MHz): 227297 Bandwidth (MHz): 4096

Specify your integration time in seconds (sometimes referred to as “dump time” or “record length”). This is not the total duration of your observation, but rather the sampling and recording interval of the data.

Integration Time (seconds): 60

What Kinds of Noise Can We Add?

Thermal Noise

SEFDs

$$\sigma = \frac{1}{0.88}\sqrt{\frac{\rho_1 \rho_2}{\Delta\nu\tau}}$$

Effect of 2-Bit Quantization

Integration Time

Bandwidth

Atmospheric Phase Error

$\cos(\omega t + \phi_1)$ $\cos(\omega t + \phi_2)$

Frequency Measurement:

$$\propto e^{i(\phi_1 - \phi_2)}$$

Systematic Gain Error

- Variation in Estimated SEFD
- Variation in SEFD over Time due to Opacity Changes

Selecting Types of Noise Added

Let’s JUST add Thermal Noise

Step 6: Add Noise and Generate Data

- [ ] Simulate Without ANY Noise
- [x] Simulate Without Atmospheric Phase Errors
- [x] Simulate Without 5 % Gain Error

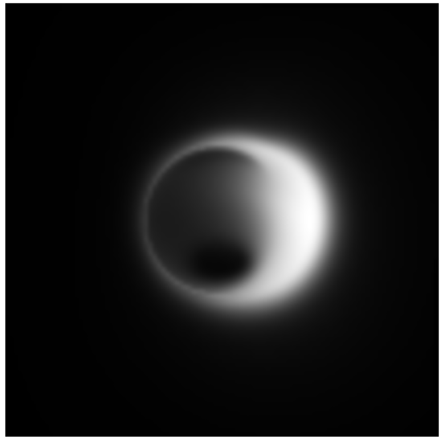
And now we generally generate our data....

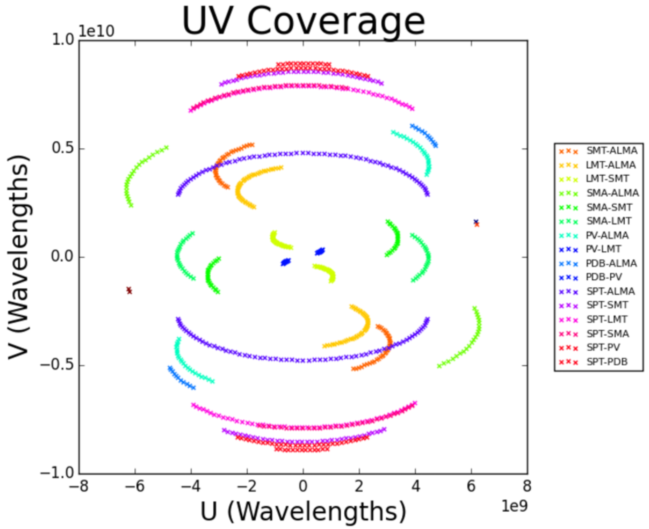
But if so many people submit at the same time we will probably bog down the machine....

So for now, please download pre-computed data and later you can generate it yourself

vlbiimaging.csail.mit.edu/myDataResults_6312

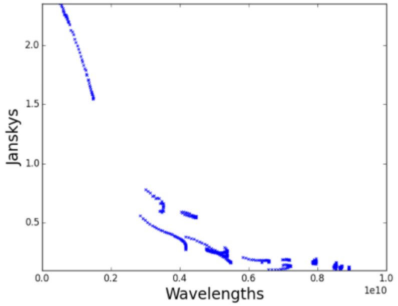
Click Here to Download Data

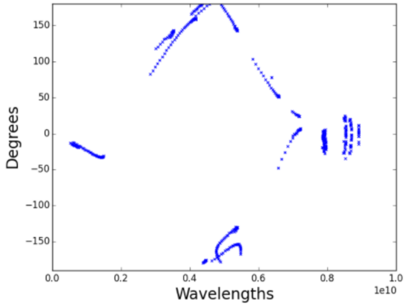


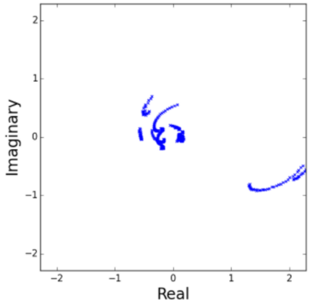


Click Here to View the Telescope and Target Source Parameters

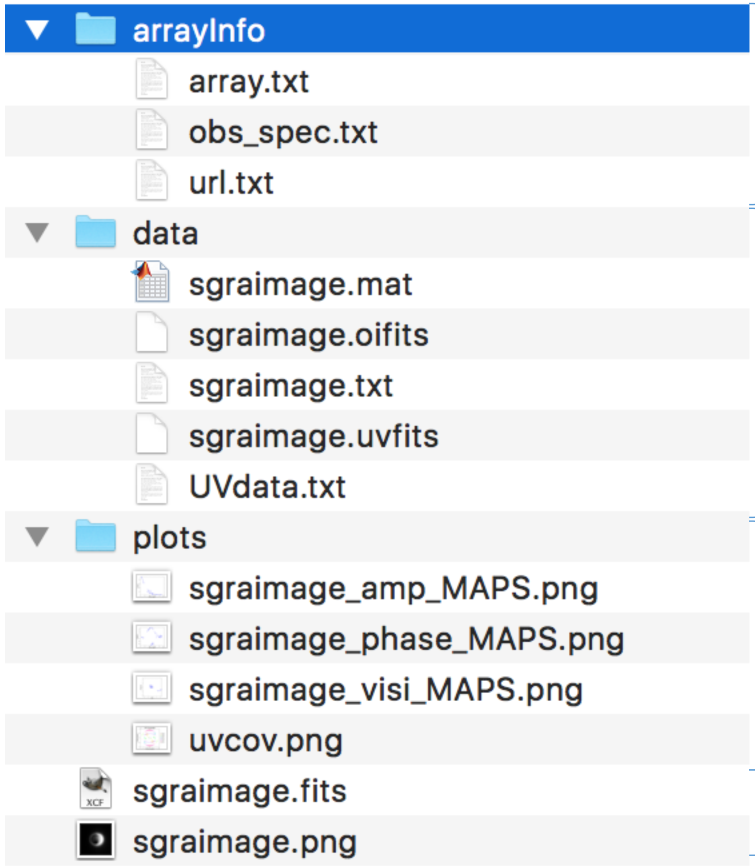
Visibility Amplitudes	Visibility Phase	Plotted on Real Imaginary Plane







What does the downloaded zip file provide?



Information to Reproduce Data

Data in a number of formats

Plots to help you understand the dat

Original Images in FITS and PNG

What are these data formats?

What are the data formats and how do I use them?

We use OIFITS, MAT, and ASCII data formats. We describe OIFITS and MAT below in detail.

OIFITS

The primary data format that we have chosen to use is OIFITS. OIFITS is a standard for exchanging data for Optical (Visible/IR) Interferometry, and is based on the FITS Standard. Since mm/sub-mm VLBI shares a lot of similarities to optical interferometry, this format is better suited for mm/sub-mm measurements than UVFITS. More information on the OIFITS format can be found here. We list the variables described in this paper in the tables below.

We provide a number of tools that may be useful in reading and writting in the OIFITS format:

- Paul Boley has written a OIFITS Python module that you can download here
- Python code that can be used to write an OIFITS file from an output MAPS text file can be downloaded here
- Python code by Andrew Chael to extract information from OIFITS and write it to a text file can be downloaded here

OI_T3 Variables	Description	Units
T3AMP	Triple-product/Bispectrum amplitude	$Jansky^3$
T3PHI	Triple-product phase	Degrees
T3AMPERR	Standard deviation of error in triple product amplitude	$Jansky^3$
T3PHIERR	Standard deviation of error in phase	Degrees
U1COORD	u coordinate of baseline AB of the triangle	meters
U2COORD	u coordinate of baseline BC of the triangle	meters
V1COORD	v coordinate of baseline AB of the triangle	meters
V2COORD	v coordinate of baseline BC of the triangle	meters
STA_INDEX	Station numbers contributing to the data	
INT_TIME	Integration time	seconds
MJD	Modified Julian Date	

Step 3: Loading and Inspecting Data

In an ipython window:

```
import numpy as np

import ehtim as eh
```

Load the observation file we generated

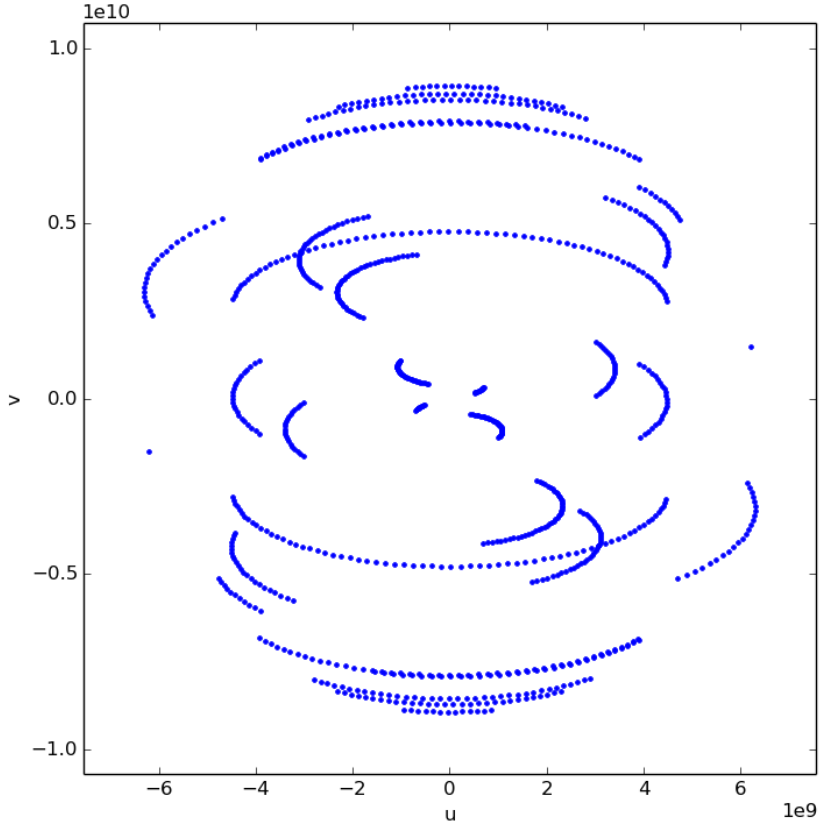
```
obs = eh.obsdata.load_uvfits('./data/sgraimage.uvfits')
```

Can also load custom text format, oifits, and MAPS output

Look at plots! UV coverage

```
obs.plotall('u','v', conj=True)
```

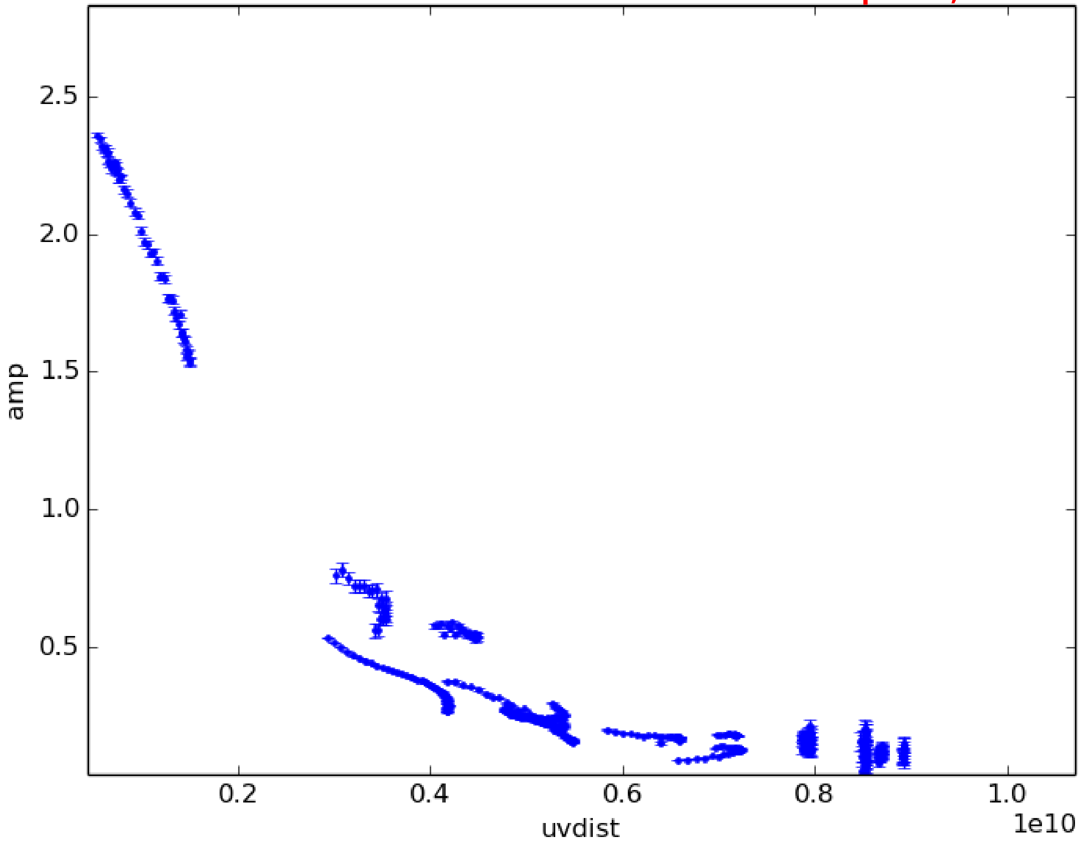
Shows both u,v and -u,-v



Look at plots: Visibility amplitudes

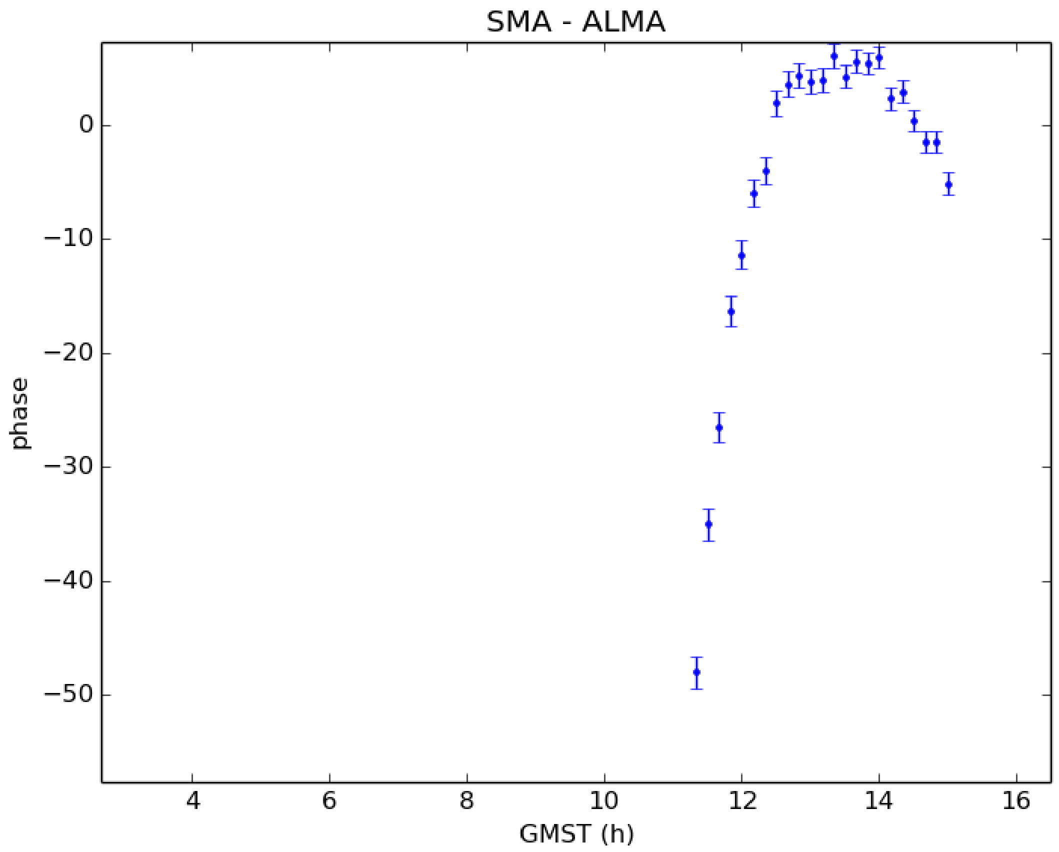
```
obs.plotall('uvdist’, ‘amp’)
```

Other possible fields include “snr”, “sigma”, “qamp”, ”uamp”,” vamp” , “m”



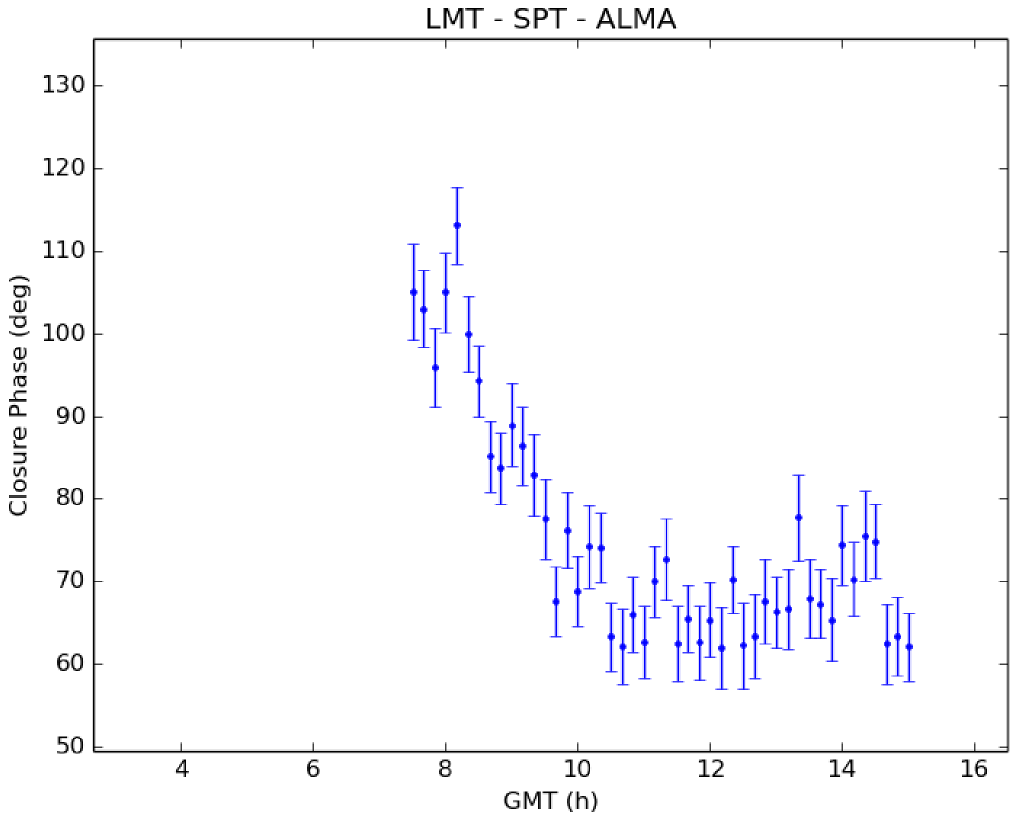
Look at plots: Baseline phase over time

```
obs.plot_bl('SMA','ALMA', 'phase')
```



Look at plots: Closure phase over time

```
obs.plot_cphase('LMT', 'SPT', 'ALMA')
```



Take a look at the dirty beam and clean beam

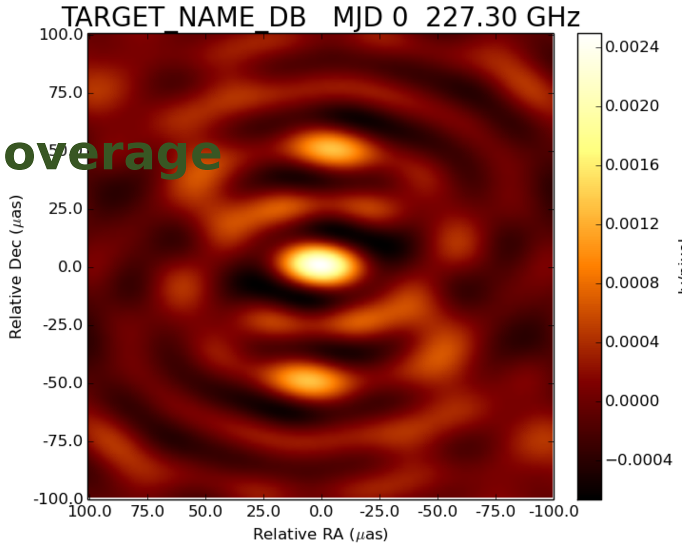
Image Parameters

```
npix = 128
fov = 200*vb.RADPERUAS
```

Dirty Beam

```
dbeam = obs.dirtybeam(npix, fov)
dbeam.display()
```

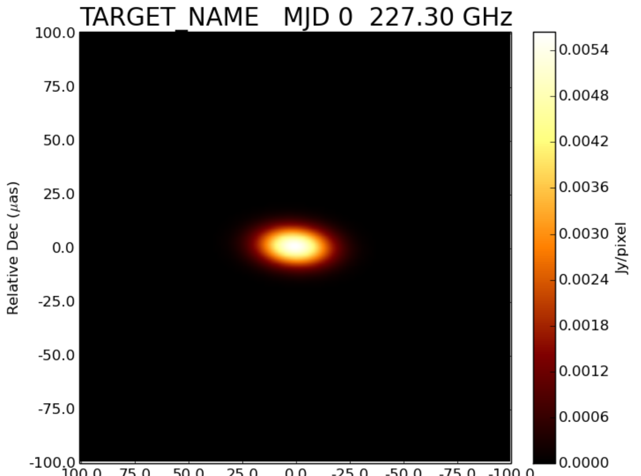
FT of the sparse u-v coverage



Clean Beam

```
cbeam = obs.cleanbeam(npix,fov)
cbeam.display()
```

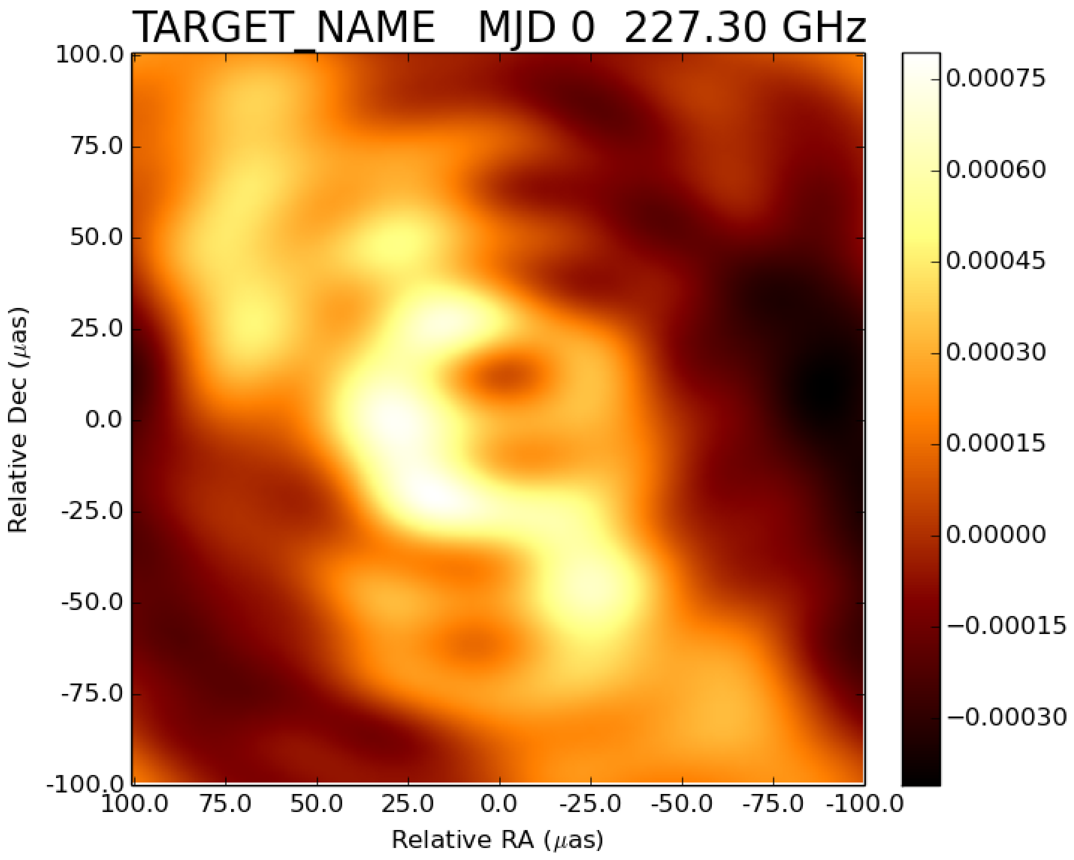
Gaussian fit to the beam component



Take a look at the dirty image

```
dim = obs.dirtyimage(npix, fov)
dim.display()
```

Sky Image convolved with Dirty Beam



What is the array resolution?

Clean Beam Gaussian parameters

```
beamparams = obs.fit_beam()
```

(FWHM_max, FWHM_min, Position angle)

Save for use in restoring beam convolution

“Maximum” resolution

```
res = obs.res()
```

1/longest baseline – can use in circular restoring beam

Step 4: Produce an Image

Generate a prior image

Image Parameters

```
npix = 128
fov =  200*vb.RADPERUAS
```

Gaussian Prior

```
zbl = 2.5
prior_fwhm = 100*vb.RADPERUAS
gaussparams = (prior_fwhm, prior_fwhm, 0.0)
```

The zero baseline flux

FWHM of our circular Gaussian Prior

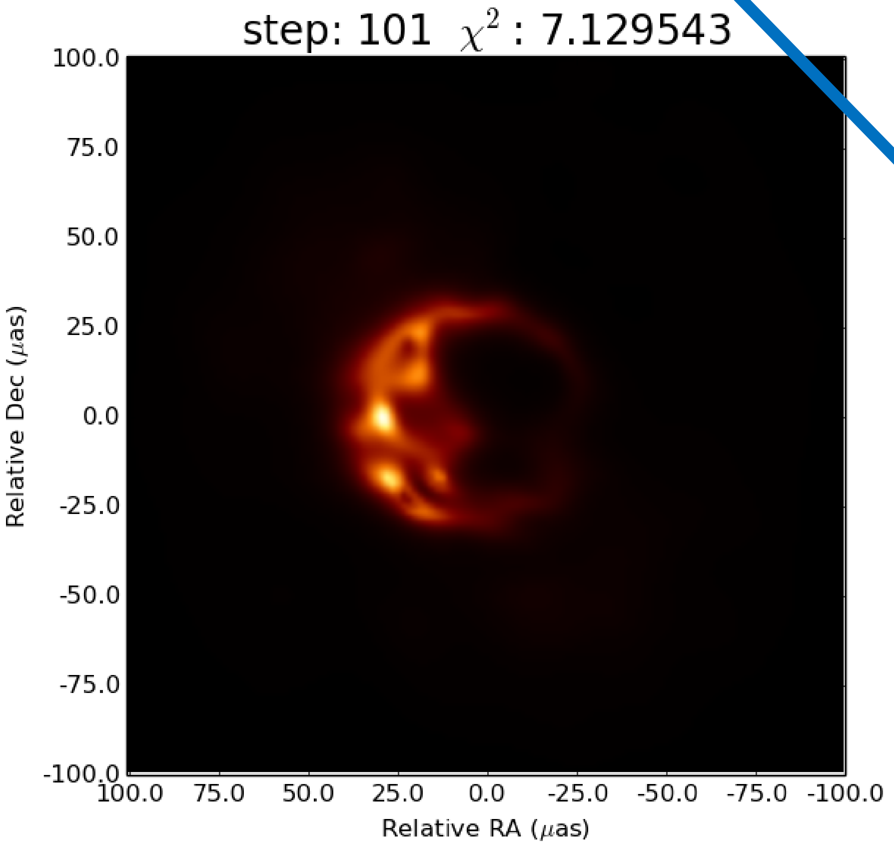
```
emptyprior = eh.image.make_square(obs, npix, fov)
gaussprior = emptyprior.add_gauss(zbl, gaussparams)
gaussprior.display()
```

Use MEM with complex visibilities

Initial Image **Prior Image** **Total flux constraint**

```
out = eh.imager_func(obs, gaussprior, gaussprior, zbl,
                     d1=”vis”, alpha_d1=50, s1="gs",maxit=100)
```

Data Term & Weight



# of iterations

Regularizer Type other options: “tv”, “l1”,…

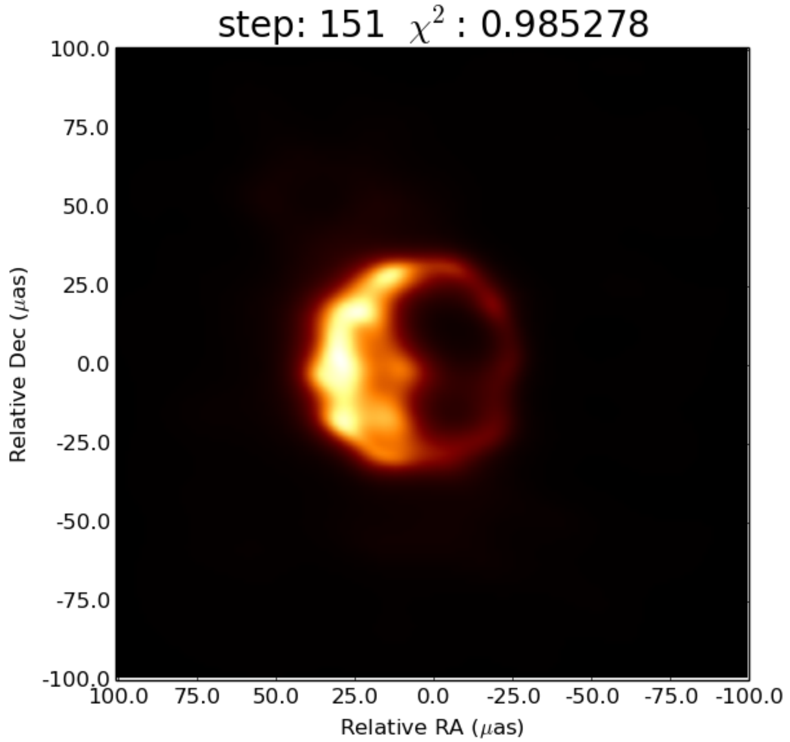
Blur and restart

Fractional beam size

```
outblur = out.blur_gauss(beamparams, 0.5)
```

We decreased data weight to prevent overfitting

```
out = outblur
out = eh.imager_func(obs, out, out,zbl, d1=”vis”, alpha_d1=10 ,s1="gs", maxit=150)
```



Final images – save to

Final “restored” image

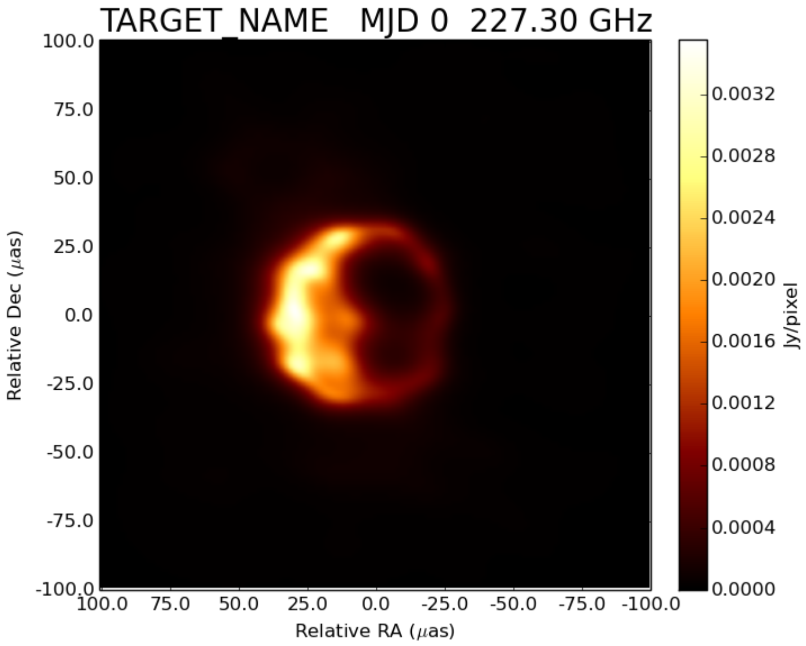
```
outblur = out.blur_gauss(beamparams, 0.5)
```

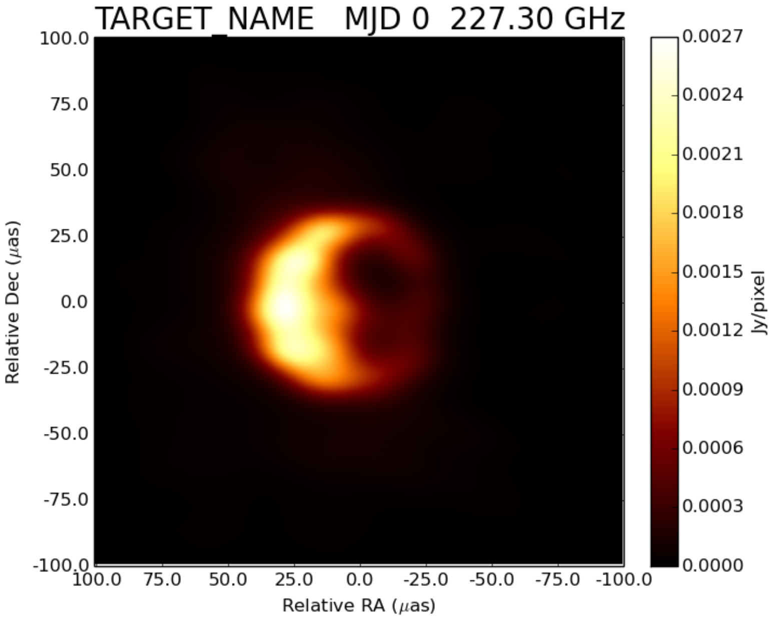
Display results

```
out.display()
outblur.display()
```

Save to FITS

```
imageout.save_fits('./sgraim.fits')
outblur.save_fits('./sgraim_blur.fits')
```

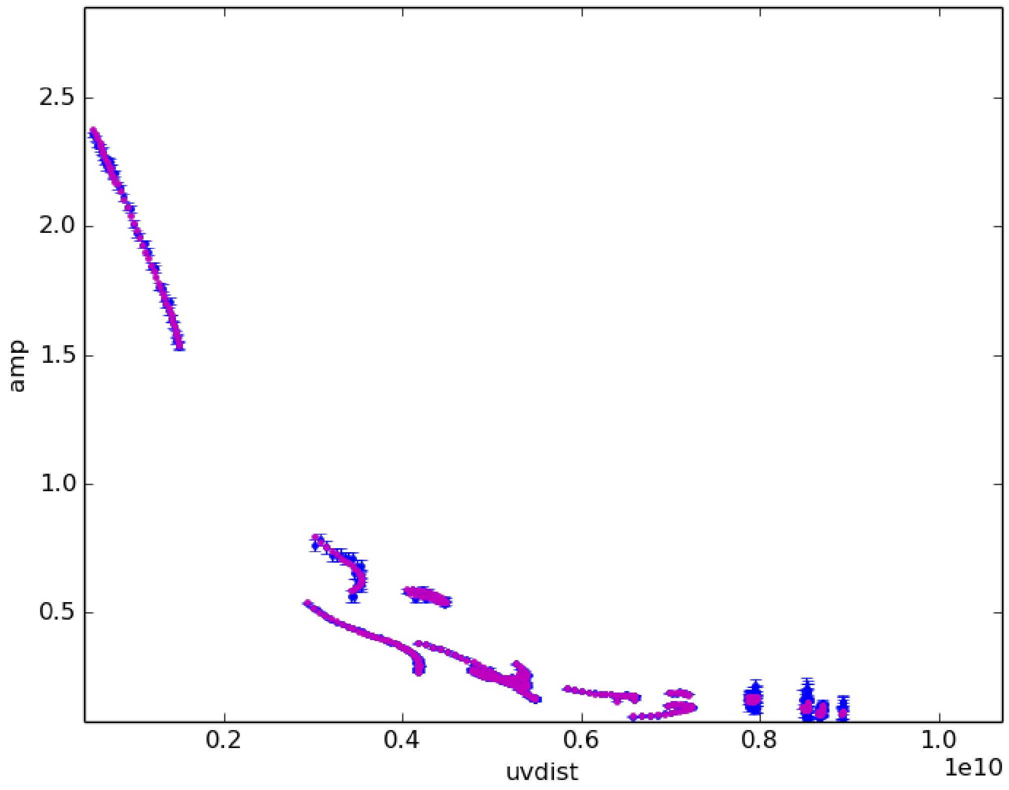




Look at fit to data - Amplitudes

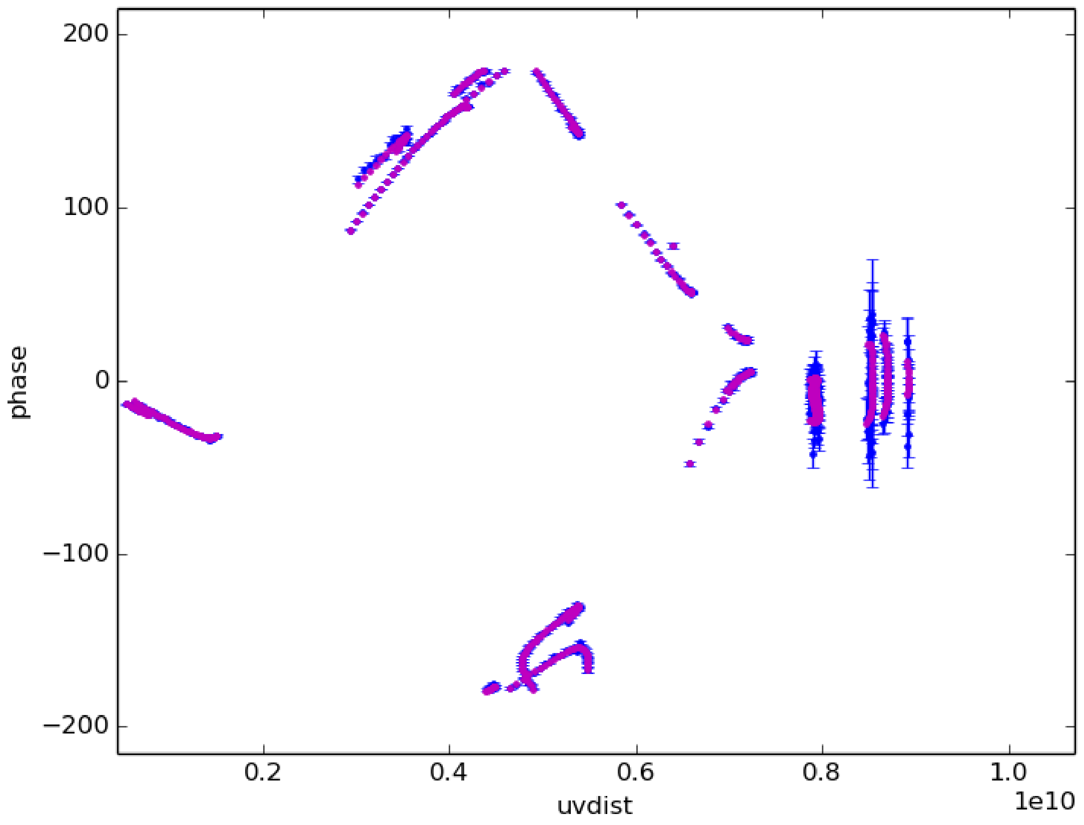
```
eh.plotting.comp_plots.plotall_obs_im_compare(obs, out, "uvdist", "amp")
```

`comp_plots.py`
has functions to overplot data from different observations



Look at fit to data - Phases

```
eh.plotting.comp_plots.plotall_obs_im_compare(obs, out, "uvdist", "phase")
```



Step 5: Generating Data with Atmospheric Noise

Selecting/Uploading an Image

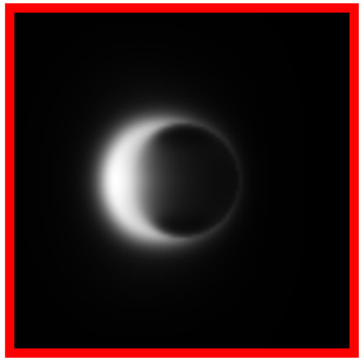
Step 1: Select Image of the Emission

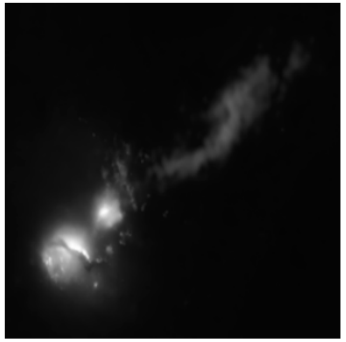
Select or upload an image that you would like to observe and specify the total flux density of the emission.

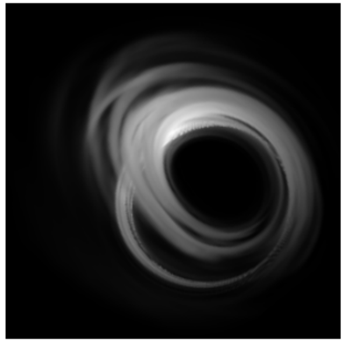
Total Flux Density (Janskys): 1.0 **Rotation** (Degrees): 180

CLICK TO UPLOAD YOUR OWN GRAYSCALE PNG/JPEG IMAGE

(under 512 x 512 pixels)







Add Image To Dataset	Sgr A* Model When using these images cite (Broderick et al., July 2011)	Celestial	Natural	User Uploaded
	Spin: 0% Inclination: 89 °	Choose Another Image	Choose Another Image	Choose Another Image

Selecting Target Location and Field of View

Step 2: Select Direction and FOV

Identify the direction to the target source. Right ascension should be in the form HH:MM:SS.SS for hours, minutes, and seconds and declination should be in the form DD:MM:SS.SS for degrees, arcminutes, and arcseconds. Field of view is specified in arcseconds. Warning: You must choose coordinates such that your region will be observable from your observatory site (the first telescope you specify below) at the start time that you specify, otherwise the resulting output will be incorrect.

Field Of View Center: Right Ascension (HH:MM:SS.SS) 12:30:49.42338
2 Declination(DD:MM:SS.SS) 12:23:28.04366

Field Of View Size: Right Ascension (arcseconds) 0.00016 Declination (arcseconds) 0.00016

Selecting Telescopes

Step 3: Specify Telescope Array

Add the telescope locations and intrinsic parameters that you would like to use to simulate data

Initilization: Select a pre-loaded telescope
Name: Unique name for each telescope station (up to 12 characters)
East Longitude/Latitude: East longitude and latitude of the array center. For locations less than 180 degrees west of Greenwich a minus sign should precede the longitude entry.
X/Y/Z Position: Absolute X, Y, Z coordinates of each station (in meters) relative to the center of the Earth
Lower/Upper Elevation: Lower and upper elevation limits of the of the antenna in degrees
SEFD: System equivalent flux denisty of the antenna
Diameter: Antenna diameter in meters

	Initilization	Name	East Longitude	Latitude	X-Position	Y-Position	Z-Position
	ALMA	ALMA	-67:45:11.4	-23:01:09.4	2225037.1851	-5441199.162	-2479303.4629
	SMT	SMT	-109:52:19	32:42:06	-1828796.2	-5054406.8	3427865.2
	LMT	LMT	-97:18:53	18:59:06	-768713.9637	-5988541.7982	2063275.9472
	SMA	SMA	-155:28:40.7	19:49:27.4	-5464523.4	-2493147.08	2150611.75
	PV	PV	-3:23:33.8	37:03:58.2	5088967.9	-301681.6	3825015.8
	PDB	PDB	05:54:28.5	44:38:02.0	4523998.4	468045.24	4460309.76
	SPT	SPT	-000:00:00.0	-90:00:00	0	0	-6359587.3

ADD TELESCOPE | DELETE SELECTED

Data Collection Settings

Step 4: Specify Date and Time Data is Collected

Specify the time of when you would like measurments to be taken, and the time interval between measurements.

Start Time: Specify the time of your first observation in Universal Time (UT). The required format is "YYYY:ddd:hh:mm:ss" where YYYY is the year, ddd is the day number (e.g., December 31 is day 365); hh is the UT hour, mm is the UT minute, and ss is the UT second.

Scan Duration: The length of a continuous scan in seconds

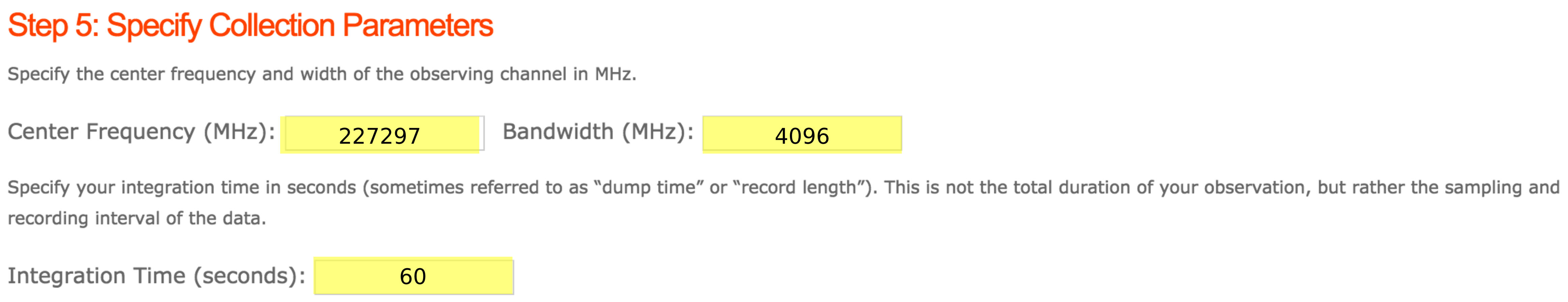
Interval Length: The time in seconds between successive scans

Number of Samples: The number of successive scans of this type

	Start Time (UT)	Scan Duration (seconds)	Interval Length (seconds)	Number of Samples
☐	2017:95:00:00:00	12	600	100

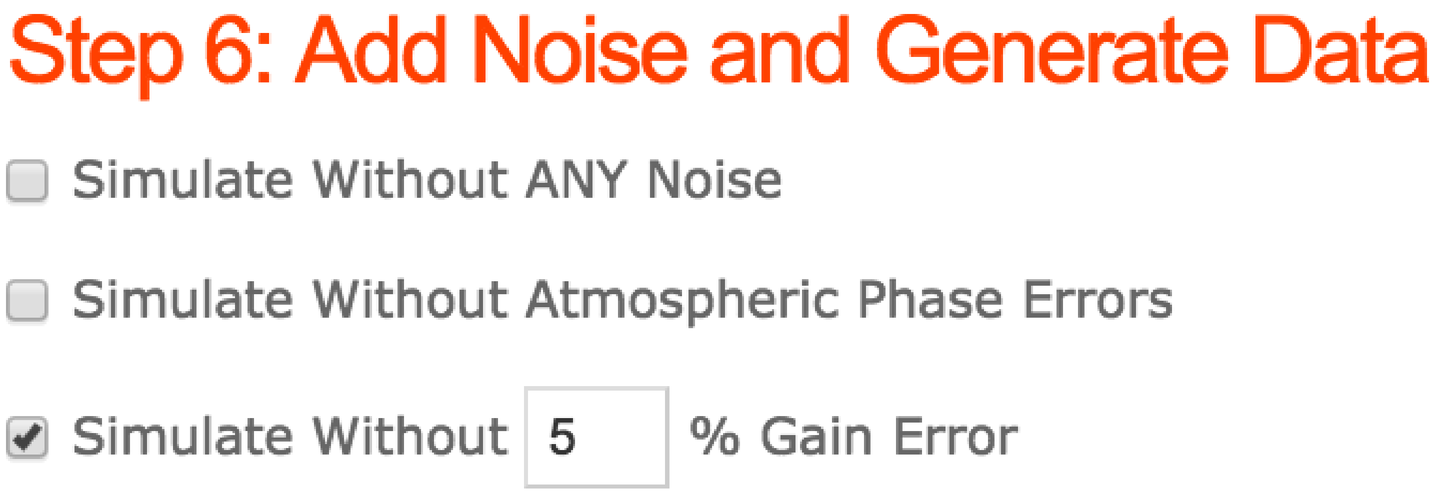
ADD DATA COLLECTION   DELETE SELECTED

Data Collection Settings



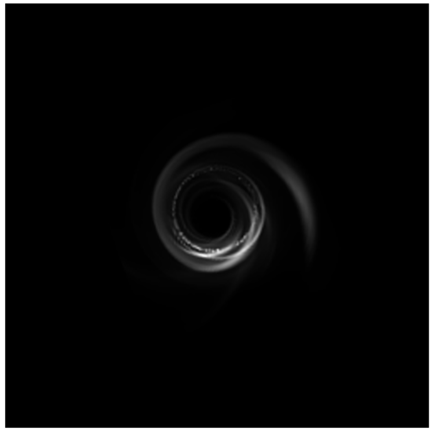
Selecting Types of Noise Added

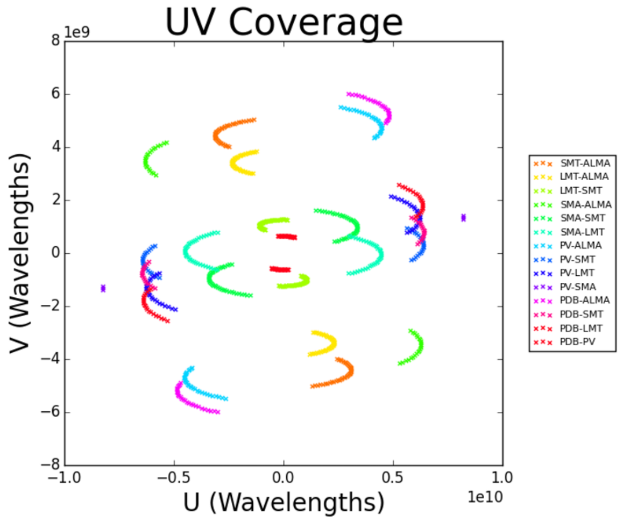
Let’s add Thermal & Atmospheric Noise



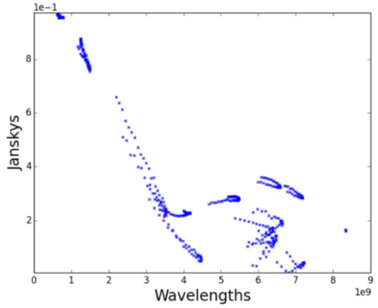
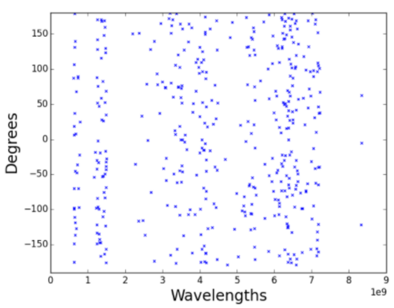
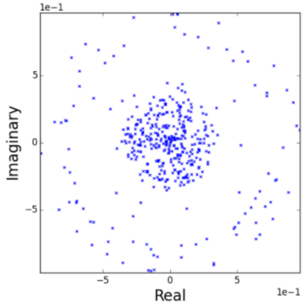
vlbiimaging.csail.mit.edu/myDataResults_3593

Click Here to Download Data





Click Here to View the Telescope and Target Source Parameters

Visibility Amplitudes	Visibility Phase	Plotted on Real Imaginary Plane
		

Step 7: Image with Closure Phase

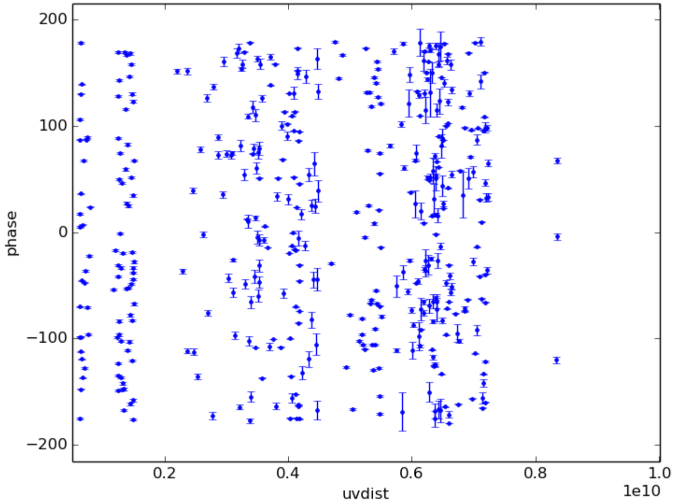
Look at the phase errors

Load the data

```
obs = eh.obsdata.load_uvfits('./data/m87image.uvfits
```

Baseline Phases

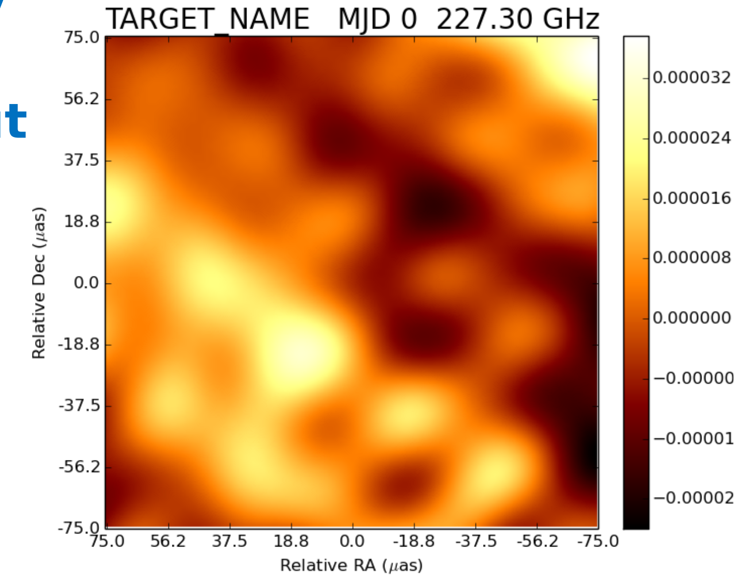
```
obs.plotall('uvdist','phase')
```



Dirty Image

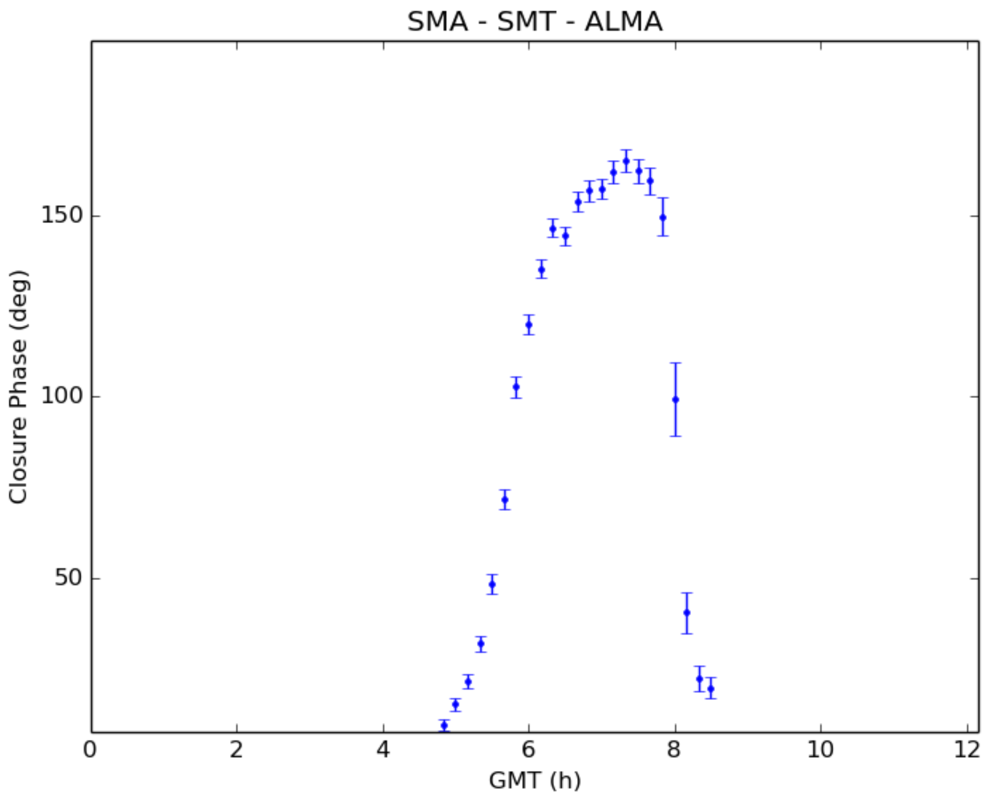
```
npix = 128
fov = 150*vb.RADPERUAS
dim = obs.dirtyimage(npix, fov)
dim.display()
```

No phase = No dirty image!
(Can’t CLEAN without Self-Cal)



Closure Phase is preserved

```
obs.plot_cphase('SMA', 'SMT', 'ALMA')
```



Array resolution and prior image

Array Resolution

```
beamparams = obs.fit_beam()
res = obs.res()
```

Prior Parameters

```
npix = 128
fov = 150*vb.RADPERUAS
zbl = 1.0
prior_fwhm = 100*vb.RADPERUAS
gaussparams = (prior_fwhm, prior_fwhm, 0.0)
```

The zero baseline flux

FWHM of our circular Gaussian Prior

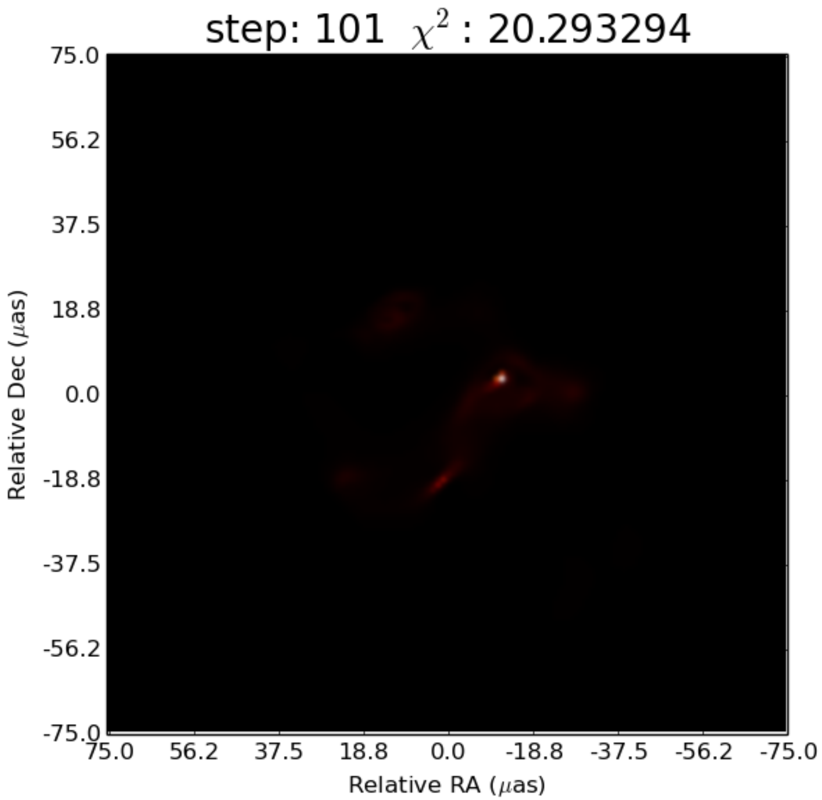
Create the Gaussian Prior

```
emptyprior = eh.image.make_square(obs, npix, fov)
gaussprior = emptyprior.add_gauss(zbl, gaussparams)
```

Image with amplitude and closure phase

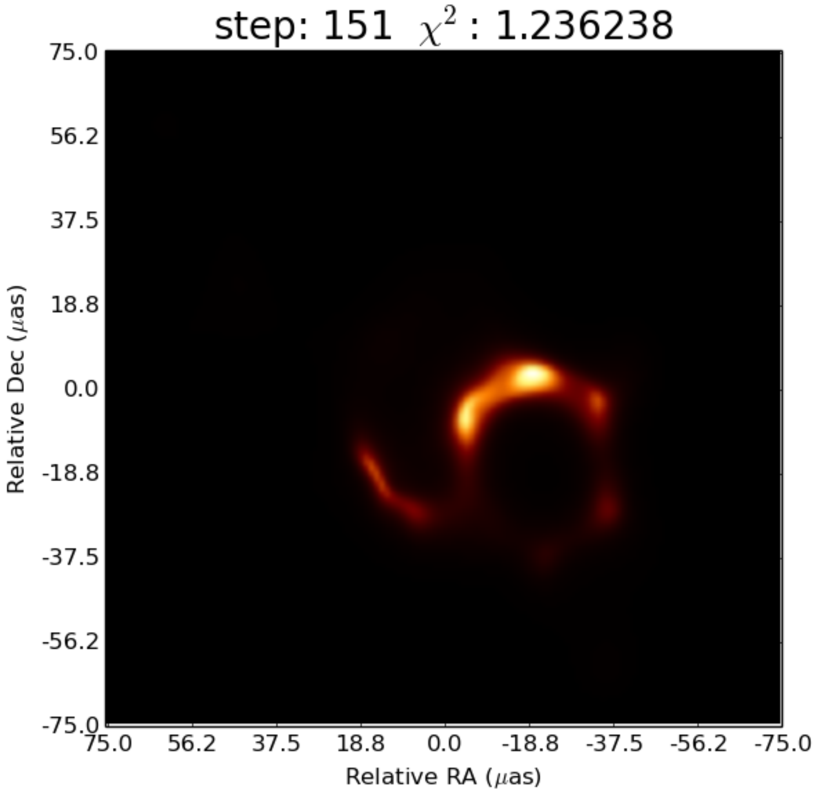
```
out = eh.imager_func(obs, gaussprior, gaussprior, zbl,d1="amp",d2="cphase", alpha_d1=100, alpha_d2=50, s1="gs",  maxit=100)
```

From experience, closure phase fits faster so we decrease its weight



Blur and re-image

```
outblur = out.blur_gauss(beamparams, 0.5)
out=outblur
out = eh.imager_func.maxen_amp_cphase(obs, out, out, zbl, d1=”amp”,
d2=”cphase”, alpha_d1=50, alpha_d2=25, maxit=150, s1="tv”)
```



Final images

Final “restored” image

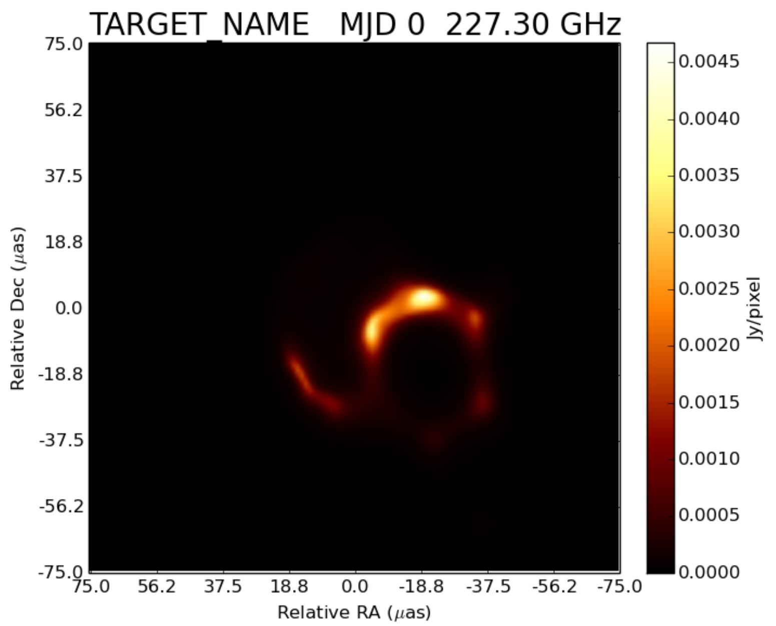
```
outblur = out.blur_gauss(beamparams, 0.5)
```

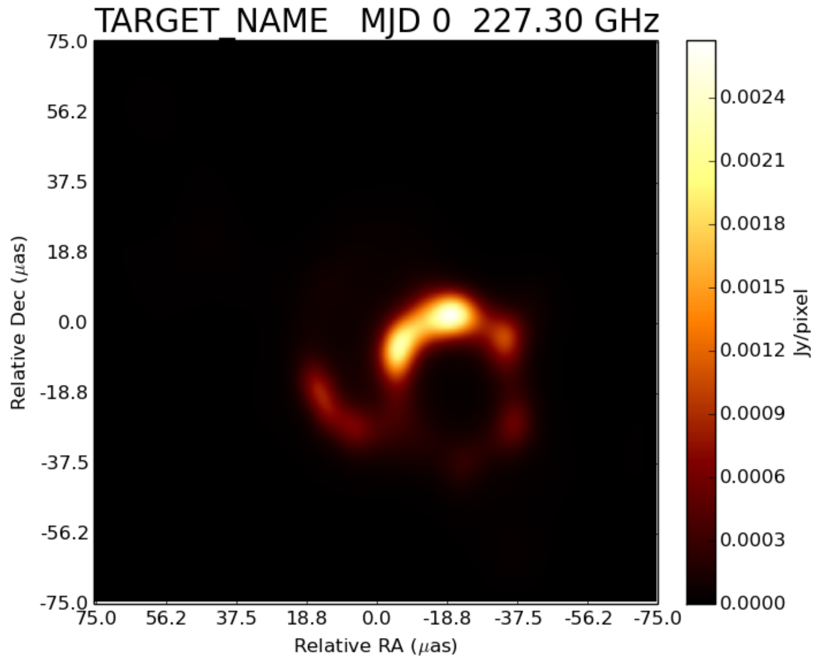
Display results

```
out.display()
outblur.display()
```

Save to FITS

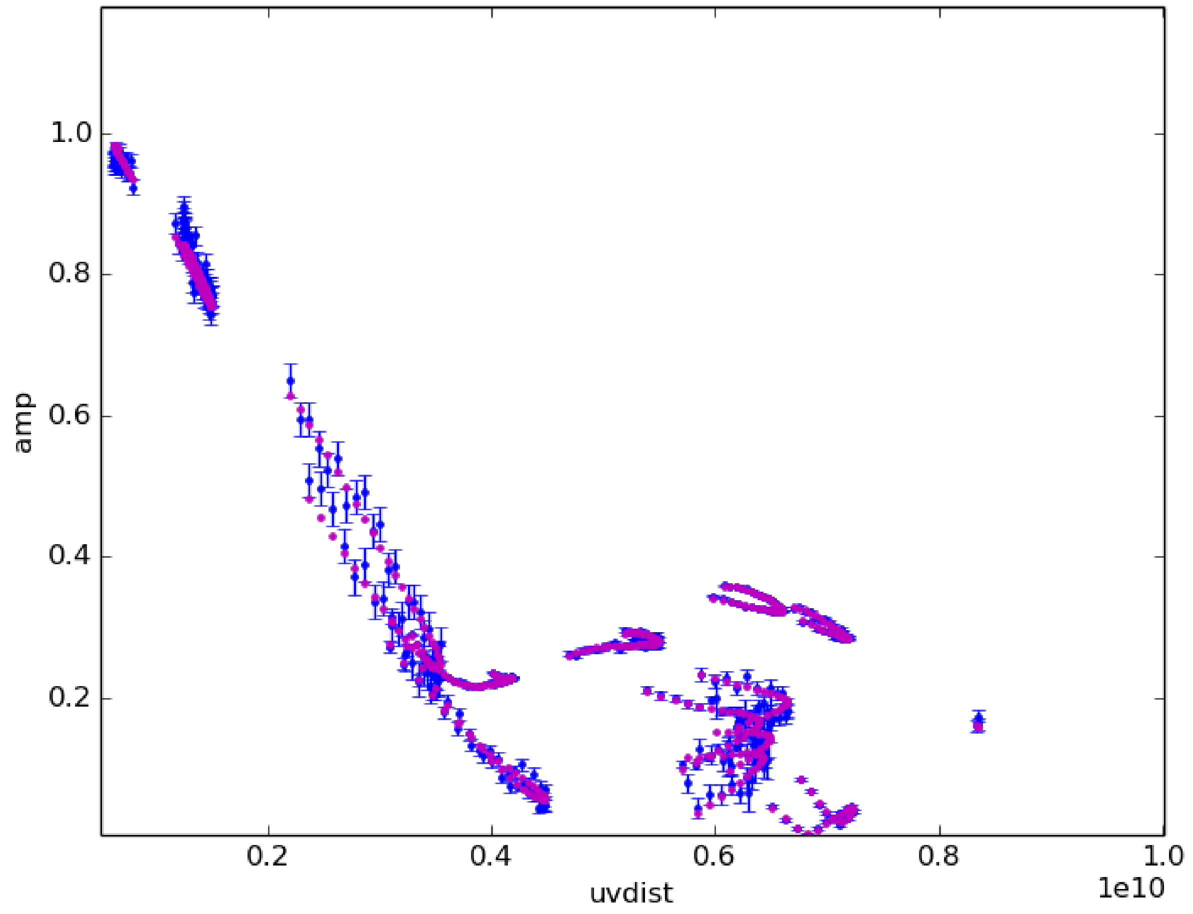
```
imageout.save_fits('./M87im.fits')
outblur.save_fits('./M87im_blur.fits')
```





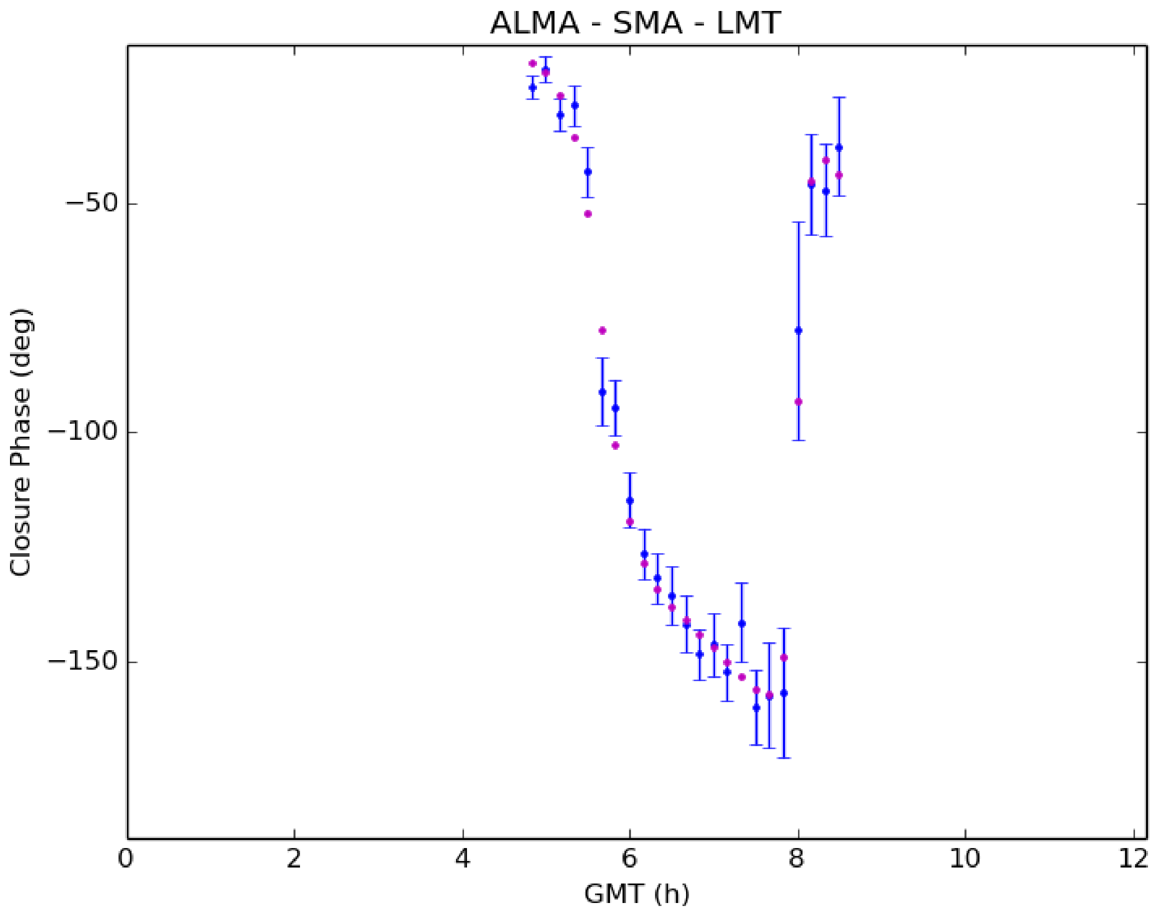
Look at fit to data - Amplitudes

```
eh.plotting.comp_plots.plotall_obs_im_compare(obs, out, "uvdist", "amp")
```



Look at fit to data – Closure Phase

```
eh.plottling.comp_plots.plot_cphase_obs_im_compare(obs, out, "ALMA","SMA","LMT")
```



Step 7: Participate in the Imaging Challenge!

New challenge out now!

Deadline: December 9^{th} , 2016

vlbiimaging.csail.mit.edu/imagingchallenge

- Blind data that you download (in uvfits, oifits, and text files)
- Sample data with truth images to help verify your algorithms are working
- Code to help you get started

Testing Data and Submission Instructions

1. **Download the test data from HERE.**
2. **Use your algorithm to generate an image for each of the data files.** For each < filename >.txt file, submit a FITS image with the name < filename >.fits and the FOV specified in the README File. Further instructions can be found in the README file.
3. **Submit your reconstructed images.** Compress all of your reconstructed FITS images into a ZIP file. Submit this ZIP file with the required additional information.

Method Name: Email: Images: Choose File No file chosen

Additional Information (such as website/code links):

SUBMIT

Data Parameters and Noise Properties

Challenge Number	Source Location	Telescopes	Total Flux (Janskys)	Noise Property
1	3C279	SMA, JCMT, SMT, LMT, ALMA, APEX, PV, PDB, SPT	3	Thermal & Atmospheric
2	M87	SMA, JCMT, SMT, LMT, ALMA, APEX, PV, PDB, SPT	2	Thermal & Atmospheric
3	Sgr A*	SMA, JCMT, SMT, LMT, ALMA, APEX, PV, PDB, SPT	2	Thermal & Atmospheric & Systematic

Advanced Topics

Other types of imaging implemented in `eht-imaging`

- CHIRP
- Closure-only (Closure Amplitudes + Closure Phases)
- Stochastic Optics (Scattering Mitigation)
- Polarimetric (using phase-robust ratios)
- Dynamic Imaging (Movie Reconstruction)
- Add your own!