CLEAN: Iterative Deconvolution

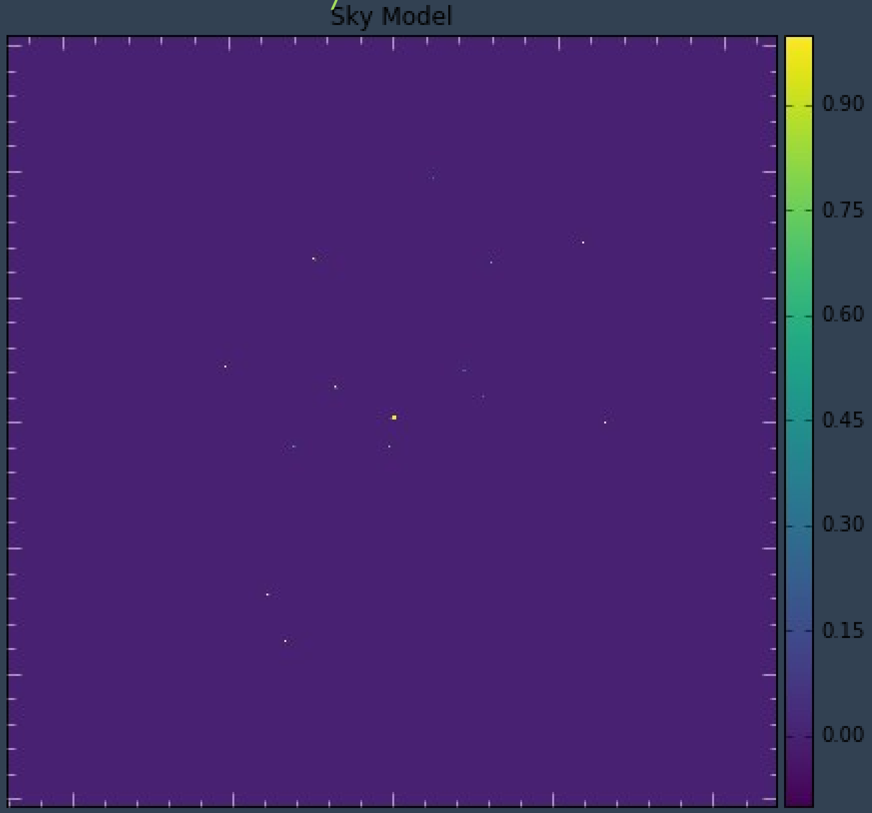
Fundamentals of Radio Interferometry (Chapter 6)

Griffin Foster
SKA SA/Rhodes University

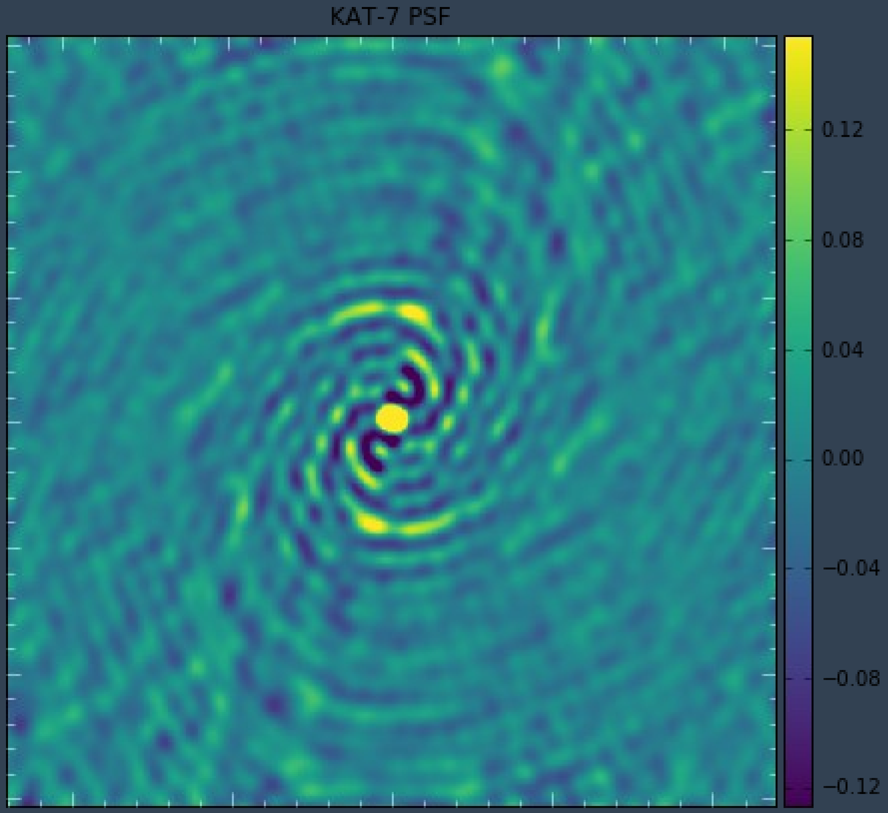
NASSP 2016

Sky Model Convolved with Array PSF

Sky Model

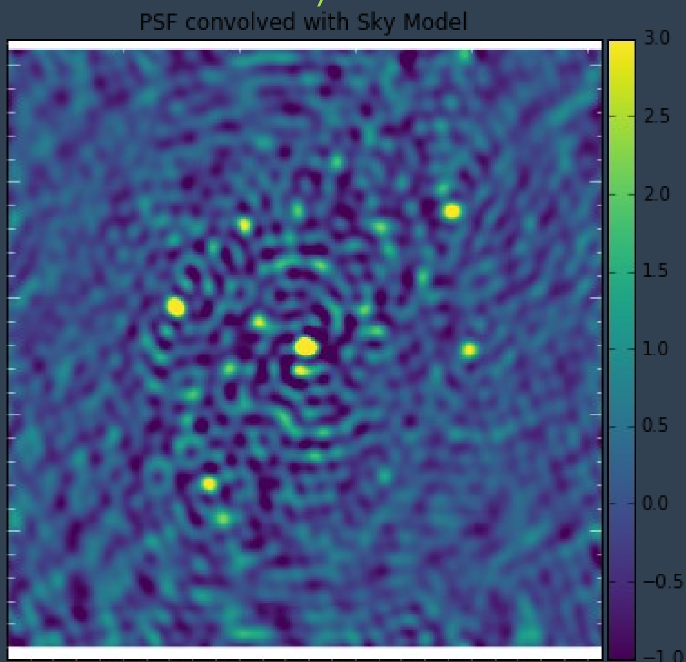


KAT-7 PSF

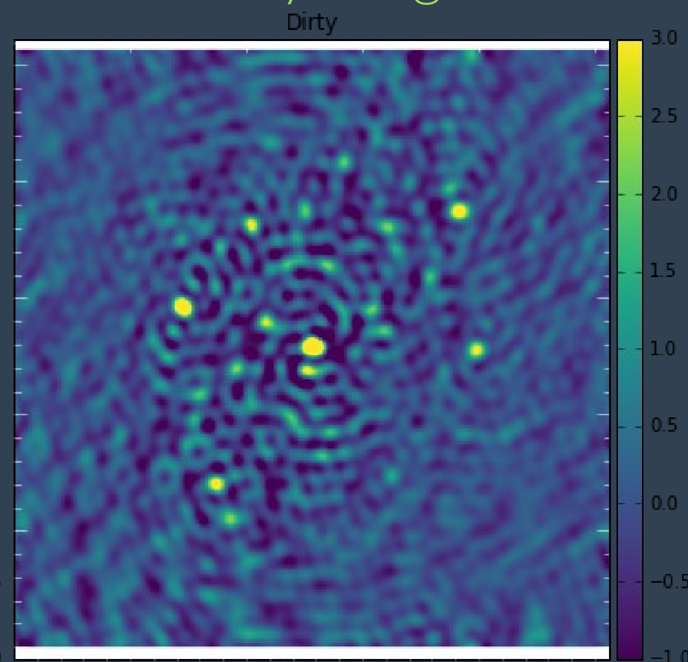


Sky Model Convolved with Array PSF

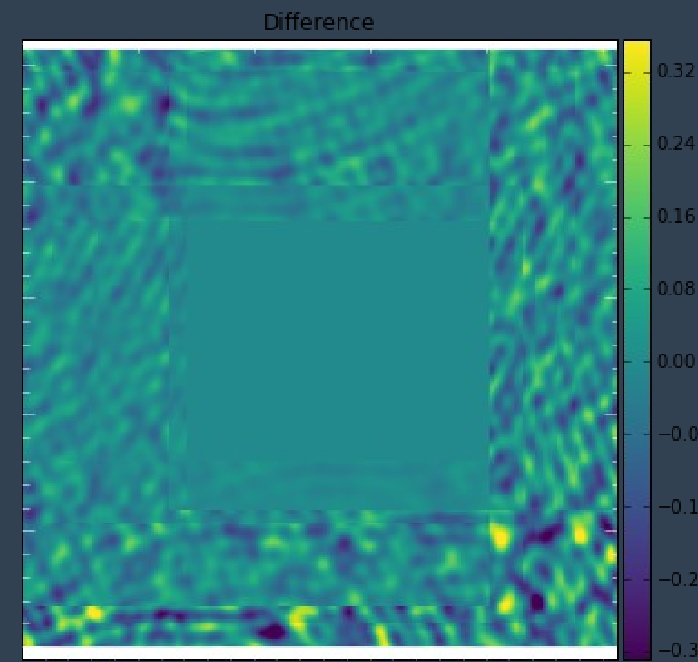
PSF Convolved with Sky Model



Dirty Image



Difference



How do we separate out the signal (the sky model) from the noise?

Naïve Deconvolution: Inverse Filtering

Given a function ***h*** with is the convolution of two other functions ***g*** and ***f***:

$$h = f \circ g$$

Given, ***h*** and one of the other functions, say ***g*** then ***f*** can be *deconvolved* by using the convolution theorem:

$$f = \mathcal{F}^{-1}\{\mathcal{F}\{f\}\} = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{h\}}{\mathcal{F}\{g\}}\right\}$$

This is called *inverse filtering*

Naïve Deconvolution: Inverse Filtering

Our deconvolution problem is to recover the true sky image from the PSF and the dirty image

$$I^D = \mathrm{PSF} \circ I_{true}$$

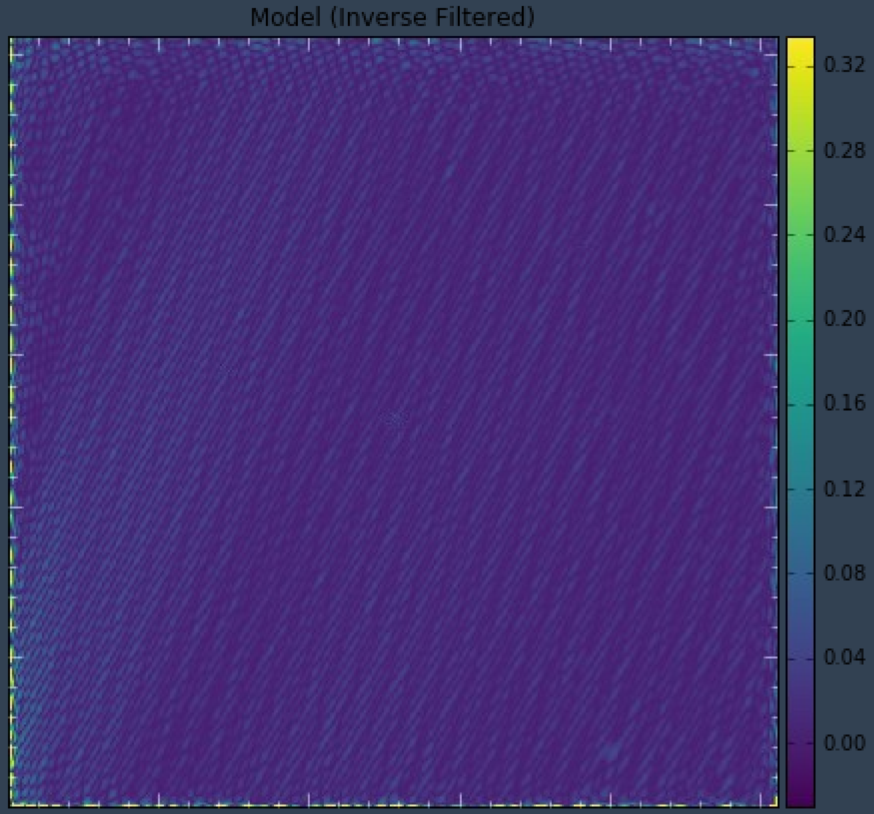
We can try to recover the true sky image with this method:

$$I_{true} = \mathcal{F}^{-1}\left\{\frac{\mathcal{F}\{I^D\}}{\mathcal{F}\{\mathrm{PSF}\}}\right\}$$

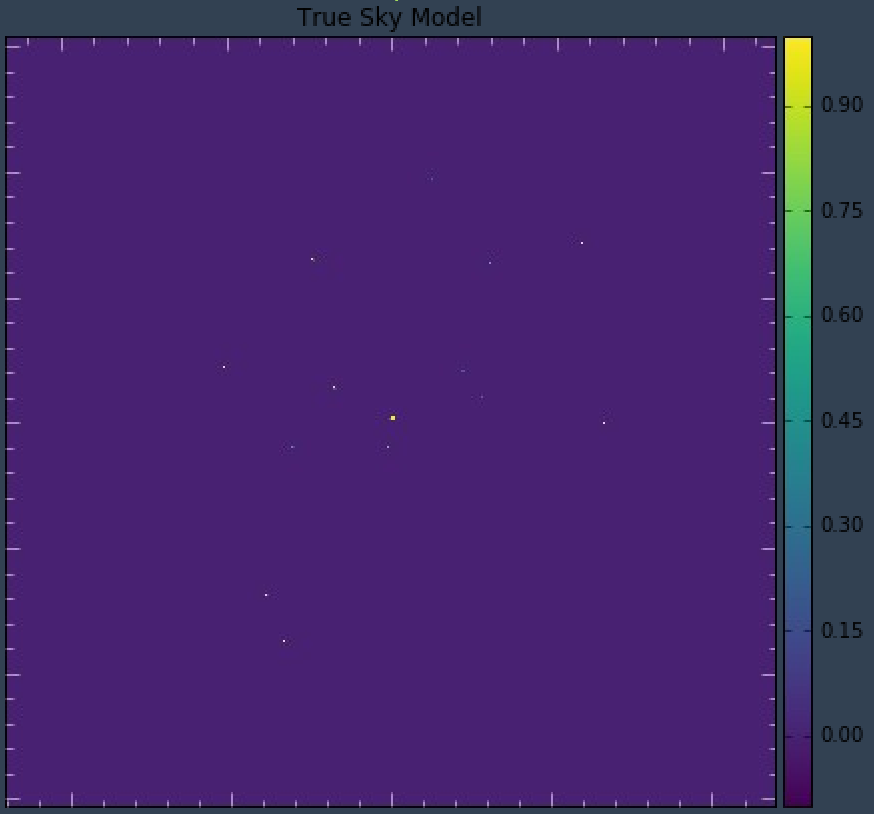
Unfortunately...

Naïve Deconvolution: Inverse Filtering

Sky Model Recovered Using Inverse Filtering



True Sky Model



Naïve Deconvolution: Inverse Filtering

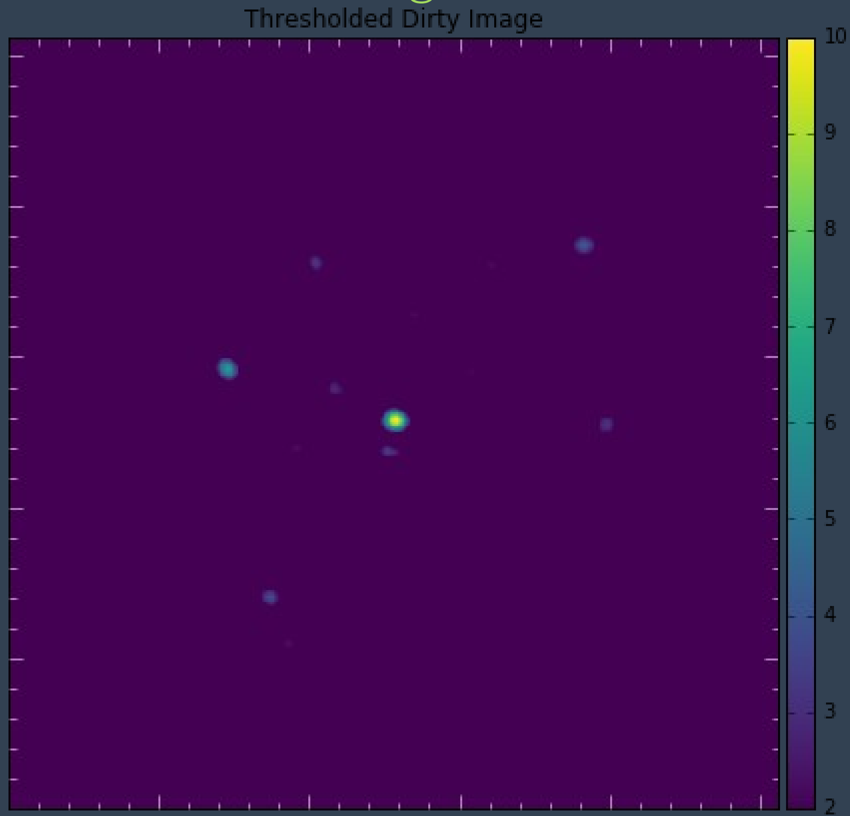
Inverse filtering only works when there is NO noise in the measurement.
Unfortunately, there is noise in any real world measurement.

$$I^D = \mathrm{PSF} \circ I_{true} + \epsilon$$

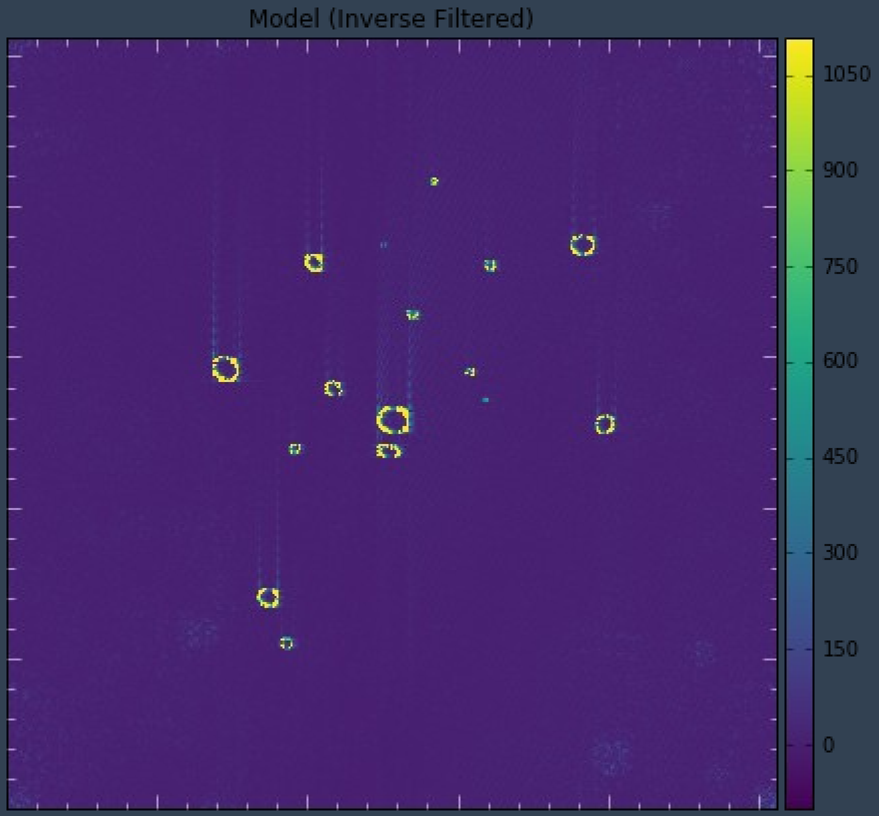
Sky noise, instrumentational noise, computation error...

Naïve Deconvolution: Thresholding

Thresholded Dirty Image



Sky Model Recovered Using Thresholding



Sky Model using Point Source Components

Fourier transform of a Dirac delta-function, by the Fourier shift theorem, is a simple complex phase function and the constant flux

$$F\{C(\nu) \cdot \delta(l - l_0, m - m_0)\}(u, v) =$$

$$C(\nu) \cdot \iint_{-\infty}^{\infty} \delta(l - l_0, m - m_0)\, e^{-2\pi i(ul+vm)}\, dl\, dm$$

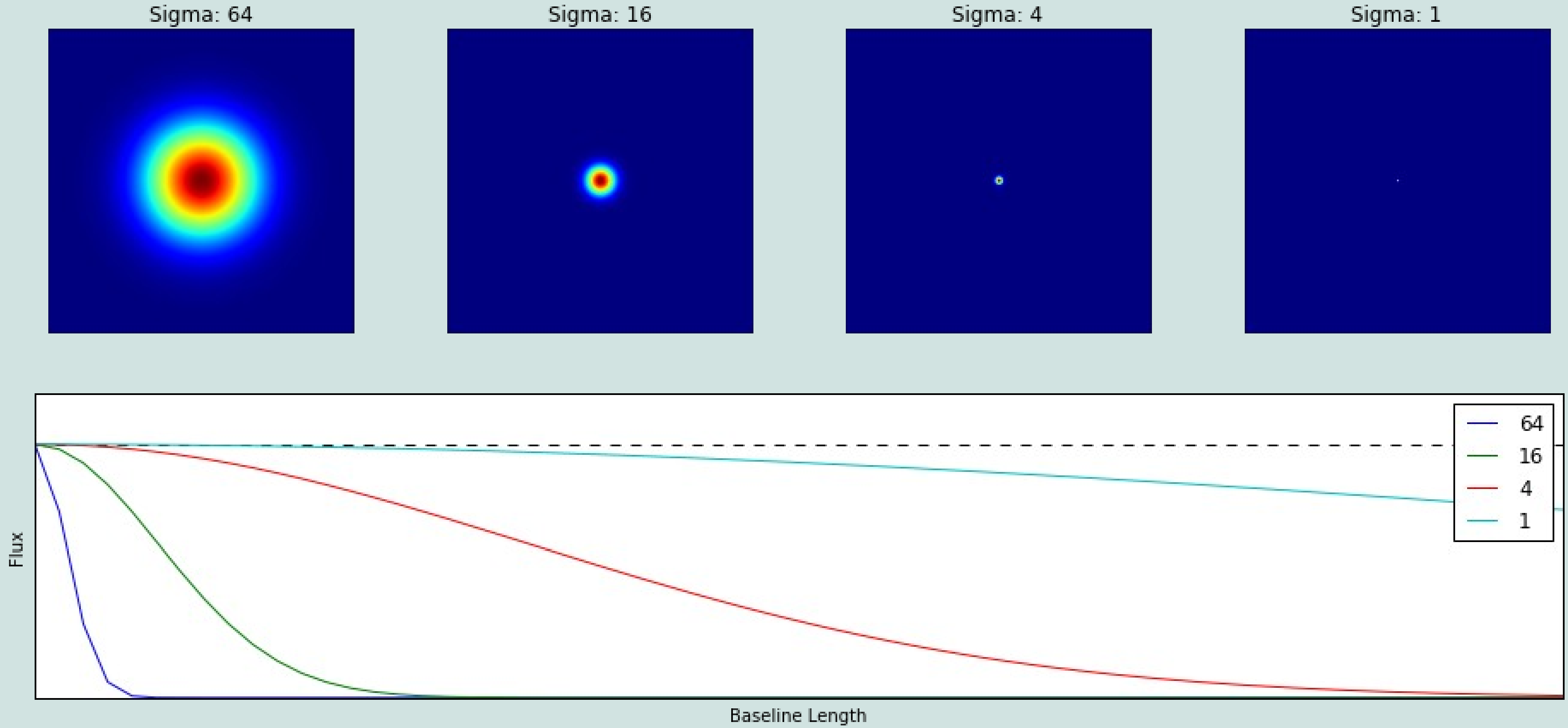
$$= C(\nu) \cdot e^{-2\pi i(ul_0+vm_0)}$$

Sky Model using Point Source Components

Source ID	RA (Hours)	Dec (degrees)	Flux (Jy)	Spectral Index
1	00:02:18.81	-29:47:17.82	3.55	-0.73
2	00:01:01.84	-30:06:27.53	2.29	-0.52
3	00:03:05.54	-30:00:22.57	1.01	-0.60
...	...	...	...	...
N	00:02:17.01	-30:01:34.57	0.001	-0.71

To compute model visibilities for deconvolution and self-calibration (in a few weeks) we want to use functions which have an analytic Fourier form (delta functions, Gaussian) to reduce computation time.

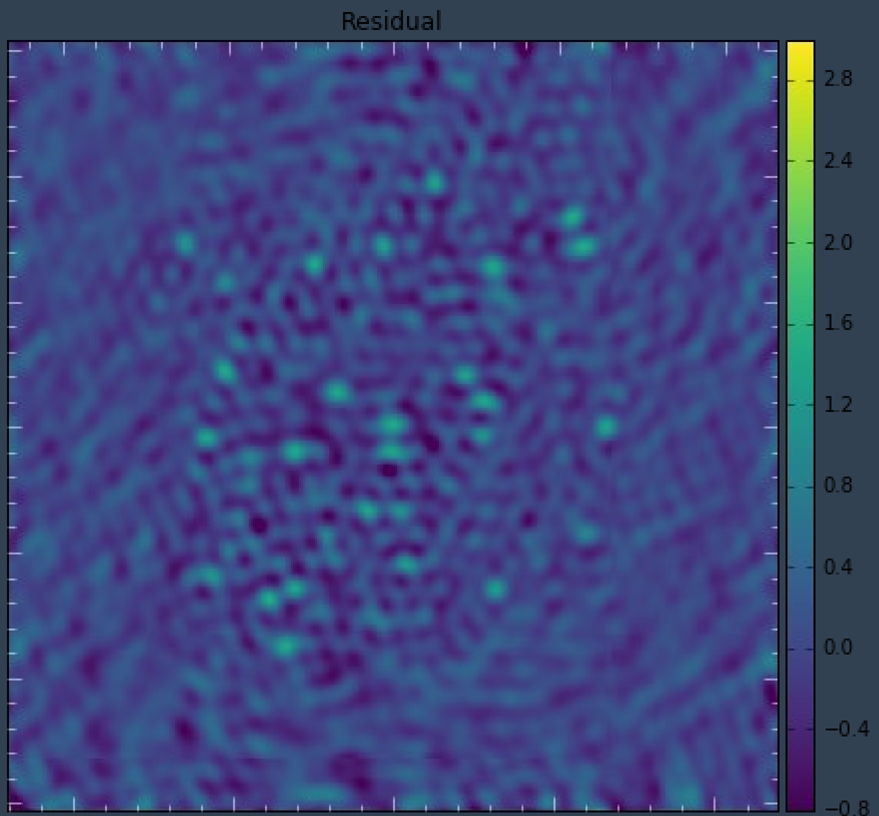
Resolved Sources



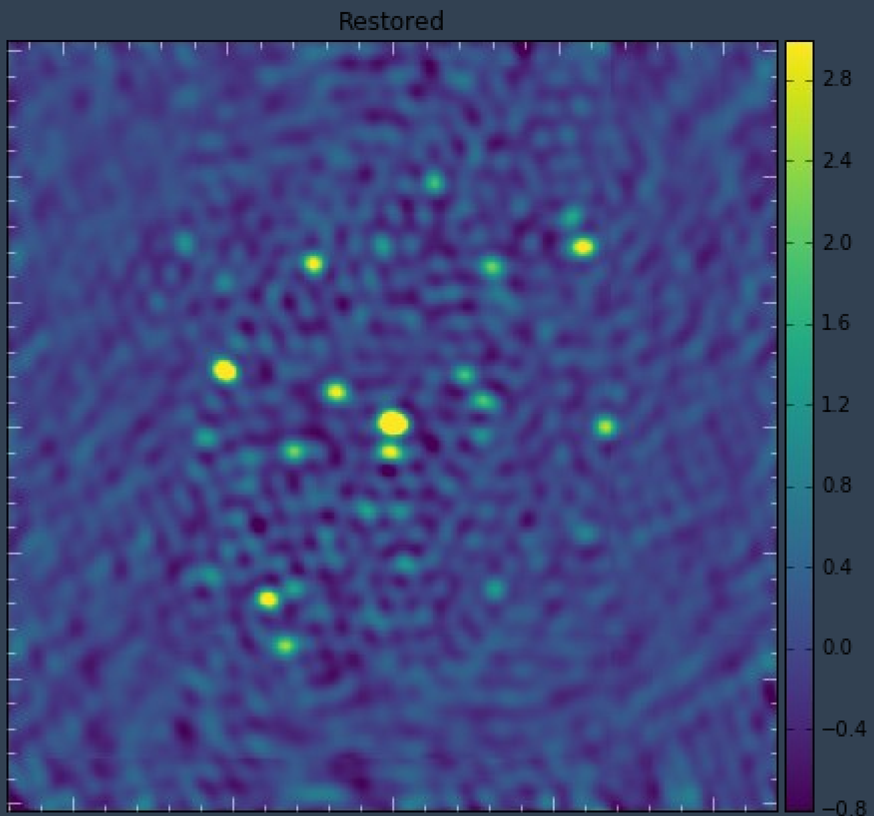
A point source will have the same flux at any baseline length. Any resolved source will have a baseline length dependent flux response.

Results of Deconvolution

Residual Image

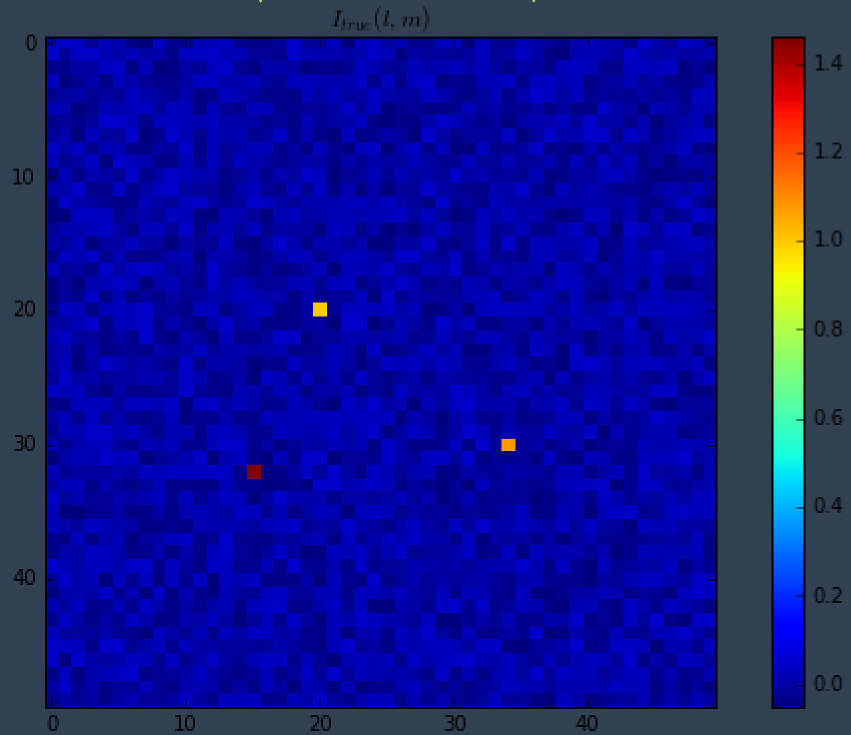


Restored Image

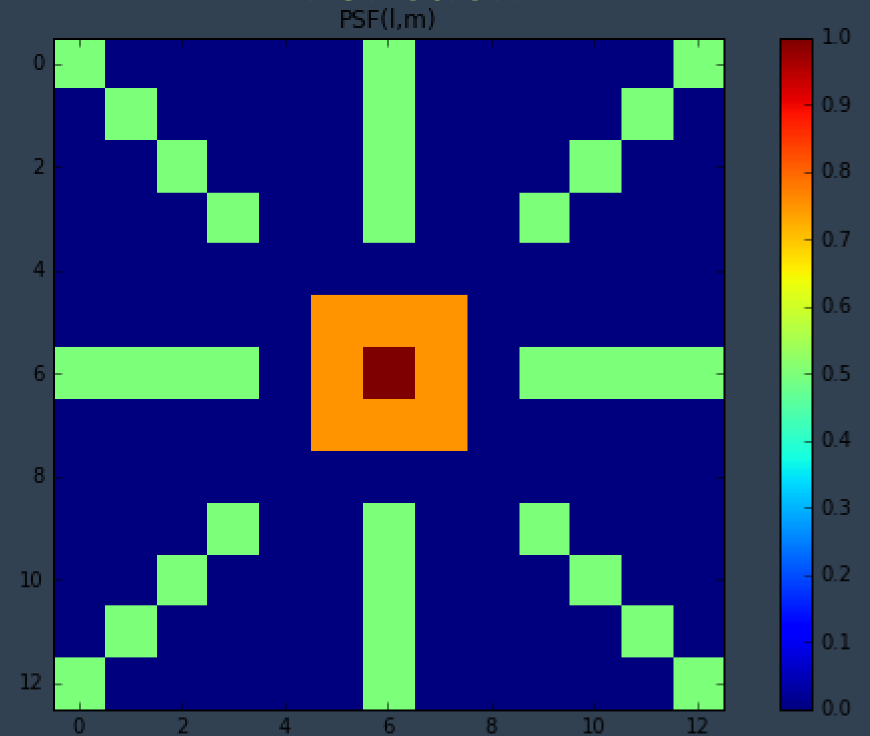


CLEAN: A Simple Example

True Sky Model (with Noise)

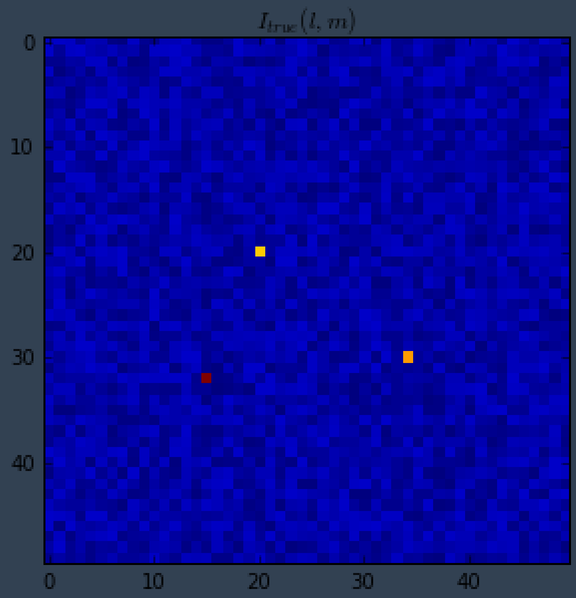


Point Spread Function

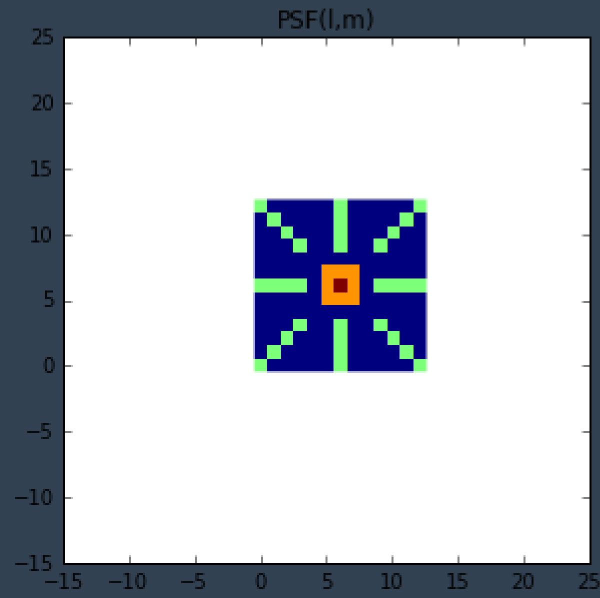


CLEAN: A Simple Example

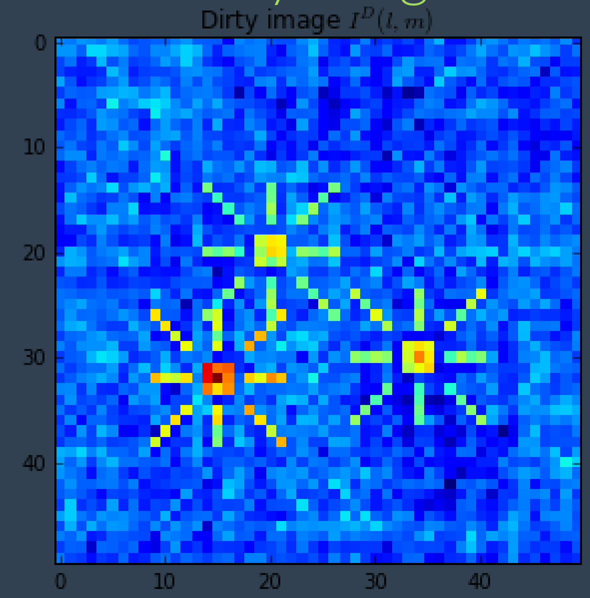
True Sky Model (with Noise)



Point Spread Function



Dirty Image



CLEAN: A Simple Example

Högbom's Algorithm (Image-domain CLEAN):

1. Make a copy the dirty image $I^D(l,m)$ called the *residual image* $I^R(l,m)$.

2. Find the maximum pixel value and position of the maximum in the residual image $I^R(l,m)$.

3. Subtract the PSF multiplied by the peak pixel value f_{max} and a gain factor **g** from the residual image $I^R(l,m)$ at the position of the peak.

4. Record the position and magnitude of the point source subtracted in a model, i.e. $g\ f_{max}$.

5. Go to (Step 2.), unless all remaining pixel values are below some user-specified threshold or the number of iterations have reached some user-specified limit.

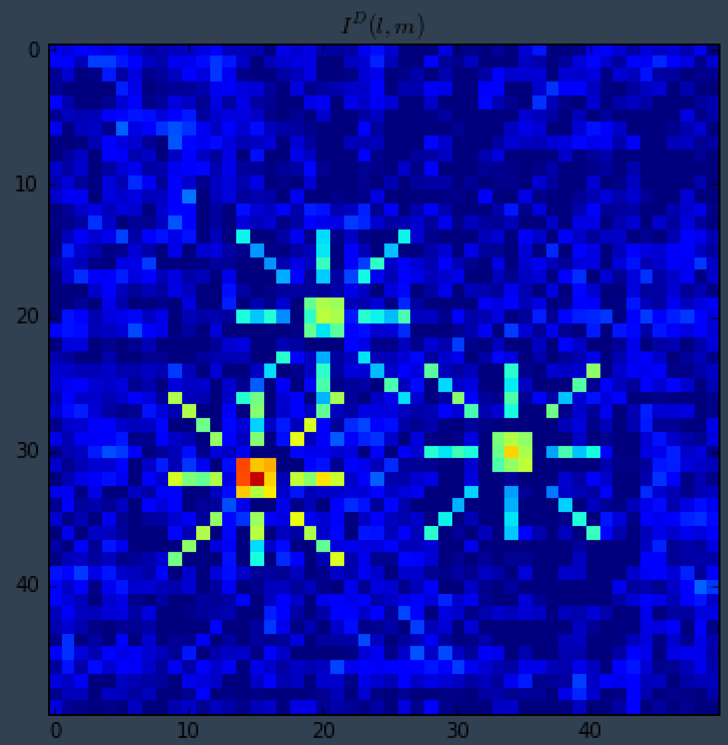
6. Convolve the accumulated point source sky model with a *restoring beam*, termed the CLEAN beam (usually a 2-D Gaussian fit to the main lobe of the PSF)

7. Add the remainder of the residual image $I^R(l,m)$ to the CLEAN image formed in (6.) to form the final *restored image*.

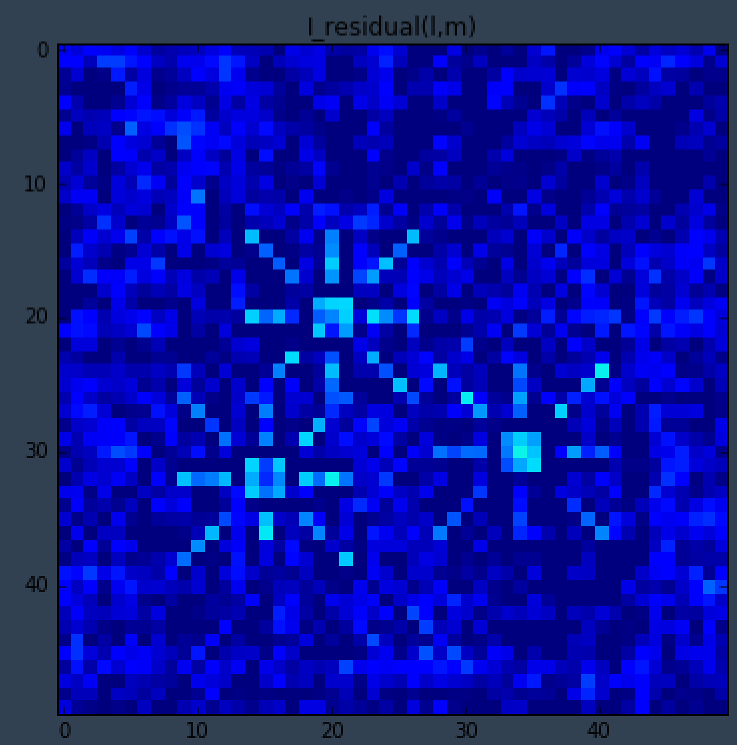
Input: Dirty image, PSF
Parameters: gain, iteration limit **OR** flux threshold
Output: Sky model, residual image, restored image

CLEAN: A Simple Example

Dirty Image

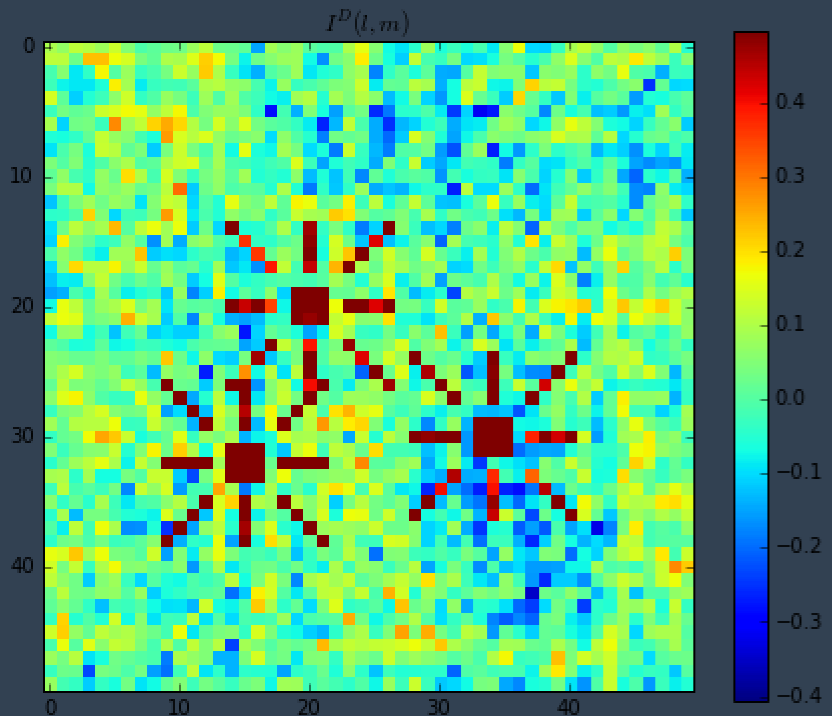


Residual Image

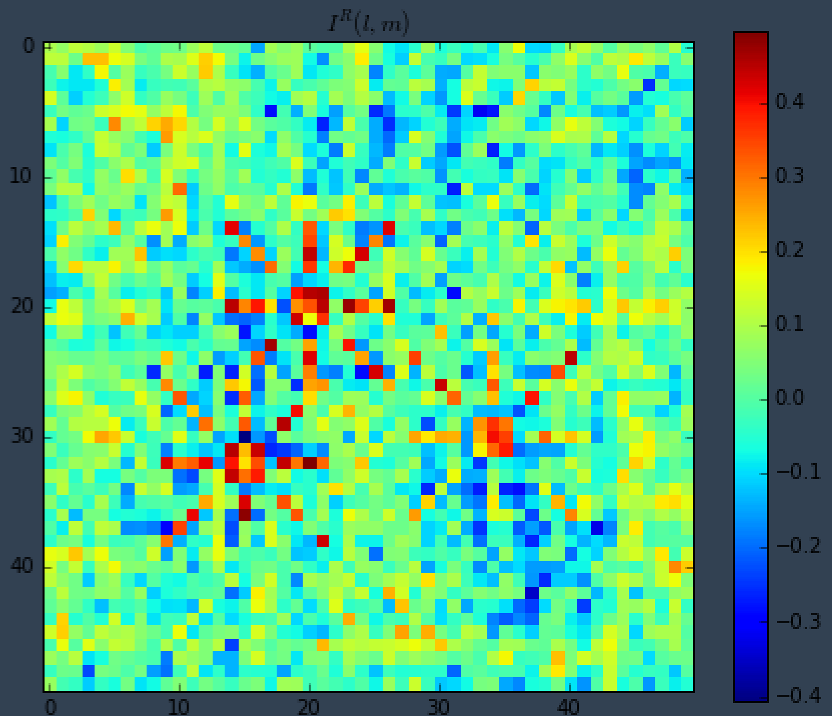


CLEAN: A Simple Example

Dirty Image



Residual Image



CLEAN: A Simple Example

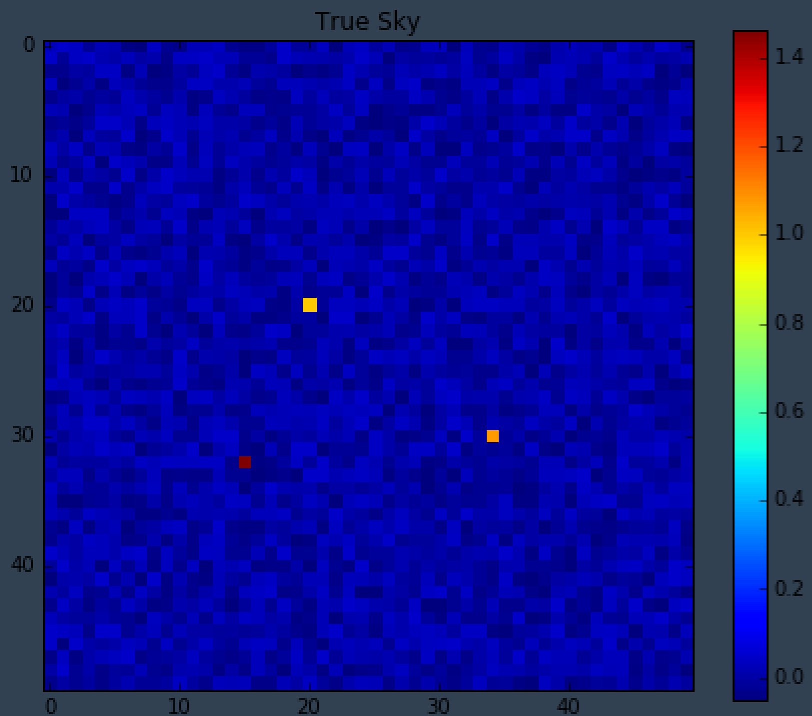
Clean components:

x	*y*	*flux*
32	15	0.273542117836
32	15	0.218833694269
30	34	0.197043506304
32	15	0.175066955415
20	20	0.164478127268
30	34	0.157634805043
31	14	0.141743159144
20	21	0.133470733705
30	34	0.126107844035
32	20	0.124271249713
31	14	0.113394527315
29	18	0.113236796988
19	20	0.11300001035
31	16	0.109407177869
38	21	0.109218346103
21	19	0.109080307468
25	25	0.106739789818
32	9	0.106513135995
30	34	0.100886275228

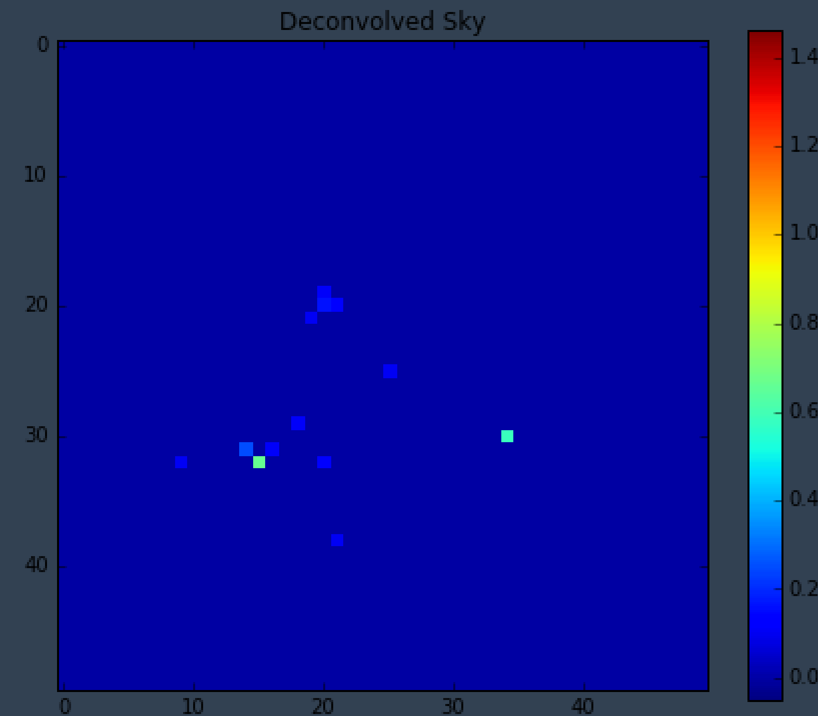
Same position, different amount of flux. The final sky model is the sum of the different components at the same position

CLEAN: A Simple Example

True Sky Model

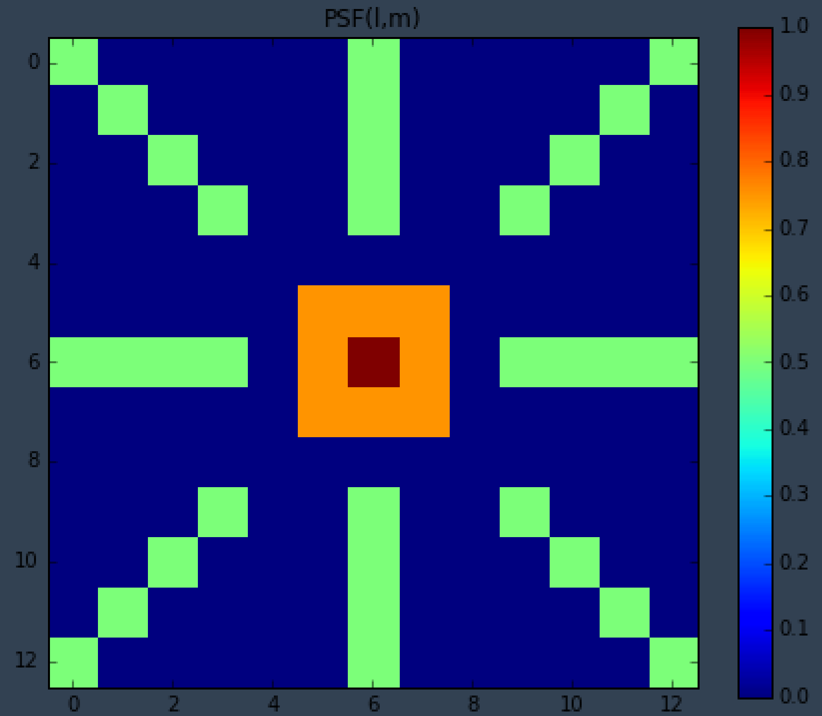


Deconvolved Sky Model

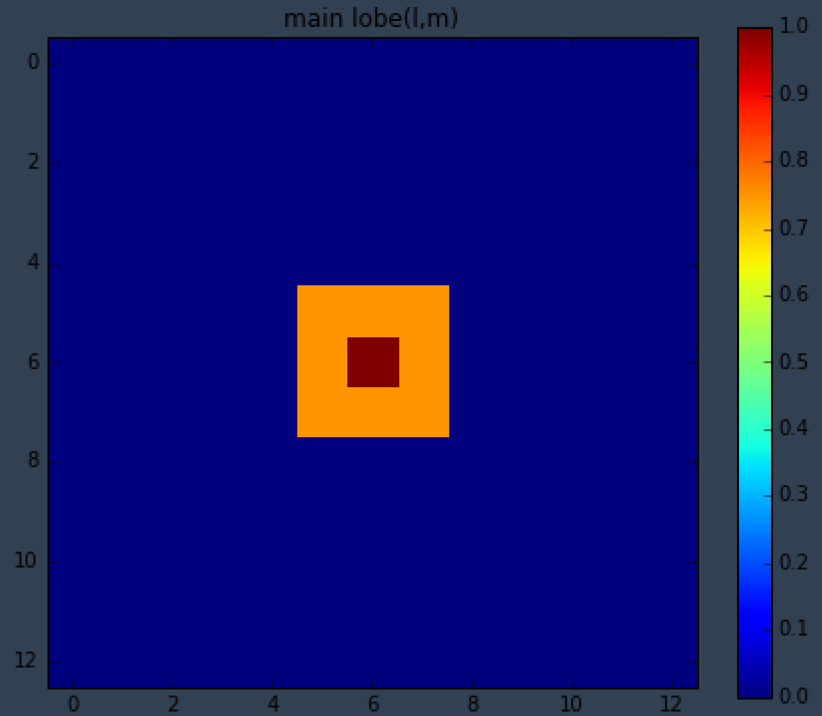


CLEAN: A Simple Example

Point Spread Function

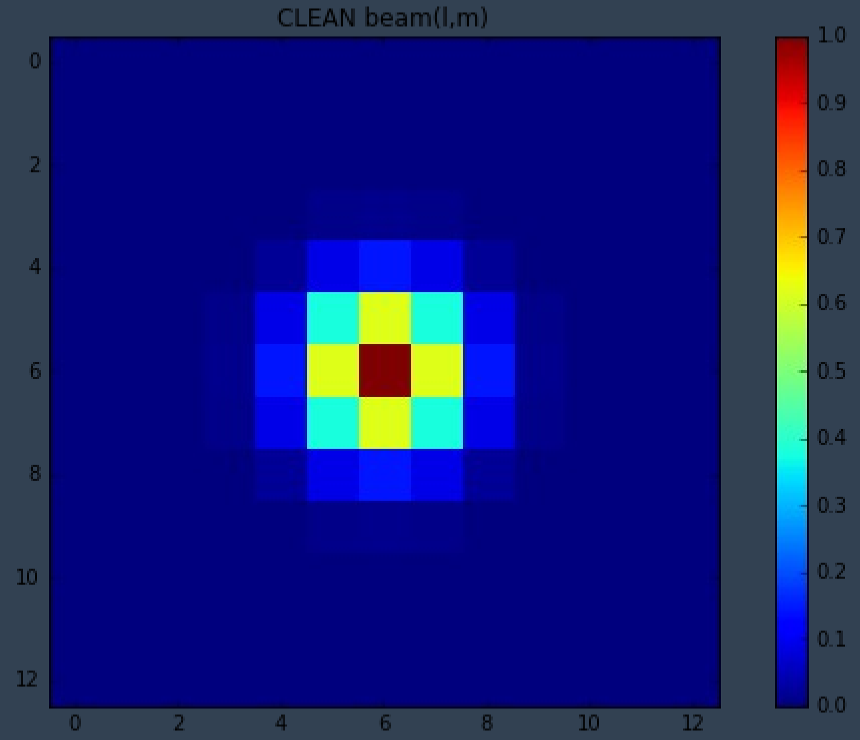


PSF Main Lobe

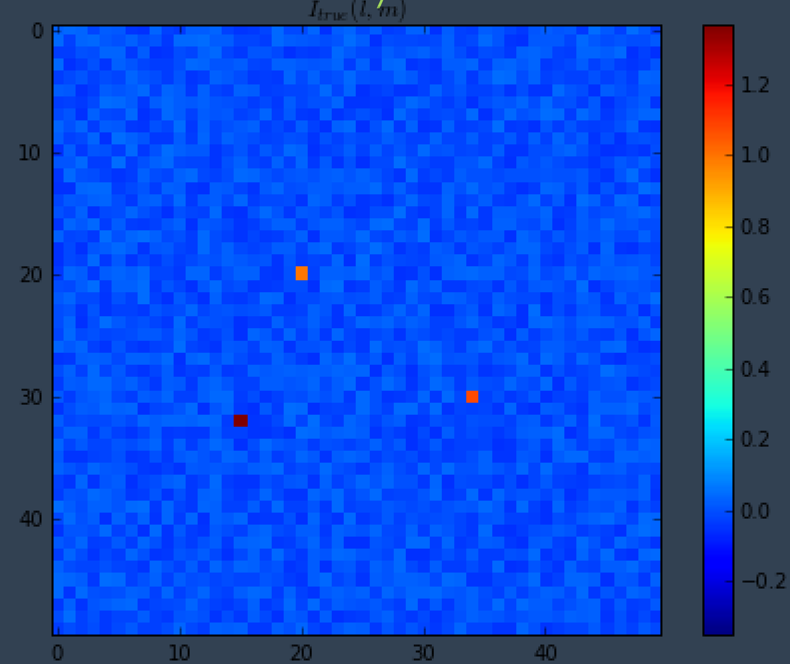


CLEAN: A Simple Example

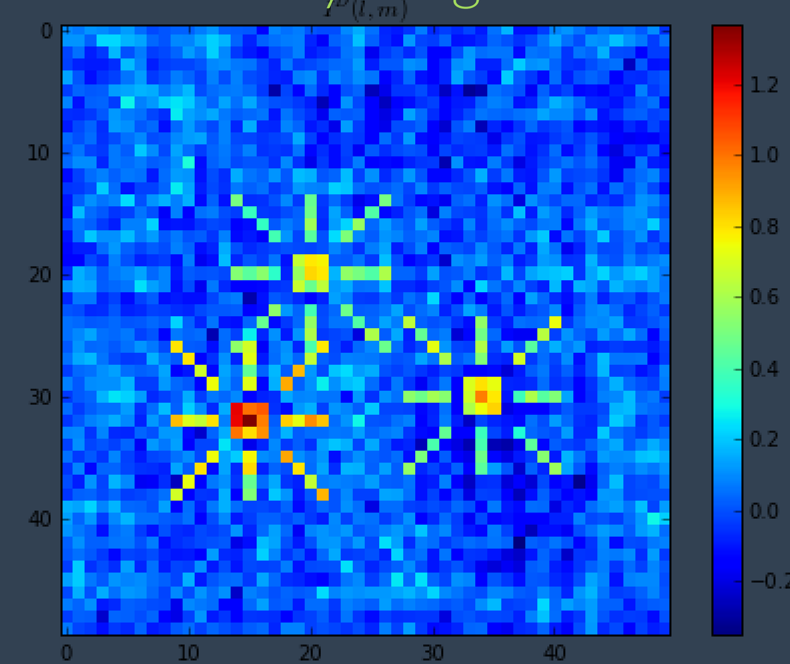
Restoring Beam/
Restoring PSF



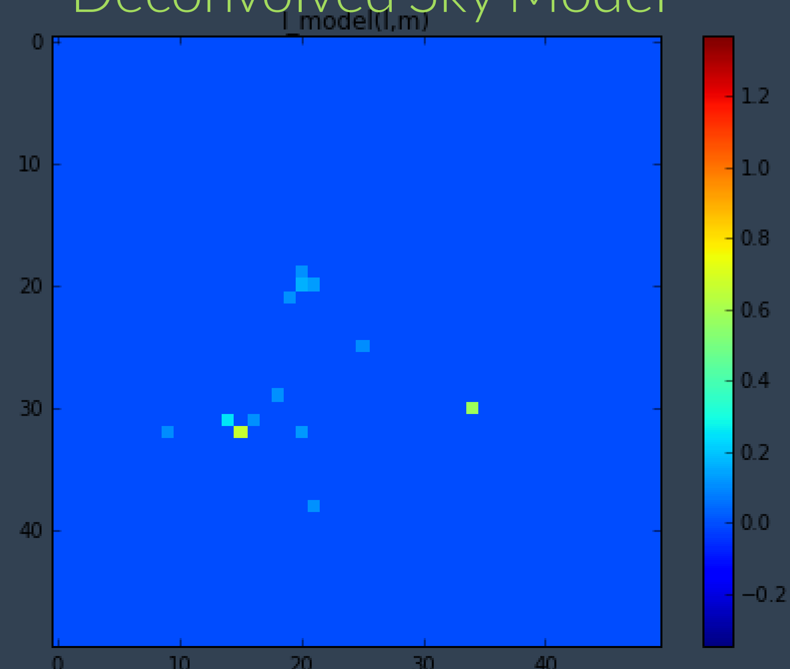
True Sky Model



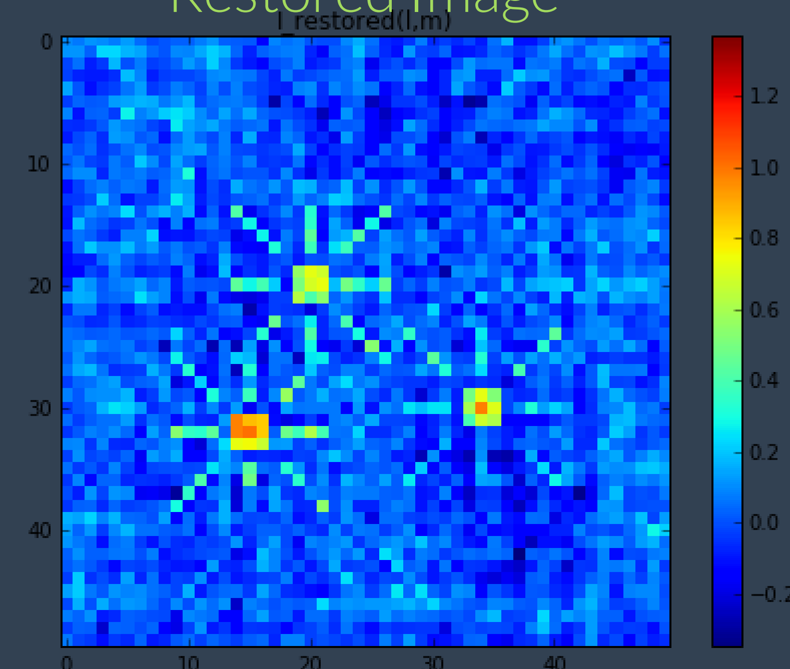
Dirty Image



Deconvolved Sky Model

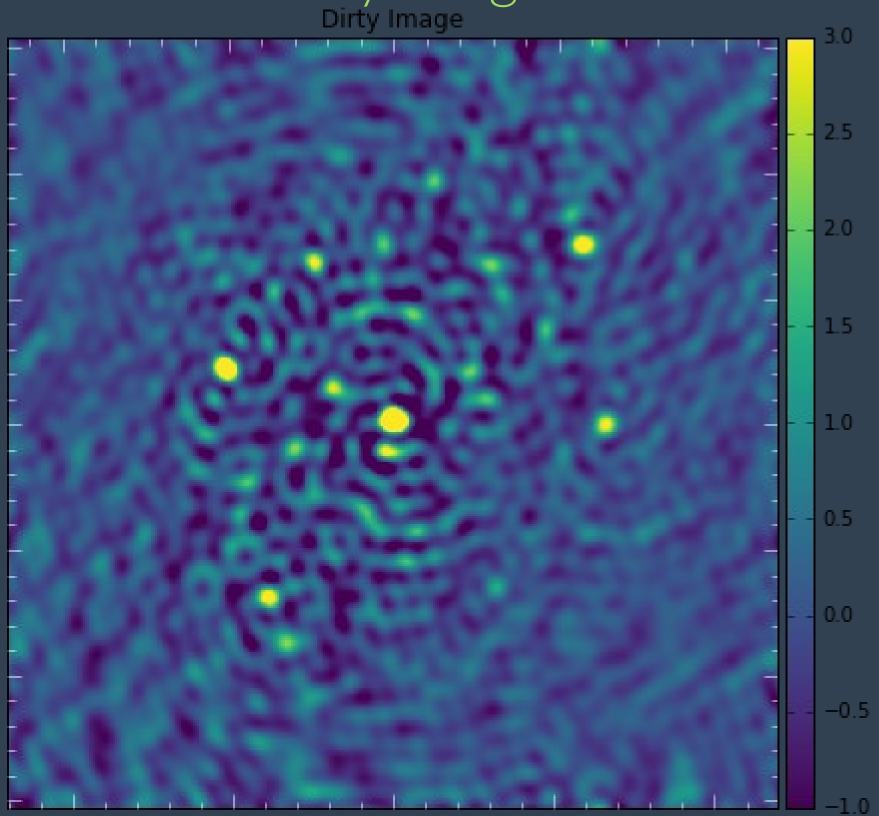


Restored Image

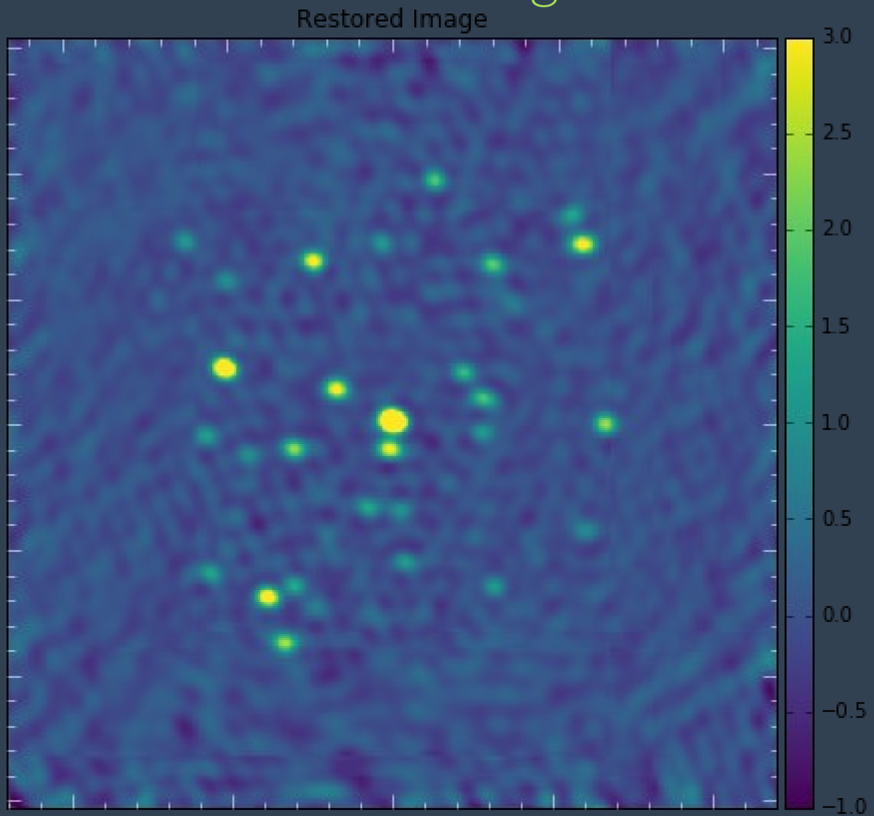


CLEAN: KAT-7 Example

Dirty Image



Restored Image



CLEAN: Högbom's Method (Image-domain)

input: $I^D(l,m),\ \mathrm{PSF}(l,m),\ \gamma,\ f_{\mathrm{thresh}},\ N$

initialize: $S^{\mathrm{model}} \leftarrow \{\}, I^{\mathrm{res}} \leftarrow I^D, i \leftarrow 0$

while any$(I^{\mathrm{res}} > f_{\mathrm{thresh}})$ or $i \leq N$ **do:**

$$l_{\max}, m_{\max} \leftarrow \underset{l,m}{\mathrm{argmax}}\, I^{\mathrm{res}}(l,m)$$

$$f_{\max} \leftarrow I^D(l_{\max}, m_{\max})$$

$$I^{\mathrm{res}} \leftarrow I^{\mathrm{res}} - \gamma \cdot f_{\max} \cdot \mathrm{PSF}(l + l_{\max}, m + m_{\max})$$

$$S^{\mathrm{model}} \leftarrow S^{\mathrm{model}} + \{l_{\max}, m_{\max} : \gamma \cdot f_{\max}\}$$

$$i \leftarrow i + 1$$

ouput: $S^{\mathrm{model}}, I^{\mathrm{res}}$

CLEAN: Input Parameters

$$\textbf{input:}\ \ I^D(l,m),\ \ \text{PSF}(l,m),\ \ \gamma,\ \ f_{\text{thresh}},\ \ N$$

Point Spread Function Image (real, positive valued N x N array)

Dirty Image (real, positive valued N x N array)

γ : gain factor, between 0 and 1, determines the rate of deconvolution. Typically set around 0.1

f_{thresh} : flux threshold stopping criteria, once the maximum flux is at this level then halt.

N : maximum number of iterations to preform.

CLEAN: Initialization and Output

initialize: $S^{\text{model}} \leftarrow \{\}, I^{\text{res}} \leftarrow I^{D}, i \leftarrow 0$

S^{model} : empty delta-function sky model

I^{res} : initialize the initial residual image to be the dirty image

i : set iteration counter to zero

ouput: $S^{\text{model}}, I^{\text{res}}$

S^{model} : final sky model of delta-function components

I^{res} : residual noise not deconvolved

I^{restored} : (optional) sky model restored image with the ideal PSF

CLEAN: Iterative Loop

$$I^{\text{res}} \leftarrow I^{\text{res}} - \gamma \cdot f_{\text{max}} \cdot \text{PSF}(l + l_{\text{max}}, m + m_{\text{max}})$$

Subtract the PSF image from the position of the peak flux, attenuated by the gain factor to update the residual image.

$$S^{\text{model}} \leftarrow S^{\text{model}} + \{l_{\text{max}}, m_{\text{max}} : \gamma \cdot f_{\text{max}}\}$$

Add the flux and position of the component subtracted from the residual image.

CLEAN: Clark's Method (Gridded Visibility-domain)

input: $I^D(l,m),\ \mathrm{PSF}(l,m),\ \gamma,\ f_{\mathrm{thresh}},\ N$

initialize: $S^{\mathrm{model}} \leftarrow \{\},\ I^{\mathrm{res}} \leftarrow I^D,\ i \leftarrow 0,\ (\mathrm{PSF}_{\mathrm{sub}}(l,m),\ R_{\mathrm{PSF}}) \leftarrow g(\mathrm{PSF}(l,m))$

while any$(I^{\mathrm{res}} > f_{\mathrm{thresh}})$ or $i \leq N$ **do:** [Major Cycle]

$$l_{\max}, m_{\max} \leftarrow \underset{l,m}{\operatorname{argmax}}\, I^{\mathrm{res}}(l,m)$$

$$f_{\max} \leftarrow I^D(l_{\max}, m_{\max})$$

$$S^{\mathrm{model}}_{\mathrm{partial}} \leftarrow \mathrm{Hogbom}(I^{\mathrm{res}},\ \mathrm{PSF}_{\mathrm{sub}},\ \gamma,\ f_{\max} \cdot R_{\mathrm{PSF}}) \quad \text{[Minor Cycle]}$$

$$V^{\mathrm{model}}_{\mathrm{partial}} \leftarrow \mathcal{F}\{S^{\mathrm{model}}_{\mathrm{partial}}\}, V^S \leftarrow \mathcal{F}\{\mathrm{PSF}\}$$

$$I^{\mathrm{res}} \leftarrow I^{\mathrm{res}} - \mathcal{F}^{-1}\{V^S \cdot V^{\mathrm{model}}_{\mathrm{partial}}\}$$

$$S^{\mathrm{model}} \leftarrow S^{\mathrm{model}} + S^{\mathrm{model}}_{\mathrm{partial}}$$

$$i \leftarrow i + 1$$

ouput: $S^{\mathrm{model}}, I^{\mathrm{res}}$

CLEAN: Clark's Method (Gridded Visibility-domain)

input: $I^D(l,m),\ \mathrm{PSF}(l,m),\ \gamma,\ f_{\mathrm{thresh}},\ N$

initialize: $S^{\mathrm{model}} \leftarrow \{\},\ I^{\mathrm{res}} \leftarrow I^D,\ i \leftarrow 0,\ (\mathrm{PSF}_{\mathrm{sub}}(l,m),\ R_{\mathrm{PSF}}) \leftarrow g(\mathrm{PSF}(l,m))$

while any$(I^{\mathrm{res}} > f_{\mathrm{thresh}})$ or $i \leq N$ **do:** [Major Cycle]

$$l_{\max}, m_{\max} \leftarrow \underset{l,m}{\operatorname{argmax}}\, I^{\mathrm{res}}(l,m)$$

$$f_{\max} \leftarrow I^D(l_{\max}, m_{\max})$$

$$S^{\mathrm{model}}_{\mathrm{partial}} \leftarrow \mathrm{Hogbom}(I^{\mathrm{res}},\ \mathrm{PSF}_{\mathrm{sub}},\ \gamma,\ f_{\max} \cdot R_{\mathrm{PSF}})$$ [Minor Cycle]

$$V^{\mathrm{model}}_{\mathrm{partial}} \leftarrow \mathcal{F}\{S^{\mathrm{model}}_{\mathrm{partial}}\}, V^S \leftarrow \mathcal{F}\{\mathrm{PSF}\}$$

$$I^{\mathrm{res}} \leftarrow I^{\mathrm{res}} - \mathcal{F}^{-1}\{V^S \cdot V^{\mathrm{model}}_{\mathrm{partial}}\}$$

$$S^{\mathrm{model}} \leftarrow S^{\mathrm{model}} + S^{\mathrm{model}}_{\mathrm{partial}}$$

$$i \leftarrow i + 1$$

ouput: $S^{\mathrm{model}}, I^{\mathrm{res}}$

Same Inputs as Högbom's Method

CLEAN: Initialization

$$\textbf{initialize:}\ S^{\text{model}} \leftarrow \{\},\ I^{\text{res}} \leftarrow I^{D},\ i \leftarrow 0,\ (\text{PSF}_{\text{sub}}(l,m),\ R_{\text{PSF}}) \leftarrow g(\text{PSF}(l,m))$$

A function which selects a subset of the PSF and reports the highest PSF sidelobes.

Most of the power in the PSF is centred around the main lobe → we only need a subset of the PSF

For the *minor cycle* we do a shallow deconvolution down to the level of the highest sidelobe.

CLEAN: Minor Cycle

$$l_{\max}, m_{\max} \leftarrow \underset{l,m}{\operatorname{argmax}}\, I^{\text{res}}(l, m)$$

$$f_{\max} \leftarrow I^{D}(l_{\max}, m_{\max})$$

$$S^{\text{model}}_{\text{partial}} \leftarrow \text{Hogbom}(I^{\text{res}},\ \text{PSF}_{\text{sub}},\ \gamma,\ f_{\max} \cdot R_{\text{PSF}})$$

The minor cycle is a shallow cycle of Högbom's method to a flux threshold determined by the highest PSF sidelobes to produce a partial sky model.

CLEAN: Major Cycle

$$\textbf{while } \text{any}(I^{\text{res}} > f_{\text{thresh}}) \text{ or } i \leq N \textbf{ do:}$$

$$\text{MINOR CYCLE}$$

$$V^{\text{model}}_{\text{partial}} \leftarrow \mathcal{F}\{S^{\text{model}}_{\text{partial}}\}, V^S \leftarrow \mathcal{F}\{\text{PSF}\}$$

$$I^{\text{res}} \leftarrow I^{\text{res}} - \mathcal{F}^{-1}\{V^S \cdot V^{\text{model}}_{\text{partial}}\}$$

$$S^{\text{model}} \leftarrow S^{\text{model}} + S^{\text{model}}_{\text{partial}}$$

After the minor cycle, Fourier transform the partial sky model into visibilities, combine with the visibility sampling function and produce a partial sky model image.

Subtract the partial sky model image from the residual image, update full sky model with partial sky model.

CLEAN: Cotton-Schwab's Method (Visibility-domain)

Standard method which is implemented in most modern deconvolving imagers.

Requires the use of gridder/de-gridder functions, computationally more expensive but produces more accurate results.

Ungridded Visibilities → $\mathcal{V}^{\text{model}}_{\text{partial}}$

$$\mathcal{V}^{\text{model}}_{\text{partial}} \leftarrow \text{degrid}(\mathcal{F}\{S^{\text{model}}_{\text{partial}}\})$$

$$\mathcal{V}^{\text{residual}} \leftarrow \mathcal{V}^{\text{residual}} - \mathcal{V}^{\text{model}}_{\text{partial}}$$

$$V^{\text{residual}} \leftarrow \text{grid}(\mathcal{V}^{\text{residual}})$$

$$I^{\text{residual}} \leftarrow \mathcal{F}^{-1}\{V^{\text{residual}}\}$$

$$S^{\text{model}} \leftarrow S^{\text{model}} + S^{\text{model}}_{\text{partial}}$$

Method Comparison

Högbom (image-domain):

- pro: easy to implement
- con: limited accuracy in PSF subtraction (e.g w-term effects)
- con: can not account for aliasing artefacts

Clark (gridded visibility-domain):

- pro: only minimally more effort to implement compared to Högbom
- pro: improved aliasing response
- con: limited accuracy in PSF subtraction (e.g w-term effects)

Cotton-Schwab (visibility-domain):

- pro: accurate subtraction of sky model, we can include beam and w-term effects
- con: computationally expensive

Method Comparison

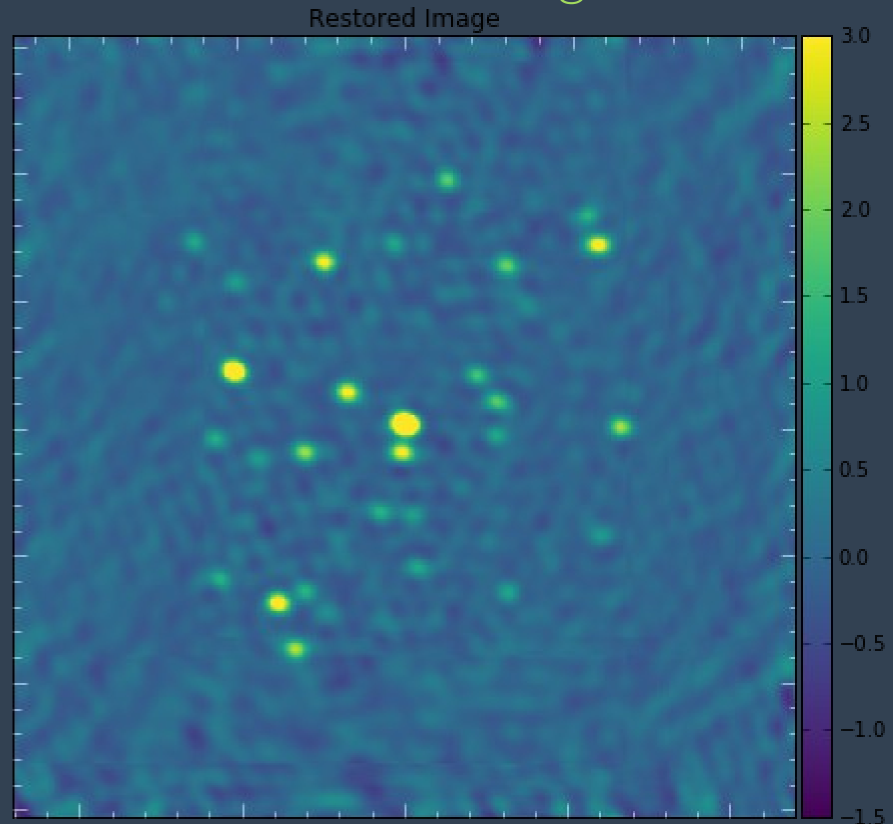
Short version:

Högbom and Clark methods are easy to implement (the next assignment is to implement a portion of Clark's method).

But, in almost **all** cases you should use the Cotton-Schwab method as computation costs are not really to much of a problem these days.

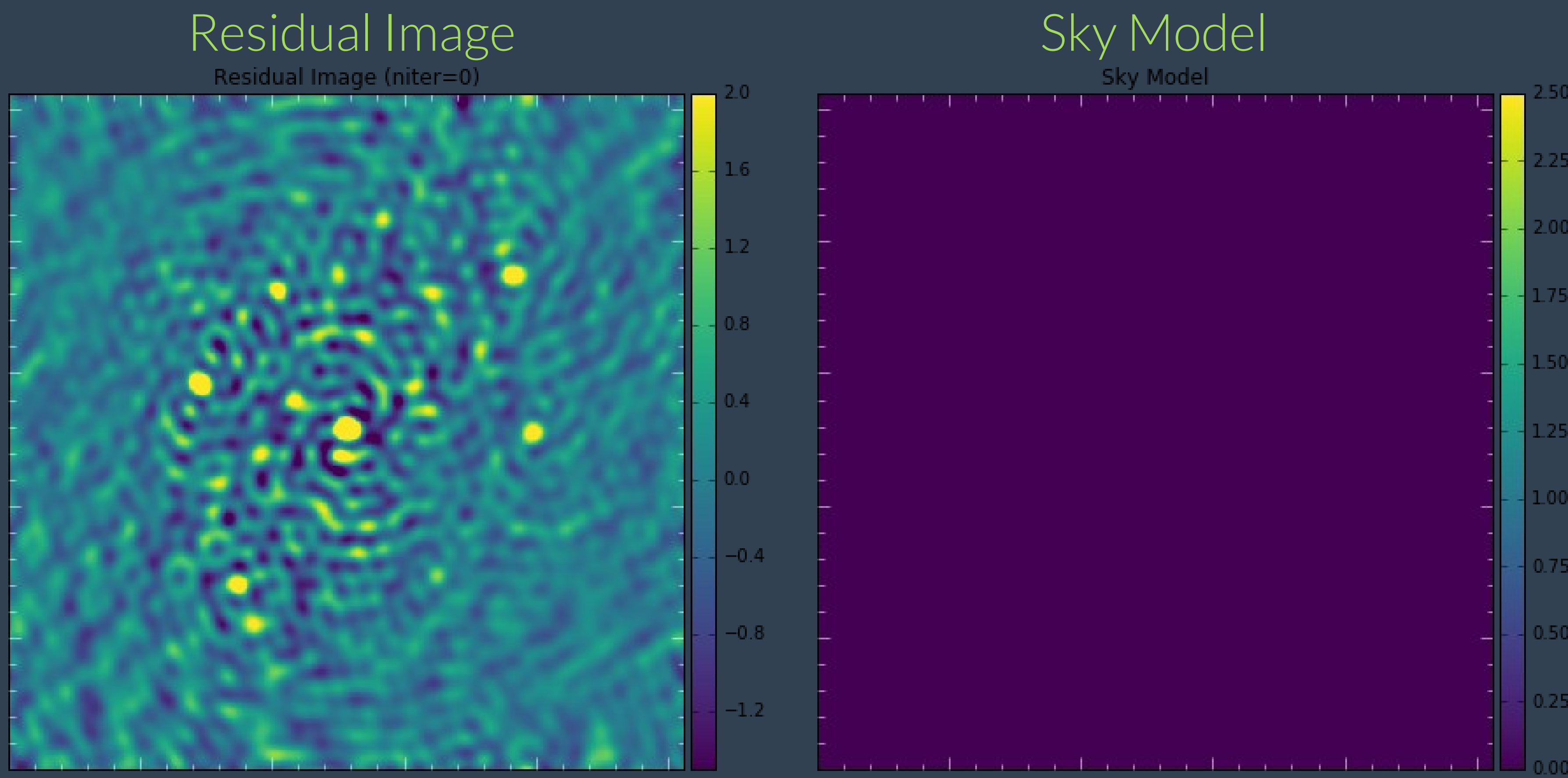
Idealized Synthesis Telescope Image

Restored Image



$$I_{\text{restored}} = I_{\text{skymodel}} \circ \text{PSF}_{\text{ideal}} + I_{\text{residual}}$$

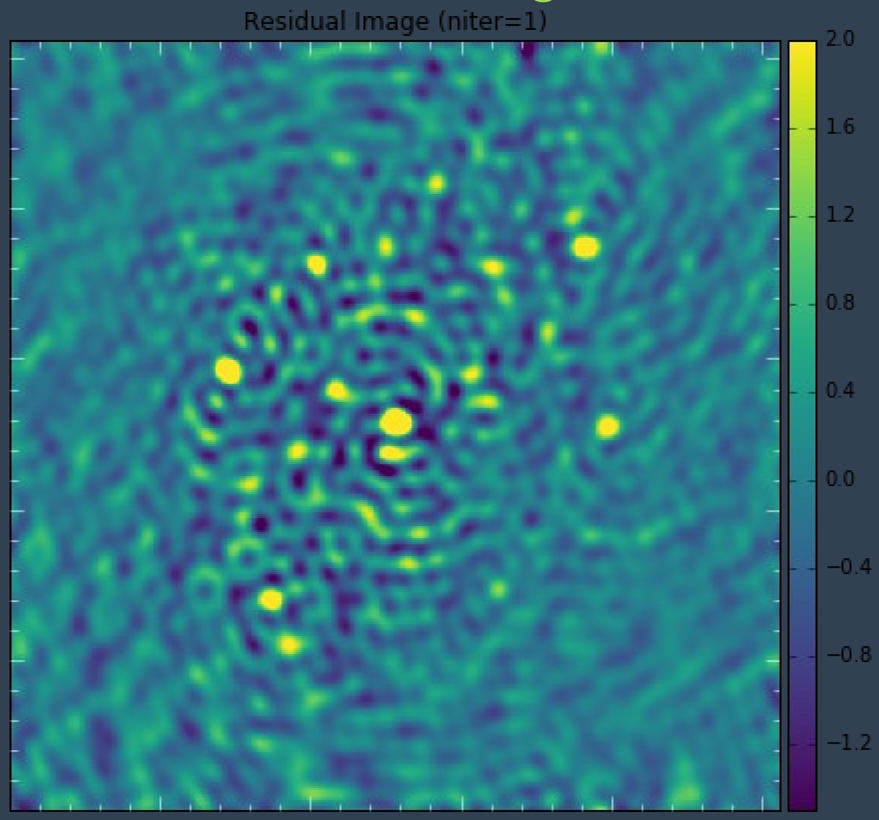
CLEAN: KAT-7 Example



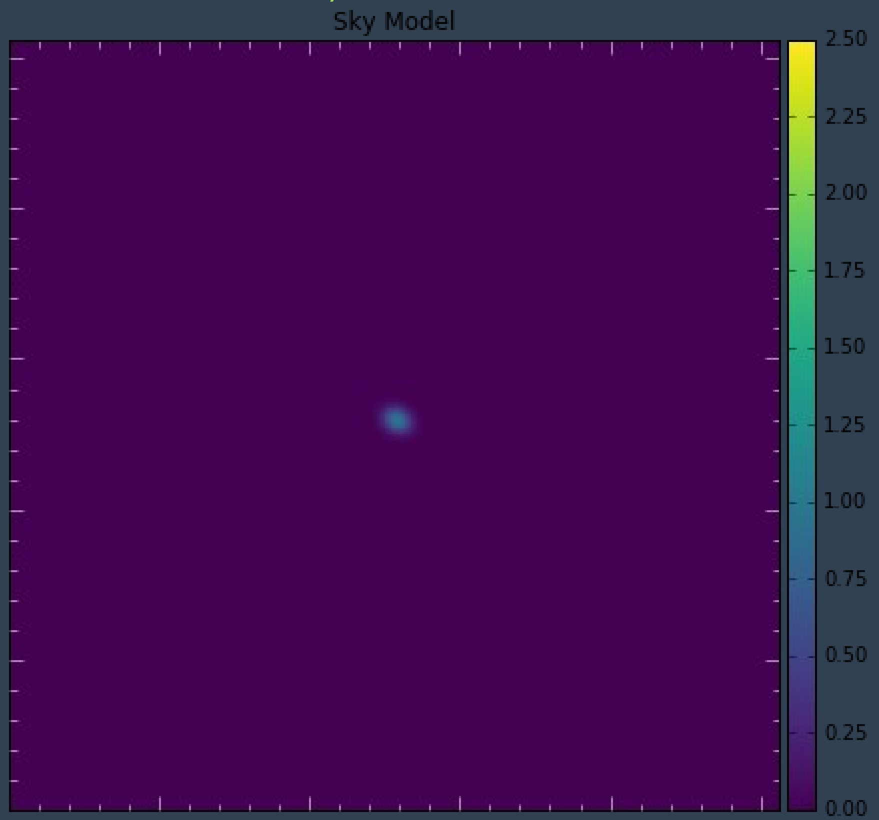
$N_{iterations} = 0$

CLEAN: KAT-7 Example

Residual Image



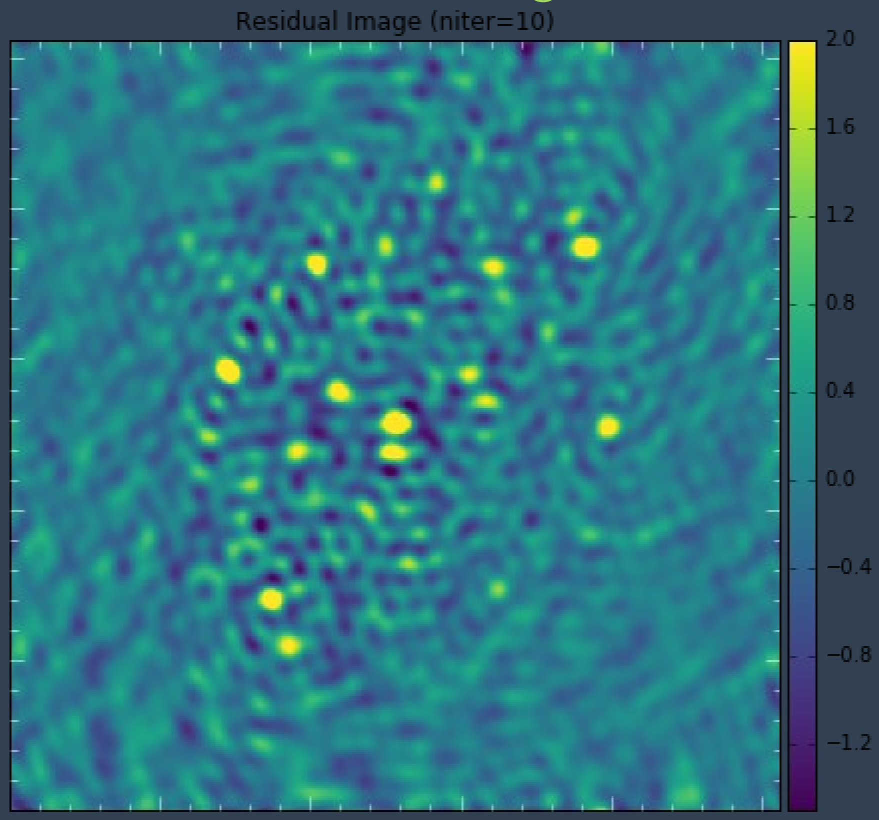
Sky Model



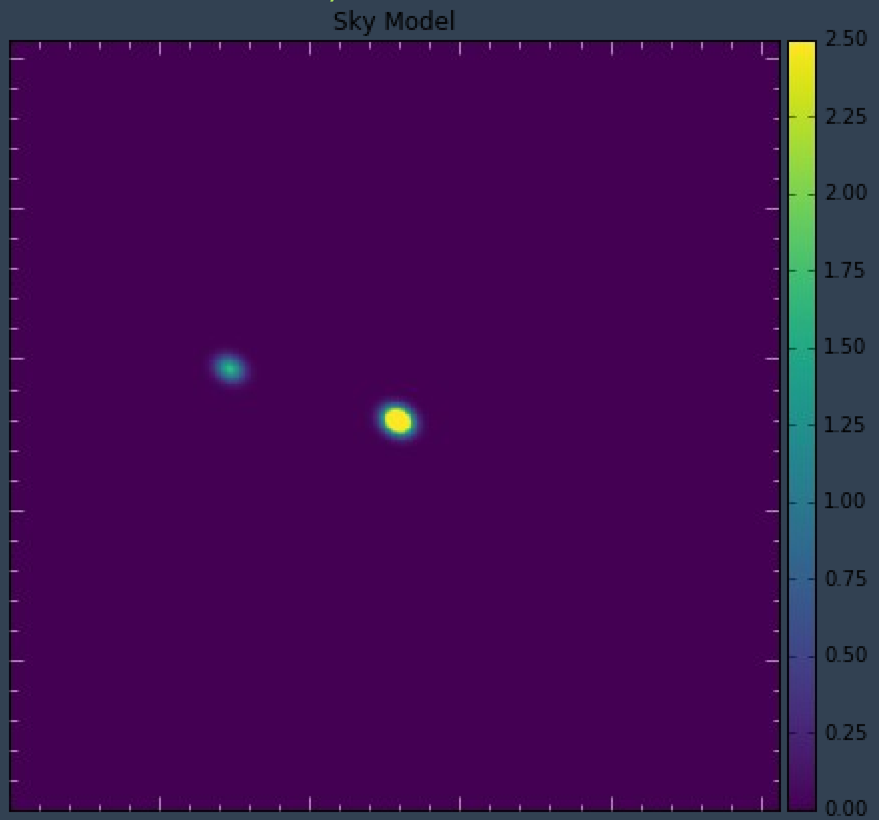
$N_{iterations} = 1$

CLEAN: KAT-7 Example

Residual Image

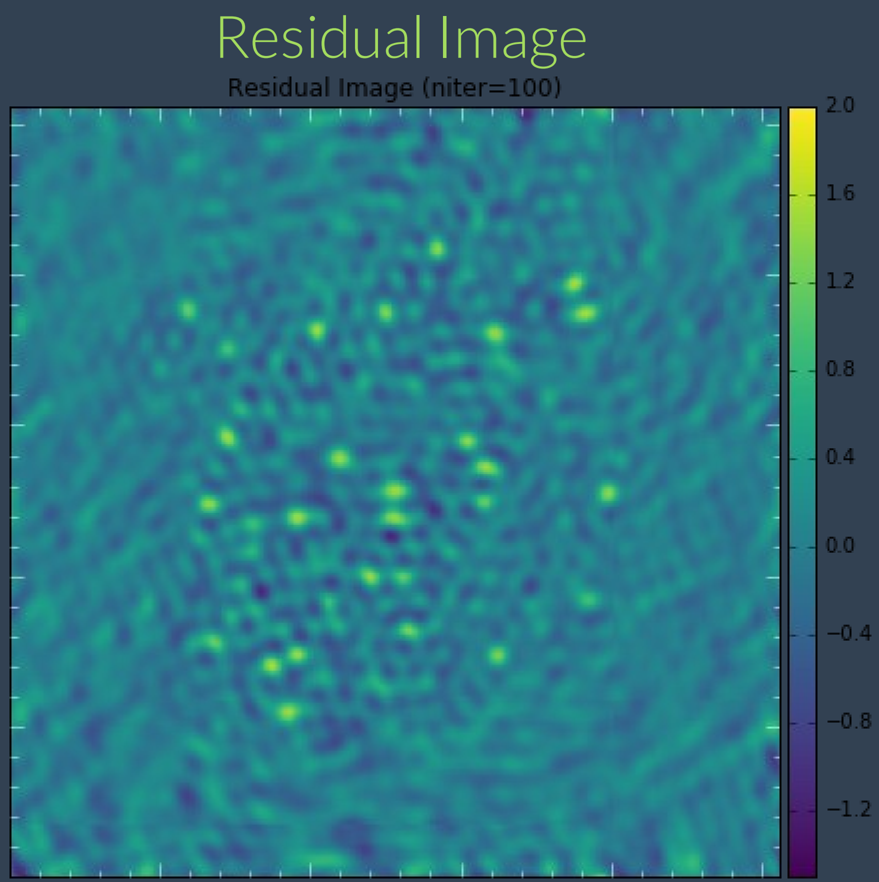


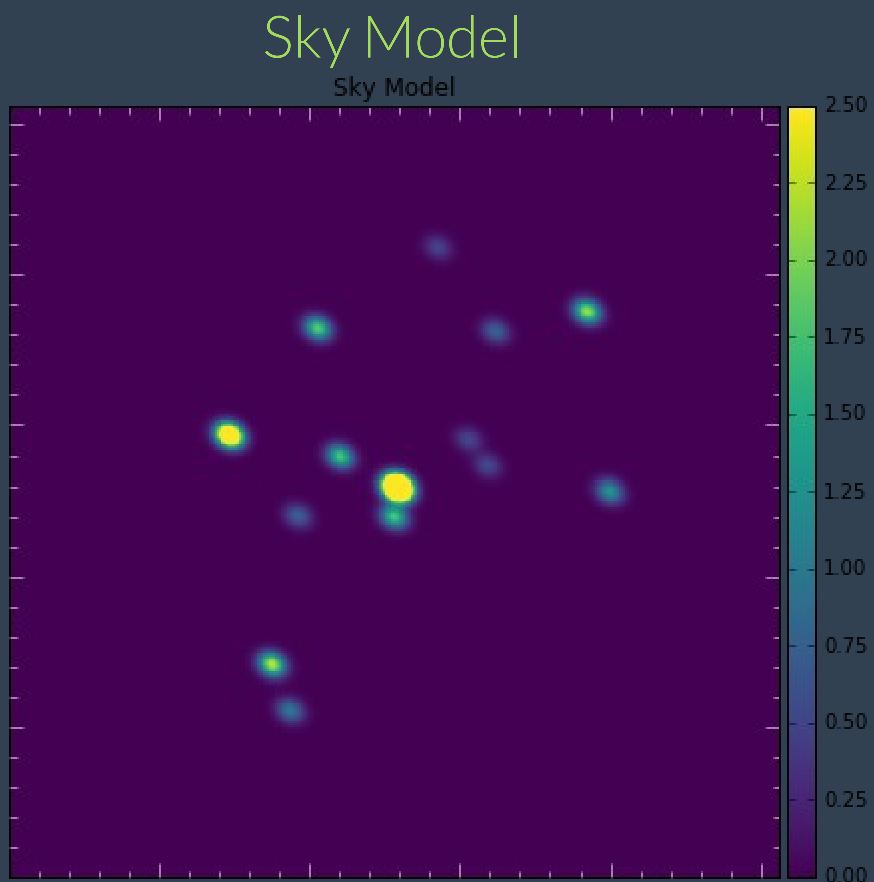
Sky Model



$N_{iterations} = 10$

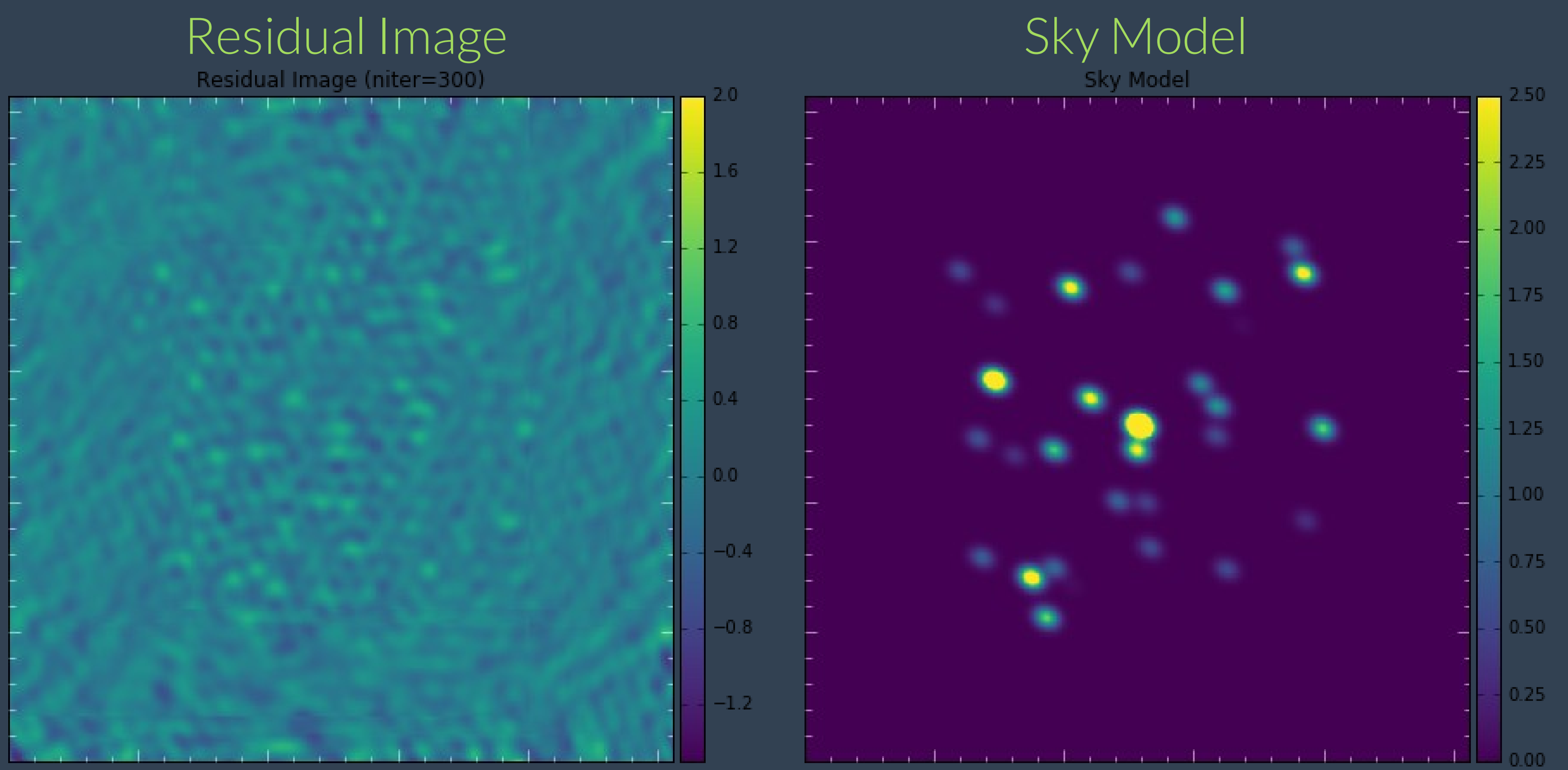
CLEAN: KAT-7 Example





$N_{iterations} = 100$

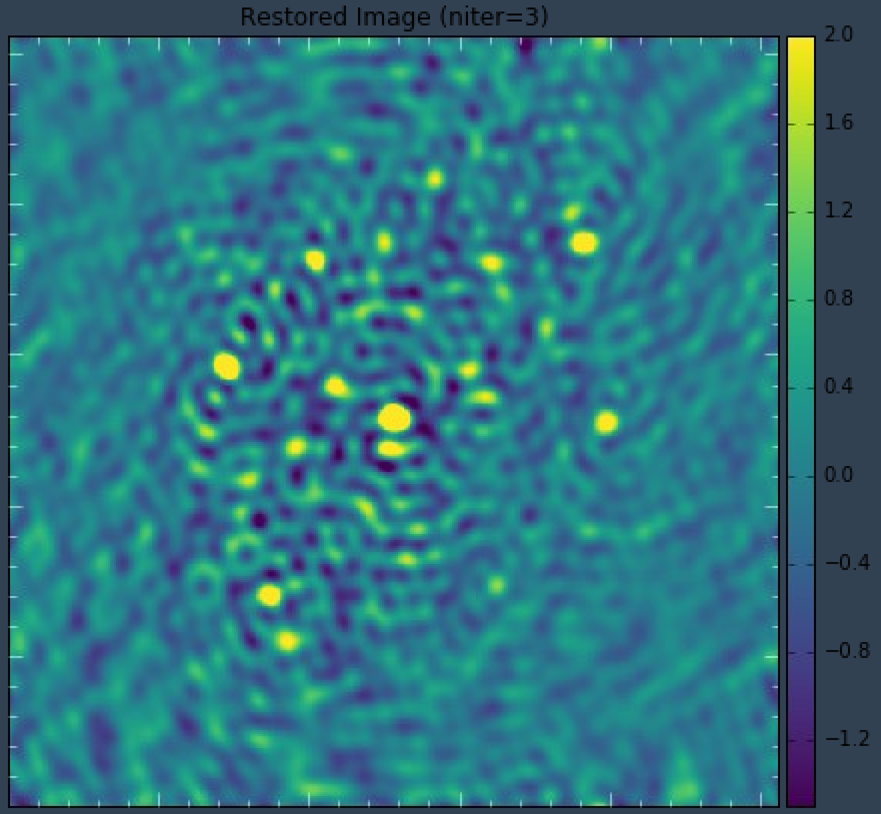
CLEAN: KAT-7 Example



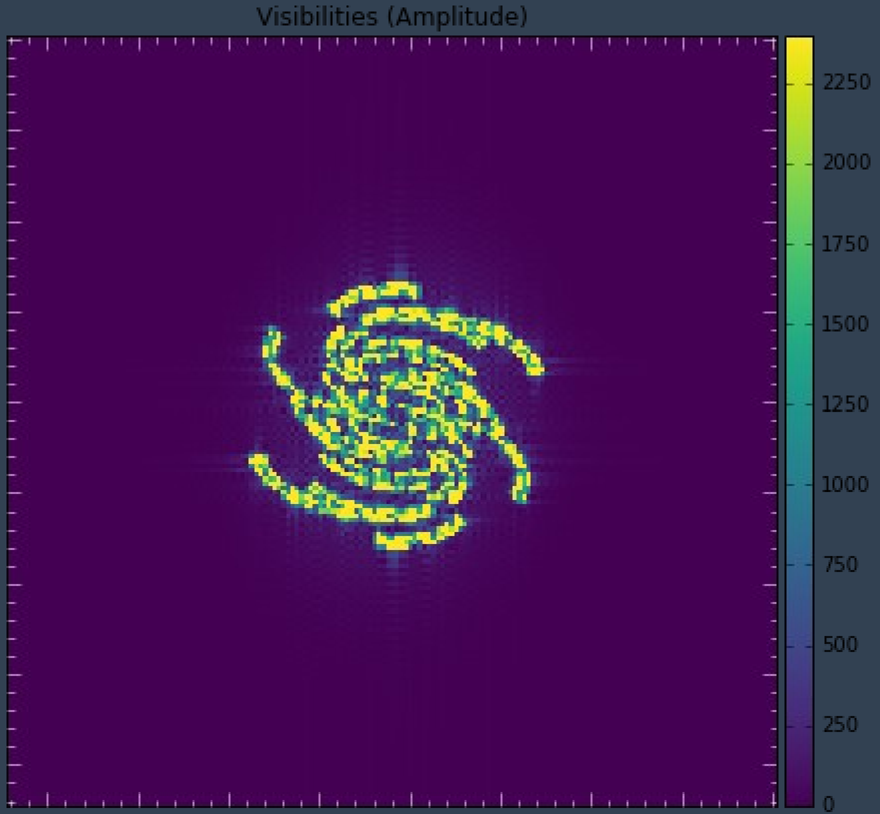
$N_{iterations} = 300$

CLEAN: Filling in the Visibility Space

Restored Image



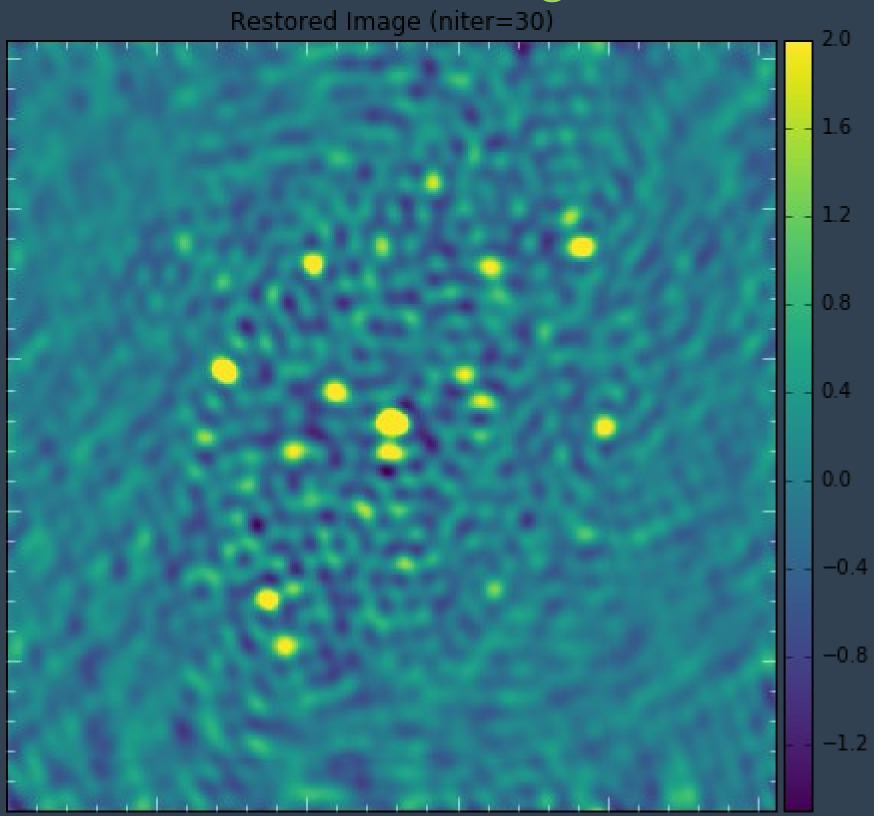
Restored Visibilities



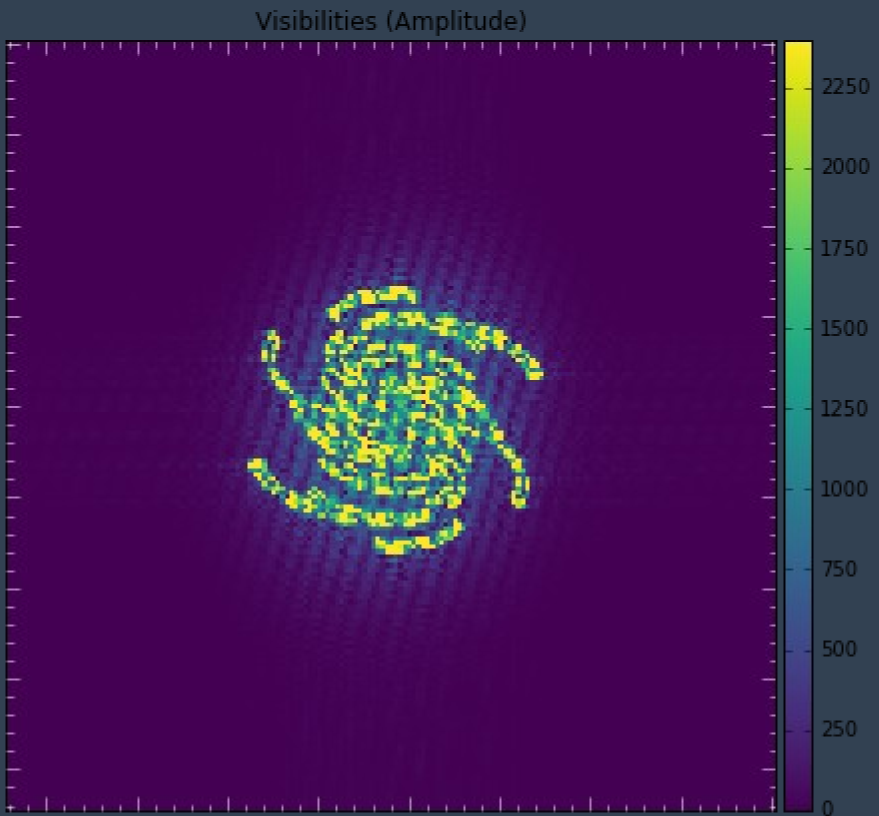
$N_{iterations} = 3$

CLEAN: Filling in the Visibility Space

Restored Image



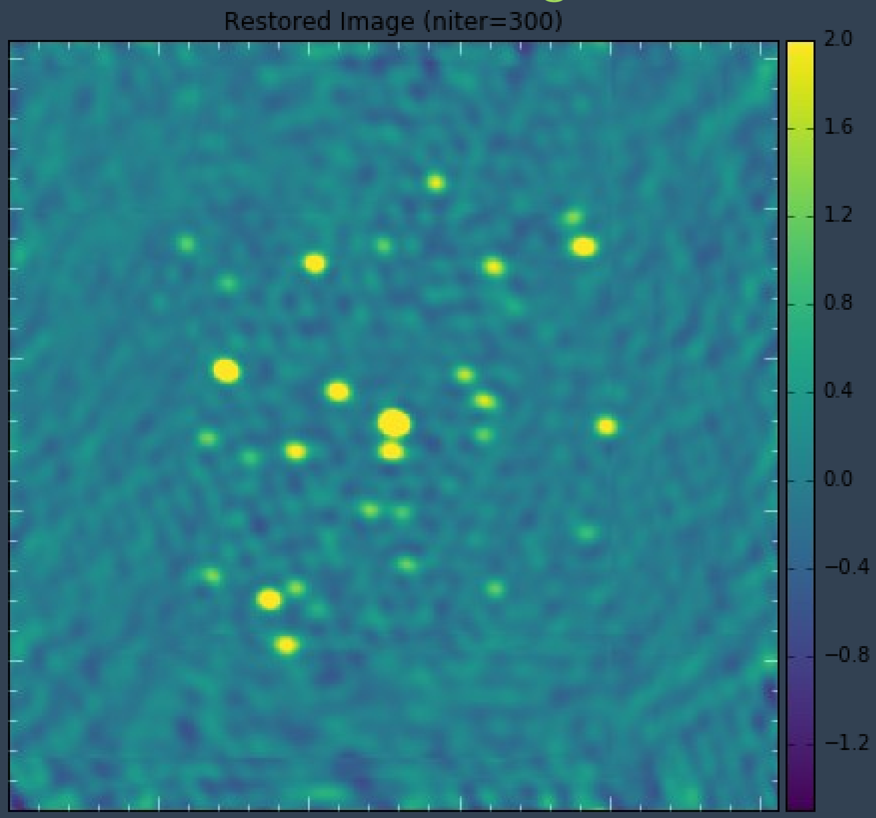
Restored Visibilities



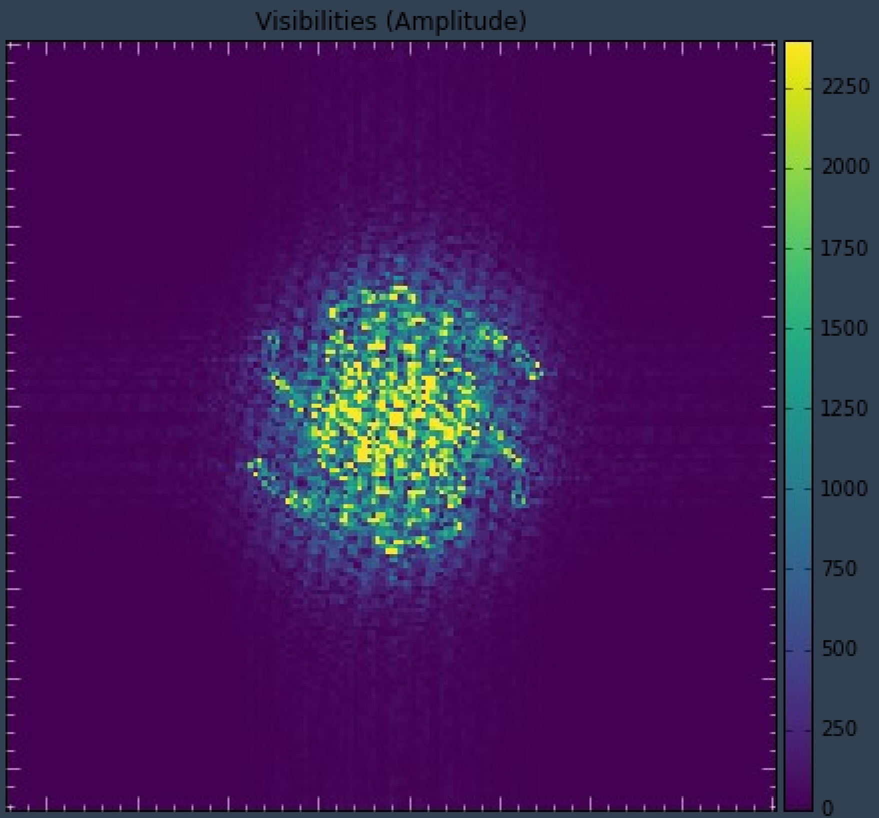
$N_{iterations} = 30$

CLEAN: Filling in the Visibility Space

Restored Image



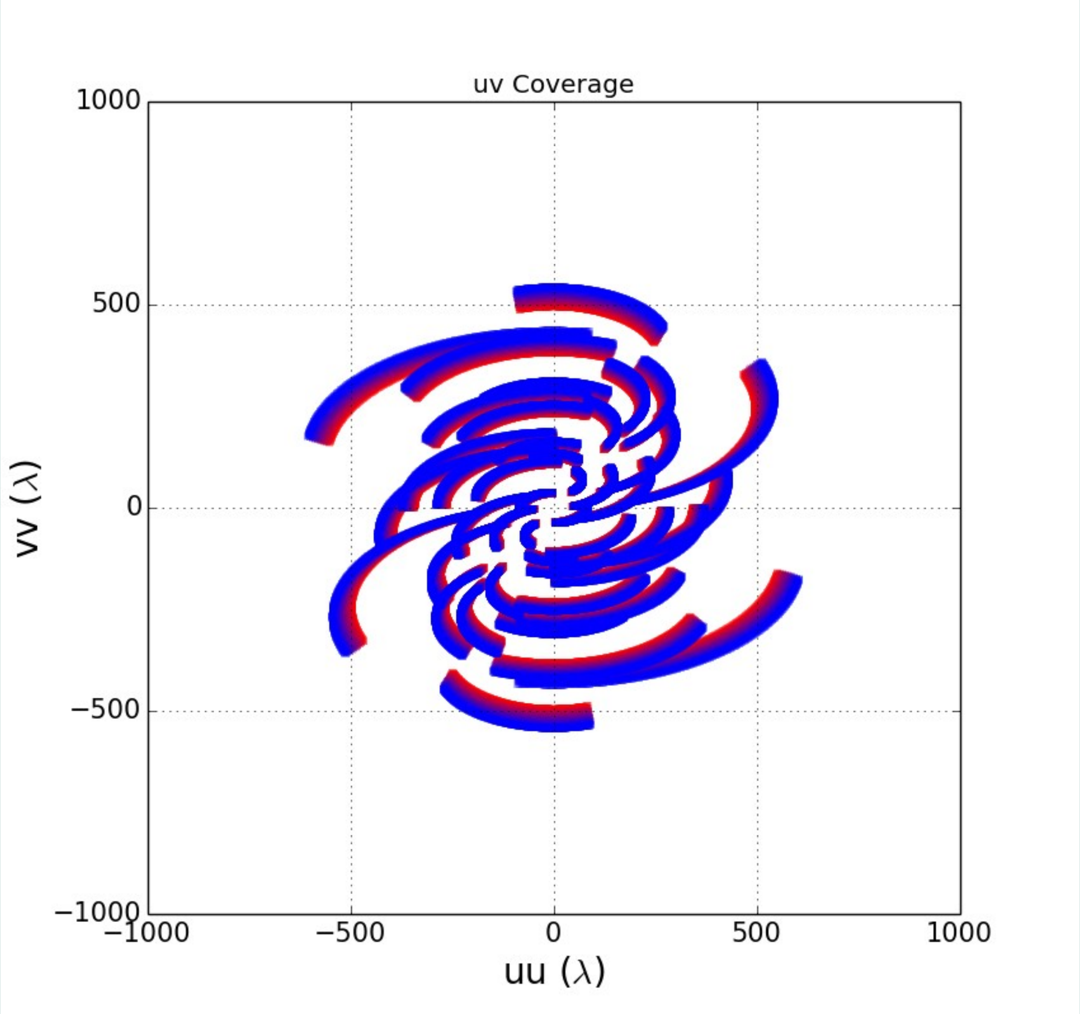
Restored Visibilities



$N_{iterations} = 300$

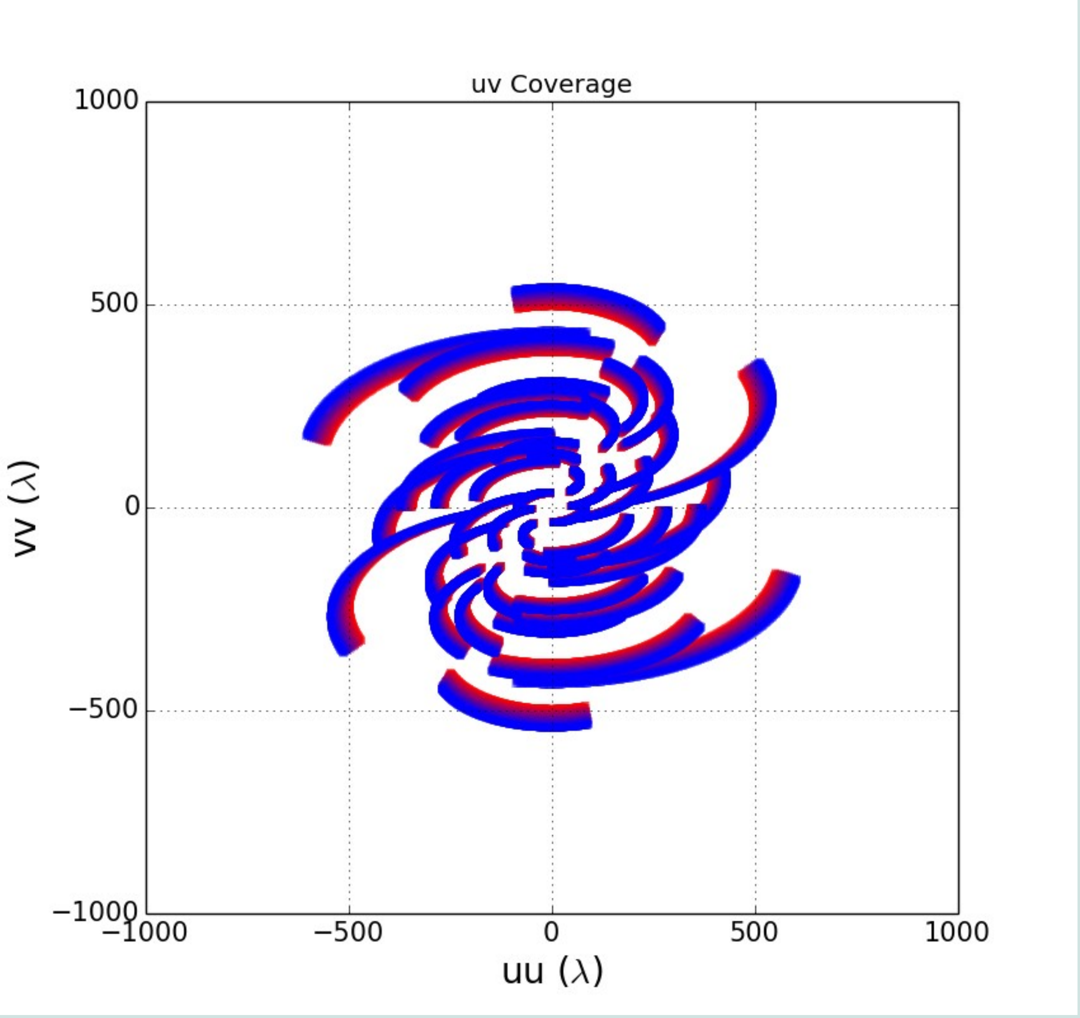
Limits of CLEAN: Multi-Frequency Deconvolution

A baseline length is in units of wavelength, for an array which observes at multiple frequencies this means that baselines are 'shorter' for lower frequencies compared to higher frequencies.



Limits of CLEAN: Multi-Frequency Deconvolution

This means that the PSF resolution *scales* as a function of frequency. What does it mean to make a multi-frequency image? What is the ideal size PSF if the PSF changes?



Limits of CLEAN: Multi-Frequency Deconvolution

Channel Imaging Method:

- make a dirty image and PSF for each frequency channel
- perform deconvolution
- average together the images to produce a single image

Pro: can account for the different PSF scale for each frequency channel

Con: reduced signal to noise by not combining all the channels leading to a shallower deconvolution

Multi-frequency Synthesis:

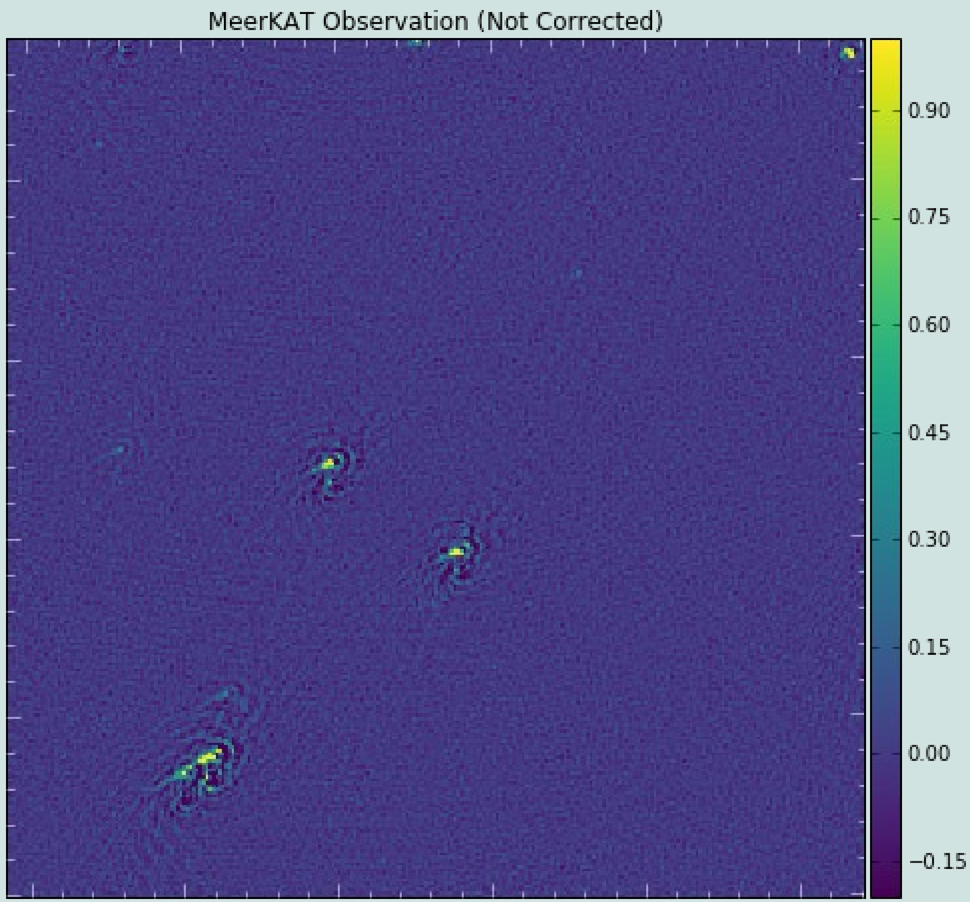
- make a dirty image and PSF using all channels
- perform deconvolution with an *average* PSF
- use an *average* ideal PSF to produce a restored image

Pro: maximizes signal to noise for a deeper deconvolution

Con: for wide band observations this leads to 'holes' around sources due to the average PSF subtraction

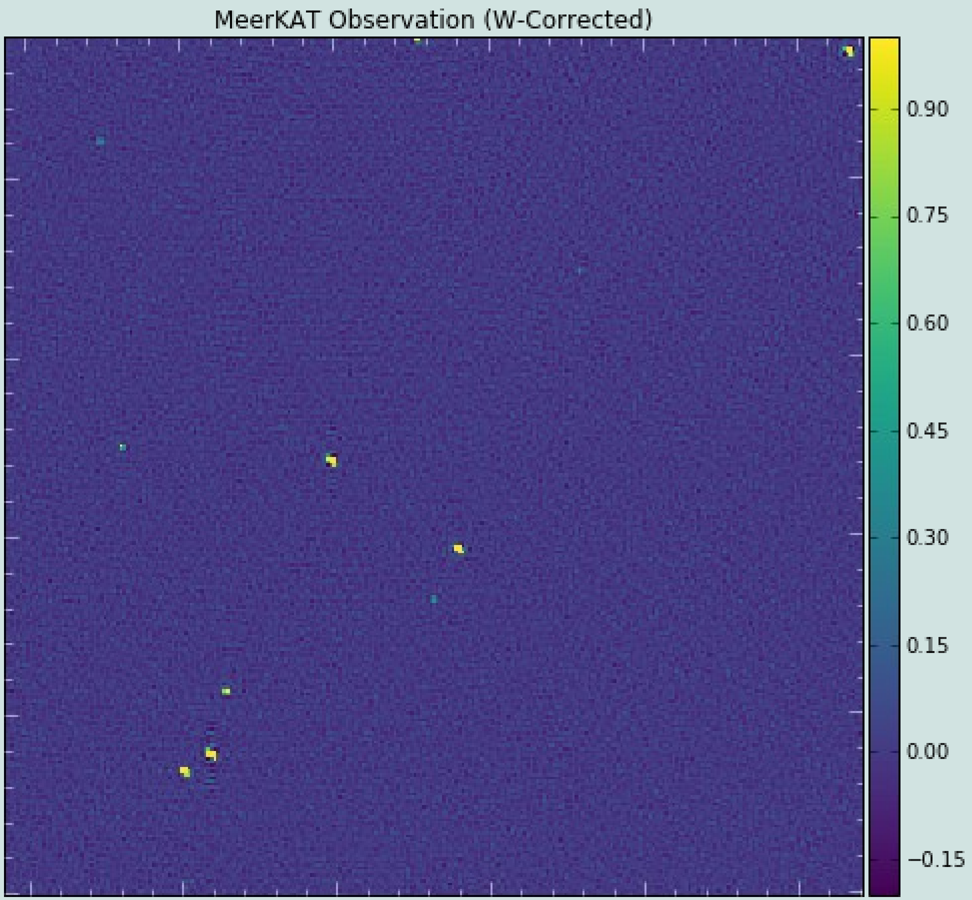
Limits of CLEAN: W-term Approximation

The flat-field approximation leads to w-term effects. The w-term can be seen as a phase offset → a phase offset is a change in position → the PSF is 'smeared' out as a function of distance from the phase centre.



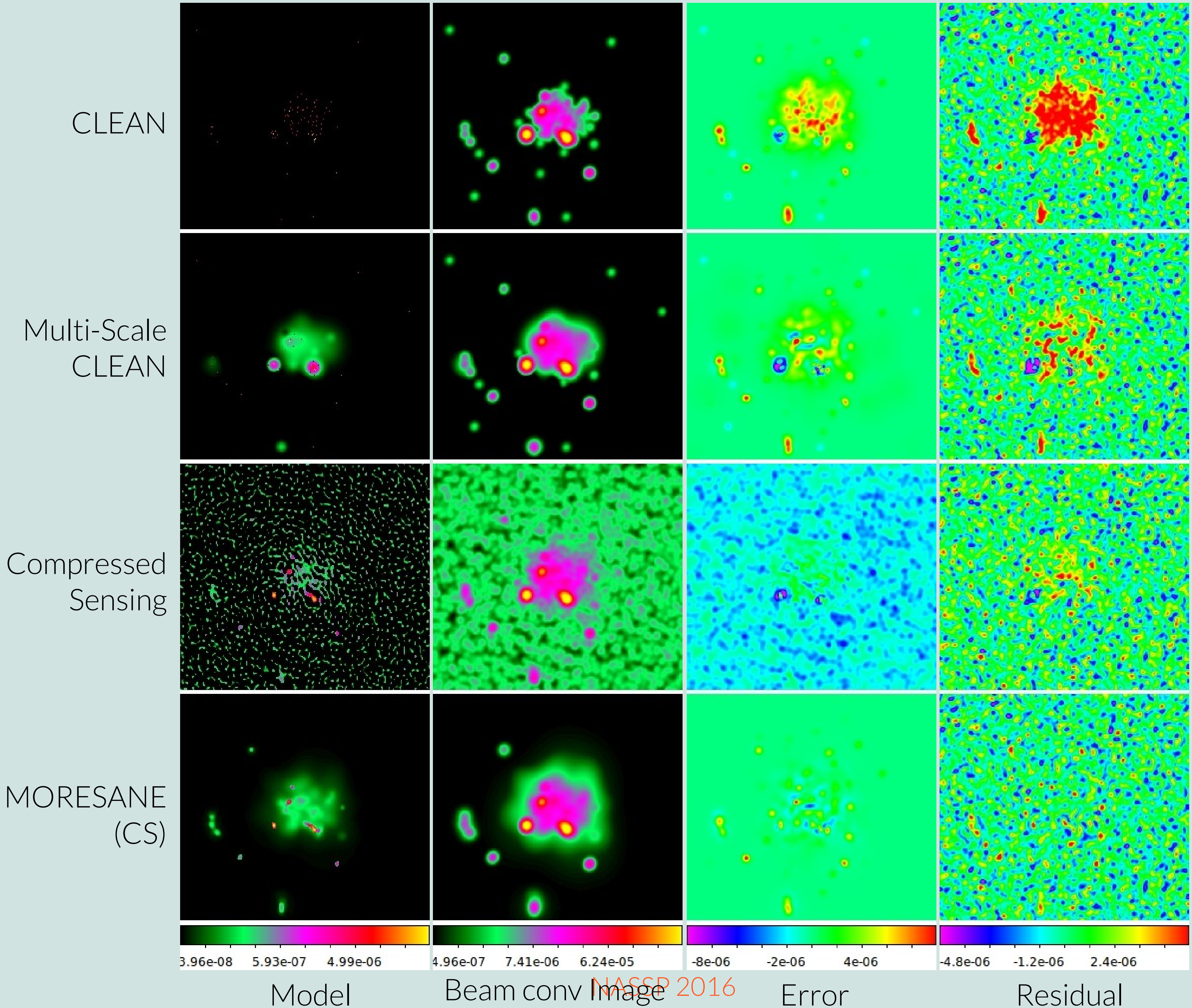
Limits of CLEAN: W-term Approximation

Using Cotton-Schwab's method to do deconvolution in the visibility domain allows for w-term correction (at a computational cost).



Limits of CLEAN: Extended Sources

Dabbech et al 2014



Standard Imagers

CASA clean : full-featured imager and deconvolver included in NRAO's CASA package. (casa.nrao.edu/docs/TaskRef/clean-task.html)

lwimager : light-weight imager and deconvolver, stable but new features are not being added. (github.com/casacore/casarest)

wsclean : generic widefield imager and deconvolver. (sourceforge.net/projects/wsclean/)

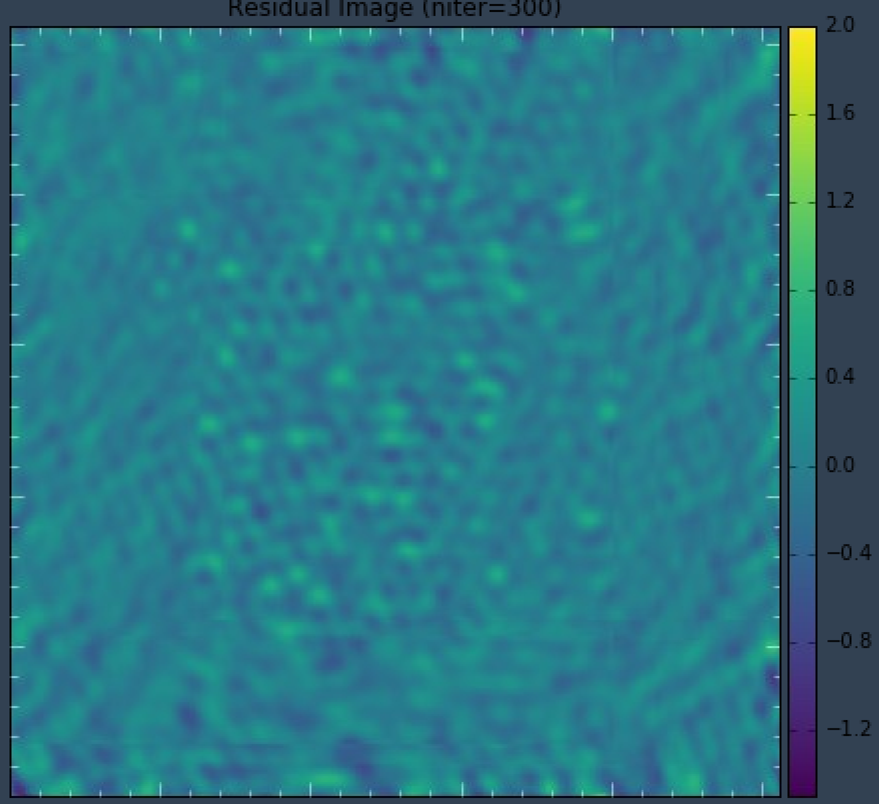
When should you halt the deconvolution process?

i.e.

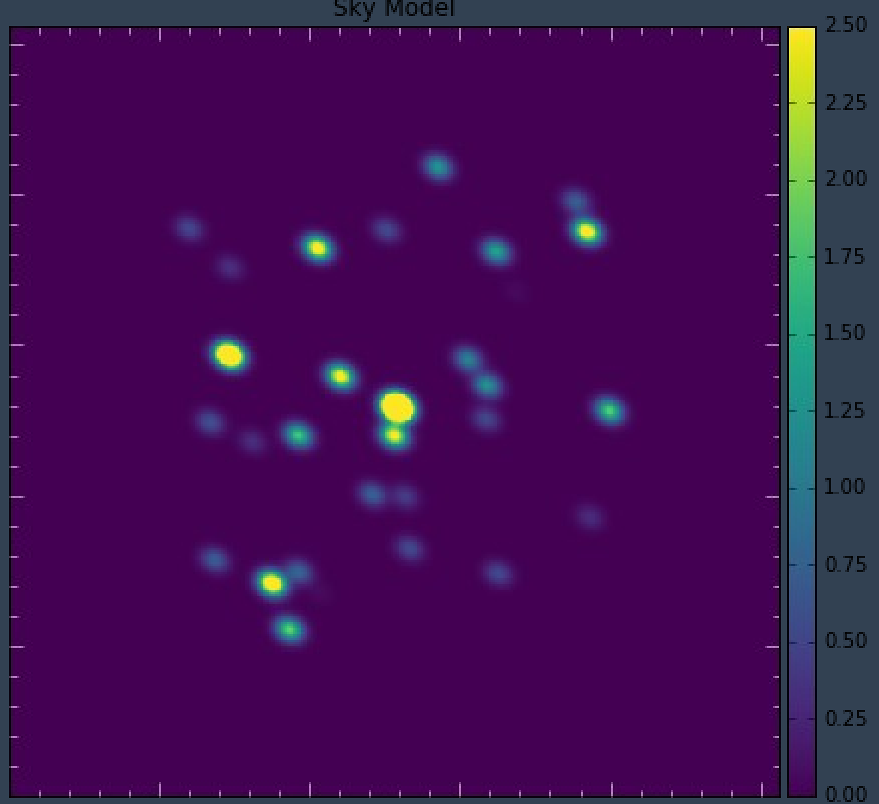
What makes a 'good' image?

Halting Deconvolution

Residual Image



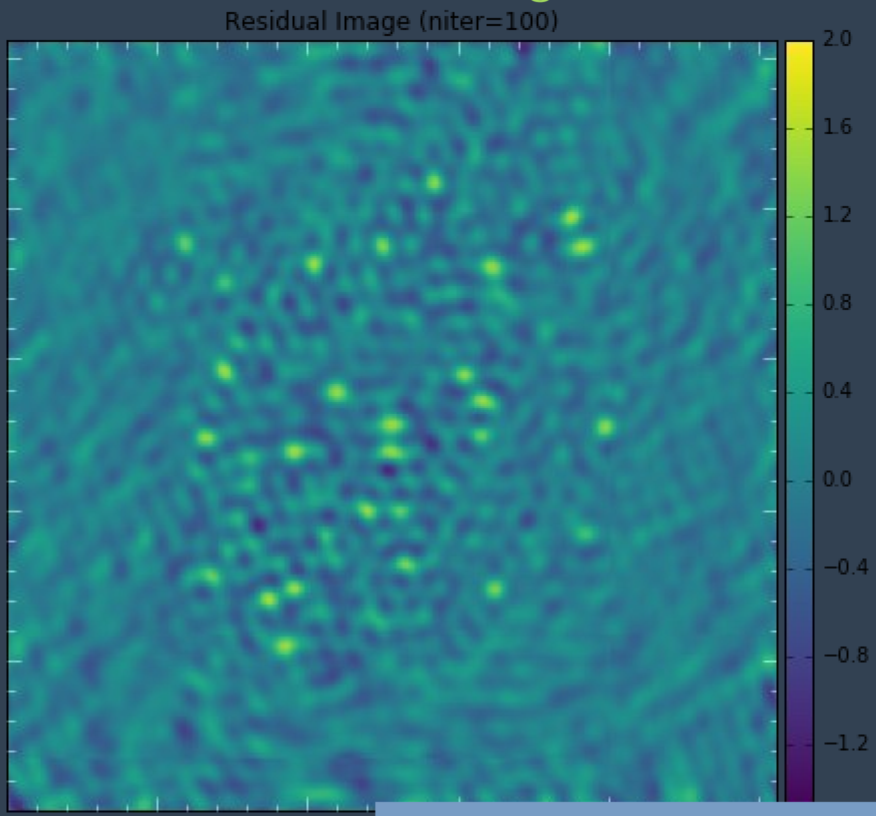
Sky Model



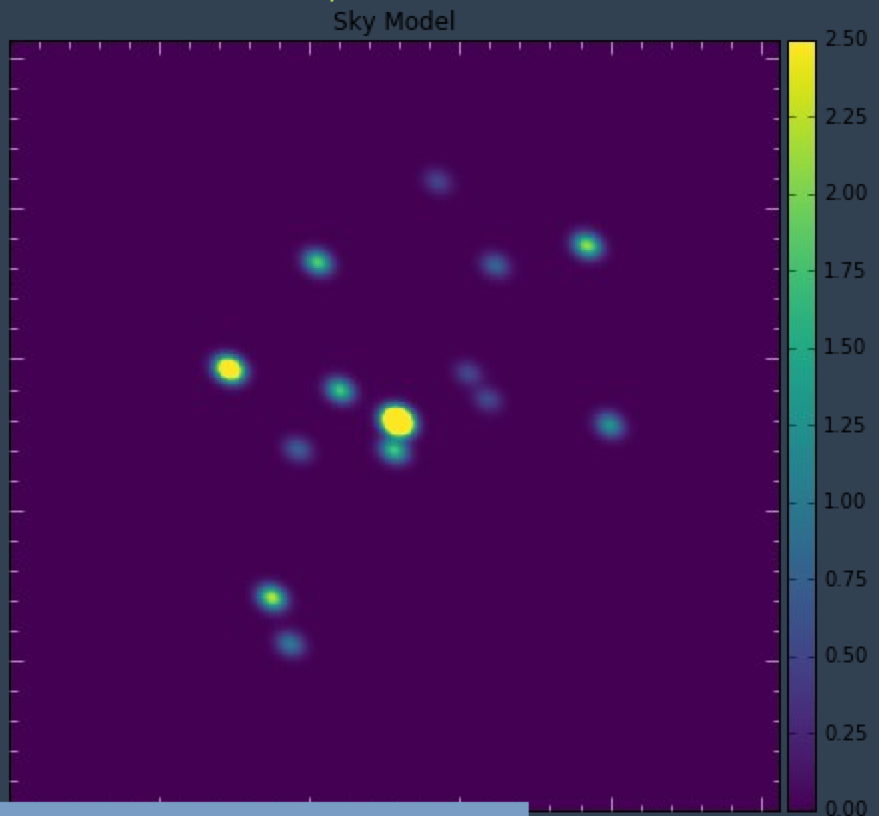
$N_{iterations} = 300$

Halting Deconvolution

Residual Image



Sky Model

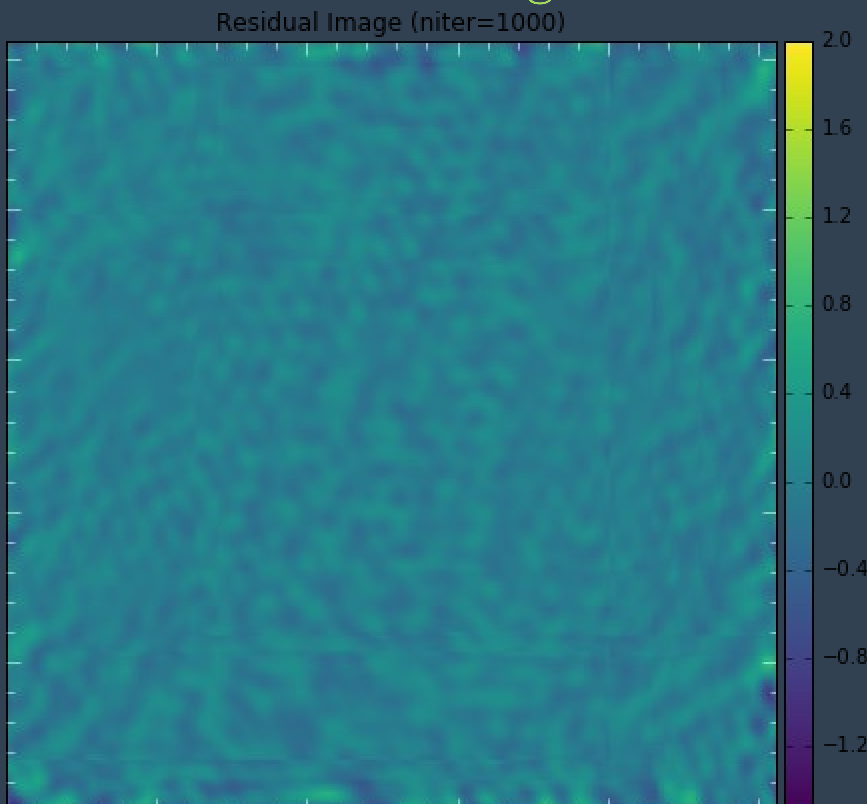


Under Deconvolved → limits the resulting sky model signal

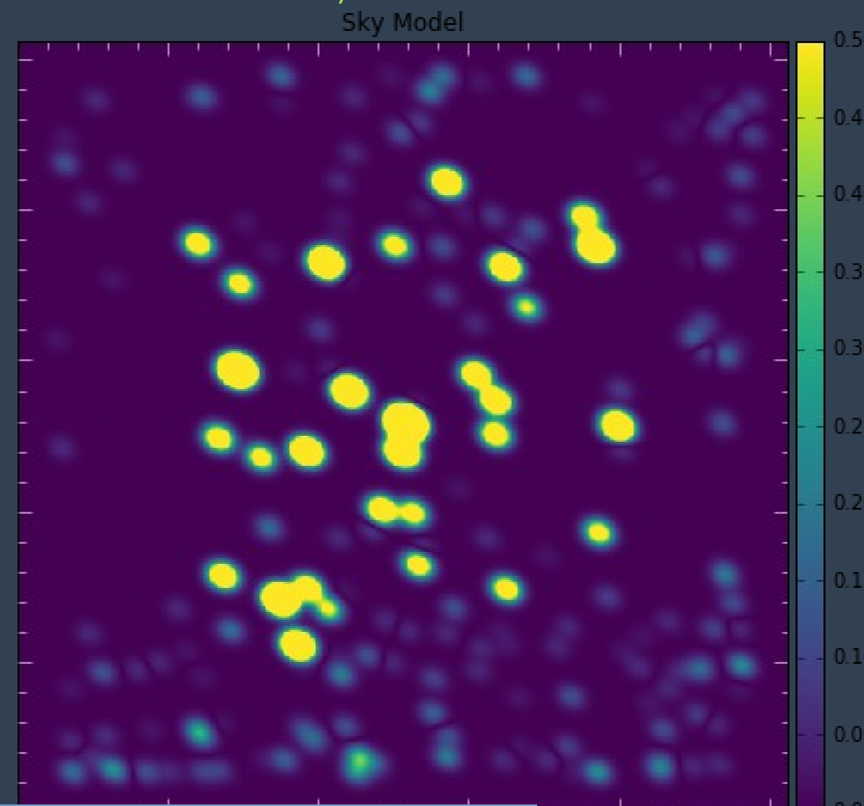
$N_{iterations} = 1000$

Halting Deconvolution

Residual Image



Sky Model



Over Deconvolved → noise being inserted into the sky model

$N_{iterations} = 1000$

Halting Deconvolution

Q: When should you halt the deconvolution process?

A: It is a bit ad-hoc, an interesting problem that has not been well solved.

Usually, it is based on intuition and examining the residuals and sky model for different levels of deconvolution.

Halting deconvolution is closely connected with calibration leading to a degeneracy issue.

What makes a 'good' image?

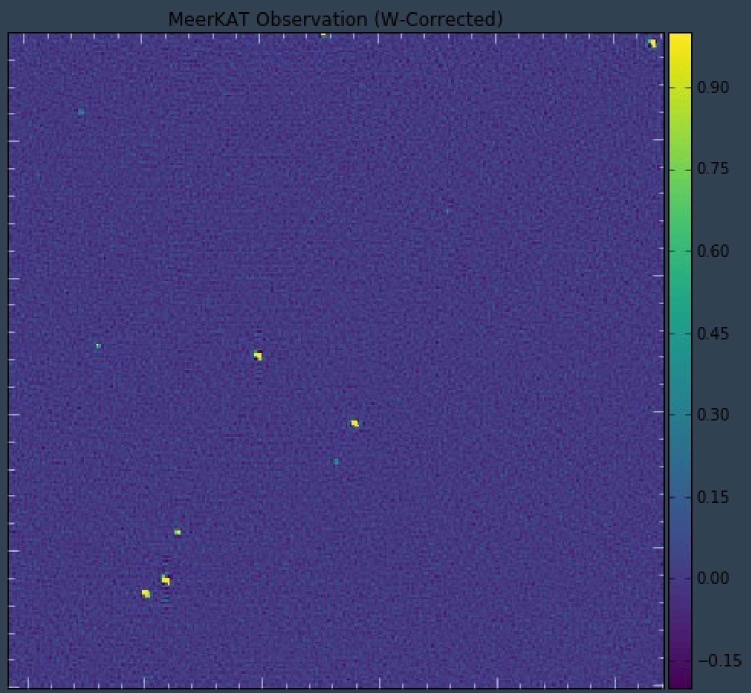
Standard metric is the *dynamic range*:

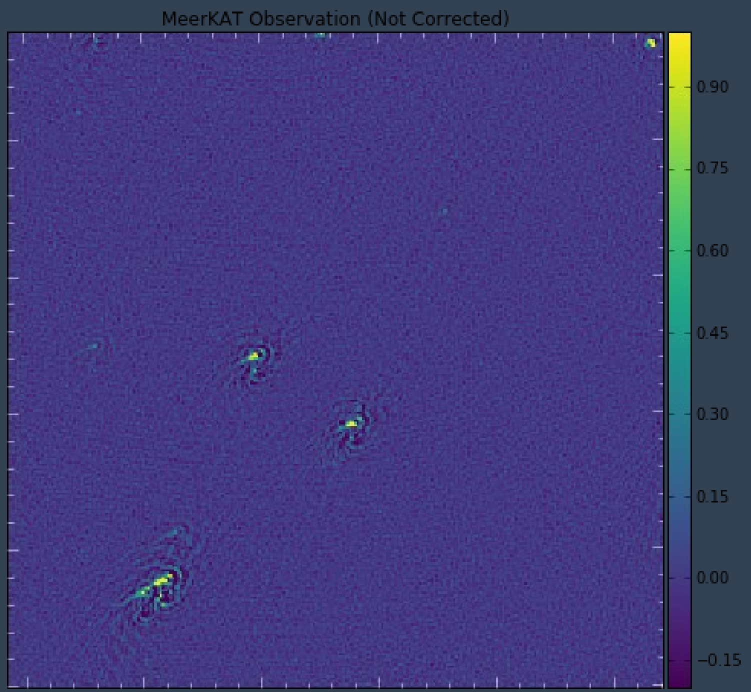
$$\mathrm{DR} = \frac{I_{\mathrm{peak}}}{\sigma_I}$$

The ratio of the peak flux of the restored image to the noise of the image.

Limitation of Dynamic Range

An overall metric which provides *no information about local variations*. In sparse images, such as interferometric images there are only a few sources and mostly noise, then *artefacts* (errors due to deconvolution, imaging or calibration) only occur in small, local regions. Dynamic range does not capture this information which the eye can clearly see.





Both images have nearly the same dynamic range,
the one on the right has w-term artefacts

Limitation of Dynamic Range

The denominator of the dynamic range is ill-defined, what is the noise of the image? To calculate the noise there are a number of methods that are **subjective**:

1. Use the entire image
2. Use the entire residual image
3. Randomly sample the image
4. Choose a 'relatively' empty region

Resulting in different dynamic ranges figures:

1. 27.6075
2. 37.6852
3. 31.2751

4a. 38.2564 (using a corner of the image)
4b. 11.8666 (using the centre)

Limitation of Dynamic Range

The denominator of the dynamic range is ill-defined, what is the noise of the image? To calculate the noise there are a number of methods that are **subjective**:

1. Use the entire image
2. Use the entire residual image
3. Randomly sample the image
4. Choose a 'relatively' empty region

Resulting in different dynamic ranges figures:

1. 27.6075
2. 37.6852
3. 31.2751

4a. 38.2564 (using a corner of the image)
4b. 11.8666 (using the centre)

Image Quality Assessment

Notes:

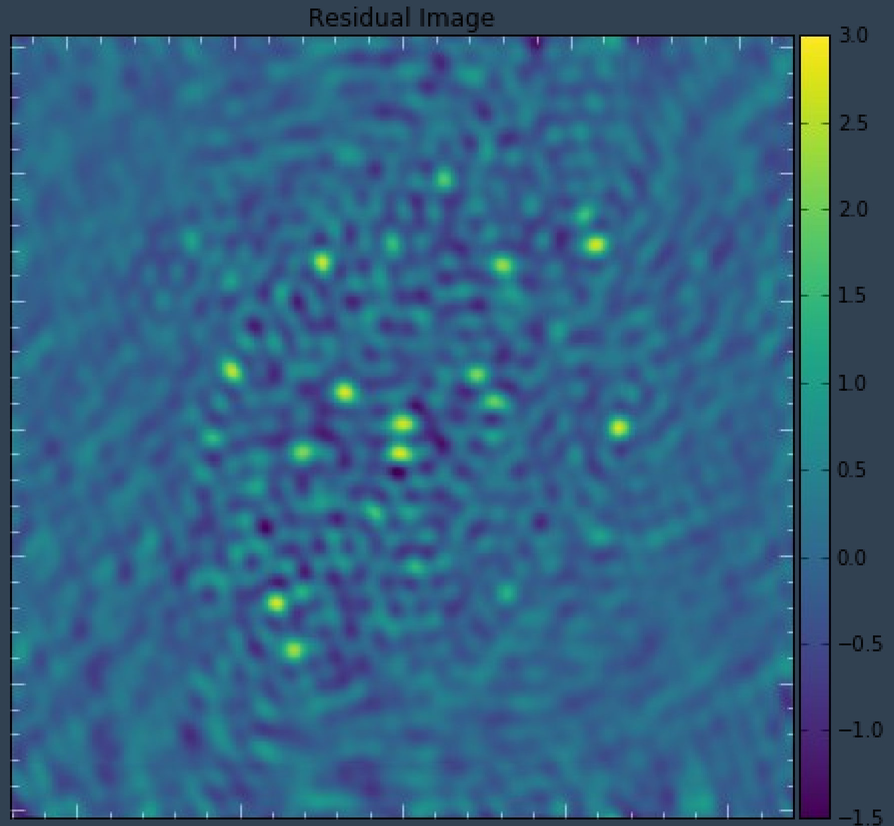
- Look at the residual image for artefacts, the restored image is just a pretty picture.

- Dynamic range is a (weak) proxy for image quality.

- Artefacts are result of imaging, deconvolution, and calibration errors in unison.

- Image Quality Assessment is under-developed in radio interferometry

Residual Image



Source Finding

Clean components:

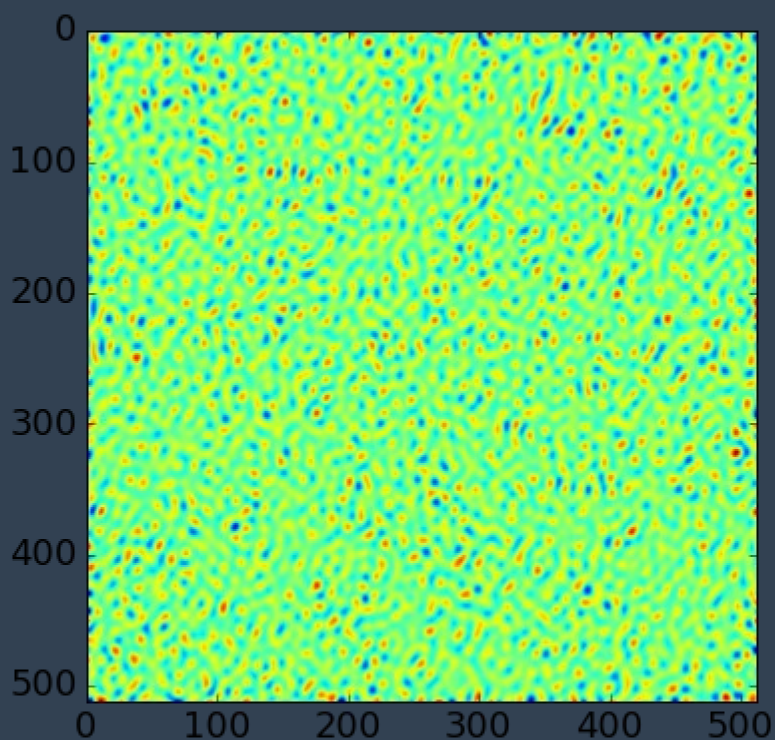
x	*y*	*flux*
32	15	0.273542117836
32	15	0.218833694269
30	34	0.197043506304
32	15	0.175066955415
20	20	0.164478127268
30	34	0.157634805043
31	14	0.141743159144
20	21	0.133470733705
30	34	0.126107844035
32	20	0.124271249713
31	14	0.113394527315
29	18	0.113236796988
19	20	0.11300001035
31	16	0.109407177869
38	21	0.109218346103
21	[illegible]	[illegible]
25	[illegible]	[illegible]
32	9	0.106513135995
30	34	0.100886275228

Same position, different amount of flux. The final sky model is the sum of the different components at the same position

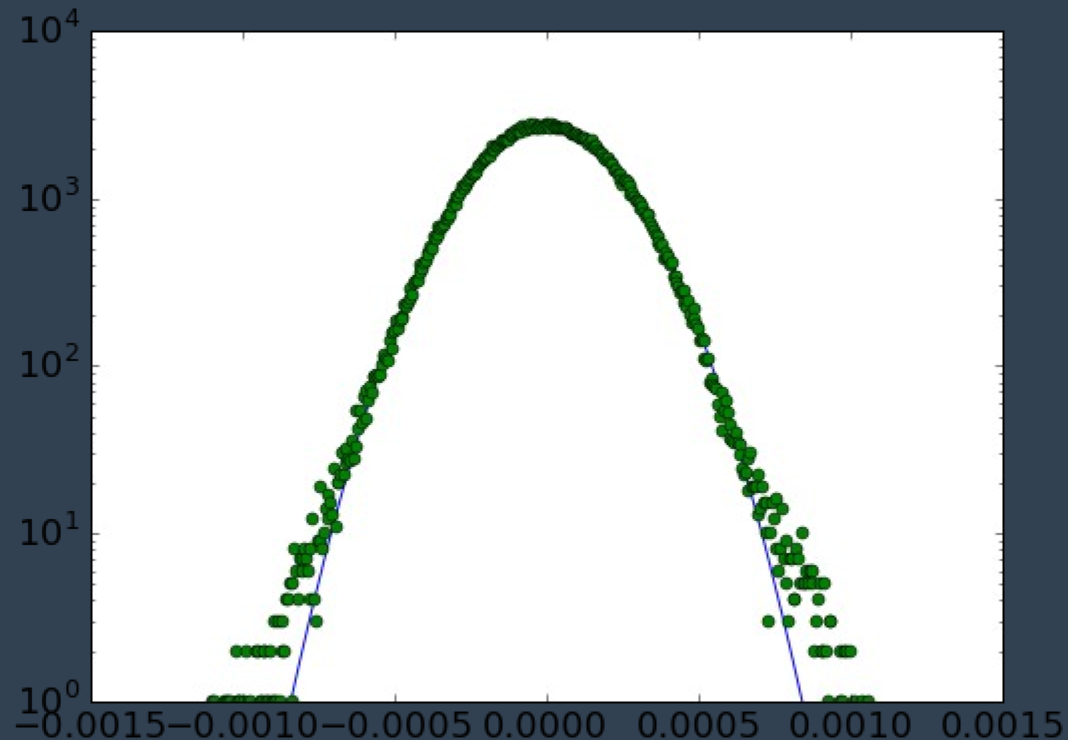
Need a way to combine 'nearby' components into a single source

Source Finding

Noise Image

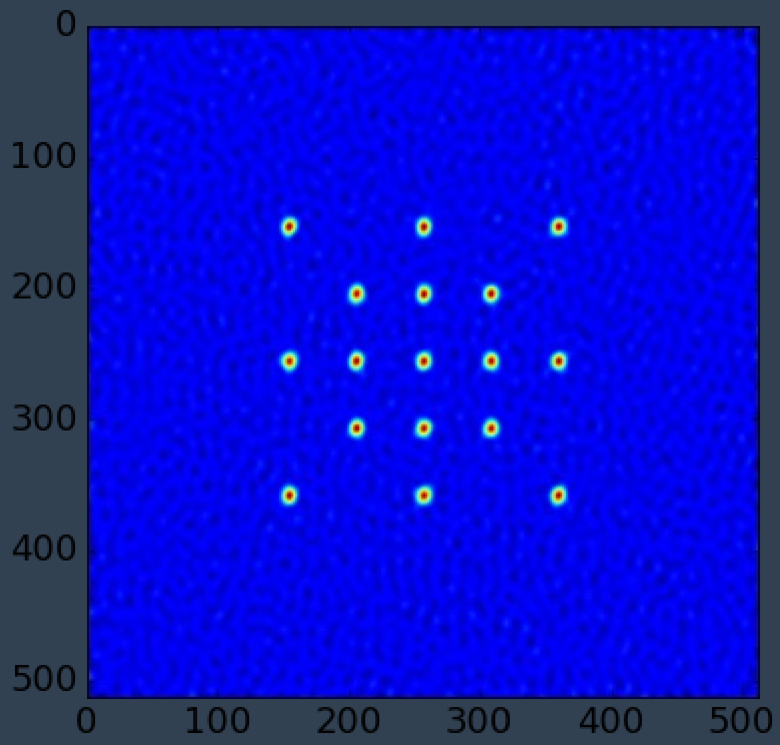


Pixel Flux Distribution (log)

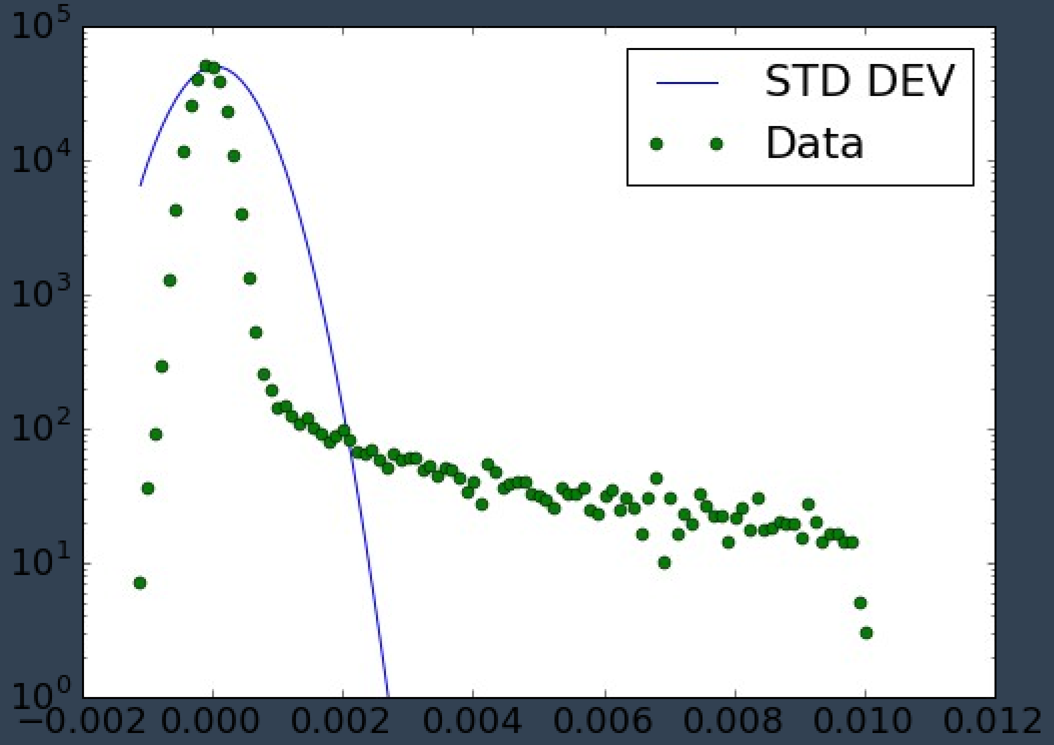


Source Finding

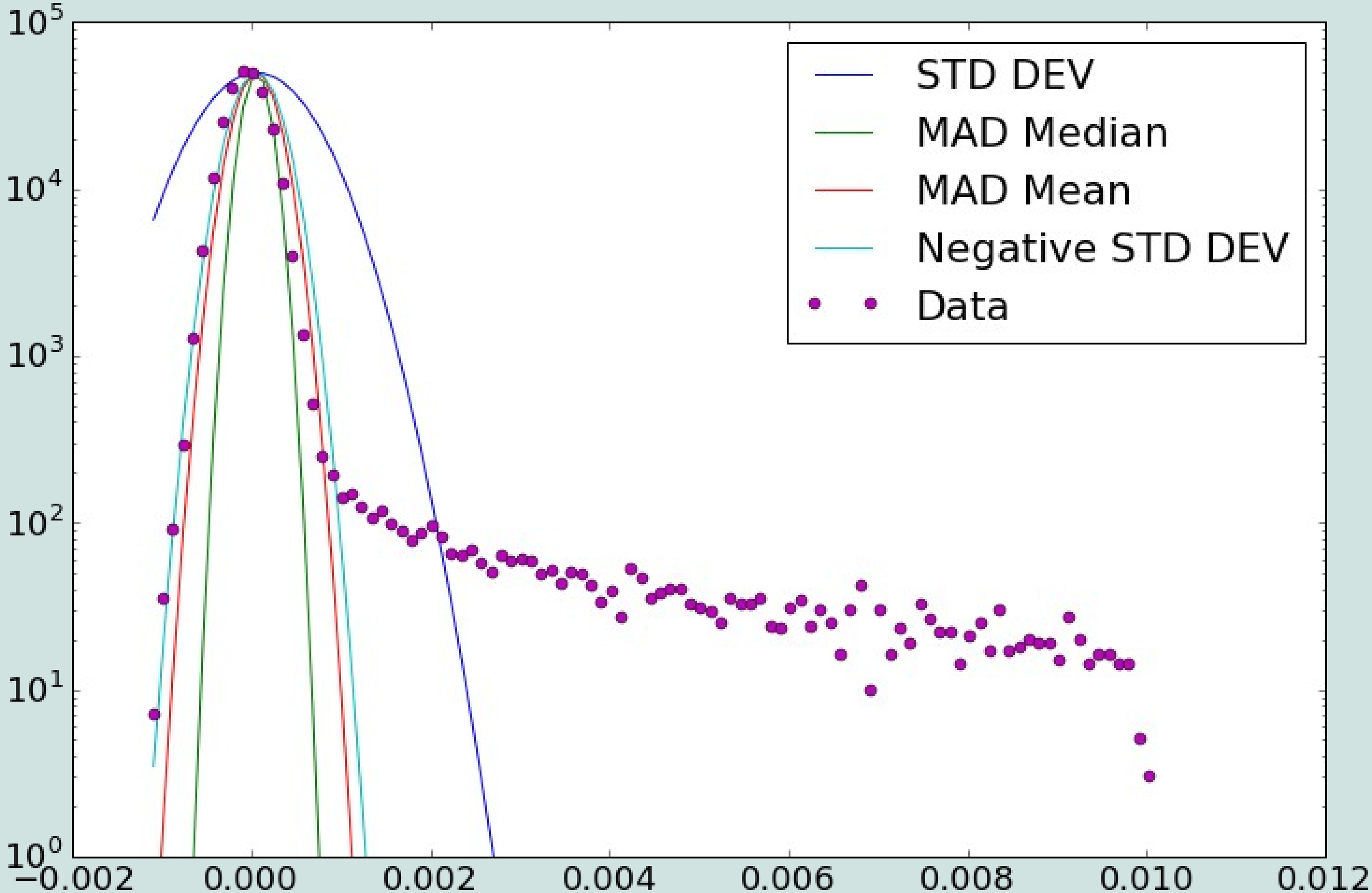
Noise w/ Sources



Pixel Flux Distribution (log)

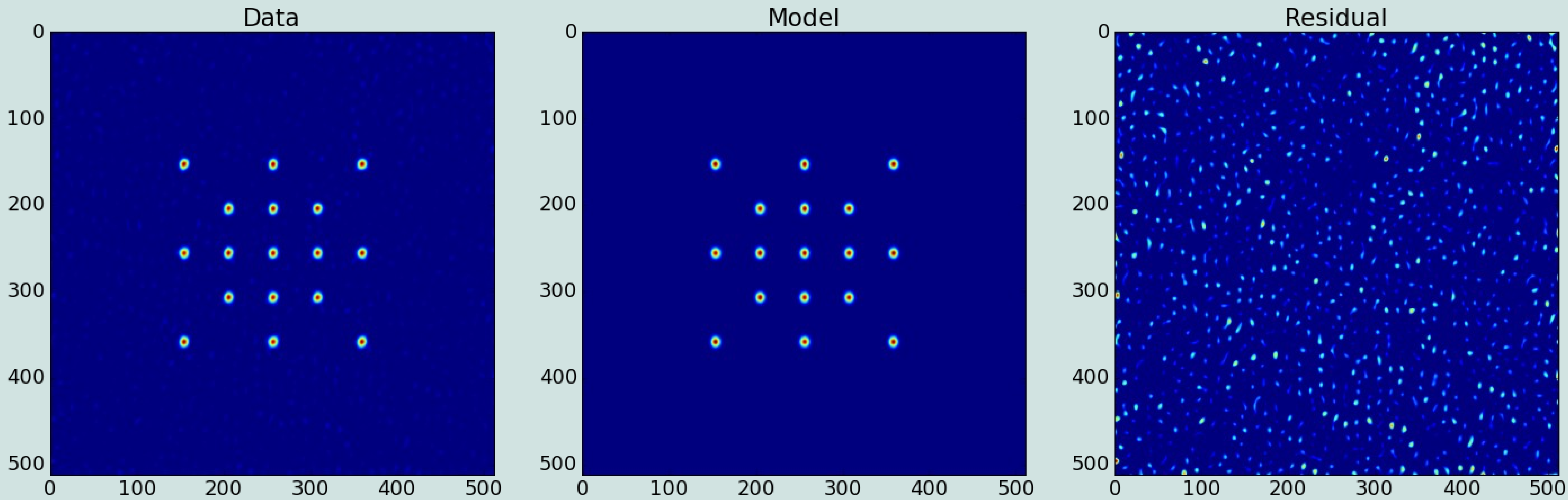


Source Finding



A Gaussian using the flux mean and standard deviation results in a poor model. Better noise models can be derived from computing the mean absolute deviation or only using the negative pixel values.

Source Finding



Peak_Flux	Pix_x	Pix_y	Size_x	Size_y
0.0103	153.5	255.8	7.85	9.80
0.0102	204.4	255.6	7.88	9.81
0.0102	306.7	204.4	8.02	9.59
0.0102	255.8	204.5	7.90	9.88
0.0101	204.3	306.8	8.30	9.48
0.0100	255.0	357.4	8.30	9.47

...

Next class, optional but suggected,
Thursday 1:00-3:00 in the computer lab.

Assignment 2: Implement Clark's
Method, see course site for link to
starting point notebook, due May 6