Continuum Analysis

ALMA Imaging Workshop 2026

Piyali Saha

Postdoctoral Fellow & EA-ALMA Taiwan ARC, ASIAA

September 2-4 , 2026

What do we want to learn?

Why ALMA continuum data are important?

How can we use ALMA continuum data?

+

Examples of usage of CASA tasks and CARTA

Outline

Brief description about:

Continuum flux density
Brightness temperature
Sensitivity
Polarization

CASA tasks:

Imhead
Imval
Imsubimage
Imstat
Imfit
….and so on

*Useful links:

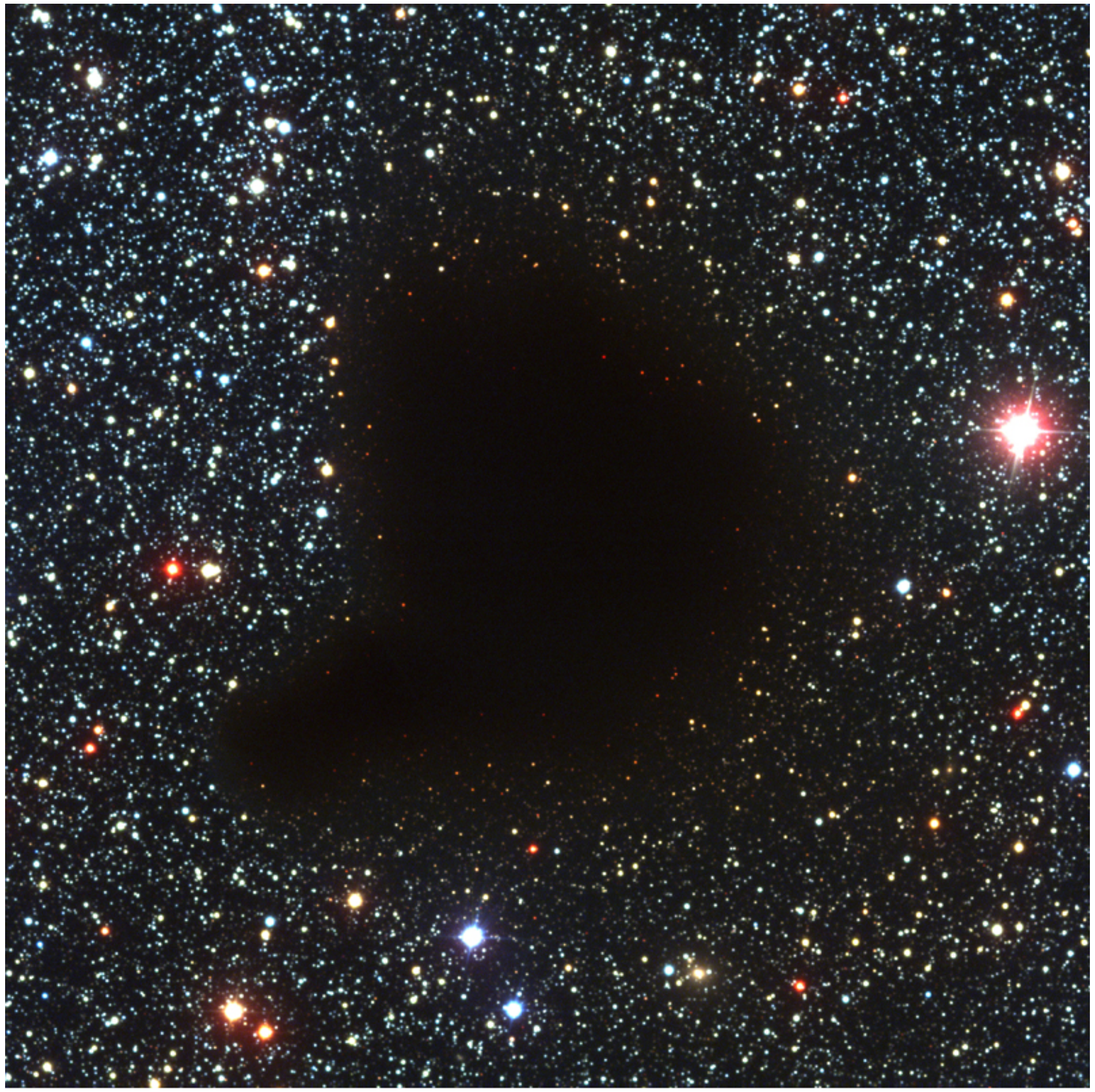
https://www2.nao.ac.jp/~eaarc/DATARED/EAARC_MAPSCI/eaarc_map2sci_diskcont_en.html
https://casaguides.nrao.edu/index.php/3C286_Polarization
https://almascience.eso.org/documents-and-tools/cycle13/alma-science-primer
https://almascience.org/tools/alma-primer-videos

Why ALMA continuum data are important?

Dark patches in the sky —-> star lights are blocked by dense cold interstellar clouds

Some examples of dark clouds in optical wavelength:

Horsehead nebula



Barnard 68

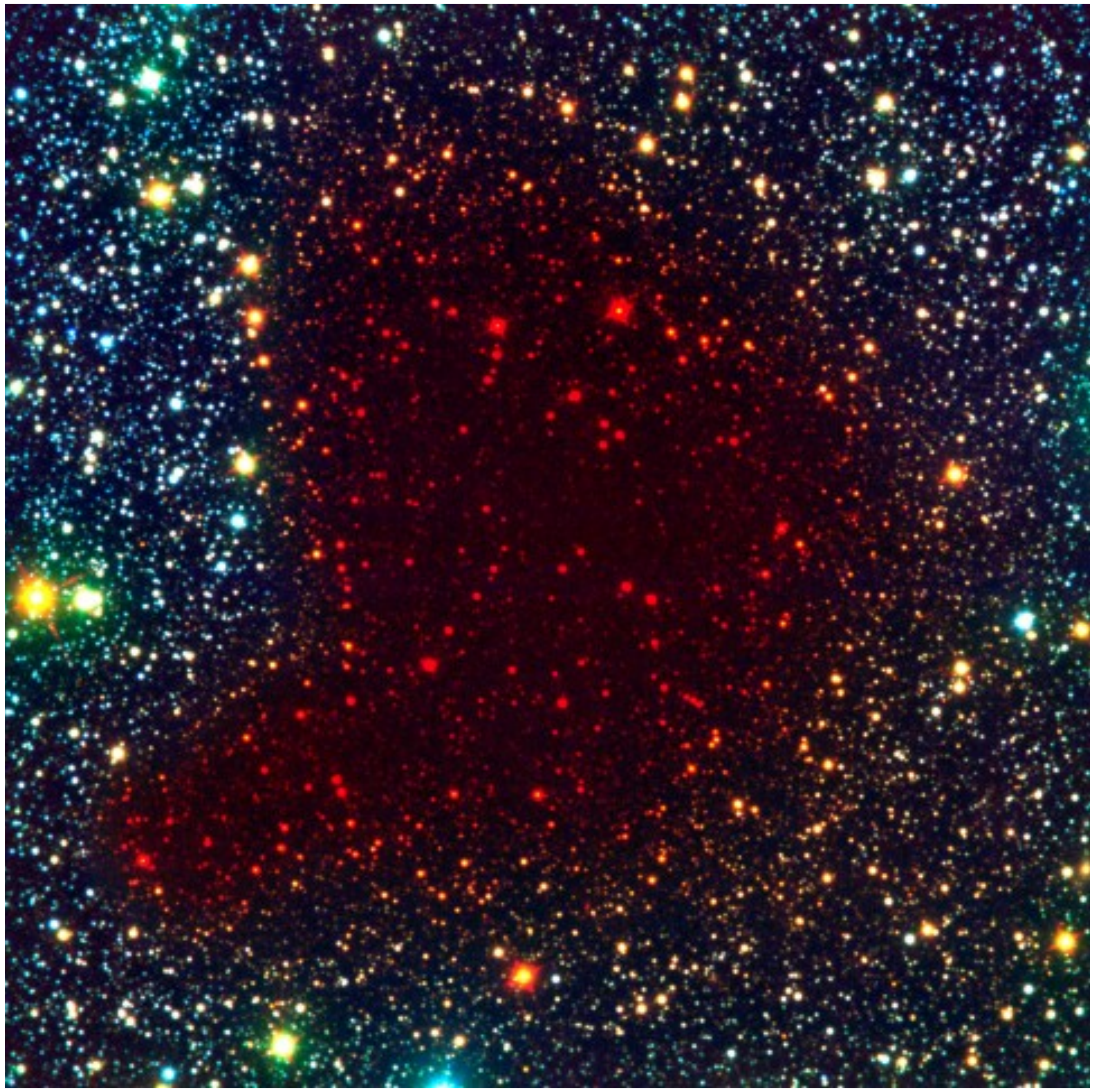
Image credit: ESO & NASA

Why ALMA continuum data are important?

Dark patches can be probed with longer wavelengths

Some examples of dark clouds in infrared wavelength:

Horsehead nebula



Barnard 68

Image credit: ESO & NASA

Why ALMA continuum data are important?

ALMA gives us access to the millimeter and submillimeter sky, where continuum emission can trace some of the coldest material in astronomical systems.

But before we can interpret that emission physically, we first need to understand what we are actually measuring.

What ALMA measures:

- continuum flux density
- spectral information

Flux density tells us how much emission we receive from the source at a given frequency.

In an ALMA observation, this is typically expressed in units of Janskys or milli Janskys.

$$1 \text{ Jansky (Jy)} = 10^{-26} \text{ W m}^{-2} \text{ Hz}^{-1} = 10^{-23} \text{ erg s}^{-1} \text{ cm}^{-2} \text{ Hz}^{-1}$$

Brightness temperature

It is the temperature of a source calculated on the assumption that it is a blackbody emitting radiation of the observed intensity at a given wavelength in the Rayleigh-Jeans limit ($h\nu/kT \ll 1$).

wavelength

$$T = \frac{\lambda^2}{2k\Omega} S$$

flux density

brightness temperature

Boltzmann constant

beam solid angle

The unit of brightness temperature is Kelvin (K).

Brightness temperature

Now, we can replace $\frac{S}{\Omega}$ by $\frac{S}{beam}\,\frac{beam}{\Omega} = I\,\frac{beam}{\Omega}$

beam solid angle $\Omega = \frac{\pi\,\theta_{maj}\,\theta_{min}}{4\;ln2}$

Substituting omega and then converting to units to cm, seconds of arc, and mJy/beam results in:

$$T = 1.36\,\frac{\lambda^2}{\theta_{maj}\,\theta_{min}}\;I$$

or in frequencies in units of GHz, and I in mJy/beam:

$$T = 1.222 \times 10^3\,\frac{I}{\nu^2\,\theta_{maj}\,\theta_{min}}$$

Sensitivity

Sensitivity is usually defined as the 1 sigma rms variation of noise in the data (ΔS) and so serves as a threshold for the detection of emission.

For ALMA, basic sensitivity depends on:

System noise

$$\Delta S \propto \frac{T_{sys}}{D^2 \left[n_p\, N(N-1)\, \Delta\nu\, \Delta t\right]^{1/2}} \ \mathrm{W\, m^{-2}\, Hz^{-1}}$$

Antenna diameter

Number of antennas

Bandwidth

Integration time

Number of polarization measurements

Sensitivity

Cycle 13 Receiver Bands				
Band	**Frequency (GHz)**	**Wavelength (mm)**	**Primary Beam (FOV; ")**	**Continuum Sensitivity (mJy/beam)**
1	*35-50*	*8.5-6*	*166-117*	*0.058*
*2**	*67-116*	*4.4-2.6*	*87-50*	*0.070*
3	*84-116*	*3.6-2.6*	*69-50*	*0.082*
4	*125-163*	*2.4-1.8*	*46-36*	*0.089*
5	*158-211*	*1.9-1.4*	*37-28*	*0.12*
6	*211-275*	*1.4-1.1*	*28-21*	*0.12*
7	*275-373*	*1.1-0.8*	*21-16*	*0.22*
8	*385-500*	*0.78-0.6*	*15-12*	*0.40*
9	*602-720*	*0.5-0.42*	*9.7-8.1*	*1.4*
10	*787-950*	*0.38-0.32*	*7.4-6.1*	*3.3*

Image credit: ALMA Science primer Cycle 13

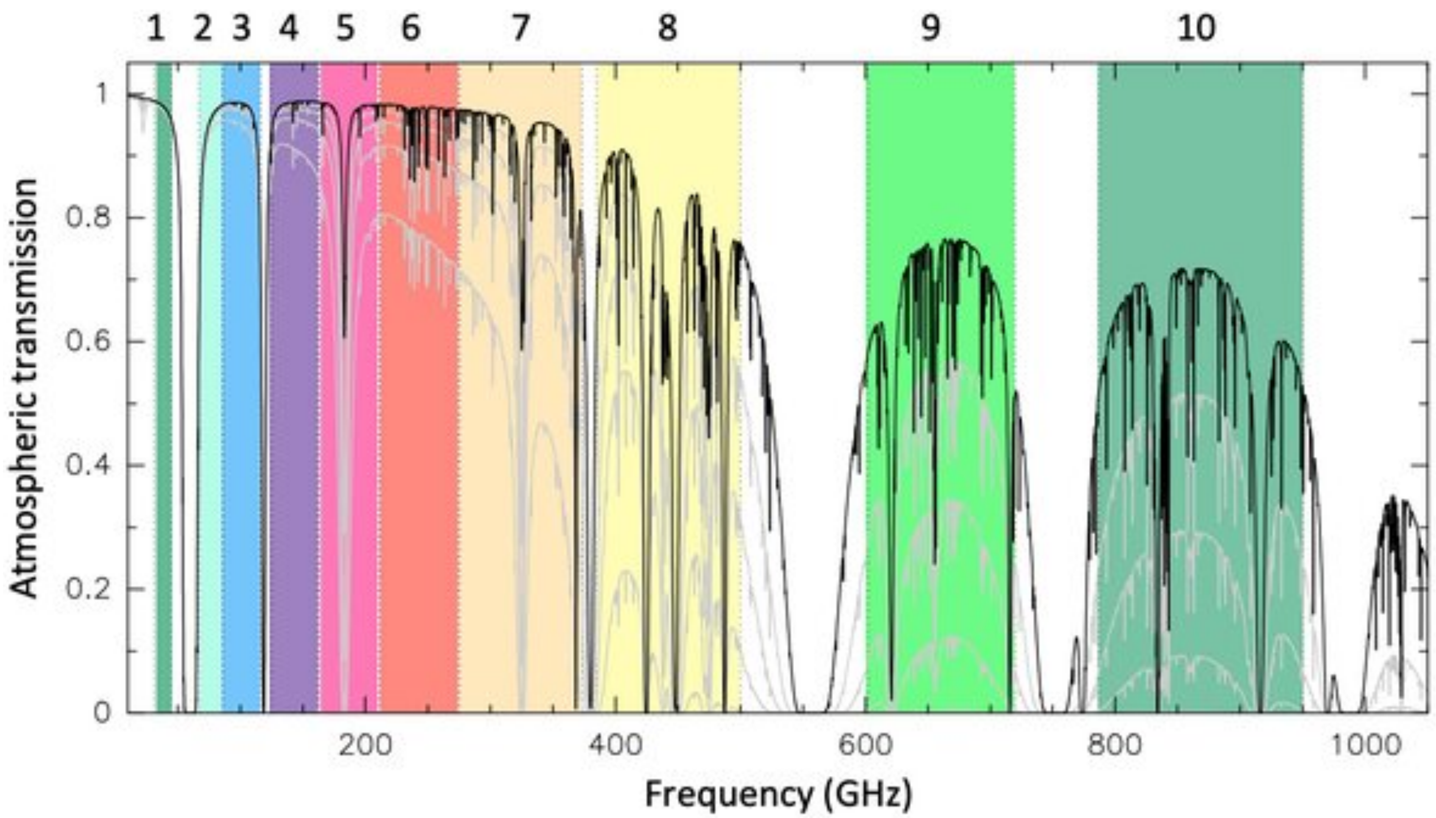


Image credit: Quadrature Hybrid Optimization for ALMA Bands 2 and 3 by Jorge Hernan Cardenas

Sensitivity worsens with increasing frequency

Sensitivity

In continuum analysis, another aspect of the observational set-up strongly affects sensitivity: **Beamsize**

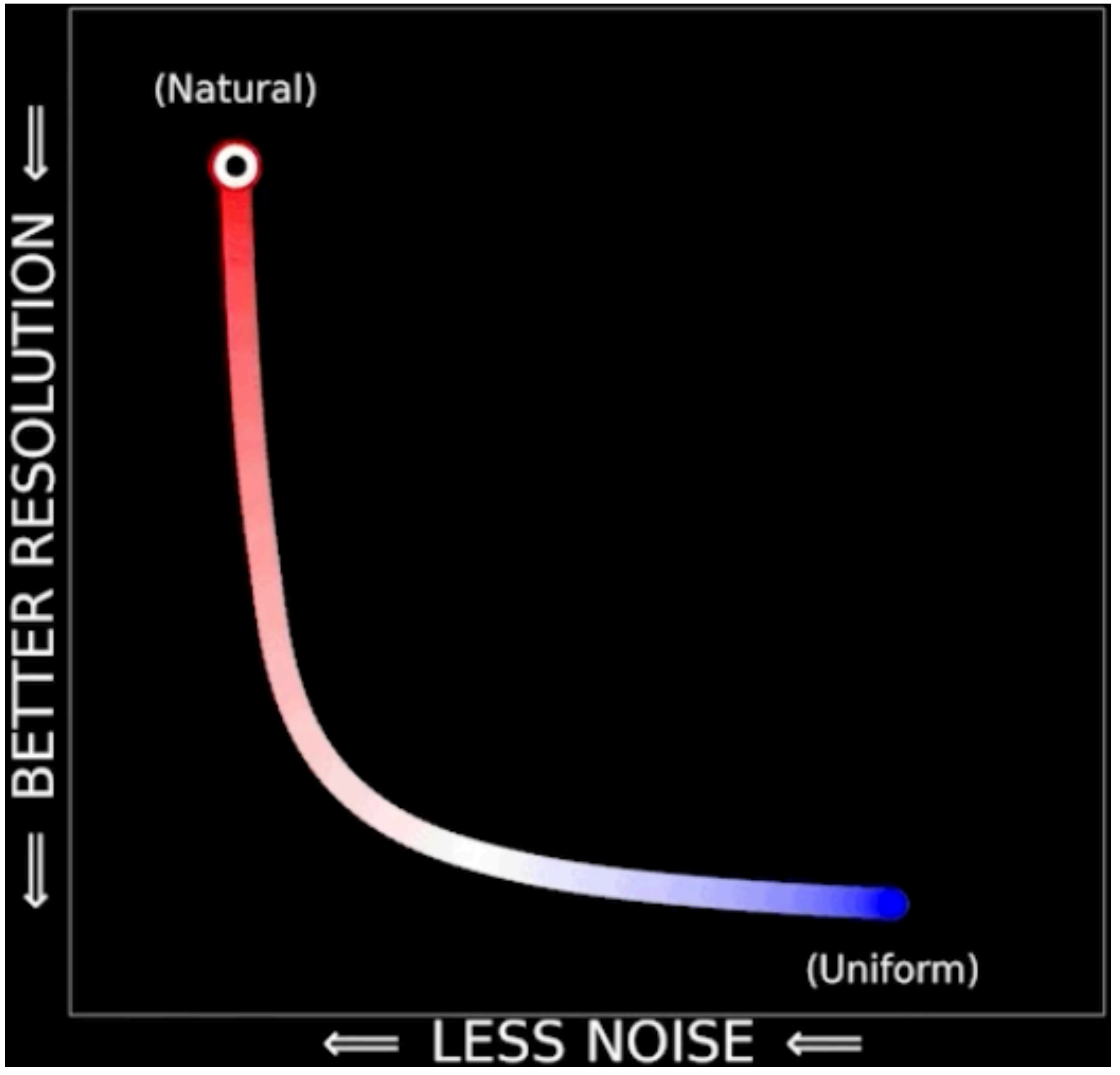


Image credit: YouTube video of ALMA Primer Series: Weighting in Imaging

Let's explore some CASA tasks with CARTA

Imhead

Provides information of the image header

```
CASA:imhead('twhya_cont.image')
Out:
{'axisnames': array(['Right Ascension', 'Declination', 'Stokes', 'Frequency'],
       dtype='<U15'),
 'axisunits': array(['rad', 'rad', '', 'Hz'], dtype='<U3'),
 'defaultmask': 'mask0',
 'hasmask': True,
 'imagetype': 'Intensity',
 'incr': array([-4.84813681e-07,  4.84813681e-07,  1.00000000e+00,
2.34445115e+08]),
 'masks': array(['mask0'], dtype='<U5'),
 'messages': array([], dtype='<U5'),
 'ndim': 4,
 'refpix': array([125., 125.,   0.,   0.]),
 'refval': array([ 2.88792330e+00, -6.05713443e-01,  1.00000000e+00,
3.72636940e+11]),
 'restoringbeam': {'major': {'unit': 'arcsec', 'value': 0.6092361807823181},
  'minor': {'unit': 'arcsec', 'value': 0.45245257019996643},
  'positionangle': {'unit': 'deg', 'value': -51.57859802246094}},
 'shape': array([250, 250,   1,   1]),
 'tileshape': array([250, 250,   1,   1]),
 'unit': 'Jy/beam'}
```

Imhead

```
CASA:imhead('twhya_cont.image', mode='list')
Out:
{'beammajor': {'unit': 'arcsec', 'value': 0.6092361807823181},
 'beamminor': {'unit': 'arcsec', 'value': 0.45245257019996643},
 'beampa': {'unit': 'deg', 'value': -51.57859802246094},
 'bunit': 'Jy/beam',
 'cdelt1': -4.84813681109536e-07,
 'cdelt2': 4.84813681109536e-07,
 'cdelt3': 1.0,
 'cdelt4': 234445114.8776245,
 'crpix1': 125.0,
 'crpix2': 125.0,
 'crpix3': 0.0,
 'crpix4': 0.0,
 'crval1': 2.887923300334642,
 'crval2': -0.6057134431858937,
 'crval3': 1.0,
 'crval4': 372636939995.9968,
 'ctype1': 'Right Ascension',
 'ctype2': 'Declination',
 'ctype3': 'Stokes',
 'ctype4': 'Frequency',
 'cunit1': 'rad',
 'cunit2': 'rad',
 'cunit3': '',
 'cunit4': 'Hz',
 'datamax': 0.39978447556495667,
 'datamin': -0.008692984469234943,
 'date-obs': '2012/11/19/07:56:26.544000',
 'equinox': 'J2000',
 'imtype': 'Intensity',
 'masks': array(['mask0'], dtype='<U5'),
 'maxpixpos': array([120, 127,   0,   0]),
 'maxpos': '11:01:51.837 -34.42.17.166 I 3.72637e+11Hz',
 'minpixpos': array([ 85, 191,   0,   0]),
 'minpos': '11:01:52.120 -34.42.10.766 I 3.72637e+11Hz',
 'object': 'TW Hya',
 'observer': 'cqi',
 'projection': 'SIN',
 'reffreqtype': 'LSRK',
 'restfreq': array([3.7263694e+11]),
 'shape': array([250, 250,   1,   1]),
 'telescope': 'ALMA'}
```

Imhead

Extracting an hdkey from the image header

```
CASA: imhead('twhya_cont.image', mode='get', hdkey='beammajor')
Out: {'unit': 'arcsec', 'value': 0.609236180782318l}

CASA: imhead('twhya_cont.image', mode='get', hdkey='beamminor')
Out: {'unit': 'arcsec', 'value': 0.45245257019996643}

CASA: imhead('twhya_cont.image', mode='get', hdkey='restfreq')
Out: {'unit': 'Hz', 'value': 372636939995.9968}
```

Adding an hdkey in the image header

```
CASA: imhead('twhya_cont.image', mode='put', hdkey='dataanalyzer',
hdvalue='piyalisaha')

CASA: imhead('twhya_cont.image', mode='get', hdkey='dataanalyzer')
Out: 'piyalisaha'
```

Exercise

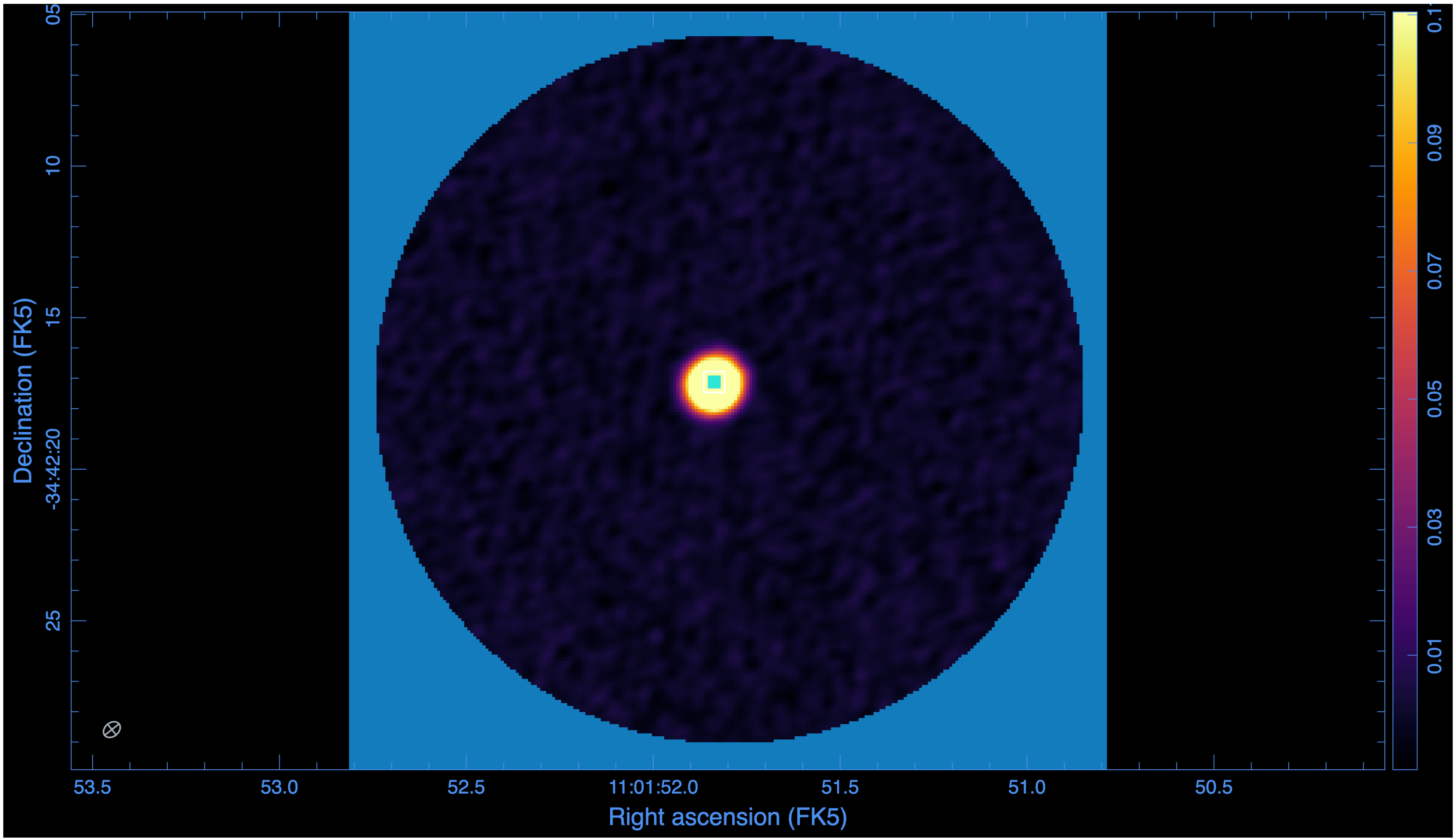
You have 6 sources with the following names:

HH212M
HOPS-11
HOPS-60
HOPS-81
HOPS-84
HOPS-224

Try to get the beamsizes using imhead

Imval

The **imval** task extracts the values of the data from a specified region of an image



Imval

The **imval** task will extract the values of the data from a specified region of an image

```
CASA: val = imval('twhya_cont.image', box='120,127')

CASA: print(val)
{'blc': [120, 127, 0, 0], 'trc': [120, 127, 0, 0], 'unit': 'Jy/beam', 'data':
array([0.39978448]), 'mask': array([ True]), 'coords':
array([[ 2.88792625e+00, -6.05712474e-01,  1.00000000e+00,
         3.72636940e+11]]), 'axes': [[0, 'Right Ascension'], [1,
'Declination'], [3, 'Frequency'], [2, 'Stokes']]}

CASA: print(val['data'])
[0.39978448]
```

Exercise

You have 6 sources with the following names:

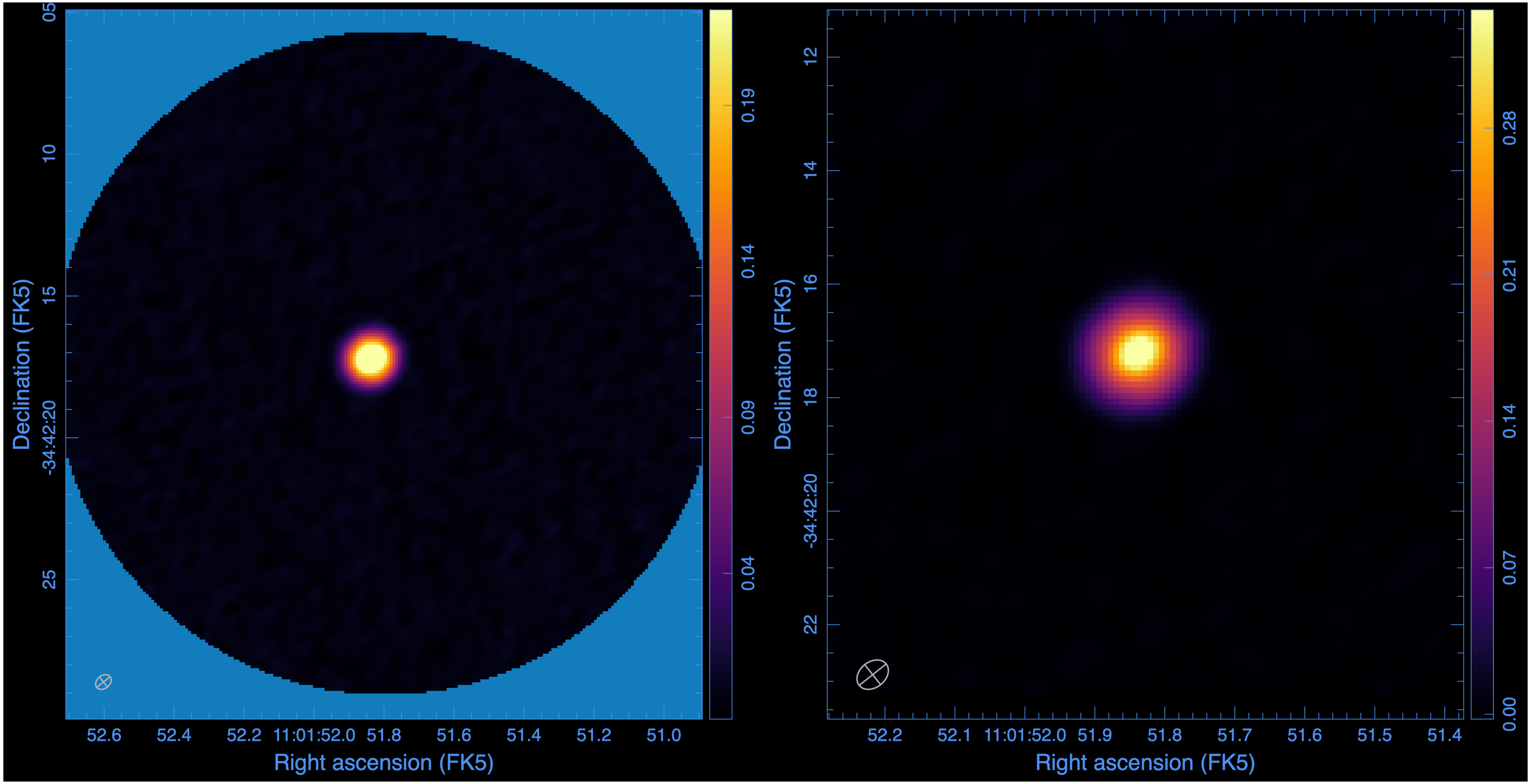
HH212M
HOPS-11
HOPS-60
HOPS-81
HOPS-84
HOPS-224

Try to get the pixel value at the centre of each image

Imsubimage

To create an image by specifying a region of an image

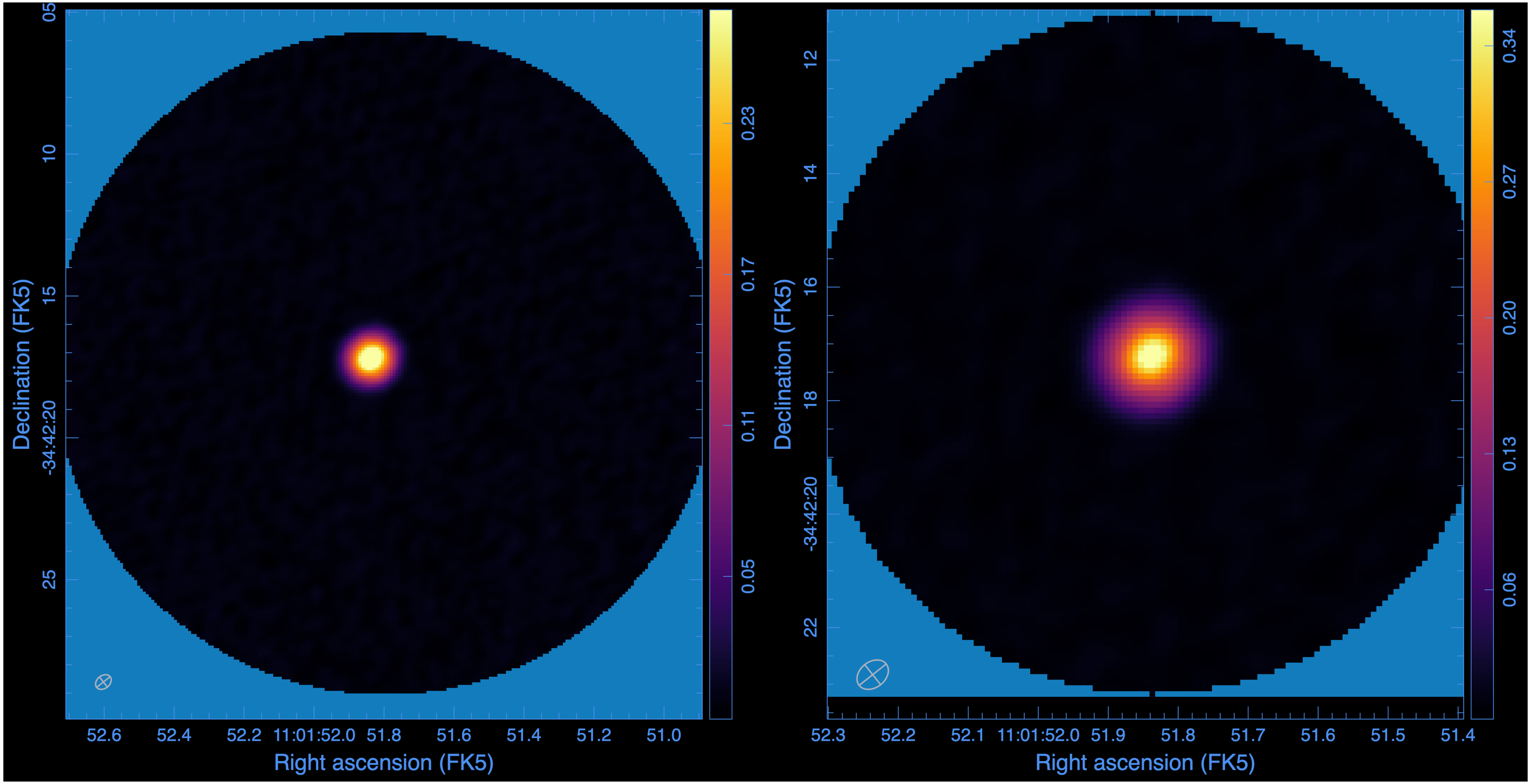
```
region='box[[60pix, 67pix], [180pix, 187pix]]'

imsubimage(imagename="twhya_cont.im", region = region,
outfile="cropped_twhya_cont.im")
```



Imsubimage

To create an image by specifying a region of an image

```
region='circle[[120pix, 127pix], 60pix]'

imsubimage(imagename="twhya_cont.im", region = region,
outfile="cropped_twhya_cont.im")
```



Imsubimage

To drop unnecessary axis of an image

```
CASA: imsubimage(imagename='HH212M.IQUV.image',outfile='HH212M.I.image',
stokes='I', dropdeg=True)
```

A degenerate axis is an image axis with only one pixel/channel.

CASA Region Format

Rectangular box; the two coordinates are two opposite corners:
`box[[x1, y1], [x2, y2]]`

Center box; [x, y] define the center point of the box and [x_width, y_width] the width of the sides:
`centerbox[[x, y], [x_width, y_width]]`

Rotated box; [x, y] define the center point of the box; [x_width, y_width] the width of the sides; rotang the rotation angle:
`rotbox[[x, y], [x_width, y_width], rotang]`

Polygon; there could be many [x, y] corners. If parts of the polygon overlap, then the pixels in that overlapping region are taken into account. Note that the last point will connect with the first point to close the polygon:
`poly[[x1, y1], [x2, y2], [x3, y3], …]`

Circle; center of the circle [x,y], r is the radius:
`circle[[x, y], r]`

Annulus; center of the circle is [x, y], [r1, r2] are inner and outer radii:
`annulus[[x, y], [r1, r2]]`

Ellipse; center of the ellipse is [x, y]; semi-major and semi-minor axes are [bmaj, bmin]; position angle of the major axis is pa:
`ellipse[[x, y], [bmaj, bmin], pa]`

Imstat

Calculates statistical information from an image or image region

```
CASA:stat=imstat(imagename='twhya_cont.image')
CASA:stat
Out:
{'blc': array([0, 0, 0, 0]),
 'blcf': '11:01:52.810, -34.42.29.866, I, 3.72637e+11Hz',
 'flux': array([1.98294114]),
 'max': array([0.39978448]),
 'maxpos': array([120, 127,   0,   0]),
 'maxposf': '11:01:51.837, -34.42.17.166, I, 3.72637e+11Hz',
 'mean': array([0.00144245]),
 'medabsdevmed': array([0.00136992]),
 'median': array([3.91167487e-05]),
 'min': array([-0.00869298]),
 'minpos': array([ 85, 191,   0,   0]),
 'minposf': '11:01:52.120, -34.42.10.766, I, 3.72637e+11Hz',
 'npts': array([42937.]),
 'q1': array([-0.00133103]),
 'q3': array([0.00140875]),
 'quartile': array([0.00273978]),
 'rms': array([0.01665415]),
 'sigma': array([0.01659175]),
 'sum': array([61.93455052]),
 'sumsq': array([11.90902983]),
 'trc': array([249, 249,   0,   0]),
 'trcf': '11:01:50.790, -34.42.04.966, I, 3.72637e+11Hz'}
```

Imstat

Calculates statistical information from an image or image region

flux	the flux or flux density. See below for details.
npts	the number of unmasked points used
max	the maximum pixel value
min	minimum pixel value
maxpos	absolute PIXEL coordinate of maximum pixel value
maxposf	Same as maxpos, but uses WORLD coordinates instead of pixels
minpos	absolute pixel coordinate of minimum pixel value
minposf	Same as minpos, but uses WORLD coordinates instead of pixels
sum	the sum of the pixel values:
sumsq	the sum of the squares of the pixel values:
mean	the mean of pixel values :
sigma	the standard deviation about the mean:
rms	the root mean square :
median	the median pixel value

Imstat

Calculates statistical information from an image or image region

```
CASA:stat["max"]
Out: array([0.39978448])

CASA:stat["maxpos"]
Out: array([120, 127,   0,   0])

CASA:stat["maxposf"]
Out: '11:01:51.837, -34.42.17.166, I, 3.72637e+11Hz'
```

Imstat can be used to measure the sensitivity, dynamic range, flux etc.

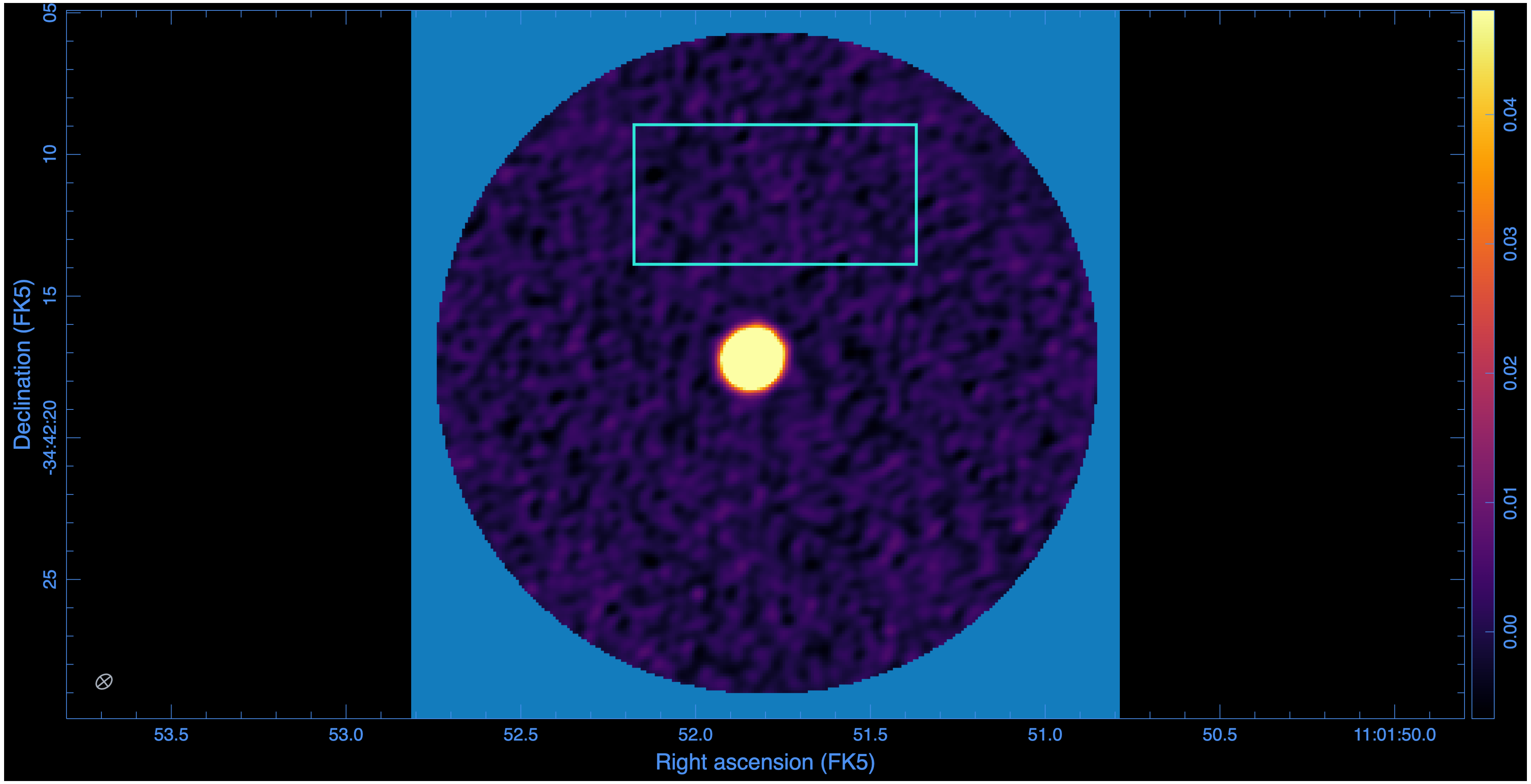
Sensitivity: rms of a box (ideally a relatively large box with no emission that is representative of the noise in the image)

Dynamic Range: ratio of the brightest signal and noise

Imstat

To estimate sensitivity of an image:

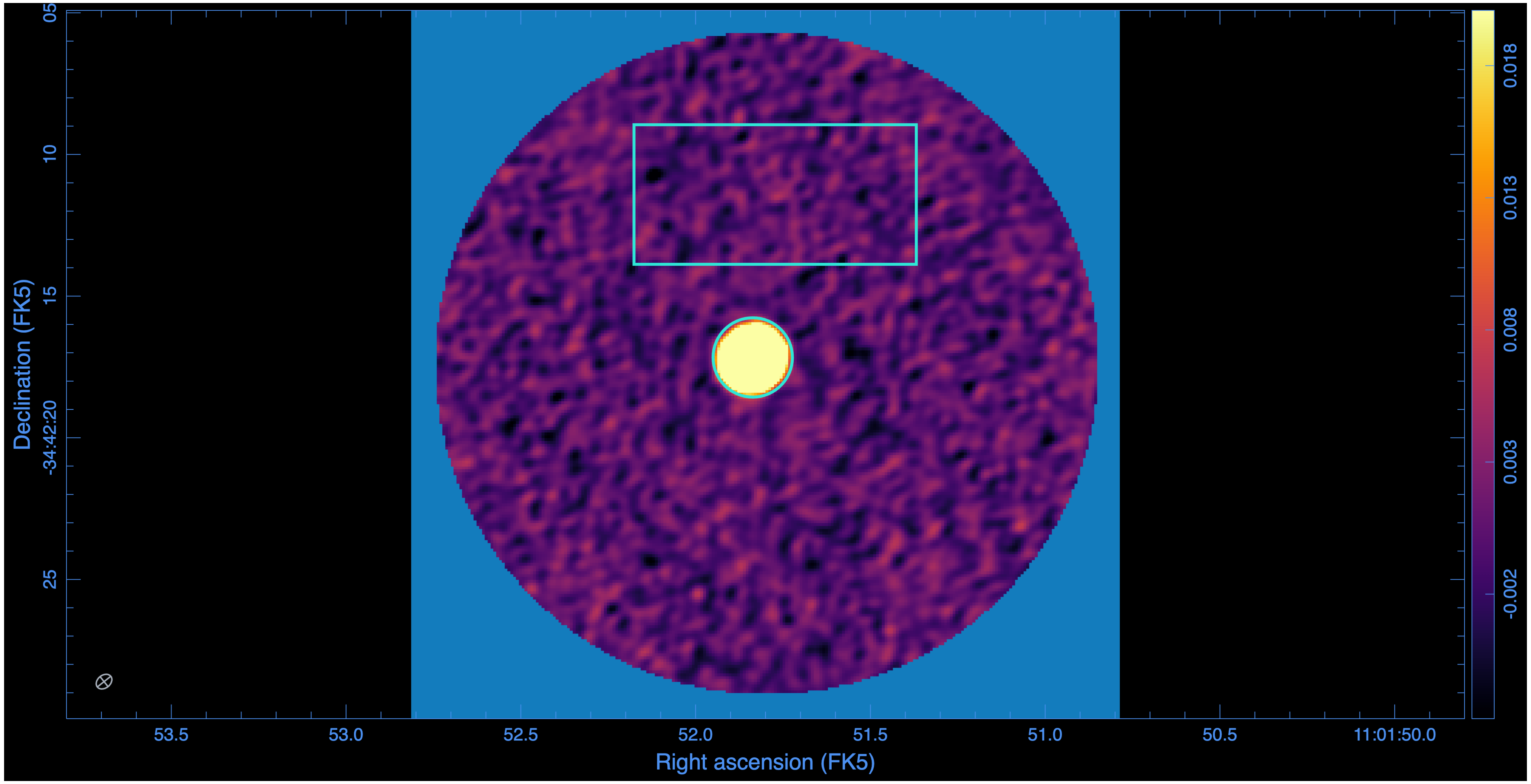
```
CASA:stat=imstat(imagename='twhya_cont.image', box='80,160,180,210')

CASA: stat["rms"]
Out: array([0.00201315])
```



Imstat

To estimate flux of a region:

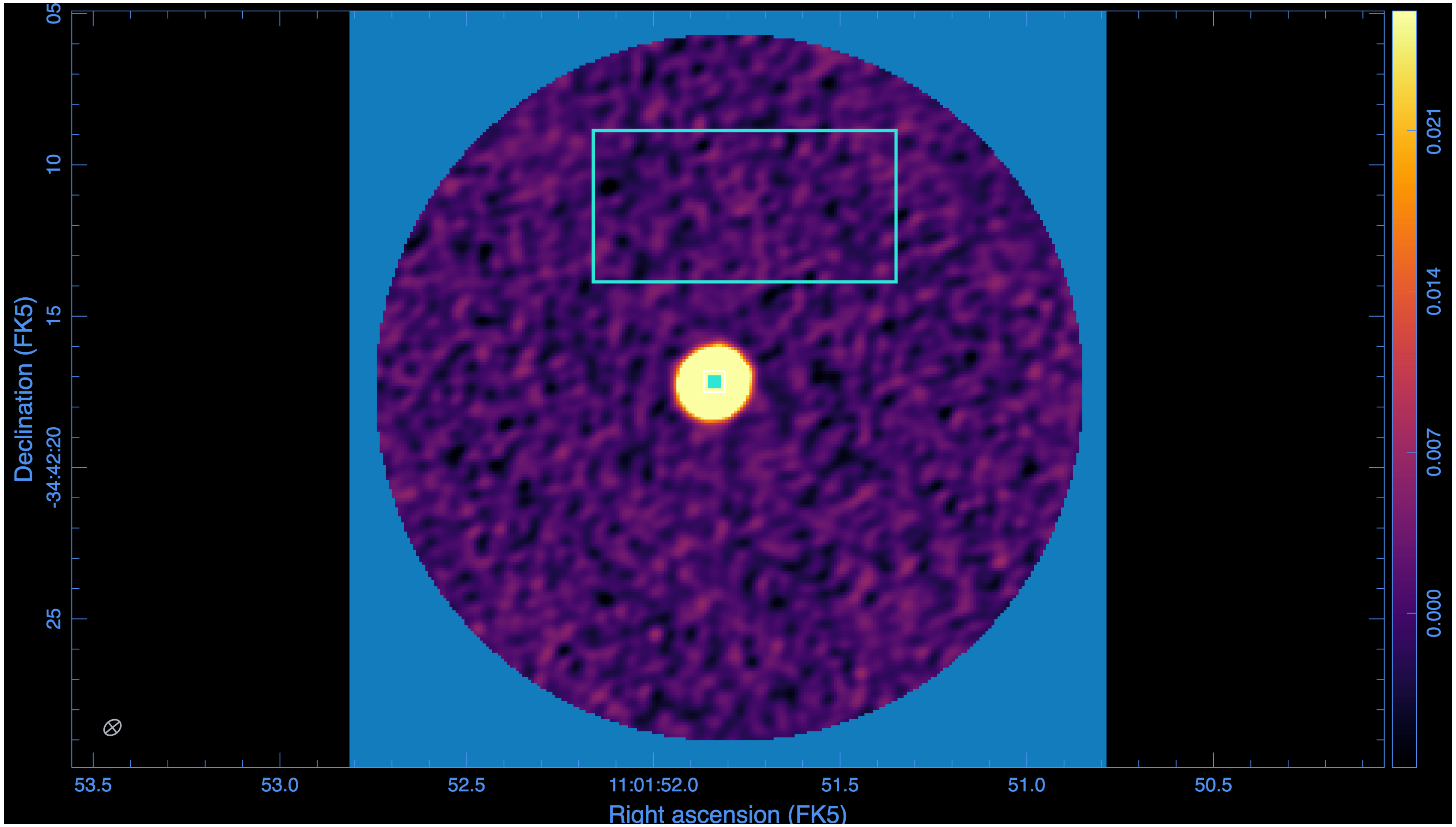
```
region2='circle[[120pix, 127pix], 14pix]'

CASA: stat=imstat(imagename='twhya_cont.image', region=region2)

CASA: stat["flux"]
Out: array([1.99139562])
```



Imstat

To estimate the dynamic range of an image:

```
Dynamic range =  0.39978448/0.00201315 ~ 200
```



Exercise

You have 6 sources with the following names:

HH212M
HOPS-11
HOPS-60
HOPS-81
HOPS-84
HOPS-224

Try to get the sensitivity and dynamic range in each target.

Imfit

```
imfit(imagename='twhya_cont.image',
box="100,107,140,147",
logfile="Gaussian_fit.log",
residual="Gaussian_fit.residual",
model="Gaussian_fit.model")
```

Region included in fitting

Name of log file (recommended), otherwise no record will be saved

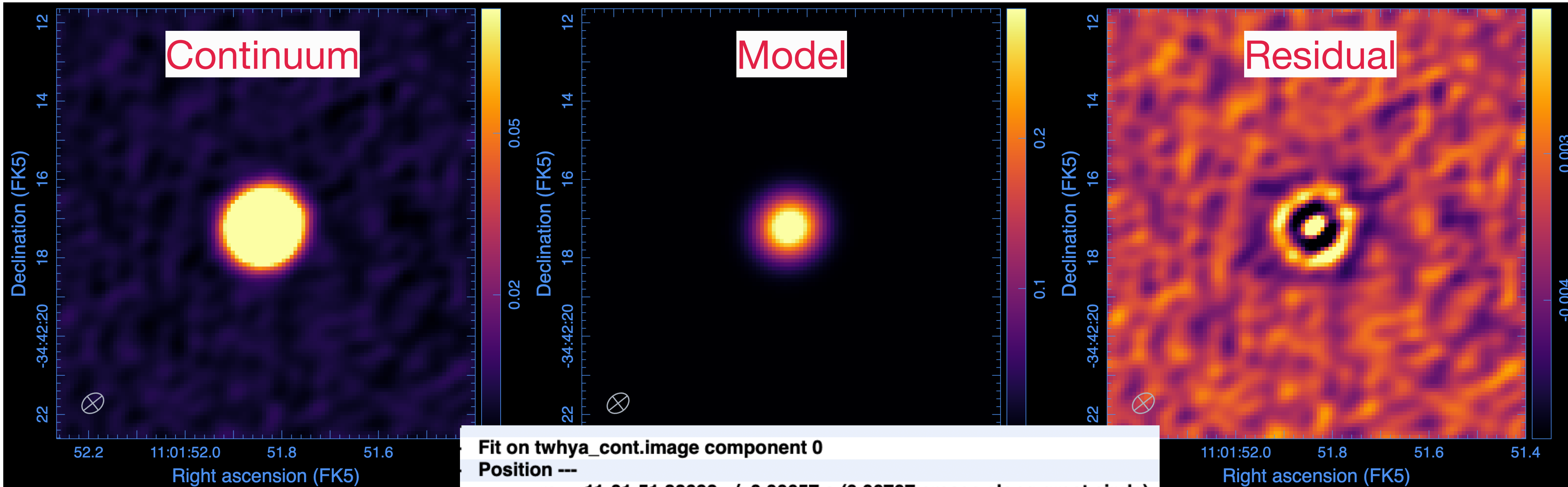
residual image

model image

Imfit provides:

- (x,y) position
- (major, minor) axis widths
- peak brightness
- position angle

Imfit



Fit on twhya_cont.image component 0
Position ---
--- ra: 11:01:51.83693 +/- 0.00057 s (0.00707 arcsec along great circle)
--- dec: -034.42.17.20754 +/- 0.00697 arcsec
--- ra: 119.953 +/- 0.071 pixels
--- dec: 126.585 +/- 0.070 pixels

Image component size (convolved with beam) ---
--- major axis FWHM: 1.281 +/- 0.017 arcsec
--- minor axis FWHM: 1.214 +/- 0.016 arcsec
--- position angle: 129.5 +/- 9.9 deg

Clean beam size ---
--- major axis FWHM: 0.61 arcsec
--- minor axis FWHM: 0.45 arcsec
--- position angle: -51.58 deg
Image component size (deconvolved from beam) ---
--- major axis FWHM: 1.129 +/- 0.031 arcsec
--- minor axis FWHM: 1.126 +/- 0.031 arcsec
--- position angle: 167 +/- 90 deg

Flux ---
--- Integrated: 2.056 +/- 0.032 Jy
--- Peak: 364.3 +/- 4.8 mJy/beam
--- Polarization: I

Exercise

You have 6 sources with the following names:

HH212M
HOPS-11
HOPS-60
HOPS-81
HOPS-84
HOPS-224

Try imfit using a gaussian fitting in these sources

Immath

To subtract the second image given from the first:

```
CASA: immath(imagename=[‘image1.im’,'image2.im'],expr='IM0-
IM1', outfile='ImageDiff.im')
```

To divide an image by another, with a threshold on one of the images:

```
CASA: immath(imagename=[‘image1.im','image2.im'], expr=‘IM0/
IM1[IM1>0.008]', outfile=‘imagediv.im')
```

Exercise

You have 6 sources with the following names:

HH212M
HOPS-11
HOPS-60
HOPS-81
HOPS-84
HOPS-224

Try to get the brightness temperature of any of the source using immath

Immath

CASA command to convert intensity to brightness temperature:

```
CASA: immath(imagename='HH212M.I.image', mode='evalexpr',
expr='1.222e6*IM0/(341.04^2*0.92*0.68)', outfile=‘HH212M.I.Tb.image')
```

Stokes I (Jy/beam)

Frequency (GHz)

Beam major (Arcsec)

Beam minor (Arcsec)

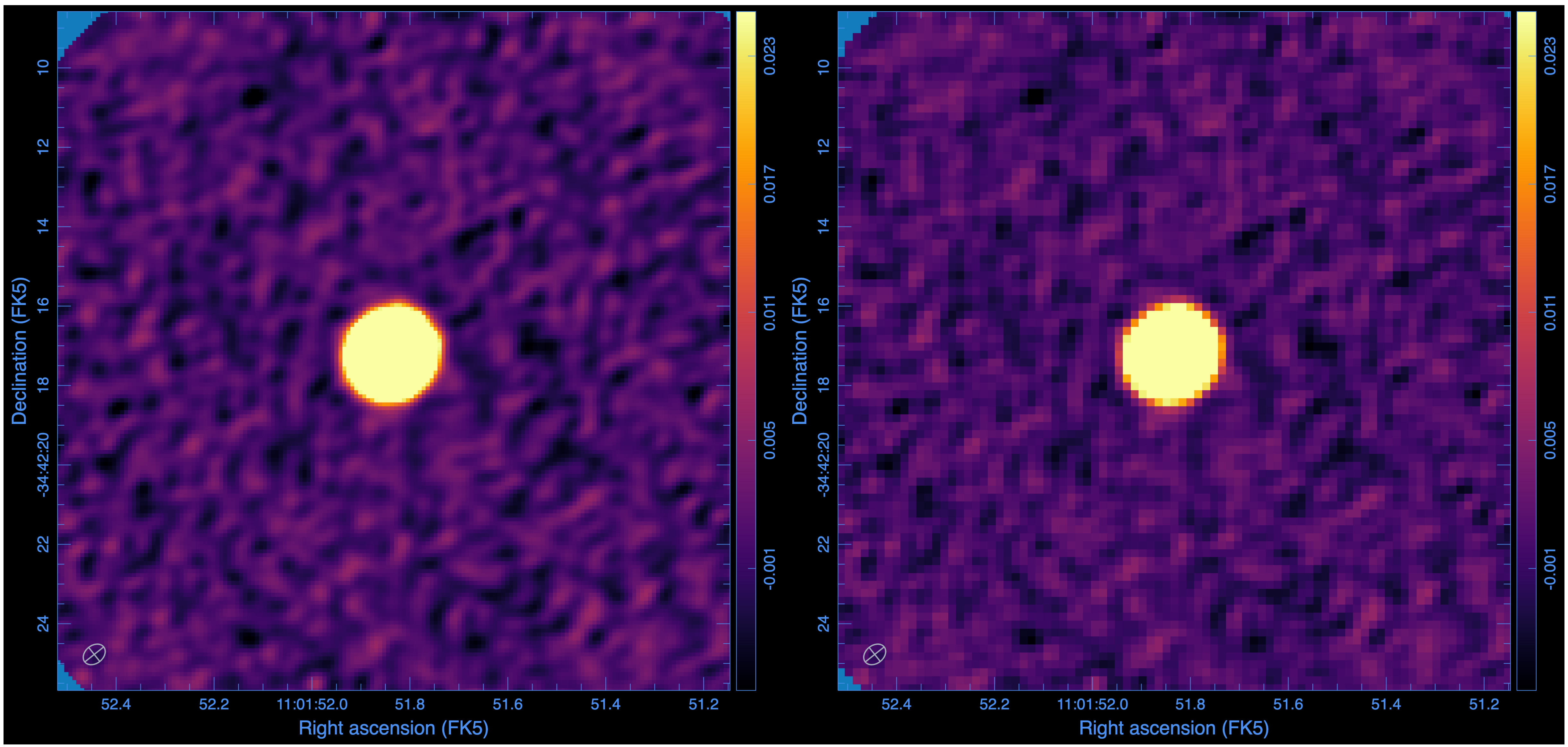
```
CASA: imhead(imagename='HH212M.I.Tb.image', mode='put',
hdkey='bunit', hdvalue='K')
```

Brightness unit

Imrebin

To rebin the first two axes (normally the direction axes) :

```
CASA:imrebin(imagename="twhya_cont.image",outfile="rebinned.image", factor=[5,5])
```



Exercise

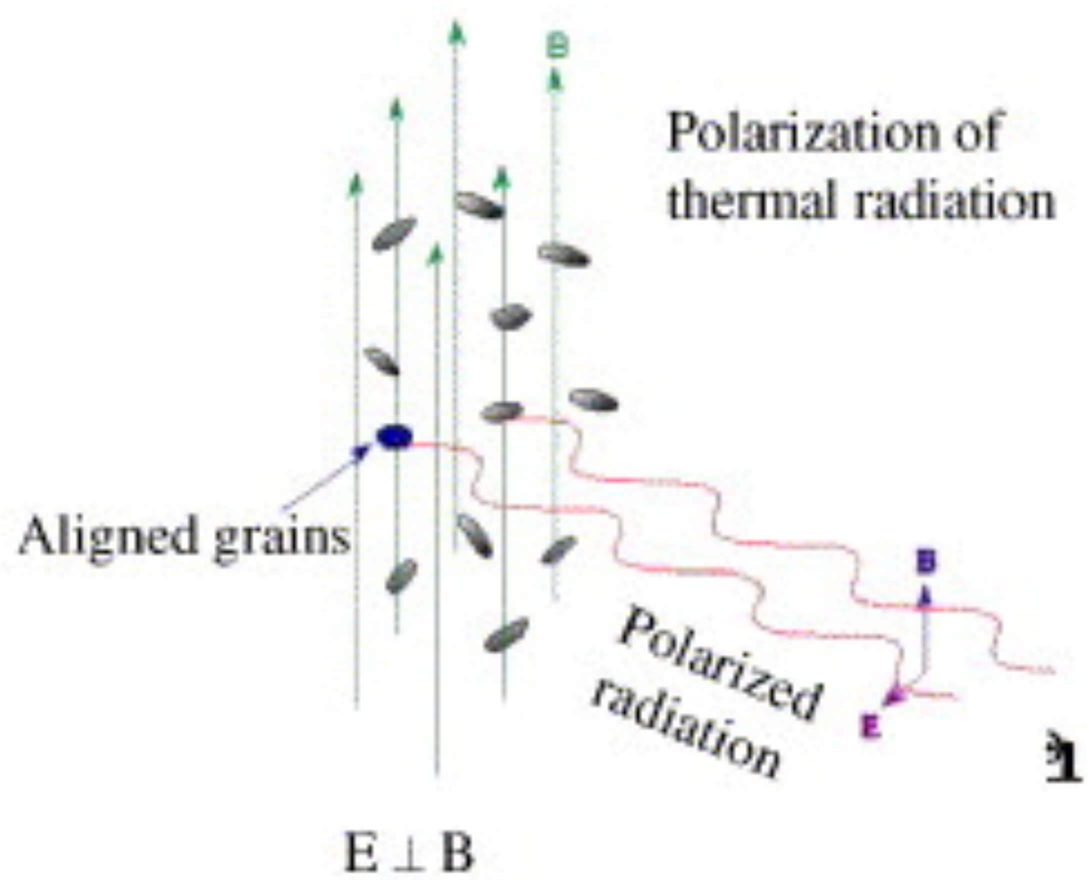
You have 6 sources with the following names:

HH212M
HOPS-11
HOPS-60
HOPS-81
HOPS-84
HOPS-224

Try imrebin using a factor of 4 in these sources, and check the rms using imstat

Polarization

ALMA polarization observations can provide information about B-fields, dust grain properties etc.



Interstellar dust grains are aligned with their minor axis parallel to the B-field direction

Credit: A. Lazarian

Polarization

Interferometric polarization observations:

- single mode: XX → High spectral resolution observations, mainly line studies
- dual mode: XX and YY → Continuum Stokes I only
- full polarization mode: XX, YY, YX, XY → Stokes I, Q, U, V

$$I = \frac{1}{2}(XX + YY)$$

total intensity
(polarized + unpolarized)

$$Q = \frac{1}{2}(XX - YY)$$

$$U = \frac{1}{2}(XY + YX)$$

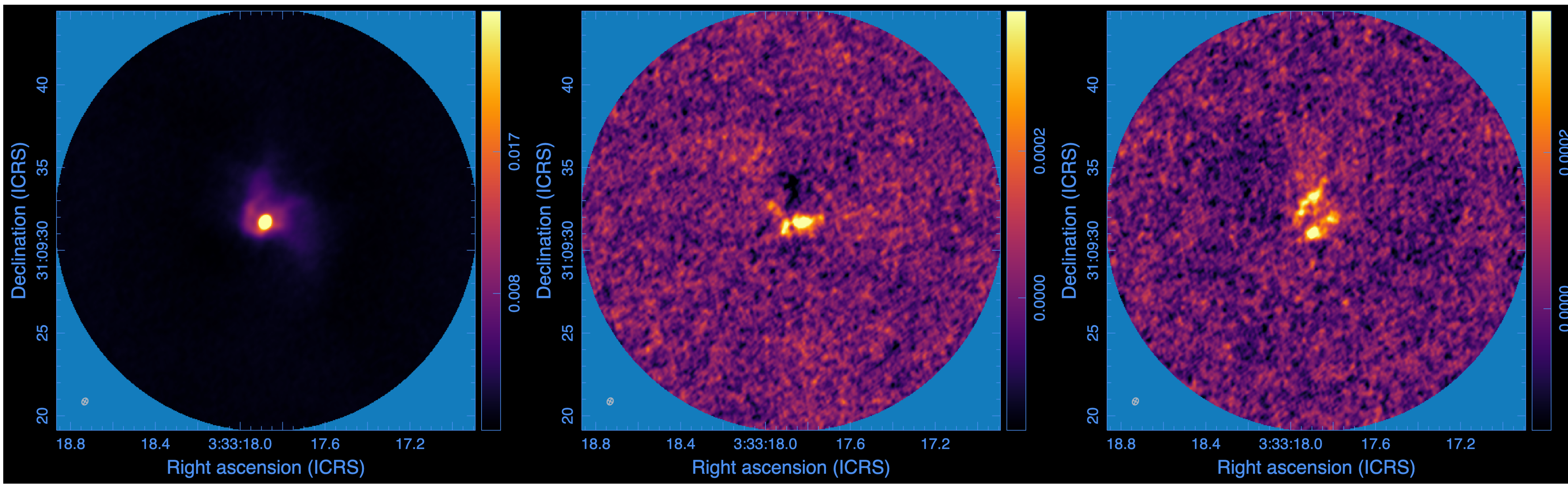
linear polarization

$$V = \frac{1}{2}(XY - YX)$$

circular polarization

Examples of Stokes Images

Stokes I Stokes Q Stokes U

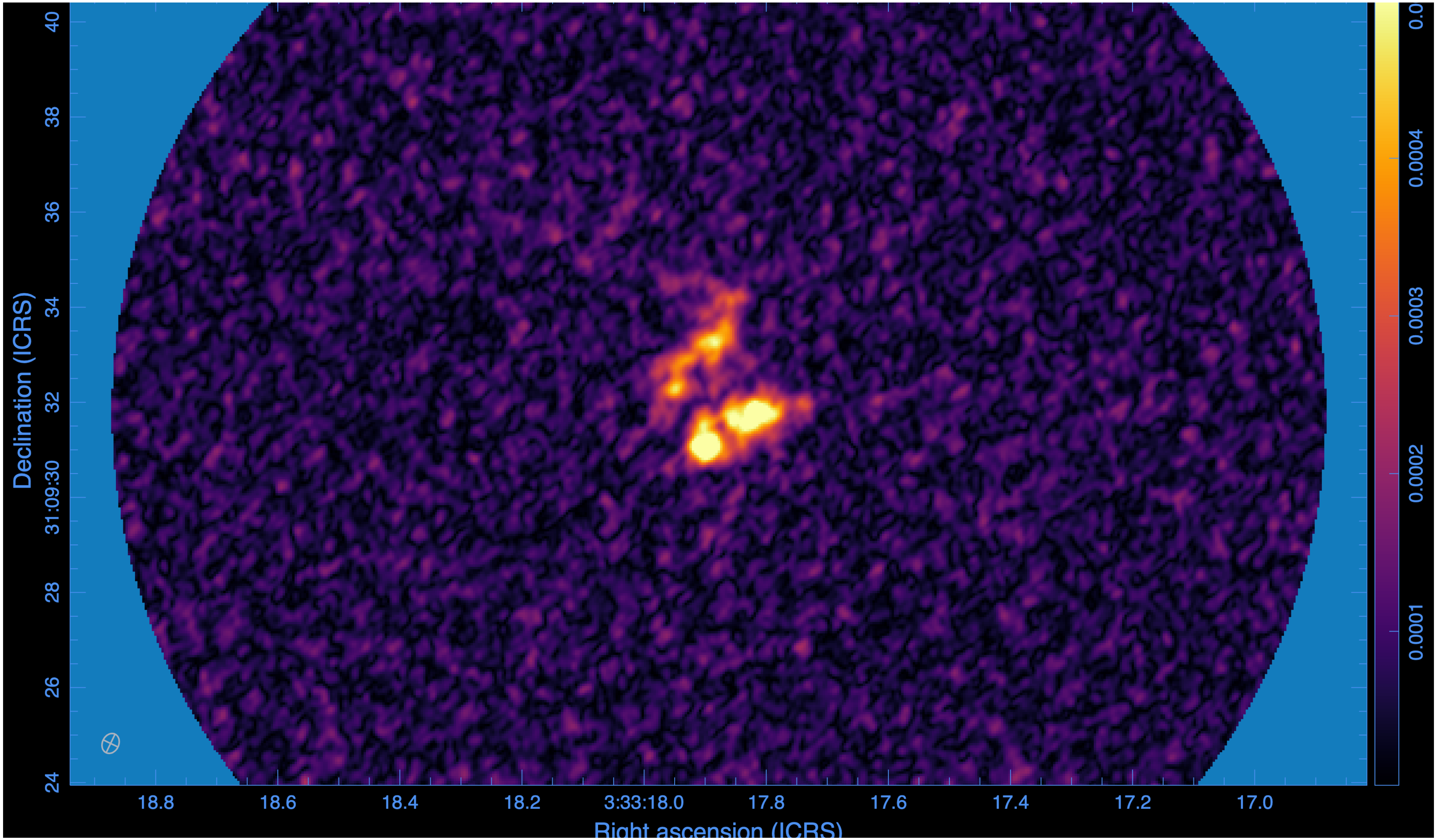


Polarized intensity image

$$PI = \sqrt{Q^2 + U^2}$$

```
CASA: immath(outfile = 'B1-c.POLI.image', mode = 'poli', imagename = ['B1-
c.StokesQ.image','B1-c.StokesU.image'])
Out: True

CASA: calstat = imstat(imagename='B1-c.POLI.image', box='240,190,450,260')

CASA: rms = (calstat['rms'][0])

CASA: print(rms)
7.350747881825672e-05
```

Polarized intensity image



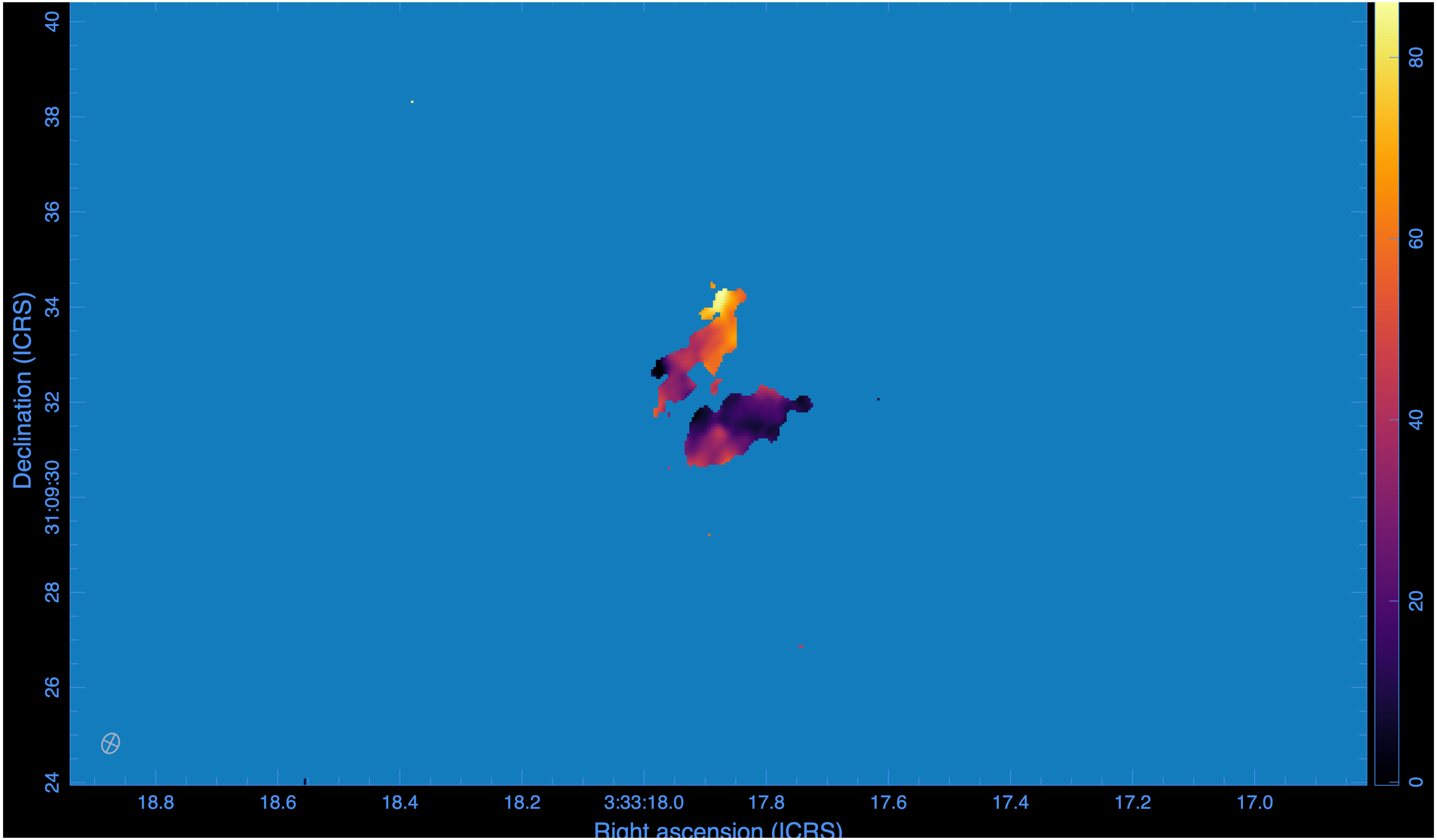
Polarization position angle image

$$\tan 2\chi = \frac{U}{Q}$$

```
CASA: immath(outfile = 'B1-c.POLA.image', mode = 'pola',imagename = ['B1-c.StokesQ.image', 'B1-c.StokesU.image'], polithresh = '0.2205mJy/beam')

Out: True
```

3 times RMS of
Polarized intensity map

Polarization position angle image



Polarization fraction image

$$P_{\mathrm{L}} = \sqrt{Q^2 + U^2}/I$$

```
CASA: immath(outfile='B1-c.POLF.image' ,mode='evalexpr',imagename=['B1-c.StokesI.image', 'B1-c.POLI.image'],expr='(IM1/IM0[IM0>0.000118])')

Out: True
```

B1-c.POLI.image

B1-c.StokesI.image

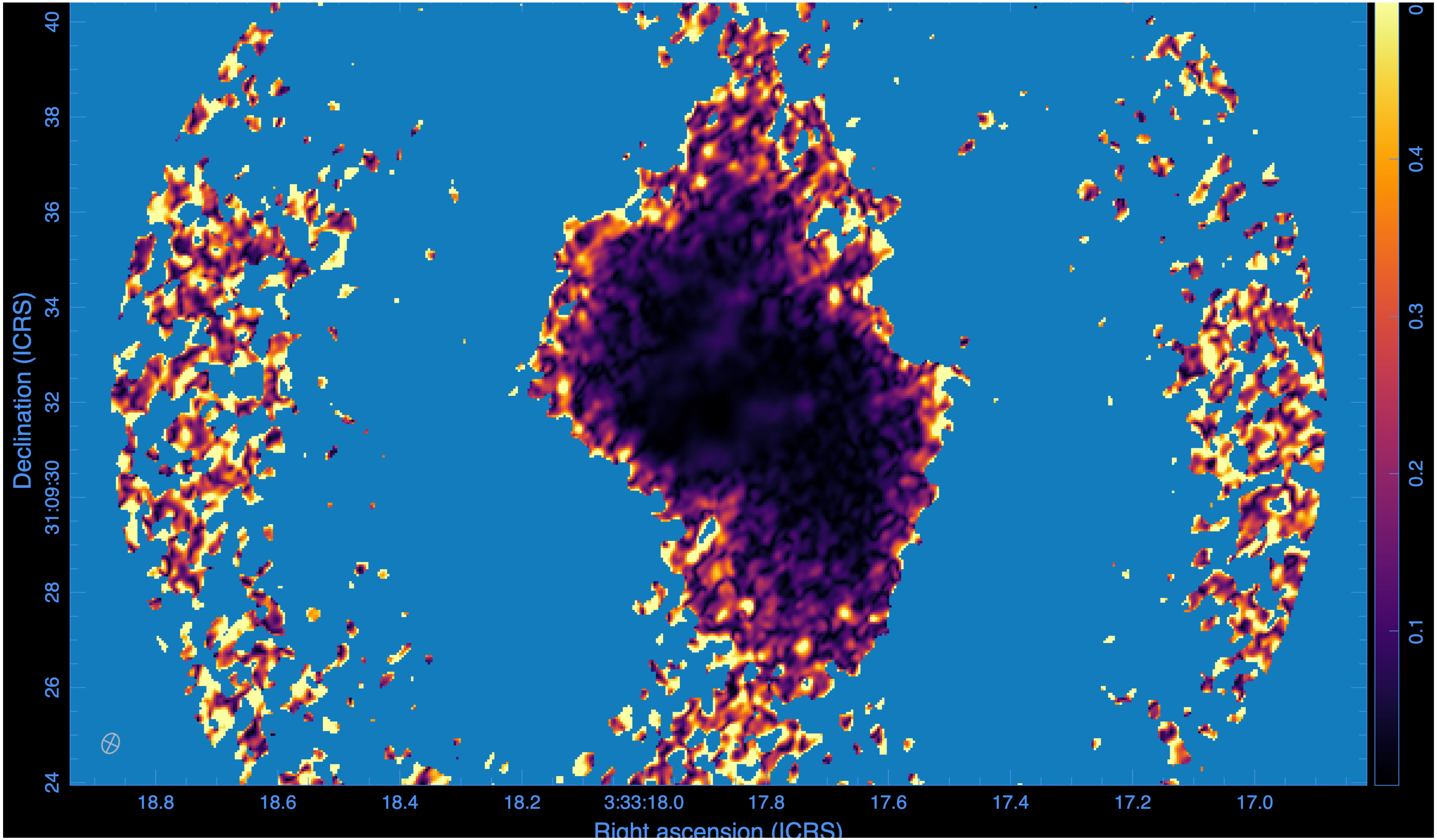
Polarization fraction image

$$P_{\mathrm{L}} = \sqrt{Q^2 + U^2}/I$$

```
CASA: immath(outfile='B1-c.POLF.image' ,mode='evalexpr',imagename=['B1-
c.StokesI.image', 'B1-c.POLI.image'],expr='(IM1/IM0[IM0>0.000118])')

Out: True
```

3 times RMS of
Stokes I map in Jy/beam

Polarization fraction image



B-field angle image

B-field angle = Polarization position angle + 90

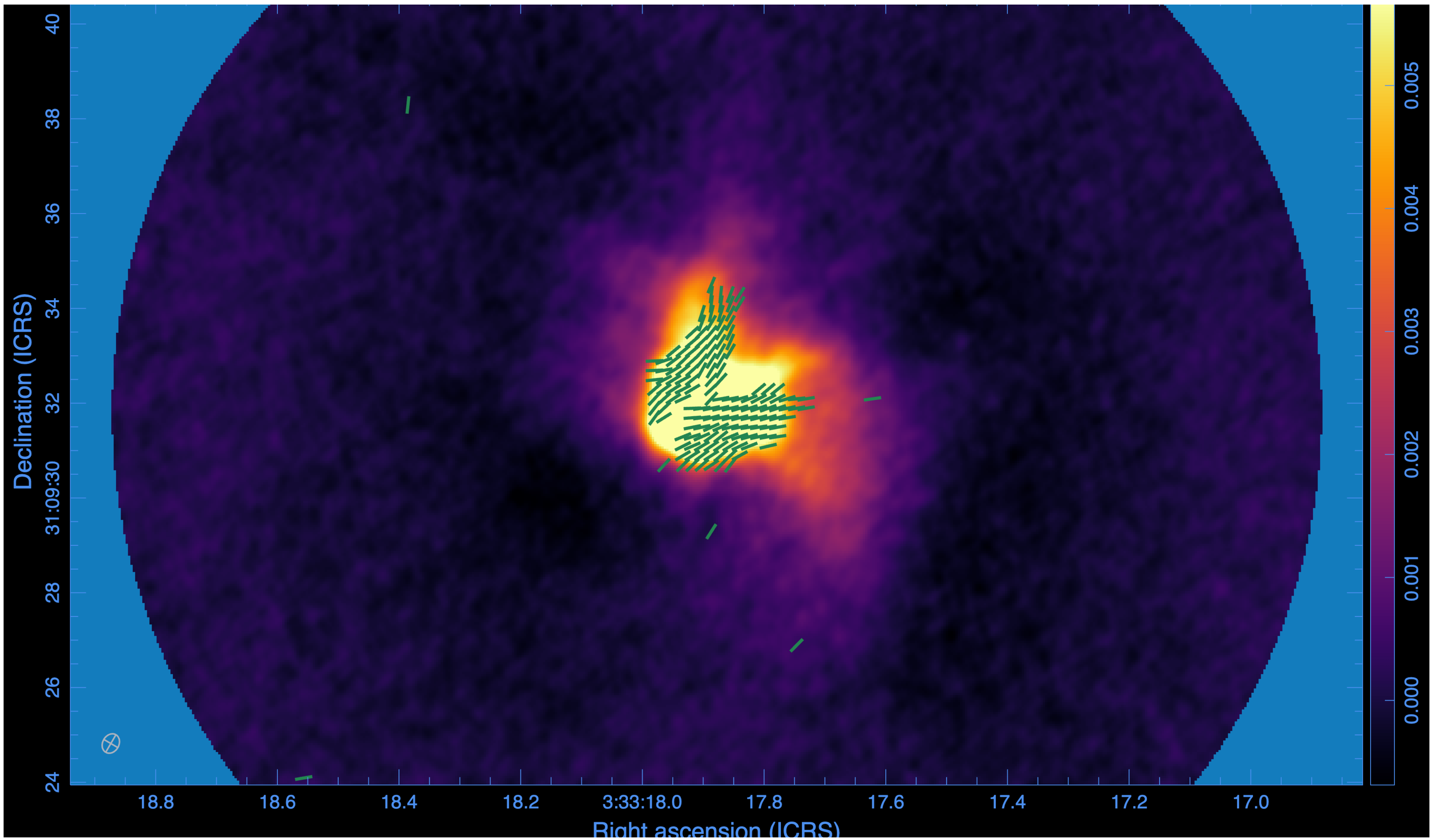
```
CASA: immath(outfile = 'B1-c.Bfield.image', imagename = 'B1-c.POLA.image',
mode='evalexpr', expr='IM0 + 90')

Out: True
```

B1-c.POLA.image

Assumption that B-field angle is perpendicular to the polarization position angle

B-field angle image

B-field angle = Polarization position angle + 90



Importfits and Exportfits

To import a FITS image into a CASA image format

```
CASA:importfits(fitsimage=‘twhya_cont.fits',imagename='tw
hya_cont.im', overwrite=True)
```

To export images from CASA image format to FITS images

```
CASA:exportfits(imagename=‘twhya_cont.image',fitsimage=‘twhy
a_cont.fits',overwrite=True)
```

For the cube we want to specify additionally that the frequency axis will be written out as velocity.

```
CASA:exportfits(imagename='twhya_n2hp.image',fitsimage='tw
hya_n2hp.fits',velocity=True,overwrite=True)
```

Exercise

You have I, Q, U images of a target named G333.46-0.16

Try to get the polarized intensity, polarization fraction and polarization position angle map of this target.

Then try to get the B-field map as well.

Exercise

B-field map should look similar to the black line segments presented in the following figure

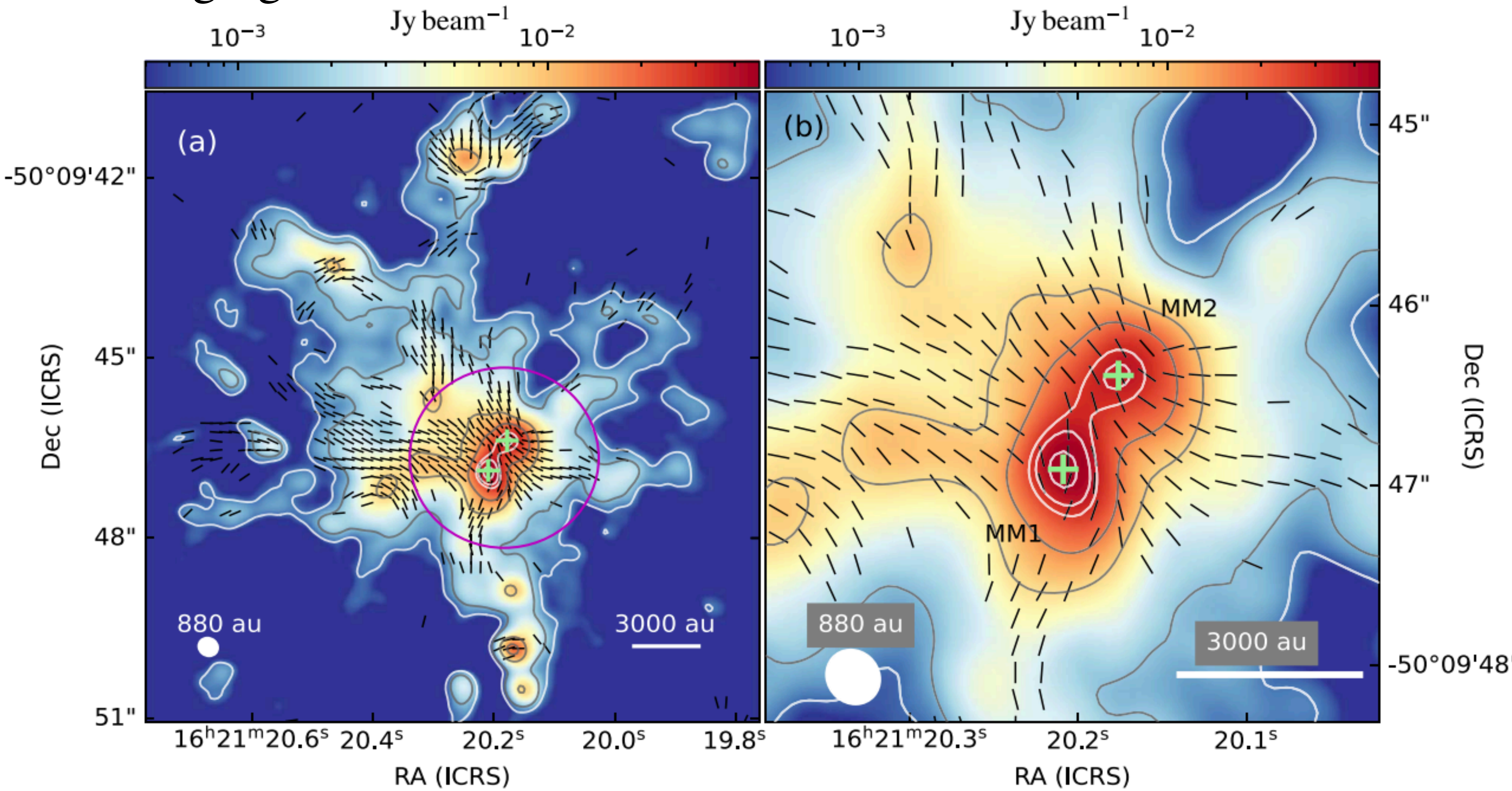


Image credit: Saha et al. 2024 ApJL, 972, L6

Finish!