VLASS Project Memo: 17

Utilizing a Customized VLASS Single Epoch Continuum Pipeline for End-User Science

E. Carlson (University of Rhode Island, NRAO[1])
A. Kimball (NRAO)
September 16, 2021

1 Introduction

The Very Large Array Sky Survey, proposed in 2014 (Lacy et al., 2020) is an all-sky radio survey that takes advantage of the new Karl G. Jansky Very Large Array (JVLA;Perley et al. (2011)). The increased capabilities of the JVLA and the upcoming Square Kilometre Array (SKA), warranted a survey that was both distinct but complimentary to SKA and previous VLA surveys such as FIRST [2] (Becker et al., 1994) and NVSS[3] (Condon et al., 1998).

With astronomy increasingly becoming a multi-wavelength discipline, the ability to compare images across observatories and across time has become critical to understanding our universe as a hole. VLASS contributes to our understanding of the universe by providing high-resolution S-Band B-Array ($\theta_{HPBW} = 2.1$ arcseconds) images of the radio sky through the use of On-the-Fly (OTF) mosaicking. OTF moasiacking, as shown in Figure 1, uses the array to take a snapshot (marked by the X's) before slewing in right ascension to the next snapshot. When the border of a predefined tile is reached the telescope repeats the process, this time at a higher declination. The primary beams of the each snapshot (marked by the orange circles), overlap to produce the equivalent time-on-source of a 5 second pointed observation. More information about the OTF Mosaicking is located here.

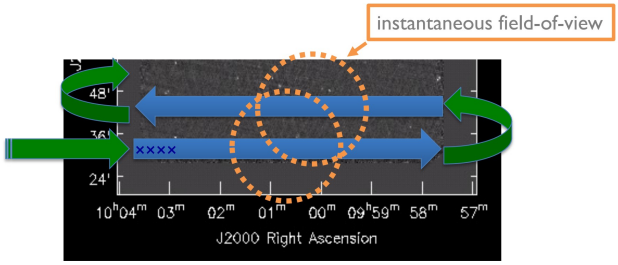


Figure 1: Diagram of the On-the-Fly mosaicking Scheme

Once the data has been collected the VLASS calibration and imaging pipeline takes the raw measurement set data, calibrates it and produces 1 square degree images of the radio sky. To prevent aliasing of radio sources when preforming fast Fourier transform operations (i.e. CLEAN or TCLEAN) on the visibility data, a buffer must be included in the desired image size. For VLASS quick look and single epoch images, a two-square degree image is produced with only the center one-square degree being used for science.

Traditionally, VLA surveys such as FIRST(Becker et al., 1994) and NVSS (Condon et al., 1998) have provided basic search capabilities of catalogs and products. Although the data from these surveys are located

[1]Graduate Student Intern; Summer 2021
[2]http://sundog.stsci.edu/cgi-bin/searchfirst
[3]https://www.cv.nrao.edu/nvss/postage.shtml

in the NRAO archive, guides related to the reduction and imaging of this data remain closed to the public.

To change this, VLASS has extended the search capability to visibility data, calibration products and deep/multi-epoch images. With the completion of the VLASS Imaging Project (VIP) (VLASS Memo 15) in April 2020, the production of ≈ 10 Single Epoch images were produced to fulfill VLASS project requirements. With the completion and validation of this pipeline (VLASS Memo 16, *in prep.*), VLASS pipeline development can begin to focus on Enhanced Data Products (EDP) as outlined in VLASS Memo 3. One such EDP is the capability for "processing on-demand" (POD) of images or image cubes. This memo will outline and evaluate the production of some of these POD images given the current[4] version of the CASA VLASS Pipeline. It will also provide more detailed, step-by-step instructions in the attached appendix.

2 Expanding the VLASS Pipeline Capabilities

To achieve EDPs as required by the VLASS Science goals the current system (as of 08/2021) needs to be upgraded to include diverse user customization. To that end, several keys areas needing customization were identified. They are:

1. Obtaining VLASS Measurement Sets
2. Generating a User-Defined Image of a Source
3. Multi-Tile Image Reduction

Each of these items and their solutions will be briefly explored in the following subsections. A dedicated "guide" has been generated for each use case and a link will be identified by the footnote corresponding with each subsection.

2.1 Identifying and Obtaining VLASS Data

The current methodology for a user to obtain pointed observations from NRAO is to utilize the data archive located at data.nrao.edu. The archive uses the metadata of each observation to determine if an observation covers the right ascension and declination as defined by the user. Given the on-the-fly moasicking utilized by VLASS, different metadata is produce preventing robust use of the NRAO data archive. This is because the OTF moasicking field metadata only contains the start position of each raster and not the right ascensions and declinations covered by that raster.

For a user to obtain data on a particular source, they must identify which tile the source exists in and then explore VLASS weblogs to identify which measurement set their source exists in. If the source lands near the edge or corner of a tile, the user will also need to identify the adjacent tile(s) and their associated measurement sets (for tile definitions see VLASS Memo 7).

To simplify this, a Jupyter Notebook was developed which creates an experience similar to the NRAO data archives. This notebook intakes the right ascension, declination as well as the desired image size and identifies all of the measurement sets whose scans contribute to the user's desired image. We did this by producing a basic CSV listing of the tiles, their boundaries, and their tile ids and utilized a Jupyter Notebook to identify our tile id. This script then used the identified tile id to generate a URL where the weblog of the quick look image exists. From this file, the measurement set is identified, appended to a list, and printed to the console.

This script works in a similar manner for sources on the edge of tiles by generating a series of right ascensions and declinations and identifying which tiles these points exist in. Duplicate measurement sets are not printed to the console by ensuring that only unique entries are contained in the list of measurement sets before being shown to the user.

[4] casa-6.1.3-3-pipeline-2021-1.1.29

Actively, this script prints all measurement sets in which scans contribute significantly[5] to the putative field of interest. This includes data across multiple epochs. For sources and fields which do not drastically vary in brightness over time, this data could be concatenated and imaged together (R. Perley and A. Kimball, *private communication*). Sources, which are excessively variable over time are not covered in the memo. This concatenation is explored further in Section 2.3. However, for the most stable use of the pipeline it is recommended that the user adhere to utilizing measurement sets from the same epoch.

2.2 Using the Image Parameters File to Modify the VLASS Pipeline

Much of the VLASS Imaging Pipeline has been developed with the primary science goals in mind. That is to say images, generated on predetermined phase centers, of a fixed size(see Section 1 for details) with many of the default values being hard-coded into the pipeline recipe. Despite this, the user can utilize an image parameters file to override many of these default values. The down side of this, is that it requires significant coordination for the end user.

For example, the pipeline task *hif_editimlist* that is run within a standard pipeline reduction is hard-coded to identify fields that contribute to the one-square degree final image and its buffer. To get around this, the user must utilize the script detailed in Section 6 to identify which fields are required for imaging. To do so, and to provide an overall more efficient pipeline experience, E.V.C modified a version of the code run by *hif_editimlist* and transitioned it into a python function and made easily available to the user. This function intakes the path of the measurement set, image size and desired phase center and provides the user with a list of fields. The user must then use the CASA task *mstransform* or *split* to generate a sub-measurement set. It is this secondary measurement set that the user can run the VLASS pipeline with a further customized image parameters file.

To generate our custom image parameters file for a custom sized image, we must first identify two items. What cell size[6] do we want for our image? And what is the final desired image size? Given these two pieces of information, as well as the size of our buffer to prevent aliasing of sources into our field, we can calculate the number of pixels in our image. This number is then modified to allow for proper usage with the *tclean* algorithm. These values are then appended to the image parameters file, which are read by the imaging pipeline and override the default parameters used by the typical VLASS imaging recipe. A sample image parameter file is provided in the appendix below.

2.3 Areas Requiring Continued Development

As part of the development of this EDP, several use cases were devised to identify further areas of pipeline development. These use cases are:

1. Non-Standard Phase Centers
2. Non-Standard Image Sizes
3. Non-Standard Phase Centers and Image Sizes
4. Images across Multiple Tiles
5. Images across Multiple Epochs
6. Images across Multiple Tiles and Multiple Epochs

For images across tiles or epochs, two or more measurement sets are required. However, as more deeply explored in Section 4, this is not supported by the pipeline. It is recommended that the user split off the required fields from each measurement set independently and manually image them through the normal process.

[5] Traditionally, for QL and SE images, this would include every field within twice the image size. However, we utilize fields whose primary beams contribute 20% or more to the phase center. This is calculated to be approximately 1000 arcseconds.

[6] The cell size should always be ≤ 0.6 arcseconds or smaller to properly sample the synthesized beam

For images from a single measurement set, , and any combination of the two, no modifications beyond those already explored in this memo are required even for non-standard phase centers, non-standard image sizes.

3 Data Products and Analysis

Given the ability to generate custom sized images with the VLASS pipeline, several images were generated of two specifically chosen quasars. These quasars were chosen due to the wide availability of pointed observations and their well-known morphology. The goals of these images were to quantify the temporal benefit of generating smaller images as well as to identify the stability of the pipeline on non-standard images. These sources outlined briefly in Table 1 are explored in more detail in Gobeille (2011) and Gobeille et al. (2014).

J2000 Name	Redshift	Right Ascension	Declination	Alternative Name	SDSS Classifciation
J0807+0432	2.876	08h07m57.5385s	4d32m34.531s	OJ+008	QSO
J0925+1444	0.896	09h25m07.271s	14d44m25.74s	OK+136	QSO

Table 1: Selected High-Powered[7]Quasars for Imaging

These images were produced using the same methodology previously outlined in this memo and more fully outlined in Section 6. The images were produced at an image size of 500, 1000 and 1500 arcseconds with a 1000 arcsecond buffer with a cell size of 1 arcsecond. The images were then cropped using CASA's (McMullin et al., 2007) *image.crop* command to include a region 500 arcseconds in size centered on the source. Additionally, difference images were produced, comparing each 500 arcsecond cutout to each other to ensure the results that were produced by the imaging pipeline were self-consistent when using different images. This was done using CASA's *immath* command. These images are present in Figure 4 and 5 respectively.

3.1 Analyzing the Images

Once the images were produced, they were exported to CARTA (Comrie et al., 2021) for analysis. Rectangular regions were created to encapsulate the putative core of the quasar, extended regions and the source as a whole. A region also encapsulated the image as a whole to gather RMS noise information on the image as a whole. The measured quantities for each image are presented in Table 2 and Table 3.

Image Size	Peak Flux	Total Integrated Flux	Total Extended Flux	Total Core Flux	RMS Noise
500"	347 mJy/beam	440 mJy	71 mJy	344 mJy	1446 μ Jy/beam
1000"	347 mJy/beam	441 mJy	71 mJy	343 mJy	1456 μ Jy/beam
1500"	348 mJy/beam	442 mJy	71 mJy	343 mJy	1461 μ Jy/beam

Table 2: Measured Image Quantities for J0807+0432

Figures 4 and 5 demonstrate there were no major changes in morphology, selecting different map sizes. We also saw consistent values across all of our measured values with in ± 1 mJy. However, we did see slight increases in the measured RMS noise within the 500 arcsecond region as the base image increased in size.

[7] $P_{Tot1.4GHz} > 10^{27.55} \frac{W}{Hz}$

Image Size	Peak Flux	Total Integrated Flux	Total Extended Flux	Total Core Flux	RMS Noise
500”	95 mJy/beam	346 mJy	299 mJy	34 mJy	626 μ Jy/beam
1000”	95 mJy/beam	345 mJy	298 mJy	34 mJy	630 μ Jy/beam
1500”	95 mJy/beam	346 mJy	300 mJy	34 mJy	633 μ Jy/beam

Table 3: Measured Image Quantities for J0925+1444

To ensure the values we collected made physical sense, we utilized L-Band and C-Band flux densities taken at similar resolution taken from Gobeille (2011) and Condon et al. (1998) to make a basic Flux Density versus Frequency plot as presented in Figure 2. Simply looking at the measured values of sources, we can identify J0925+1444 as a lobe-dominated quasar and due to the synchrotron emission present in the lobes, we expect the source will be steep in spectrum. This is confirmed by the data we have collected. On the other hand, J0807+0432 is a core-dominated quasar and is expected to have a flatter spectrum. Given this, we should expect our S-Band fluxes to be greater than our C-Band fluxes, while being less than our L-Band fluxes. The values presented in Figure 2 support these claims. Thus given these two examples, the imaging pipeline produces physical flux values [8]in line with what is expected from these sources. Given that the changes to

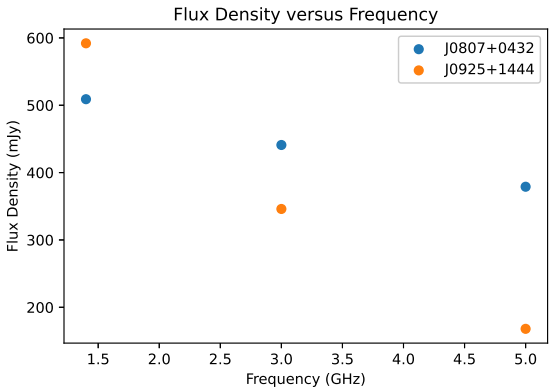


Figure 2: Flux Density versus Frequency
Sources: Gobeille (2011) & Condon et al. (1998)

the production of the VLASS images do not change the scientific results produced, we also looked at the computational benefits to both NRAO and the end-user. Given the limited disk space available to standard observer accounts at NRAO, the use of entire VLASS measurement sets is unreasonable. By processing the measurement sets with the archive, splitting off the required fields and then removing the original measurement sets, we are saving hundreds of gigabytes of disk-space from being unnecessarily utilized. Additionally the reduction of these data sets for a one-square-degree image can take on the order of several days. By utilizing the recipe outlined in this memo, there are several areas in which the run-time of the pipeline is reduced. First, by splitting off the required fields before the pipeline is run, CASA task *hif_importdata* references a much smaller measurement saving, in some cases several hours of compute time. Additionally, by selecting a smaller image size, the *hif_makeimages* command can be run much more efficently. This is mainly due to a faster *tclean* run. This frequently reduces compute time from several days to several hours. By using the custom recipe presented in this memo, the user can utilize significantly less disk space and computational resources.

In Figure 3 we demonstrate some of these gains by showing the production time for the three images that we produced for each source. It is important to note from this plot that the relationship for time is not as linear as it appears. As a user selects a smaller and smaller image size, there is always the requirement for the 1000 arcsecond image buffer which will contribute to the overall size of the image passed to *tclean*.

[8]Flux densities in VLASS 1.1 can be very unreliable and should not be used. Reference VLASS Memo 13 for more information.

This manifests itself as diminishing returns as the user produces smaller and smaller image sizes.

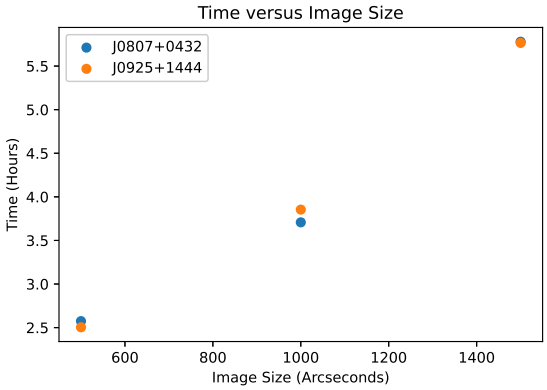


Figure 3: Production Time versus Image Size

4 Detailed Recommendations

4.1 Image Masking Failures

Over of the course of this exploration, several areas of development were identified in the VLASS Imaging Pipeline. One of the most common failures during a single-tile run of the VLASS imaging pipeline is the failure to produce an imaging mask. This frequently happens in sources with extended structure at the default (0.6 square arcsecond) pixel size. To troubleshoot this issue, we reran the imaging pipeline with a more coarse cell size (1 square arcsecond) which resulted in the proper imaging of our source. Although untested in this memo, we hypothesize that selecting even finer cell sizes will result in similar issues. Identifying and quantifying this value for user's will enable them to more appropriately select their desired cell size for their source at the start of an imaging run.

4.2 Multi-Tile Imaging

A further area of development is imaging across tiles through the use of multiple tiles or mulitple epochs. Concatenation of two files leads to an error during the *hif_importdata* command and prevents further use of the imaging pipeline. This error is not presented when one singular measurement set is split apart and then recombined, leading us to postulate is is related to how the pipeline queries the measurement sets' metadata. To troubleshoot this, we attempted to pass a list of measurement sets through the pipeline. Running the entire job until completion resulted in an error with the *tclean* command. We identified that *tclean* commmand was only being passed a string literal containing a list of fields from one measurement set instead of a list containing the strings for, in our case, both measurement sets. We did identify however, that abandoning the imaging pipeline entirely and manually imaging the mosaic was sufficient.

4.3 Unmanageable Data Retrieval for Large Surveys

Given the whole sky nature of VLASS, we foresee scientists using in to preview large populations of sources. At this this time, the archive must apply calibration tables to each requested VLASS measurement set. By doing so, CPU time is being redirected from other critical processes, frequently for 2-4 days at a time, to apply these tables. If a user requests even a modest (≈ 100) number[9] of VLASS measurement sets from the archive, this ties up a significant amount of compute resources. If the VLASS team cached measurement sets with applied calibration tables on disk, bypassing the archive entirely, the user would simply be able to point the scripts generated in this memo at these measurement sets saving significant amounts of CPU time

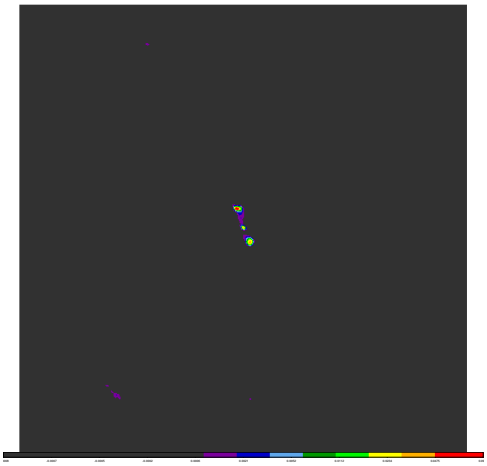
[9]This is roughly 25% of the size of the 3C Sample (Laing et al., 1983)

on the NRAO compute cluster.

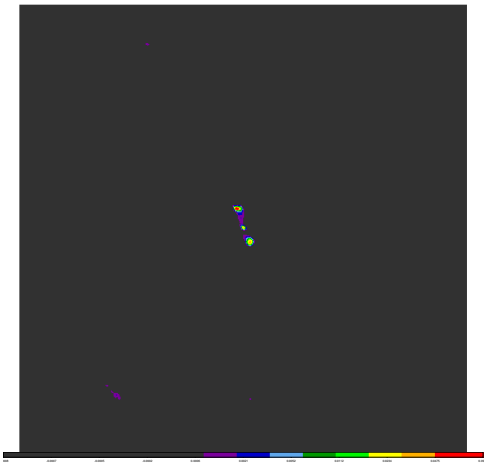
Although disk-space intensive, hundreds of hours of compute time will be save for both the user and NRAO as calibration tables will not have to be constantly reapplied to the VLASS measurement sets. Additionally, this would save additional compute hours for the end user as they must download and decompress their data from the archive the NRAO compute cluster. It would also save the user from having to juggle the limited disk quota that they are subjected to as they load and offload measurement sets for sources in their sample.

5 Summary

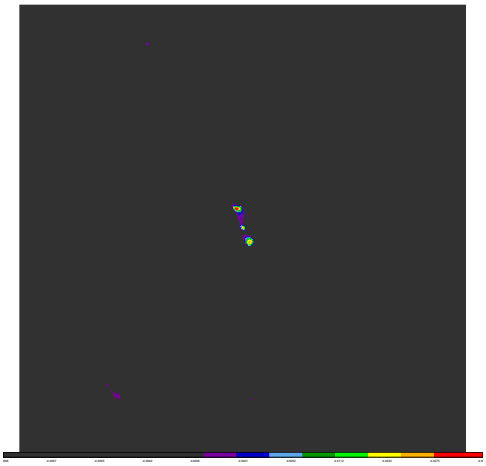
In summary, we have presented several use cases, tools and recommendations to produce EDPs to achieve VLASS program requirements. First, for basic use cases such as non-standard phase centers, non-standard image sizes and the combination of non-standard phase centers and image sizes, the VLASS Single Epoch pipeline is able to successfully produce an image that can be used for end-user science. The process requires minimal user-effort with a complete process being compiled in Section 6. We explored images from two sources to ensure that changing the image size did not drastically effect any of the imaging algorithms. Finally, we presented several recommendations to the VLASS team, to continue to troubleshoot remaining areas of in which the pipeline does not run as intended.



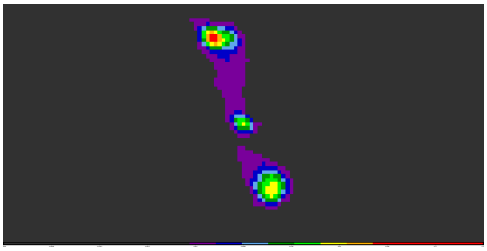
(a) J0925+1444 created at 500”



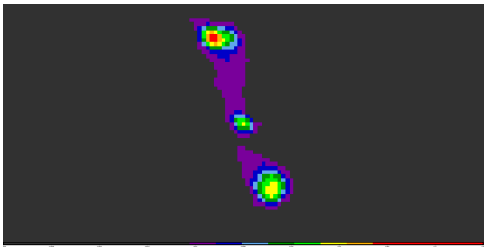
(b) J0925+1444 created at 1000”



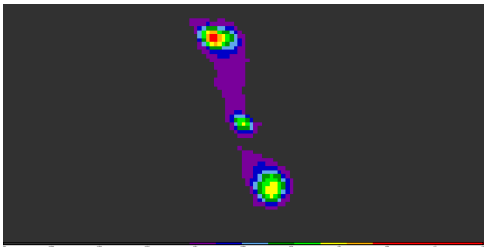
(c) J0925+1444 created at 1500”



(d) Zoomed in View of J0925+1444 created at 500”



(e) Zoomed in View of J0925+1444 created at 1000”



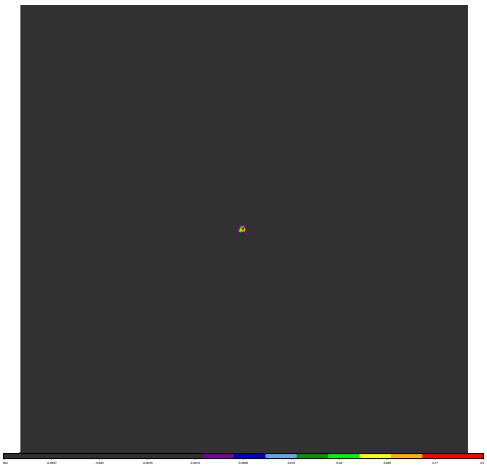
(f) Zoomed in View of J0925+1444 created at 1500”

(g) 500” Image Minus 1000” Image

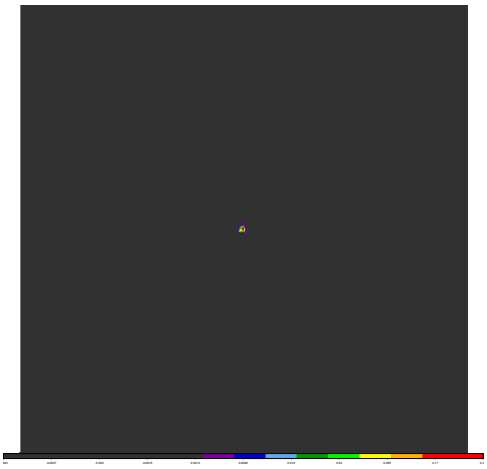
(h) 500” Image Minus 1500” Image

(i) 1500” Image Minus 1000” Image

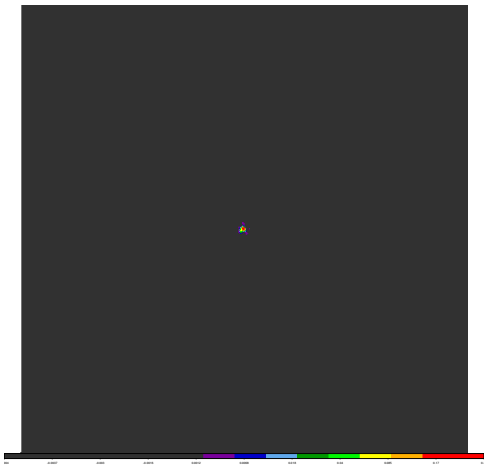
Figure 4



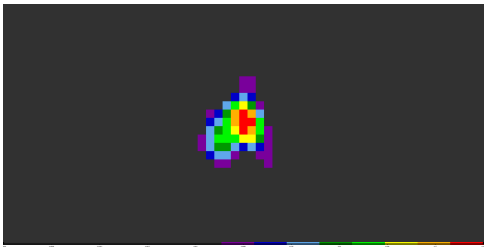
(a) J0807+0432 created at 500”



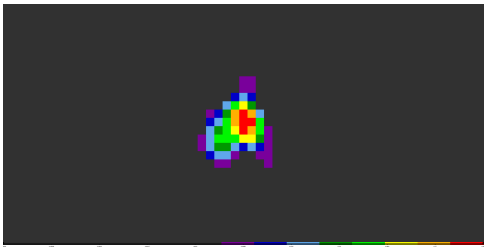
(b) J0807+0432 created at 1000”



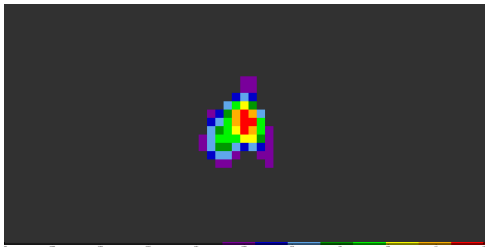
(c) J0807+0432 created at 1500”



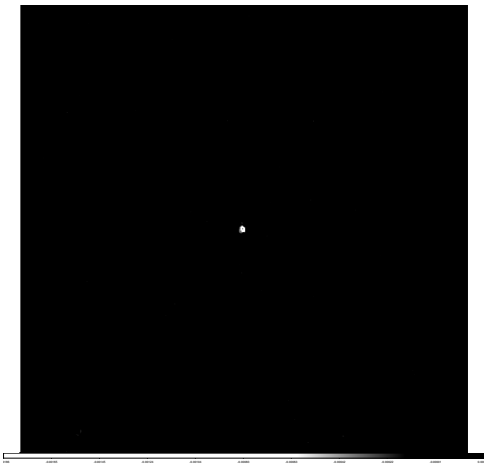
(d) Zoomed in View of J0807+0432 created at 500”



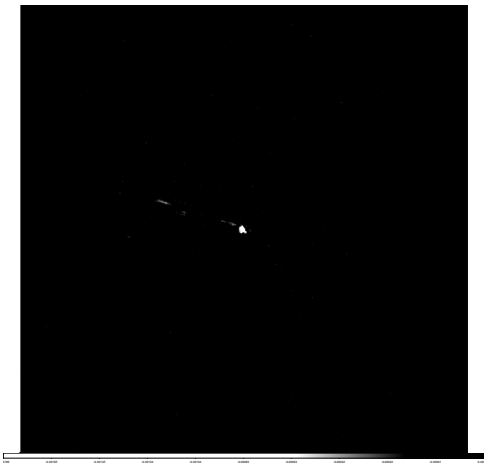
(e) Zoomed in View of J0807+0432 created at 1000”



(f) Zoomed in View of J0807+0432 created at 1500”



(g) 500” Image Minus 1000” Image



(h) 500” Image Minus 1500” Image

(i) 1500” Image Minus 1000” Image

Figure 5

6 Appendix: Guide for Generating Images

The following documentation outlines how to generate a custom-image utilizing a VLASS observation of a source. For the first half of this documentation we will be looking at the high-powered quasar [HB89] 0805+046, known in this guide as J0807+0432.

The first step in generating a custom image is to identify the required measurement sets for which data is to be pulled from. The code for this is located at the following link to my Github repository. This file as well as the CSV defining tile locations located here, should be placed in a working directory. It is recommended that the user use an online iPython interface like a Jupyter Notebook or Google Collab. Although utilizing Python in a command-line setting will also work, it may require changes which are beyond the scope of this guide.

Running the code, and following the prompts to enter the Right Ascension and Declination in decimal degrees as well as an integer with the number of arcseconds for which we want to produce an image, we are given the output in Figure 6 identifying which measurement sets are required.

```
Please Enter your Right Ascention in Decimal Format: 121.98974
Please Enter your Declination in Decimal Format: 4.54293
Please Enter the Proposed Image Size to the nearest Arcsecond: 250

The unique measurement sets required are:
VLASS2.1.sb38561374.eb38565040.59070.62333981482
VLASS1.1.sb34647560.eb34700758.58075.26425702547
```

Figure 6: Prompts and Results Presented by the Jupyter Notebook

For the purposes of this guide, we will use the VLASS2.1.sb38561374.eb38565040.59070.62333981482 measurement set. To obtain our calibration products we will go to data.nrao.edu. Simply copying and pasting the string of our measurement set we are presented with the screen in Figure 7.

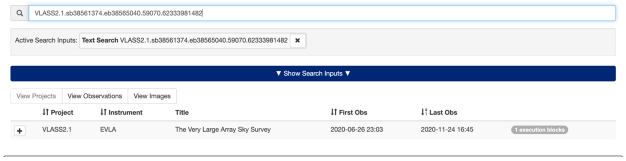


Figure 7: Copying and Pasting the Measurement Set Name into the Archive

Selecting the cross, the clipboard and "download" we are presented with the screen presented in Figure 8.

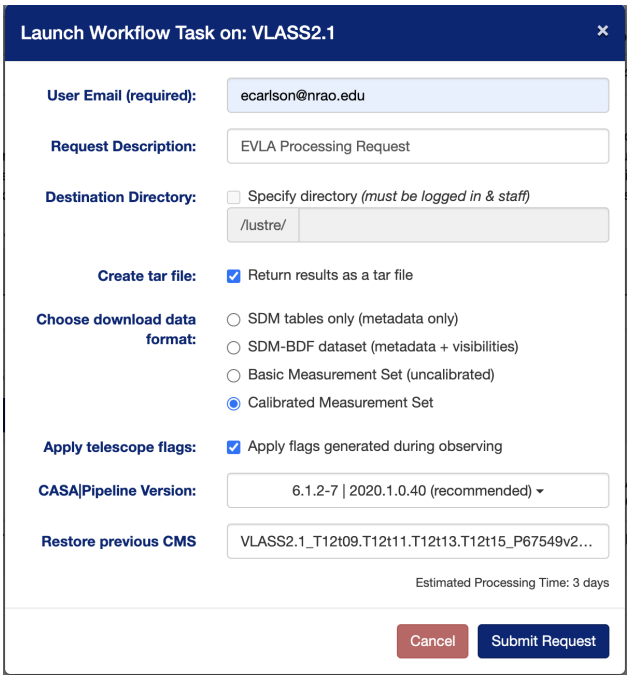


Figure 8: Copying and Pasting the Measurement Set Name into the Archive

We must then enter our email and submit the request.

After the archive applies the calibration tables, which will take roughly 3-4 days, we will receive an email similar to the one in Figure 9.

Figure 9: A sample email provided by the archive alerting the user their data is ready.

Logging into our NRAO compute cluster account, we can use the *wget* command to transfer our data to a working directory. For the case below (which is a different measurement set and a unique location from what the user will receive) is:

```
wget -r https://dl-dsoc.nrao.edu/anonymous/810337739/
    qt7b2k0qbsbvvqbtl2m117i3hi/VLASS2.1.sb38561374.eb38565040
    .59070.62333981482
```

Once the data has transferred (generally >100 minutes), we can begin to run some of our custom scripts before we begin the VLASS pipeline. The first of these scripts is used to identify what fields are required. To utilize the script, located here, the user must first open a version of casa containing the pipeline. This must be done in the working directory containing the VLASS measurement set.

Additionally, the user must have also generated a text file which will contain their list of image parameters. The text file should contain the desired image name, image center, image mode and image size. To calculate the image size this Python script can be used. If a custom cell size is to be selected it must be put

into the image parameters file. A sample of a complete image parameters file is located here. This text file must also be in the working directory.

Once these preliminary steps are complete, the user can copy and paste the carlson_editimlist_prep.py file into CASA to initialize the function. The user should then run the following command:

```
field_list = carlson_editimlist_prep('VLASS2.1.sb38561374.eb38565040
    .59070.62333981482.ms', 500, 'J2000 08:07:57.5 +04.32.34.6', matchregex
    =['^0', '^1', '^2'])
```

Where the first argument is the string for the measurement set, the second is an integer with the desired image size in arcseconds and the desired phase center. The matchregex argument should be left as the default value.

Once the script is done running, the user can now run the split command on the original dataset and remove it from disk. This is done using the following command:

```
split(vis = 'VLASS2.1.sb38561374.eb38565040.59070.62333981482.ms', outputvis =
    'name_for_ms.ms', field = field_list)
```

Now that the data has been properly split off, the user can now run a normal VLASS SE imaging pipeline script with the exception of custom image parameters. To complete this, the user must generate a working directory with the following structure:

```
SourceName
├── working
└── products
    ├── command_script.py
    ├── run_SE_SourceName.sh
    └── image_parameter.list
```

Examples of the command script, run script and image parameter script are located here. To calculate the required image size, the user must first choose a cell size. The default for quick-look images is 1 arcsecond with the default for Single Epoch images being 0.6 arcseconds. Once the cell size has been determined, the user must then calculate an image size using the following formula:

$$P = (I + 1000)/C \tag{1}$$

Where I is the image size in arcseconds, C is the cell size in arcseconds and P is the image size in pixels. It is this value P which is appended to the image_parameter.list file. It is important to note that *tclean* is optimized for imsize values that are even and factorizable by 2,3,5,7 only. A helpful tool to assist with determining the proper image size is located here.

Once the directory structure, image parameter list, command script and shell script are properly formatted, the user can now submit a non-interactive job using the methodology outline here. Upon the completion of the run, the user should inspect the weblog for warnings and errors. Interpreting and troubleshooting the results of the VLASS imaging pipeline is examined here.

References

Becker, R. H., White, R. L., & Helfand, D. J. 1994, in Astronomical Society of the Pacific Conference Series, Vol. 61, Astronomical Data Analysis Software and Systems III, ed. D. R. Crabtree, R. J. Hanisch, & J. Barnes, 165

Comrie, A., Wang, K.-S., Hsu, S.-C., et al. 2021, CARTA: The Cube Analysis and Rendering Tool for Astronomy, v2.0.0, Zenodo, doi:10.5281/zenodo.3377984

Condon, J. J., Cotton, W. D., Greisen, E. W., et al. 1998, AJ, 115, 1693

Gobeille, D. B., Wardle, J. F. C., & Cheung, C. C. 2014, arXiv e-prints, arXiv:1406.4797

Gobeille, D. B. P. 2011, PhD thesis, Brandeis University

Lacy, M., Baum, S. A., Chandler, C. J., et al. 2020, Publications of the Astronomical Society of the Pacific, 132, 035001. https://doi.org/10.1088/1538-3873/ab63eb

Laing, R. A., Riley, J. M., & Longair, M. S. 1983, MNRAS, 204, 151

McMullin, J. P., Waters, B., Schiebel, D., Young, W., & Golap, K. 2007, in Astronomical Society of the Pacific Conference Series, Vol. 376, Astronomical Data Analysis Software and Systems XVI, ed. R. A. Shaw, F. Hill, & D. J. Bell, 127

Perley, R. A., Chandler, C. J., Butler, B. J., & Wrobel, J. M. 2011, ApJ, 739, L1