***To get drizzle and dash working check***

Conda --version

Source activate astroconda

Conda --version

Ipython

Follow <https://nbviewer.org/github/spacetelescope/notebooks/blob/master/notebooks/DrizzlePac/align_to_catalogs/align_to_catalogs.ipynb#Inspect-the-image-header>

import os

import glob

import numpy as np

import matplotlib.pyplot as plt

from astropy.io import fits

from astropy.table import Table

from astropy.io import ascii

from astroquery.mast import Observations

from drizzlepac import astrodrizzle

from reduce\_dash import DashData

proposal\_id=15153

obstype='all'

filters=['F126N', 'F128N', 'F164N']

obs\_id=['idn115010', 'idn123010', 'idn107010']

obsTable = Observations.query\_criteria(obs\_id=obs\_id, proposal\_id=proposal\_id, obstype=obstype, filters=filters)

products = Observations.get\_product\_list(obsTable)

filtered\_products = Observations.filter\_products(products, productSubGroupDescription='FLT')

Observations.download\_products(filtered\_products, mrp\_only=**False**)

filtered\_products\_ima = Observations.filter\_products(products, productSubGroupDescription='IMA')

Observations.download\_products(filtered\_products\_ima, mrp\_only=False)

for flt in glob.glob('./mastDownload/HST/\*/\*flt.fits'):

flt\_name = os.path.split(flt)[-1]

os.rename(flt, flt\_name)

for ima in glob.glob('./mastDownload/HST/\*/\*ima.fits'):

ima\_name = os.path.split(ima)[-1]

os.rename(ima, ima\_name)

flt\_files = glob.glob('mastDownload/HST/\*/\*\_flt.fits')

ima\_files = glob.glob('mastDownload/HST/\*/\*\_ima.fits')

OR IF I KNOW THE FRAMES…[need to sort because ends up out of order for some reason]

flt\_files = np.sort(glob.glob('\*flt\*'))

ima\_files = np.sort(glob.glob('\*ima\*'))

os**.**environ['CRDS\_SERVER\_URL'] **=** 'https://hst-crds.stsci.edu'

os**.**environ['CRDS\_SERVER'] **=** 'https://hst-crds.stsci.edu'

os**.**environ['CRDS\_PATH'] **=** './crds\_cache'

os**.**environ['iref'] **=** 'iref/'

**if** **not** os**.**path**.**exists('iref'):

os**.**mkdir('iref')

**for** file **in** ima\_files:

command\_line\_input **=** 'crds bestrefs --files {:} --sync-references=1 --update-bestrefs'**.**format(file)

os**.**system(command\_line\_input)

myDash\_list = []

for i in range(len(flt\_files)):

myDash\_list.append(DashData(ima\_files[i], flt\_files[i]))

threshold\_list = [1., 5., 10., 50., 100.]

threshold = 20.

for myDash in [myDash\_list[0]]:

myDash.split\_ima()

myDash.make\_pointing\_asn()

asn\_filename = 'diff/{}\_asn.fits'.format(myDash.root)

asn\_table = Table(fits.getdata(asn\_filename, ext=1))

asn\_table.show\_in\_notebook()

myDash.create\_seg\_map()

rootname = myDash.root

#segmap\_name = ('segmentation\_maps/'+ rootname + '\_seg.fits')

#segmap = fits.getdata(segmap\_name)

sourcelist\_name = ('segmentation\_maps/' + rootname + '\_source\_list.dat')

sourcelist = ascii.read(sourcelist\_name)

print(sourcelist)

diffpath = os.path.dirname(os.path.abspath('diff/{}\_\*\_diff.fits'.format(rootname)))

cat\_images = sorted([os.path.basename(x) for x in glob.glob('diff/{}\_\*\_diff.fits'.format(rootname))])

sc\_diff\_files = [diffpath + '/' + s for s in cat\_images]

myDash.diff\_seg\_map(cat\_images=sc\_diff\_files)

myDash.subtract\_background\_reads()

myDash.fix\_cosmic\_rays()

myDash.align(threshold=threshold, drz\_output= myDash**.**root **+** '\_thresh' + str(threshold)[:-2] **+** '\_nodrz\_sci.fits', updatehdr**=True**, updatewcs**=True**, astrodriz**=True**)

sci\_name **=** myDash**.**root **+** '\_thresh' + str(threshold)[:-2] **+** '\_nodrz\_sci.fits'

#og\_flt\_name **=** 'mastDownload/HST/' **+** myDash**.**root **+** '/' **+** myDash**.**root **+** '\_ima.fits'

og\_flt\_name **=** myDash**.**root **+** '\_ima.fits'

sci **=** fits**.**getdata(sci\_name)

og\_flt **=** fits**.**getdata(og\_flt\_name)

fig **=** plt**.**figure(figsize**=**(9, 4))

ax1 **=** fig**.**add\_subplot(1,2,2)

ax2 **=** fig**.**add\_subplot(1,2,1)

ax1**.**set\_title('DASH Pipeline Reduced Science File')

ax2**.**set\_title('Original IMA (not reduced using pipeline)')

ax1**.**set\_xlim(**-**10,1120)

ax2**.**set\_xlim(**-**10,1120)

ax1**.**set\_ylim(**-**10,1050)

ax2**.**set\_ylim(**-**10,1050)

ax1**.**imshow(sci, vmin**=**0, vmax**=**40, cmap**=**'Greys\_r', origin**=**'lower')

ax2**.**imshow(og\_flt, vmin**=**0, vmax**=**40, cmap**=**'Greys\_r', origin**=**'lower')

plt**.savefig(sci\_name +** '.png'**)**

for threshold in threshold\_list:

myDash = myDash\_list[0]

myDash.split\_ima()

myDash.make\_pointing\_asn()

asn\_filename = 'diff/{}\_asn.fits'.format(myDash.root)

asn\_table = Table(fits.getdata(asn\_filename, ext=1))

asn\_table.show\_in\_notebook()

myDash.create\_seg\_map()

rootname = myDash.root

segmap\_name = ('segmentation\_maps/'+ rootname + '\_seg.fits')

segmap = fits.getdata(segmap\_name)

sourcelist\_name = ('segmentation\_maps/' + rootname + '\_source\_list.dat')

sourcelist = ascii.read(sourcelist\_name)

print(sourcelist)

diffpath = os.path.dirname(os.path.abspath('diff/{}\_\*\_diff.fits'.format(rootname)))

cat\_images = sorted([os.path.basename(x) for x in glob.glob('diff/{}\_\*\_diff.fits'.format(rootname))])

sc\_diff\_files = [diffpath + '/' + s for s in cat\_images]

myDash.diff\_seg\_map(cat\_images=sc\_diff\_files)

myDash.subtract\_background\_reads()

myDash.fix\_cosmic\_rays()

myDash.align(threshold=threshold, drz\_output= myDash**.**root **+** '\_thresh' + str(threshold)[:-2] **+** '\_drz\_sci.fits')

sci\_name **=** myDash**.**root **+** '\_thresh' + str(threshold)[:-2] **+** '\_drz\_sci.fits'

#og\_flt\_name **=** 'mastDownload/HST/' **+** myDash**.**root **+** '/' **+** myDash**.**root **+** '\_ima.fits'

og\_flt\_name **=** myDash**.**root **+** '\_ima.fits'

sci **=** fits**.**getdata(sci\_name)

og\_flt **=** fits**.**getdata(og\_flt\_name)

fig **=** plt**.**figure(figsize**=**(9, 4))

ax1 **=** fig**.**add\_subplot(1,2,2)

ax2 **=** fig**.**add\_subplot(1,2,1)

ax1**.**set\_title('DASH Pipeline Reduced Science File')

ax2**.**set\_title('Original IMA (not reduced using pipeline)')

ax1**.**set\_xlim(**-**10,1120)

ax2**.**set\_xlim(**-**10,1120)

ax1**.**set\_ylim(**-**10,1050)

ax2**.**set\_ylim(**-**10,1050)

ax1**.**imshow(sci, vmin**=**0, vmax**=**40, cmap**=**'Greys\_r', origin**=**'lower')

ax2**.**imshow(og\_flt, vmin**=**0, vmax**=**40, cmap**=**'Greys\_r', origin**=**'lower')

plt**.savefig(sci\_name +** '.png'**)**

and so on, remember for every step going forward anything with **flc** has to be swapped with **flt**

***References (the highlighted materials are most important)***

<https://arxiv.org/pdf/1603.00465.pdf> - the original paper by eva momcheva

<https://mast.stsci.edu/search/ui/#/hst/results?resolve=true&target=HH%206&radius=3&radius_units=arcminutes&data_type=image&observations=S,C&active_instruments=wfc3&proposal_id=15153&select_cols=ang_sep,sci_aper_1234,sci_central_wavelength,sci_data_set_name,sci_dec,sci_actual_duration,sci_spec_1234,sci_hlsp,sci_instrume,sci_pi_last_name,sci_preview_name,sci_pep_id,sci_ra,sci_refnum,sci_release_date,scp_scan_type,sci_start_time,sci_stop_time,sci_targname&useStore=false&search_key=483686f4e8faa> – joel example using the api

<https://mast.stsci.edu/search/ui/#/hst/results?resolve=true&target=HH%206&radius=25&radius_units=arcminutes&data_type=image&observations=S,C&active_instruments=wfc3&proposal_id=15153&select_cols=ang_sep,sci_aper_1234,sci_central_wavelength,sci_data_set_name,sci_dec,sci_actual_duration,sci_spec_1234,sci_hlsp,sci_instrume,sci_pi_last_name,sci_preview_name,sci_pep_id,sci_ra,sci_refnum,sci_release_date,scp_scan_type,sci_start_time,sci_stop_time,sci_targname&useStore=false&search_key=c70378862b57> – all of ngc 1333 (to look for the ones I want, which have the same IDs as [https://mast.stsci.edu/portal/Mashup/Clients/Mast/Portal.html?searchQuery=%7B%22service%22%3A%22CAOMBYOBS%22%2C%22inputText%22%3A%22idn107010%2C%20idn115010%2C%20idn123010%22%2C%22paramsService%22%3A%22Mast.Caom.SearchByObsID%22%2C%22title%22%3A%22Observation%20IDs%3A%20idn107010%2C%20id...%22%2C%22columns%22%3A%22\*%22%2C%22caomVersion%22%3Anull%7D](https://mast.stsci.edu/portal/Mashup/Clients/Mast/Portal.html?searchQuery=%7B%22service%22%3A%22CAOMBYOBS%22%2C%22inputText%22%3A%22idn107010%2C%20idn115010%2C%20idn123010%22%2C%22paramsService%22%3A%22Mast.Caom.SearchByObsID%22%2C%22title%22%3A%22Observation%20IDs%3A%20idn107010%2C%20id...%22%2C%22columns%22%3A%22*%22%2C%22caomVersion%22%3Anull%7D) the normal mast portal just with more info)

<https://www.stsci.edu/files/live/sites/www/files/home/hst/instrumentation/wfc3/documentation/instrument-science-reports-isrs/_documents/2021/2021-02.pdf> eva’s example steps on how to process dash mode data

<https://github.com/spacetelescope/wfc3_dash/blob/8aa7b925ab5ca23f57a830175b489ab6b1aa5af6/wfc3_dash/notebooks/Step_1-Creating_Science_Data_from_single_exposure_IMAs.ipynb> an example notebook for this process

<https://github.com/spacetelescope/notebooks/blob/master/notebooks/DrizzlePac/align_sparse_fields/align_sparse_fields.ipynb> or <https://github.com/spacetelescope/gaia_alignment/blob/master/Gaia_alignment.ipynb> an example for details on alignment for sparse fields and what to change when using drizzlepac and tweakreg…

although the claims by the above is to use <https://github.com/spacetelescope/wfc3tools> that actually does not work for dash mode! For that we must refer to <https://github.com/spacetelescope/WFC3Library> !

4/25/22 [**stuff to add to logbook**]

* Tried to get environment.yml from WFC3Library as per Joel’s rec
* It didn’t work because curl can’t work with that specific github /blob/master site. You need to use raw.github or the github api, which I can’t figure out
* I copy and pasted the raw text into a txt file, moved it via winscp
* Tried it using conda env create, got resolvepackagenotfound, which is a dependency error of some kind
* The dependencies likely involve some of it requiring pip…but this gets too complicated, randomly outputs in no order…
* Opting to just run the reduce\_dash.py and dash.ipynb by hand, fixing any incorrect packages by hand. These packages included
  + Lacosmicx
    - Solved with git clone lacosmicx repo, and then pip install ./lacosmicx
  + Utils
    - …you have to install as a separate file duh…sigh
* Next where I get stuck…user-wise…
  + All of step 2 works perfectly, though I set proposal\_id and obs\_id as a separate variable (see top of document)
  + The plotting doesn’t work while in ssh, that’s on me, I just opened the specific files
  + Querying CRDS doesn’t work unless I move the \_ima files out of the mastdownloads directory!
  + Only issue now is going through all the IMA files?
* The main frames to focus on (rather than waiting for them all to run) are
  + Idn107jhq
  + Idn115fqq
  + Idn123irq

4/26/22

Plan: try to work on just those three frames, not tilt them (maybe wcschange = False flag?) and work from there

Tests:

* Crds search only returns errors (still runs but not sure what it returns)
  + Issue was actually I wasn’t running the os.environ crds info…
* Running mydash.align and changing threshold changes nothing
* Next update I tried was to swap the segmentation map methods all with DAOstarfinder to create the star source catalog (instead of one with point like and extended sources)

**Tomorrow**:

* Need to add plot from <https://photutils.readthedocs.io/en/stable/detection.html> and clean up some of the imports in reduce\_dash.py (and/or find a way to produce fits file…but we’ll see)
* Possibly need to swap all mentions of “segmentation\_map” with some other word since I’m never using segmentation maps at all if I use daostarfinder

4/27/22

Me: “it can detect extended emission…but problem is if there’s lots of nebulosity…” Dan: “no but those are the same, what you mean is problem that tom megeath has which is about reflection nebulosity (which is where we’d get into issues with HH 12)

So my questions about segmentation maps…”

If it breaks, email with joel and fred…try to fix it…do tell fred he saw data before and dan says hi!

Break in a very specific way and ask why this set of steps didn’t work?

* It’s not rectilinear
* Going back a step to .align, why is it not aligning?

4/28/22

Sent email to joel et al…found out should defer to hst help desk

Also see below since Joel requested help for installing drizzlepac

Drizzlepac Installation notes:

(I have notes on this, also basically ignored the environment.yml on the WFC3Library page since it wouldn't work anyway).

Since I had no admin privilege, I ended up getting a fresh install of miniconda, setup an astroconda environment, checked versions for both conda (4.11.x) and python (3.7+ since <3.6 has issues) to make sure they're up to date, and then...I found out on github that a library called six is needed for some reason (makes no sense to me because six is for compatibility with py2, which is defunct? I don't think about these things too much). Anyway, I asked on github, and they said just try pip install six and somehow all that together worked...