Astrometric Registration

Story

Minimal: As a scientist reducing grism data, I'd like to be able to combine data from different visits. This absolutely requires that the direct images associated with the visits are registered on the same astrometric system. The astrometric solution changes need to be then propagated to the grism images appropriately.

Better: Correct for astrometry against an external catalog such as GAIA or PANSTARRS.

Inputs

Two or more different direct-grism visits of the same area on the sky.

CASE A: Simple example with the same filter-grism combination:

Visit 1:

Image 1: F140W / Grism 1: G141 / Image 2: F140W / Grism 2: G141 / Image 3: F140W / Grism 3: G141 / Image 4: F140W / Grism 4: G141

Visit 2:

Image 1: F140W / Grism 1: G141 / Image 2: F140W / Grism 2: G141 / Image 3: F140W / Grism 3: G141 / Image 4: F140W / Grism 4: G141

CASE B: Can be different filters and grisms:

Visit 1:

Image 1: F140W / Grism 1: G141 / Image 2: F140W / Grism 2: G141 / Image 3: F140W / Grism 3: G141 / Image 4: F140W / Grism 4: G141

Visit 2:

Image 1: F105W / Grism 1: G102 / Image 2: F105W / Grism 2: G102 / Image 3: F105W / Grism 3: G102 / Image 4: F105W / Grism 4: G102

CASE C: Not every grism image has a corresponding direct image:

Visit 1:

Image 1: F140W / Grism 1: G141 / Grism 2: G141 / Grism 3: G141 / Grism 4: G141

Visit 2:

Image 1: F105W / Grism 1: G102 / Grism 2: G102 / Image 3: F105W / Grism 3: G102 / Grism 4: G102

Outputs

The same files as the inputs but with new WCSs such that:

- All **direct images** are aligned to the same reference frame and a new WCS is written in the header. In the minimal case, the frame can be arbitrarily chosen such as the first direct image. There are small but measurable offsets even in dithered direct images within the same visit, typically fractions of a pixel (~0.1-0.2) so even images within the same visit should be alligned. In the better case, this can be a two-step process such that the images are aligned internally first and then to the external reference frame.
- All **grism images** are assigned the WCS offsets of the closest direct image. If there's no direct image for every grism image (Case C), apply the calculated offsets of the direct image WCS to all the subsequent grism images. For Case C Visit 1 all grism images will get the F140W WCS offsets. For Case C Visit 2, Grism 1 and Grism 2 will get Image 1 WCS offsets, Grism 3 and Grism 4 will get the Image 2 WCS offsets.

Computations

The script that does this for a single visit is in grizli.prep.process_direct_grism_visit. The script that loops over multiple visits as in the examples above is in girzli.pipeline.auto_script.preprocess. If you are writing this code, do not reinvent the wheel and just look there. There are lots of small things that need to be done for this to work for ACS and WFC3. For ACS I think it currently only calculates the offsets for the 1st detector for simplicity.

- Run source detection on each FLT/FLC direct image. Source detection can be done in the distorted FLT/FLC frame. All you care are the centroids of the sources. This can be done once for all the direct images in all the filter, make sure there is a consistent header parameter for the filter zero-point in the header. Might need to mask parts of the image with the DQ array and subtract a median to allow for cleaner source detection.
- To align to one of the images, say 0-th, for each [1...n] image in the list:
  - Trim the 0th and i-th image catalogs to the brightest (say 100) sources
  - Convert the x,y coordinates of the catalog sources of image i to RA, Dec (wcs_i.all_pix2world)
  - Convert the RA, Dec to x,y in the frame of image 0 (wcs_0.all_world2pix)
  - Calculate the offsets relative to the 0th image (tristars.match.match_catalog_tri)
  - Apply shifts to i-th image (drizzlepac.updatehdr.updatewcs_with_shift)
  - Apply the same shifts for all grism images associated with the i-th image
  - Drizzle all direct images together and generate a visit-level source catalog.

- To allign to an external catalog (grizli.prep.align drizzled_image) or a user provided catalog or a catalog from a different visit (the use case of an external catalog is pretty much the same as aligning multiple visits such as Case C):
    - Query the catalog server at the position of the image. This can also be user-supplied catalog, e.g., when you want to align to CANDELS.
    - Trim reference catalog to match image image footprint. Trim reference catalog to limit number of sources either magnitude limit (min and max for many catalog, saturated objects at bright magnitures) or a number of sources or source density.
    - Convert RA, Dec positions of reference catalog sources to X,Y positions in the image
    - Calculate transform between the 2 sets of x,y coordinates (grizli.prep.match_lists), usually want to calculate shift, scale and rot here! Good to iterate here a few times.
    - Update FLT/FLC images with tranform, update corresponding grism images.

Drawbacks

Some of these tasks are not grism specific. However this is really important for being able to combine grism visits that these tasks should exist somewhere. As can be seen from the several hundred lines of code grizli takes to do this, the task is not straightforward and even an expert user will struggle stringing all of these together.