

Streak connector

3 June 2018: Gary has completed a working version of *connect_streaks.py* to be added to pixcorrect. This code reads the streak tables produced for each individual CCD of the exposure (from files whose names are listed in an input text file) and determines which of these single-CCD streak "segments" are likely to be part of the same longer streak. It then fits a quadratic path to the endpoints of all the segments determined to be part of the same streak. The code then looks to see if the streak has crossed CCDs that do not already have a streak segment recorded in the immask tables. A new streak segment is created whenever such a "missed" segment is

- Between two detected segments (interpolated), OR
- Extrapolated beyond the extent of detected segments, but would reach within *max_extrapolate* degrees of the endpoint of a detected streak. This is a command-line parameter of the program.

New copies of the streak tables are then written to disk, appending info on any newly masked segments.

Then new copies of the images are written to disk, with the MSK plane of each image being updated with the STREAK bit set in the pixels corresponding to the newly masked streaks.

Here is the output of *connect_streaks.py*:

```
usage: connect_streaks.py [-h] [-s SAVECONFIG] [-l LOG] [-v]
                        [--streak_file STREAK_FILE]
                        [--image_file IMAGE_FILE]
                        [--streak_name_in STREAK_NAME_IN]
                        [--streak_name_out STREAK_NAME_OUT]
                        [--image_name_in IMAGE_NAME_IN]
                        [--image_name_out IMAGE_NAME_OUT]
                        [--add_width ADD_WIDTH]
                        [--max_extrapolate MAX_EXTRAPOLATE]
                        [--plotfile PLOTFILE]
                        [config]
```

Predict missed streak detections and mask them

positional arguments:

config Configuration file filename

optional arguments:

-h, --help show this help message and exit

-s SAVECONFIG, --saveconfig SAVECONFIG

output config file

-l LOG, --log LOG the name of the logfile

-v, --verbose be verbose

--streak_file STREAK_FILE

File holding list of input streak file names

--image_file IMAGE_FILE

File holding list of input image file names

--streak_name_in STREAK_NAME_IN

String to replace in input streak filenames

--streak_name_out STREAK_NAME_OUT

Replacement string for output streak filenames

--image_name_in IMAGE_NAME_IN

String to replace in input image filenames

```

--image_name_out IMAGE_NAME_OUT
                        Replacement string for output image filenames

--add_width ADD_WIDTH
                        number of pixels to grow (or shrink) streak width

--max_extrapolate MAX_EXTRAPOLATE
                        farthest to start a new streak from endpoint of an
                        existing one (degrees)

--plotfile PLOTFILE    filename for diagnostic plot, if desired

```

Here is a command line that I have used:

```

./connect_streaks.py --image_file i367226 --streak_file s367226 --image_name_in _immasked --image_name_out
_immasked2 --streak_name_in _streak --streak_name_out _streak2 --plotfile test.pdf -l test.log -v

```

The optional *plotfile* will hold a pdf illustration of the streaks on this exposure, such as this one:



The rectangles mark the corners of all the CCDs whose images were given to the routine in the *image_file*. The **input** streaks are marked as either

- **connected** (blue) are sets of >1 segments that line up and are assumed to represent a contiguous streak
- **edge** (magenta) streaks are close to and nearly parallel to the long edges of the CCDs (there are none on this exposure). These are probably spurious detections by immask and are ignored by *connect_streaks*.
- **unconnected** streaks (cyan) are singletons that do not line up with any other *immask* detections. We do not try to extend these into neighboring CCDs.

The connect streaks are extended across the full array, and their intersections with individual CCDs are marked as either:

- **new masked** streaks (red) are those that are within the *max_extrapolation* limit and will be masked on the outputs.
- **new skipped** streaks (dotted) are extrapolations of detected streaks that are too far from detections to be kept, and are *not* masked on outputs.

The command line arguments *streak_file* and *image_file* are text files holding the names of the FITS files holding the streak tables and immasked images, respectively (one input per line). The command line arguments *streak_name_in* and *streak_name_out* are used to generate the name for the new streak files, by replacing the former string with the latter one. If there is not exactly one match of *streak_name_in* in the input filename, an exception is generated. For example if these two arguments are `'_streak'` and `'_streaks'` respectively, then an input filename like

```
streaks/D00242750_i_c57_r2381p01_streak.fits
```

will be converted to

```
streaks/D00242750_i_c57_r2381p01_streak2.fits
```

and the output file will get this name. Note that I included the leading underscore to prevent a match with the leading directory name. The same system is applied to the image names.

The width of the new streak is taken to be the maximum of the widths of the detected segments from which it was made. If you want to grow or shrink this default, specify a positive or negative number in the *add_width* argument.

The standard *pixcorrect* mechanisms for logging, config files, and verbosity are used.