SAMOS PIPELINE STATUS

Dana Koeppe
MSU
2018-05-07

GITHUB-WRITTEN IN PYTHON 3

- https://github.com/SAMOS-Pipeline/SAMOS_pipeline_draft1
- Pipeline is on the GitHub page.
- Repository includes current SAMOS pipeline, Flame_DRP paper (Belli et. al., 2017), test data from LDSS3.
- I am following routines from the Flame pipeline (https://github.com/siriobelli/flame), written in IDL.

MAIN REDUCTION ROUTINES

- WhichFITSFiles
  - separates and sorts data based on header keys.
  - creates directories for storage during reduction.
- OverscanAndTrim
  - overscan/bias subtraction
  - trims edges of data
- NormDivFlats
  - master flat is created and divided out of the science frames.
  - creates thumbnail images of output data frames.
- OutlineSlits
  - Uses text files of approximate slit positions to create a region file to outline slits.
  - Cuts out region surrounding individual slits and creates new FITS image.

PYTHON MODULES

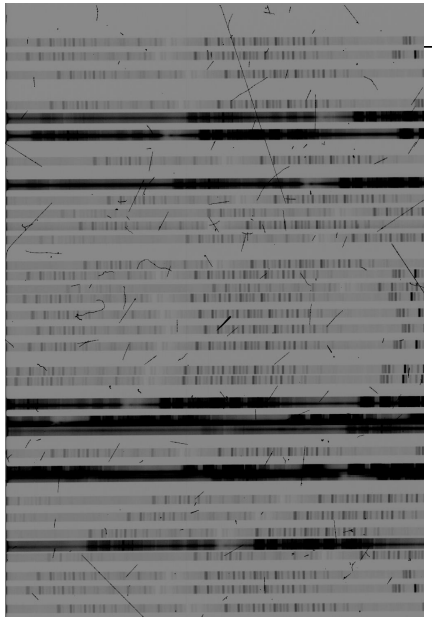
- SAMOSHelpers.py
  - checks file paths
  - function for making thumbnails
- Overscan.py
  - contains main routine for trimming and overscan subtraction
- FlatNorm.py
  - routine for generating the master flat
  - routine for dividing master flat from science frames
- cleaner.sh
  - run this to clear the directory of generated files and start over

```
$ ./WhichFITSFiles 2017-11-30 LMask2

Scanning LDSS3/2017-11-30

Found flats:
['ccd8161c1.fits', 'ccd8162c1.fits', 'ccd8163c1.fits']
Found lamps:
['ccd8159c1.fits', 'ccd8160c1.fits']
Found science:
['ccd8157c1.fits', 'ccd8158c1.fits']
With exposure times:
[ 900.  900.]
Keeping science:
['ccd8157c1.fits', 'ccd8158c1.fits']

$ ./OverscanAndTrim LMask2.db

 Working on LDSS3/2017-11-30/ccd8159c1.fits
 LDSS3/2017-11-30/ccd8159c1.fits
 Writing LMask2/LMask28159c1.fits
 LMask2/LMask28159c1.fits written.
 Working on LDSS3/2017-11-30/ccd8160c1.fits
 LDSS3/2017-11-30/ccd8160c1.fits
 Writing LMask2/LMask28160c1.fits
 LMask2/LMask28160c1.fits written.
 Working on LDSS3/2017-11-30/ccd8161c1.fits
 LDSS3/2017-11-30/ccd8161c1.fits
 Writing LMask2/LMask28161c1.fits
 LMask2/LMask28161c1.fits written.
```

LMask2/ and **LMask2.db** created

LMask2/jpeg also created

5

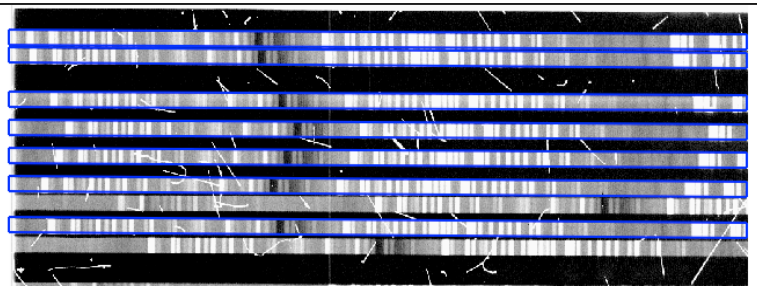


LMask2/jpeg for LMask28157c1.jpg

- jpeg directories created after `OverscanAndTrim` and `NormDivFlats`.
- This way the user can see progress without going into DS9.

6

OUTLINE SLITS



- Examples of reg_txt.reg output for LMask2 and a slit cutout image.
- Slit cutouts (Lmask2/flat_fielded/slit_cutouts) need fine-tuning.

7

FITS HEADER FOR SLIT INFORMATION

- Optimally, the SAMOS FITS headers will contain information about the slit positions.
  - This will help save time with slit identification and isolation.
- With the test data I'm using, I had to create text files of approximate slit edges based on the field mask image in DS9.

current slit cutout result for SAMOS

slit cutout from Flame

8

FITS HEADER FOR SLIT INFORMATION

- Example of FITS header for LUCI, read by Flame into pipeline data structures.

```
TGT08NAM= '124139  '           / name of target
MOS08SHA= 'STRAIGHT'           / shape of the slit
MOS08WAS=                   1. / width in arcsec
MOS08LAS=                  16. / length in arcsec
MOS08PA =                   0. / position angle on mask
MOS08RA = '100015.584'         / RA (hhmmss.ss)
MOS08DEC= '+021502.370'        / DEC (ddmmss.ss)
MOS08WMM=                0.601 / width in mm
MOS08LMM=                 9.61 / length in mm
MOS08XPO=              -52.586 / x pos in mm on mask
MOS08YPO=              -57.986 / y pos in mm on mask
TGT09NAM= '121033  '           / name of target
MOS09SHA= 'STRAIGHT'           / shape of the slit
MOS09WAS=                   1. / width in arcsec
MOS09LAS=                  12. / length in arcsec
MOS09PA =                   0. / position angle on mask
MOS09RA = '100014.625'         / RA (hhmmss.ss)
MOS09DEC= '+021251.050'        / DEC (ddmmss.ss)
MOS09WMM=                0.601 / width in mm
MOS09LMM=                7.207 / length in mm
MOS09XPO=              -43.952 / x pos in mm on mask
MOS09YPO=               20.886 / y pos in mm on mask

GRATPOS =                   0 / grating position
GRATNAME= 'G210 HiRes'        / grating name
GRATWLEN=        1.2699999809 / grating central wavelength
GRATORDE=                   4 / grating order
GRATVOLT=        2.5623989105 / grating tilt voltage
CAMPOS  =                   0 / camera position
CAMERA  = 'N1.8 Camera'       / camera name
CAMSCALE=               0.2496 / camera pixel scale [arcsec/pixel]
PIXSCALE=               0.2496 / camera pixel scale [arcsec/pixel]
```

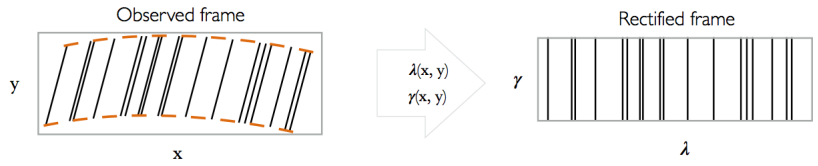

CURRENT

- Currently working on fine tuning OutlineSlits and Slit_id.py.
- Will organize code to use modules and saved data structures.
  - slit structure, instrument structure, etc…

```
; create 'empty' slit structure
slit = { $
    number:0, $
    name:'NN', $
    skip:0, $
    PA:!values.d_nan, $
    approx_bottom:0, $
    approx_top:0, $
    approx_target:0, $
    width_arcsec:!values.d_nan, $
    approx_R:0.0, $
    range_lambda0:[0.0, 0.0], $
    range_delta_lambda:[0.0, 0.0] }
```

```
; create the instrument structure - - - - - - - -
instrument = { $
  instrument_name: instrument_name, $
  grating: grating, $
  grating_order: grating_order, $
  central_wavelength: central_wavelength, $
  camera: camera, $
  pixel_scale: pixel_scale, $
  filter1: filter1, $
  filter2: filter2, $
  readnoise: readnoise, $
  gain: gain, $
  resolution_slit1arcsec: 0.0, $
  linearity_correction: linearity_correction, $
  trim_edges: 4, $
  default_badpixel_mask: default_badpixel_mask, $
  default_dark: default_dark, $
  default_pixelflat: default_pixelflat, $
  default_illumflat: default_illumflat, $
  default_arc: default_arc $
  }
```


NEXT MAIN STEP: COORDINATE TRANSFORMATION/RECTIFICATION



from Belli et. al. (2017)

- Need to correct distortions along wavelength and vertical axes.
- Timeline: I will aim to have a working version of a coordinate transformation module by the end of the summer.

GOODMAN DRP ANNOUNCEMENT

- SOAR just released the Goodman Data-Reductio pipeline on April 29.
- http://www.ctio.noao.edu/soar/content/goodman-data-reduction-pipeline
- Written in python3.6.
- Pipeline relies heavily on reading/updating FITS headers.

Keywords definition

- **GSP_COMB**: Image combined
- *GSP_VERS*: Pipeline Version
- *GSP_FNAM*: Original File Name
- *GSP_PATH*: Path to were the data was processed
- *GSP_TECH*: Observing Technique
- *GSP_DATE*: Processing date
- *GSP_OVER*: Overscan Correction
- *GSP_TRIM*: Default trim using TRIMSEC
- *GSP_SLIT*: Slit trim applied
- *GSP_BIAS*: Bias corrected (master bias file name??)
- *GSP_FLAT*: Master flat name
- *GSP_NORM*: Master Flat Normalization
- *GSP_COSM*: Cosmic ray rejection method (DCR, LaCosmic, None)
- *GSP_WRMS*: Wavelength solution Error RMS
- *GSP_WPOI*: Number of points used to calculate wavelength solution
- *GSP_WREJ*: Wavelength solution points rejected
- *GSP_IC00*: Image combined number where 00 goes from 01 to 99.

SUMMARY

- Uses python3.6 for data reduction.
- Working DRP code up to slit identification.
- User runs each step of the pipeline process, but eventually will be able to run all through one driver code.
- Need to know what information the DRP will be able to read from headers.
- Need to know what calibration files (darks, arcs, pixel flats, etc…) are needed for reduction.
- Main focus for this summer: coordinate transformation calculation.
- Future: wavelength calibration (line list?), sky subtraction, spectrum extraction.