Leibniz-Institut für
Astrophysik Potsdam

Data Calibration I

Imaging Instruments

Christoph Kuckein

Learning goals for today

1. Familiarize with ground-based imaging data
2. Get to know the basic data reduction concepts
3. Run the basic data reduction pipeline for two GREGOR imaging instruments
4. Obtain the wavelength array
5. … and learn some IDL tips

Examples of Imaging Instruments

Examples of Imaging Instruments

Telescope	Instruments
GREGOR	
SST	
DST	
DKIST	

Examples of Imaging Instruments

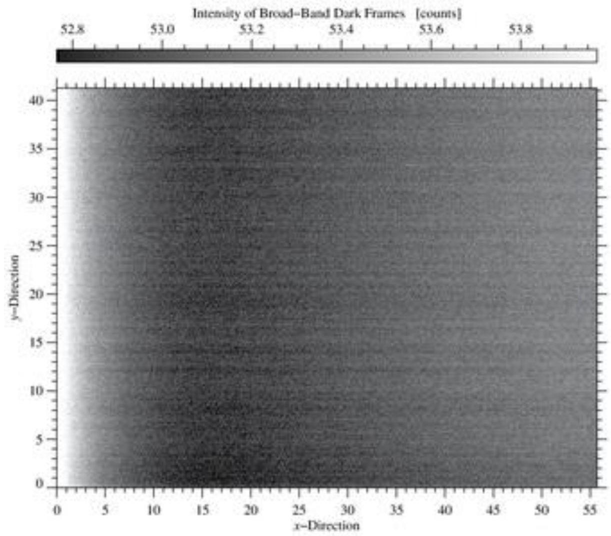
Telescope	Instruments
GREGOR	BIC (2014-2016) HiFI (2016 -) GFPI
SST	CRISP CHROMIS
DST	IBIS ROSA
DKIST	VTF VBI

Examples of Imaging Instruments

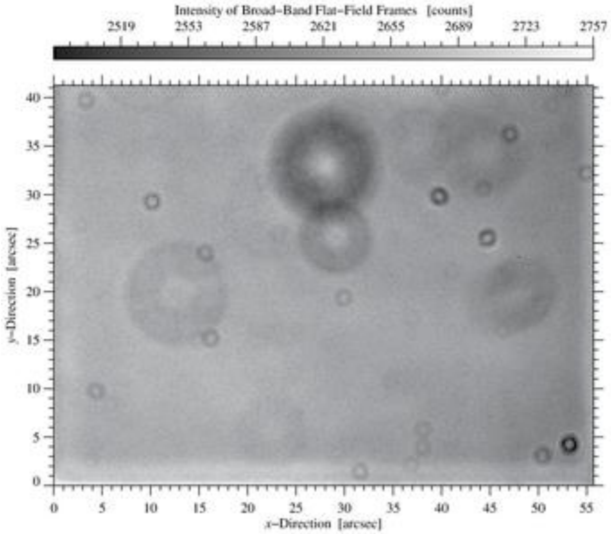
Telescope	Imaging	Spectroscopic Imaging
GREGOR	BIC (2014-2016) HiFI (2016 -)	GFPI
SST	-	CRISP CHROMIS
DST	ROSA	IBIS
DKIST	VBI	VTF

Basic Data Reduction

- Imaging data
  - Dark correction
    - Detectors accumulate counts even if no light is falling on the detector (because the detector is not at a temperature of 0 K)

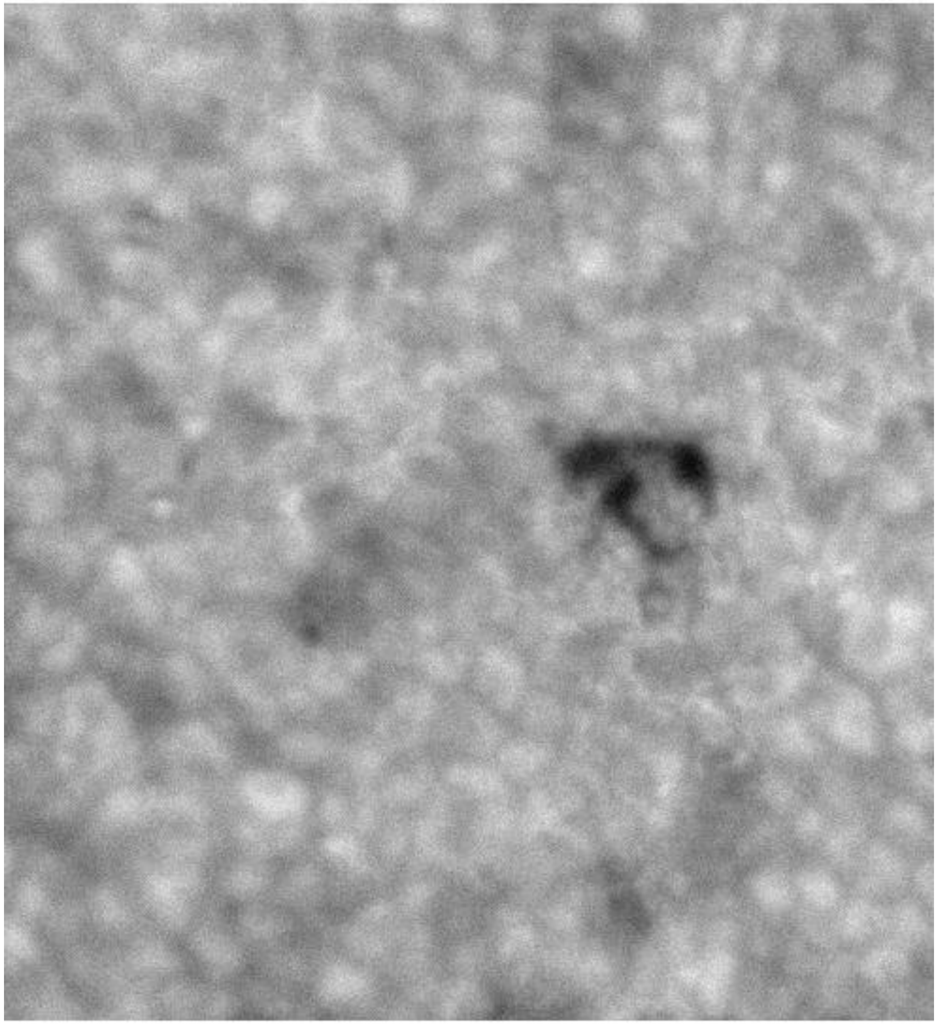


  - Flat-field correction
    - Each pixel of a detector (CCD, SCMOS,…) has a certain gain, which might be different from the neighbor pixel (also owing to dust along the light). The flat-field image shows the response of the detector to a uniform light source.

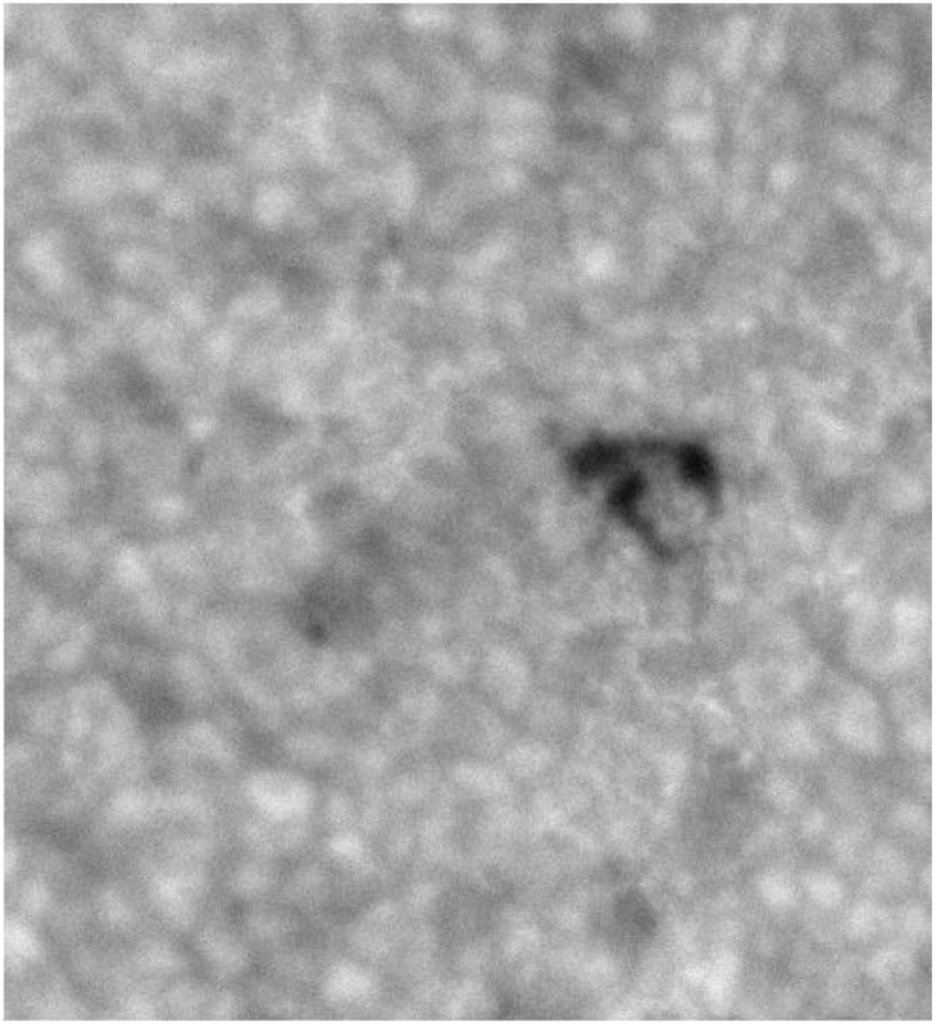


Flat-field correction

Find the differences?



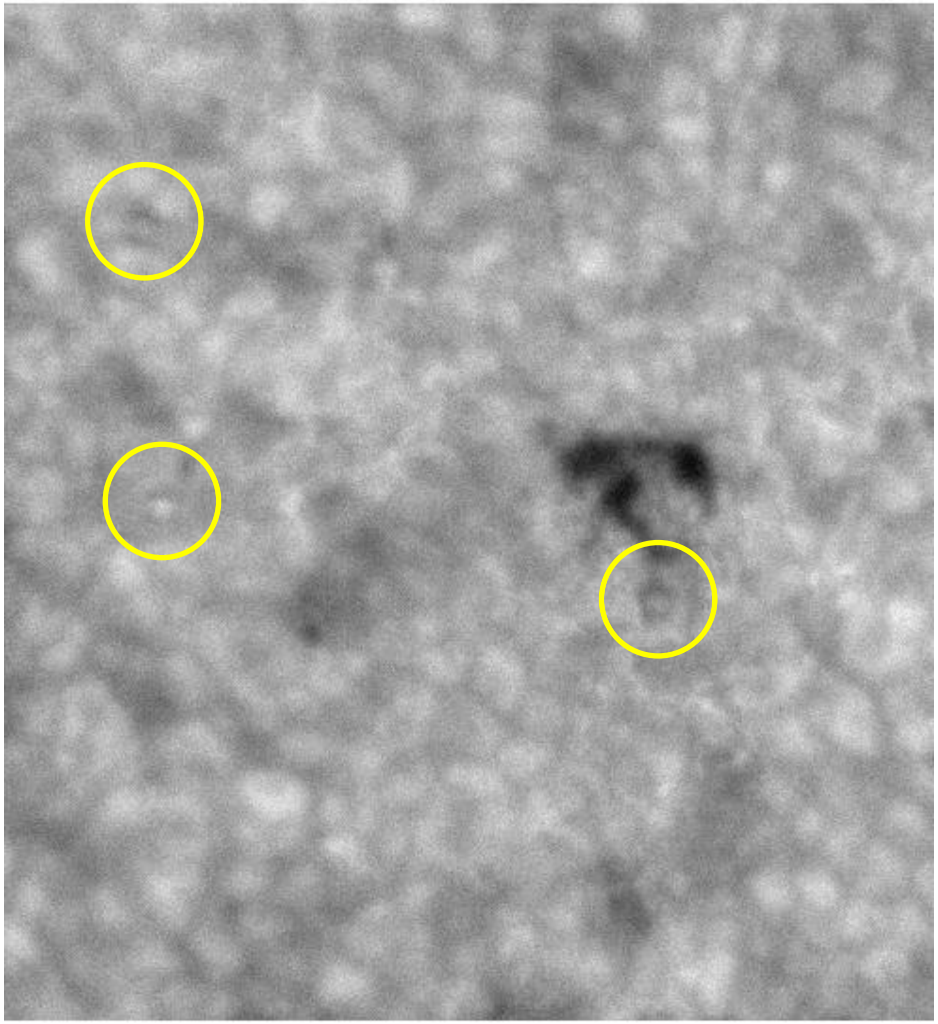
Without flat-field correction



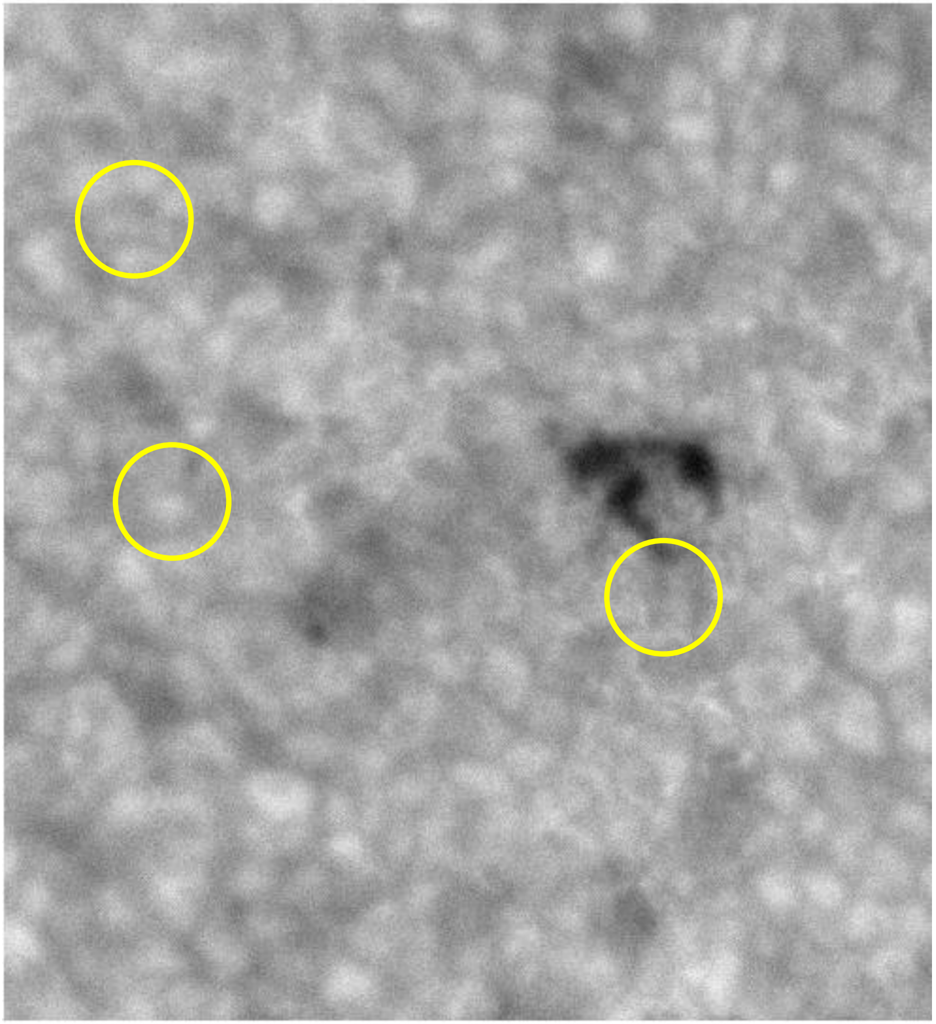
Flat-field corrected

GREGOR Fabry-Pérot Interferometer (GFPI; May 2018)

Flat-field correction



Without flat-field correction



Flat-field corrected

GREGOR Fabry-Pérot Interferometer (GFPI; May 2018)

Basic Data Reduction

- Imaging data
  - Dark correction
    - Detectors accumulate counts even if no light is falling on the detector (because the detector is not at a temperature of 0 K)
  - Flat-field correction
    - Each pixel of a detector (CCD, SCMOS,…) has a certain gain, which might be different from the neighbor pixel (also owing to dust along the light). The flat-field image shows the response of the detector to a uniform light source.

*next slide…

- Spectroscopic Imaging data
  - Dark correction
  - Flat-field correction
  - Prefilter curve correction
  - Instrument specific corrections*
- Spectropolarimetric Imaging data
  - Dark correction
  - Flat-field correction
  - Prefilter curve correction
  - Instrument specific corrections*
  - Polarimetric calibration to obtain Stokes profiles

Types of Fabry-Pérot Interferometers

- Different steps for data reduction

Spectroscopic Imaging	Type	(T)elecentric (C)ollimated
GFPI @ GREGOR	→	C
IBIS @ DST	→	C
CRISP @ SST	→	T*

→ Blueshift Correction (the wavelength of each pixel is field dependent)

→ Cavity and reflectivity corrections

*Telecentric setup: etalons are located close to the focal plane

Pipelines

Spectroscopic Imaging		
GFPI, HiFI @ GREGOR	→	sTools
CRISP @ SST	→	CRISPRED
CHROMIS @ SST	→	CHROMISRED

Pipelines

Spectroscopic Imaging		
GFPI, HiFI @ GREGOR	→	sTools
CRISP @ SST	→	CRISPRED
CHROMIS @ SST	→	CHROMISRED

CRISPRED / CHROMISRED ↓ https://dubshen.astro.su.se/wiki/index.php/SSTRED

Pipelines

https://ui.adsabs.harvard.edu/abs/2017IAUS..327...20K/abstract

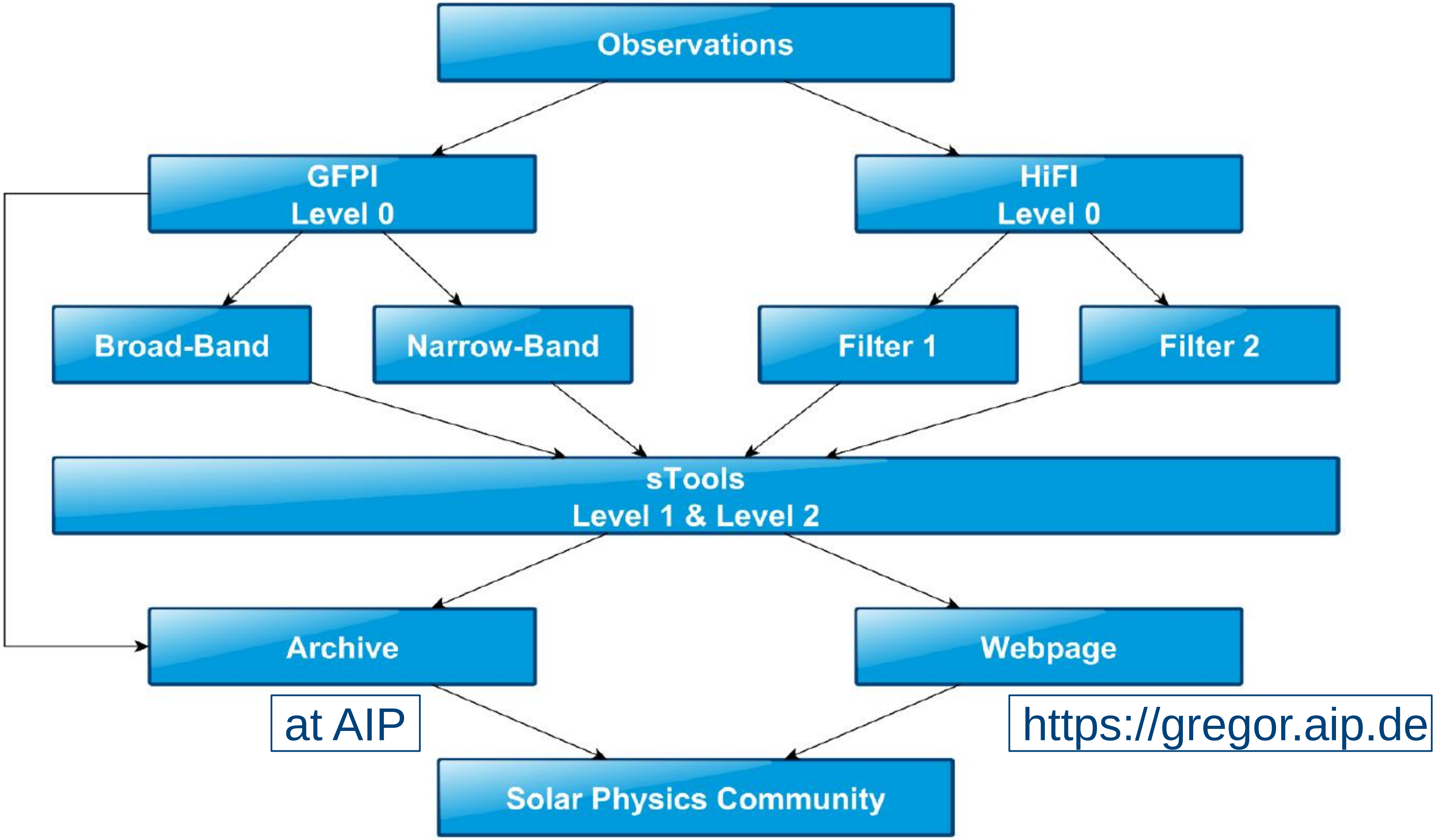
Spectroscopic Imaging		
GFPI, HiFI @ GREGOR	→	sTools
CRISP @ SST	→	CRISPRED
CHROMIS @ SST	→	CHROMISRED

A few words to sTools

- Under constant development at the Solar Group at Leibniz-Institute for Astrophysics Potsdam (AIP)
- Is more than a data reduction pipeline for HiFI and GFPI
- Has many IDL programs and tools to analyze your data
- Is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License
  - You are free to use and build your own programs upon sTools, but please acknowledge the creators (AIP)
- A very brief description can be read at the proceeding: *Kuckein, C., Denker, C., Verma, M., et al. 2017, Fine Structure and Dynamics of the Solar Atmosphere, IAU Symposium 327, page 20*

  https://ui.adsabs.harvard.edu/abs/2017IAUS..327...20K/abstract
- You have the latest version. Usually updates are uploaded at gregor.aip.de (members section; free registration required)
- Use it!

Imaging Instruments at GREGOR



Access to the archive

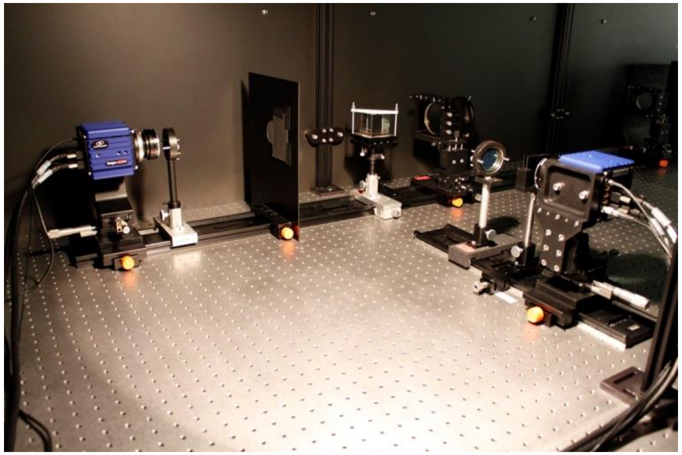
- Access is currently granted via ssh after registration
  - Register at https://gregor.aip.de
    - AIP approves registration → account is created on the website and home directory is setup on the data archive
  - ssh -p 2222 username@minos.aip.de

- Archive structure
  - Data is located at /store/gregor/…
  - Substructure: **HiFI**
    - Level0.0 (raw data; only exceptionally stored)
    - Level1.0 (reduced and frame selected)
    - Level2.0 (speckle restored data)
  - Substructure: **GFPI**
    - Level0.0 (raw data)
    - Level1.0 (reduced data)
    - Level2.0 (MOMFBD restored)

Data Reduction
Imaging

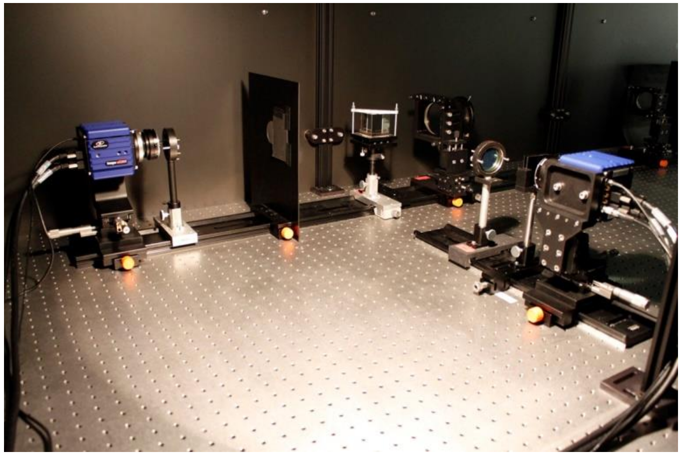
Data Reduction Imaging

- We will use as an example the current High-resolution Fast Imager (HiFI) instrument at GREGOR

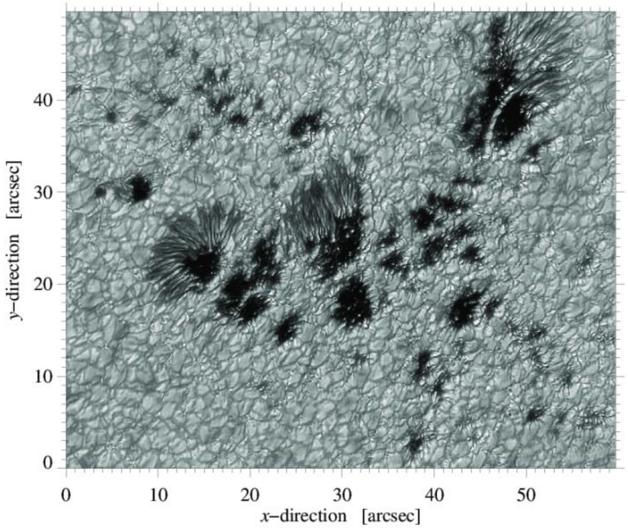


Data Reduction Imaging

- We will use as an example the current High-resolution Fast Imager (HiFI) instrument at GREGOR



- Characteristics
  - 2 synchronized sCMOS cameras
    - Images written into the same file
  - 2560 x 2160 pixels
  - 370 – 500 nm (together with GFPI)
  - Frame rate at full res: 50Hz
    - With smaller FOV: 100 - 135 Hz



Checklists…

- Checklist include recommendations from AIP after working with the cameras for 3 years
- Every day create a new project directory inside DAVIS
- If you decide to change the filters please use a new project
- <u>Pay attention to the suffix. They are crucial for the data reduction pipeline *sTools*</u>
- Every project should have the following files:
  - Flats (suffix: _ff)
  - Darks (_dk)
  - _df, _tg, _ph ,_pl, pg

Checklist HiFI at GREGOR

Recommended settings for science data (Suffix: _sd):

- **Full FOV**:.......................................# images: 500, Rate: 47 Hz
- **FOV 1280 X 1024**:
  (x= 640,1919; y= 568,1591).........# images: 500, Rate: 100 Hz
  (x= 640,1919; y= 566,1593).........# images: 500, Rate: 100 Hz
- **HD FOV 1920 X 1080**:
  (x= 302, 2239; y= 539, 1620).......# images: 500, Rate: 98 Hz

Note: Always use same exposure times in both cameras!

Calibration data:

---

1. flat field (Suffix: _ff)
(disk center; watch out for solar structures such as pores, sunspots, etc.)
Telescope moving speed as high as possible (100”). Big radius 100”.
Images: 2000
Rate: 10 Hz

Move up M2 to defocus telescope (arrow up 1.2 mm).

2. flat field defocused (Suffix: _df)
Images: 500
Rate: 10 Hz

3. target defocused (Suffix: _tg)
Images: 100
Rate: 49 Hz

4. pinholes defocused (small (_ph) and large (_pl))
Images: 100
Rate: 49 Hz
Move back M2 to original value (arrow down 1.2 mm).

5. dark (Suffix: _dk)

Checklists…

- Checklist include recommendations from AIP after working with the cameras for 3 years
- Every day create a new project directory inside DAVIS
- If you decide to change the filters please use a new project
- Pay attention to the suffix. They are crucial for the data reduction pipeline *sTools*
- Every project should have the following files:
  - Flats (suffix: _ff)
  - Darks (_dk)
  - _df, _tg, _ph ,_pl, pg

Checklist HiFI at GREGOR

Recommended settings for science data (Suffix: _sd):

- **Full FOV**:..# images: 500, Rate: 47 Hz
- **FOV 1280 X 1024**:
  (x= 640,1919; y= 568,1591).........# images: 500, Rate: 100 Hz
  (x= 640,1919; y= 566,1593).........# images: 500, Rate: 100 Hz
- **HD FOV 1920 X 1080**:
  (x= 302, 2239; y= 539, 1620).......# images: 500, Rate: 98 Hz

Note: Always use same exposure times in both cameras!

Calibration data:

---

1. flat field (Suffix: _ff)
(disk center; watch out for solar structures such as pores, sunspots, etc.)
Telescope moving speed as high as possible (100"). Big radius 100".
Images: 2000
Rate: 10 Hz

Move up M2 to defocus telescope (arrow up 1.2 mm).
2. flat field defocused (Suffix: _df)
Images: 500
Rate: 10 Hz

3. target defocused (Suffix: _tg)
Images: 100
Rate: 49 Hz

4. pinholes defocused (small (_ph) and large (_pl))
Images: 100
Rate: 49 Hz
Move back M2 to original value (arrow down 1.2 mm).

5. dark (Suffix: _dk)

Calibration Data

Running the data reduction pipeline

- The basic setup (https://cloud.aip.de/index.php/s/dtLdEyyc5yaE8q4)
  - A few SolarSoft routines are needed

  - HiFI and GFPI write images in a special file type: .im7
    - A special program is needed for GFPI data to read them: ReadIMX. This program needs to be compiled first on your computer

  - sTools → the configuration files: *stools_cfg_??????.pro*
    - *stools_cfg_setup.pro* (path to ReadIMX needs to be set)
    - *stools_cfg_observer.pro* (Observer name can be set)
    - *stools_cfg_filter.pro* (list of available filters for HiFI)

Running the data reduction pipeline

- sTools
  - *stools_hifi_stand_alone_nodisplay.pro*

All routines start with *stools*

Instrument

Pipeline

Running the data reduction pipeline

- sTools
  - OPEN: *stools_hifi_stand_alone_nodisplay.pro*

```
!QUIET = 0

sSetup = stools_cfg_setup()

sSetup.date = 'YYYYMMDD'   ; <--- input date
sSetup.dir  = '/instruments/hifi/Prj_Type=Imaging_Date=.../' ; <- input project
sSetup.out  = '/dat/USERNAME/hifi/level1/'+sSetup.date+'/'   ; <- create directo

sSetup.verbose = 0 ; to not display images

sTelescope = stools_cfg_telescope('GREGOR')

; Look up observername in stools_cfg_observer.pro
sObserver = stools_cfg_observer('CHRISTOPH')    ; <-- change observer

; Look up filter names in stools_cfg_filter.pro
sFilter1 = stools_cfg_filter('3968B')  ; <-- change filter tag 1
sFilter2 = stools_cfg_filter('4506')   ; <-- change filter tag 1

; Notes:
; - If no pinhole grid images were taken (extension 'pg'), then please comment
;   below the "align pinhole grid frames" routine
```

Running the data reduction pipeline

- sTools
  - OPEN: *stools_hifi_stand_alone_nodisplay.pro*

```
!QUIET = 0

sSetup = stools_cfg_setup()

sSetup.date = 'YYYYMMDD'  ; <--- input date
sSetup.dir  = '/instruments/hifi/Prj_Type=Imaging_Date=.../' ; <- input project
sSetup.out  = '/dat/USERNAME/hifi/level1/'+sSetup.date+'/'   ; <- create directo

sSetup.verbose = 0 ; to not display images

sTelescope = stools_cfg_telescope('GREGOR')

; Look up observername in stools_cfg_observer.pro
sObserver = stools_cfg_observer('CHRISTOPH')    ; <-- change observer

; Look up filter names in stools_cfg_filter.pro
sFilter1 = stools_cfg_filter('3968B')  ; <-- change filter tag 1
sFilter2 = stools_cfg_filter('4506')   ; <-- change filter tag 1

; Notes:
; - If no pinhole grid images were taken (extension 'pg'), then please comment
;   below the "align pinhole grid frames" routine
```

Where is the data and where you want to store it

Add your name In stools_cfg_observer.pro

Select filter

Running the data reduction pipeline

- sTools
  - OPEN: *stools_hifi_stand_alone_nodisplay.pro*

```
; dark frames ----------------------------------------
stools_hifi_dark_frames, sSetup, sTelescope, sObserver, $
                         sFilter1, sFilter2
timing[1] = systime(1)
print, ''
print, 'Dark frames [s]:                   ', string(FORMAT='(F8.2)', $
    (timing[1] - timing[0]))


; flat-field frames ----------------------------------------
stools_hifi_flat_field_frames, sSetup, sTelescope, sObserver, $
                               sFilter1, sFilter2
timing[2] = systime(1)
print, 'Flat-field frames frames [s]:      ', string(FORMAT='(F8.2)', $
    (timing[2] - timing[1]))


; Defocused flat-field frames ----------------------------------------
stools_hifi_defocused_flat_field_frames, sSetup, sTelescope, sObserver, $
                                         sFilter1, sFilter2
```

exclude

Running the data reduction pipeline

- *stools_hifi_stand_alone_nodisplay.pro*

```
; pinhole frames ----------------------------------------
stools_hifi_pinhole_frames, sSetup, sTelescope, sObserver, $
                            sFilter1, sFilter2
timing[4] = systime(1)
print, 'Pinhole frames frames [s]:         ', string(FORMAT='(F8.2)', $
    (timing[4] - timing[3]))


; target frames ----------------------------------------
stools_hifi_target_frames, sSetup, sTelescope, sObserver, $
                           sFilter1, sFilter2
timing[5] = systime(1)
print, 'Target frames frames [s]:          ', string(FORMAT='(F8.2)', $
    (timing[5] - timing[4]))


; align target frames ----------------------------------------
stools_hifi_target_frames_align, sSetup
timing[6] = systime(1)
print, 'Align target frames [s]:    ', string(FORMAT='(F8.2)', $
    (timing[6] - timing[5]))


; align pinhole grid frames ----------------------------------------
stools_hifi_pinhole_grid_frames_align, sSetup
timing[7] = systime(1)
print, 'Align pinhole grid frames [s]:    ', string(FORMAT='(F8.2)', $
    (timing[7] - timing[6]))
```

Running the data reduction pipeline

- *stools_hifi_stand_alone_nodisplay.pro*

```
; science data frames ----------------------------------------
stools_hifi_science_frames, sSetup, sTelescope, sObserver, $
                            sFilter1, sFilter2
```

nBest parameter: number of kept images (the others are lost)

Running the data reduction pipeline

❑ *stools_hifi_stand_alone_nodisplay.pro*

- ❑ Try to run now the HiFI pipeline:
  - Data from 2018 May 07
    - Set correct path for in and output data and filters
  - Needed libraries:
    - sTools
    - SolarSoft ssw/gen
    - Oslolib (for align.pro)
  - If everything is ready then open IDL and type:
    - .r *stools_hifi_stand_alone_nodisplay*

Running the data reduction pipeline

- sTools
  - *stools_hifi_stand_alone_nodisplay.pro*

```
!QUIET = 0

sSetup = stools_cfg_setup()

sSetup.date = 'YYYYMMDD'   ; <--- input date
sSetup.dir  = '/instruments/hifi/Prj_Type=Imaging_Date=.../' ; <- input project
sSetup.out  = '/dat/USERNAME/hifi/level1/'+sSetup.date+'/'   ; <- create directo

sSetup.verbose = 0 ; to not display images

sTelescope = stools_cfg_telescope('GREGOR')

; Look up observername in stools_cfg_observer.pro
sObserver = stools_cfg_observer('CHRISTOPH')     ; <-- change observer

; Look up filter names in stools_cfg_filter.pro
sFilter1 = stools_cfg_filter('3968B')  ; <-- change filter tag 1
sFilter2 = stools_cfg_filter('4506')   ; <-- change filter tag 1

; Notes:
; - If no pinhole grid images were taken (extension 'pg'), then please comment
;   below the "align pinhole grid frames" routine
```

Where is the data and where you want to store it

Add your name
In stools_cfg_observer.pro

Select filter

Output of the pipeline

```
-rw-rw---- 1 ckuckein gre  43M Jun 25  2018 calibration20180507.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_082542_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_082542_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_082706_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_082706_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_082846_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_082846_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_082908_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_082908_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_082931_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_082931_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_082953_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_082953_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_083016_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_083016_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_083038_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_083038_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_083101_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_083101_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_083123_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_083123_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_083145_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_083145_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_083208_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_083208_sd.sav
-rw-rw---- 1 ckuckein gre 2.1G Jun 25  2018 hifi_20180507_083231_sd.fts
-rw-rw---- 1 ckuckein gre 5.9K Jun 25  2018 hifi_20180507_083231_sd.sav
```

Alignment information of both cameras (IDL structure)

Fits files: 200 images
0,2,4,6, … n → filter 1
1,3,5,7, … n+1 → filter 2

Save files:
- mfgs_stat

(IDL structure with statistics of the quality of the image according to MFGS)

Output of the pipeline

```
IDL> restore, 'calibration20180507.sav',/v
% RESTORE: Portable (XDR) SAVE/RESTORE file.
% RESTORE: Save file written by ckuckein@perot, Mon Jun 25 11:12:33 2018.
% RESTORE: IDL version 8.1 (linux, x86_64).
% RESTORE: Restored variable: TARGET1.
% RESTORE: Restored variable: TARGET2.
% RESTORE: Restored variable: STARGET.
IDL> help, starget
** Structure <18b9848>, 6 tags, length=48, data length=42, refs=1:
   DIRECTION        INT              0
   DXY              FLOAT     Array[2]
   MAG              FLOAT          1.00570
   COR              DOUBLE         0.97710854
   PXY              DOUBLE    Array[2]
   THETA            FLOAT         0.477696
IDL> print, starget.pxy
       1304.1194       1107.2721
IDL> print, starget.dxy
      11.0690      6.47623
```

Alignment information of both cameras (IDL structure)

Fits files: 200 images

0,2,4,6, … n → filter 1

1,3,5,7, … n+1 → filter 2

To align image of camera 1 to camera 2:

img1new = rot(img1, theta, mag, pxy[0], pxy[1], CUBIC=-0.5, /PIVOT)

Tip: You can check the alignment blinking both images

blink, [0,1], 0.5

Output of the pipeline

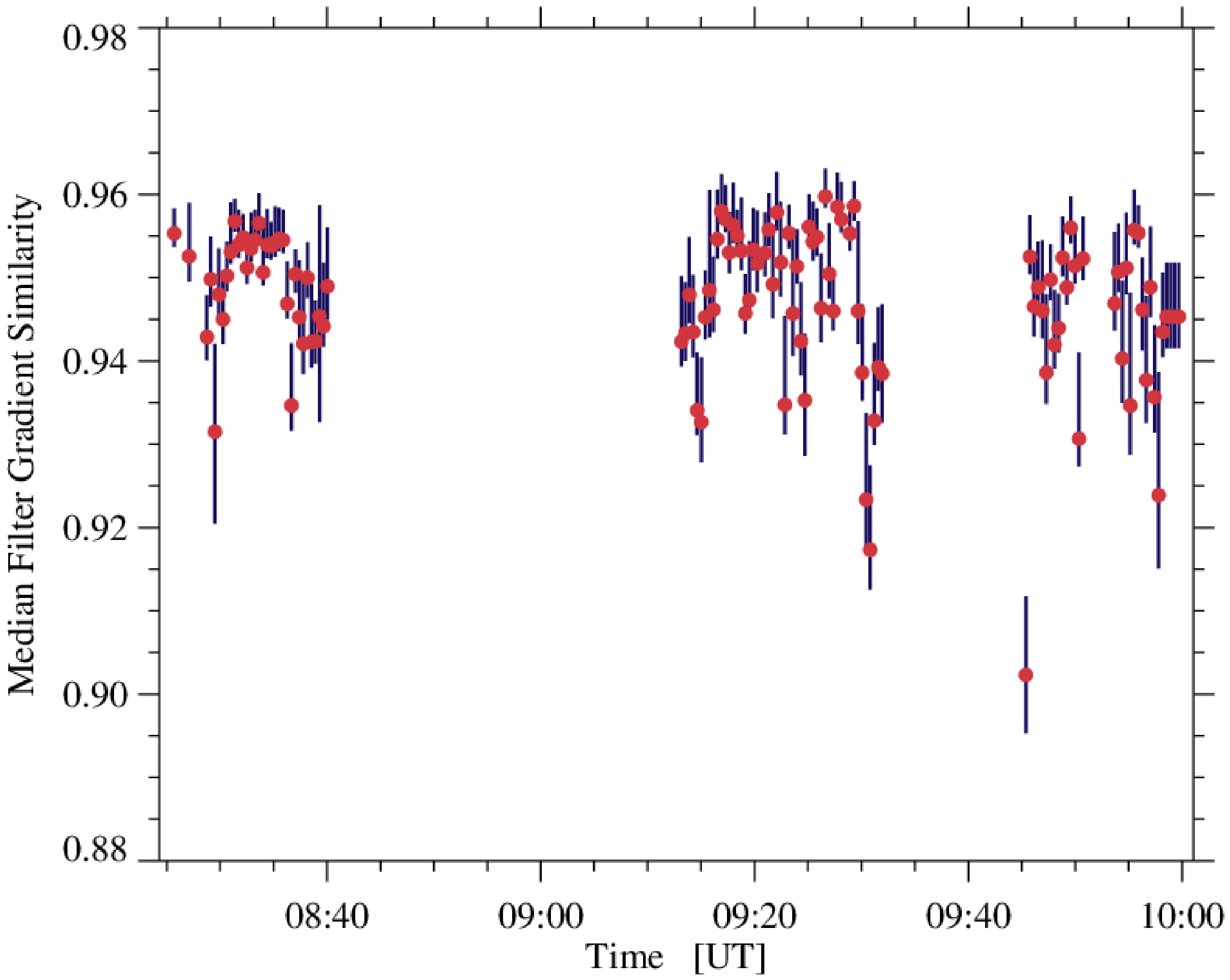
- Fits file
  - Images are not sorted by time, they are sorted by image quality
  - Image quality / seeing is computed using the MFGS value
    - Median Filter-Gradient Similarity
    - https://link.springer.com/article/10.1007%2Fs11207-015-0676-1
    - The higher the **mean** MFGS value, the better the seeing quality

Alignment information of both cameras (IDL structure)

Fits files: 200 images
0,2,4,6, … n → filter 1
1,3,5,7, … n+1 → filter 2

Save files:
- mfgs_stat
(IDL structure with statistics of the quality of the image according to MFGS)

MFGS



stools_median_filter_gradient_similarity.pro

Output of the pipeline

- ❑ Fits file
  - Images are not sorted by time, they are sorted by image quality
  - Image quality / seeing is computed using the MFGS value
    - Median Filter-Gradient Similarity
    - https://link.springer.com/article/10.1007%2Fs11207-015-0676-1
    - The higher the **mean** MFGS value, the better the seeing quality

- ❑ To read the fits files with IDL:

img = readfits('filename.fts', ext=n, header)

Alignment information of both cameras (IDL structure)

Fits files: 200 images
0,2,4,6, … n → filter 1
1,3,5,7, … n+1 → filter 2

Save files:
- mfgs_stat

(IDL structure with statistics of the quality of the image according to MFGS)

Output of the pipeline

Alignment information of both cameras (IDL structure)

- ❑ Fits file
  - ■ Images are not sorted by time, they are sorted by image quality
  - ■ Image quality / seeing is computed using the MFGS value
    - – Median Filter-Gradient Similarity
    - – https://link.springer.com/article/10.1007%2Fs11207-015-0676-1
    - – The higher the **mean** MFGS value, the better the seeing quality

Fits files: 200 images
0,2,4,6, … n → filter 1
1,3,5,7, … n+1 → filter 2

Save files:
- mfgs_stat
(IDL structure with statistics of the quality of the image according to MFGS)

- ❑ To read the fits files with IDL:

```
img = readfits('filename.fts', ext=n, header)
```

- ■ use a "for" loop to read the 100 images of each filter (200 images in total)

❑ To read the fits files with IDL:

img = readfits('filename.fts', **ext=0**, **header**)

```
SIMPLE  =                    T /
BITPIX  =                   16 /
NAXIS   =                    2 /
NAXIS1  =                 2560 /
NAXIS2  =                 2160 /
BZERO   =              0.00000 /
BSCALE  =              1.00000 /
EXTEND  =                    T /
DATE    = '2018-06-27T17:23:14' / File creation date [UTC]
DATE-OBS= '2018-05-07T09:58:56' / Data acquisition date [UTC]
INSTRUME= 'HiFI    '           / High-resolution Fast Imager
WAVELNTH=              396.800 / Observed wavelength [nm]
TELESCOP= 'GREGOR Solar Telescope' / Name of telescope
OBS-SITE= 'Observatorio del Teide' / Name of observatory
OBSERVER= 'Christoph Kuckein'  / Observer name
OBJECT  = 'Sun     '           /
BUFFNAME= 'B00086.im7'         / Original image file name
TIMEOFFS=              3000.06 / Time offset after first image [ms]
EXPTIME =              6.00000 / Exposure time [ms]
NFRAMES =                  100 / Number of saved frames
NIMGORIG=                  500 / Number of initially acquired images
NCAMERAS=                    2 / Camera 1 (even), Camera 2 (odd)
CAMERAID=                    1 / Camera ID
NAMECAM = 'Imager sCMOS'       / Name of cameras
PIXSIZE =              6.50000 / Size of camera pixel [microns]
DAVISVER= 'DaVis ver. 8.3.0'   / DaVis software version
```

```
MFGSMEAN=              0.93865 / MFGS mean (based on 4x4 binned i
MFGSSDEV=              0.12152 / MFGS standard deviation
MFGS10  =              0.82174 / MFGS 10th percentile
MFGS50  =              0.98692 / MFGS 50th percentile
MFGS90  =              0.99974 / MFGS 90th percentile
MFGSVAR =              0.01477 / MFGS variance
MFGSMDEV=              0.07369 / MFGS absolute mean deviation
MFGSSKEW=             -3.43847 / MFGS skewness
MFGSKURT=             13.82297 / MFGS kurtosis
DATAMIN =              0.52979 / Data minimum
DATAMEAN=              0.98757 / Data mean
DATAMEDN=              0.98150 / Data median
DATA10  =              1.06343 / Data 10th percentile
DATAMAX =              1.21863 / Data maximum
DATAVAR =              0.00361 / Data variance
DATASDEV=              0.06012 / Data standard deviation
DATAMDEV=              0.04506 / Data absolute mean deviation
DATASKEW=             -0.69839 / Data skewness
DATAKURT=              5.32838 / Data kurtosis
COMMENT
COMMENT   Contact information:
COMMENT
COMMENT   Carsten Denker (cdenker@aip.de)
COMMENT   Leibniz Institute for Astrophysics Potsdam (AIP)
COMMENT   An der Sternwarte 16
COMMENT   14482 Potsdam, GERMANY
O_BSCALE=    1.05108765638E-05 / Original BSCALE Value
O_BZERO =       0.874206572771 / Original BZERO Value
END
```

❑ To read the fits files with IDL:

img = readfits('filename.fts', **ext=0**, **header**)

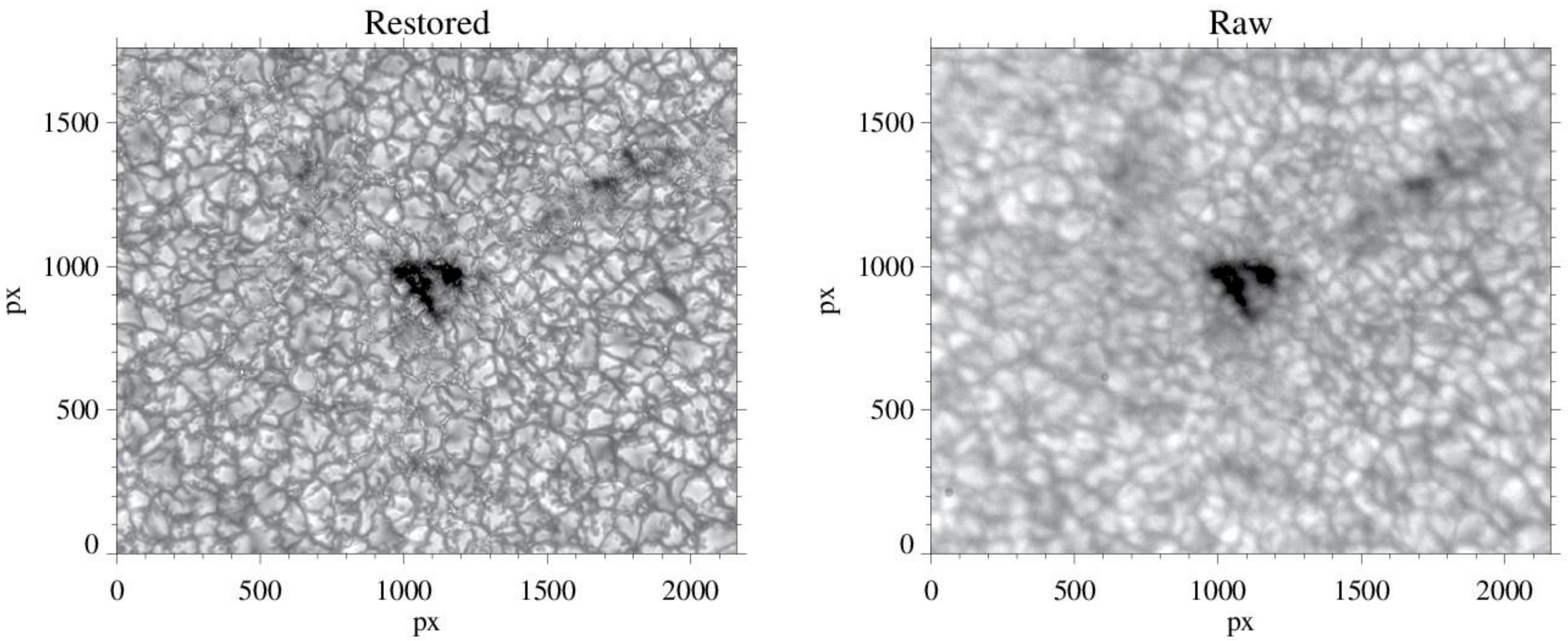
Important keywords!

```
SIMPLE  =                    T /
BITPIX  =                   16 /
NAXIS   =                    2 /
NAXIS1  =                 2560 /
NAXIS2  =                 2160 /
BZERO   =              0.00000 /
BSCALE  =              1.00000 /
EXTEND  =                    T /
DATE    = '2018-06-27T17:23:14' / File creation date [UTC]
DATE-OBS= '2018-05-07T09:58:56' / Data acquisition date [UTC]
INSTRUME= 'HiFI    '           / High-resolution Fast Imager
WAVELNTH=              396.800 / Observed wavelength [nm]
TELESCOP= 'GREGOR Solar Telescope' / Name of telescope
OBS-SITE= 'Observatorio del Teide' / Name of observatory
OBSERVER= 'Christoph Kuckein'  / Observer name
OBJECT  = 'Sun     '           /
BUFFNAME= 'B00086.im7'         / Original image file name
TIMEOFFS=              3000.06 / Time offset after first image [ms]
EXPTIME =              6.00000 / Exposure time [ms]
NFRAMES =                  100 / Number of saved frames
NIMGORIG=                  500 / Number of initially acquired images
NCAMERAS=                    2 / Camera 1 (even), Camera 2 (odd)
CAMERAID=                    1 / Camera ID
NAMECAM = 'Imager sCMOS'       / Name of cameras
PIXSIZE =              6.50000 / Size of camera pixel [microns]
DAVISVER= 'DaVis ver. 8.3.0'   / DaVis software version
```

```
MFGSMEAN=              0.93865 / MFGS mean (based on 4x4 binned i
MFGSSDEV=              0.12152 / MFGS standard deviation
MFGS10  =              0.82174 / MFGS 10th percentile
MFGS50  =              0.98692 / MFGS 50th percentile
MFGS90  =              0.99974 / MFGS 90th percentile
MFGSVAR =              0.01477 / MFGS variance
MFGSMDEV=              0.07369 / MFGS absolute mean deviation
MFGSSKEW=             -3.43847 / MFGS skewness
MFGSKURT=             13.82297 / MFGS kurtosis
DATAMIN =              0.52979 / Data minimum
DATAMEAN=              0.98757 / Data mean
DATAMEDN=              0.98150 / Data median
DATA10  =              1.06343 / Data 10th percentile
DATAMAX =              1.21863 / Data maximum
DATAVAR =              0.00361 / Data variance
DATASDEV=              0.06012 / Data standard deviation
DATAMDEV=              0.04506 / Data absolute mean deviation
DATASKEW=             -0.69839 / Data skewness
DATAKURT=              5.32838 / Data kurtosis
COMMENT
COMMENT  Contact information:
COMMENT
COMMENT  Carsten Denker (cdenker@aip.de)
COMMENT  Leibniz Institute for Astrophysics Potsdam (AIP)
COMMENT  An der Sternwarte 16
COMMENT  14482 Potsdam, GERMANY
O_BSCALE=    1.05108765638E-05 / Original BSCALE Value
O_BZERO =       0.874206572771 / Original BZERO Value
END
```

Output of the pipeline

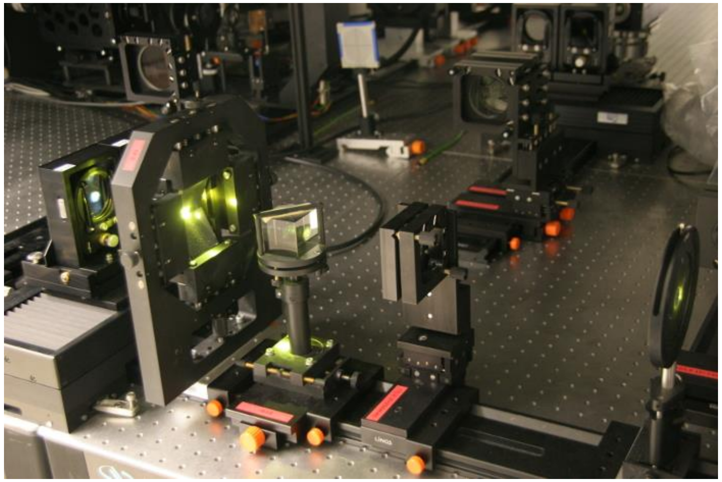
- Next steps to get the data science ready: **restoration**
  - Speckle restoration (e.g., KISIP example below)
    - *stools_hifi_science_frames_speckle.pro* (requires KISIP installation)
  - MFBD restoration



Data Reduction
Spectroscopic Imaging

Data Reduction GFPI

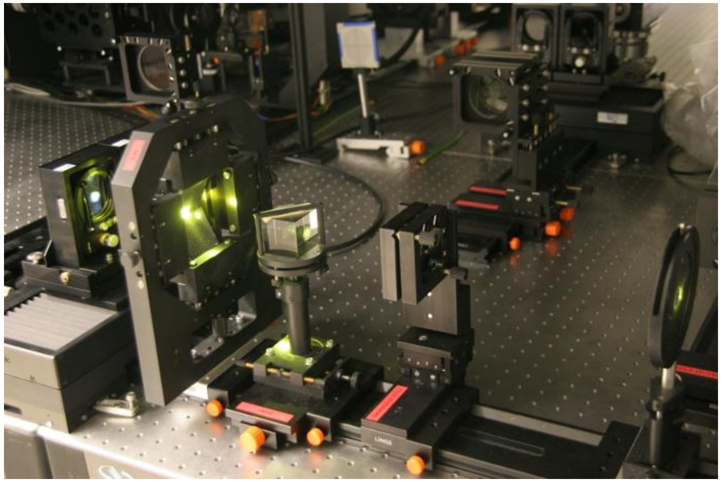
- We will use as an example the GFPI at GREGOR



- Characteristics (before 2019)
  - 2 synchronized CCD cameras
    - Broad-band camera (to have high S/N images)
    - Narrow-band camera to scan the line
  - 1376 x 1040 pixels
  - 530 – 650 nm (together with GRIS)

Data Reduction GFPI

- We will use as an example the GFPI at GREGOR

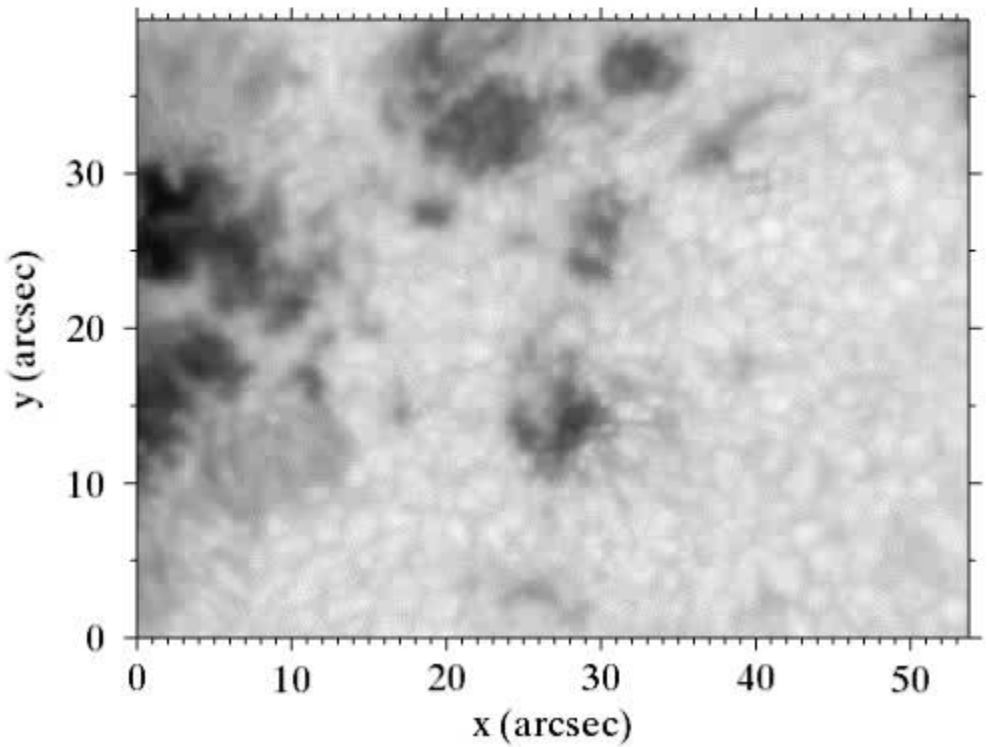


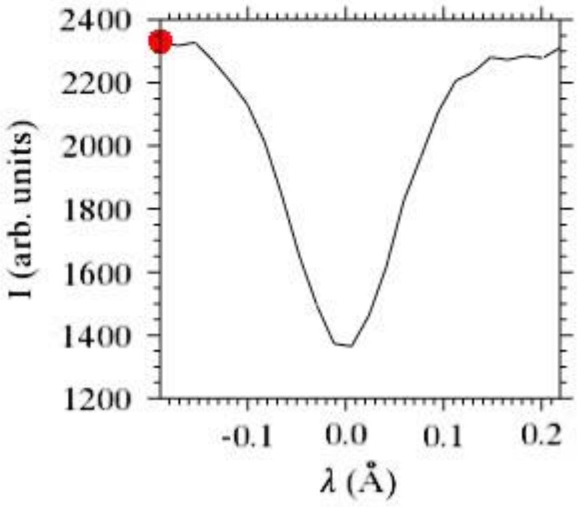
- Characteristics (before 2019)
  - 2 synchronized CCD cameras
    - Broad-band camera (to have high S/N images)
    - Narrow-band camera to scan the line
  - 1376 x 1040 pixels
  - 530 – 650 nm (together with GRIS)

Optimized for MOMFBD restoration

Data Reduction GFPI

- Scan along the line





Running the data reduction pipeline

- Make sure:
  - sTools is in your IDL path
  - SSW is in your IDL path
  - Oslolib is in your IDL path
  - The external program ReadIMX is in /home/username/tools/bin
    - Open stools_cfg_setup.pro and modify line 132 to make sure that the “bin” path point toward ReadIMX
  - For the prefilter curve you need an atlas profile
    - Workaround: fts_atlas_6563.sav will be handed out and needs to be manually changed in stools_gfpi_prefilter_curve.pro (line 202)

Running the data reduction pipeline

- sTools
  - OPEN: *stools_gfpi_stand_alone_solarnet.pro*

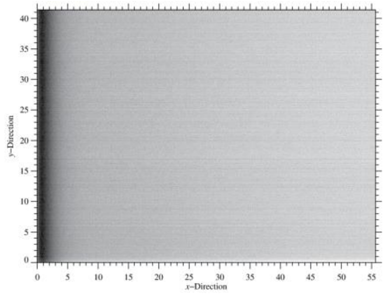
```
sSetup = stools_cfg_setup()

; Halpha
sSetup.date = '20140812'
;sSetup.dir =
;'/store/gregor/gfpi/level0.0/Prj_Type=FPI_Date=20140812_Time=071212/'
sSetup.dir = '/store/gregor/gfpi/level0.0/tmp/summerschool/Prj_Type=FPI_Date=20140812_Time=071212/'
sSpectralLines = stools_cfg_spectral_lines('6563')
sSetup.out ='/store/gregor/gfpi/level1.0/tmp/summerschool/20140812/'

sSetup.verbose = 1

sCamera = stools_cfg_camera('SENSICAM')
sCamera.nx = 688
sCamera.ny = 512
sCamera.ibx = 1  ; binning
sCamera.iby = 1
sCamera.scale = 0.0405

sTelescope = stools_cfg_telescope('GREGOR')
```

Change to your directories

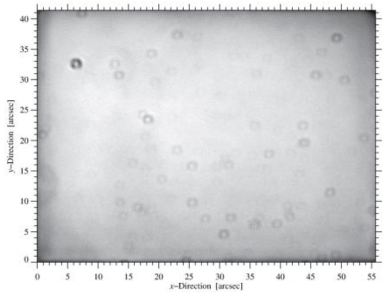
Running the data reduction pipeline

- *stools_gfpi_stand_alone_solarnet.pro*

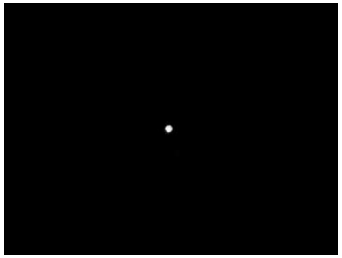
```
; dark frames
stools_gfpi_dark_frames, sCamera, sSetup
timing[1] = systime(1)
print, ''
print, 'Dark frames [s]:                    ', string(FORMAT='(F8.2)', $
    (timing[1] - timing[0]))
```



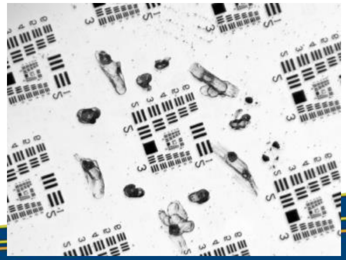
```
; flat-field frames
stools_gfpi_flat_field_frames, sCamera, sSetup
timing[2] = systime(1)
print, 'Flat-field frames frames [s]:       ', string(FORMAT='(F8.2)', $
    (timing[2] - timing[1]))
```



```
; pinhole frames
stools_gfpi_pinhole_frames, sCamera, sSetup
timing[3] = systime(1)
print, 'Pinhole frames frames [s]:          ', string(FORMAT='(F8.2)', $
    (timing[3] - timing[2]))
```



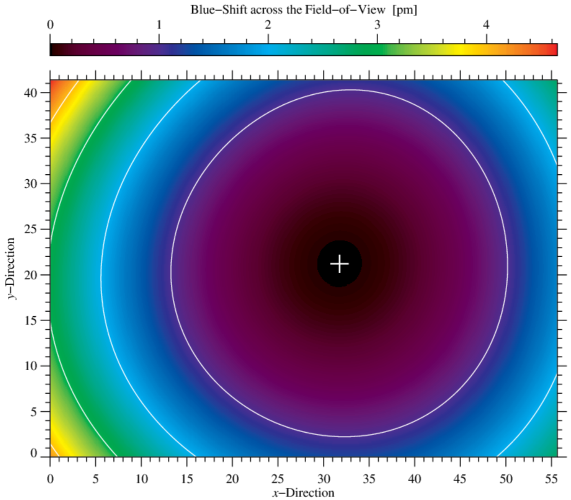
```
; target frames
stools_gfpi_target_frames, sCamera, sSetup
timing[4] = systime(1)
print, 'Target frames frames [s]:           ', string(FORMAT='(F8.2)', $
    (timing[4] - timing[3]))
```

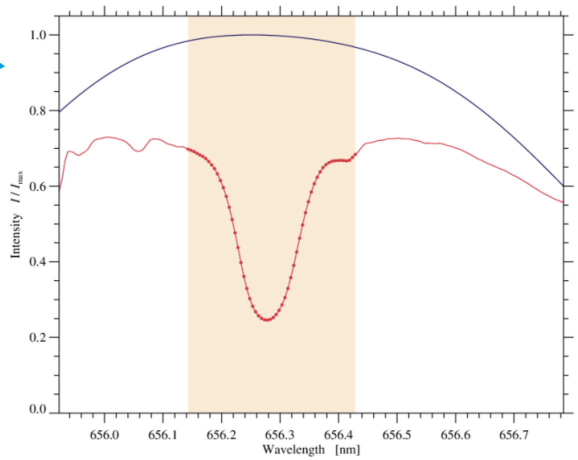


Running the data reduction pipeline

- *stools_gfpi_stand_alone_solarnet.pro*

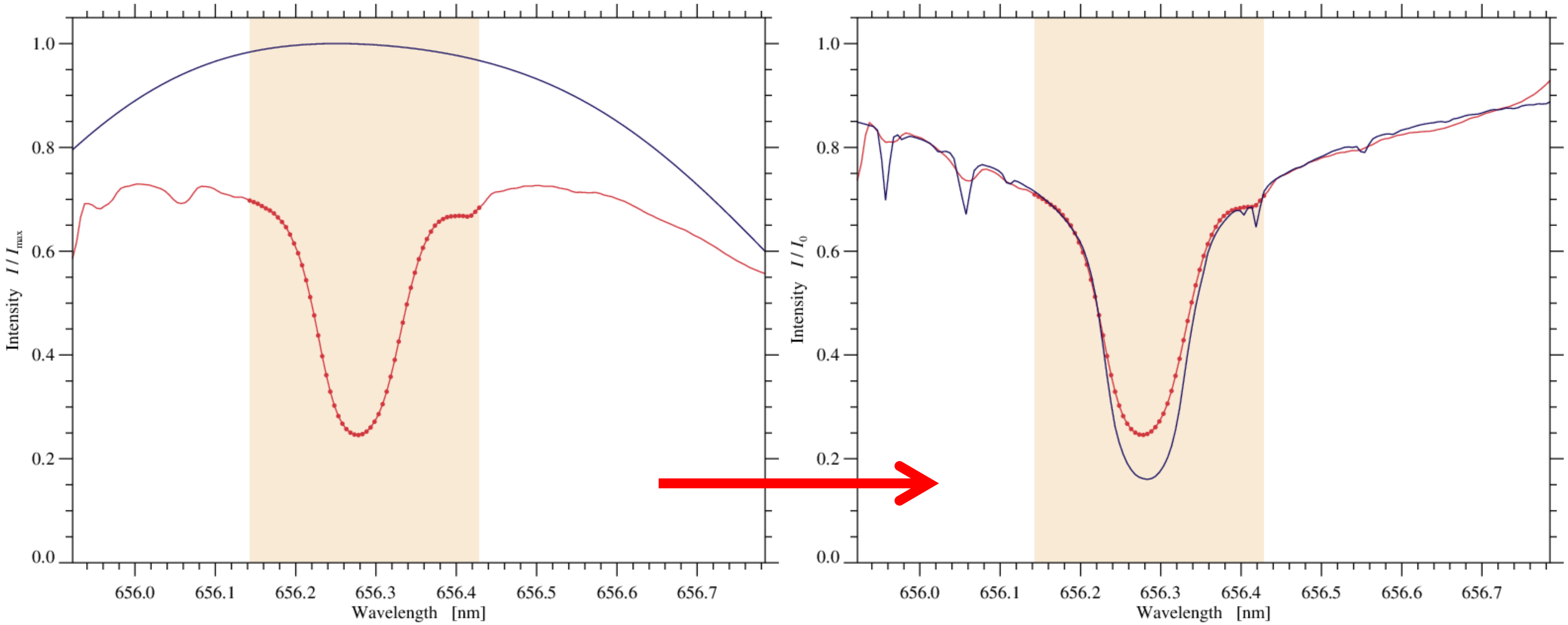
```
; align target frames
stools_gfpi_target_frames_align, sCamera, sSetup
timing[5] = systime(1)
print, 'Align target frames frames [s]:     ', string(FORMAT='(F8.2)', $
    (timing[5] - timing[4]))


; compute blue-shift
stools_gfpi_blue_shift, sCamera, sSetup, sSpectralLines
timing[6] = systime(1)
print, 'Compute blue-shift [s]:             ', string(FORMAT='(F9.3)', $
    (timing[6] - timing[5]))


; compute prefilter curve
stools_gfpi_prefilter_curve, sCamera, sSetup, sSpectralLines
timing[7] = systime(1)
print, 'Compute prefilter curve [s]:        ', string(FORMAT='(F9.3)', $
    (timing[7] - timing[6]))

; science data frames
stools_gfpi_science_frames, sCamera, sSetup, /masterflat3d
timing[8] = systime(1)
print, 'Science data frames [h]:            ', string(FORMAT='(F8.2)', $
    (timing[8] - timing[7])/3600.)
```





Running the data reduction pipeline

- *stools_gfpi_stand_alone_solarnet.pro*




```
; compute prefilter curve
stools_gfpi_prefilter_curve, sCamera, sSetup, sSpectralLines
timing[7] = systime(1)
print, 'Compute prefilter curve [s]:       ', string(FORMAT='(F9.3)', $
    (timing[7] - timing[6]))
```

Output of the pipeline

```
total 178G
-rw-rw-r-- 1 ckuckein gre 173M Jul 24 18:10 calibration20140812.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 09:46 gfpi20140812_075543.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 09:53 gfpi20140812_075633.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:01 gfpi20140812_075722.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:08 gfpi20140812_075812.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:15 gfpi20140812_075900.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:22 gfpi20140812_075948.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:29 gfpi20140812_080036.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:37 gfpi20140812_080126.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:44 gfpi20140812_080215.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:51 gfpi20140812_080305.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:56 gfpi20140812_080355.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:01 gfpi20140812_080445.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:06 gfpi20140812_080534.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:10 gfpi20140812_080622.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:15 gfpi20140812_080710.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:20 gfpi20140812_080759.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:25 gfpi20140812_080847.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:29 gfpi20140812_080935.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:34 gfpi20140812_081023.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:39 gfpi20140812_081110.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:43 gfpi20140812_081158.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:48 gfpi20140812_081246.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:53 gfpi20140812_081335.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:58 gfpi20140812_081424.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:02 gfpi20140812_081513.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:07 gfpi20140812_081601.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:12 gfpi20140812_081649.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:16 gfpi20140812_081737.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:21 gfpi20140812_081824.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:26 gfpi20140812_081913.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:31 gfpi20140812_082000.sav
```

Output of the pipeline

```
total 178G
-rw-rw-r-- 1 ckuckein gre 173M Jul 24 18:10 calibration20140812.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 09:46 gfpi20140812_075543.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 09:53 gfpi20140812_075633.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:01 gfpi20140812_075722.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:08 gfpi20140812_075812.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:15 gfpi20140812_075900.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:22 gfpi20140812_075948.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:29 gfpi20140812_080036.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:37 gfpi20140812_080126.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:44 gfpi20140812_080215.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:51 gfpi20140812_080305.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 10:56 gfpi20140812_080355.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:01 gfpi20140812_080445.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:06 gfpi20140812_080534.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:10 gfpi20140812_080622.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:15 gfpi20140812_080710.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:20 gfpi20140812_080759.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:25 gfpi20140812_080847.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:29 gfpi20140812_080935.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:34 gfpi20140812_081023.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:39 gfpi20140812_081110.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:43 gfpi20140812_081158.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:48 gfpi20140812_081246.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:53 gfpi20140812_081335.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 11:58 gfpi20140812_081424.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:02 gfpi20140812_081513.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:07 gfpi20140812_081601.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:12 gfpi20140812_081649.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:16 gfpi20140812_081737.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:21 gfpi20140812_081824.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:26 gfpi20140812_081913.sav
-rw-rw-r-- 1 ckuckein gre 1.4G Jul 25 12:31 gfpi20140812_082000.sav
```

Output of the pipeline

- calibrationYYYYMMDD.sav

```
BLUESHIFT         FLOAT     = Array[688, 512]
BLUESHIFTFIT      FLOAT     = Array[688, 512]
DARKBB            FLOAT     = Array[688, 512, 2]
DARKNB            FLOAT     = Array[688, 512, 2]
FLATBB            FLOAT     = Array[688, 512]
FLATNB            FLOAT     = Array[688, 512]
FLATNBMASTER      FLOAT     = Array[688, 512, 58]
FLATNBSCAN        FLOAT     = Array[688, 512, 58]
LAMBDA_GFPI       FLOAT     = Array[173]
NACC              INT       = Array[1]
NINC              INT       = Array[1]
NPOS              INT       = Array[1]
PINHBB            FLOAT     = Array[688, 512]
PINHNB            FLOAT     = Array[688, 512]
POSITIONS         INT       = Array[58]
POSITIONSLONG     INT       = Array[173]
PREFILTER         FLOAT     = Array[2753]
PREFILTER_COEF    FLOAT     = Array[6]
SPECTRUM_FTS      FLOAT     = Array[173]
SPECTRUM_GFPI     FLOAT     = Array[173]
SSPECTRALLINES    STRUCT    = -> <Anonymous> Array[1]
STARGET           STRUCT    = -> <Anonymous> Array[1]
SVN               STRING    = Array[5]
TARGBB            FLOAT     = Array[688, 512]
TARGNB            FLOAT     = Array[688, 512]
```

Output of the pipeline

- calibrationYYYYMMDD.sav

```
BLUESHIFT        FLOAT     = Array[688, 512]
BLUESHIFTFIT     FLOAT     = Array[688, 512]
DARKBB           FLOAT     = Array[688, 512, 2]
DARKNB           FLOAT     = Array[688, 512, 2]
FLATBB           FLOAT     = Array[688, 512]
FLATNB           FLOAT     = Array[688, 512]
FLATNBMASTER     FLOAT     = Array[688, 512, 58]
FLATNBSCAN       FLOAT     = Array[688, 512, 58]
LAMBDA_GFPI      FLOAT     = Array[173]
NACC             INT       = Array[1]
NINC             INT       = Array[1]
NPOS             INT       = Array[1]
PINHBB           FLOAT     = Array[688, 512]
PINHNB           FLOAT     = Array[688, 512]
POSITIONS        INT       = Array[58]
POSITIONSLONG    INT       = Array[173]
PREFILTER        FLOAT     = Array[2753]
PREFILTER_COEF   FLOAT     = Array[6]
SPECTRUM_FTS     FLOAT     = Array[173]
SPECTRUM_GFPI    FLOAT     = Array[173]
SSPECTRALLINES   STRUCT    = -> <Anonymous> Array[1]
STARGET          STRUCT    = -> <Anonymous> Array[1]
SVN              STRING    = Array[5]
TARGBB           FLOAT     = Array[688, 512]
TARGNB           FLOAT     = Array[688, 512]
```

- Blueshift correction (to apply after image restoration by MOMFBD (BLUESHIFTFIT)
- Number of images per wavelength position (accumulations) (NACC)
- Step size (unit: positions) (NINC)
- Wavelength points along the spectral line (POSITIONS)
- Coefficients to correct the prefilter curve (PREFILTER_COEF)

Output of the pipeline

- Observations: gfpiYYYYMMDD_hhmmss.sav
  - IDL → restore, ‘filename’, /v

```
SCANBB          FLOAT     = Array[688, 512, 464]
SCANNB          FLOAT     = Array[688, 512, 464]
SCANNBMEAN      FLOAT     = Array[688, 512, 58]
SPHYSMAPS       STRUCT    = -> <Anonymous> Array[1]
SSCANTABLE      STRUCT    = -> <Anonymous> Array[1]
```

  - scanBB: all reduced broad-band images
  - scanNB: all reduced narrow-broad images

} Used for MOMFBD

Output of the pipeline

- Observations: gfpiYYYYMMDD_hhmmss.sav
  - IDL → restore, ‘filename’, /v

```
SCANBB          FLOAT     = Array[688, 512, 464]
SCANNB          FLOAT     = Array[688, 512, 464]
SCANNBMEAN      FLOAT     = Array[688, 512, 58]
SPHYSMAPS       STRUCT    = -> <Anonymous> Array[1]
SSCANTABLE      STRUCT    = -> <Anonymous> Array[1]
```

- scanBB: all reduced broad-band images
- scanNB: all reduced narrow-broad images

} Used for MOMFBD

- scanNBmean: scanNB but images of the same wavelength position were summed up (averaged). In other words, the accumulated images at each wavelength position are averaged to increase the S/N. In addition, blueshift and prefilter curve were corrected. This data can be used to make science (but be aware that no image restoration was done yet!)

Output of the pipeline

- Observations: gfpiYYYYMMDD_hhmmss.sav
  - IDL → restore, 'filename', /v

```
SCANBB          FLOAT     = Array[688, 512, 464]
SCANNB          FLOAT     = Array[688, 512, 464]
SCANNBMEAN      FLOAT     = Array[688, 512, 58]
SPHYSMAPS       STRUCT    = -> <Anonymous> Array[1]
SSCANTABLE      STRUCT    = -> <Anonymous> Array[1]
```

  - scanBB: all reduced broad-band images
  - scanNB: all reduced narrow-broad images

  } Used for MOMFBD

  - scanNBmean: scanNB but images of the same wavelength position were summed up (averaged). In other words, the accumulated images at each wavelength position are averaged to increase the S/N. In addition, blueshift and prefilter curve were corrected. This data can be used to make science (but be aware that no image restoration was done yet!)

Output of the pipeline

- Observations: gfpiYYYYMMDD_hhmmss.sav
  - IDL → restore, ‘filename’, /v

```
SCANBB          FLOAT     = Array[688, 512, 464]
SCANNB          FLOAT     = Array[688, 512, 464]
SCANNBMEAN      FLOAT     = Array[688, 512, 58]
SPHYSMAPS       STRUCT    = -> <Anonymous> Array[1]
SSCANTABLE      STRUCT    = -> <Anonymous> Array[1]
```

  - sPhysMaps: quicklook data such as intensity, equivalent width, LOS velocities, etc. Cannot be used for science, but gives an impression of the content and interesting features in the data

Output of the pipeline

- ❑ Observations: gfpiYYYYMMDD_hhmmss.sav
  - sPhysMaps (only quicklook data!)

```
IDL> help, SPHYSMAPS,/str
** Structure <2360ee8>, 18 tags, length=17266120, data length=17266120, refs=1:
   CWL              FLOAT          656.281
   CHI2             FLOAT     Array[688, 512]
   EQUIV            FLOAT     Array[688, 512]
   GRAN             BYTE      Array[688, 512]
   FWHM             FLOAT     Array[688, 512]
   IMGBB            FLOAT     Array[688, 512]
   IMGNB            FLOAT     Array[688, 512]
   I_CORE           FLOAT     Array[688, 512]
   I_LOS            FLOAT     Array[688, 512]
   INTQS            FLOAT         0.736049
   MFGS             FLOAT     Array[464]
   MFGS_GRAN        FLOAT     Array[464]
   MFGS2D           FLOAT     Array[688, 512]
   RMSBB            FLOAT     Array[464]
   V_CORE           FLOAT     Array[688, 512]
   V_LOS            FLOAT     Array[688, 512]
   V_LPFF           FLOAT     Array[688, 512]
   SUMBB            FLOAT     Array[688, 512]
```

Doppler shifts computed from the scanNBmean data cube using 3 different meth.: *Polynomial fit to the core, Gaussian fit to the line, & LPFF technique*

Output of the pipeline

- Observations: gfpiYYYYMMDD_hhmmss.sav
  - IDL → restore, ‘filename’, /v

```
SCANBB          FLOAT     = Array[688, 512, 464]
SCANNB          FLOAT     = Array[688, 512, 464]
SCANNBMEAN      FLOAT     = Array[688, 512, 58]
SPHYSMAPS       STRUCT    = -> <Anonymous> Array[1]
SSCANTABLE      STRUCT    = -> <Anonymous> Array[1]
```

  - sScanTable: information about the parameters of how the GFPI scanned the spectral line

Output of the pipeline

- Observations: gfpiYYYYMMDD_hhmmss.sav
  - sScanTable

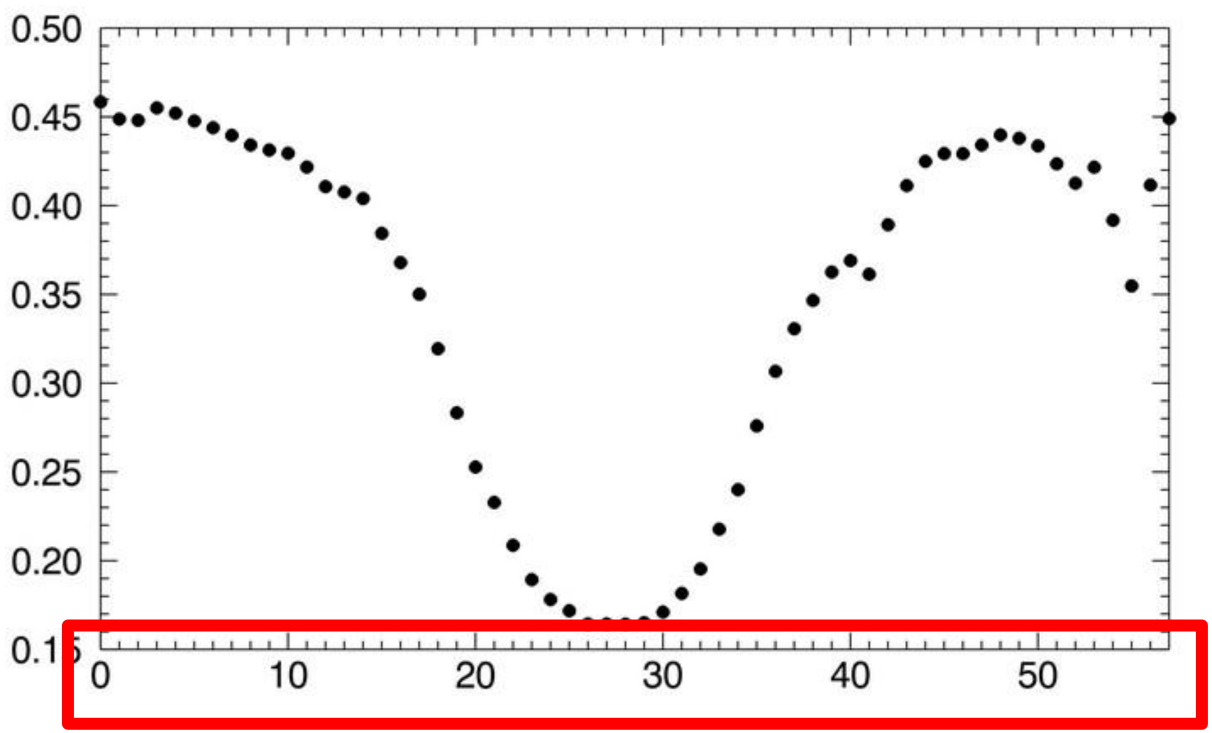
```
IDL> help, SSCANTABLE,/str
** Structure <2259938>, 17 tags, length=224, data length=212, refs=1:
   FILE            STRING    'Cam_Date=140812_Time=075540_sd.set'
   TYPE            STRING    'sd'
   ISEQUENCE       INT              0
   NROWS           LONG                 1
   IDFILTERNB      INT              1
   FILTERNB        FLOAT          6563.00
   IDFILTERBB      INT              0
   FILTERBB        FLOAT          6563.00
   START           INT           -448
   INCREMENT       INT             16
   FINAL           INT            464
   ACCUMULATIONS   INT              8
   POSITIONS       INT       Array[58]
   NIMAGES         INT            472
   EXPTIME         LONG             12000
   DATE            STRING    '2014-08-12'
   TIME            STRING    '07:55:43.597'
```

Exposure time in milliseconds

Recording time

To make science we need…

…a wavelength array!



To make science we need…

…a wavelength array!

```
IDL> help, SSCANTABLE,/str
** Structure <2259938>, 17 tags, length=224, data le
   FILE            STRING    'Cam_Date=140812_Time=(
   TYPE            STRING    'sd'
   ISEQUENCE       INT              0
   NROWS           LONG                 1
   IDFILTERNB      INT              1
   FILTERNB        FLOAT          6563.00
   IDFILTERBB      INT              0
   FILTERBB        FLOAT          6563.00
   START           INT           -448
   INCREMENT       INT             16
   FINAL           INT            464
   ACCUMULATIONS   INT              8
   POSITIONS       INT       Array[58]
   NIMAGES         INT            472
   EXPTIME         LONG             12000
   DATE            STRING    '2014-08-12'
   TIME            STRING    '07:55:43.597'
```

?

Discuss in pairs and come up with an idea to calculate the wavelength array

Wavelength array (solution)

```
cwl  = 6562.8                                   ; units: Å (from NIST for example)
disp = stools_gfpi_dispersion(cwl*1e-10)        ; spectral dispersion
disp = disp *1d13                               ; Å → mÅ
```

```
positions     = sScantable.positions
nInc          = sScantable.increment
npositions    = n_elements(positions)
```

Find the wavelength at rest selecting an area of quiet Sun and fitting the profile (e.g., gaussfit in IDL)

```
lambda =  (findgen(npositions) - pos_core_QS) * disp * nInc + cwl   ; [mA]
```