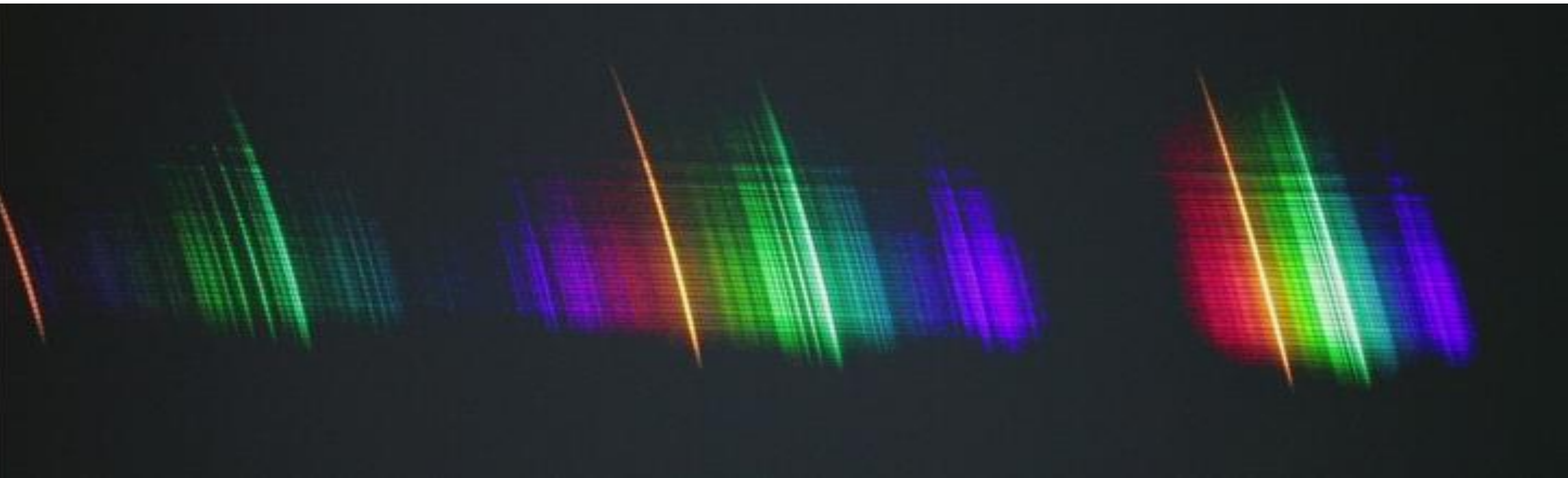
Fachgruppe Meteorastronomie

Meteor Spectroscopy,
Bellinzona, 12. Oct. 2019

Martin Dubs, images by Koji Maeda

SAG, FMA

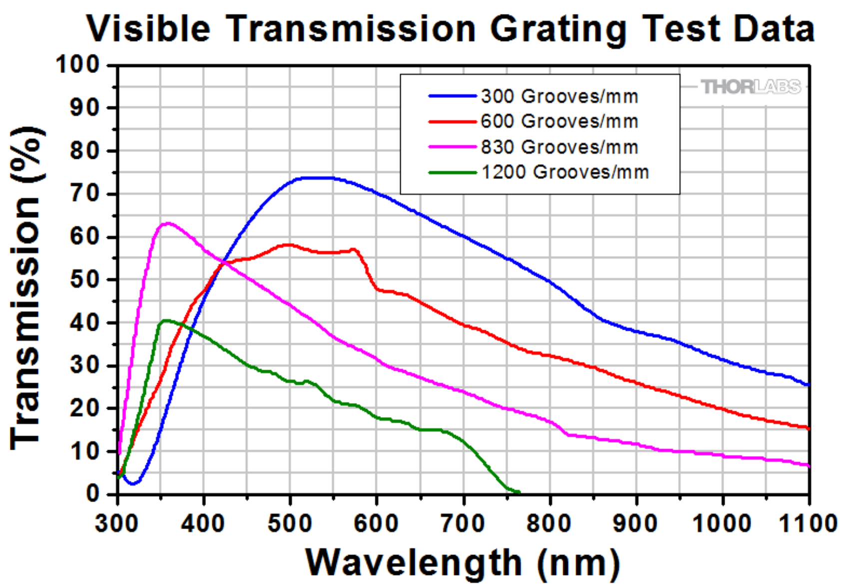


Content

- Hard- Software
- Wavelength calibration
- Spectrum extraction, old method
- Processing pipeline with Python
- Results
- Summary
- Live demo, Python pipeline m_pipe62.py

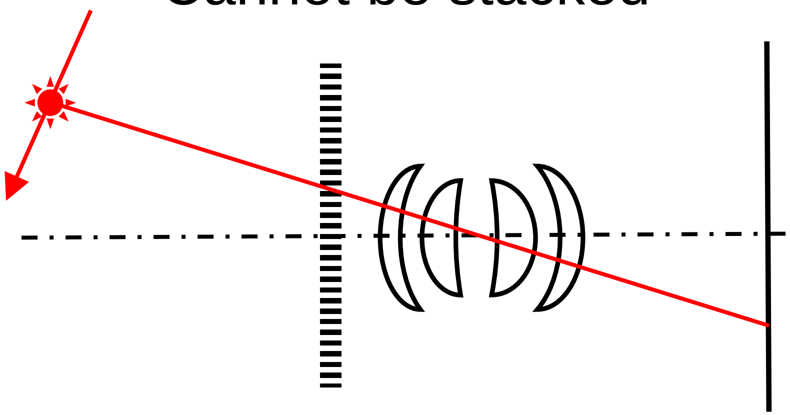
Hardware

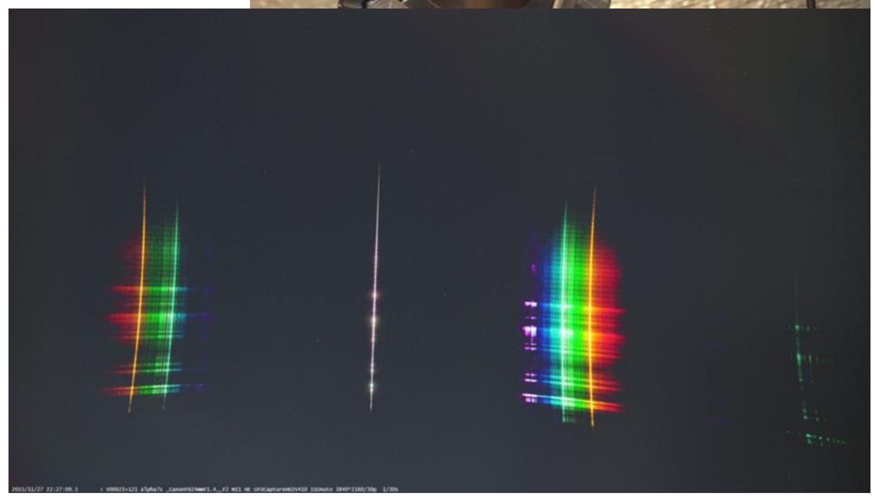
- Watec 902H2 ult. Computar HG2610AFCS-HSP F/1 2.6mm fl
- 902H2 ultimate (spectroscopy) Tamron 12VG412ASIR F/1.2, ≈7mm fl
- 2nd camera with transmission grating for spectroscopy Thorlabs 600 L/mm



Starting point

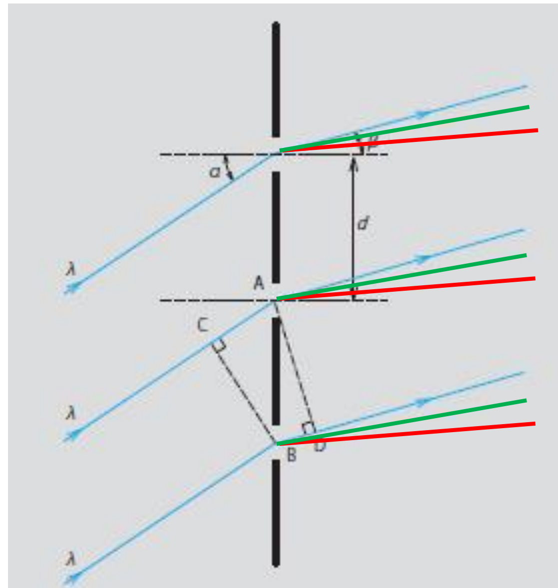
- Camera with wide angle lens
- Transmission grating
  - mounted perpendicular to optical axis!
- Problem:
  - Moving meteor
  - Curved spectra with nonlinear dispersion
  - Cannot be stacked





Spectrograph, theory

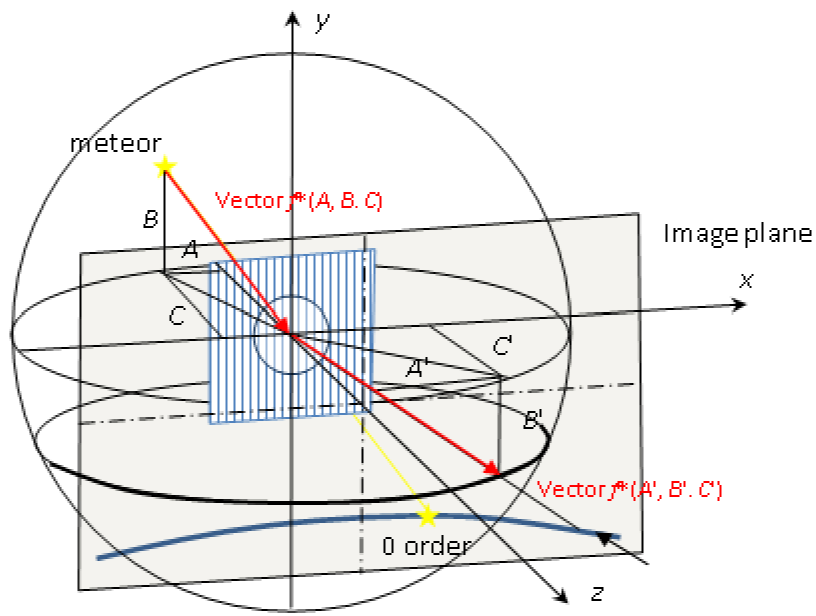
- Video camera with transmission grating in front of lens
- Grating equation:
  - **m * λ * G = (sin α - sin β) * cos γ**
  - m: grating order, G: grating lines / mm
  - λ: wavelength
  - α, β: angle of incidence, transmitted beam
    γ: cross, out of plane angle



- Inverse dispersion per pixel:
  dλ/dx = (cos β cos γ)/(m*G*f)*p (p: pixel size)
  - Example: f = 7 mm, p = 8.6 µm, G: 600L/mm β = 0 → dλ/dx = 2.1nm/pixel

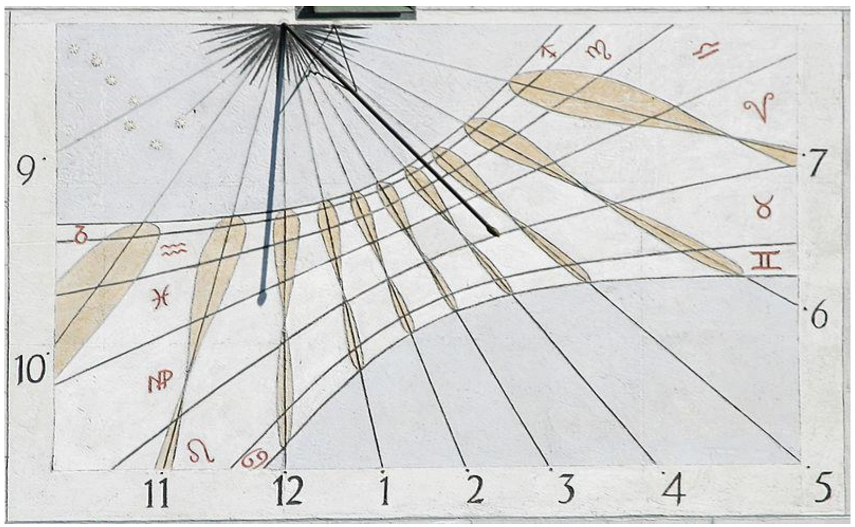
Vector theory wavelength calibration

- Grating in front of lens perpendicular to optical (z-)axis
- Unit vector (A B C) for incident direction
- Diffracted beam
  $A' = A + m\lambda G$ *(x-axis)*
  $B' = B$ *(y-axis)*
  $C' = \mathrm{sqrt}(1 - A'^2 - B'^2)$
- Spectrum on CCD plane
  - Nonlinear dispersion
  - Hyperbolic curvature
- Spectrum straight linear in A',B'

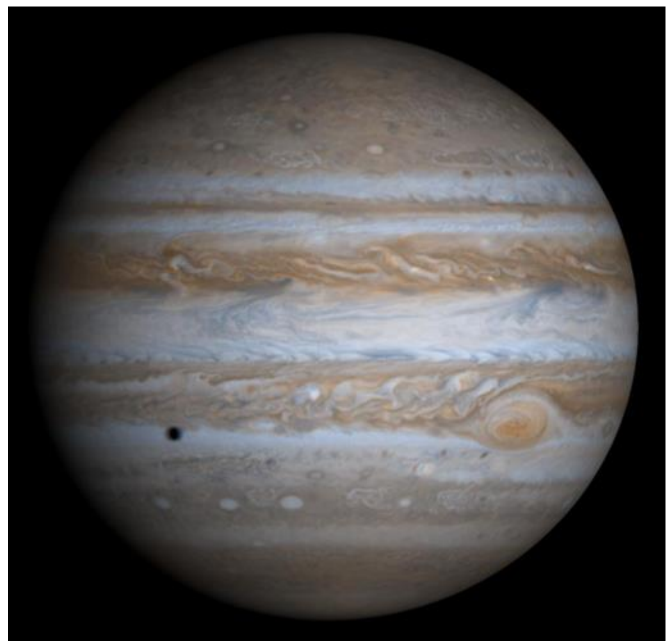


Gnomonic and orthographic projection

- Gnomonic, TAN
  - R = f*tan(ρ)
  - Great circles → straight
  - Optimum for path, radiant
  - Latitude circles → hyperbola

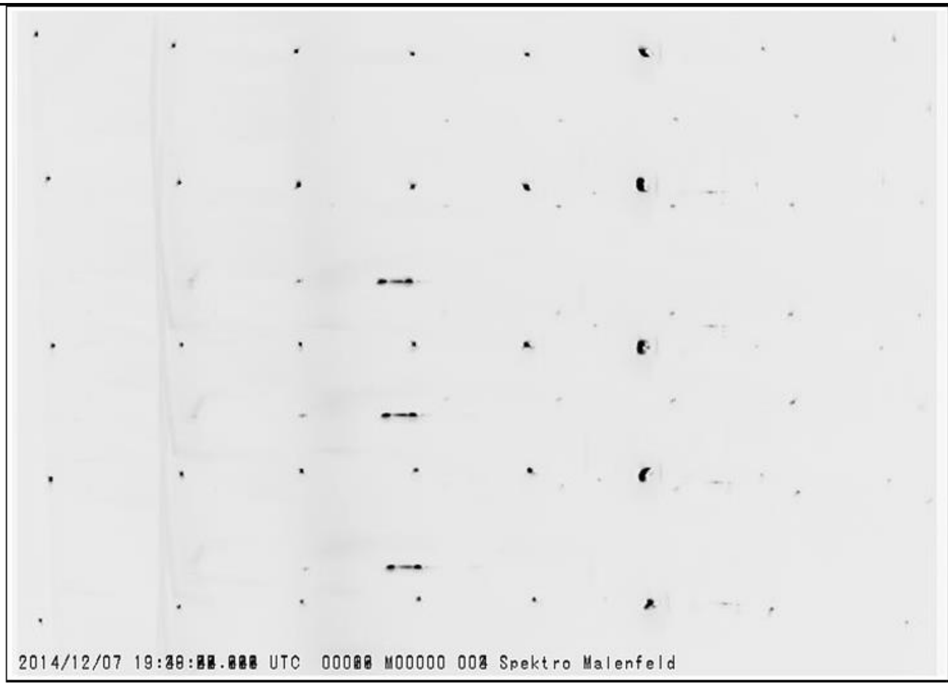


- Orthographic, SIN
  - R = f*sin(ρ)
  - Great circles →ellipses
  - Latitude circles → straight
  - Optimum for spectroscopy

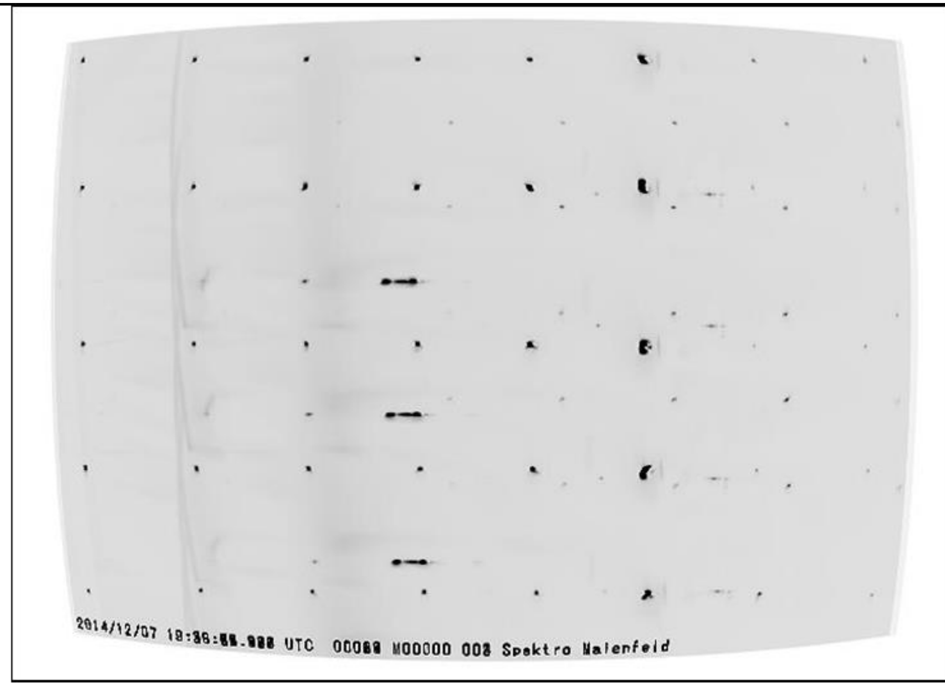


Calibration spectrum HeNe laser

- HeNe laser λ = 633 nm, f = 4 mm, grating 300L/mm
- Fit with polynom r = r' * [1 + 3.94E-07*r'2 + 2.01E-12*r'4]
- Fit center x_0, y_0

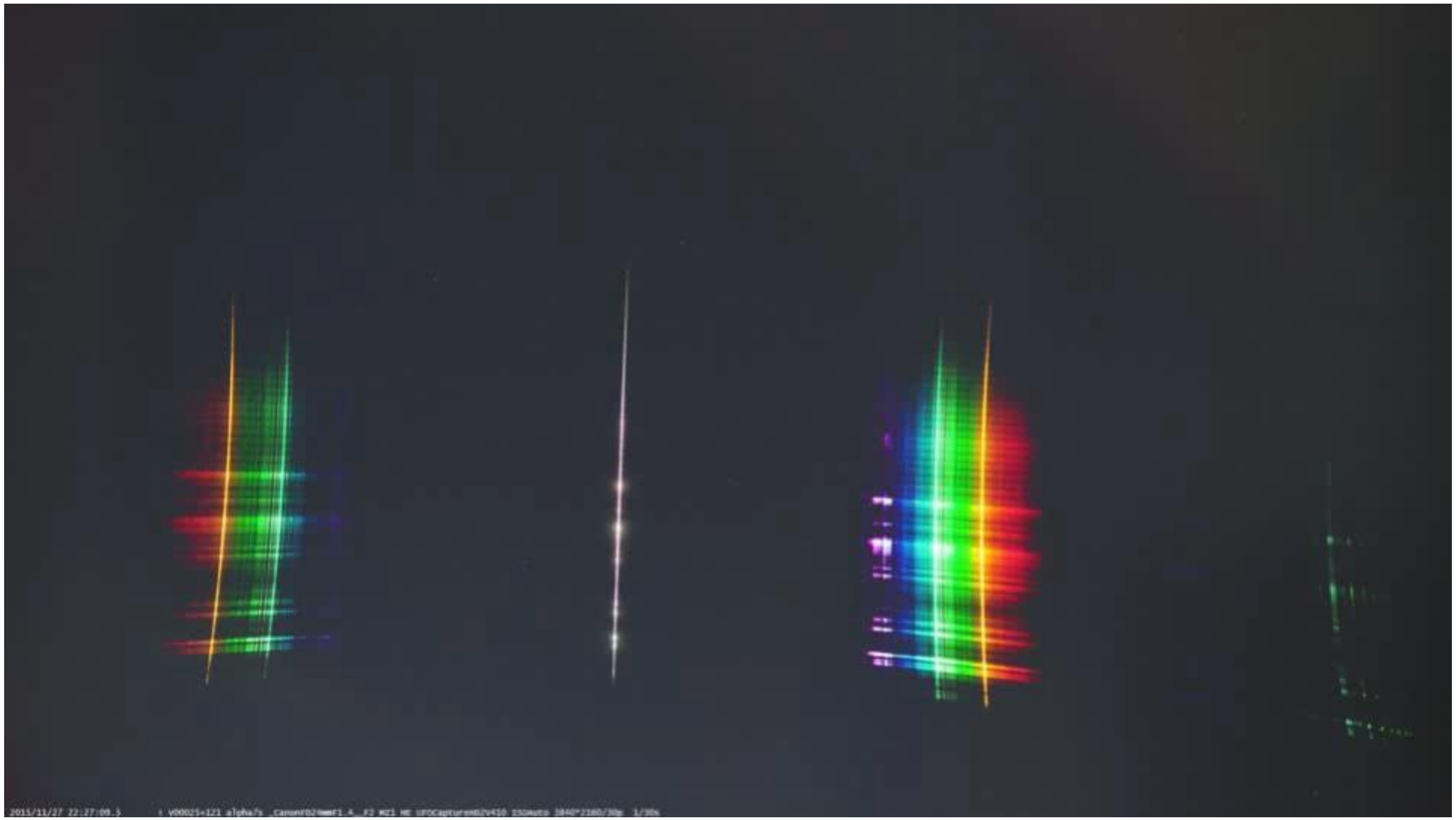


Composite spectra original

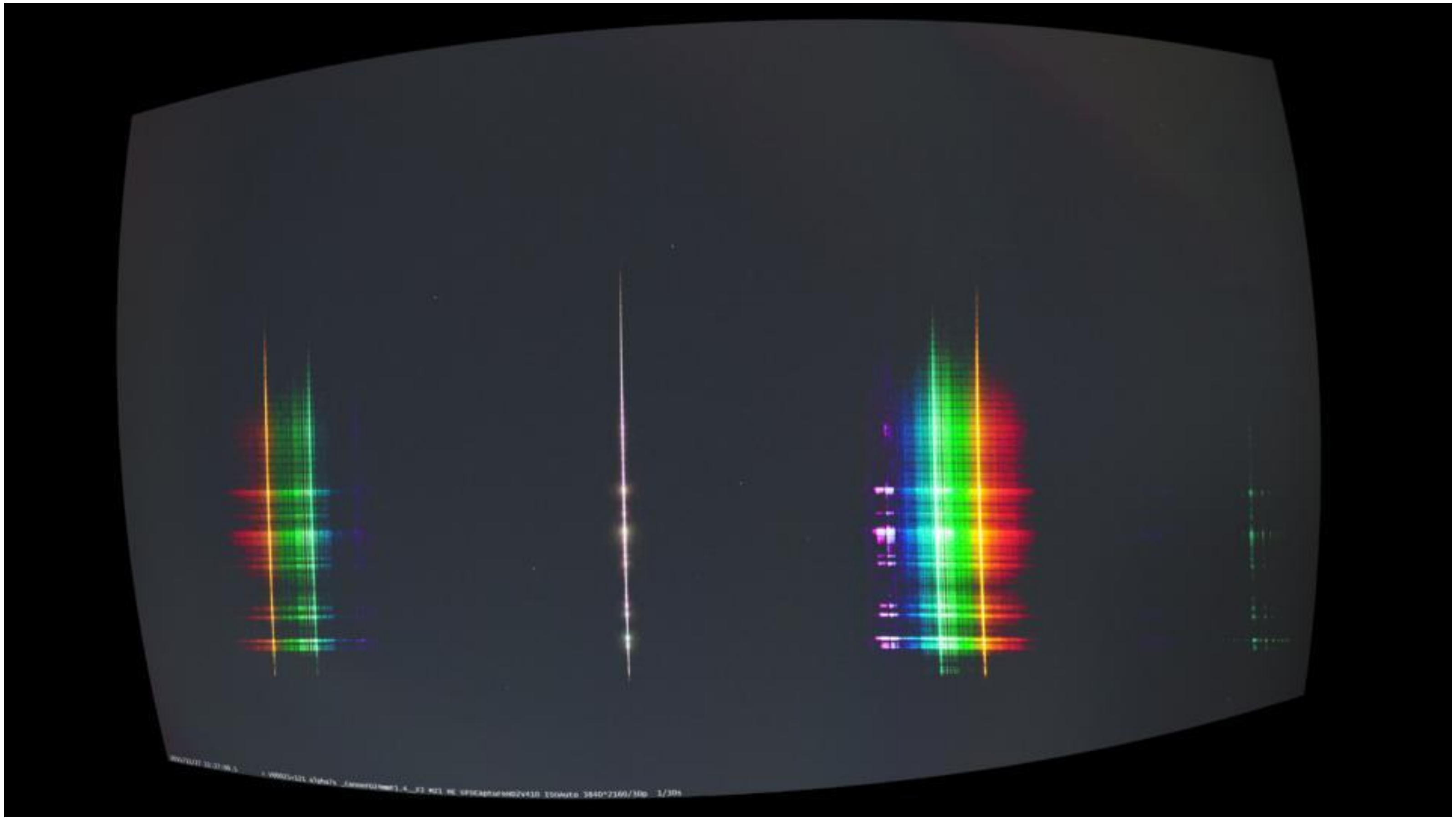


After applying transformation

Orthographic transformation, original



Orthographic transformation, result

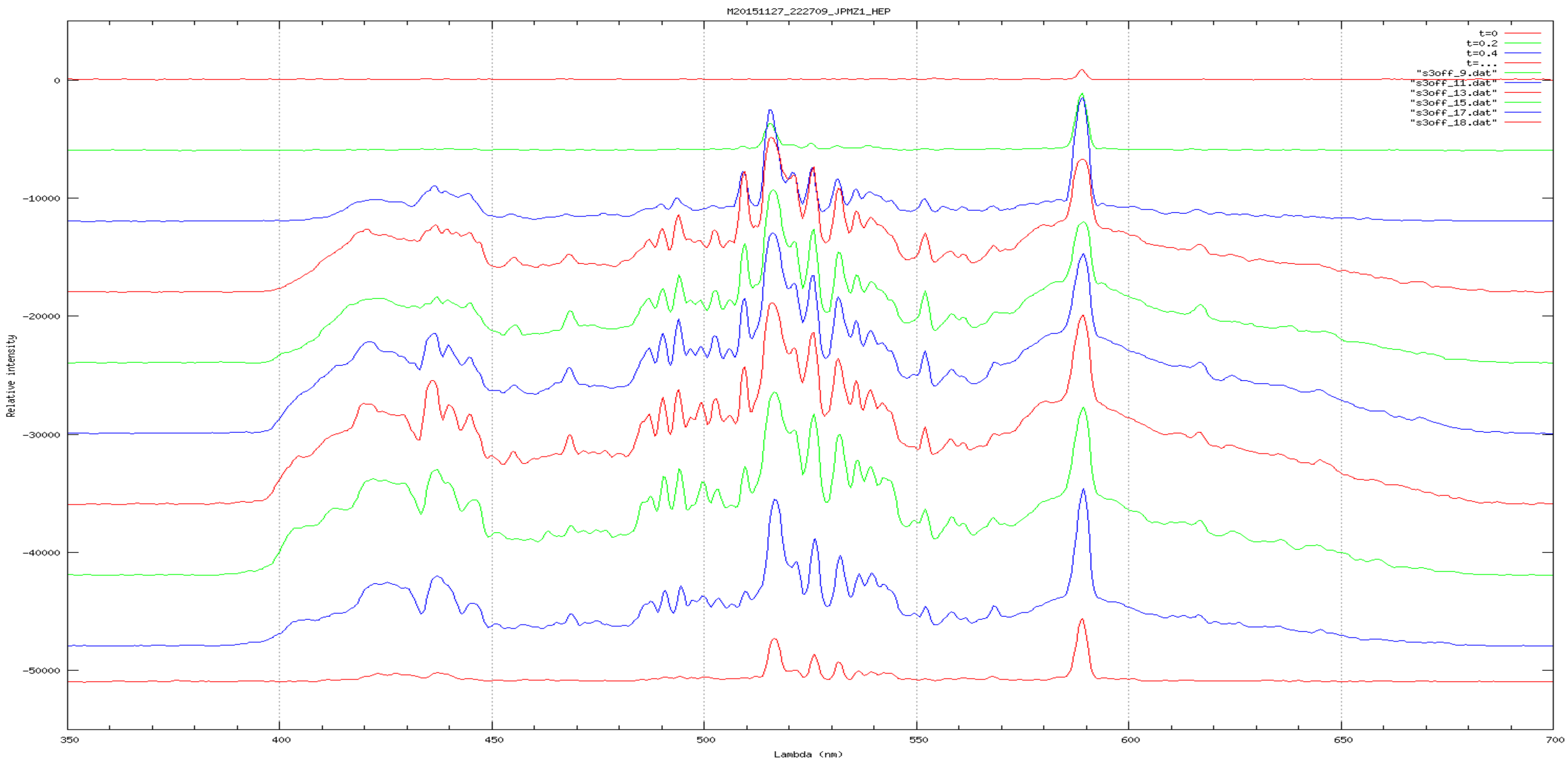


Orthographic transformation, result

- Frames converted to b/w, linearized,registered, M20151127_222709

- color

Extraction of spectra



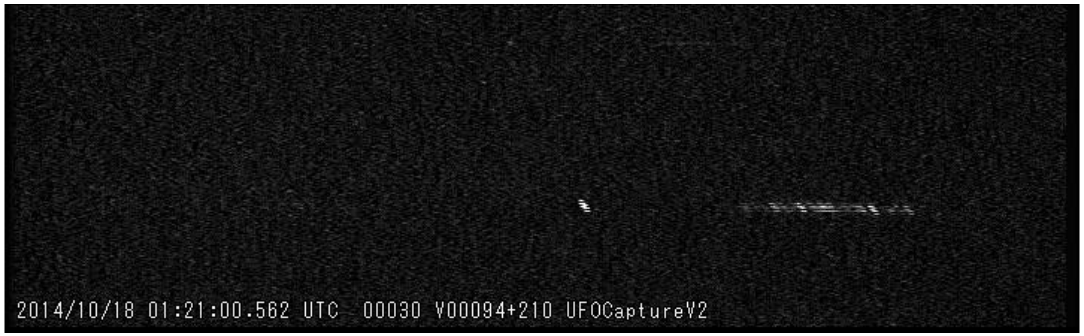
Full processing, using different software

- Image extraction with VirtualDub
- preprocessing:
  - Background subtraction (IRIS)
  - Image transformation (ImageTools by Peter Schlatter)
  - Stacking of spectra (IRIS)
- Extract spectrum, calibrate wavelength (SpectraTools)
- (correction of instrument response, ISIS)
  - Grating efficiency
  - Camera spectral sensitivity (lens, CCD)
  - Atmospheric transmittance

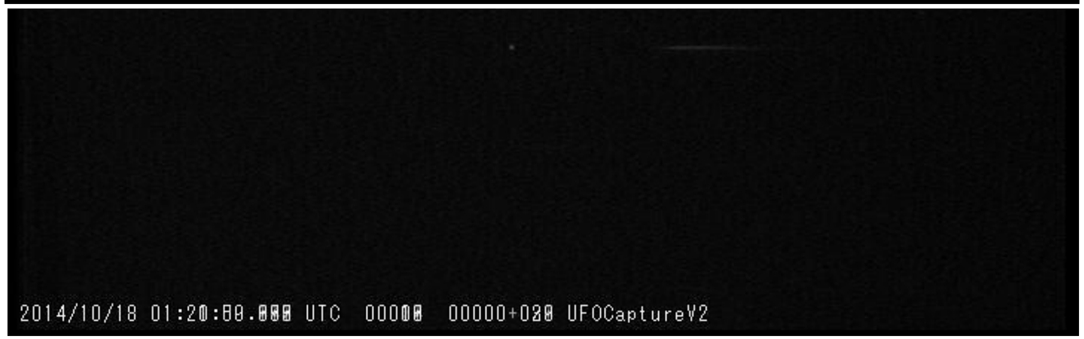
} instrument response

Preprocessing

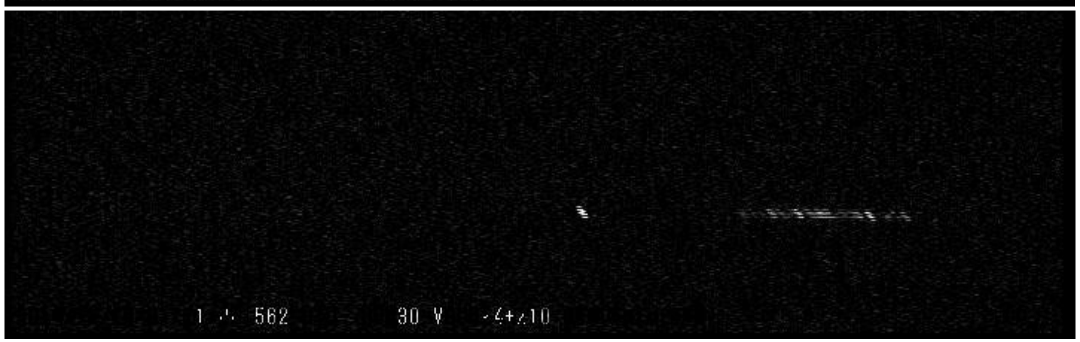
- Extract image (i30)



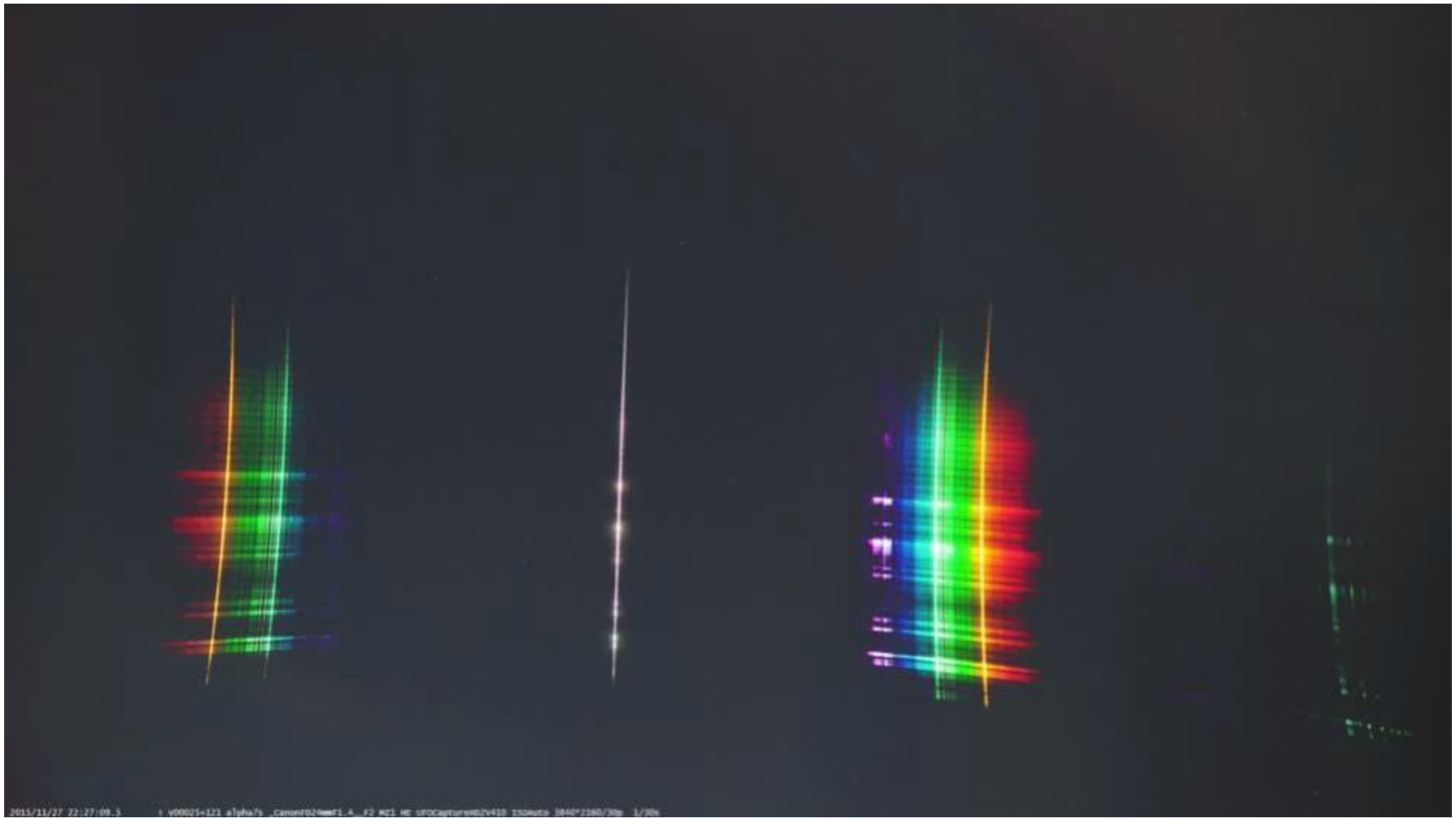
- Background ADD_MEAN < I1 … I20 >



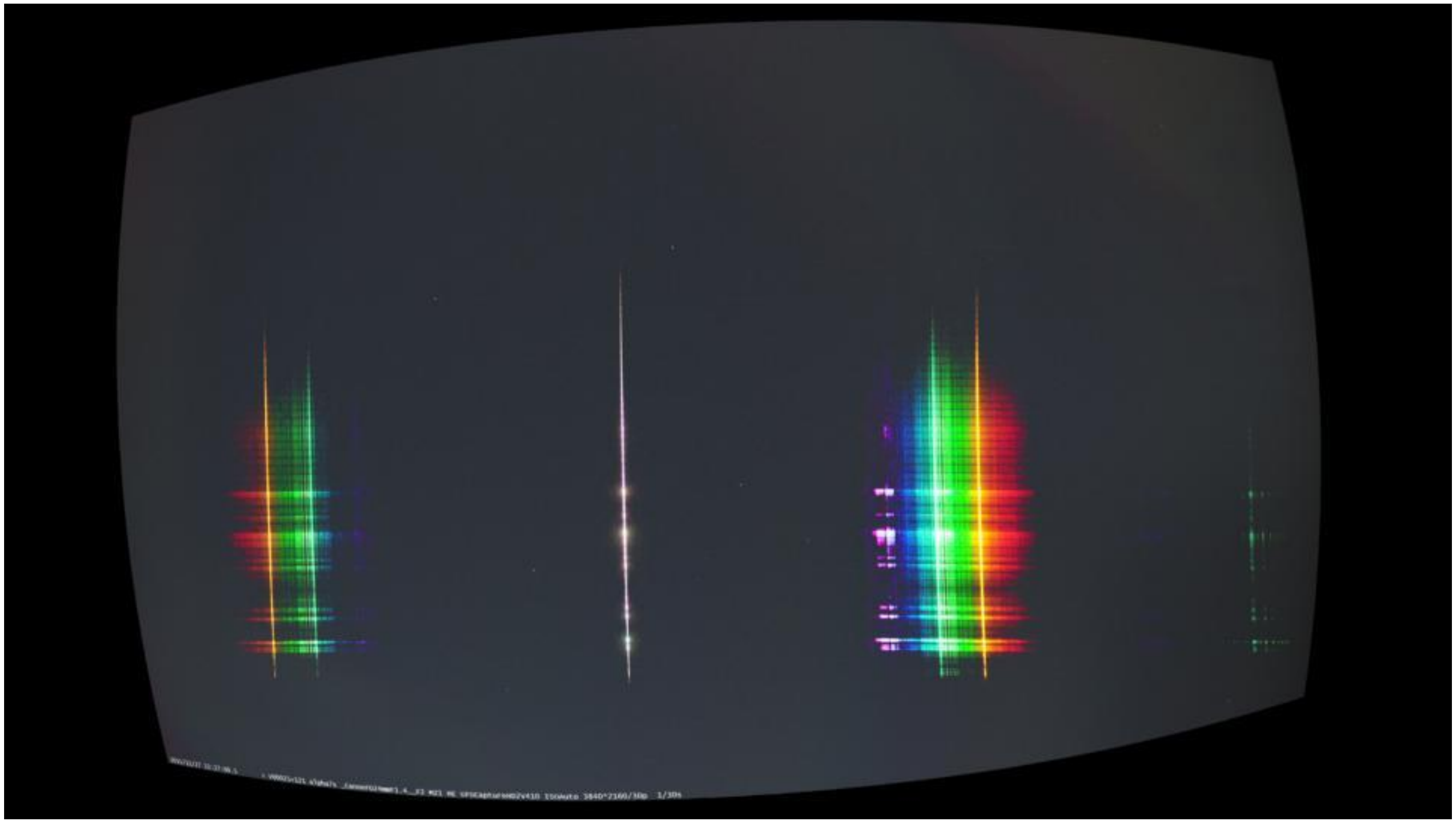
- Subtraction SUB2



Orthographic transformation, original

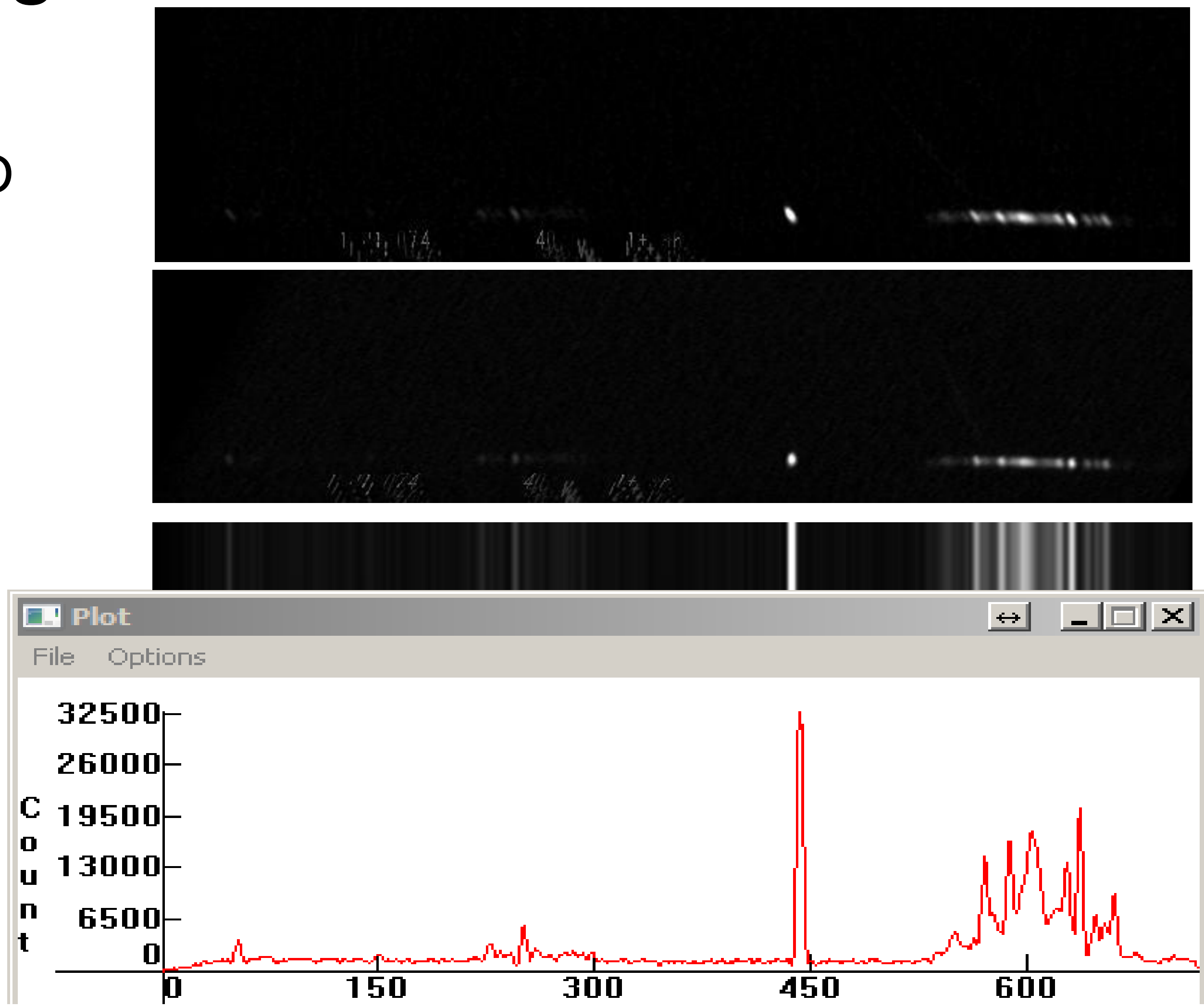


Orthographic transformation, result



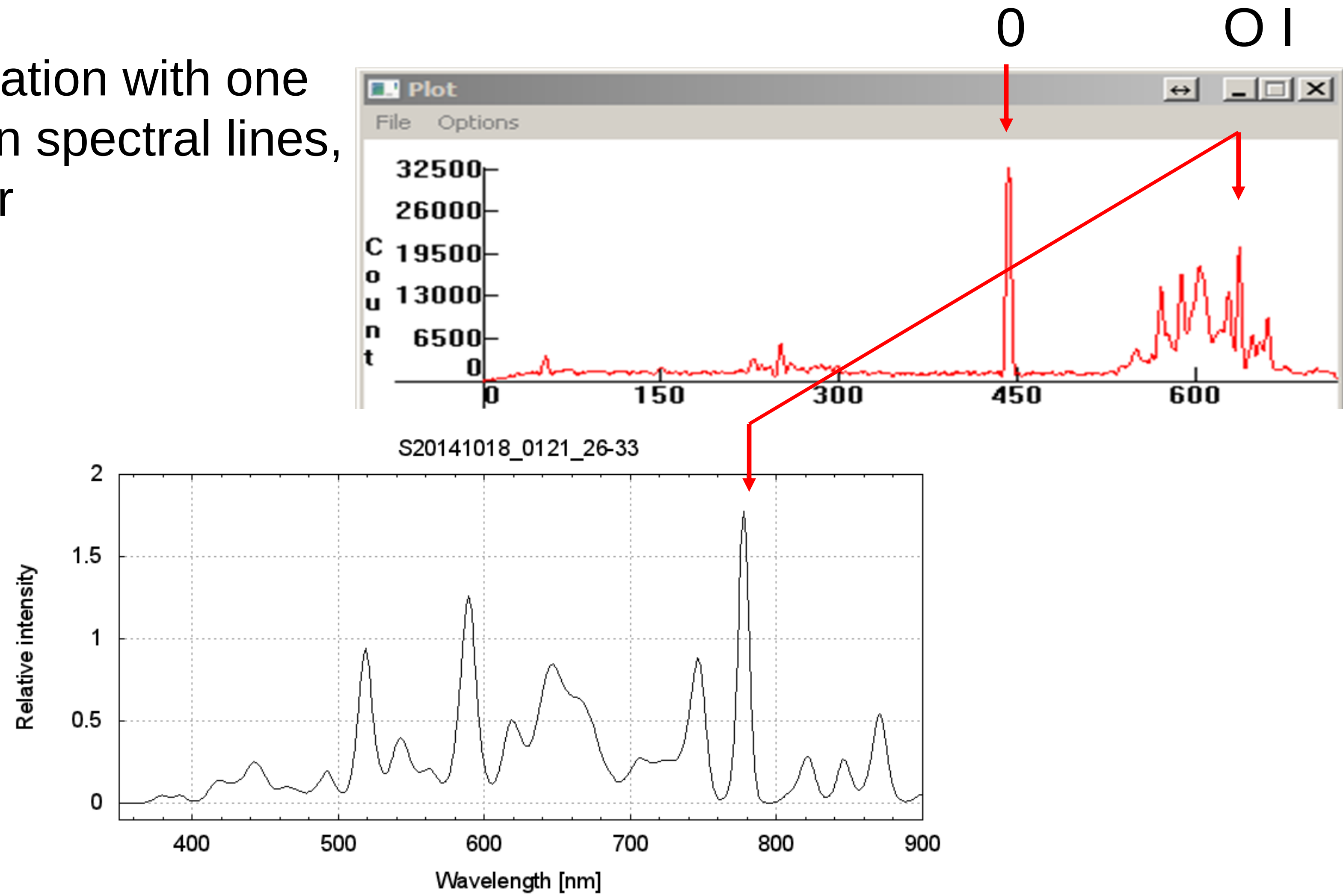
Processing 2

- Register
  Add
  TRANS, ADD
  ib34-ib40
- Slant 28 30
- L_ADD
- L_PLOT
  save file.dat



Wavelength calibration

- Linear calibration with one or two known spectral lines, or zero order



Meteor spectrum processing

- Moving meteor requires special treatment of spectra
  - Nonlinear curved spectra, dispersion varies with meteor position
  - Linear spectra after orthographic transformation
  - Standard spectroscopy software only partially useful
  - Combination of different software required
- New approach: processing pipeline with Python script
  - One script for laser calibration → calibration parameters
  - One script for spectrum extraction
    video file → plot of wavelength calibrated meteor spectrum
- For info see:
  https://meteorspectroscopy.org/...meteor-spectraanalysed-with-python

Why Python

- it contains all the necessary tools to do the analysis
  - Image processing
  - Astronomical packages (FITS-format)
  - Fitting algorithms (peak positions, least square fit)
  - Plotting
- it finds widespread use in the astronomy community
- it is free
- it runs on different platforms

Laser calibration

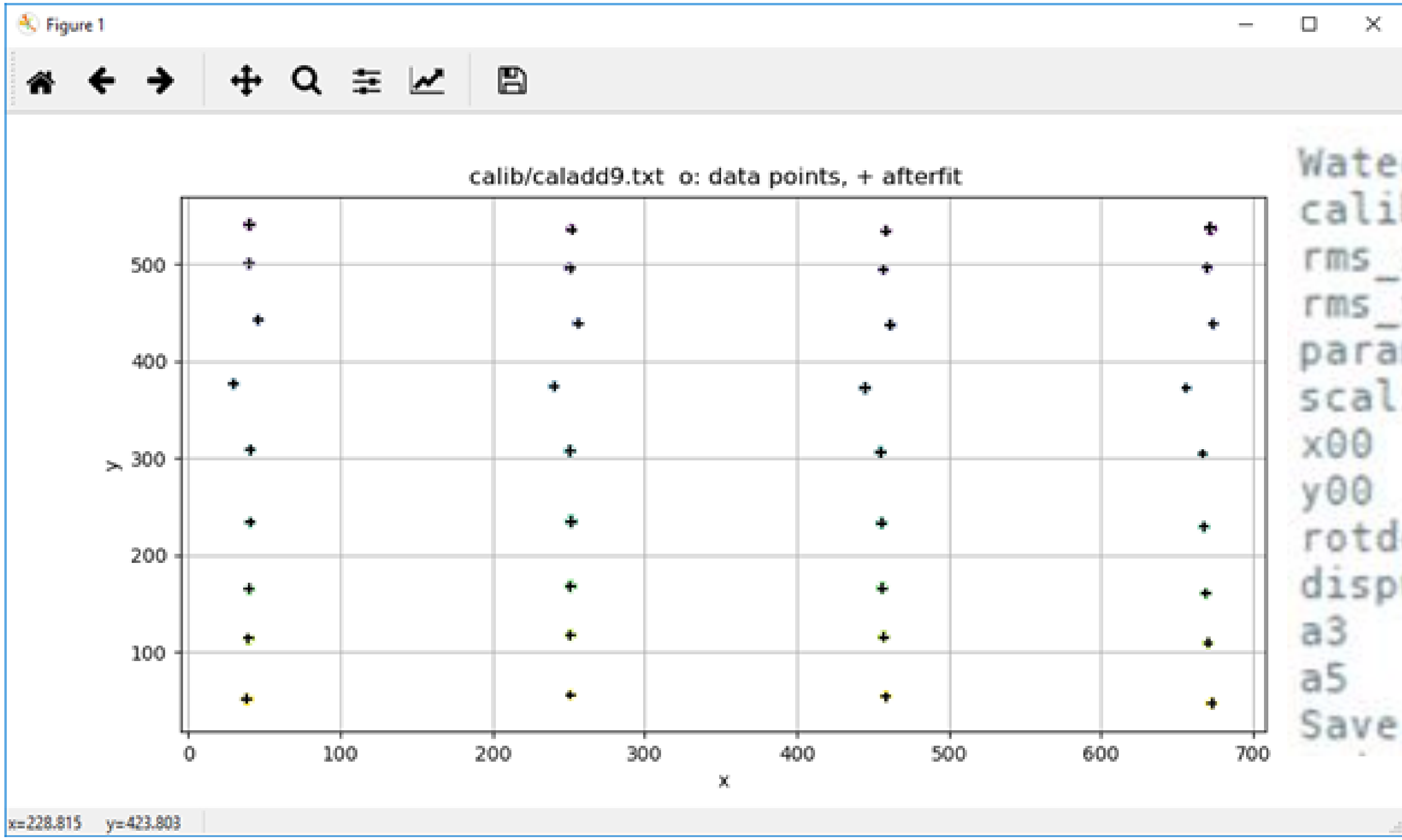
- Script for least square fit of laser spectra
  - Set initial parameters
  - Load image
  - Mark different orders
  - Least square fit
  - Save results to *.ini file

Run calibration script, s_calib.py

- Mark laser lines:

- Fit equation:
  - Calculate in polar coordinates
  - Radial transformation to orthographic projection
    $r = r' * (1 + a3*r'^2 + a5*r'^4)$ (includes lens distortion)
- Results:
  - x00, y00: coordinates of optical axis
  - rot: angle of rotation of spectra
  - disp0: dispersion [nm/pixel]
  - a3, a5: radial transformation parameters
  - rmsx, rmsy: fit errors

Calibration results




```
Watec 902 H2 ultimate, Tamron
calib/caladd9.txt
rms_x =  0.3042
rms_y =  0.4483
parameters after fit:
scalxy =      0.9200
x00    =    344.4033
y00    =    324.2399
rotdeg =     -0.3588
disp0  =      1.9928
a3     =  3.1525e-07
a5     = -3.2818e-13
Save config in directory [] n:
```

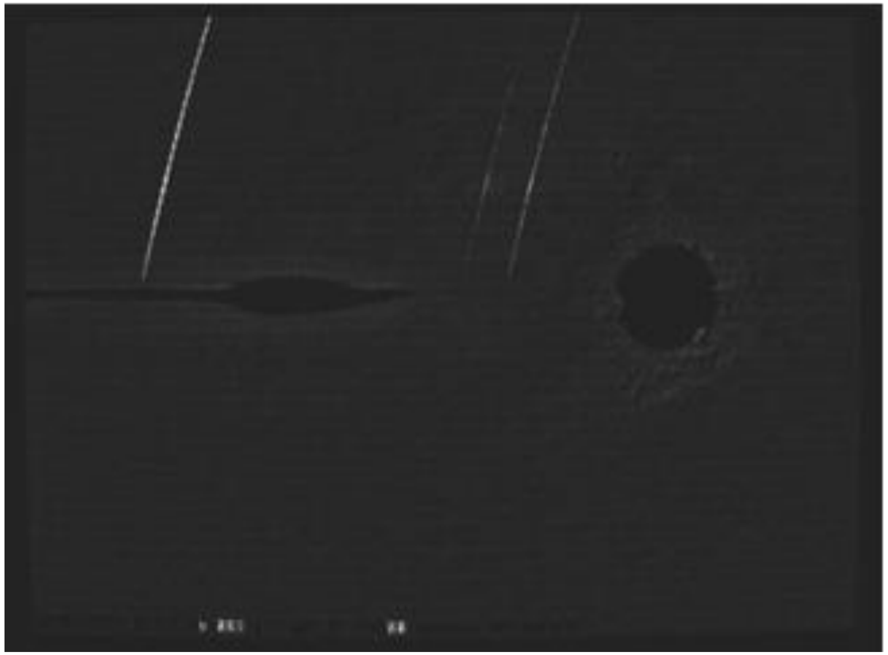
Meteor spectra processing with Python

- Processing pipeline:
  - Image extraction from video file
  - Preprocessing
  - Image transformation
  - Stacking (tilt, slant correction)
  - Wavelength calibration
  - Plotting, save data
- (→ spectrum analysis)
  - Instrument response
  - Line intensities (Na, Mg, Fe- ratios)

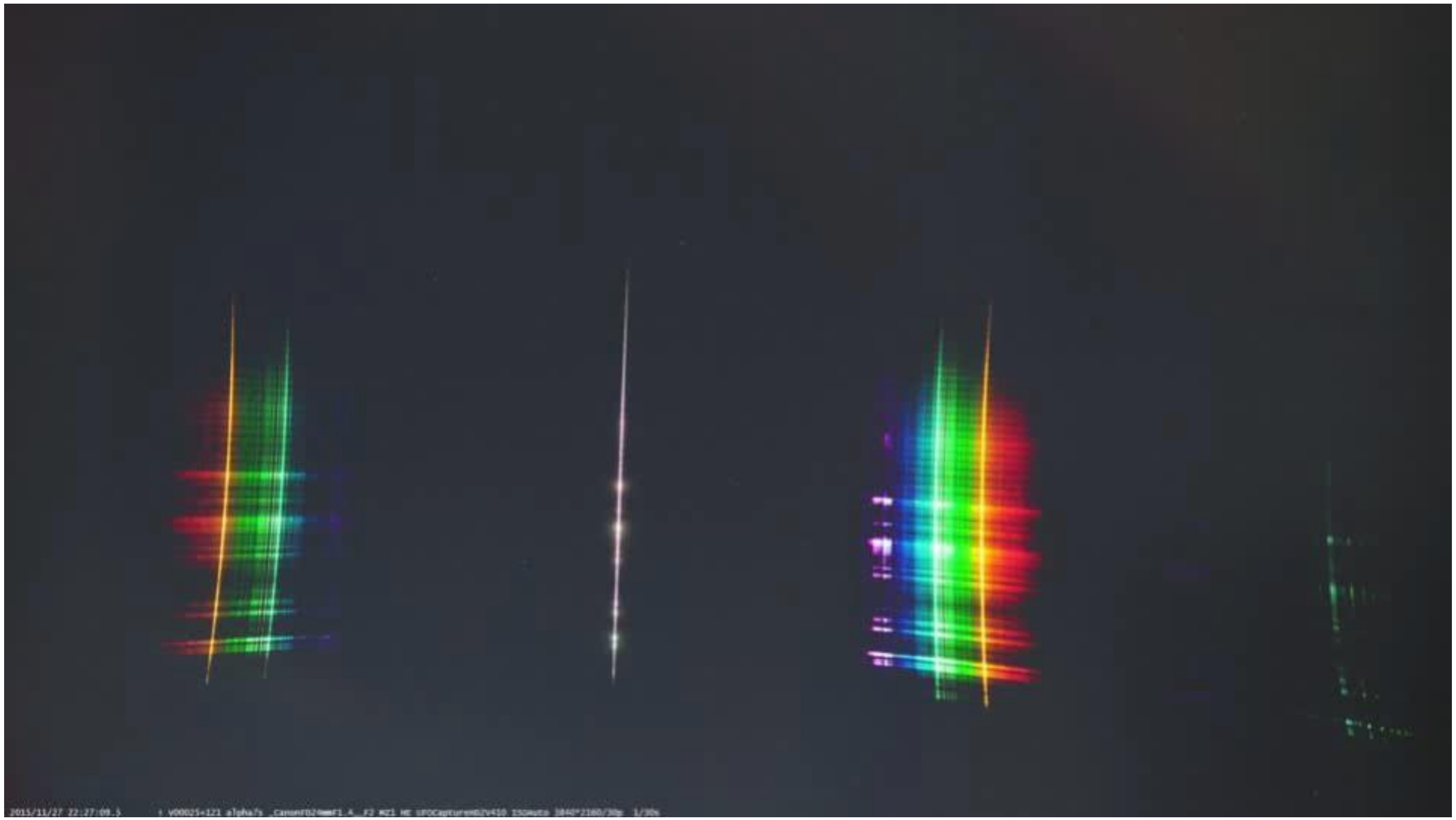
Preprocessing with Python

- Before background subtraction with moonlight

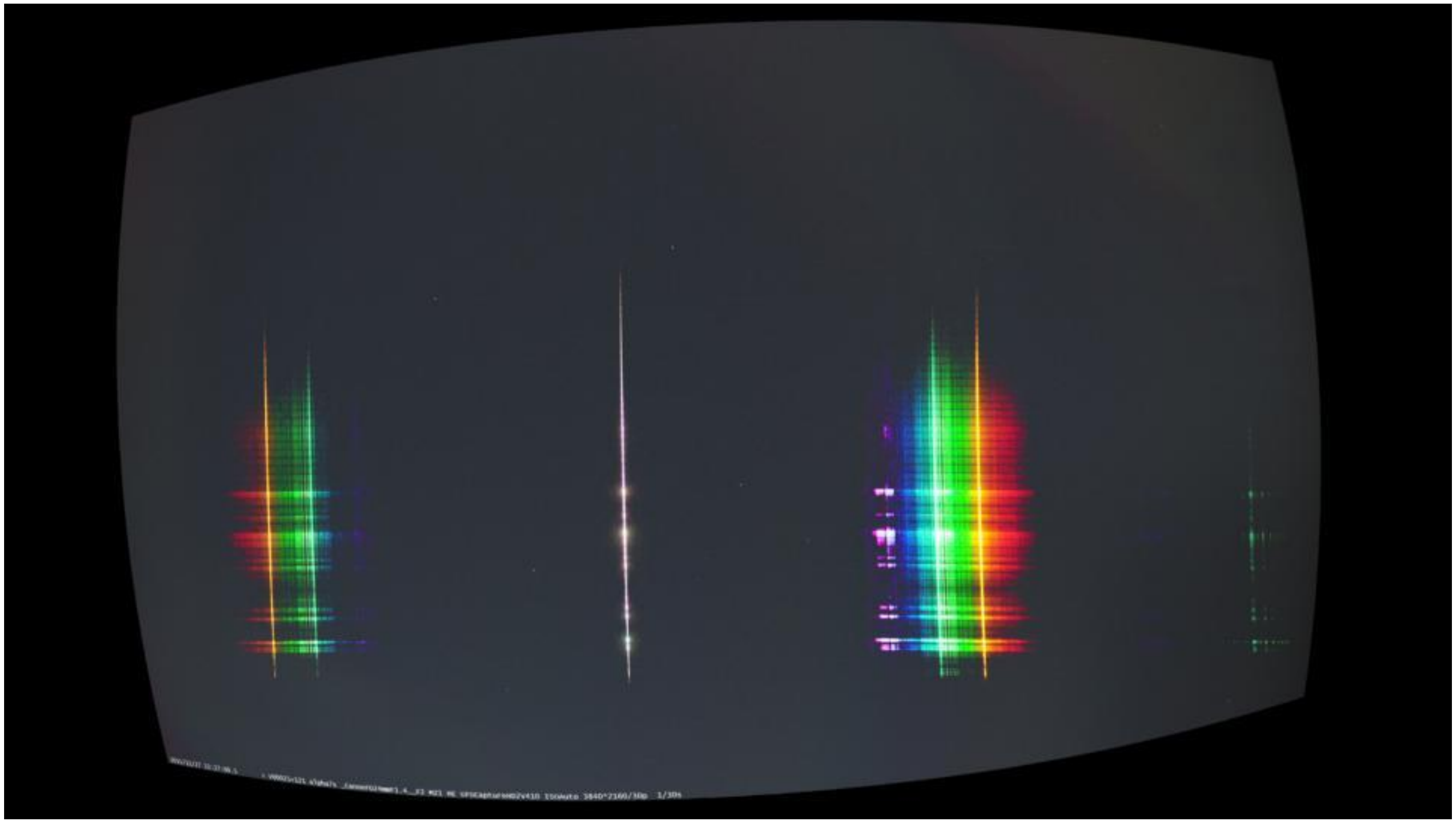
- After background subtraction



Orthographic transformation, original

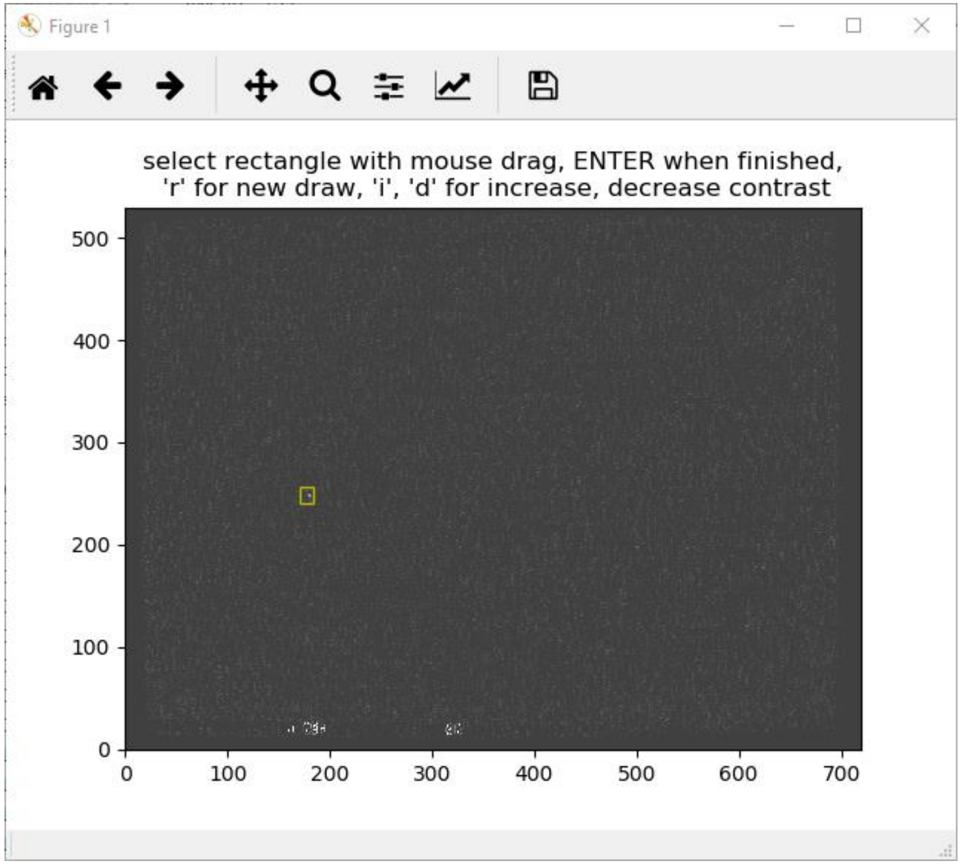


Orthographic transformation, result

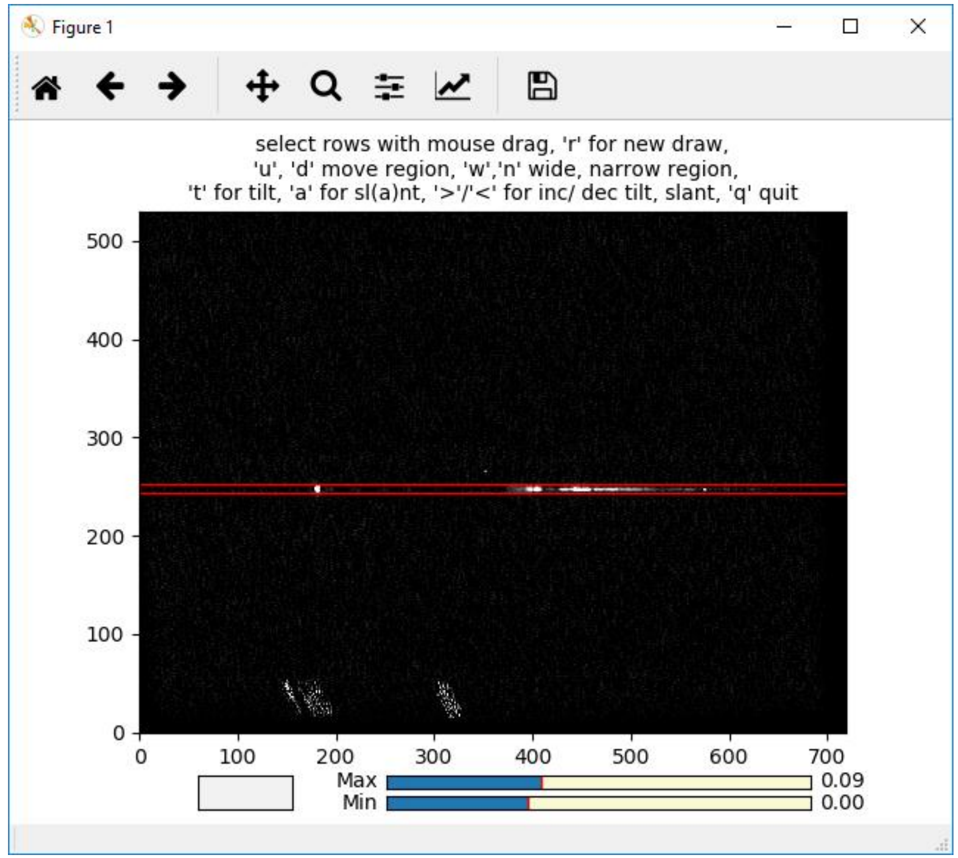


Registering and stacking spectra

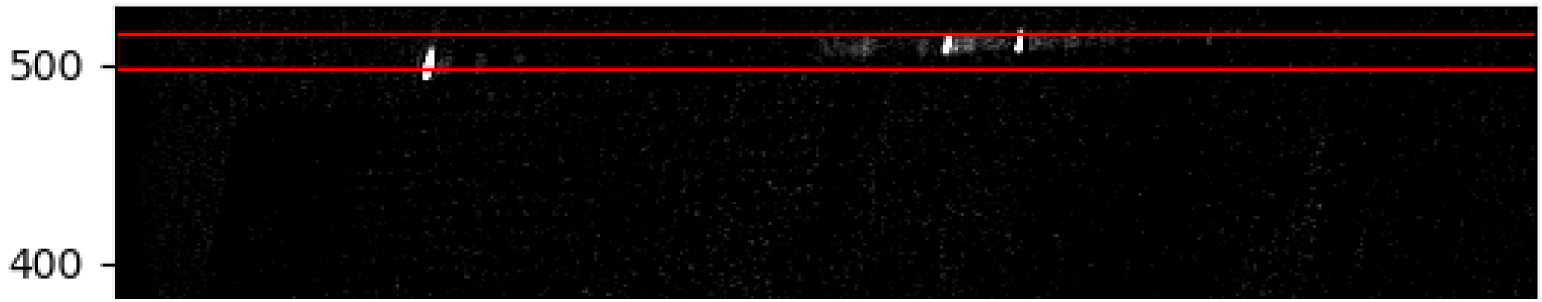
- Select 0 order for registering spectra



- Select range of rows for addition



Tilt and slant correction

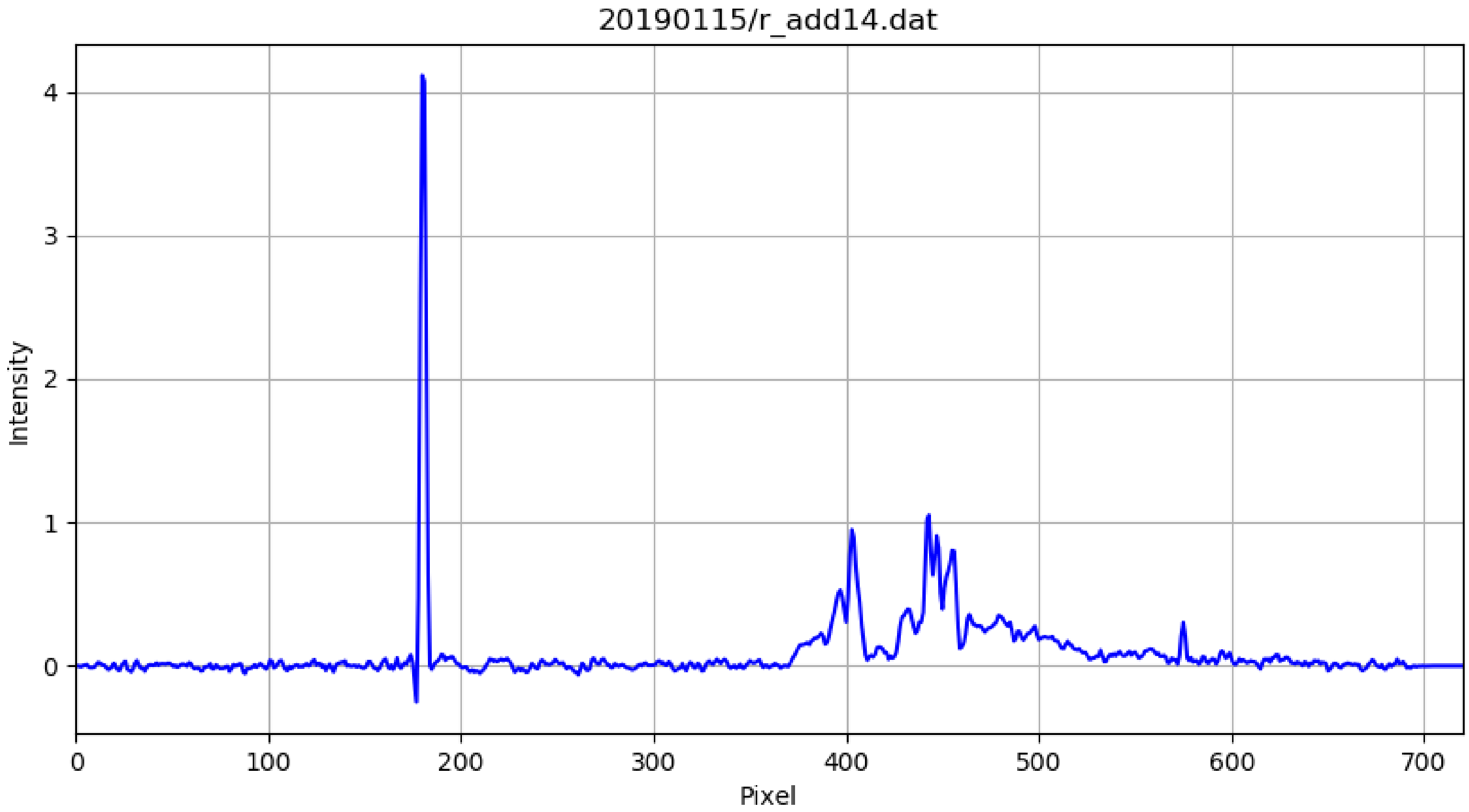


- Tilt correction

- Slant correction

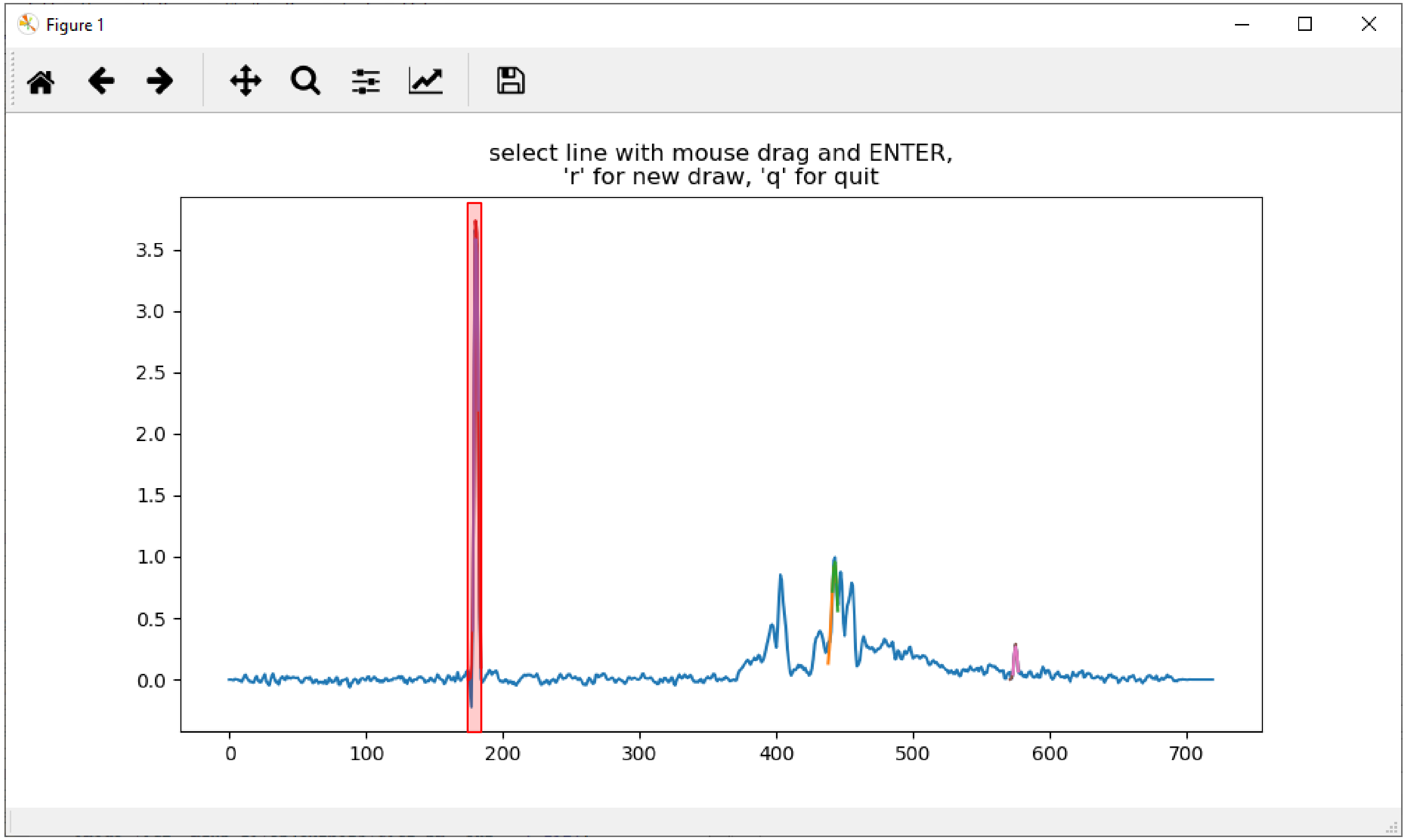
Extract raw spectrum

- Add lines → 1d-spectrum, uncalibrated



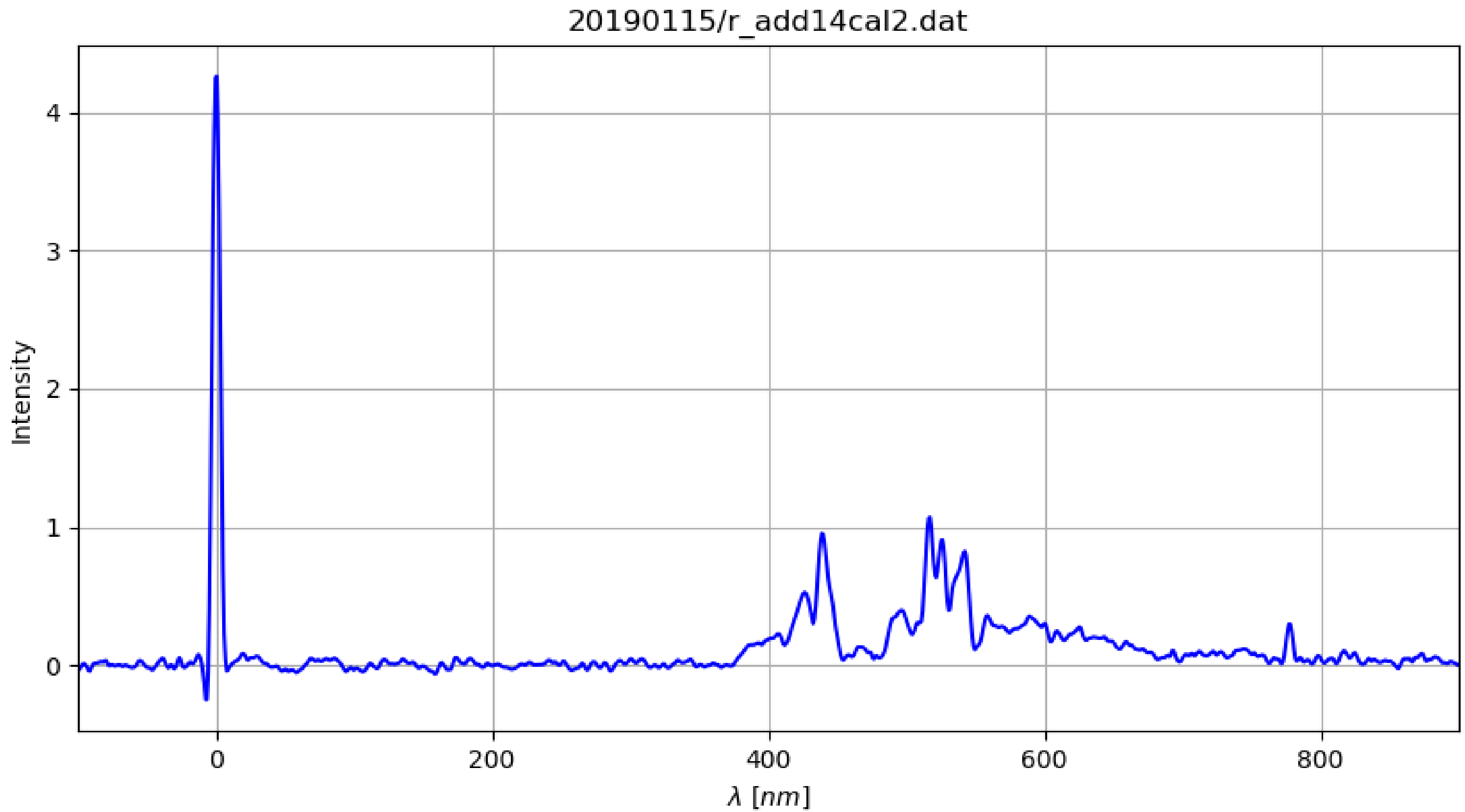
Wavelength calibration

- Select known lines, assign wavelength (0 for zero order)



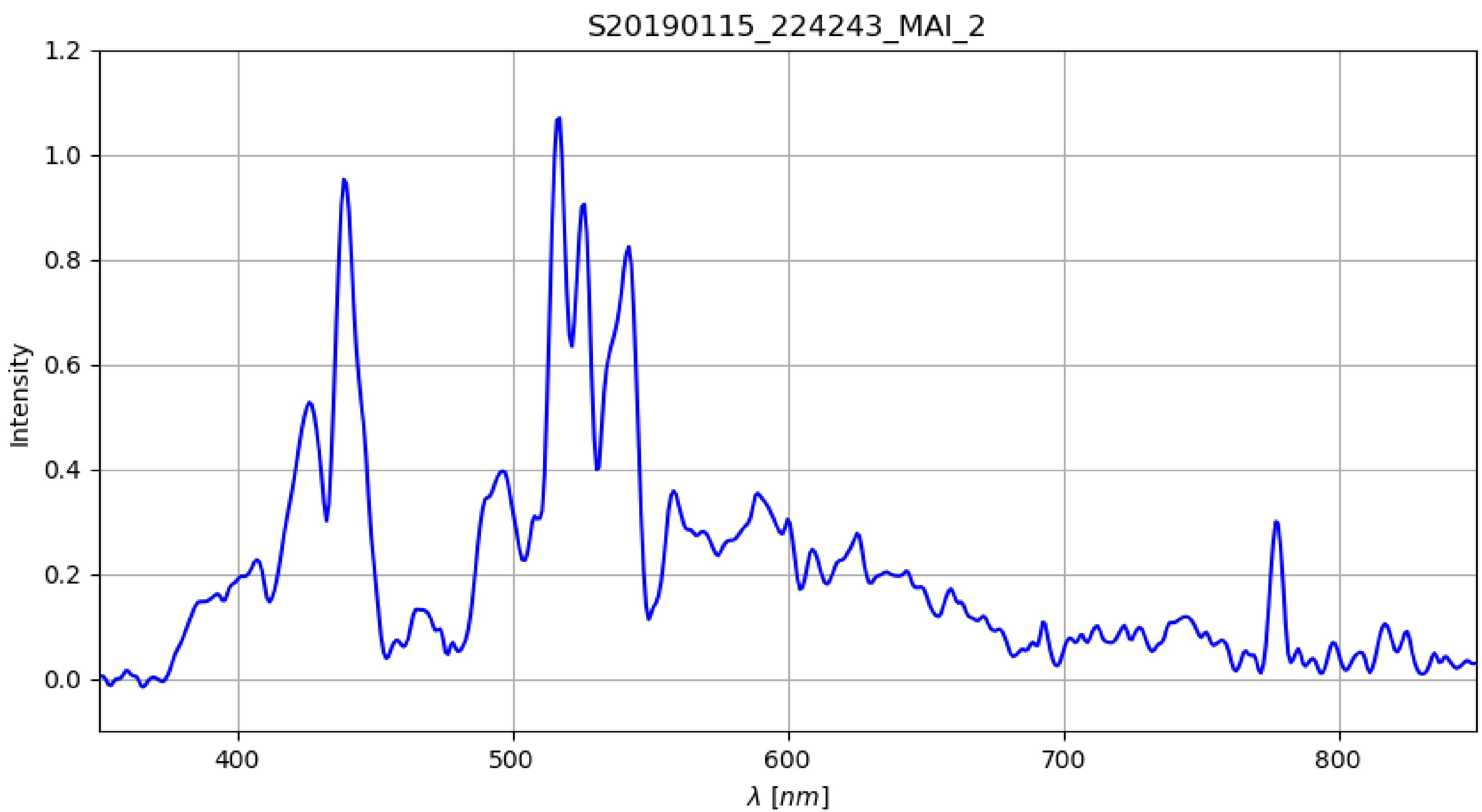
Calibrated spectrum

- After linear or polynomial fit → calibrated spectrum



Plot final spectrum

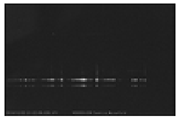
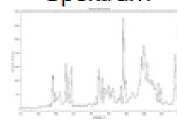
- Select range, set title etc.



Results

- See: http://www.meteorastronomie.ch/ergebnisse_spektroskopie.html

ERGEBNISSE:

	Bild	Video	Spektrum
			
15. Januar 2019, 22:42 UT low Na / high Fe	MAI (Spektrum)		MAI (Spektrum)
31. Dezember 2018	Overview	Meteor Spectra 2018 MAI	Analysis
20. Juni 2018, 20:40 UT mit Aufblitzen	MAI (Spektrum) / MAI / VTE	MAI (Spektrum) / MAI / VTE	
07. Februar 2018, 02:05 UT majestätisch	MAI	MAI	full range / 1. order / doc
31. Januar 2018, 03:05 UT über Norditalien	MAI	MAI	full range / -1. order / doc
08. December 2017, 19:12 UT Aufnahme: Koji Maeda (Miyazaki) Analyse: Martin Dubs (MAI)	Koji Maeda	Koji Maeda linearisiert (MAI)	1. + 2. Ordnung anim. (MAI) Spektrum gesamt (MAI) Beschreibung (MAI)

Outlook

- Instrument response, flat field
- Graphical user interface for Python
- Improve spectral resolution, HD-Video camera

Characteristic meteor spectra

- Catalogue of meteor spectra: V. Vojacek et. Al.
  http://adsabs.harvard.edu/abs/2015A%26A...580A..67V

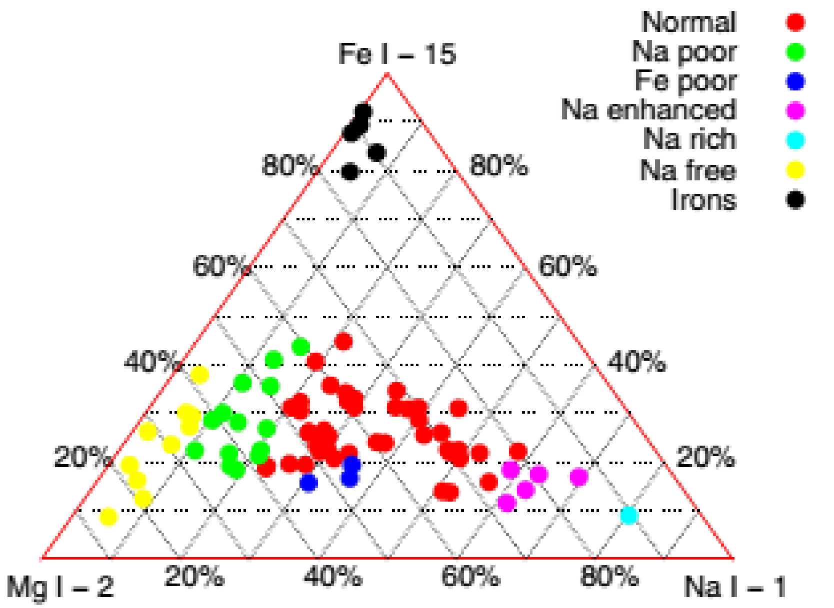


Fig. 8. Classification of meteor spectra. The ternary graph of the Mg I (2), Na I (1), and Fe I (15) multiplet relative intensities. Every group of meteoroids is represented with a different symbol.

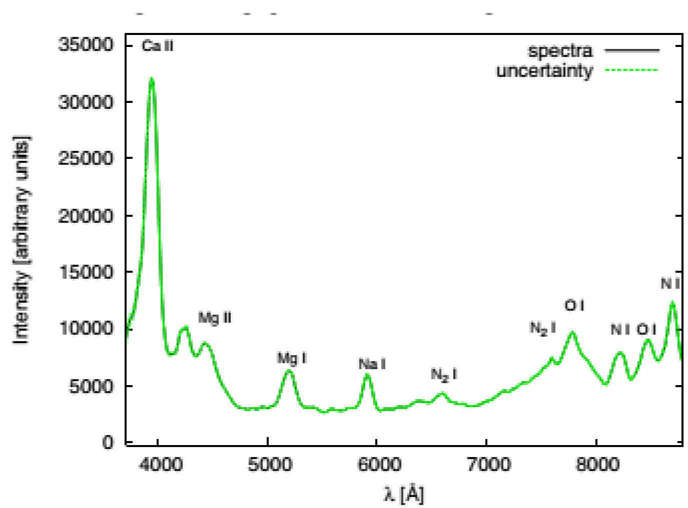


Fig. 5. Spectrum SX1837 of a bright Perseid. The meteor had a maximum brightness of −9.2 mag. Because the spectra were oversaturated on the video frames around the brightness maximum, one frame of the sequence was chosen. The brightness of the meteor in this frame was −7.5 mag.

Python meteor spectrum processing

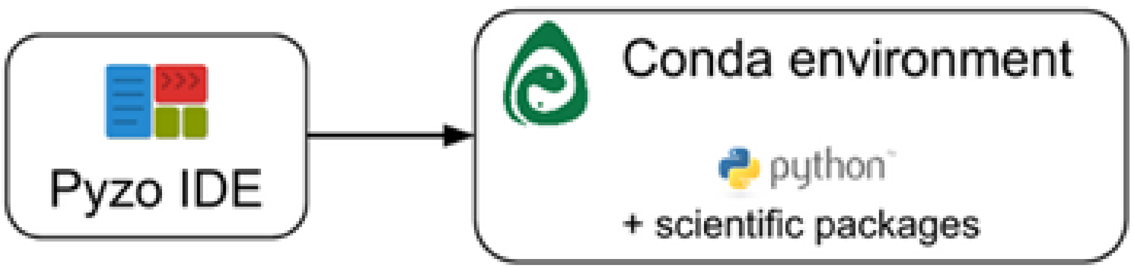
- Python: https://www.python.org/
  (https://www.anaconda.com/download/#windows)
  https://repo.anaconda.com/archive/
- Spectrum processing manual short or detailed look at:
  https://meteorspectroscopy.org/welcome/documents/
  in the section Meteor Spectroscopy, manuals
- For more info, scripts and demo spectra contact the author for Dropbox link

Testrun m_pipe62.py

- Install Pyzo as programming environment
- Install Python from Anaconda
- Copy example from Dropbox\Python\s_calib.zip
- Copy example from Dropbox\Python\Python_demo.zip
- Setup working directory in Pyzo
- Run s-calib.py
- Run m_pipe62.py
  - m_set.ini
  - Directories: tmp, out
- Results, logfile

Install Pyzo

- http://www.pyzo.org/start.html
- See Meteor spectra Python manual_V6.pdf for details



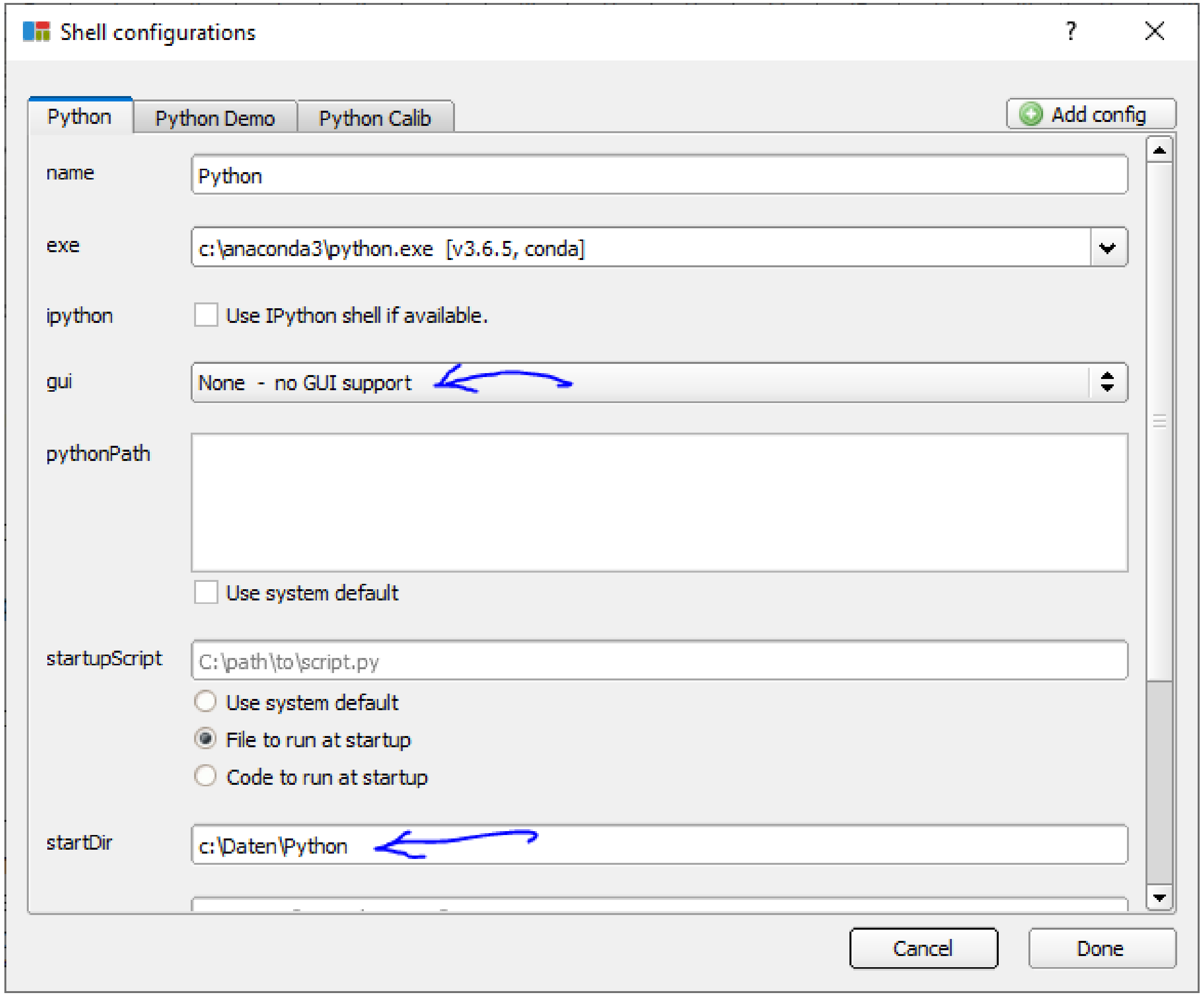
To get started with Pyzo, you need to install the Pyzo IDE (in which you write your code) and a Python environment (in which you run your code).

Install Anaconda

- https://www.anaconda.com/download/
- Or better: download the version 5.2.0-windows-x86 from the Anaconda archive: https://repo.anaconda.com/archive/
  Select the 32 or 64 bit version, depending on your computer.
- Version 3.7 did not work with my script
- Install libraries: >>> conda install -c conda-forge lmfit

Select working directory in Pyzo

- In Pyzo shell:
- Different configurations possible



Check configuration file

- Default: m_set.ini
- Edit if necessary
- Or run s_calib.py first
- [Lasercalib]
  start values for s_calib.py
- [Calib] parameters for
  m_pipe62.py

m_set.ini - Editor

Datei Bearbeiten Format Ansicht Hilfe

```
[Lasercal]
f_lam0 = 405.0
f_scalxy = 1.0
b_fitxy = 1
i_imx = 720
i_imy = 576
f_f0 = 8.0
f_pix = 0.0086
f_grat = 600.0
f_rotdeg = 0.0
i_binning = 1
s_comment = Watec 902 H2 ultimate, Tamron12VG412ASIR, f = 8mm
s_infile = calib\cal190320\20190320cal1
s_outfil = calib\cal190320\190320cal
s_linelist = 1
b_sqrt = 0

[Calib]
scalxy = 0.9543847
x00 = 355.5084
y00 = 296.72482
rot = -0.018013034
disp0 = 2.0338101
a3 = 1.557944e-07
a5 = 2.499677e-13
```

Calibration parameters

Run example from Python Demo

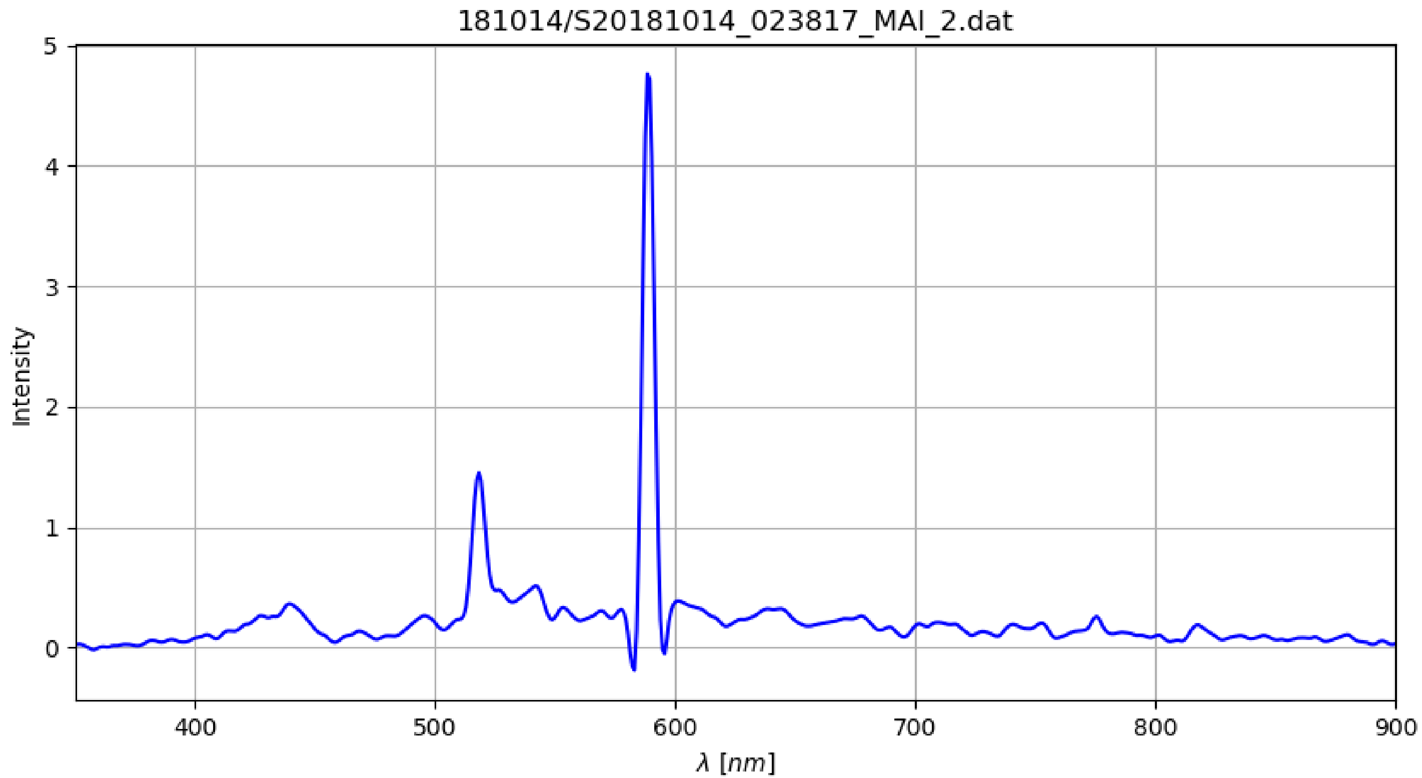
- Input file: M181014/M20181014_023817_MAI_2 (.avi)
- Live demo, use shell 'Python Demo'

```
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 13:32:41) on Wind
ows (64 bits).
This is the Pyzo interpreter.
Type 'help' for help, type '?' for a list of *magic* commands.

>>> run m_pipe62.py
version m_pipe, m_pipefun 0.6.2 0.6.2
***-> path to configuration file: []:
dir .
Configuration m_set.ini read
--> ['Lasercal', 'Calib', 'Fits']
[Lasercal]
- [f_lam0] =  405.0
- [f_scalxy] =  0.92
```

Result

- Calibrated spectrum



Conclusion

- Grating mounted perpendicular to camera axis
- Orthographic image transformation gives linear spectra!
- Python pipeline gives fast spectrum analysys
- Looking for low cost, sensitive, high resolution, high dynamic range video camera

Acknowledgment

- FMA (division of Swiss (Amateur) Astronomical Society) for data, discussion
- Koji Maeda (HD color videos)
- Giovanni Leidi for inspiration to use Python

Thank you!