M_SPEC Python manual

Martin Dubs, FMA

Version 0.9.27

Summary

This manual explains the use of a Python script for the analysis of video meteor spectra. The theory behind it has been described in (Dubs and Schlatter, 2015): meteor_spectroscopy_wgn43-4_2015.pdf).
Until now, the method had been applied successfully to analyse meteor spectra obtained with high sensitive video cams such as the Watec 902 H2 ultimate or colour cameras such as the Sony alpha 7S, which allow video recording. Unfortunately the method required the use of different software for the various steps of the spectra processing, since standard spectroscopy was not adapted to the special requirements of meteor spectroscopy with a fast moving spectral source and nonlinear spectra caused by oblique incidence of the light to be analysed. By combining the different steps of the analysis in a single program, using Python, the analysis should be easier, faster and more reliable. Processing meteor spectra requires the calibration of the camera – lens – grating combination, in order to obtain transformation parameters for the orthographic transformation. This has to be done once for each different camera setup and can be done also with this program.
The processing steps include extraction of the images from the video file, defining a background image, subtraction of background, division by flat image, transformation of the images to orthographic projection, resulting in linear spectra. These are superposed (registered) and converted to 1-dimensional spectra by summing over several rows. The intermediate images of the processing are stored as FITS-images. The advantage of this format is that it also contains information about the image creation and processing, important for spectra of professional quality. The last steps are the wavelength calibration and plotting of the final spectra. The procedure follows the processing as described in (Dubs and Maeda, 2016: Calibration_of_Meteor_Spectra_Dubs_IMC2016.pdf).
Some processing, such as the flat correction and the instrument response have been added in the latest version.
The program is still in a development stage, so the user is asked to report problems and errors, for correction in later versions. Since version 0.9.22 it was changed from a pipeline based approach to a windows GUI, using PySimpleGUI (https://pysimplegui.readthedocs.io/en/latest/) for creating the windows interface. This allows more flexibility in solving the tasks of meteor spectra processing. The style is similar to ISIS, well-known and widely used astronomical spectroscopy software by Christian Buil (http://www.astrosurf.com/buil/isis-software.html).

Introduction

For amateurs, the analysis of meteor spectra has been quite difficult. Standard spectroscopy software is not particularly suited for meteor spectra. Therefore different software had to be used for parts of the processing, which discouraged potential users and made the analysis quite complex (see the manual: processing-meteor-spectra-v151.pdf for the old method). Most of the processing steps are the same for different meteors, therefore a program which combines the different steps can simplify the meteor spectra processing. The program is written as a Python script with a graphical user interface (GUI).

Python was chosen, because

- it contains all the necessary tools to do the analysis
- it finds widespread use in the astronomy community
- it is free
- it runs on different platforms

I was inspired to use Python by Giovanni Leidi, who was giving a talk about the use of Python for the analysis of spectra in the spectroscopy workshop at OHP 2018 (https://www.shelyak.com/ohp-spectro-star-party-2018/).

For the processing of meteor spectra it is necessary to calibrate the equipment beforehand. This is included in this version of the program.

Note of caution: I am new to Python, so the script presented here may not be the best solution. Some things have been done in a complicated way, copying examples from different sources and trying to make it work. I try to improve it for clarity and safety of operation. Therefore I hope you will suggest improvements.

The manual is divided into four main sections and additional features. First a general description of the processing steps is given, which are followed by any processing software. The next section describes the installation of Python, such that the specific Python scripts for analysing meteor spectra can be run. In the third section the determination of the parameters for the orthographic transformation is described. In the fourth section a detailed description how to process a video file into a calibrated meteor spectrum is given. At last some hints and tricks are given for efficient use of the script. The calculation of instrument response and atmospheric transmission correction is treated in a separate manual, “instrument response.pdf”.

M_spec is intended for a first analysis of meteor spectra. For more detailed studies and reanalysis of processed meteor spectra standard spectroscopy such as ISIS (http://www.astrosurf.com/buil/isis-software.html) or Visual Spec (http://astrosurf.com/vdesnoux/) may be used. The registered images and spectra produced by the Python script are in a format, which can be read by these spectroscopy programs.

Processing of meteor spectra, overview

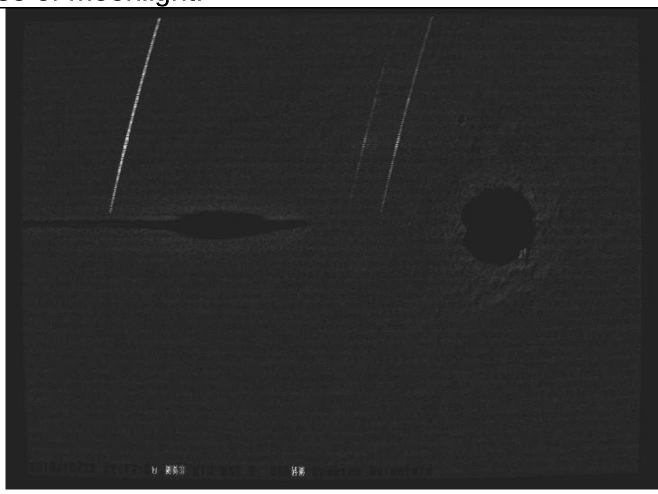
Image extraction from video sequence

The starting point for the analysis is a video file. Typically these are recorded with UFO capture, a program to detect meteors in real time. The program uses a pretrigger to record one second before the meteor appears until the end of the meteor. This video is separated into an image sequence which is stored as *.PNG images. In Python this is done with a call to ffmpeg (https://www.ffmpeg.org/). For the Watec camera, the images are either single frames or the interlaced frames can be separated into two fields, each containing a half frame (even and odd scan lines). These have to be arranged in the correct order (bottom field first or top field first). The use of fields with double image rate is useful for fast meteors, having a large velocity component in the dispersion direction, which reduces the spectral resolution.
For colour videos, there are two processing methods, in colour or black and white. Processing in colour produces nice colour images of the spectra and makes the assignment of spectral lines easier. On the other hand it is three times slower and uses lots of disk space. For the extraction of 1d-spectra the colour images have to be converted to b/w images. Therefore it is usually more convenient to do the whole processing in b/w. Another possibility to speed up the processing is by binning the images in the video extraction in case of 4k videos (2x2, 3x3, 4x4 binning supported at present).

Background image

An important step in the processing of meteor spectra is background subtraction. The video records not only the meteor spectrum, but also other light sources such as sky background, starlight and light pollution from different sources. Most of this does not vary rapidly and can be subtracted before further processing of the spectra.
The background image is made by averaging a number of images without the meteor. For this, the first images of the video can be used. The pretrigger in UFO capture is adjusted normally to about one second. At a frame rate of 25 images per second this means that the first 20 images can be used for the background, leaving some safety margin if the detection algorithm misses the first few meteor images. Averaging of 20 images reduces the statistical noise in the images by a factor square root of 20 or about 4.5. In the case that the video is converted to fields the image rate is doubled and also the number of images for the background can be doubled.
This background image is subtracted from the remaining images containing the meteor spectra. The effect of the background subtraction in the case of moonlight:

Peak image of meteor images as displayed by UFO capture (no background subtraction)



Peak image of meteor images after background subtraction and orthographic transformation (see next section)

Flat image

The processing of astronomical images or spectra includes a step of normalizing sensor sensitivity over the area of the detector. This is done by illuminating the telescope or camera with a uniform (flat) screen. The resulting image, after subtraction of a dark or offset image, gives pixel sensitivity for each pixel. For wide angle images, oblique incidence of the light on the detector and possible vignetting of the light towards the corners of the sensor also contributes to variable sensitivity.

For meteor spectra wide angle lenses are used to cover a large area of the sky. Therefore a reduction of sensitivity away from the image centre is quite noticeable and should be corrected for quantitative (response corrected) spectra. In addition, the meteor spectrum is recorded at different positions on the detector due to the movement of the meteor over the sky. The reference star used for the determination of the spectral response is placed at a different position with different optical efficiency. On the other hand, pixel to pixel sensitivity variations of modern detectors are quite small and averaged out by the movement of the meteor. They can therefore be ignored for the present analysis.

In addition to these sensitivity variations caused by the optics, the image transformation described in the next section changes the size of pixel areas as a function of radius from the image centre. This change of area ratio is partially included in the flat intensity. An additional correction step in the transformation to the orthographic projection can be included in the calculation of the flat, as will be shown below.

The flat correction is applied by dividing the original meteor images by the flat image (pixel by pixel). The numpy package in Python has very efficient procedures for these calculations, which are used to speed up the code.

Image transformation to the orthographic projection

As described in the above mentioned paper: meteor_spectroscopy_wgn43-4_2015.pdf, the recorded spectra have a nonlinear dispersion and are curved differently in different image areas. These spectra are linearized and made parallel by a transformation to an orthographic projection. This transformation can also correct lens distortions. In the same processing step the spectra can be rotated, so that the resulting linear spectra are parallel to the image x-axis. The transformation is described by an axial symmetric radial transformation of the form

$r = r' * (1 + a3*r'^2 + a5*r'^4)$ (plus higher terms in the future), where r and r' are polar coordinates in the original and transformed images. The equation describes the inverse transformation, since for every image point in the resulting image the original coordinates have to be calculated. The polar coordinates are measured from the coordinates of the optical axis (x0, y00 and the polar angle is corrected by the spectrum rotation. The transformation is done in three steps, conversion to polar coordinates, radial and angular transformation, conversion back to Cartesian coordinates. For the interested reader the Python code is given here:

```
x, y = xy.T
x0, y0 = center
y0 = y0 * yscale # the center in the original image has to be scaled as well
# y has been scaled in a previous step with resize image
rp = np.sqrt((x - x0) ** 2 + (y - y0) ** 2)
phi =  np.arctan2(y - y0, x - x0) + rotation
r = rp*(1 + rp**2*(a3+a5*rp**2))
xy[..., 0] = x0 + r * np.cos(phi)
xy[..., 1] = y0 + r * np.sin(phi)
```

The parameters x0, y0, a3, a5 and rotation have to be determined beforehand in a one time calibration for each camera – lens – grating configuration. In addition, in the case of non-square pixels a transformation (stretching in y-direction) by a factor yscale has to be applied before the transformation to orthographic coordinates. When fields (half frames) were extracted from the interlaced video, the images have to be scaled by an additional factor 2 in y-direction, in order to compensate for the missing lines. For the laser calibration, the script M_calib.py has been integrated into m_spec (see tab “Laser Calibration”). The results of this calibration can be

directly used for the processing of meteor spectra, under the condition that the camera setup has not been changed after the calibration.
The radial transformation conserves the dispersion or image scale in the center of the image (at x0, y0). The dispersion disp0 can be used to convert the spectra from pixel to wavelength (more about this see below).

Intensity correction to preserve flux after transformation

As described above, the area of an image element is changed by the transformation. In the centre, the area is conserved (r' ≅ r). For larger r the image is squeezed (for positive a3), the area of the orthographic image is smaller than the original (tangential) image. In order to conserve the light flux (number of photons in a given area), the signal intensity has to be increased by the inverse area ratio. A part of these intensity changes is corrected by dividing by the flat image. This produced a correct flux compensation of the flux measured in the sky to the flux measured on the sphere of radius f within the camera. This sphere is parallel projected onto the detector. For a light ray incident at an angle ρ to the optical axis, the area is reduced by a factor cos(ρ). This factor can be included in the flat correction by multiplying the measured flat with a factor cos(ρ).

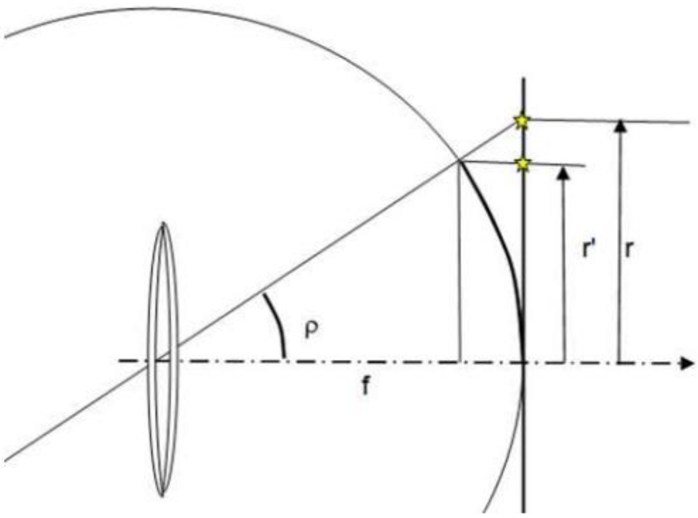


Figure 3 – Relation between the gnomonic projection (r) and the orthographic projection (r') of a point on the sphere with radius f. The prime denotes the coordinates in the orthographic projection coordinate system.

(From: A practical method for the analysis of meteor spectra,
Martin Dubs, Peter Schlatter in: WGN 43:4 (2015), p94-101 https://arxiv.org/abs/1509.07531)
For the mathematically interested reader, a short outline of the calculation is given here.
It is assumed that the grating is mounted perpendicular to the optical axis, which intersects the image plane at pixel position (x0, y0). In that case, both lens distortion and the transformation to the orthographic projection are rotation symmetric about the optical axis and can be corrected in a single step.
For the calculation of cos(ρ) we use the following relation[1]
ρ = arc sin(r'/f), where the same units for r' and f have to be used, for r' in pixels and f in mm →
ρ = arc sin(r'/f/pixel_size). (1)
This gives
cos(ρ) = sqrt(1 - (r'/f/pixel_size)2). (2)

[1] In principle, also the relation ρ = arc tan(r/f) could be used, which however is only valid in the absence of lens distortion. Therefore the relation (1) is used, which is also valid with lens distortion.

Unfortunately, this equation contains the transformed coordinate r‘, which has to be calculated from the coordinate r by the inverse transformation of

$r = f(r') = r' * (1 + a3*r'^2 + a5*r'^4 + O(r'6))$. (3)

The inverse function can also be expressed as a polynomial

$r' = g(r) = r * (1 + b3*r^2 + b5*r^4 + b7*r^6 + \ldots)$ (4)

The inverse polynomial is calculated numerically, by fitting a polynomial in r through points r = f(r') with r in the range (0, rmax), with $rmax = 0.5*sqrt(xmax^2 + ymax^2)$ for the image plane with dimension (xmax, ymax).

Inserting this into the equation for the flux gives:

$flux\_correction(x, y) = sqrt(1 - (g(r(x, y))/f/pixel\_size)^2)$, (5)

with

$r(x, y) = 0.5*sqrt((x-x0)^2 + (y-y0)^2)$. (6)

This correction is required in addition to the flat correction:

I(x, y) = I0(x, y) / flat(x, y), (7)

resulting in a total correction:

I'(x', y') = I0(x, y) / (flat(x, y) * flux_correction(x, y)), (8)

I'(x', y') = I0(x,y) / flat_distortion_corrected(x, y), (8a)

where the distortion correction is calculated first and combined with the flat correction into a new flat:

flat_distortion_corrected(x, y) = flat(x, y) * flux_correction(x, y) (9)

This flat (eq. 9) has to be calculated only once for a given camera configuration and includes vignetting, oblique incidence of light on detector and orthographic transformation correction. It is applied with (eq. 8a) to each meteor image of the video sequence before the transformation to the orthographic projection.

An example for a wide angle lens (f = 7 mm for a Wateg 902 H2 ultimate will be given in a later section.

Both the calculation of a polynomial on a 2-dimensional image array and the division of an image by another image can be done very efficiently with numpy functions in Python, about 100x faster than with a double loop over x and y.

Image registration and summation

In stellar spectroscopy, the position of the star image can be kept in a fixed place, usually defined by a narrow slit, which allows a one-time calibration of the spectrum. With meteor spectra this is hardly possible, the meteor appearing without warning and moving fast enough to make capture and tracking impossible, at least for the amateur. Therefore the spectra are recorded anywhere in the image plane, in a different position from image to image. For the analysis they can be shifted in such a way that they overlap. This works only for parallel spectra which have the same linear dispersion. This is exactly what does the transformation to the orthographic projection. Aligning therefore the zero order will automatically align the whole spectrum. Summing these aligned (registered) spectra allows to improve signal to noise ratio.

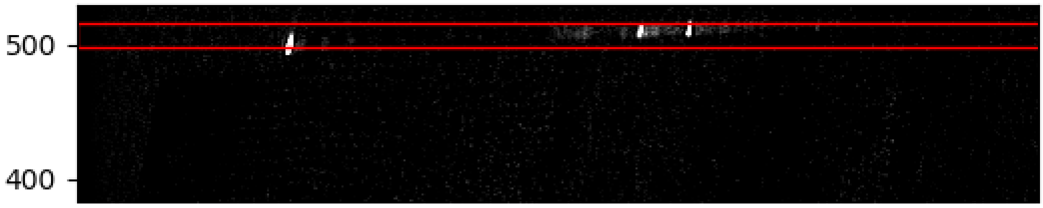
The aligning is done by measuring the position of a prominent line (the zero order if visible, or another prominent line if the zero order is outside the image area) in each image. This position is determined by fitting a Gaussian peak shape to the line and using the centre as line position. In the newest version this Gaussian fitting has been replaced by a correlation method to determine the shift of the meteor spectra, the Gaussian fit can still be used as an option. The line position in the first spectrum is used as a reference, to which all the following spectra are shifted.

In the next processing step a suitable number of registered images are added, which results in a 2-D spectrum image, which can be analysed further in the following steps.

Correction of tilt and slant

Tilt

The deviation of the meteor spectrum from the horizontal axis is called tilt. For the extraction of the spectrum from the 2-dimensional image a number of rows corresponding to the width of the spectrum are added. This works correctly only if the spectrum is aligned parallel to the rows. During calibration this angle has been determined and the images were rotated so that the spectra should be aligned parallel. In the case that the grating has been rotated since the calibration, the spectra will not be parallel to the rows. This can be corrected by applying a “rotation” to the added spectrum. Mathematically a shear is applied; the columns are shifted vertically in proportion to the column number:
dy = tilt * (x – x0)
x0 is selected in the centre of the image.

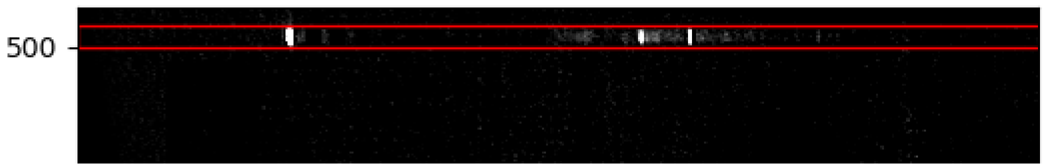


Spectrum before tilt correction

Spectrum after tilt correction, tilt = -0.04, spectrum “rotated” clockwise.
The same effect would be obtained by changing the rotation angle in the configuration file by increasing rot by 0.04 (angle measured in radians)

Slant

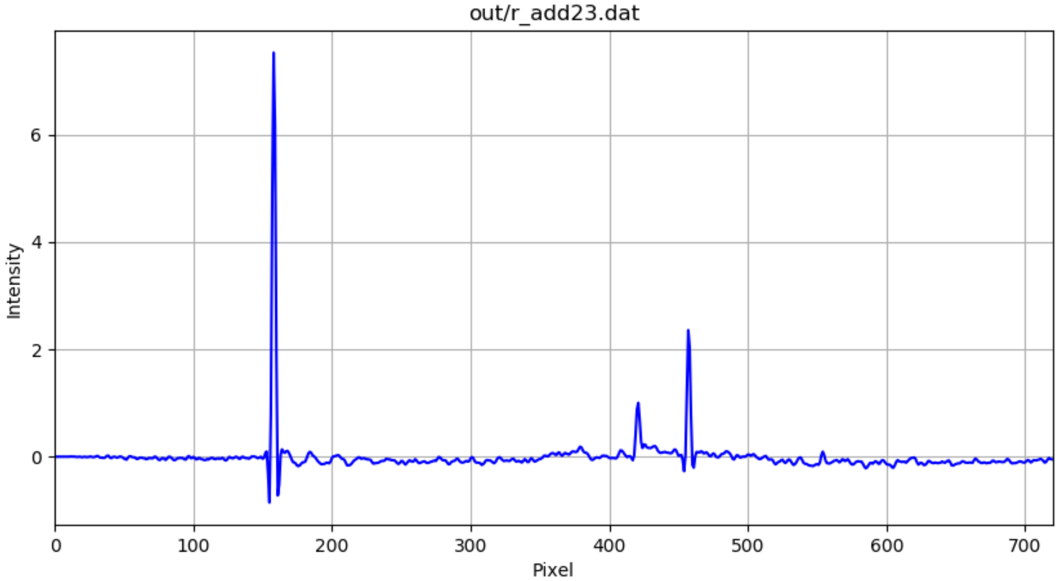
Notice that the spectral lines are not vertical. This is caused by the movement of the meteor. The deviation from the vertical is called slant and is defined as the slope
Slant = dx/dy
For the correction of the slant the rows are shifted horizontally in proportion to the row number.
dx = slant * (y – y0).
y0 is chosen as the center of the selected rows for adding the spectrum. Therefore the spectrum is shifted only minimally by the slant operation.



Spectrum after slant correction, slant = -0.2, spectral lines “rotated” anti-clockwise.
It is advisable to perform the operations in this order and adjust the row selection before the slant correction.
With these corrections, the processing may continue with the spectrum extraction.

Spectrum extraction

The sum image can be converted to a 1-d spectrum by adding a suitable number of rows of the image which contain the spectrum (marked with red lines). The number of rows is determined by the width of the spectrum. In order to optimize signal to noise, one wants to add all the rows which contain the information but avoid the rows which contain only background noise. The choice is somewhat arbitrary and depends on the skill of the operator, but it is not too critical. There exist methods of optimal extraction of the spectrum by measuring the width of the spectrum and signal to noise ratio in comparison with the background. This has not been implemented yet.
The result of this summing over rows is a 1-d spectrum: intensity versus pixel.



With the known dispersion disp0 [nm/pixel] wavelength intervals can be calculated by measuring separations in pixels of prominent features (zero order, Na-line, Mg-line etc.).This helps in a course assignment of spectral features.

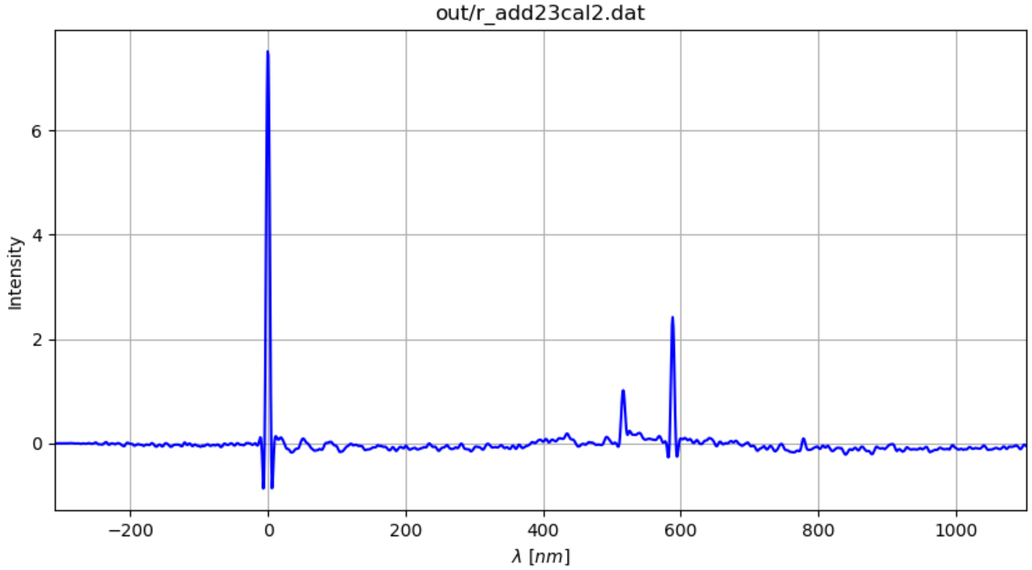
Spectrum calibration

Apart from the instrument calibration, which corrects for lens distortion and transforms to the orthographic projection with constant dispersion, each spectrum has to be converted to a proper wavelength scale, since the meteor and its spectrum can appear anywhere in the image.
In principle, the meteor spectrum can be calibrated in wavelength by setting the zero point at the zero order and convert the pixels to nm by multiplying the distance in pixels of a spectral line from the zero order by the factor disp0, the (inverse) dispersion. In practice, some changes in alignment of the grating, changes of focus and focal length (for zoom lenses) may occur and it is advisable to calibrate the spectrum with the help of known spectral lines of the meteor. Although in theory a linear function should fit the spectrum, in order to correct for small calibration errors a second order polynomial may give a better result. Before fitting a higher order polynomial it is advisable to check if a lower order polynomial works as well. Checking with more lines than required for a fit is always advisable in order to be able to reduce measurement errors or to avoid assignment of the wrong lines.

Plotting the spectra

Data format

Once calibrated, the spectrum can be plotted and saved as *cal.dat file ("cal" is added as a suffix to the filename, to distinguish it from the uncalibrated file). This is a common data format, a text file with two columns for wavelength and intensity, each data point in a separate row. Some programs which read spectra expect a constant wavelength interval; therefore the spectra are resampled with a constant wavelength interval, about half as large as the average original sampling interval (saved as *cal2.dat). Quadratic interpolation is used. This helps to prevent loss of information with the resampling.



Calibrated spectrum, the zero order peak is at 0nm, the Mg line at 517.5 nm and the Na line at 589 nm are also clearly visible.

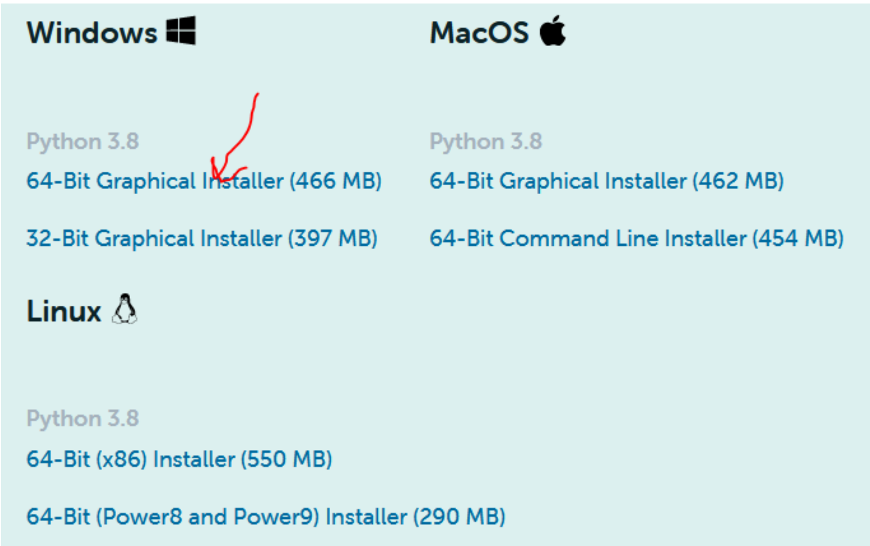
For the calibration in this example a linear fit of the zero order and the two spectral lines of Mg and Na were used.

Python Installation

Python comes in different versions. Apart from the base version it requires additional packages, in particular:
Numpy
Astropy
Etc.
Some versions are not quite compatible. For that reason it is highly recommended to install a pre-packaged version, such as Anaconda, unless you are familiar with Python installing packages. This is a large package, but it contains almost all the required libraries for running the script m_spec.

Installation of Anaconda

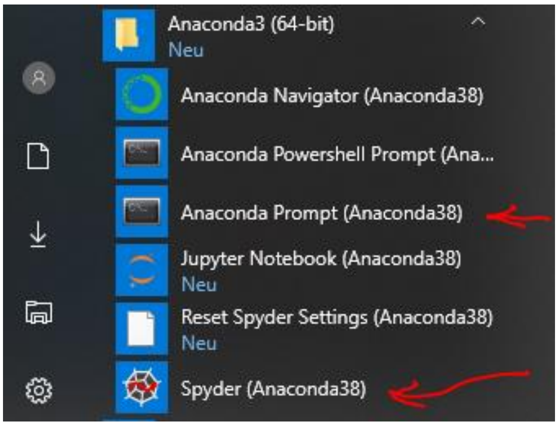
The advantage of using Anaconda is that it comes with a compatible set of almost all of the required packages needed for running the spectroscopy scripts. This saves a lot of trouble with different versions of Python libraries.
Go to https://www.anaconda.com/download/
Install the latest version (64-bit in my case). The Python 3.6 or 3.7 version also works, if already installed (in case you have used my earlier scripts):



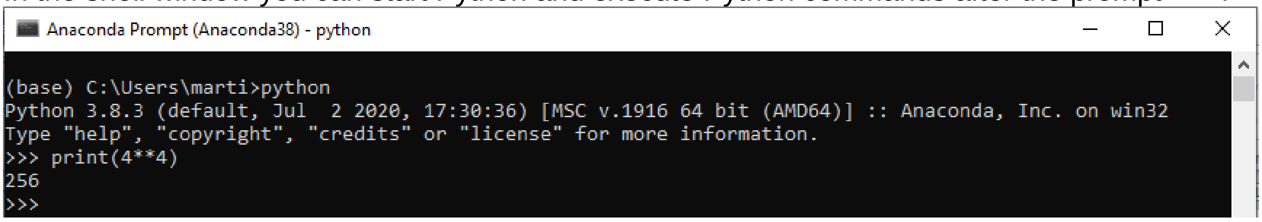
Follow the tips for installation
https://docs.anaconda.com/anaconda/install/windows

- Required system architecture: Windows- 64-bit x86, 32-bit x86; MacOS- 64-bit x86; Linux- 64-bit x86, 64-bit Power8/Power9.
- Install Anaconda in C:\, otherwise you may have problems finding it on your computer (makes it easier to write path to anaconda, but you are free to install it elsewhere).

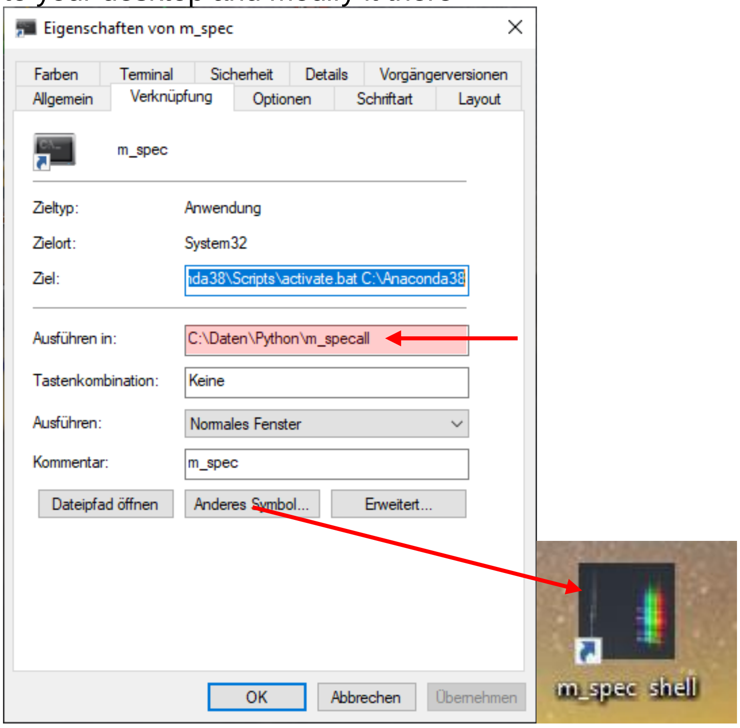
Note: **Install Anaconda to a directory path that does not contain spaces** (such as: “Program Files”) or Unicode characters.
This will take a while.
There are several methods to start Python. You may use Spider (included in Anaconda), which gives you also a programming environment (similar to Pyzo, which I used for the initial development and described in earlier versions of this manual). For writing and editing scripts I use PyCharm, another free tool https://www.jetbrains.com/pycharm/download, which has advanced features such as error correction, renaming variables, version control etc. I do not recommend it for beginners).
Probably the easiest way is to open a shell window with the Anaconda Prompt:



In the shell window you can start Python and execute Python commands after the prompt >>>:

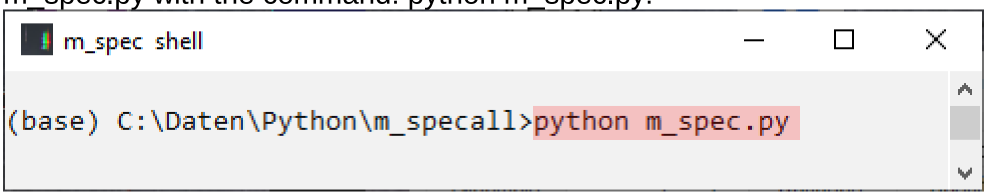


If you want to start Python in a different directory you may copy the Anaconda prompt shortcut to your desktop and modify it there

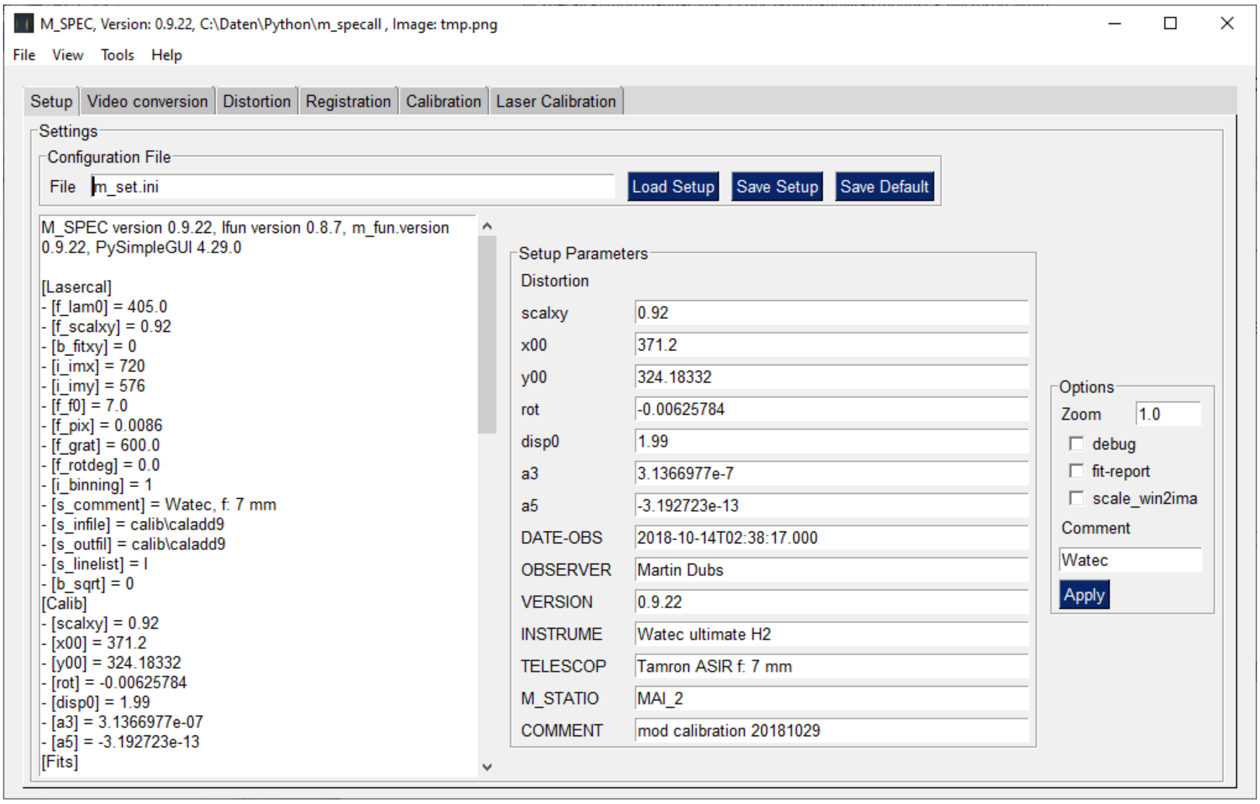


Important here are the command (Ziel) and the working directory (ausführen in). Change the working directory to the directory where your m_spec scripts are located. You may also change the icon and the shell appearance (colors, size, font etc.)

Start the shortcut and the shell should open in the correct directory. You can then start the script m_spec.py with the command: python m_spec.py:



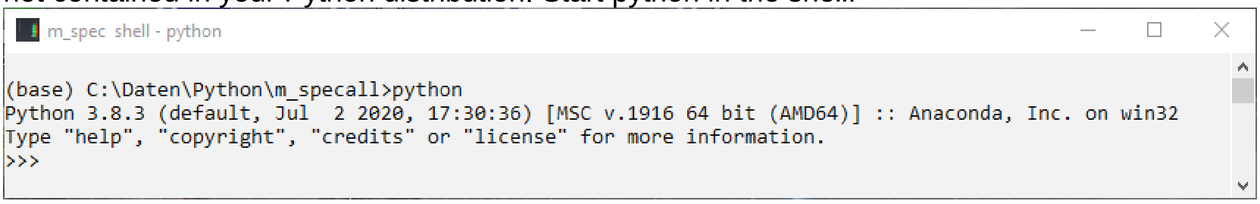
And the program should start:



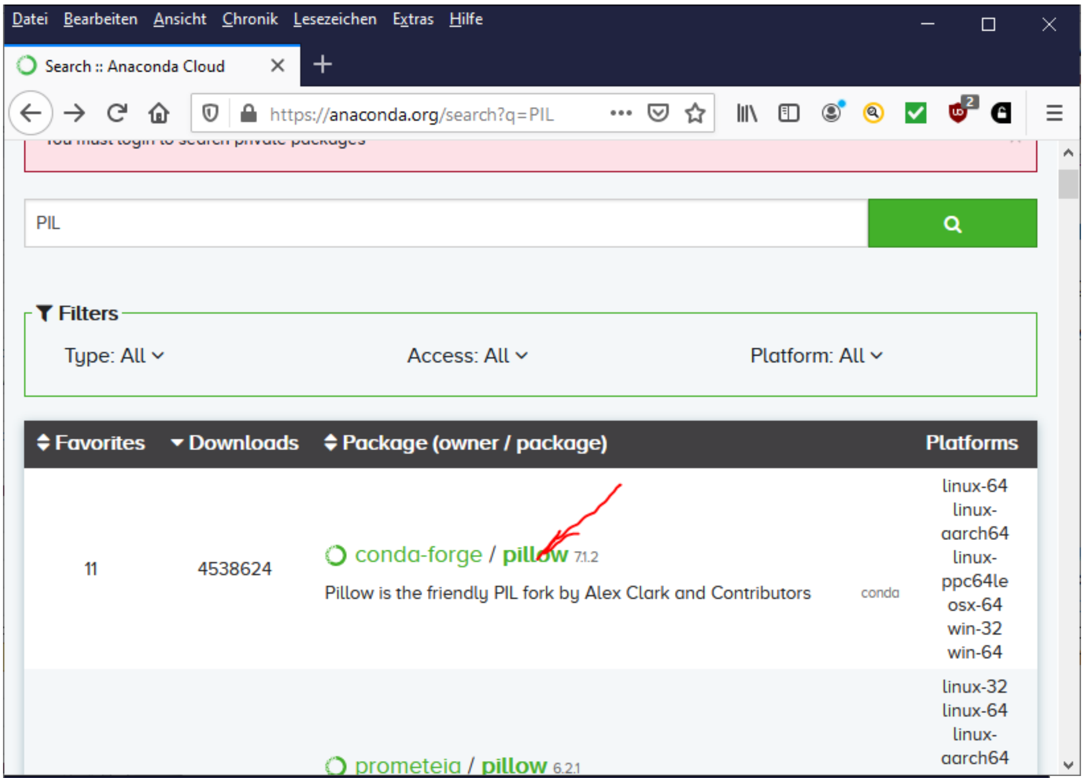
The first time it probably will not start, because some libraries are missing. You also need to install ffmpeg, otherwise the video conversion will not work (see below) .

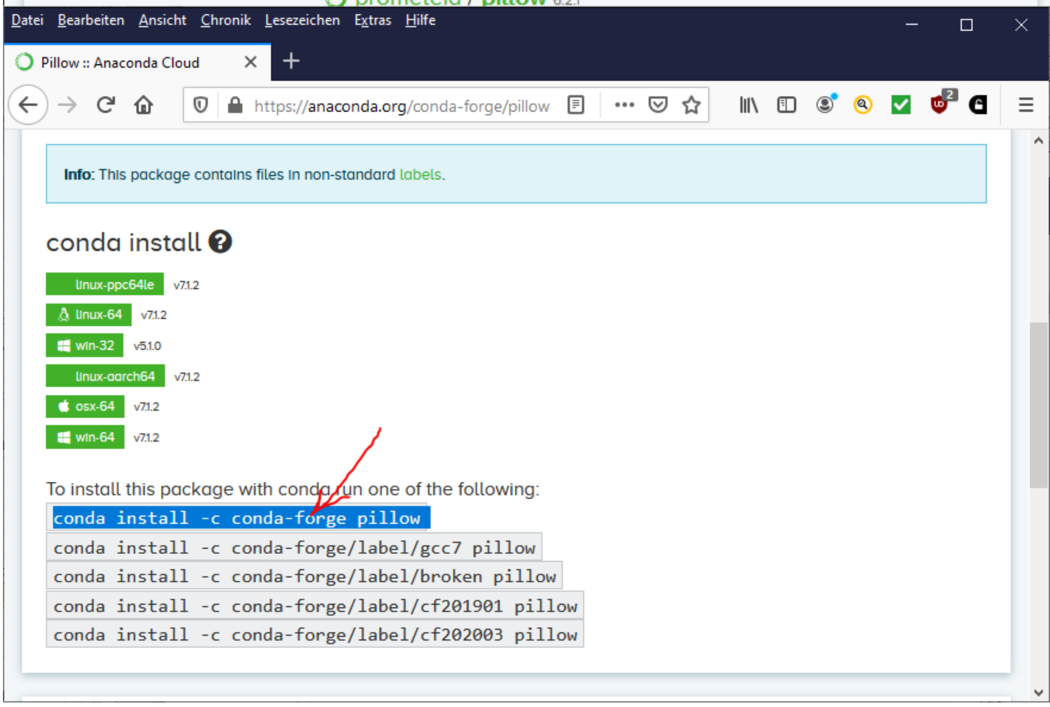
Load libraries

If the script does not run, it may be that a library is missing. You have to load libraries, which are not contained in your Python distribution. Start python in the shell:



With Anaconda the command is, e.g. for the installation of lmfit:
>>> conda install -c conda-forge lmfit
or
>>>conda install -c conda-forge pillow (for the image library PIL, included in Anaconda 3 2020.07, in case you use an older version of Anaconda)
You find the installation command by going to the Anaconda cloud https://anaconda.org/ and search for “pillow”.

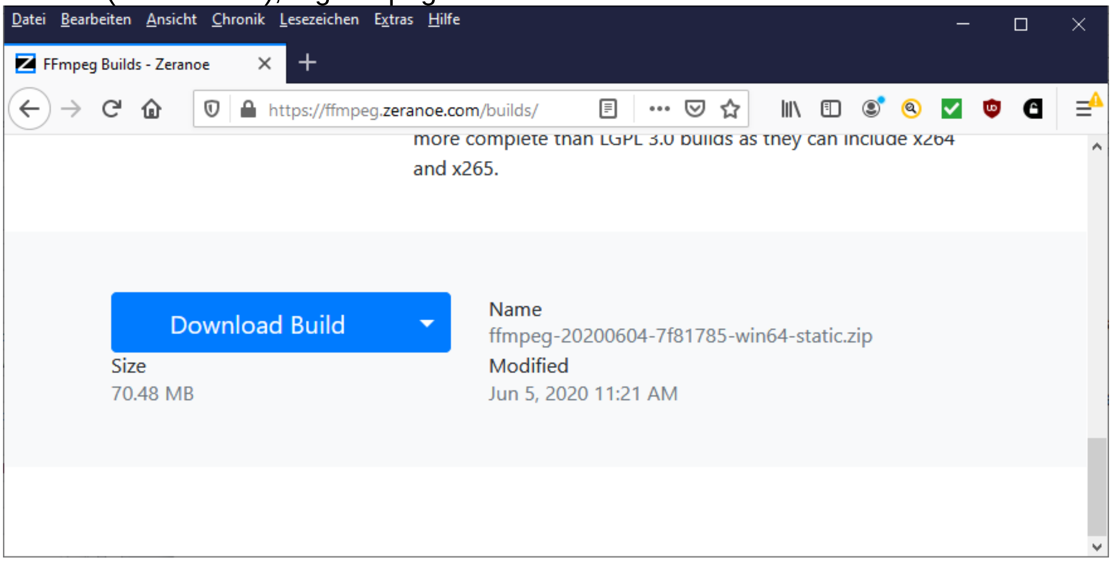




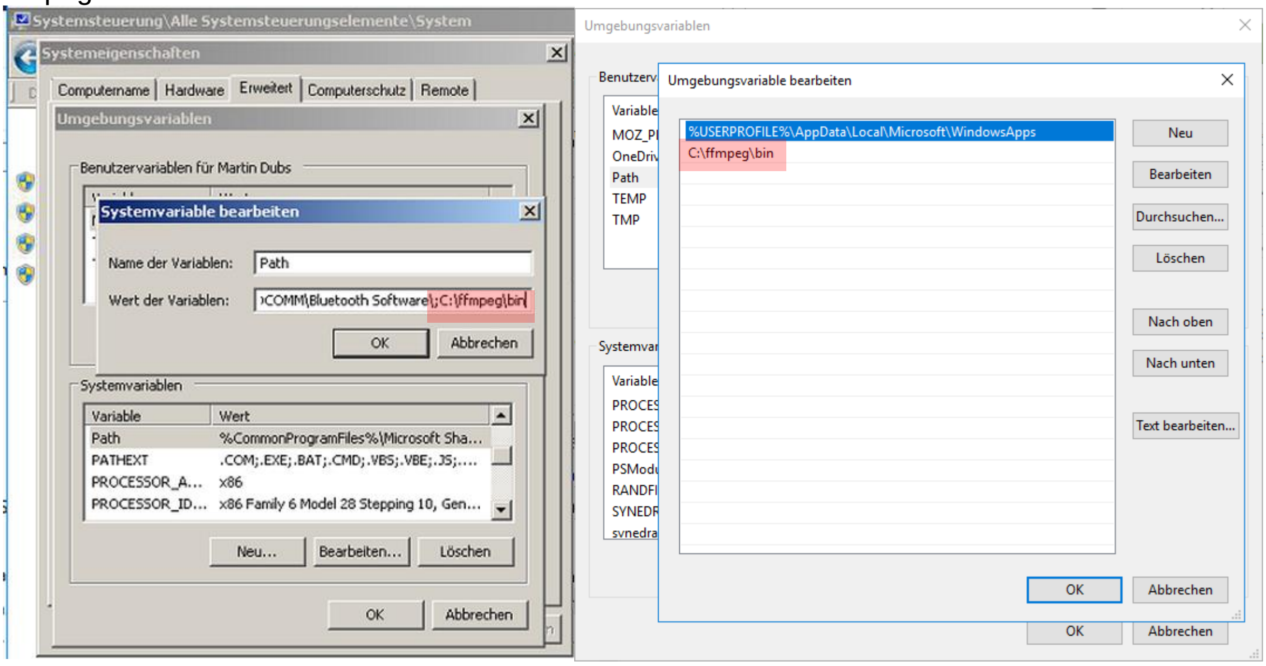
The installation checks for the compatibility with the different already installed packages and makes the necessary updates.
In addition you need to copy other scripts and files (m_specfun.py, m_set.ini etc.). By unpacking the zip-file from Github in a folder you should have all the necessary files for a start.
Fortunately most of the needed library packages are already contained in Anaconda, namely: Numpy, matplotlib, astropy, scipy, skimage (is contained in scikit-image).
For the GUI I use a slightly modified form of PySimpleGUI.py (with titles in the dialog boxes), which is included in my Github folder. I hope the current version is compatible with your Python environment. In case you should have a problem, install it directly from where you installed Python.

Install ffmpeg

The script uses ffmpeg.exe for the conversion of the AVI-files to BMP-images.
Download ffmpeg.exe (I used: https://ffmpeg.zeranoe.com/builds/) select the correct Windows version (32 or 64-bit), e.g. ffmpeg-4.0.2-win64-static.exe and install it.



Add the ffmpeg.exe directory to the Windows system path variable. This is slightly different in Windows7 and 10. In system variable, double click on the path and add the location of ffmpeg.exe:



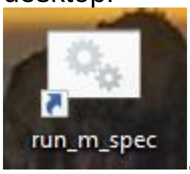
Installing Python in LINUX

The installation under Linux (Ubuntu 18.04) without Anaconda works also.
This has not been tested yet for m_spec, but worked well for m_calib according to a Linux user of Python.

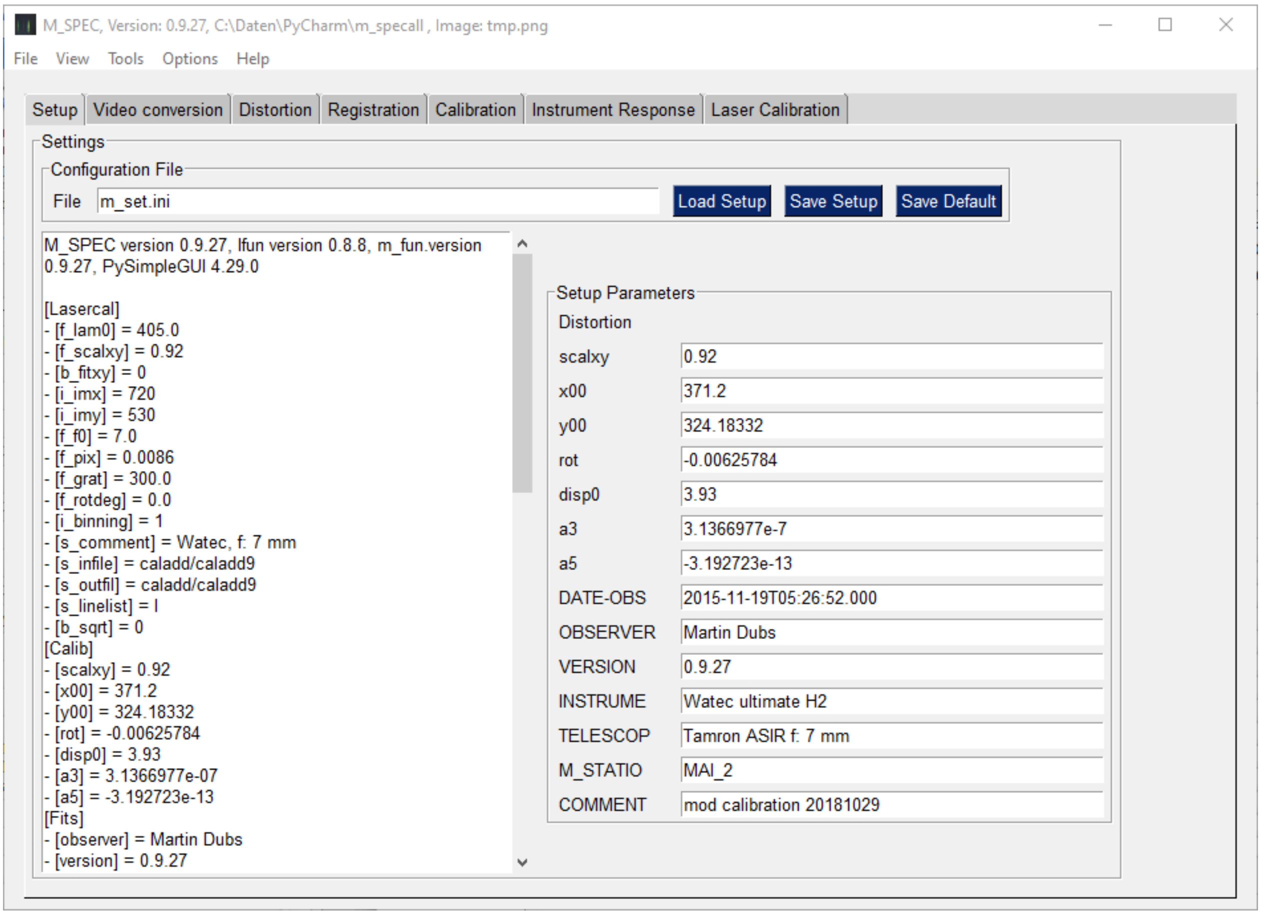
Starting the script

Instead of opening the shell window and start the script, there are other methods to start m_spec.

1st method:
You can modify the shortcut by replacing the Anaconda command:
%windir%\System32\cmd.exe "/K" C:\Anaconda3\Scripts\activate.bat C:\Anaconda3
with a combined command:
%windir%\System32\cmd.exe /C C:\Anaconda3\Scripts\activate.bat C:\Anaconda3 & python m_spec.py
(with “&” you can add more commands to be executed in the new shell. Also change the directory to your actual Anaconda directory. /C closes the command window after finishing)

2nd method:
Write a single line batch file containing:
%windir%\System32\cmd.exe /C "C:\Anaconda3\Scripts\activate.bat C:\Anaconda3 & python m_spec.py"
(Notice the double quotes, which are required to start the new shell window and adjust the Anaconda directory)
Save it as e.g. “run_m_spec.cmd” in the m_spec directory. You may copy a shortcut to the desktop:


. This should work as well. An example is given in the meteor-spectrum-calibration folder. Modify the directories I that file according to your environment. Once all packages are installed, you should be able to start the script and the window for processing the meteor spectra appears:



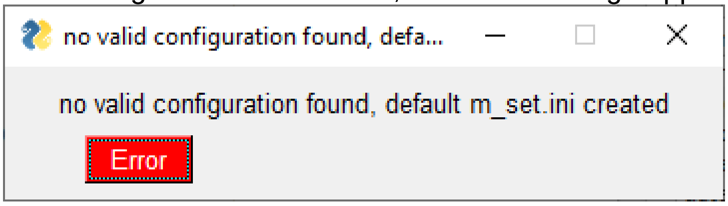
The window allows to select the different processing steps in separate tabs. At startup the tab Setup is shown, where the parameters from the configuration file are displayed. By default, the configuration file m_set.ini from the work folder (where m_spec.py resides) is loaded. This file Is also stored when you leave the program with File – Exit in the menu or you can store changes to it with “Save Default”. This file contains many program settings and, most important, also the distortion parameters from a calibration with s_calib.py or m_calib.py. In the case you have different cameras or setups, it is advisable to store them in separate configuration files.
The content of the configuration file you see in the left memo. Since the same configuration file is used for the calibration and for the spectrum processing, it starts with parameters for the laser calibration, followed by the distortion parameters and fits-header values and finally the settings of m_spec under [Options].
The most important setup parameters are shown in the center frame. Most values can be edited, some values such as DATE_OBS, VERSION and M_STATIO are updated by the program in case you use UFO Capture for recording the meteor spectra (VERSION shows the actual version of the script as written to the fits-header).

Fits-header

A good practice in astronomical photography and spectroscopy is to record information about the image content in the Fits-header, in particular observation time and date, observer, location and equipment. In the case of the orthographic transformation also the used transformation parameters may be useful for a later check on the processing history.
DATE-OBS is a standard keyword; its value is derived from the UFO capture filename.
M_STATIO is also derived from this filename and gives the location of the camera.
D_DISP0 is the dispersion from the laser calibration. If you have a better value from actual meteor spectra, you may enter the correct value here (used for one line calibration).
Others, such as COMMENT you may edit to your taste.

Save configuration

If you save the changes in the default directory you can use them next time you start the script. It is good practice to save them also in the output directory, together with the other processed files, in case you would like to reprocess some files later on.
If no configuration file is found, an error message appears:



Load a valid configuration file or edit the default values with correct values.
If you use different cameras or lenses, make sure you are using the correct configuration. I use different directories for different cameras. Also make a backup of the configuration in a safe place, it is possible to overwrite or erase the file by accident, if something goes wrong.
(The script is not yet foolproof, not even without bugs).
Check the other settings, then save the configuration under a meaningful name.
With this preparation, you are ready to use the script, either for laser calibration and the determination of the parameters for the orthographic transformations, or for the processing of meteor spectra using this transformation. We start with the laser calibration.

Laser calibration of meteor spectra

The laser calibration tab contains the calibration of meteor spectra with laser calibration images replacing the older method with IRIS for the determination of laser line coordinates and EXCEL for the fitting of parameters for the distortion model. These parameters are required for the change from image coordinates to the orthographic projection, which is used to linearize meteor spectra. It is derived from l_calib6.py, described in Calib_Python_manual_2.pdf. It can also be used as a standalone script: https://github.com/meteorspectroscopy/calibrate-spectrum.

Laser calibration images

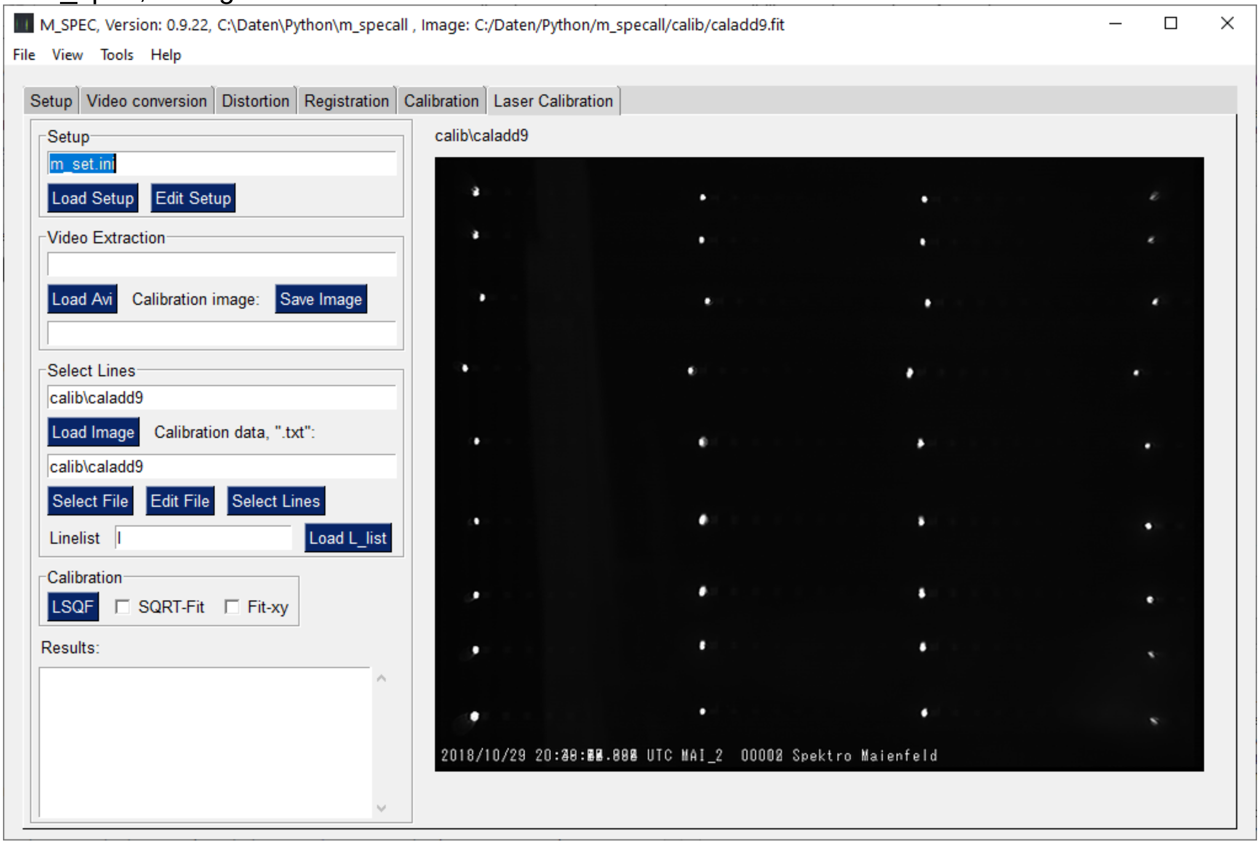
For the calibration of meteor spectra you need some calibration images, containing laser spectra with different orders in different positions of the image area. Ideally they cover most of the image area, in order to avoid extrapolation of the fitting functions. Details are described in the old manual
https://meteorspectroscopy.files.wordpress.com/2018/01/processing-meteor-spectra-v151.pdf
and are not repeated here. The theory is described in the following paper:
A practical method for the analysis of meteor spectra
Martin Dubs, Peter Schlatter (WGN Journal, Ausgabe 43-4, 2015):
Meteor_Spectroscopy_WGN43-4_2015
For simplicity, the spectra are in a single image, obtained earlier with ADD_MAX2 in IRIS (get peak values from several images). In M_CALIB or m_spec, these images can be created directly from video files.
Calibration with a laser is particularly useful for wide angle lenses and/or low resolution spectra and is easier to interpret:

.\calib\caladd9.png
On each horizontal row 1 spectrum is displayed, with 4 laser lines in different orders, for 8 spectra in total. This was recorded with
Camera: WATEC 902 H2 ultimate image size: 720x576 pixels
Lens: Tamron 12VG412ASIR ½", f: 4-12mm f/1.2, used at f: 8 mm
Grating: 50x50 600 lines/mm Thorlabs
Calibration laser: 405 nm wavelength from eBay:
http://www.ebay.com/itm/Diode-Laser-405nm-20mW-Violet-Purple-Laser-Dot-Module-w-cable-13x42mm-3-5-5V-/191174143732?ssPageName=ADME:L:OC:CH:3160

Note: the possession and/or use of this laser may be illegal in your country! If this is the case, use a different light source instead.

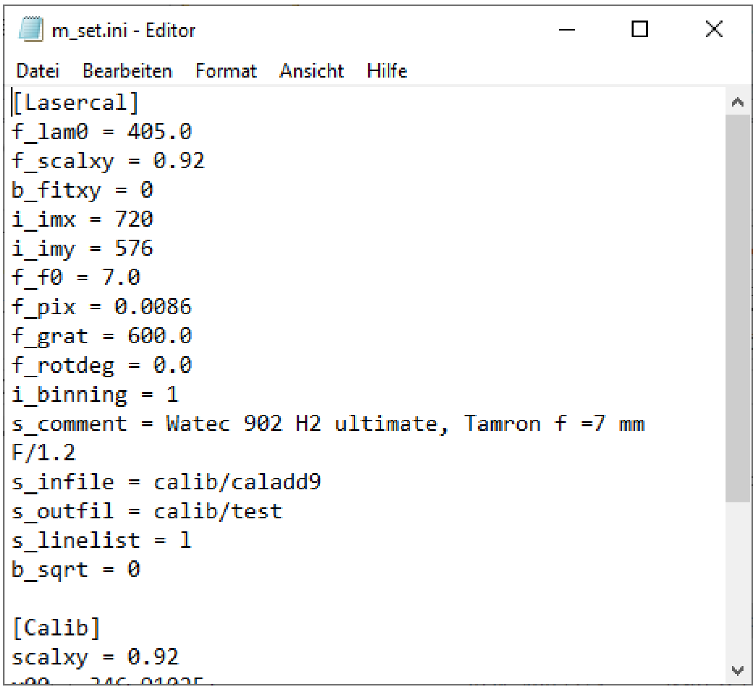
Run the test example

For simplicity we start with the laser calibration as shown in the above figure with the image caladd9.fit or caladd9.png. The creation of this image will be shown later. This avoids uploading large video files. The files are in a folder calib.
Download the test example from the demo zip file or obtain a copy of the files from the author.
In m_spec, change to the laser calibration tab:



Setup

The exact look may vary, depending on the content of the configuration file. At startup, it uses the configuration m_set.ini. Its content is updated at the end of each session with the button Exit, so you can close a session and resume the work later with the same configuration.
The configuration file is a text file like this:




```
[Lasercal]
f_lam0 = 405.0
f_scalxy = 0.92
b_fitxy = 0
i_imx = 720
i_imy = 576
f_f0 = 7.0
f_pix = 0.0086
f_grat = 600.0
f_rotdeg = 0.0
i_binning = 1
s_comment = Watec 902 H2 ultimate, Tamron f =7 mm
F/1.2
s_infile = calib/caladd9
s_outfil = calib/test
s_linelist = l
b_sqrt = 0

[Calib]
scalxy = 0.92
```

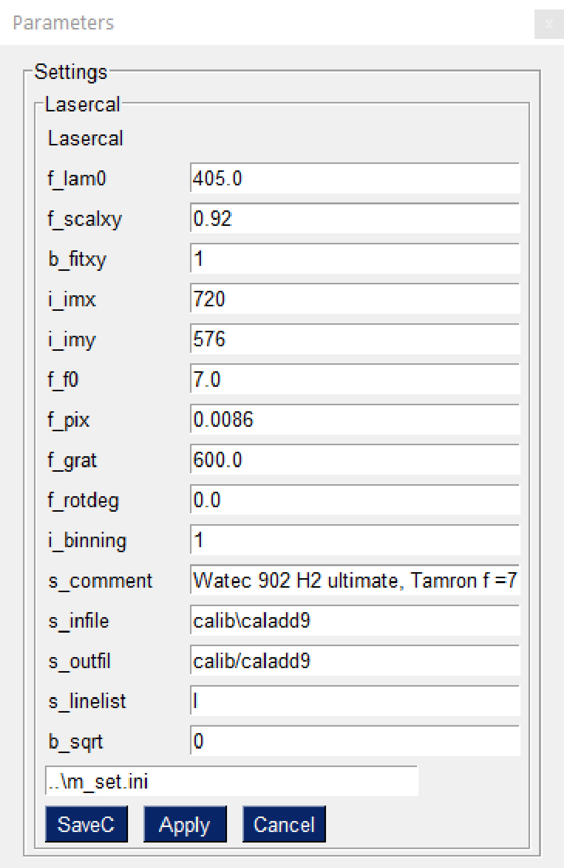
The meaning of the parameters is as follows:

- **F_lam0**: laser wavelength [nm] in case of a calibration with a laser, ignored for calibration lamp. The f_ at the beginning indicates a floating point value
- **F_scalxy** gives the ratio pixel height to pixel width of the camera (or the video capture device, 0.92 is the value determined from UFO analyzer.
- **B_fitxy** is a boolean value, 0 for False, do not use scalxy as a fit parameter, keep it fixed, 1 for True, obtain a better fit with scalxy as fit parameter. If scalxy deviates far from the expected value, set b_fitxy to 0
- The following parameters (imx, imy, f0, pix and grat) are image size, focal length [mm], pixel size [mm] and number of lines /mm of the grating [mm^{-1}]. Its approximate values are used for the start of the fit, but the least square fit works with a wide range of values, takes a bit longer maybe.
- F_rotdeg is the tilt of the spectrum (deviation angle from x-axis, measured in degrees [°].
- **F_binning** indicates if the calibraation image was binned. In that case the distortion parameters are converted to an unbinned image. The spectrum script works with the correct value if it given the correct value of binning (binning is useful for 4k videos, it speeds up the processing considerably, without loosing much precision).
- S_comment gives a short description of the equipment, in case you use different, each with its own configuration file.
- **S_infile** gives the path and filename of the calibration image (without extension). At present images with extension PNG and FIT can be read. PNG is used because it is compatible with the user interface based on PySimpleGUI and FIT is used for internal calculations and as a standard astronomical image format. It contains detailed information about the processing in its header, important for the documentation.
- **S_outfil** (s_for string) gives the path and name of the output of the line positions, as determined by the next step Select Lines (extension .txt added by the script).
- **S_linelist** (.txt) is the name of the file used for the calibration of lamp or meteor spectra. It contains the (wavelength*order) of the selected lines and a comment for each line. A value 'l' (lower case 'L') for the linelist means a laser calibration with lam0.
- **B_sqrt** is a boolean flag: 0 for polynomial fit, 1 for SQRT fit. The latter has one parameter less to fit, is more stable and useful for low distortion lenses and/or longer focal length.
- Under the heading [Calib] follow the results of the calibration, to be discussed later.

All these parameters may be edited with a text editor or easier within the GUI.

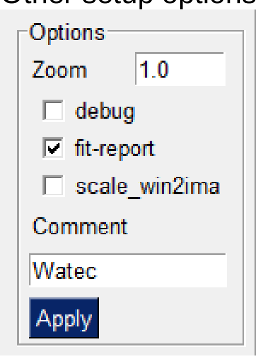
Select configuration

The actual configuration used by the script can be checked with the button 'Edit Setup' in the Laser Calibration tab, Setup at top left.



Important in the setup is the field 's_outfil', which gives the path and filename to the textfile, where the results of the line positions are stored. When starting a new calibration, it might be advisable to give it the same name or a similar name as the image file 'infile' or a name indicating specifics like date or camera setup. With 'Apply' the configuration is used. A different name for the configuration file can be saved with SaveC. With Cancel the configuration remains as before Edit Setup was chosen, changes are ignored. The files s_infile and s_outfil can also be changed in the laser calibration tab with "Load Image" and "Select File". Best to try it out, easier than explaining!

Other setup options may be changed in Menu – Options – Setup Options:



Debug is mostly for testing purpose. Fit-report gives detailed information about the least square fit in the shell window. You probably do not need it either. If you check scale_win2ima the image is scaled by the factor zoom and the window size changed that the image fits inside. Use a zoom value according to original image size and screen size. It works only after closing the program and restarting it (peculiar to PySimpleGUI). If this is not checked, the image fits the available space in the window, the zoom factor is ignored (For the determination of line position, the original image is used, with the binning as given by i_binning).

Select points of the different orders

Once the configuration has been set, we can load a calibration image for analysing with “Load Image”.
If you would like to do another analysis, you can type the new filename in the input field Calibration data or select an existing file with “Select File”.
Start the calculation of the line positions with ‘Select Lines’. A new window appears:

- Start with the lowest order at the left and click with the mouse on the spot you would like to select:

  The program calculates a Gaussian fit of the spectrum and saves the result. A yellow circle shows the selected point. A label shows wavelength*order. Continue with the next point. Since all lines have the same spacing, it is not important if the first point is actually the zero order. This is different with a calibration lamp, where the lines have to be selected precisely as given in the list.
- If you have selected all orders of one spectrum, press “OK”, the coordinates and widths of the different orders will be saved to the text file.

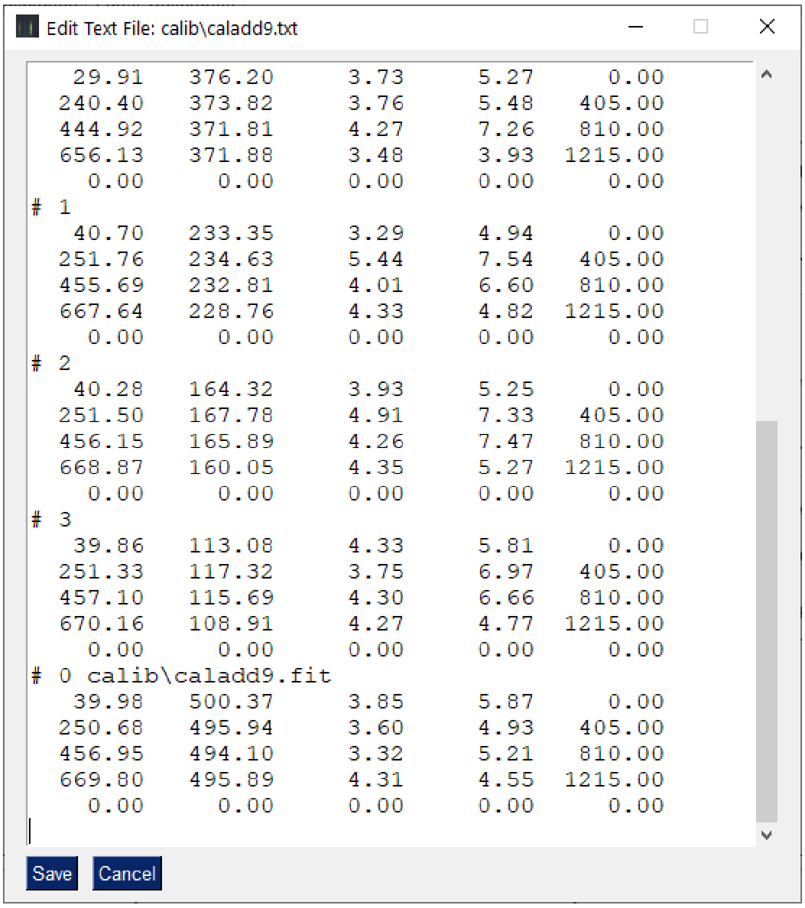
- If you made a mistake, press ‘Cancel’, to start with the same or another spectrum
- With ‘i(ncrease)’ and ‘d(ecrease)’ you can adjust the image contrast
- With ‘Skip Line’ you can skip a line (not needed here,but useful if you do not see a line corresponding to the linelist, or when it overlaps another line)
- Press “Finish” after the last spectrum
  In the Results window it should say ‘Finished, saved caladd/caladd9.txt’.

Note:
If you have problems with the selection of points, you may change the size of the rectangle, in which the Gaussian peak is searched and fitted (default is a rectangle with corners x0, y0 +/- 10 pixels. You may draw a rectangle in the image by selecting one corner and draw with the mouse down to the opposite corner and release the button there:

If the rectangle is too small, you may miss the point when selecting it, if the rectangle is too large, the Gaussian fit may fail. You will get some error messages in the shell, try with another size! Drawing the rectangle does not select the point. Type ‘r’, then start a new spectrum. Notice, this is still a test version.

Inspection of the calibration text file

With “Edit File” you can check the measurements.
It should look like this:




```
   29.91   376.20     3.73     5.27     0.00
  240.40   373.82     3.76     5.48   405.00
  444.92   371.81     4.27     7.26   810.00
  656.13   371.88     3.48     3.93  1215.00
    0.00     0.00     0.00     0.00     0.00
# 1
   40.70   233.35     3.29     4.94     0.00
  251.76   234.63     5.44     7.54   405.00
  455.69   232.81     4.01     6.60   810.00
  667.64   228.76     4.33     4.82  1215.00
    0.00     0.00     0.00     0.00     0.00
# 2
   40.28   164.32     3.93     5.25     0.00
  251.50   167.78     4.91     7.33   405.00
  456.15   165.89     4.26     7.47   810.00
  668.87   160.05     4.35     5.27  1215.00
    0.00     0.00     0.00     0.00     0.00
# 3
   39.86   113.08     4.33     5.81     0.00
  251.33   117.32     3.75     6.97   405.00
  457.10   115.69     4.30     6.66   810.00
  670.16   108.91     4.27     4.77  1215.00
    0.00     0.00     0.00     0.00     0.00
# 0 calib\caladd9.fit
   39.98   500.37     3.85     5.87     0.00
  250.68   495.94     3.60     4.93   405.00
  456.95   494.10     3.32     5.21   810.00
  669.80   495.89     4.31     4.55  1215.00
    0.00     0.00     0.00     0.00     0.00
```


- The columns denote x-coordinate, y-coordinate, FWHM(x) and FWHM(y) of the spots, as determined by the Gaussian fit
  The last column is wavelength*order (added for analysis of calibration spectra)
- The lines with # are comments and ignored for the fit.

- Each spectrum ends with a line of zeros as separator
- Additional lines of zeros in between are ignored, also spectra containing only one line (there is nothing to fit there)
- If you forgot one or more spectra, you can run the script again and add the missing spectra (with each run, data are appended to the old file).
- If you recorded one spectrum twice or entered wrong data points, you may erase these in the text file manually and save the file again for the following step.
  This can be done directly in the Edit File window, followed by 'Save'

Least square fit

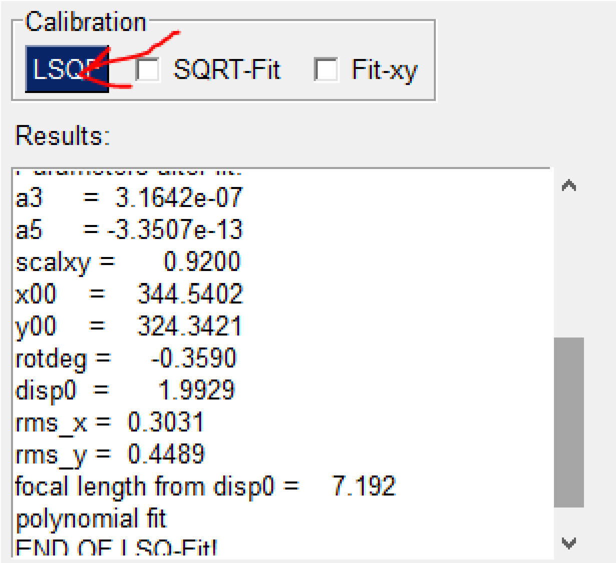
For the determination of the distortion parameters, a linear dispersion law is postulated in the transformed image
(x' = x0 + i*lam0*disp0, i = 0, 1, 2, … for laser calibration, used in old version
In present version:
x' = x0 + lam[i]*disp0, i = 0, 1, 2, … , where lam[i] = wavelength*order of line[i] from line list, both for laser calibration and spectral line calibration
y' = y0
from these coordinates the transformation to the image coordinates (as required in the coordinate transformation of the meteor images) is calculated by a transformation to polar coordinates around the image axis at coordinates (x00, y00), rotation of the image by an angle rotdeg (converted to rot, angle in radians, rot = rotdeg*Pi/180)), transforming the radial coordinate by
$r = r' * (1 + a3*r'^{2} + a5*r'^{4})$ (plus higher terms in the future),
transforming back to Cartesian coordinates with the axis remaining at (x00, y00)
For those interested in the details, here is the corresponding code from the Python function.
(x0, y0s) correspond to (x', y').
xyr +xy00 corresponds to (x, y) in the calibration image, to be fitted to the coordinates in the calibration file (testcal.txt in the example)

```
    xy00 = [x00,y00]
    xyl = [x0s+lam/disp0,y0s] - np.array(xy00) # coordinates of laser lines wrt, optical axis
    r = np.sqrt(xyl[0]**2+xyl[1]**2) # polar coordinates
    phi =np.arctan2(xyl[1],xyl[0])   # polar coordinates
    phi += rot                       # apply rotation
    r = r*(1 + a3*r**2 + a5*r**4)    # transform radial coordinate
    xyr = np.multiply(r,[np.cos(phi), np.sin(phi)/scalxy]) # return to cartesian coordinates
    return (xyr + xy00)
#------------------------------------------------------------------------------
```

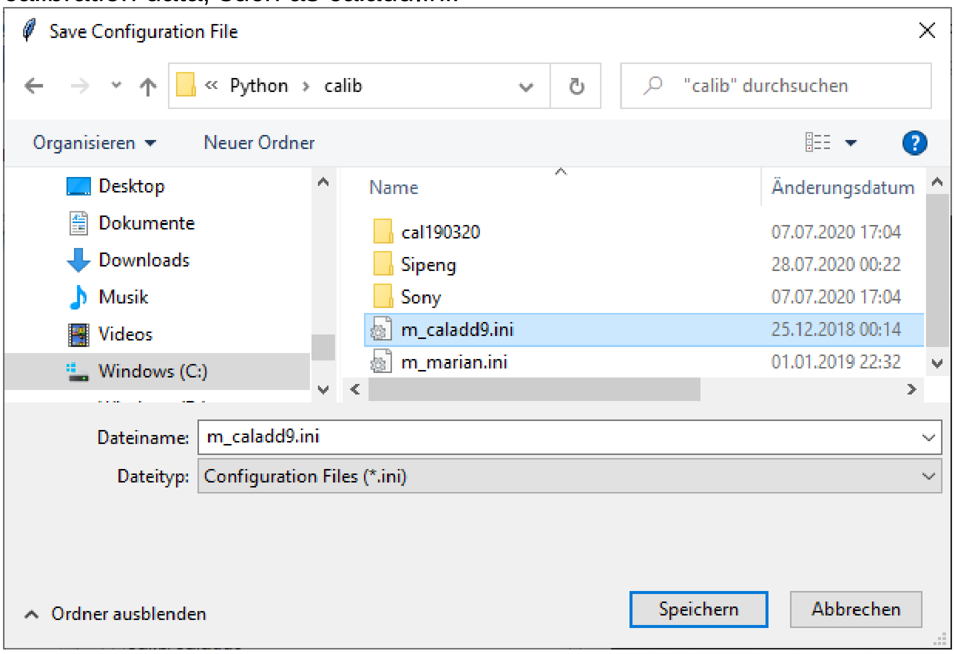
NB: For the sqrt-fit the following equation is used for r':
$r = r' * (1 + a3 * r'^{2} + 1.5*a3^{2} * r'^{4})$

Run the fit for the example caladd9.txt with m_calib.py

Let us assume that you have edited the file caladd9.txt and corrected all the errors (wrong points, wrong order, etc.) and you would like to start the least square fit with these data.
Select 'LSQF'.
After the fit the results of the least square fit appear in the log window:



At the same time they are stored in m_cal.ini in the working directory (in order not to overwrite m_set.ini, which may be used as a backup). If you like the result and want to use it with m_spec, you may save the new configuration under m_set .ini or another meaningful name (e.g. m_set_yymmdd.ini or m_set_sony24mm.ini etc.)
These parameters you may use to transform your meteor spectra to the orthographic projection. Notice that the fit runs quite fast, the Levenberg Marquard algorithm is very efficient for this kind of problem. In addition the results of the fit are shown in graphical form (helpful, if you have made any errors in the assignment of any lines).
I recommend to save the configuration under a suitable name, perhaps in the directory with the calibration data, such as caladd.ini:



The results are also stored in the Logfile m_spec_yymmdd.log
You can check the logfile with Menu – Tools – Edit log file:

Edit Text File: C:/Daten/Python/m_spec200810.log

```
a5      =  9.2516e-14
scalxy =      0.8219
x00    =    345.7423
y00    =    327.4165
rotdeg =     -0.3198
disp0  =      1.9871
rms_x =  0.3308
rms_y =  0.4239
focal length from disp0 =    7.213
for sqrt fit: feff =      12.20

2020-08-10 20:56:45,432 outfil: calib/caladd9 START LSQF
2020-08-10 20:56:45,831  configuration saved as m_cal.ini
2020-08-10 20:56:45,831 Result LSQF:
2020-08-10 20:56:45,831 Watec 902 H2 ultimate, Tamron f =7 mm F/1.2
calib/caladd9.txt
Parameters after fit:
a3      =  2.9510e-07
a5      = -1.9674e-13
scalxy =      0.8523
x00    =    345.0316
y00    =    327.3211
rotdeg =     -0.3320
disp0  =      1.9905
rms_x =  0.3227
rms_y =  0.4214
focal length from disp0 =    7.201
polynomial fit
```

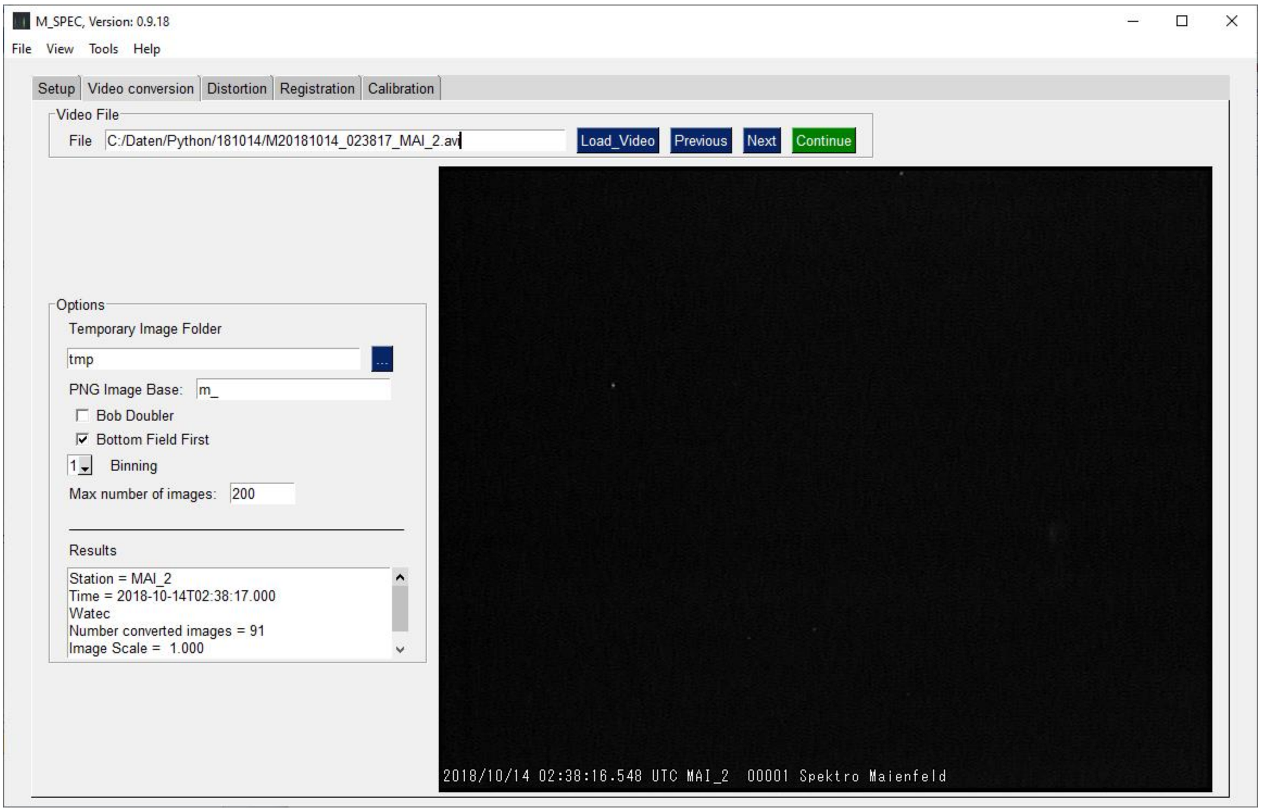
Save Cancel

If the results look ok and the rms-errors (measured in pixels) are reasonably small, you are ready to process actual meteor spectra.

Processing of meteor spectra, step by step

Conversion of *.AVI to PNG

With the correct configuration we can start the meteor spectra processing, first converting the video file into a sequence of *.PNG images. This is done in the next tab, “Video conversion”:



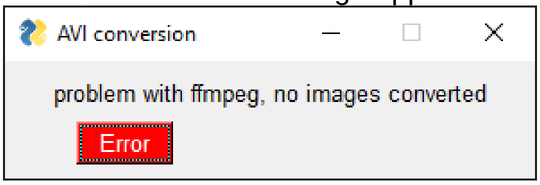
Before you load the video file you have to set the options!

- You can select a folder to store the images. Since they are used only once they are stored in a temporary folder tmp.
- You can also choose the filename of the images, the index starting from 1 is added: m_1.png, m_2.png, …
  This is the same naming convention as in IRIS, so you may use these files also in IRIS, for processing steps not included in this pipeline. You can change these names, but this is not really useful as these are only intermediate files you need once or a few times if some errrors occurs later in the processing.
  The images are erased before a new video conversion, so do not select a folder with images you want to keep.
- With Bob Doubler you can read half frames for interlaced videos, in this case you also have to choose which field is read first. For Watec cameras this is usually Bottom Field First. With the wrong selection, the meteor jumps around and it is difficult to register the images correctly. This increases time resolution and therefore spectral resolution if the meteor has a velocity component parallel to the dispersion direction (recommended for WATEC).
  Important: the calibration parameters are adjusted automatically; in particular the factor scalxy is multiplied by 2 to correct for the different aspect ratio.
- 2x2 binning is useful to reduce the file size for 4k images (for 4k color cameras the color pixels are often interpolated, so you loose little information by reducing the size to 2k).
  Important: The distortion parameters are not adjusted automatically, it is assumed that

you use the same binning as for the calibration (keep 2 sets of calibration parameters for 2k and 4k images!)

- It is also possible to limit the length of the video sequence. This is helpful in case of a corrupted file or when you want to save time or disk space. You may use Virtual Dub (https://www.virtualdub.org/) to find the relevant meteor images and convert only these.

With the options set, you can load an AVI-file. The conversion starts automatically and the button Continue turns green when completed. For this step ffmpeg.exe has to be installed, otherwise an error message appears:

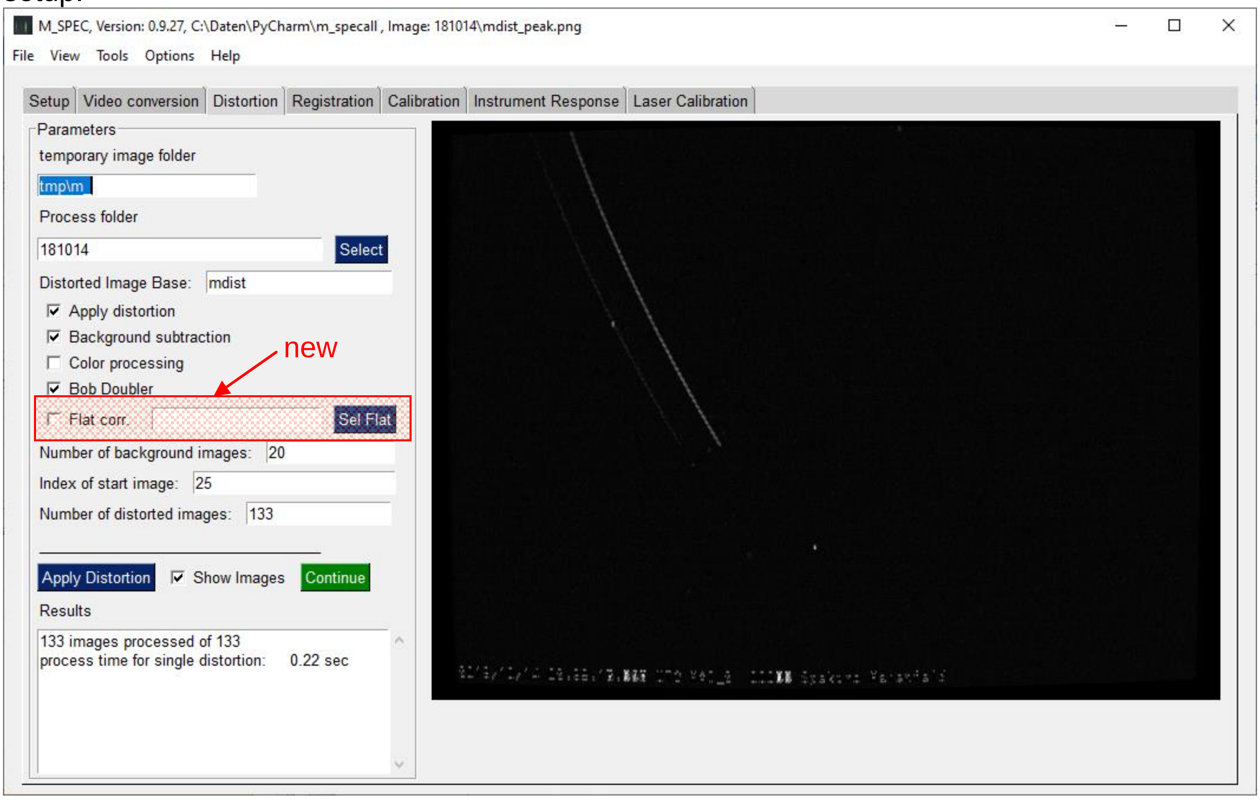


Notice that the path of the selected file is written with forward slash: /, so users of LINUX should not have a problem with it. In most cases also a backslash \ is accepted. I personally use a different directory for all meteors of a day, with the image files having standard names. In case I need the images again, I also store the video clip and related files (*.xml, *.jpg from UFOCapture).

In the memo at left some details of the conversion are given. With Continue you go to the next tab, “Distortion”. Before you do that you may look at the images with the very basic image browser. With the Next button you jump from frame 1 to 25 and then advance in steps of 1. With Previous you go backward in the sequence. This is because the first 25 images only contain background and the meteor appears around frame 25. With Bob Doubler selected, the Next button selects field 50 after the first field, corresponding to frame 25.

Distortion

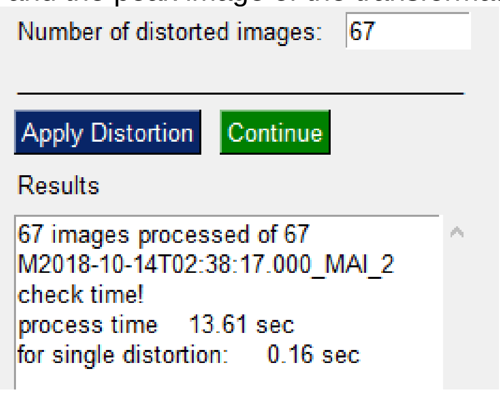
In the distortion tab you can apply the transformation to an orthographic projection, which linearizes the spectra. For this to work, you have to set the correct distortion parameters in the setup.



In addition you have to set some additional parameters in the frame at left.

- Temporary image folder, PNG image base: if you have just done the video conversion, these values are pre-set; otherwise you can choose them to convert other images. Note that they should conform to the naming convention, file base index .png, e.g. m_1.png, m_2.png, …
- Process folder and distorted image base: Folder and name for the processed images. Naming convention as above. Notice that files with the same name are removed from this folder before a new distortion is performed. This prevents a mix of different spectra. For the process folder I use a separate folder for each date, where I also store the AVI file(s), setup-file, processed spectra and logfile(s) for later reference.
- Apply distortion, background subtraction: These are normally checked, but you can do only one operation if desired.
- Color processing is mostly used for making nice images, the spectra are evaluated from b/w images. Use b/w, it is three times faster than colour processing and for the extraction of the spectra you have to convert the images anyway.
- Bob Doubler: This is set the same value as in Video Conversion. Be careful to set the right value when you do a distortion correction at a later time. It is advisable to do the distortion right after Video Conversion.
- **New in version 0.9.27:**
  Flat correction: If this is checked, the flat file in the text field next to it will be applied to each image before the application of the distortion, i.e. each image is divided by the flat field image and then the distortion is applied, if the corresponding checkbox is selected. If no flat file is selected, the checkbox will be ignored.
  Flat file: Name of the flat file *.fit, make sure the flat file has the same size as the meteor images. Only black/white flats are supported. A suitable flat can be created with Tools – Image Tools (see below in the Tools section).
  Select Flat: Button to select the desired flat file.
- Number of images: For background and start images use the default values. The number of distorted images is calculated from the number of available images after video conversion. Select a reasonable number. It will be adjusted to a lower value, if not enough images are found.
- Background images:
  As described in the general processing section, a background image is created from the first second of video. The default value of the number of images used for the averaging is given for a pretrigger of 1 sec and a frame rate of 25 images/sec. With bob doubling, the image rate (fields/second) is doubled, therefore also the double number of images can be used for the background. Make sure that the default value does not use any images with the meteor appearing early.

Once everything seems ok you press the button “Apply Distortion” and wait for the end of the processing. This is indicated by the button Continue turning green (means enabled) and some information is output in the result memo at left bottom:
and the peak image of the transformation is displayed:

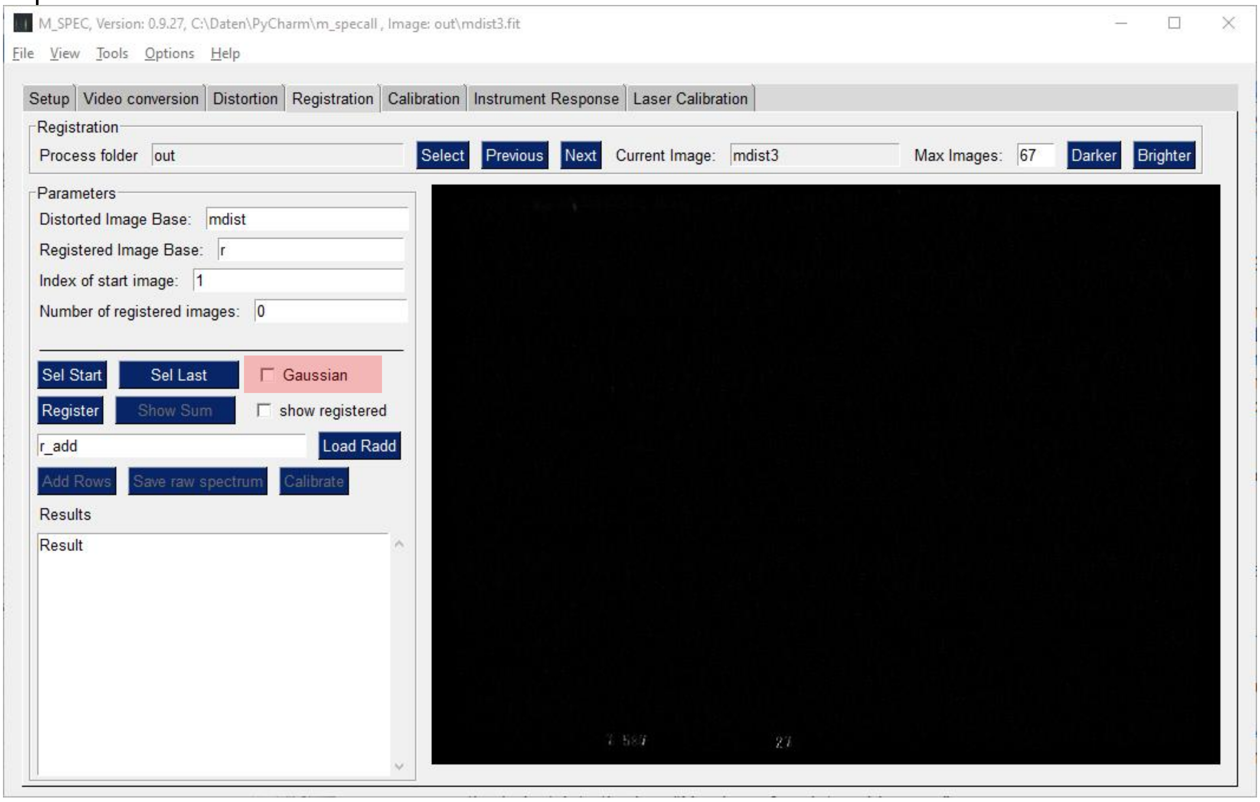


The process time is the time between the start of the background image and the end of the transformation. The log shows also the average process time per image.

The contrast and brightness of the peak image has not been adjusted, so you should not be disappointed by this result. If you continue you can adjust the brightness of the transformed images in the registration tab.

Registering the meteor spectra

As explained above, the meteor spectra of the different images can be added after registration, because they are all aligned parallel to a common dispersion direction and they all have the same linear dispersion. This is because of the transformation in the previous step. The meteor movement causes a displacement of the spectra. This is removed by registration, which will be explained in detail.



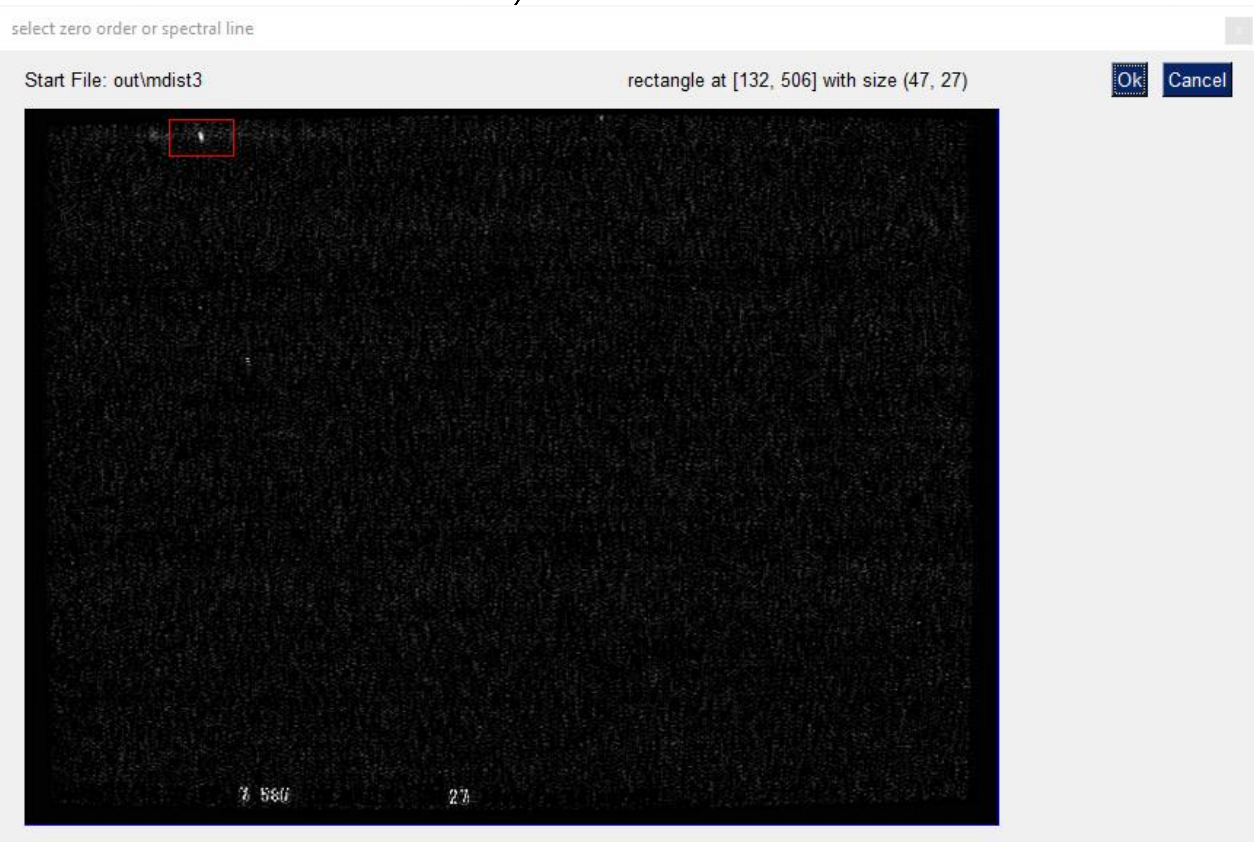
The idea is to identify a clearly visible line in the meteor spectrum and shift the following images so that this line is in the same position in all images. The position was determined with a Gaussian line fit, similar to the determination of line position in the calibration script. The problem is that this determination of line centre is not fool-proof. It has been replaced by default by determining the shift with a cross correlation method. In case you prefer the old method, you must check the box Gaussian (marked in red in the image above. Important is the selection of the first image used for registration. If the 1st image is not useable a different one may be chosen.

The registration frame at the top is a simple image browser, which helps to select the correct images.

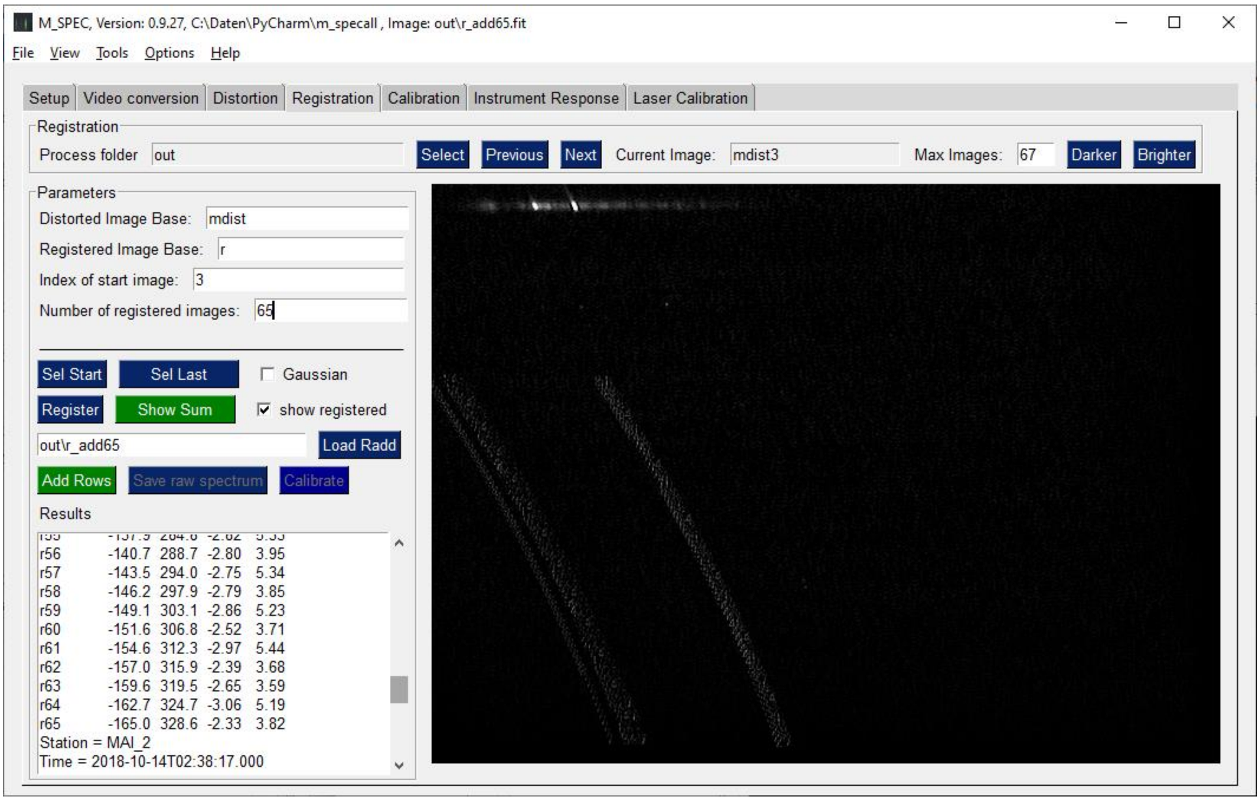
- The process folder and file base are pre-set from the distortion tab. If you want to register some other files you can choose the folder with the button "…" and change the base file name of the images you would to register (here mdist, for the files mdist1.fit, mdist2.fit, …, with the now familiar file naming convention)
- You can also change the name of the registered file base, r for r1.fit, r2.fit, …
  This is useful if you want to register several spectra in the same folder or different meteor images from the same meteor
- With the buttons "Next" and "Previous" you can click through the images to find a suitable start image, here mdist3 was selected
- With the buttons "Brighter, "Darker" the image contrast can be changed

- The current index is shown and the number of images can be set. If this is larger than the actual number of images it is adjusted automatically.
  Notice that the numbering of the meteor spectra mdist is offset from the image number in the original video file by: first meteor image – 1

If you have found the correct start image, press the button “Sel Start” or enter the image number (here 3) into the text box “Index of start image”. Browse through the images with next until your last image to register and press “Select last” or enter the corresponding number of images (not the index) into the box “Number of registered images”.
If you check the box show registered you may follow the registering in action or browse through registered images.
With all settings done you can start the registering procedure by pressing “Register”.
A new window appears with the start image. Here you can draw with the mouse a rectangle enclosing the zero order or whatever region you would like to use for the registering (You may also select several spectral lines in a wider rectangle, this reduces random errors, but only if no artefacts disturb the correlation function):



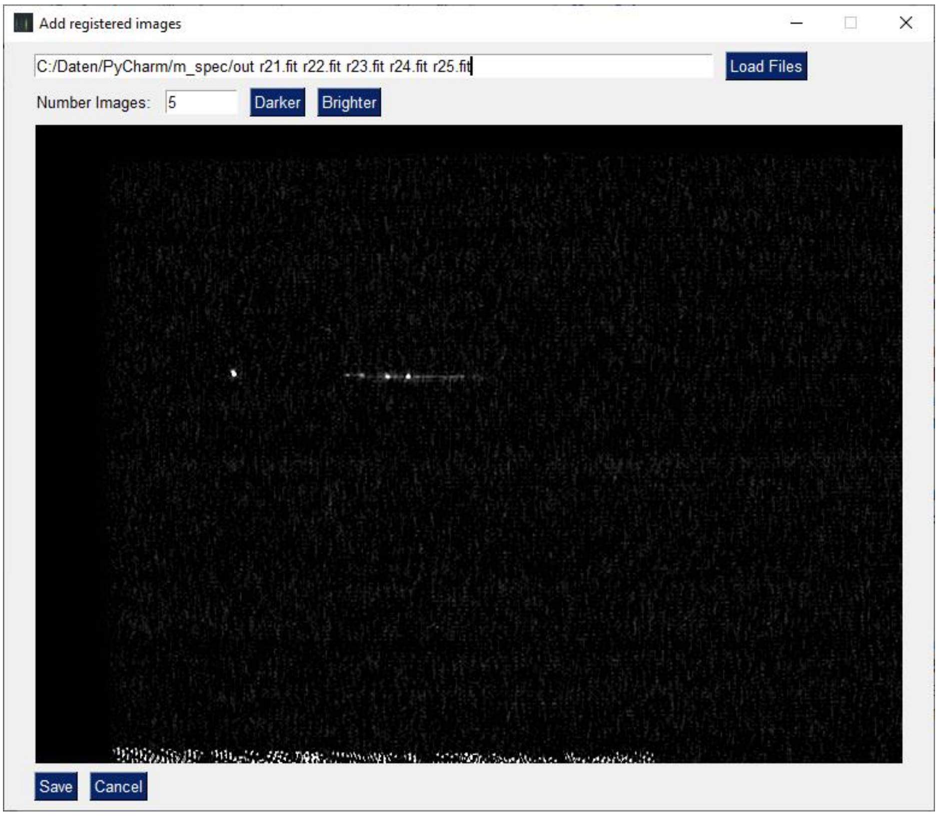
If you do not see the zero order either the image is too dark or you selected the wrong image. With Cancel you can try different parameters. With Ok, the registering starts. In case of success you see the sum spectrum of the registered images:



With “Previous” and “Next” you can look through the registered images, “Show Sum” shows the sum image again and by unchecking “show registered” you can browse through the original images. In the Results section at left you can check if the registering was successful.
The result window shows the start position and size of the rectangle, the original filename.
Also the filename, calculated shift (x, y) from the initial image and the relative shift wrt. the previous image (dx, dy) for each image. These values should vary only slightly, in the present example (-2.5±1, 4.5±1) pixel. If you see a large variation here, the correlation may have lost the target, try again with a different start image, rectangle or number of images.

For the Gaussian fit a list of the selected files, line intensities, positions and line widths of the Gaussian fit is given. The size of the rectangle should be at least as large as the size of the line and the movement from frame to frame (whichever is larger), but not too large. For the cross correlation method the rectangle size is not critical. Notice that in this particular case not the zero order was chosen but the Na-line, often one of the strongest lines in the spectrum. The zero order was outside of the image area to the left.

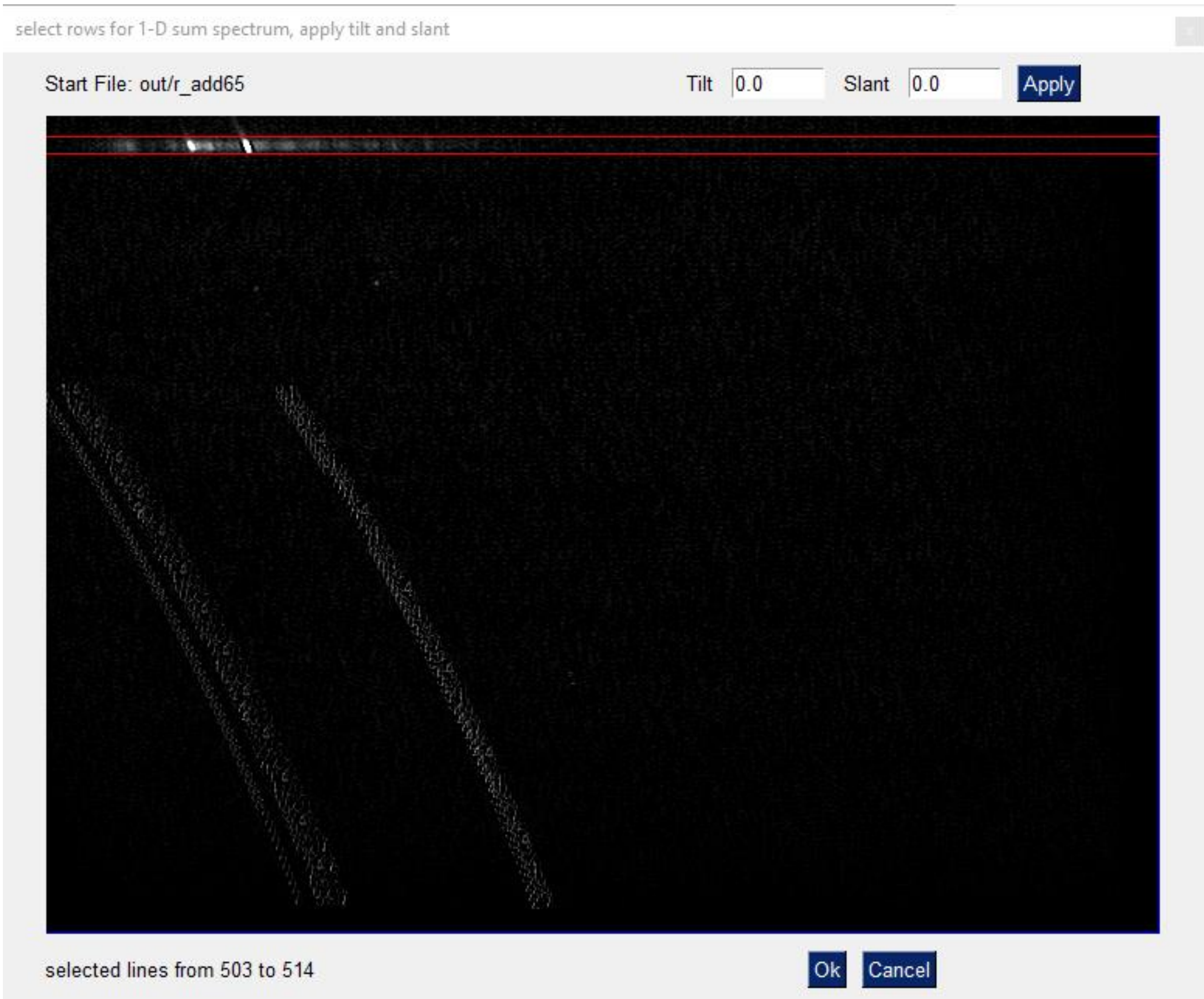
If you have registered a series of images, but only want to add some of them (e.g. to use only unsaturated images or to create a time series r1-r5, r6-r10, r11-r15, …) you can select in the menu – Tools – Add images
A new window opens, where you can choose the desired files to add with ‘Load Files’:



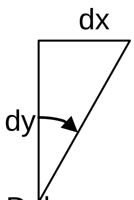
You can save the averaged sum image under a suitable name (e.g. r21-25.fit)

Spectrum extraction

Now you are almost there. Adjust the contrast of the coadded spectra until you see the spectrum strongly but only the strongest lines saturated. Continue by pressing “Add Rows” or load a previously registered image by pressing “Load Radd”. A new window with the sum spectrum appears. Next you can select the rows you would like to add, by dragging the mouse over the width of the spectrum:

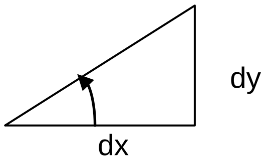


For the width of the selection it is important to add all the rows which contain the spectrum, but not the rows which contain only noise or other signals (The stripes in the lower half were produced by the text in the images, avoid these). If you do not like your choice, try again. In this case mdist3 to mdist67 were registered as r1 to r65 (notice the IRIS compatible naming) and added to r_add65.fit. In this particular spectrum you may notice that the spectral lines are not vertical because of the movemont of the meteor. You can improve this by processing fields with half exposure time (bobdoubler) or by correcting the slant (making the lines vertical). More about this later.
A look at the FITS-header (e.g. with IRIS: image info or Audela:) shows that the information about the time and observation as well as the distortion parameters are stored for documentation. Starting from version 0.9.18 this is also possible from within m_spec (see additional features)
With Ok you could select the rows and the resulting 1-d spectrum is stored as r_add65.dat, with the summed intensities as a function of column number (starting at column 0 to 719) for further processing with any spectra software (ISIS, Vspec, EXCEL …). Before doing that you should read the next paragraph.
You can adjust tilt and slant by typing a suitable value into the text boxes for Tilt and Slant. For a clockwise change of slant you enter a positive value. The value is the tangens of the angle or the slope dx/dy applied to the image

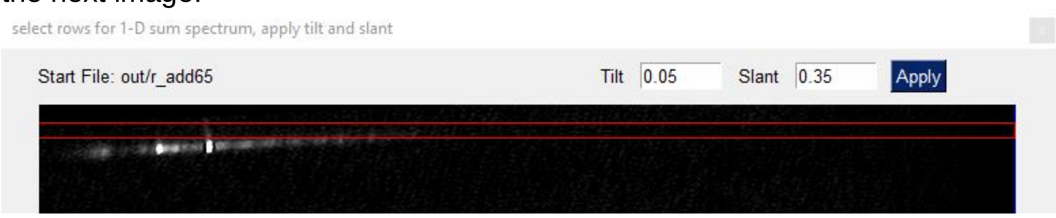


You confirm the value with Apply. The modified image is displayed. If you do not like the result you may type r and start all over or you type a new value for the slant and ↵ until you are satisfied with the result:

For the tilt the function is very similar. It is defined as the the tangens of the angle of the direction of the dispersion with respect to the x-axis or the slope dy/dx:



A positive value of the tilt rotates the spectrum anti-clockwise, as shown for demonstration in the next image:



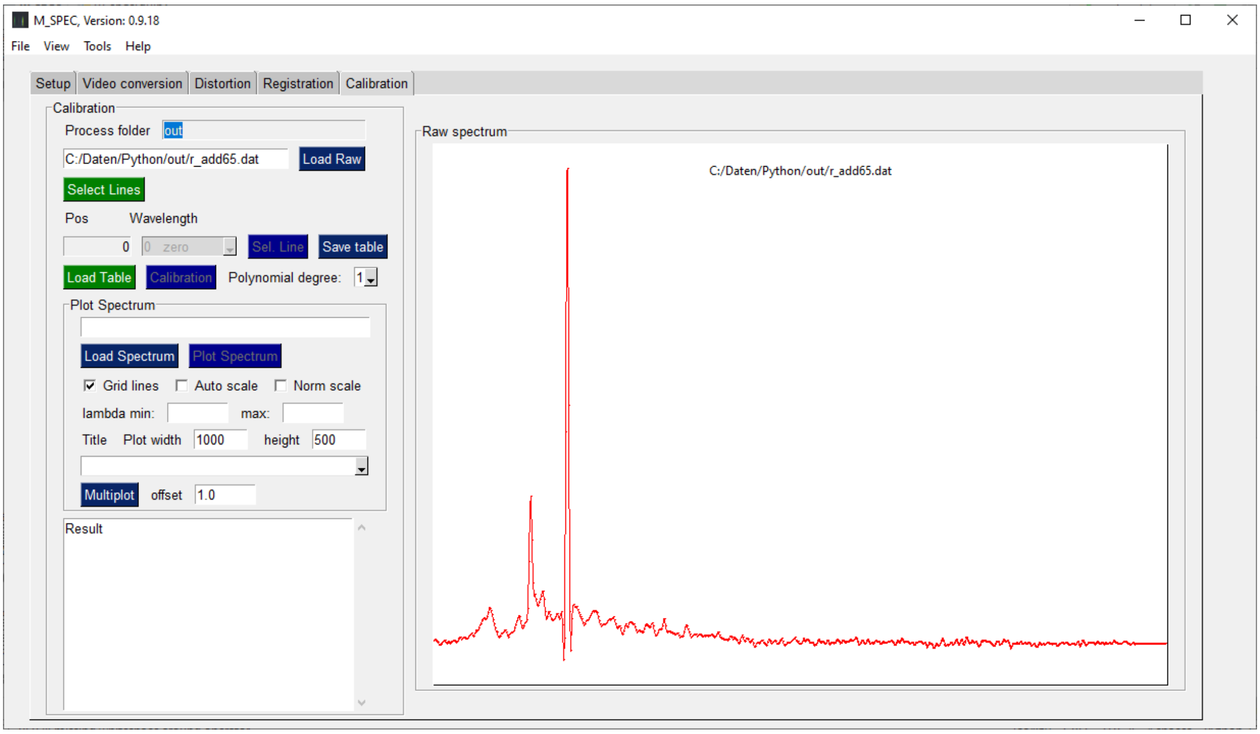
Of course, in this case a tilt of zero is the correct value, since the spectrum was aligned initially.. Notice that you can restore the original image by entering 0 for tilt and slant. When everything is adjusted, you type Ok to save the modified image and the 1-dimensional spectrum obtained by adding the rows column by column. The corrected image is stored as r_add65**st**.fit for processing with other software and the 1-dimensional spectrum obtained by adding the selected rows is stored as r_add65.dat (uncalibrated spectrum). You may also save it under a different name with “Save Raw Spectrum”

The slant is needed for correction of the meteor moving diagonally across the image field. Making the spectral lines vertical improves the spectral resolution. The tilt is used to correct for misalignment of the disperssion direction. This can also be corrected by adjusting the angle in the setup parameters. If the grating rotates slightly in front of the camera, a correction of tilt during the analysis is easier here, where the result is seen immediately. If a consistent tilt angle is observed in the spectra, it is advisable to correct the angle “rot” in m_set.ini. For small angles the tangens of the tilt is equal to the required correction of “rot”, measured in radians.

If you are satisfied with the result, you finish this process step with “Ok”. This saves the corrected image as r_add65st.fit. The image is normalized to a maximum of 32767 (2^15-1) in order to be viewed easily in IRIS. In the fits-header of this image the selected values of slant and tilt as well as the selected rows and the selected width (half width) are displayed:

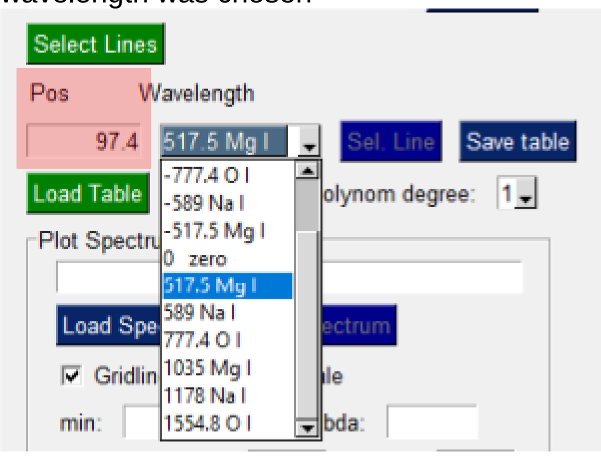
The original image r_add65.fit is simply the average of the added images, with 255 ADU’s scaled to 32767, for readable display in any FITS-viewer.

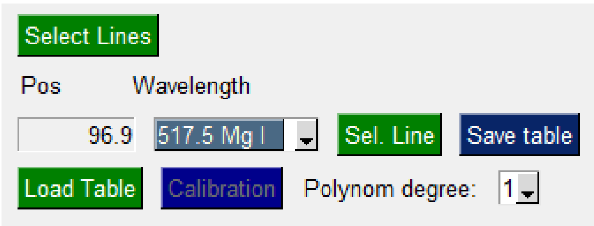
By addition of the selected rows for each column the 1-dimensional spectrum is calculated and saved as r_add65.dat in a format often used in spectroscopy. It contains two columns for pixel number and intensity, separated by a blank. In calibrated spectra the two columns denote wavelength and intensity. The raw (uncalibrated) spectrum is displayed when you press “Calibrate”. This brings you to the last tab for wavelength calibration of the raw (Intensity vs. pixel) spectrum:



Wavelength calibration

In the graph the raw spectrum is shown. Start the calibration with “Select Lines”. This activates the mouse clicks in the graph and starts a new table with wavelengths and positions of calibration lines. Select a line in the raw spectrum by dragging the mouse with pressed left button across it. 5 points around the maximum intensity are used for a quadratic fit. The position of its maximum determines the line position in pixels. In the text window Pos is the center of the selected range given. Make sure the highest point of the line is within the range indicated by green lines. Select the correct wavelength from the table “Wavelength” or enter a wavelength not listed in the table (you have to enter the table first with a mouse click, then you can edit the line in the text field). Confirm the choice with the button “Sel Line” (it turns green when a wavelength was chosen

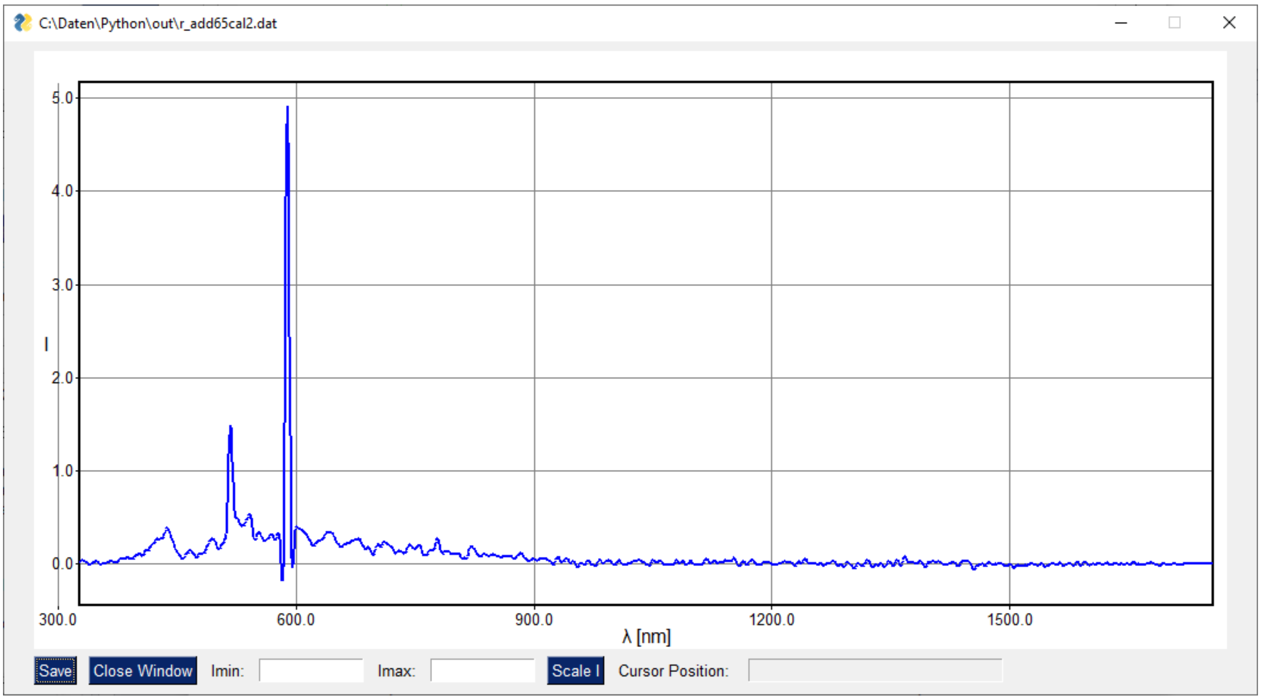




In the memo below the position, width (calculated from the parabolic fit) and the selected calibration wavelength together with information about the line are displayed. Continue by selecting more lines (those already used are shown in blue). When finished, press “Save Table”. The list of positions and wavelengths is stored under the raw file name.TXT

```
#     x     lambda
  95.78   517.50
```

131.67 589.00
You may edit this table with a text editor or the built-in text editor, in case you have entered wrong lines or wavelengths. Load the raw spectrum again if you would like to make a new calibration. The table is overwritten if you do a new line measurement with the same raw spectrum.
With two lines only a linear fit is possible. Select 1 for polynom degree (default). The polynom is of the form $a_n*x^n + a_{n-1}*x^{n-1} + \ldots + a_1*x + a_0$.
For the linear polynom: $a_1*x + a_0$, with dispersion a_1 and offset a_0.
With “Calibration” (green if you have saved or loaded a table) a least square fit of the polynom to the selected line positions is done.
The error of the fit is zero, because with 2 lines a linear fit is determined, there are no additiional lines for evaluating the error. The calibrated spectrum is saved as r_add65cal.dat, unless you give a different name to the raw spectrum. With “Plot Spectrum” the calibrated spectrum is shown:

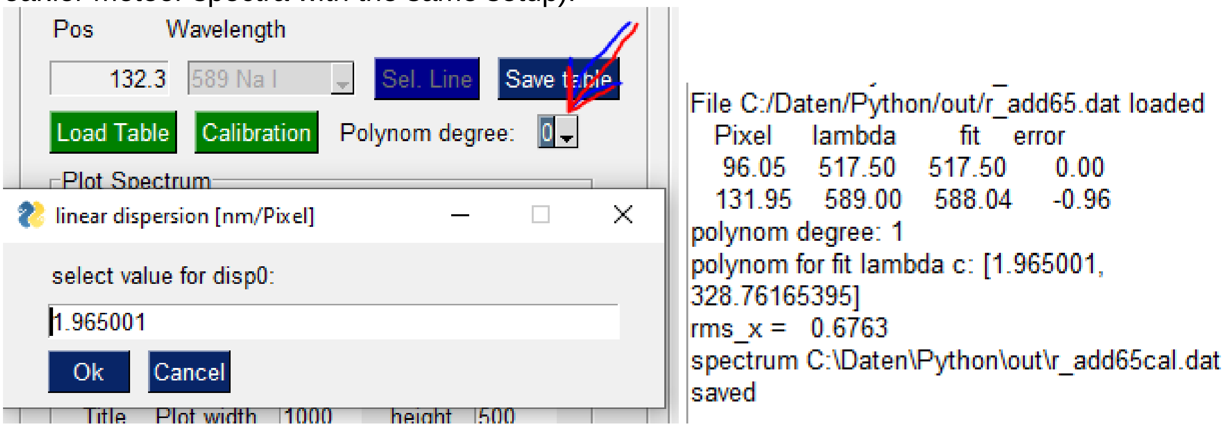


You also have the possibility to select the plot range. The intensity scale you set in the Plot window, with “Scale I” you plot the new intensities. By closing the window and replotting it you also set the correct scales. The wavelength range and the window title are set in the main window. The title can be selected from a list of the latest converted avi-files and the latest raw spectrum or typed into the list manually. The spectrum can be saved with “Save”. Choose a filename, a png image is saved. If no filename is entered, r_add65cal_plot.png or the corresponding raw file name + “_plot.png” is saved. Further possibilities to plot spectra are given below, in additional features.
Important:
In spectroscopy it is good practice that the wavelength increases to the right, to higher pixel values. Therefore orient the grating in front of your camera that the first order is on the right side with respect to the zero order. The orders to the left of the zero orders have negative values, therefore the corresponding wavelengths are negative. Always enter the line wavelengths as wavelength*order (0 for zero order, wavelength for first order, 2*wavelength for 2nd order, etc.).

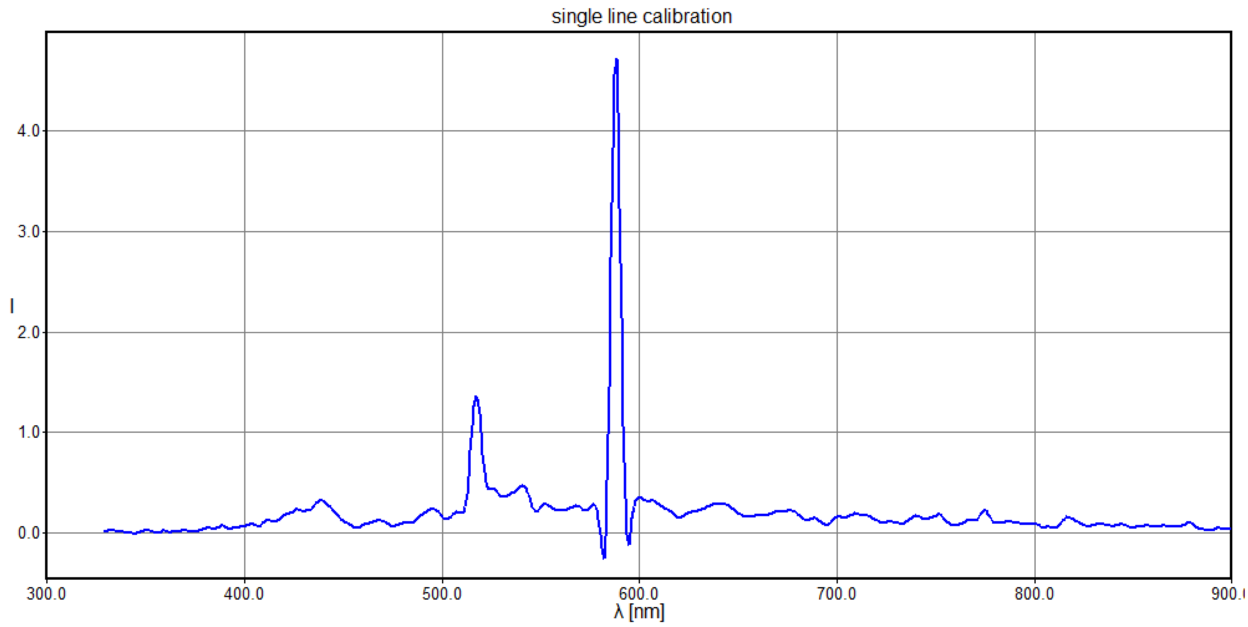
Single line calibration

The dispersion calculated from these two lines is similar to the dispersion from the laser calibration: disp0 = 2.005. However, it was derived from two lines in a narrow region, therefore the value is not very precise. In that case it may be better to use the dispersion disp0 for the fit.
This is done by entering 0 for the polynom degree:

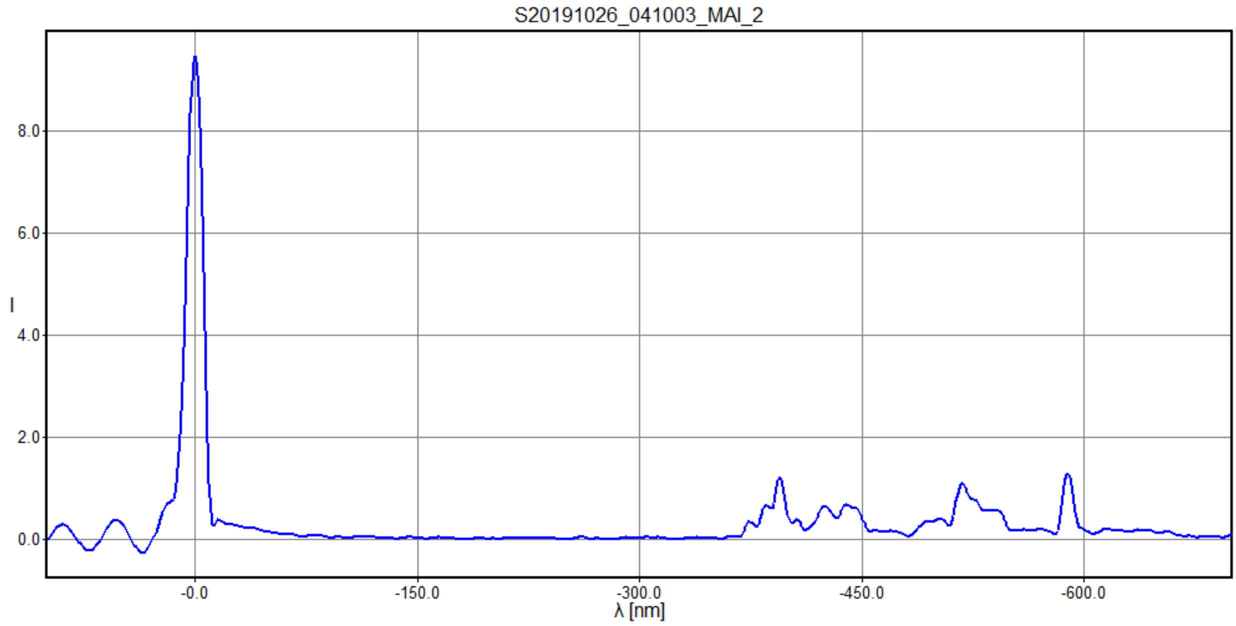
The value 2.005 from the setup file is suggested for disp0. In case you know the dispersion from other meteor spectra, you may enter here the correct value (1.965 was selected, based on earlier meteor spectra with the same setup):



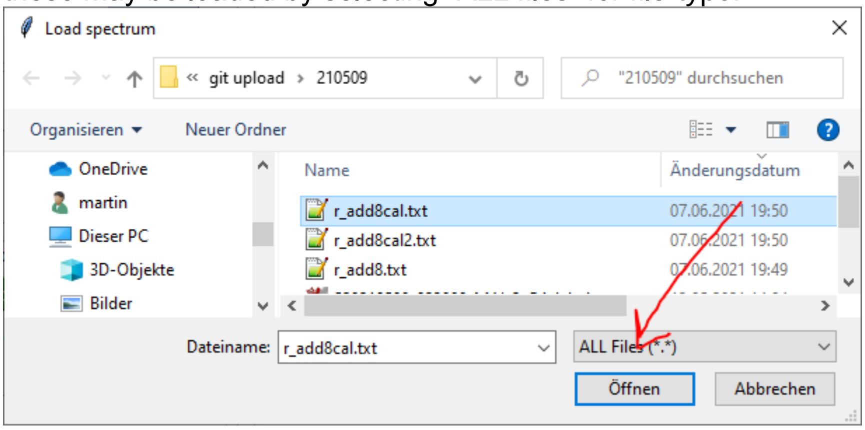
We use the same linelist, but only the first entry is used, therefore the following lines will have an error. I use the Mg-line for calibration



For plotting a negative order you may reverse the plot by plotting from e.g. +100 nm to -900 nm, plotting the zero order and negative first order:



Note:
Raw and calibrated spectra may be loaded with the buttons “Load Raw” and “Load Spectrum”. This accepts the default extension ‘.dat’ for spectroscopy files. If you would like to import files with the same file structure (two columns separated by blanks) but with a different extension, these may be loaded by selecting “ALL files” for file-type:



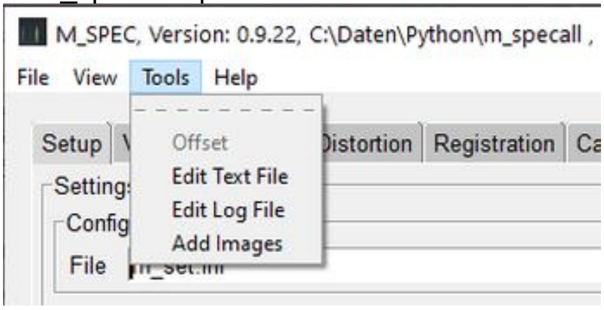
After loading, these files are saved as *.dat for further use. This is convenient for files uploaded from github, which does not allow the extension ‘.dat’. Spectra file are stored there as plain ‘.txt’ files. Remember to select ‘ALL files’ when loading github spectra.

Additional features of the script

Tools

Edit text files

In m_spec it is possible to edit text files:



With Save the files are saved under the same name

Logfile

The main inputs and results of the script execution are stored in a logfile. You may review what you have done by reading the file m_spec(yymmdd).log, where (yymmdd) stands for the actual date in year, month day format. There you will find the main input and output for the complete processing from converting the AVI-file to the wavelength calibrated spectrum:

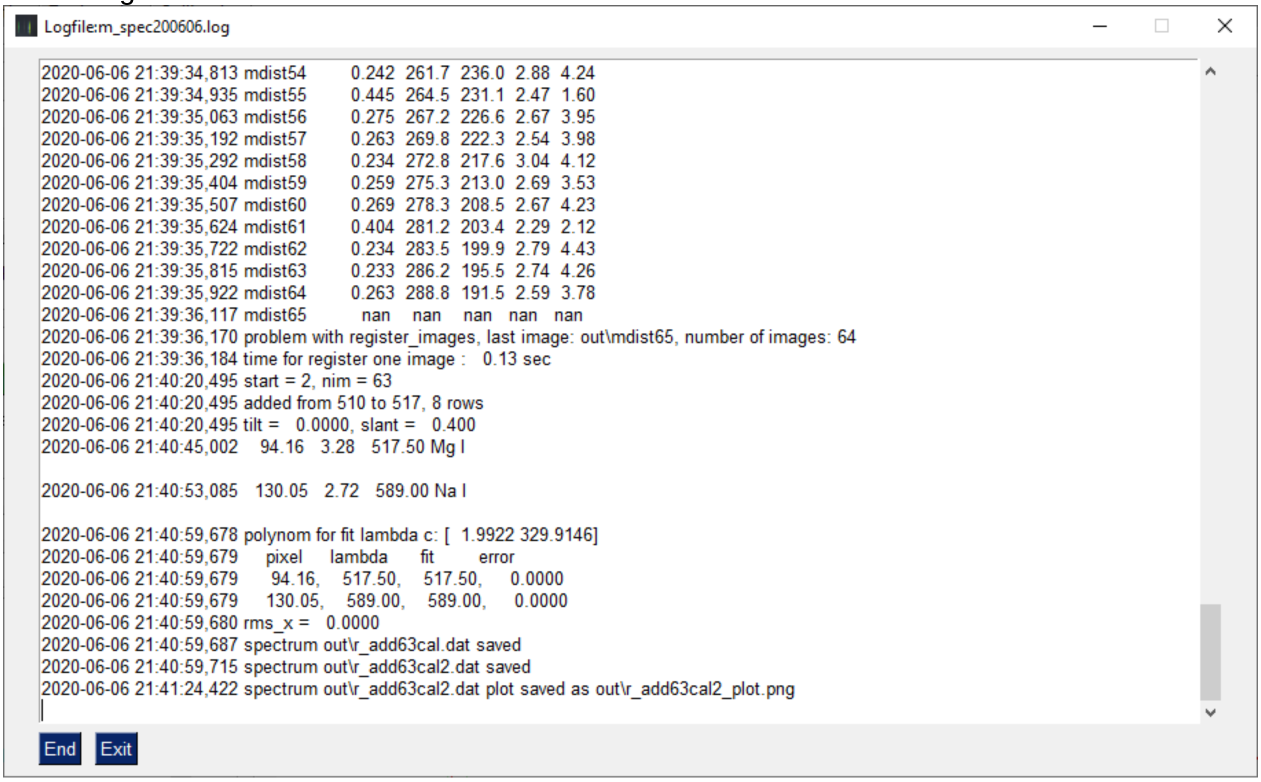
2020-04-03 15:59:51,525 M_SPEC START
+++
2020-04-03 15:59:51,525 M_SPEC version 0.9.16, 0.9.16 ++++++++++++++++++++++++++++
2020-04-03 16:00:10,565 Platorm: Windows
2020-04-03 16:00:13,779 converted C:/Daten/Python/181014/M20181014_023817_MAI_2.avi
91 images
2020-04-03 16:00:13,779 Station = MAI_2 Time = 2018-10-14T02:38:17.000
2020-04-03 16:00:47,500 scalxy =   0.920
2020-04-03 16:00:47,500 x00 =   371.200
2020-04-03 16:00:47,500 y00 =   324.183
2020-04-03 16:00:47,500 rot =   -0.006
2020-04-03 16:00:47,500 disp0 =   1.965
2020-04-03 16:00:47,500 a3 = 3.137e-07
2020-04-03 16:00:47,500 a5 = -3.193e-13
2020-04-03 16:00:47,500 'DATE-OBS' = 2018-10-14T02:38:17.000
2020-04-03 16:00:47,500 'M-STATIO' = MAI_2
2020-04-03 16:00:57,735 67 images processed of 67
2020-04-03 16:00:57,735 process time for single distortion: 0.15 sec
2020-04-03 16:01:50,690 start x y, dx dy, file: 132 508,14 14, out/mdist
2020-04-03 16:01:50,700   out/mdist3 31.33 131.91 507.01   3.18   6.10
2020-04-03 16:01:50,776   out/mdist4 47.43 134.01 500.65   2.66   6.52
2020-04-03 16:01:50,998   out/mdist5 41.72 136.11 495.62   2.71   6.01
…
2020-04-03 16:01:55,362   out/mdist65 65.64 291.37 187.57   2.58   3.38
2020-04-03 16:01:55,414   out/mdist66 59.70 294.40 183.00   2.72   3.70
2020-04-03 16:01:55,488   out/mdist67 66.45 297.08 178.37   2.56   3.25
2020-04-03 16:01:55,532 time for register one image :   0.07 sec
2020-04-03 16:02:43,981 start = 3, nim = 65
2020-04-03 16:02:43,981 added from 503 to 514, 12 rows
2020-04-03 16:02:43,981 tilt =   0.0000, slant =   0.400

2020-04-03 16:04:31,615 Pixel: 95.75 FWHMp = 3.540 lambda = 517.500 Mg I

2020-04-03 16:04:41,126 Pixel: 131.44 FWHMp = 2.910 lambda = 589.000 Na I

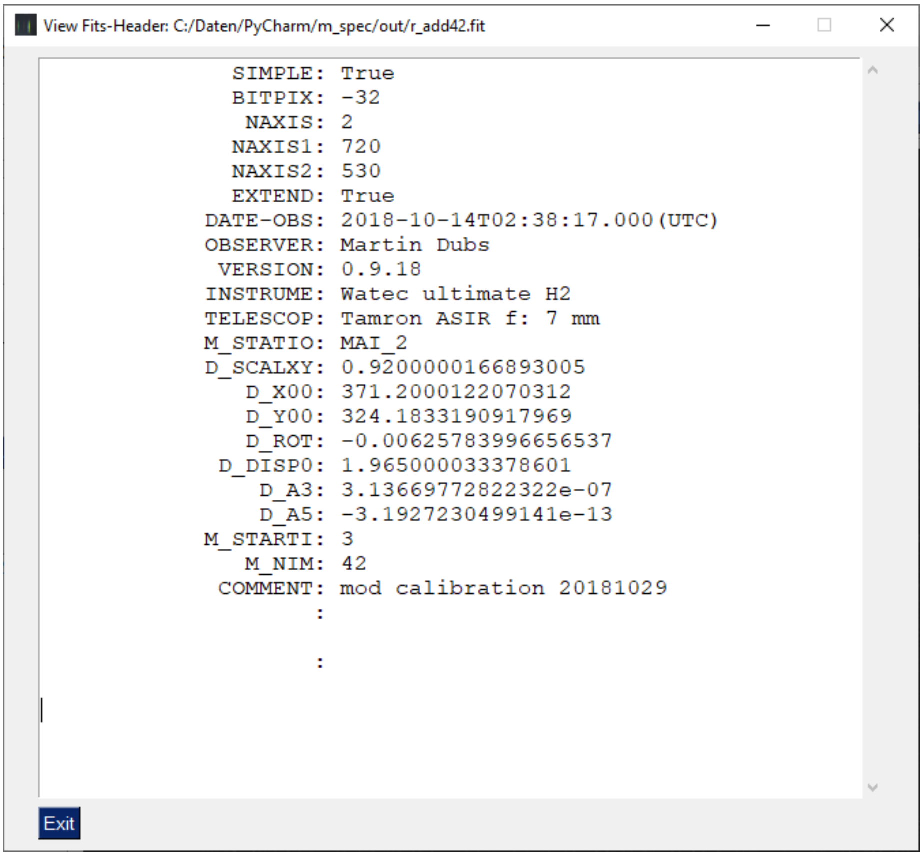
2020-04-03 16:04:45,728 polynom for fit lambda c: [2.003 325.678]
2020-04-03 16:04:45,729   pixel   lambda   fit   error
2020-04-03 16:04:45,729   95.75,   517.50,   517.50,   0.0000
2020-04-03 16:04:45,730   131.44,   589.00,   589.00,   0.0000
2020-04-03 16:04:45,730 rms_x =   0.0000
2020-04-03 16:04:45,738 spectrum C:\Daten\Python\out\r_add65cal.dat saved
2020-04-03 16:04:45,758 spectrum C:\Daten\Python\out\r_add65cal2.dat saved
2020-04-03 16:06:29,804 spectrum C:\Daten\Python\out\r_add65cal2.dat plot saved as 20181014_023817_MAI_2.png

You may edit the logfile with a text editor. For a quick view of the actual logfile you can use the built in logfile viewer with menu – View:



Built in Fits viewer

In the Menu – View – Fits header you can select any Fits File and look at its header information. This is quite useful, if you are not sure about the observation date or the used calibration:



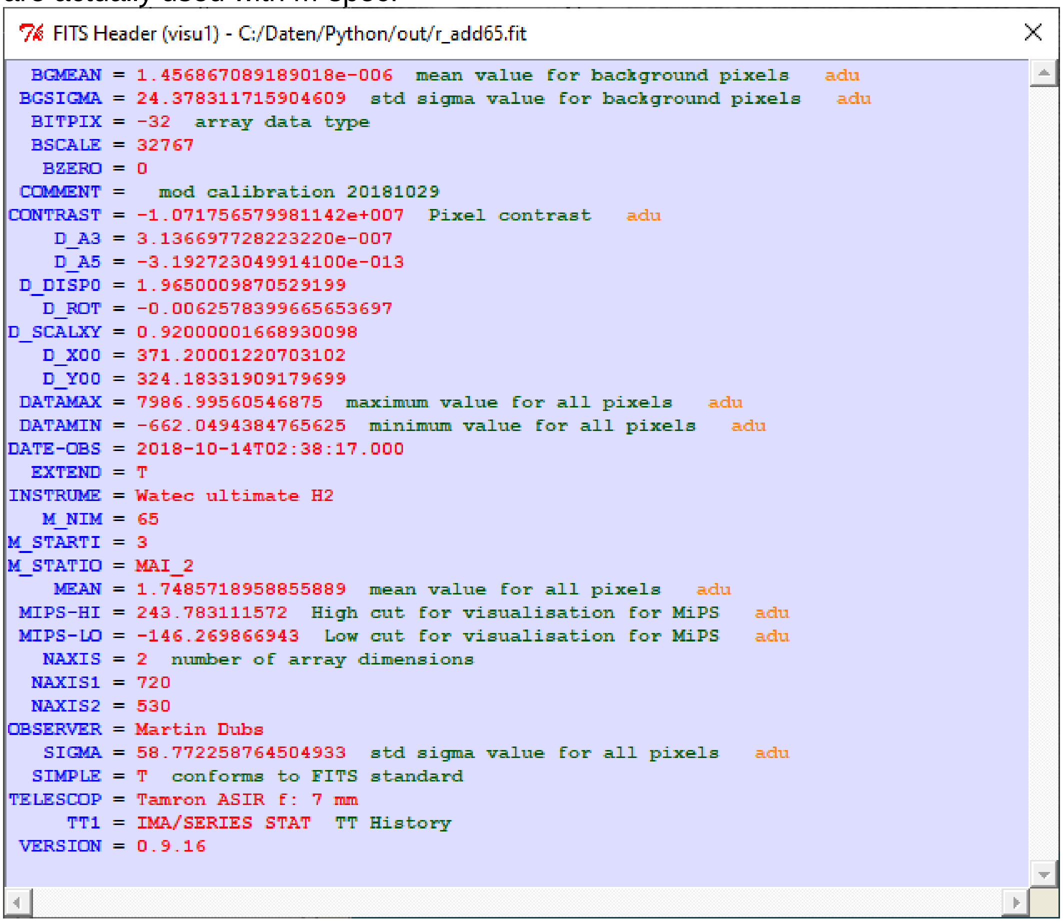
View Fits-Header: C:/Daten/PyCharm/m_spec/out/r_add42.fit

```
       SIMPLE: True
       BITPIX: -32
        NAXIS: 2
       NAXIS1: 720
       NAXIS2: 530
       EXTEND: True
     DATE-OBS: 2018-10-14T02:38:17.000(UTC)
     OBSERVER: Martin Dubs
      VERSION: 0.9.18
     INSTRUME: Watec ultimate H2
     TELESCOP: Tamron ASIR f: 7 mm
     M_STATIO: MAI_2
     D_SCALXY: 0.9200000166893005
        D_X00: 371.2000122070312
        D_Y00: 324.1833190917969
        D_ROT: -0.006257839966565837
      D_DISP0: 1.965000033378601
         D_A3: 3.136697728223222e-07
         D_A5: -3.1927230499141e-13
     M_STARTI: 3
        M_NIM: 42
      COMMENT: mod calibration 20181029
             :

             :
```

Exit

If you compare with another Fits-viewer, you will notice that you see only the keywords which are actually used with m-spec:



FITS Header (visu1) - C:/Daten/Python/out/r_add65.fit

```
   BGMEAN = 1.456867089189018e-006  mean value for background pixels   adu
  BGSIGMA = 24.378311715904609  std sigma value for background pixels   adu
   BITPIX = -32  array data type
   BSCALE = 32767
    BZERO = 0
  COMMENT =   mod calibration 20181029
 CONTRAST = -1.071756579981142e+007  Pixel contrast   adu
     D_A3 = 3.136697728223220e-007
     D_A5 = -3.192723049914100e-013
  D_DISP0 = 1.965000987052919
    D_ROT = -0.006257839966565369
 D_SCALXY = 0.92000001668930098
    D_X00 = 371.20001220703102
    D_Y00 = 324.18331909179699
  DATAMAX = 7986.99560546875  maximum value for all pixels   adu
  DATAMIN = -662.0494384765625  minimum value for all pixels   adu
 DATE-OBS = 2018-10-14T02:38:17.000
   EXTEND = T
 INSTRUME = Watec ultimate H2
    M_NIM = 65
 M_STARTI = 3
 M_STATIO = MAI_2
     MEAN = 1.7485718958855889  mean value for all pixels   adu
  MIPS-HI = 243.783111572  High cut for visualisation for MiPS   adu
  MIPS-LO = -146.269866943  Low cut for visualisation for MiPS   adu
    NAXIS = 2  number of array dimensions
   NAXIS1 = 720
   NAXIS2 = 530
 OBSERVER = Martin Dubs
    SIGMA = 58.772258764504933  std sigma value for all pixels   adu
   SIMPLE = T  conforms to FITS standard
 TELESCOP = Tamron ASIR f: 7 mm
      TT1 = IMA/SERIES STAT  TT History
  VERSION = 0.9.16
```

Image Tools, prepare flat image

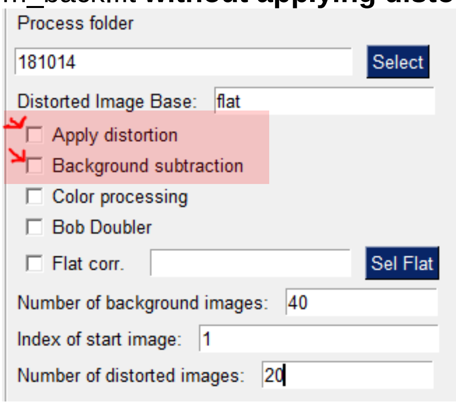
In astrophotography and spectroscopy it is common to produce a “flat image” to correct for unequal detector sensitivity over the image area. In this section it will be described how to make a flat image and how to apply it to the meteor spectra.
Due to the movement of the meteor, the meteor spectrum may appear anywhere in the image area. Wide angle lenses, which are commonly used for meteor spectroscopy, usually show a decrease of light intensity towards the corners of the detector, caused on one hand by spreading the light from a fixed solid angle over a larger area for the tangential projection and vignetting of the lens at larger angles to the optical axis.
In addition, the transformation to an orthographic projection changes the size of image elements depending on the angle ρ, by a factor $\cos(\rho)$, specific to the orthographic projection. All this has to be taken into account when analysing spectra quantitatively.

Flat image

A flat image represents the relative sensitivity of each pixel for a given detector and imaging lens. It is affected by pixel to pixel sensitivity variations, variations of image scale by image distortions and vignetting of the lens (reduction of off-axis lens throughput). It is typically made by photographing an evenly illuminated white screen, preferably out of focus in order to wash out screen irregularities in reflection or put a white matte surface in front of the lens and illuminate it evenly from the other side. Another possibility is to photograph the sky on a foggy day or the sky without clouds in the zenith.
The pixel to pixel sensitivity variations are usually small in modern sensors. In addition, the movement of the meteor averages them out. Therefore they are ignored in the present treatment. On the other hand, the illumination may have a gradient over the image area. A linear gradient may be eliminated by mirroring the image about a vertical and/or horizontal axis or rotating the image by 180° (inversion) and averaging the resulting images. This however requires that the optical axis is close to the image centre and the grating perpendicular to the optical axis for the following transformation to an orthographic projection to work.
Depending on the sensor type, it may be also necessary to obtain an offset image. Some sensors output a non-zero value without illumination in order not to clip noise values for low signal. An offset image is obtained with the same camera settings as the flat, but without illumination and the shortest possible exposure time. It is subtracted from the flat before further processing. Another possibility is to measure the offset (average intensity of the offset image) and subtract this constant value from the flat image.
The following example is for a Watec 902 H2 ultimate with a Tamron 12VG412ASIR f1.2 Zoom lens at f = 7mm. Unfortunately, the Watec camera produces a dark edge which leads to division by zero when applying the flat to the meteor images. This problem will be dealt in the processing shown below. Also visible are some hot pixels.

Step by step procedure to make a flat

- Record short video (1 sec) of white screen with the same camera settings (F, Gamma) as for meteor spectra, but adjust exposure time so that image is **not saturated**, **turn off title** in UFO capture
- Record short video with lens cap with the same camera settings except shortest exposure time for offset image.
- Extract videos of flat and offset in the same format as for meteor spectra (bob-doubler, binning), 20 – 50 *.png images are sufficient
- Use the distortion tab to convert the *png images to make a background image m_back.fit **without applying distortion:**



or rename the image m_back.fit to flat_b.fit or offset_b.fit before converting the next video.

Another possibility to make a flat is to use the same settings (**no background subtraction, no distortion**) giving the name flat and selecting a suitable start image and number of images and use the resulting flat_sum.fit for the raw flat image. Both methods give the same image, apart from a scale factor and some rounding errors.

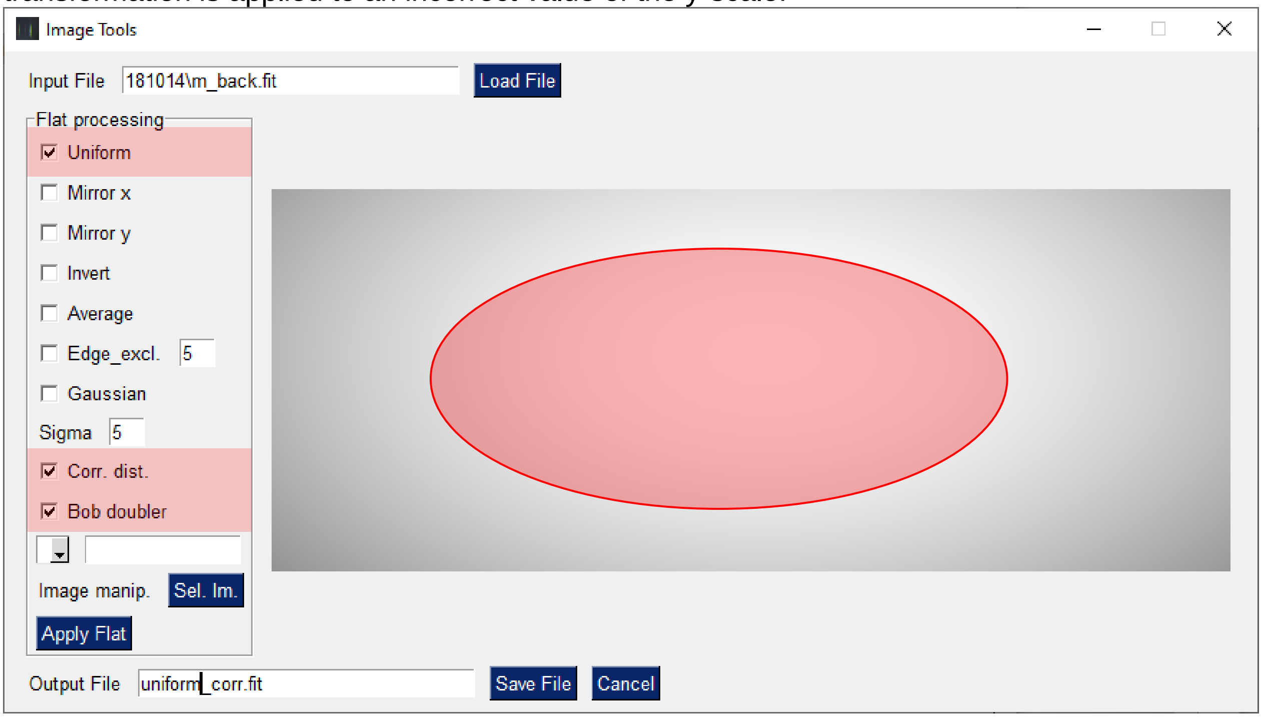
m_back.fit ≈ flat_sum.fit

During further processing the normalization will be taken care of.

- Use Tools – Image Tools to process the flat
  Load either m_back or flat_sum with Load File. The size of this image must be equal to the size of the unprocessed video frames, to which it is applied later:

The flat processing has the following selections:

- Uniform: this generates a flat with constant intensity, e.g. if you only want to correct the distortion (transformation from spherical to orthographic coordinates, multiplication by cos(ρ).Mirror x, mirror y: mirror the image abaout a vertical or horizontal axis to remove a horizontal or vertical linear gradient if averaged
- Invert: rotate the image by 180° (mirror x and y axis simultaneously), with average for removing linear gradient
- Average: used with mirror x, mirror y or invert to remove gradient, make flat symmetrical (works correctly only if optical axis is in the center of the image or very close), used for elimination of gradients in the illumination flat screen.
- Edge exclusion: The dark edge from the Watec sensor can be eliminated by a border with constant width and intensity equal to the nearest pixel inside the border. These border pixels are not used for the flat correction of the meteor spectra, but they avoid division by zero for the flat corrected images.
- Gaussian: smooth (filter) the image with a Gaussian of Sigma. Reduces noise and artefacts of the flat image (hot or dead pixels)
- Correct distortion: Apply cos(ρ) correction to the flat
- Bob doubler: see above. If the original video was interlaced and half frames extracted, they are scaled by a factor 0.5 vertically. Accordingly the distortion correction has to be scaled by the same factor. For the mirror and filter operation this has no effect. Note: in this case you also have to check the correct value of Bob Doubler; otherwise the transformation is applied to an incorrect value of the y-scale:

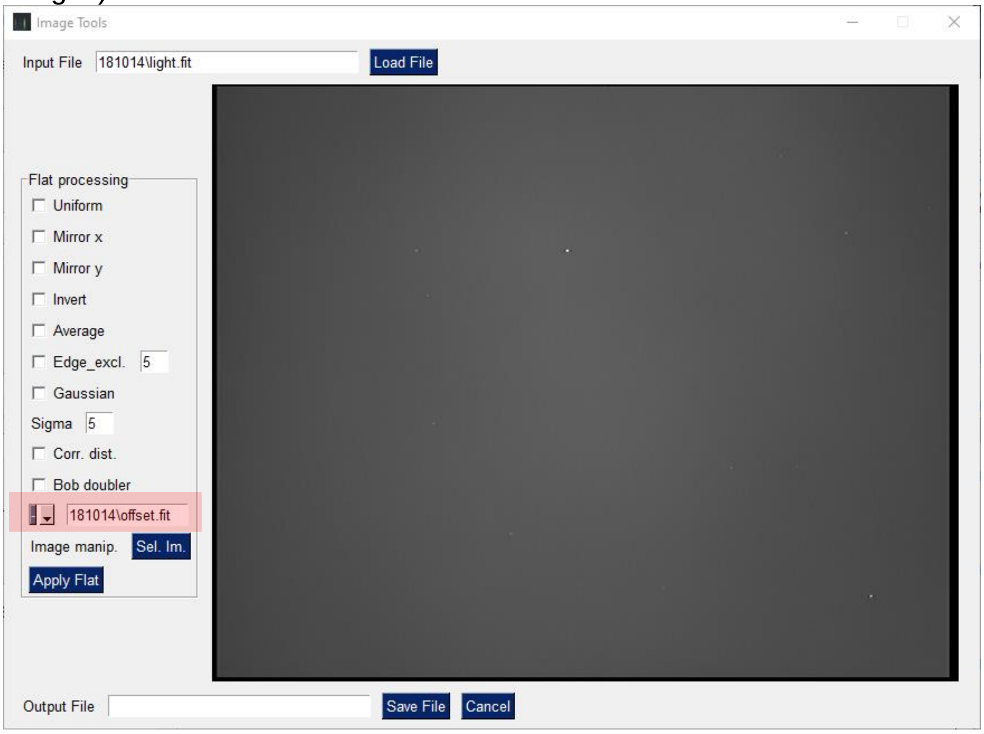


Original image with Bob Doubler, correction applied with Bob Doubler

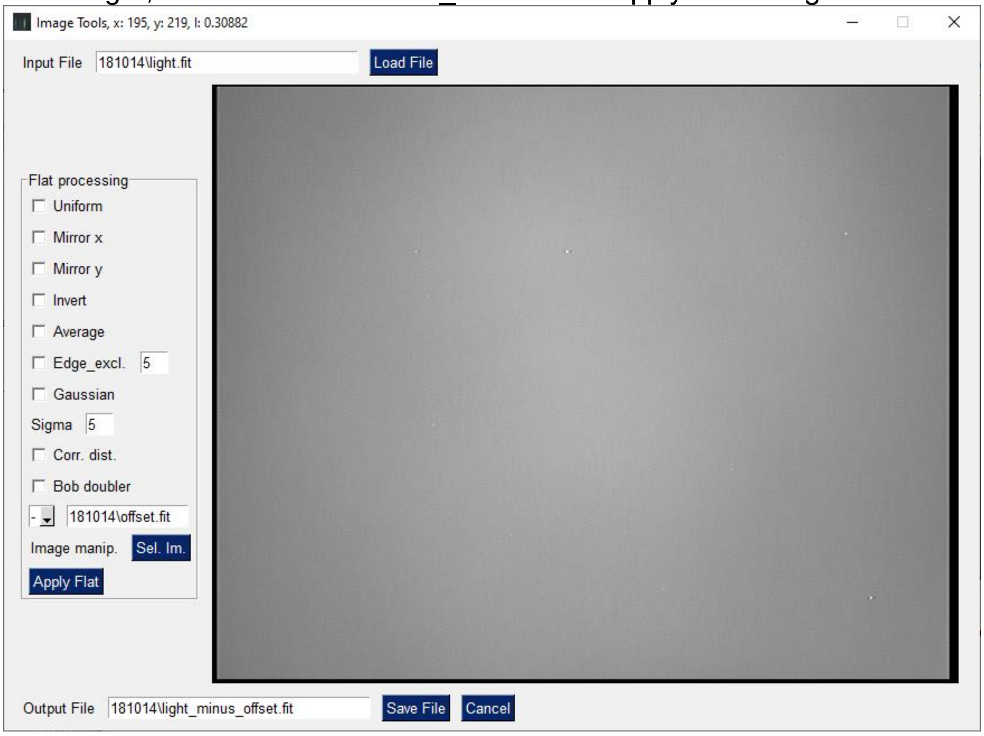
Several operations can be applied in the same step, e.g. inversion, average, Gaussian and correct distortion.
If the original flat was a png image, even no operation may be applied and a fit image can be saved.
With Apply Flat the selected operations are applied and the file stored as _flat.fit. It may be saved with the proposed filename, which shows the operations, e.g.

“181014\m_back_mx_my_ave_g5.fit”, or by selecting a different name, e.g. specifying the equipment, such as “Tamron_7mm”.

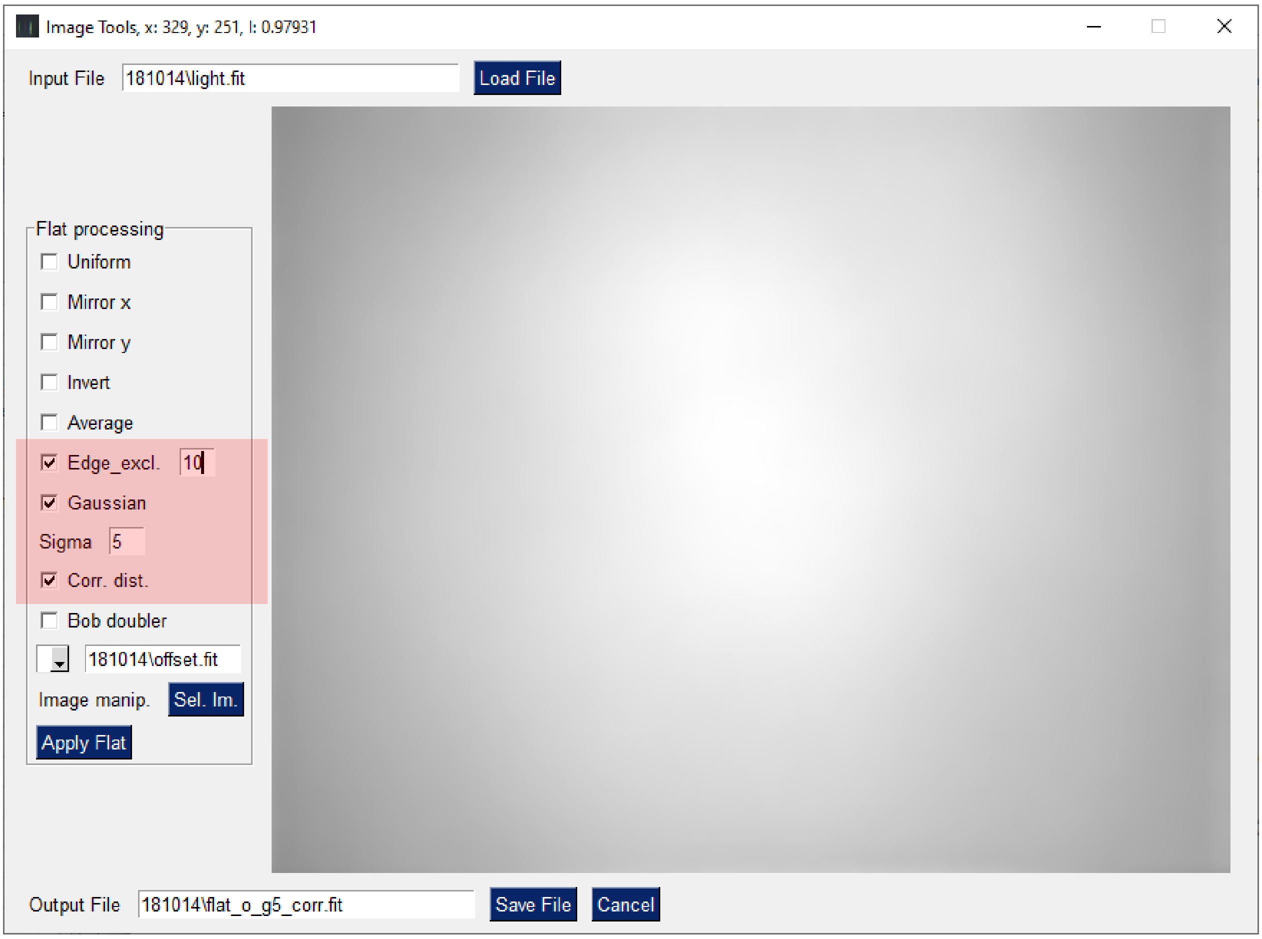
As an example, we load a flat image named light, renamed from m_back (average of 20 images)



Next we subtract an offset (dark) image, produced with lens cap and the same camera settings as the light, also renamed from m_back. With “Apply Flat” we get the result light – offset:



It is saved as flat_o.fit. Without reloading it it can be processed further by removing the border, width 10, Gaussian filtering and correct distortion. Make sure to uncheck the offset subtraction:

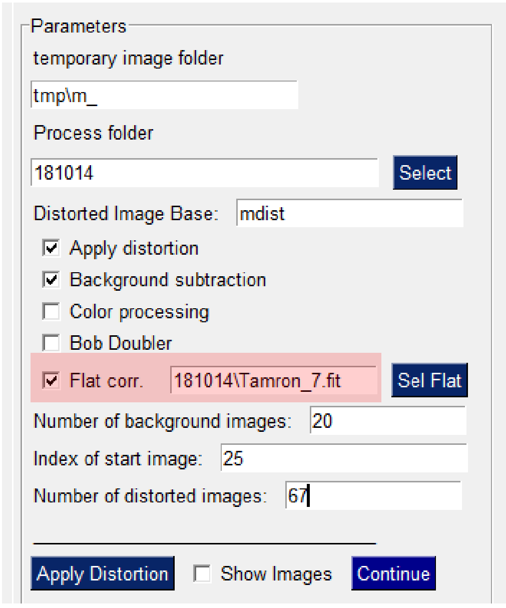


By correcting the distortion it is also automatically normalized to peak intensity 1.0. It can be saved under the proposed name, which shows the processing steps and for further use under the name Tamron_7.fit.
The log file shows the corresponding manipulations also:

20 1816 Image file: 181014\light.fit
20 1816 Image file: 181014\offset.fit
20 270 181014\light minus 181014\offset written to _flat.fit
20 1816 Save image: 181014\flat_o.fit
20 198 rp_max, rp_poly, 453.1, [8.487e-25 -8.680e-19 5.903e-13 -3.128e-07 1.000e+00]
20 270 181014\flat_o.fit gauss sigma=5 correct distortion written to _flat.fit
20 1816 Save image: 181014\flat_o_g5_corr.fit
20 1816 Save image: 181014\Tamron_7.fit

In the absence of fit files for the flat and offset you may also use PNG or BMP files for the flat and offset to create a flat in the required FIT format.

The created Tamron_7.fit can now be used for the flat correction of meteor files:



New plotting features

Since the manual was written (23.11.2020) some features have been added to the plotting of spectra

- Clip wavelength range
- Label calibration lines (replaces Label Peak)
- Label meteor lines
- Linewidth tool

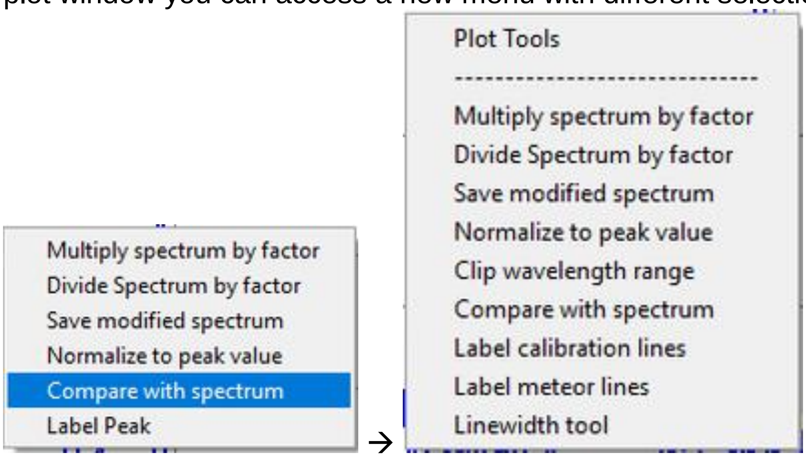
With Clip wavelength range the meteor spectrum can be limited to a useful wavelength range, e.g. to the first order. If the zero order is cut off, normalize to peak value produces a spectrum which can easily be plotted either with “Auto scale” or “Norm scale”.
“Label Peak” has been renamed “Label calibration lines”. It is used to check the calibration of the spectrum and indicate the meteor lines used for calibration or the absorption lines (Balmer series of hydrogen) for stellar spectra used for the determination of the response function.
For labelling other meteor lines, the list from the IMO handbook below has been converted to nm and written in the m_spec linelist format as “meteror_lines.txt”. The lines of this list can be indicated in the plot as well with “Label meteor lines”

Clip wavelength range

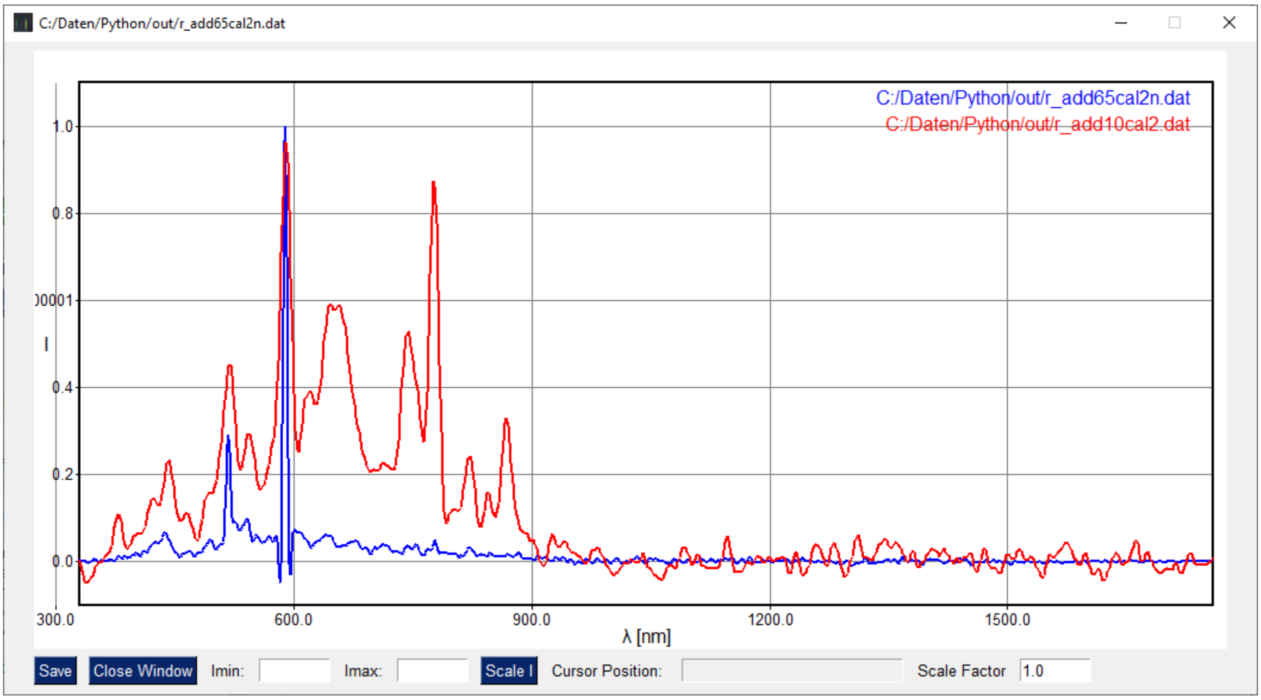
First load a spectrum, select the wavelength range and plot the spectrum. With a right click into the plot window select “Clip wavelength range”. In the plot nothing changes, but the data points outside the plotted wavelength range are removed and the new file automatically saved with appendix “C” as *oldname*C.dat. This is particularly useful if you normalize the file with right click – “Normalize to peak value”. This file is automatically saved with appendix “N” as oldnameCN.dat, for easy plotting with either Norm scale (with your default wavelength range) or Auto scale if you would like to view the spectrum later.

Compare spectra

In addition to plot a single spectrum, you also have the possibility to compare two spectra. First select wavelength range, plot one spectrum and select intensity range. With a right click into the plot window you can access a new menu with different selections:



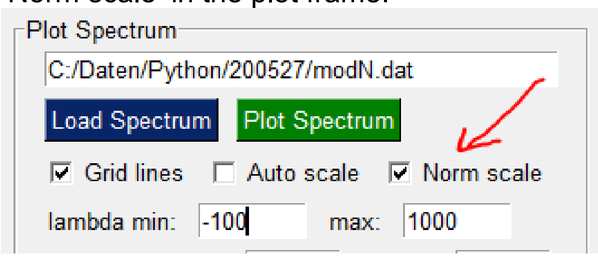
select Compare with spectrum and load a second spectrum:

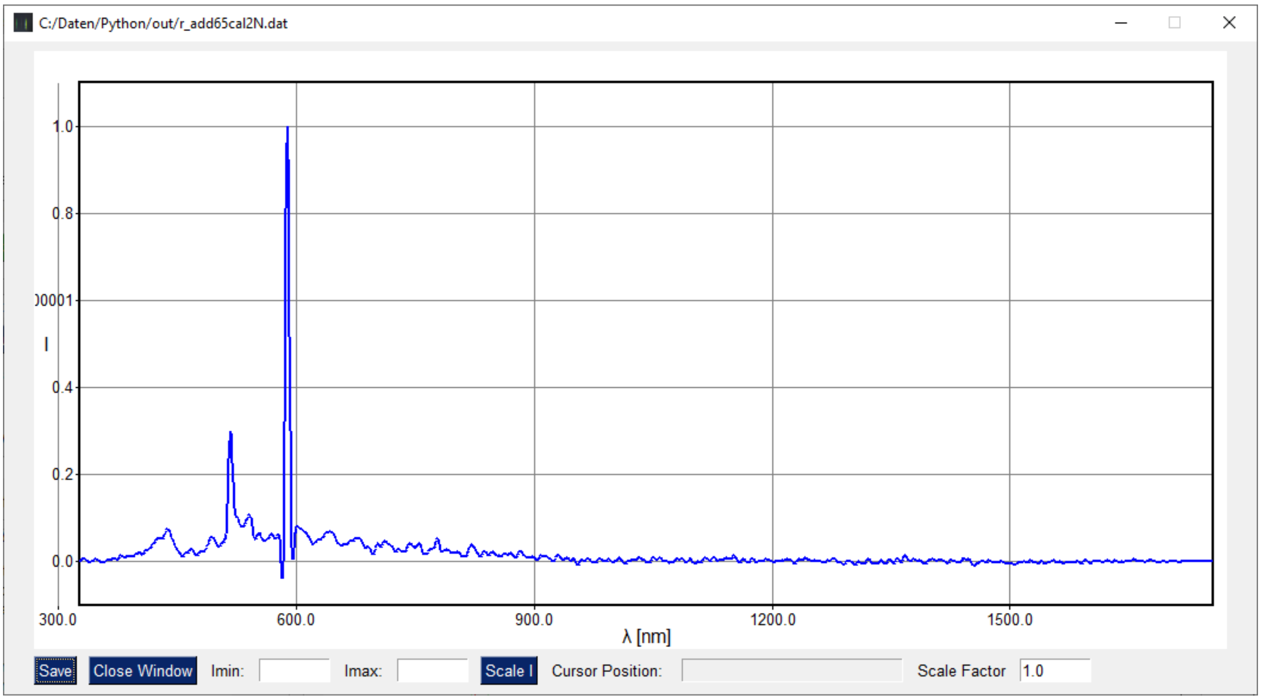


In order to get a nice view it may be necessary to scale the intensity of the spectra. This can be done with Multiply spectrum by a factor or Divide spectrum by a factor. The factor you select in the text input box at the bottom right of the plot window.

After the division or multiplication you have to store the scaled spectrum with right click – 'Save modified spectrum'

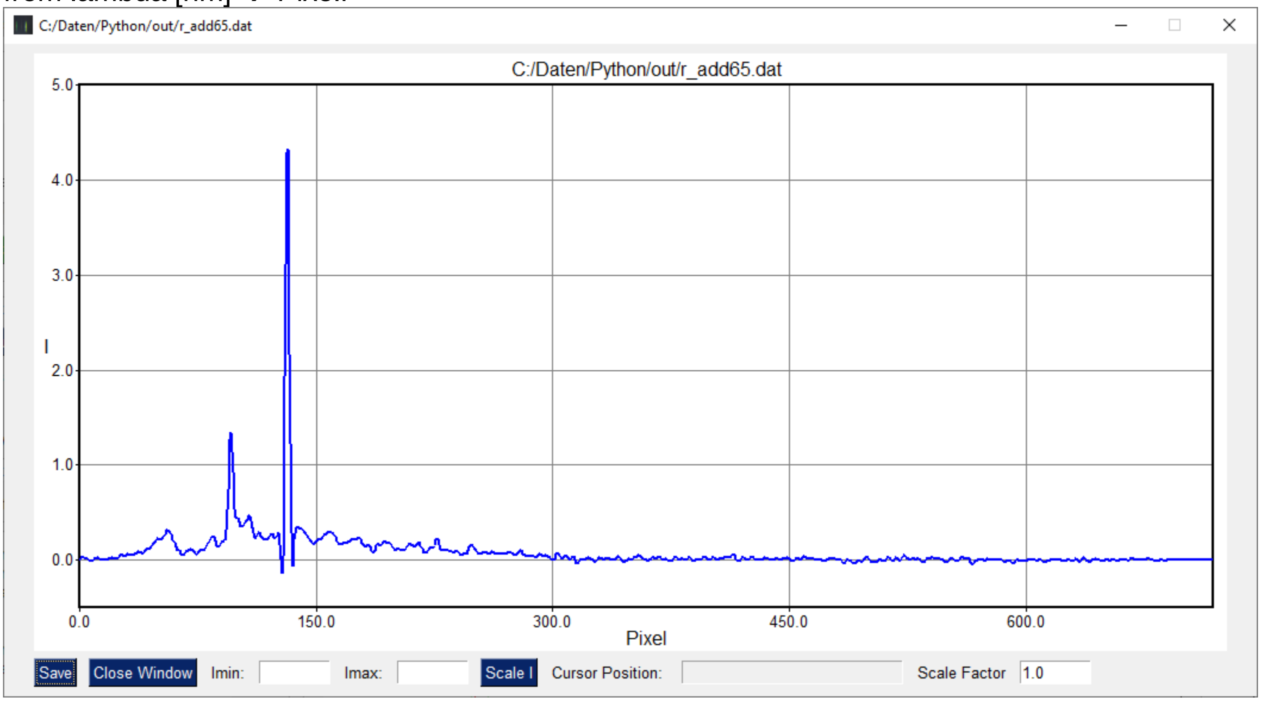
You also have the possibility to normalize the spectrum, so that the highest peak (often the zero order) has intensity one. Select right click – 'Normalize spectrum'. The file is stored automatically with 'N' added to the filename. r_add65cal2.dat → r_add65cal2N.dat. Close the window and plot again. In this case you can also use a normalized plot range by checking 'Norm scale' in the plot frame:





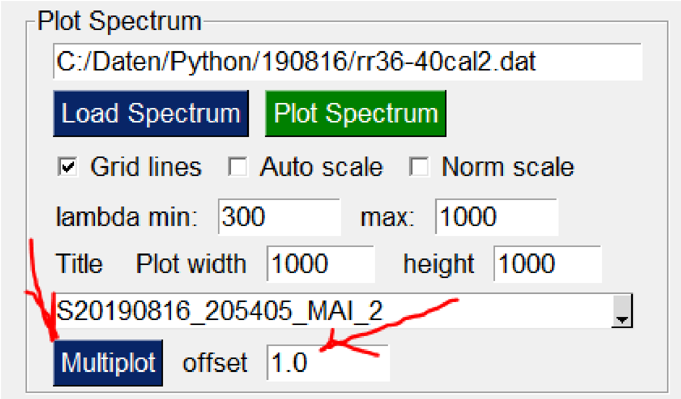
Plot raw spectrum

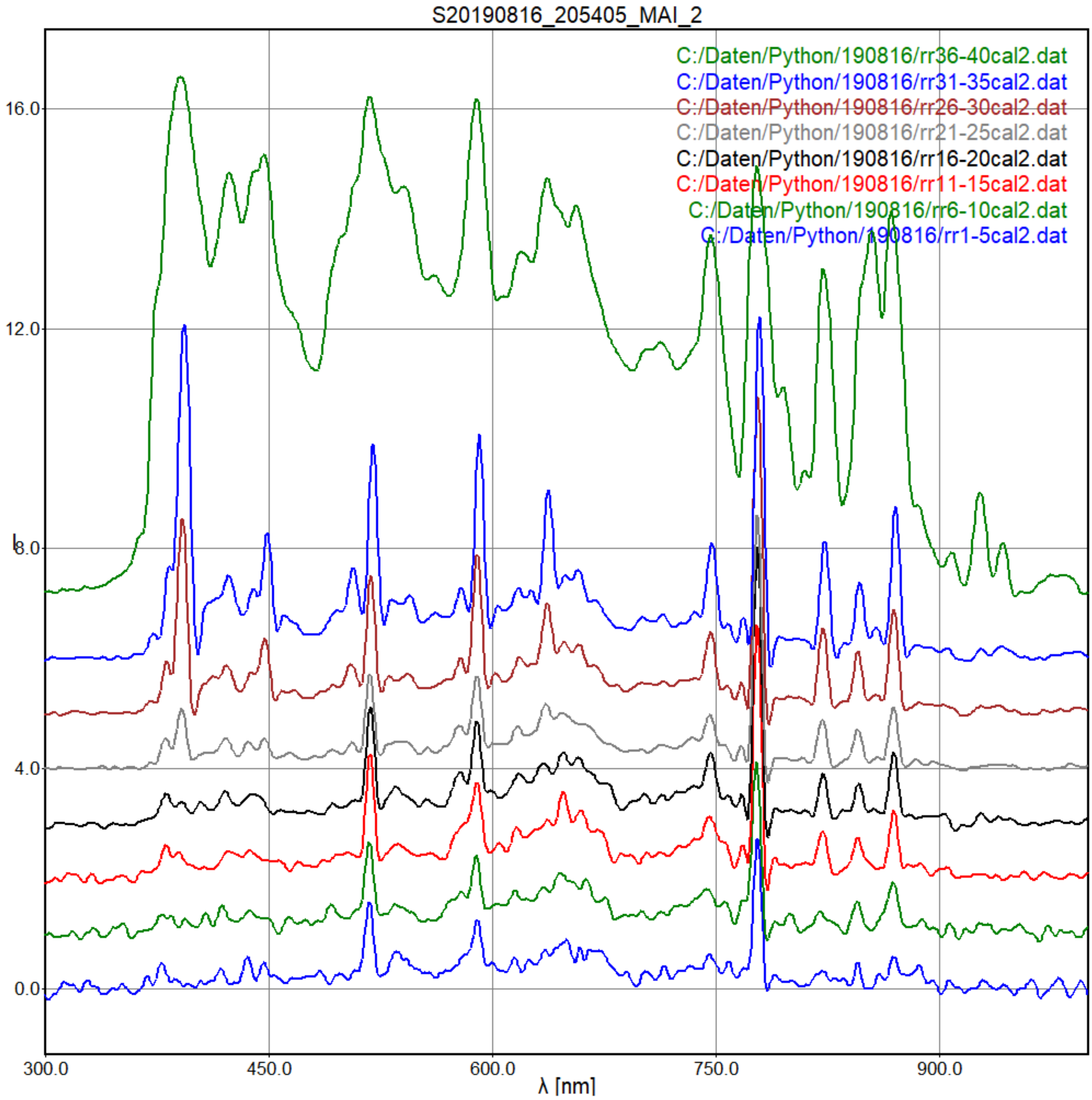
You may also plot the raw (uncalibrated) spectrum. In this case the label on the x-axis changes from lambda [nm] → Pixel:



Multiplot

If you would like to plot several spectra for comparison, as for a time series, it is also possible to plot several spectra in a single figure. Go to the calibration tab, select the wavelength range and the spacing of the different spectra with offset. A positive offset means that spectra are plotted from bottom to top, a negative offset plots them from top to bottom. The legends are sorted accordingly. With ‘Multiplot’ you can select the desired spectra and plot them:

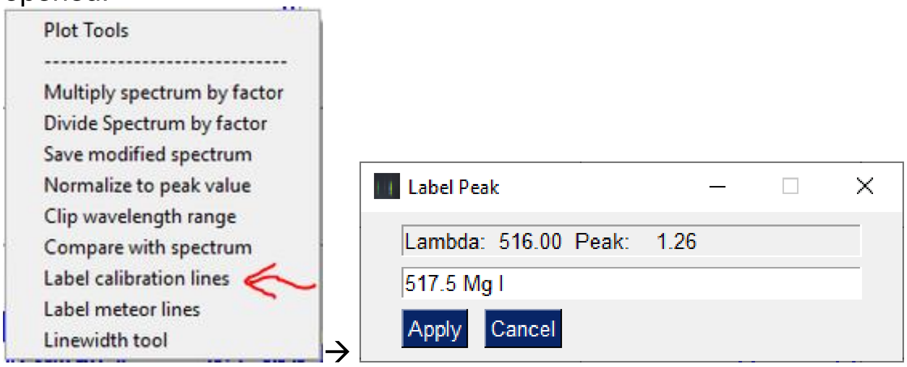




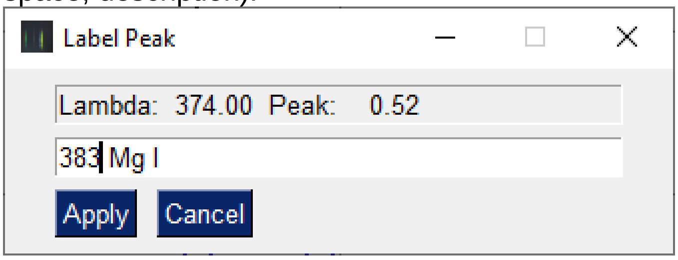
the last image (at the top) is quite strongly saturated! notable is also the increase of the Ca-Line.

Label spectra

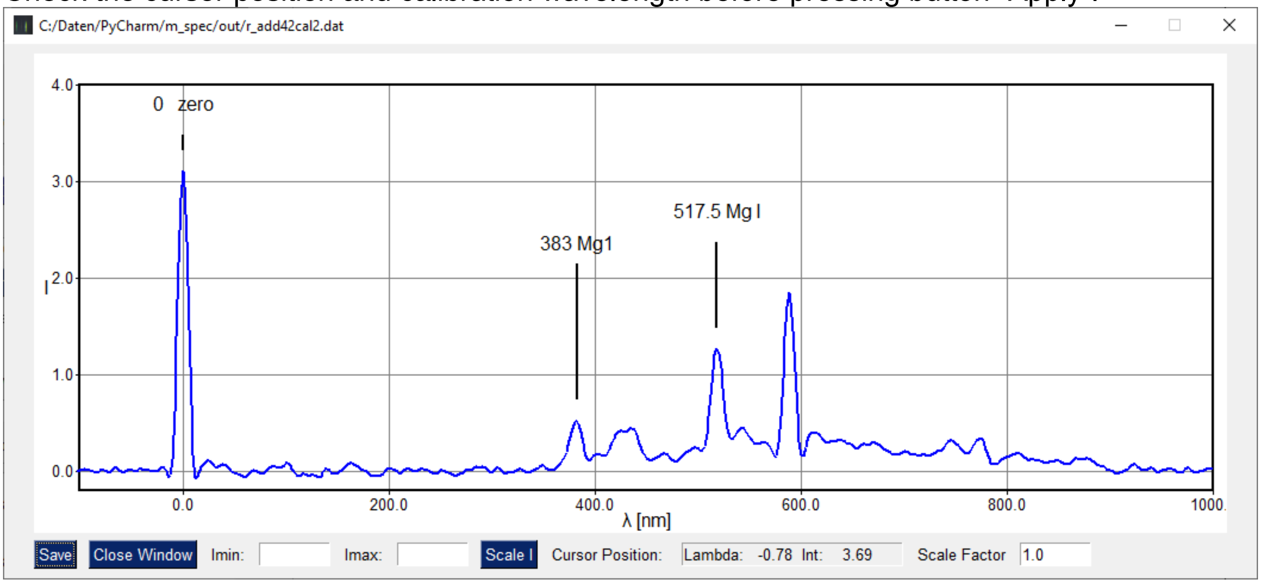
Sometimes it is useful to label peaks in the spectrum. This can be done by clicking above a peak in the spectrum. This records the click position. With a right click the label window can be opened:



In the upper field the click position of the wavelength and peak intensity at this wavelength is shown. The script searches in the file m_linelist.txt for the closest matching reference line. If the chosen peak is not in the linelist, you can edit this field (use the same format: wavelength, space, description):



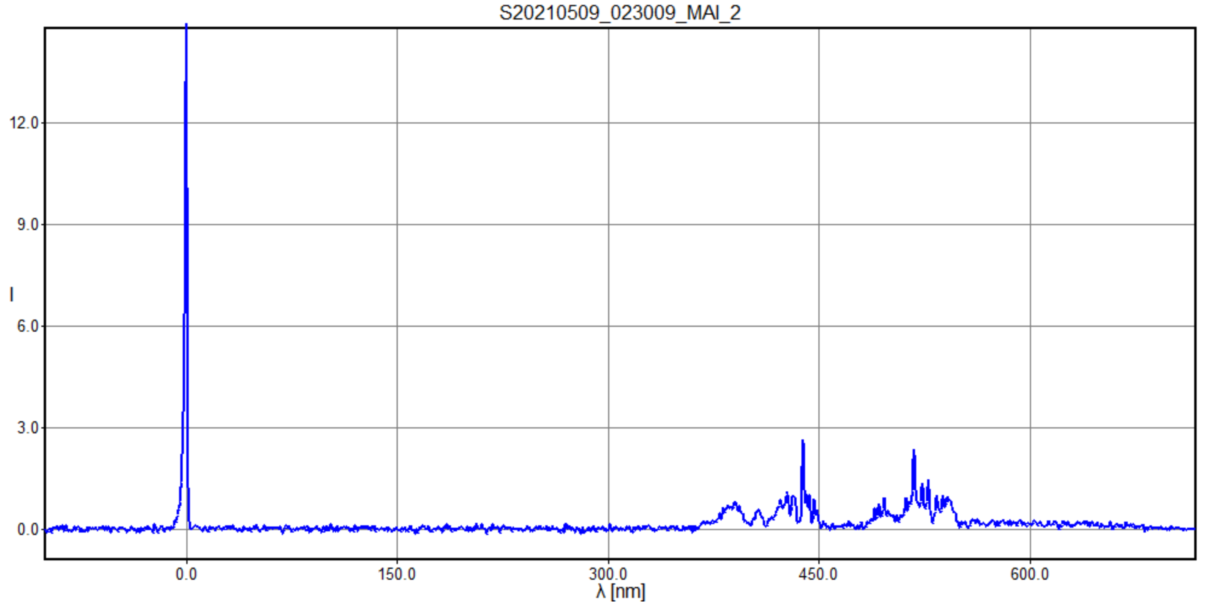
The peak is marked at the indicated wavelenth. The y-coordinate of the label is given by the y-coordinate of the click position (shown in the plot window Cursor position at bottom right). Check the cursor position and calibration wavelength before pressing button “Apply”.



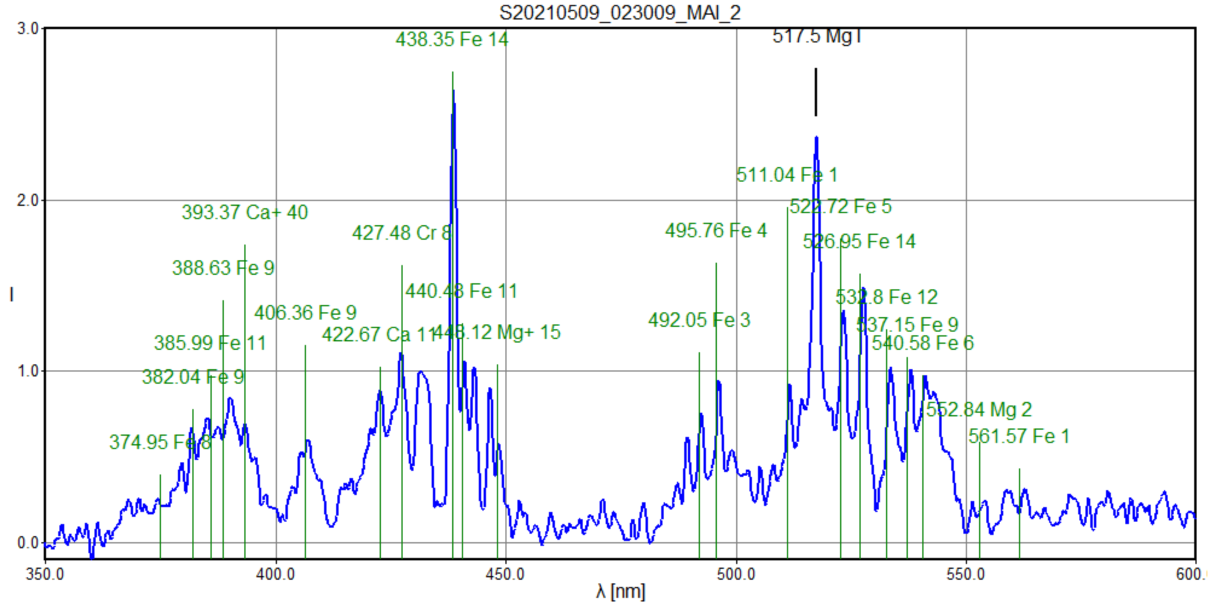
If this feature is used often, the calibration line list can be edited. This may also be necessary if you work at higher resolution. Right now, the wavelength for close-spaced lines is given as a weighted average (for Mg and Na lines). If these lines are resolved in the spectrum, it is advisable to list the components separately, with wavelengths taken e.g. from NIST: https://physics.nist.gov/PhysRefData/ASD/lines_form.html

Label meteor lines

For labelling more lines the following list has been added to m_spec as meteor_lines.txt. The wavelengths have been converted to nm as used throughout m_spec. They can be added to the plot by clicking above the chosen line and then right click – Label meteor lines. The following spectrum was recorded with a TIS DMK 33GX249 camera and a Kowa LM16HC f: 16mm F/1.4 lens with a 600 L/mm grating:



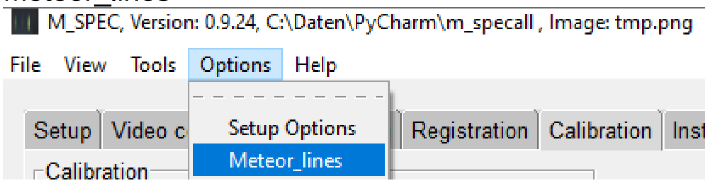
With the first order above 550 nm outside the field of view:



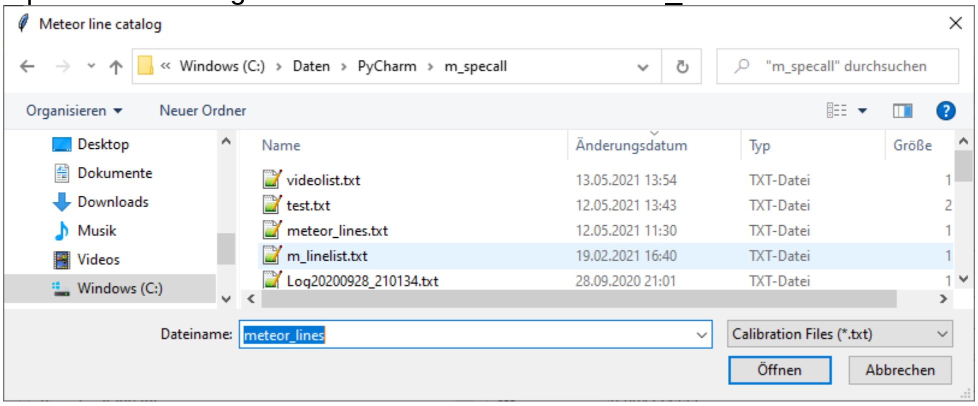
Showing most of the lines from the IMO photographic handbook list.

The file meteor_lines.txt is given as a template for your own list. The intensities after the element identification may be omitted for clarity. You may edit this list for your convenience, adding more lines, depending on your resolution. The format of the list is wvelength (floating point variable), space, line name (text string).

meteor_lines.txt is loaded as default. Another list can be loaded with Menu – Options – Meteor_lines



Opens a file dialog to choose another list of meteor_lines:



If you use always your own list, make this your default by renaming it to meteor_lines.txt.

Table 3-7: List of spectral lines frequently found in meteor spectra and their relative intensities. Th tification of the lines (numbers) in our example is also given. Lines marked with an asterisk appear b spectra of fast meteors, such as the Perseids, but much fainter in spectra of slow meteors.

Laboratory data			ident.	Laboratory data			ident.
λ_{lab}, [Å]	atom/ion	intensity	number	λ_{lab}, [Å]	atom/ion	intensity	number
3719.9	Fe	10	2	4923.9	Fe^+	2*	
3734.9	Fe	8		4957.6	Fe	4	
3737.1	Fe	9	3	5012.1	Fe	1	
3745.6	Fe	8		5018.4	Fe^+	3*	
3749.5	Fe	8		5110.4	Fe	1	
3820.4	Fe	9		5167.3	Mg	17	
3825.9	Fe	8		5172.7	Mg	25	
3829.4	Mg	10		5183.6	Mg	28	
3832.3	Mg	11		5208.4	Cr	10	
3838.3	Mg	12		5227.2	Fe	5	
3859.9	Fe	11		5269.5	Fe	14	
3886.3	Fe	9		5328.0	Fe	12	
3933.7	Ca^+	40*	8	5371.5	Fe	9	
3968.5	Ca^+	35*	9	5397.1	Fe	5	
4030.8	Mn	10		5405.8	Fe	6	
4045.8	Fe	10		5429.7	Fe	6	
4063.6	Fe	9		5434.5	Fe	4	
4131.0	Si^+	1*		5446.9	Fe	4	
4226.7	Ca	11	12	5455.6	Fe	4	
4254.4	Cr	9		5528.4	Mg	2	
4271.8	Fe	10		5615.7	Fe	1	
4274.8	Cr	8		5890.0	Na	40	
4289.7	Cr	7		5895.9	Na	35	
4307.9	Fe	10		6156.8	O	1*	
4325.8	Fe	10		6162.2	Ca	1	
4383.5	Fe	14	15	6347.1	Si^+	6*	
4404.8	Fe	11		6371.4	Si^+	3*	
4481.2	Mg^+	15*		6495.0	Fe	1	
4920.5	Fe	3		6562.9	H	2*	

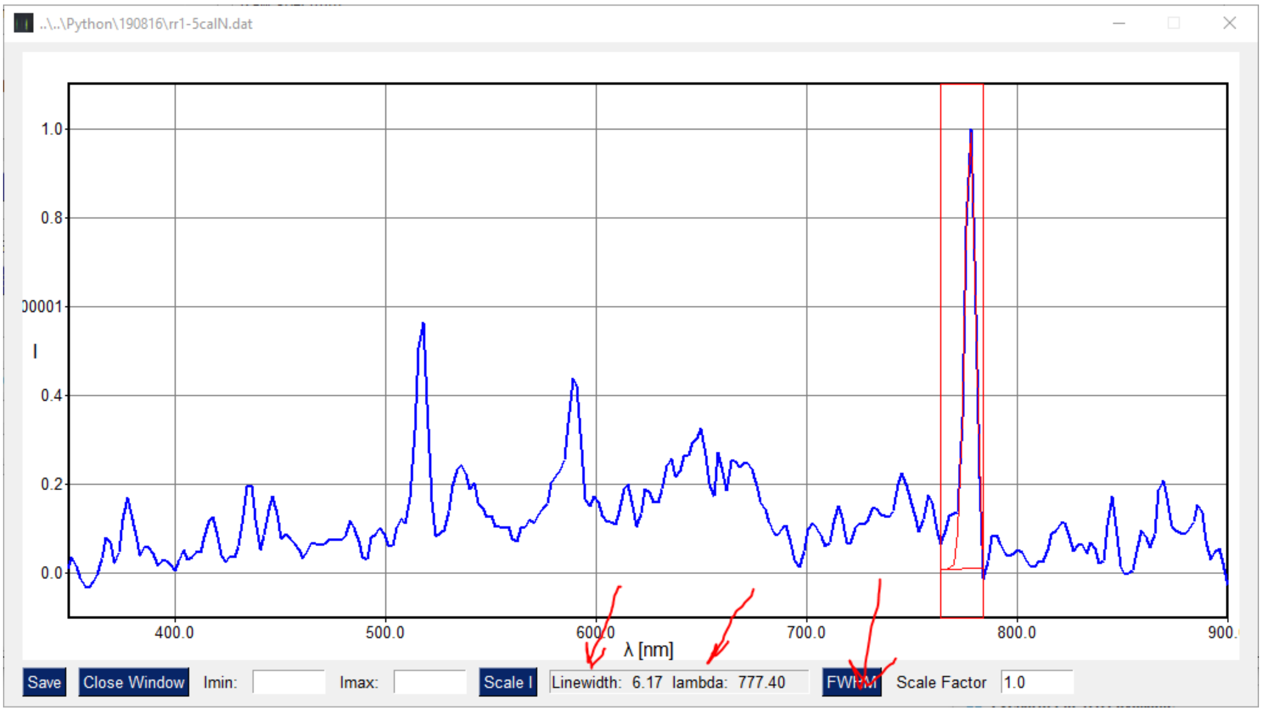
From: Spectral lines, (IMO PhotographicHandbook 03 Spectra, p 47) http://www.imo.net/docs/03spectra.pdf

Example

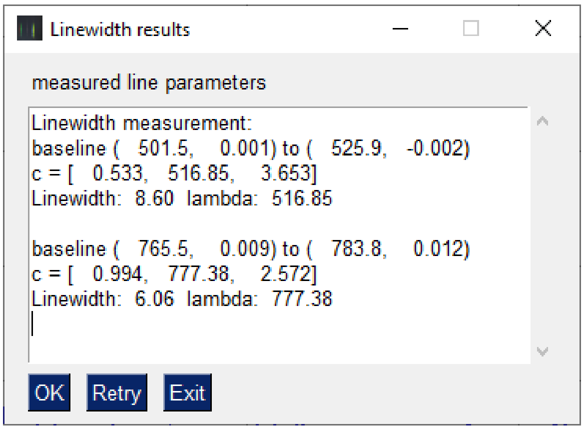
For testing the new features, a folder 210509 has been added with a spectrum r_add8.txt, obtained from the spectrum M20210509_023009_MAI_2.avi (the same as shown above). Normally the spectra file have an extension '.dat', but this is not allowed in github. Therefore the extension has been changed to '.txt', which can be loaded by selecting file type = ALL. After loading the txt-file, a file with extension '.dat ' is created. This can be calibrated with the zero order and the known dispersion of the camera DMK 33GX249 of 0.599 nm/pixel. The spectrum does not look familiar, but with the rough calibration the Mg-line at 517.5 nm may be identified and used for a two-line linear calibration r_add8cal.dat. With this spectrum the clipping and normalization can be tested as well as the labelling of many of the lines.

Linewidth tool

The linewidth of meteor lines is an important measure of the quality of a spectrum. It depends on the camera pixel size, focal length of the camera lens, number of lines of the grating, meteor movement, focusing and quality of the lens. A common measure for the linewidth is FWHM (full width at half maximum). As there are only a few data points over the full width, the measurement requires interpolation. This can easily be done by fitting a Gaussian line-shape. This works actually quite well. The biggest problem is the correct level of the baseline. Many lines are overlapping or on top of a continuum background, where the selection of the baseline is critical. In the case of overlapping lines it also advisable to fit only the non-overlapping part of the line.
The linewidth tool is selected by right click – linewidth tool
This opens a result window.
The line to be measured is selected by drawing the mouse over the width of the lines.
The baseline is then defined by the start and endpoint of the mouse draw. For a well isolated line the width of the draw should be about 2 to 3 FWHM and is not very critical. More critical is the level of the baseline. Check it out yourself by measuring different lines with different baselines.



The result (Linewidth FWHM and center wavelength) are indicated in a text window, normally used for the cursor position. For a permanent record, press the the FWHM button.
A small result window opens:



For saving the result, press OK. This writes the baseline, FWHM and center wavelength to the logfile.
With “Retry”, the measurement is shown in the linewidth results window, but not saved to the logfile
In order to finish the linewidth measurement you have to press the button FWHM and then the “Exit” button in the Linewidth results window.
A note on the measurement of spectral resolution. The zero order line is well isolated, but often overexposed, which leads to line broadening, the Na or Mg lines are actually unresolved multiple lines (at least for low resolution spectra). This should be remembered when measuring the spectral resolution.

Different calibration methods

Use of spectrum lamp for calibration

For cameras with high dispersion, either with a long focal length or with a grating with large number of grooves/mm, not enough orders of the laser may fit on the detector size. Maybe a suitable laser is not at hand. In these cases it is possible to use a calibration lamp with several emission lines for calibration. Suitable lamps are low pressure Hg-lamps or Hg-Ar lamps. I have used successfully a lamp from Ocean Optics (http://oceanoptics.com/product/hg-1/) with several lines from UV to near-IR. An example is given in https://meteorspectroscopy.files.wordpress.com/2018/01/meteor_spectroscopy_wgn43-4_2015.pdf, although with low dispersion and somewhat problematic separation of spectral lines.

Here an example with longer focal length is used (spectra recorded by Sipeng Yang). With a violet laser, only three orders fit on the image area, not enough for a good determination of the distortion parameters. With a green or red laser, only two orders fit in the image area, definitely not enough for a determination of the distortion parameters.
With the Hg-lamp, several spectral lines in addition to the zero order can be used for the fit:

Load image: calib/Sipeng/cal1_peak_5.png
5 spectra of a mercury argon lamp are shown, zero order at left, with first and order lines to the right. The spectra were recorded by Sipeng Yang[2], thanks for using these files. Equipment:
1. Watec 902H2 ULTIMATE 1/2' CCD, 720*576 Resolution
2. Lens: Hikvision HV0733D-6MP, F/0.95, f = 7-33mm zoom lens, operated at approx. f = 10mm
3. Grating: Edmund 600l/mm transmission grating
4. Camera FOV: 30.4 degree

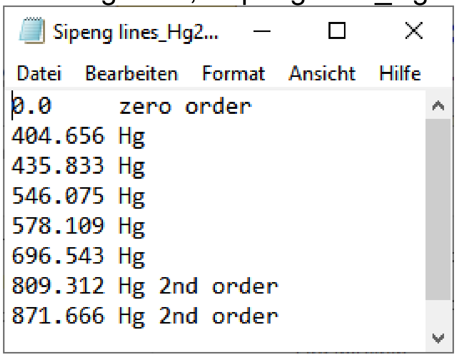
This is particularly useful for lenses with longer focal length and/or high resolution spectra and needs some knowledge about the spectrum of the lamp (which spectral line is where)
The bright line at left is the zero order, 5 Hg-lines 1st order are visible. Some second order lines are visible as well and can also be used for calibration.

Line list

When working with a spectral lamp, a slightly different procedure from the laser calibration will be used. Instead of a single laser wavelength, a list with the wavelengths of the spectral lamp in

[2] Beijing National Observatory for Space Environment
Institute of Geology and Geophysics, Chinese Academy of Sciences
NO.19 Beitucheng Xilu Street, Chaoyang District, Beijing, China 100029

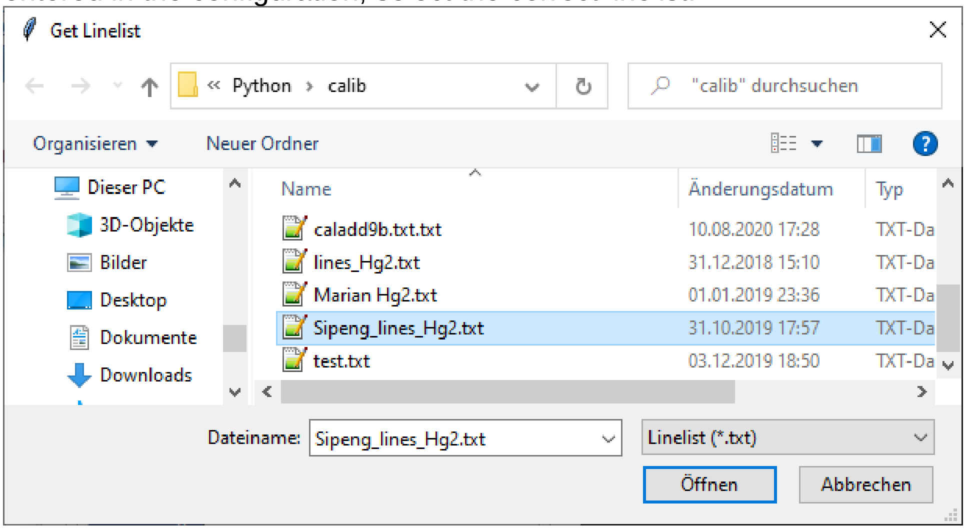
different orders is required. This is a text file with two columns, the first column contains the wavelength*order (negative for negative orders, zero for zero order, wavelength for first order, 2*wavelength for 2nd order etc. as used here also for the plotting of spectra. The 2nd column must be separated by a tab and contains text information (element, order or whatever you like to add). The script actually reads only the first column, but make sure you end the column with a tab for compatibility with other scripts, which will use this file. It can be created with a text editor (Windows Editor or Notepad++ which I prefer). For the image of Hg spectra above I used the following table, "Sipeng lines_Hg2.txt":



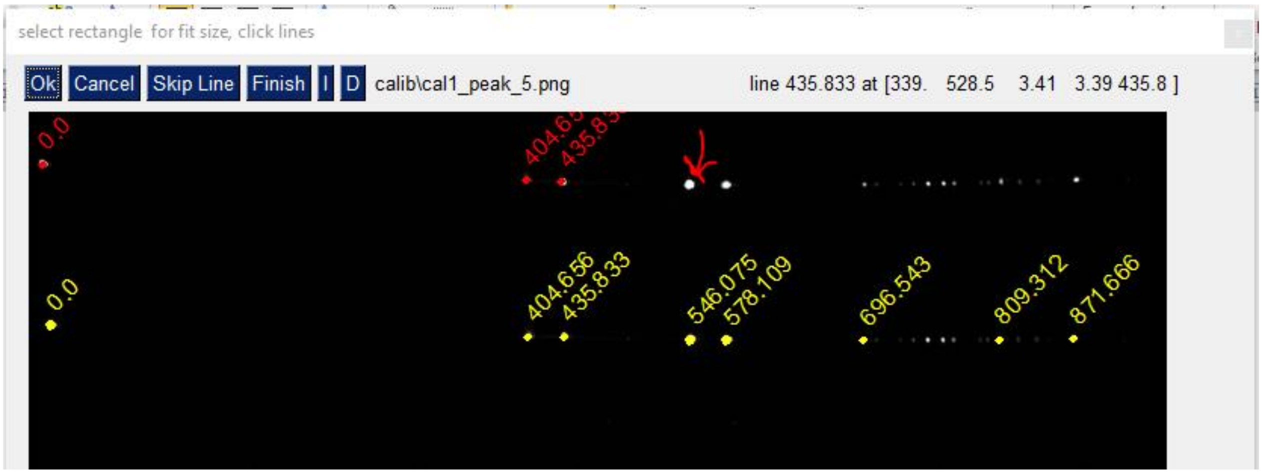
The line at 578.109 nm is actually the weighted average of two lines at 576.96 and 579.066 nm.

Run the script for spectral lamp calibration

Run the script as before, with the configuration 'm_set_sipeng.ini' and if the linelist has not been entered in the configuration, select the correct linelist:



You can also change the linelist manually in the setup.
Finish the setup with 'Apply'.
Load the image and go to 'Select Lines'.

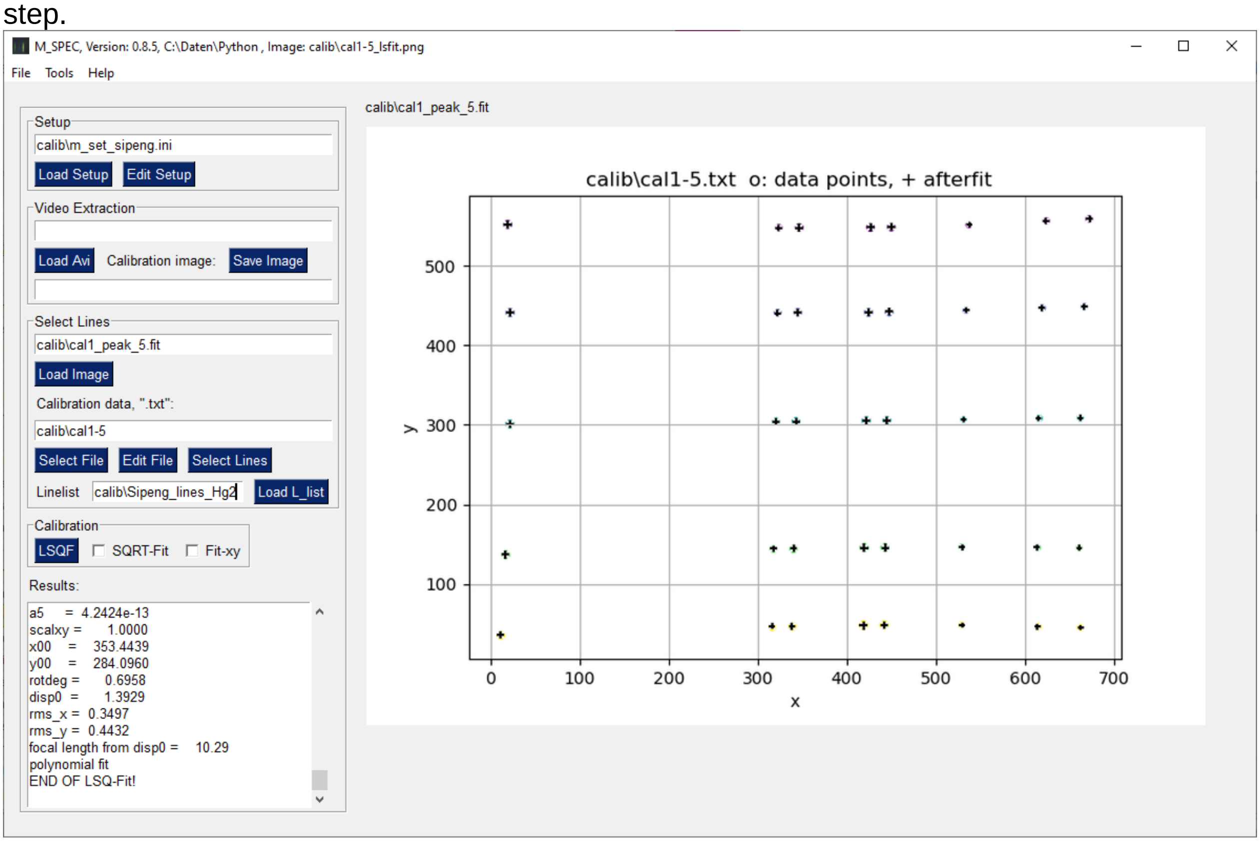


The selected lines are marked by a circle with wavelength*order indicated. They have been selected to cover the full width of the sensor with regular spacings. Some weak lines have been omitted. Finish each spectrum with **OK**

Continue with the other spectra, Skip missing lines with ‘Skip Line’ (e.g. if the leftmost line is out of the image area or if a line is too weak). Select only the lines which you can assign correctly. Finish the list with **OK** again.

It is important that you choose the correct lines! If you select the wrong lines, the errors will be large. Start over with a new calibration file and select the correct lines!

Proceed the same way as for the laser calibration (edit the list if necessary, then go to the LSQF step.



If you would like to run the least square fit with other parameters (a sqrt fit would be suitable here for the fairly long focal length) go to the Setup and edit the corresponding field(s).

The results are from the log file:

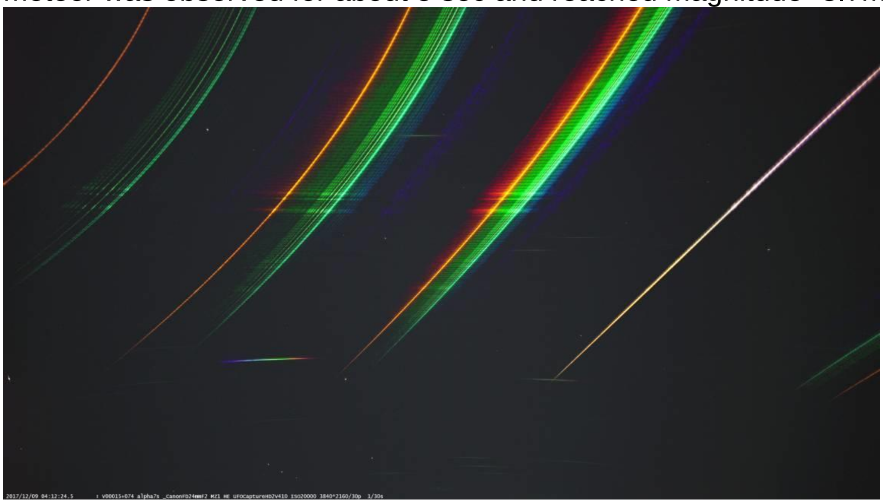
Result LSQF: **polynomial fit**	Result LSQF: **square root fit**
============	============
Watec 902 H2 ultimate, f = 10mm	Watec 902 H2 ultimate, f = 10mm
Sipeng\lamp_video\cal1-5.txt	Sipeng\lamp_video\cal1-5.txt

Parameters after fit:	Parameters after fit:
a3 = 1.7052e-07	a3 = 2.7372e-07
a5 = 6.5356e-13	a5 = 1.1239e-13
scalxy = 0.9240	scalxy = 0.9671
x00 = 352.6166	x00 = 354.5030
y00 = 283.4155	y00 = 282.9148
rotdeg = 0.6431	rotdeg = 0.6752
disp0 = 1.3906	disp0 = 1.3969
rms_x = 0.3387	rms_x = 0.3248
rms_y = 0.4269	rms_y = 0.4764
focal length from disp0 = 10.31	focal length from disp0 = 10.26
polynomial fit	for sqrt fit: feff = 11.62
END OF LSQ-Fit!	END OF LSQ-Fit!

Since the errors of the fit are similar, I would prefer the sqrt fit, but this is a matter of taste. Another possibility would be to constrain the value of scalxy to a more realistic value, as determined by UFO capture or calculated from the image size and pixel size and detector size. An rms error of < 1 pixel is the acccuracy which can be expected, subpixel accuracy is achieved with determining the line positions with a Gaussian fit.

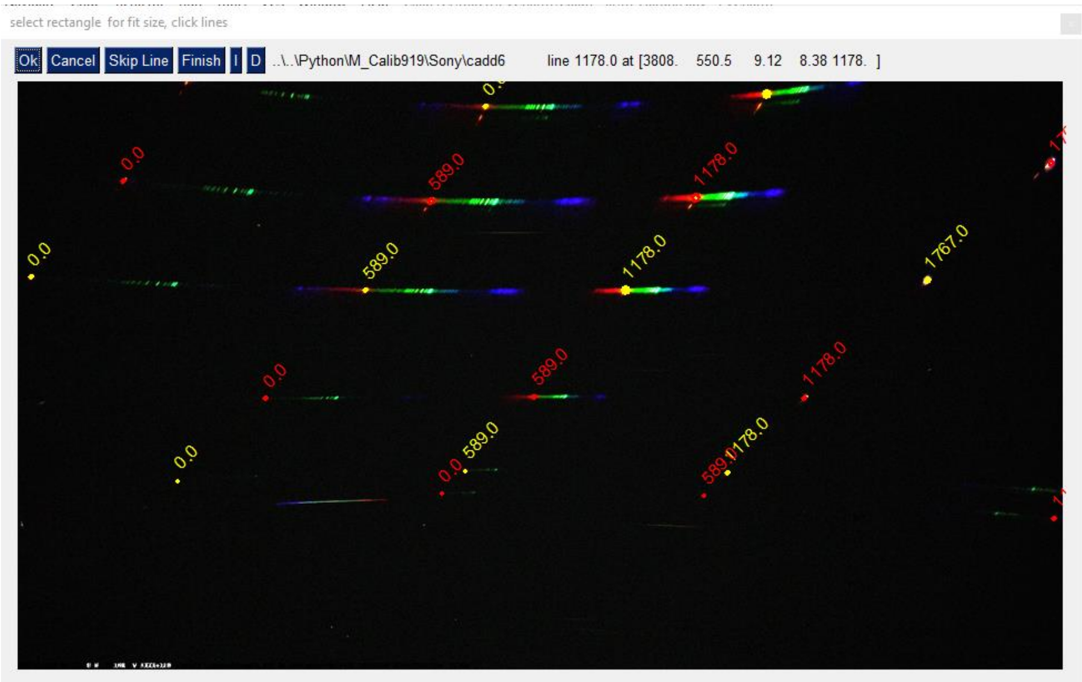
Use of meteor spectra for calibration

When a meteor spectrum with identified spectral lines covers a large portion of the image area, it can also be used for the determination of the distortion parameters. This is useful if no laser or spectral lamp spectra have been recorded or if the camera setup has been changed in the meantime. I present an example which has previously been analysed with the old method. The spectrum was recorded by Koji Maeda with a Sony Alpha 7S equipped with a Canon 24 mm F/1.4 lens (used at F/2) and a 600 L/mm grating. The meteor spectrum was captured as a 4K video (image size 3840 x 2160 pixels) at 30 images/sec with SonotaCo UFOCaptureHD. The meteor was observed for about 5 sec and reached magnitude -3.7m.



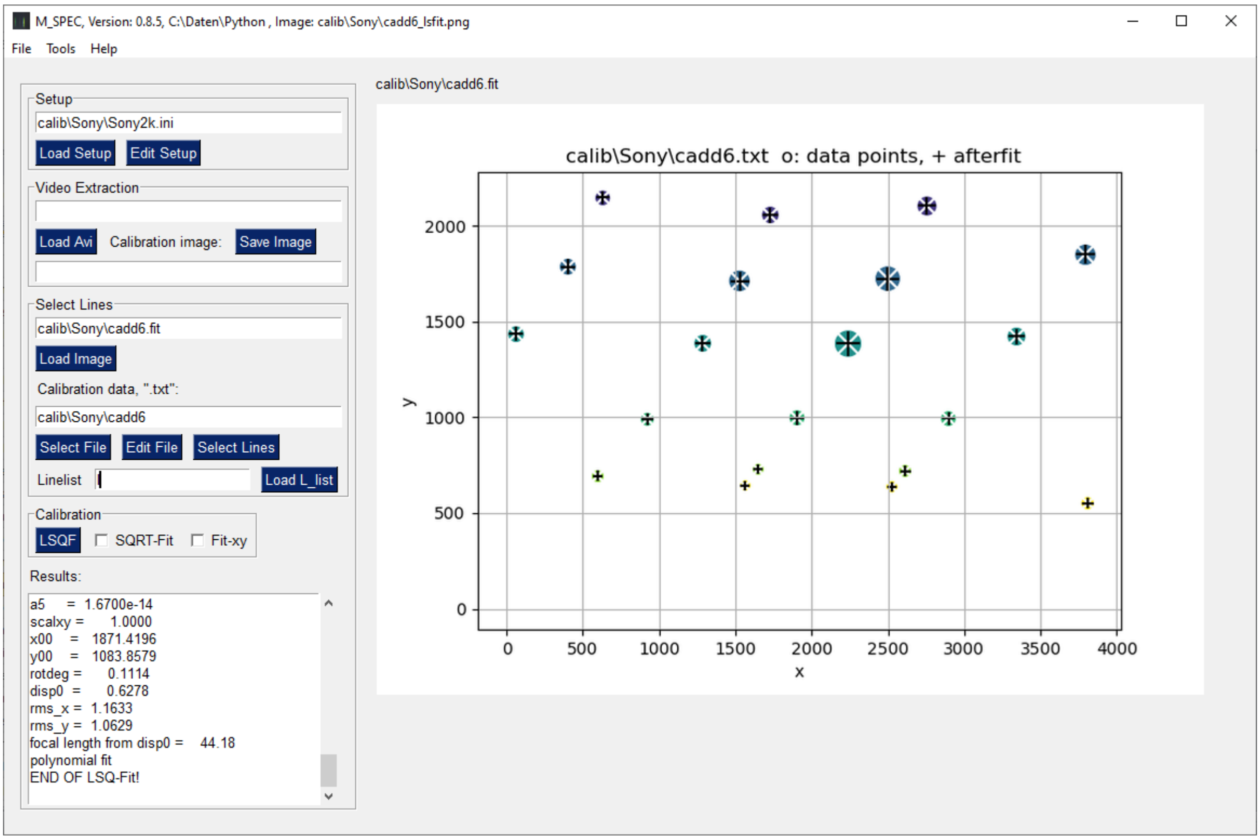
M20171209_041224_JPMZ1, Koji Maeda

The Na-lines are easily identified and can be used for calibration. The spectrum pre-processing was done with m_pipe6 including the AVI conversion and background subtraction.
Several colour fit-images were peak-added and saved as cadd6.fit.



For simplicity I only used the Na-lines, therefore no list of spectral lines was needed, I used the “laser” calibration. Notice that the orders are not correct, important is that the wavelength differences are constant between orders.

With the fit:



The size of the data points corresponds to the Gaussian width of the lines, increased by saturation, movement of the meteor and meteor train

The fit parameters agree with earlier calculations:

Result LSQF:

============

Sony alpha, f 24 mm

sony/cadd6.txt
Parameters after fit:
a3 = 4.3292e-08
a5 = 1.6700e-14
scalxy = 1.0000
x00 = 1871.4196
y00 = 1083.8579
rotdeg = 0.1114
disp0 = 0.6278
rms_x = 1.1633
rms_y = 1.0629
focal length from disp0 = 22.09
polynomial fit
END OF LSQ-Fit!

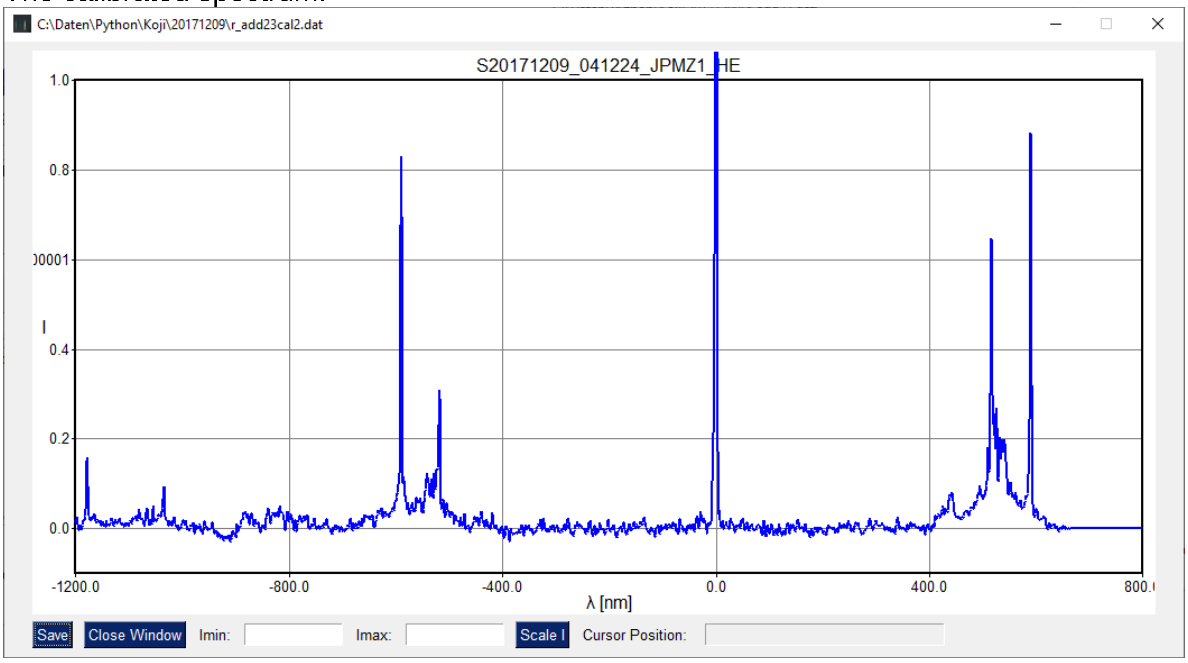
With these parameters, also saved as m_set.ini, the full spectrum can be analysed.

Analysis of spectrum M20171209_041224_JPMZ1

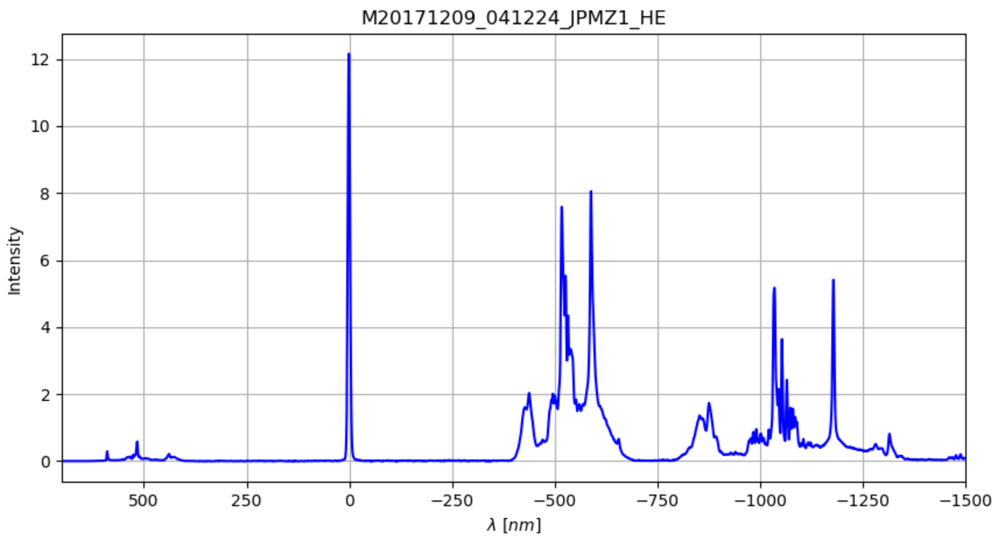
(see: http://sonotaco.jp/forum/viewtopic.php?p=47795#47795)
The distortion parameters obtained from the calibration with the Na-lines were used to calibrate the meteor spectrum, analysed with m_spec.
After adding 23 registered files (about the 1st second of the meteor (near the bottom of the image9 and calibrating with Mg- and Na-lines the following result was obtained:
A linear fit was used:
polynom for fit lambda c: [1.259e+00 -1.503e+03]

pixel	lambda	fit	error
257.95,	-1178.00,	-1177.83,	0.1668
725.21,	-589.00,	-589.48,	-0.4819
1193.86,	0.00,	0.62,	0.6196
1603.37,	517.50,	516.25,	-1.2451
1661.89,	589.00,	589.94,	0.9405

rms_x = 0.7847
spectrum C:\Daten\Python\Koji\20171209\r_add23cal.dat saved
Not surprisingly, a linear fit works well.
The calibrated spectrum:



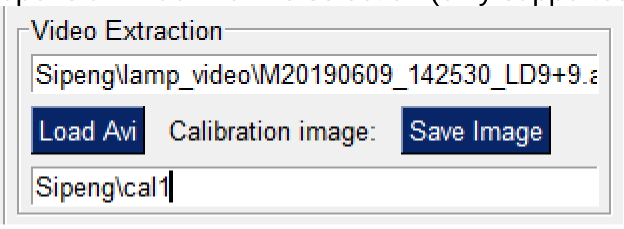
Below, the spectrum obtained from adding all image frames(123)



The spectrum was plotted with reverse wavelength scale, to show the strongest lines to the right of the zero order. The actual first order visible around 500 nm was very weak, because it was mostly outside the image area.

Creating calibration images from video files

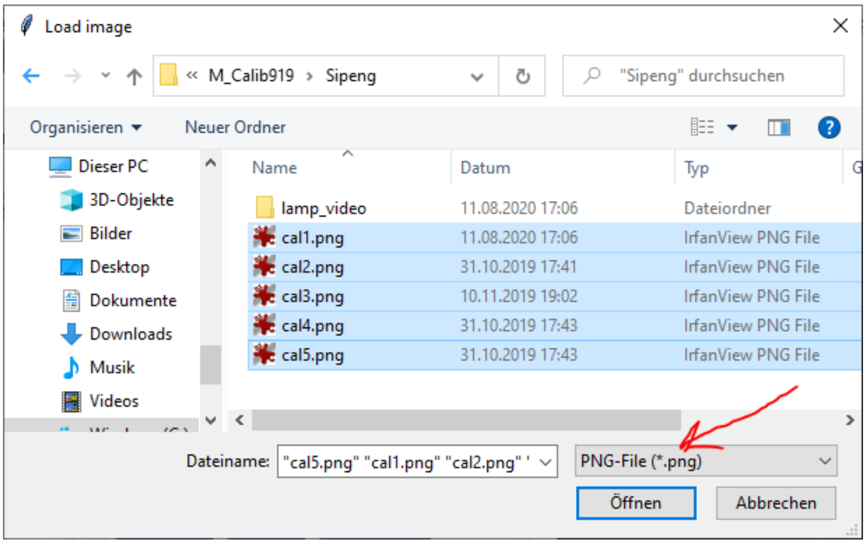
For the previous versions of this calibration script, the images used for calibration had to be created with different software (IRIS). In order to streamline the workflow, the creation of calibration images has been included in this script. For calibration images, short video clips of the laser or spectral lamp are recorded. These can be read with the button ‘Load Avi’, which opens a window for file selection (only supported format AVI, as produced by UFO capture):



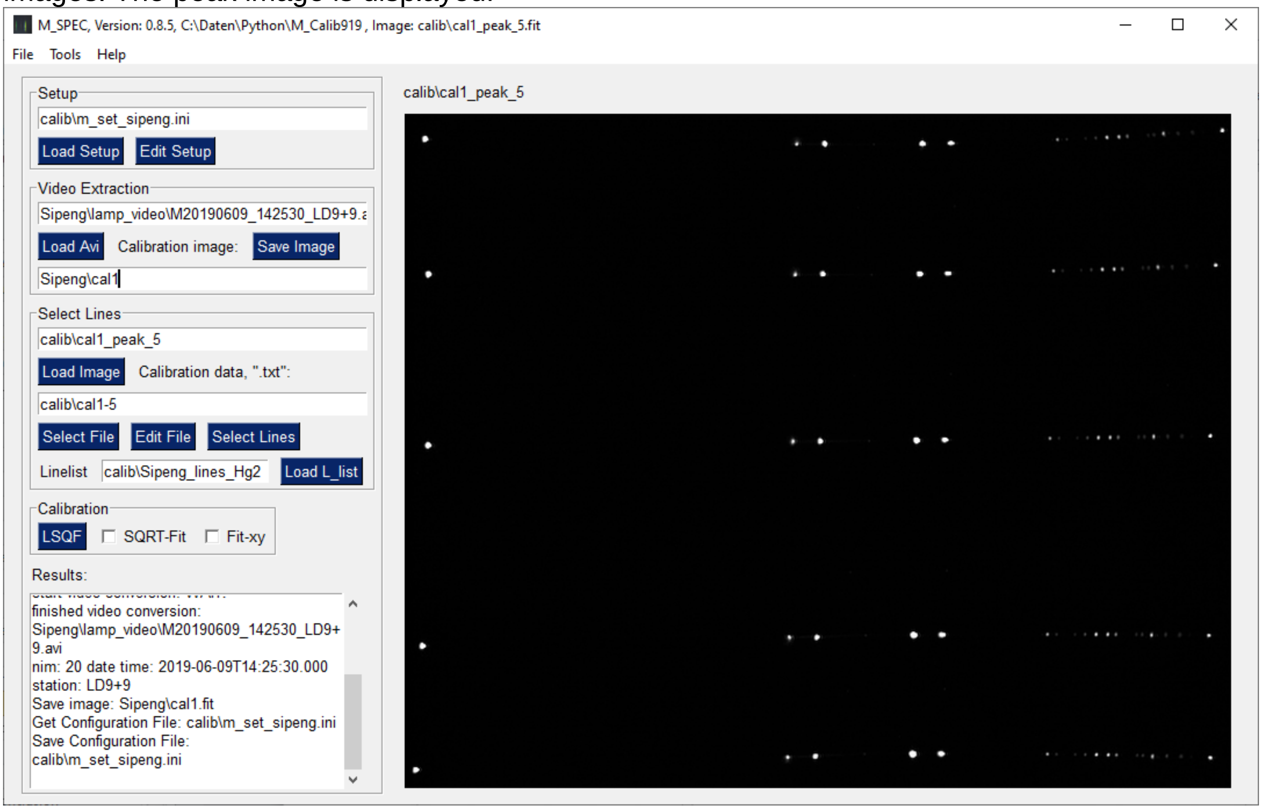
From the selected video file, a number of frames (20 at present) are converted to png images and from these a peak image and an average image are calculated with ‘Load Avi’. After a short delay the image can be saved with an appropriate filename by selecting the directory with the browser button ‘…’ and adding the filename in the image window:
If the videos are not in the demo files (for saving file size), try with your own meteor video file (a background image will be produced).

Making peak images from single calibration images

You can load an image with a single spectrum, determine the line positions and then load another image, repeating these steps until all spectra are measured. However, it is more convenient to create a peak image from all the relevant calibration spectra. Previously this had to be done with IRIS or some other imaging software. In the present version of m_calib, this is simplified considerably. Instead of loading the image files one by one, the whole series can be selected at once in the browser ‘…’ for the actual image:

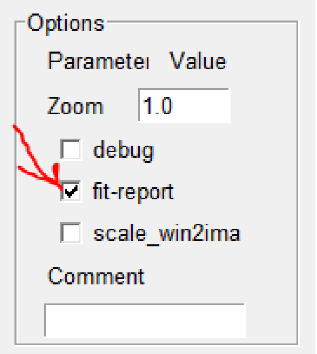


This creates two images (both in FIT and PNG format) named cal1_peak_5.png and cal1_ave_5.png(the filename is created from the name of the first image plus ',-peak-' plus number of images. Notice that the peak images can be produced from *.fit, *.png and *.bmp images. The peak image is displayed:



Detailed results of Least Square Fit

A detailed view of the results of the fit is given with the command “**report_fit(out)**”, which shows statistical errors, correlations etc. in the shell window for the interested reader. It can be switched on in the setup options:



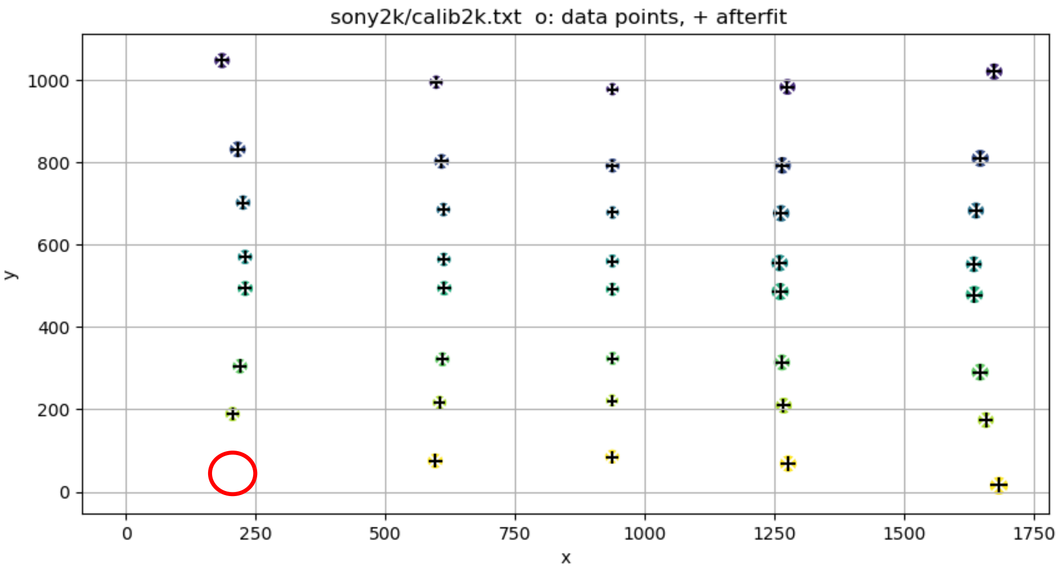
For practical purposes the average rms error of the fit in x- and y-coordinates rms_x and rms_y are shown. With a well corrected and focused lens these values should be < 1 pixel. With noise free and aberration free synthetic data produced with ImageTools, rms-errors were < 0.05 pixel.

```
>>> report_fit(out)
[[Fit Statistics]]
    # fitting method   = leastsq
    # function evals   = 210
    # data points      = 108
    # variables        = 25
    chi-square         = 10.1468068
    reduced chi-square = 0.12225068
    Akaike info crit   = -205.416995
    Bayesian info crit = -138.363714
[[Variables]]
    x0_0:    49.0181527 +/- 0.40095395 (0.82%) (init = 48.39159)
    y0_0:    501.714124 +/- 6.76732160 (1.35%) (init = 515.1632)
...
    scalxy:  0.85217823 +/- 0.03405730 (4.00%) (init = 0.92)
    x00:     344.900732 +/- 1.73905549 (0.50%) (init = 345.386)
    y00:     327.219815 +/- 3.81784835 (1.17%) (init = 322.3328)
    rot:    -0.00578993 +/- 3.1517e-04 (5.44%) (init = -0.006249105)
    disp0:   1.99038480 +/- 0.00260282 (0.13%) (init = 1.987447)
    a3:      2.9385e-07 +/- 3.1388e-08 (10.68%) (init = 2.402625e-07)
    a5:     -1.8900e-13 +/- 1.8562e-13 (98.21%) (init = 8.658912e-14)
[[Correlations]] (unreported correlations are < 0.100)
    C(y0_7, y0_8)    =  1.000
    C(y0_0, y0_1)    =  0.999
```

Notice, that the scalexy and the coefficient a5 are not well defined by the chosen data points (4 points in one spectrum cannot define the 5th power of a polynomial.

Binning, directory structure

If you have different cameras, lenses, image formats you may need more than one configuration. In this case you can use different subdirectories or different ini-filenames for each case, in this example there is a directory “sony4k” for 4k images and a directory “sony2k” for binned 2k images. The only difference in the configuration is the binning = 2 for the 2k images, as compared to binning = 1 for the 4k images. Of course the results for [Calib] are different for the different binning factors; the values of x00 and y00 are 2 times larger for the unbinned images, the values of a3 and a5 4x respectively 16x smaller for the unbinned images. Notice also that the errors are much smaller in the 2k images. This is because the first line in the lowest laser spectrum was omitted in the 2k analysis (this is permitted for the first or the last order, but you may not skip intermediate orders), since it overlapped partly with the image caption:



The results are included in the configuration file, because they will be used in the analysis of the meteor spectra, for the calculation of the distortion to transform to orthographic projection.

```
m_set.ini - Editor
Datei  Bearbeiten  Format  Ansicht  ?
[Lasercal]
f_lam0 = 405.0
f_scalxy = 1.0
b_fitxy = 0
i_imx = 3840
i_imy = 2160
f_f0 = 24.0
f_pix = 0.00832
f_grat = 600.0
f_rotdeg = 0.0
i_binning = 2
s_comment = Sony alpha, f 24 mm
s_infile = sony2k/calib2k
s_outfil = sony2k/calib2k

[Calib]
x00 = 953.3508684132236
y00 = 547.0873636640123
rot = -0.012352211369287523
disp0 = 1.2763356272969484
a3 = 1.8386668878523818e-07
a5 = 2.031462462324112e-13
```

```
Sony alpha, f 24 mm
sony2k/calib2k.txt
rms_x = 0.2861
rms_y = 0.4682
parameters after fit:
…
\sony2k\m_set.ini saved
```

```
m_cal.ini - Editor
Datei  Bearbeiten  Format  Ansicht  Hilfe
[Lasercal]
f_lam0 = 405.0
f_scalxy = 1.0
b_fitxy = 0
i_imx = 3840
i_imy = 2160
f_f0 = 24.0
f_pix = 0.00832
f_grat = 600.0
f_rotdeg = 0.0
i_binning = 1
s_comment = Sony alpha, f 24 mm
s_infile = Sony/calib4k
s_outfil = Sony/calib4k
s_linelist = 1
b_sqrt = 0

[Calib]
scalxy = 1.0
x00 = 1908.5919
y00 = 1099.8146
rot = -0.0051855845
disp0 = 0.6342405
a3 = 5.0031932e-08
a5 = 1.3980166e-14
100%   Windows (CRLF)   UTF-8
```

```
Sony alpha, f 24 mm
Sony/calib4k.txt
rms_x =  1.5168
rms_y =  1.8173☹
parameters after fit:
… Save config in directory
Sony\Sony4k.ini
```

Content