

SpecINTI and specINTI Editor V4

Processing guide for spectra

Christian Buil & Valerie Desnoux

12 juillet 2026

Contents

1	Foreword	3
2	Organisation of this guide	3
3	Background	3
3.1	Extracting the spectral profile	4
3.2	Spectral calibration	4
3.3	Flux calibration	4
3.4	Reduction scenarios	5
4	Standard interface : low-resolution spectroscopy	8
4.1	Low-resolution calibration	9
4.2	Processing at low resolution	16
4.3	Processing the spectrum of an extended object	21
5	Standard interface : use in high spectral resolution	27
5.1	Calibration at high resolution	27
5.2	Processing at high resolution	35
5.3	Revisiting the instrumental response	36
6	Advanced interface : use in low spectral resolution	38
6.1	Understanding the configuration and observation files	40
6.2	Evaluating the spectral calibration law and the instrumental response	44
6.3	Variations on computing the calibration parameters	49
6.4	Reducing the spectrum at low resolution	52
6.5	Advanced configurations	55

6.5.1	line_type: absorption	56
6.5.2	line_type: emission	57
6.5.3	line_type: external	57
6.5.4	line_type: lateral	61
6.6	Accounting for atmospheric transmission	66
6.7	Revisiting the calibration technique : the case of a cool star	69
7	The advanced interface : use in high spectral resolution	73
7.1	General points	73
7.2	Evaluating the instrumental response	77
7.3	Routine processing of high-resolution spectra	83
8	Summary	86
8.1	First pitfall : the indispensable flat-field	86
8.2	Second pitfall : the calibration star is not of spectral type A	87
8.3	Fourth pitfall : watch out for the narrowness of the spectrum	89
8.4	Fifth pitfall : watch out for the binning width	92
9	The case of the Alpy 600 spectrograph	95
10	The case of infrared observation	111
11	Appendix 1 : Utility modes	114
11.1	Mode -2	115
11.2	Mode -3	115
11.3	Mode -4	116
11.4	Mode -5	116
12	Appendix 2 : Parameter reference manual	117
12.1	Parameters relating to the observing session	117
12.2	Parameters relating to spectra	117
12.3	Operational parameters	124
13	Appendix 3 : Function reference manual	125
14	Appendix 4 : Reference spectra	129

1 Foreword

This guide describes the reduction of astronomical spectra using the **specINTI** software and its interface **specINTI Editor** (version V4 and later). The software is offered free of charge.

It can be downloaded from the official website:

https://solex.astrosurf.com/specinti1_en.html

The application runs under Windows, macOS, and Linux.

To make the most effective use of this guide, we recommend first becoming familiar with the interface and general operation of the software by consulting the user manual, downloadable in PDF format from the following link:

https://buil.astrosurf.com/specinti4/manuel_specinti_EN.pdf

2 Organisation of this guide

Numerous worked examples illustrate the different parts of this guide. The corresponding data are available for download; we **encourage** readers to reproduce them step by step for effective learning.

Although **specINTI** can handle a very wide variety of spectral data, for the sake of clarity we restrict ourselves here to two broad categories: spectra acquired at **high resolution** ($R > 10\,000$) and those acquired at **low resolution** ($R < 1\,500$).

After a brief reminder of what data reduction sets out to achieve, this guide explains, in order, how to reduce spectra using:

- the so-called “Standard” interface, introduced from version V4 onward and designed for quick, intuitive use;
- the so-called “Advanced” interface, intended for more precise reduction whenever the situation calls for it.

The document concludes with a dedicated discussion of processing data from the **Alpy 600** spectrograph, followed by a set of appendices.

3 Background

Spectral data *reduction* — the term used for this processing work — consists in turning a raw image into usable physical information: the flux of the source (in relative intensity, in our case) as a function of wavelength. This process breaks down into three fundamental steps:

1. extracting a curve representing the intensity of the spectrum along the trace recorded on the image, once instrumental biases have been removed;
2. associating a wavelength with each point of this curve — this is spectral calibration;
3. correcting for instrumental and atmospheric effects in order to recover the true intensity distribution of the source. This is where the correction for the instrumental response to the incoming optical flux is applied.

In other words, the raw two-dimensional image is progressively transformed into a calibrated physical measurement.

3.1 Extracting the spectral profile

Everything begins with correcting the defects intrinsic to the sensor: subtraction of the offset signal and the dark signal, followed by correction of pixel-to-pixel sensitivity variations using a “flat-field” image. This set of corrections, known as **DOF** processing (*Dark, Offset, Flat-field*), is identical to that used in deep-sky imaging.

The spectral signal of the object can then be isolated: the sky background is estimated in the regions on either side of the spectral trace and subtracted.

Next comes extraction proper: the flux is summed column by column along the trace, producing a one-dimensional curve known as the *spectral profile*.

This step amounts to an intelligent form of “compression”: all the information contained in the two-dimensional (2D) image is condensed into a 1D signal without sacrificing spectral information. It nevertheless calls for particular care, especially regarding the geometry of the trace and control of noise.

3.2 Spectral calibration

At this stage, the spectral profile is expressed as intensity per pixel; the pixel axis must now be converted into wavelength, an operation that lies at the heart of all quantitative spectroscopy.

The key lies in establishing a **dispersion law**, that is, a relation linking the position x , in pixels, on the detector to the wavelength λ . In most cases this law is not strictly linear and is modelled by a polynomial, usually of degree 2, which is generally sufficient for a grating spectrograph¹:

$$\lambda(x) = ax^2 + bx + c$$

The coefficients (a, b, c) are determined from reference sources — spectral lamps or stars — some of whose lines have precisely known wavelengths. By identifying these lines in the measured spectrum, the correspondence between pixels and wavelengths is established naturally.

The analogy is a telling one: it amounts to graduating a ruler, assigning each pixel a precise position within the spectral domain. More precisely:

- a degree-2 polynomial is sufficient in the great majority of cases, as noted above;
- line positions are measured with sub-pixel accuracy;
- the coefficients a and b characterise the dispersion of the instrument;
- the term c reflects an overall offset, which may vary slightly from one acquisition to the next owing to instrumental flexure. This is known as the *constant* term of the polynomial.

3.3 Flux calibration

Once the spectral axis has been calibrated, the measured intensity still needs to be corrected — a step that is likewise essential for any quantitative interpretation.

The raw signal recorded by the detector, $S_{\text{raw}}(\lambda)$, is in fact the product of the true signal and two perturbing factors:

1. When the dispersive element of the spectrograph is a *GRISM*, as in the Alpy 600, a degree-3 polynomial may be required to correctly fit the dispersion law. See section 9 of this document.

- the instrumental response $\mathcal{R}(\lambda)$, which encompasses the effects of the optics (mirrors, lenses, grating, ...) and of the detector;
- the atmospheric transmission $T_a(\lambda)$, which varies according to the observing conditions.

This can be written as:

$$S_{\text{raw}}(\lambda) = S_{\text{true}}(\lambda) \mathcal{R}(\lambda) T_a(\lambda)$$

The physical spectrum of the source is then obtained by simple division:

$$S_{\text{true}}(\lambda) = \frac{S_{\text{raw}}(\lambda)}{\mathcal{R}(\lambda) T_a(\lambda)}$$

This correction amounts to “deforming in reverse” the measured signal in order to recover what the source actually emitted. Two points deserve to be emphasised:

- the instrumental response is generally stable; it characterises the instrument and changes very little over time;
- the atmospheric transmission, by contrast, depends closely on the observing conditions: air-mass (that is, angular elevation above the horizon), aerosols, humidity, ...

These three steps — extraction, spectral calibration, and flux calibration — together form the complete reduction chain: at the end of this process, the raw image has been transformed into a curve, rich in information, that remains to be analysed.

3.4 Reduction scenarios

`specINTI` is designed to handle the whole set of reduction steps described above. While the initial operations — DOF processing, geometric corrections, and extraction of the spectral profile — remain broadly similar regardless of the target and the instrument, differences appear as soon as spectral calibration and flux calibration come into play.

Two broad families of situations can be distinguished: high-resolution spectra and low-resolution spectra (see the definition given in section 2). Depending on the case, the calibration operations do not follow exactly the same path; these are referred to as distinct reduction *pipelines* (or scenarios). In practice these approaches can sometimes overlap, but only the most common cases are covered here.

Table 1 summarises the reduction options best suited to each resolution regime.

The number of options may seem daunting at first sight; there is, however, no cause for concern: their underlying logic will quickly become clear through the simple, concrete examples given in what follows.

The essential difference between the two reduction modes, as implemented in `specINTI`, lies in how calibration is handled:

- **At high resolution**, the spectral calibration law is determined within the reduction pipeline itself. The instrumental response, by contrast, is evaluated beforehand, in a separate step. Although this choice might appear to add complexity, it is in fact advantageous: the instrumental response varies little over time and can be reused for a large number of spectra, whether they come from the same session or from different observing nights.
- **At low resolution**, the pipeline requires both the instrumental response and the spectral dispersion law to be supplied. Fortunately, `specINTI` offers largely automated tools for determining these; putting them to use is quick and presents no particular difficulty, as the following sections will show.

The role of the pipeline is then essentially limited to adjusting the constant term c of the dispersion law, so as to compensate for the slight mechanical and thermal deformations that can shift the position of the trace on the detector depending on where the instrument is pointed on the sky, or simply with the passage of time. It is this requirement — finding the term c — that gives rise to the various reduction sub-modes: *none*, *external*, *lateral*, *emission*, and *absorption*.

Table 1 – Table of reduction modes

Function	Mode	Line_type	Calibration source	Configuration settings
Reduction of high-spectral-resolution spectra	0	External	Use of an emission-line lamp before or after the exposure	If a neon lamp is available: auto_calib ; otherwise use the wavelength and line_pos parameters. The spectral calibration law is evaluated automatically. Supply the instrumental-response file (instrumental_response parameter)
		Lateral	Use of an emission-line lamp switched on continuously or intermittently during the exposures	Same configuration as above
Reduction of low-spectral-resolution spectra	1	None	None	Direct use of the wavelength-calibration-law coefficients (calib_coef parameter to be supplied). Supply the instrumental-response file (instrumental_response parameter)
		External	Emission-line lamp before or after the exposure. The object and calibration spectra are acquired separately	The wavelength-calibration-law polynomial must be supplied, together with at least one wavelength and one line position (wavelength and line_pos parameters). These lines are used to adjust the constant term A_0 . Supply the instrumental-response file (instrumental_response parameter)
		Lateral	Emission-line lamp illuminating the instrument during the exposure, or use of airglow or light-pollution lines. The science image and the calibration image coincide	Same configuration as above
		Emission	Natural calibration source: use of emission lines of known wavelength present in the object's spectrum	The wavelength-calibration-law polynomial must be supplied. The emission lines of the observed object are used to adjust the term A_0 of the polynomial. Supply the instrumental-response file (instrumental_response parameter)
		Absorption	Natural calibration source: use of absorption lines of known wavelength present in the object's spectrum	The wavelength-calibration-law polynomial must be supplied. The absorption lines of the observed object are used to adjust the term A_0 of the polynomial. Supply the instrumental-response file (instrumental_response parameter)

4 Standard interface: low-resolution spectroscopy

This section describes the procedure for processing spectra acquired in **low-resolution** mode.

We begin with the “standard” interface of **specINTI Editor** (see Fig. 1). This interface will be familiar to anyone who has previously used astronomical image-processing software, for example for deep-sky imaging. It is also the interface displayed when the software is launched.

The more feature-rich “advanced” interface is introduced later (Section 6). Once the principles of the standard interface have been understood, the advanced interface is not necessarily more difficult to use.

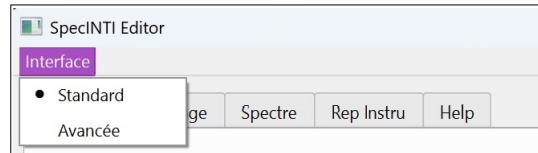


Figure 1 – Selection of the “standard” interface of **specINTI Editor**.

The example used throughout this section consists of a sequence of spectral images of the star HD 158261 ($V = 5.9$). This is a hot A0V-type star whose hydrogen Balmer absorption lines are particularly prominent. Like other stars of the same spectral class, it provides an excellent reference for simultaneously determining the wavelength calibration and the instrumental response, as will be shown below.

The observations were carried out with a 200-mm $f/5$ Newtonian telescope. The spectrograph is a **TinyStar’Ex**—a compact version of the **Star’Ex**—equipped in its standard configuration with a $300 \text{ lines mm}^{-1}$ diffraction grating. This grating provides complete coverage of the visible and near-infrared spectral range in a single exposure, with a spectral resolving power of approximately $R = 700$ in the green (20- μm slit, GEN2 version of the **Star’Ex** system). The camera is a **Player One Uranus-M Pro**, fitted with the excellent Sony IMX585 sensor.

The complete observing setup is shown in Fig. 2.



Figure 2 – The **TinyStar’Ex** spectrograph mounted at the focus of a 200-mm Newtonian telescope.

All the images required for this tutorial can be downloaded from:

https://buil.astrosurf.com/specinti4/demo/antibes_tse_20260521.zip

The compressed archive contains:

- eight raw spectral images of HD 158261, each with an exposure time of 18 s, exactly as acquired. The FITS files are named HD158261-1, HD158261-2, etc.;
- four spectra of the planetary nebula Messier 57;
- three bias frames, named o-1, o-2, and o-3;
- five dark frames, each acquired with an exposure time of 1800 s at a detector temperature of -15°C , identical to that used during the observations (n1800_15-1, n1800_15-2, ...).

These data were acquired using the CCDCiel software (<https://www.ap-i.net/ccdciel/en/start>), which provides several features particularly useful for astronomical spectroscopy and which we highly recommend.

A few additional remarks regarding these observations are in order.

Image format and cropping. The images were recorded at the full resolution of the IMX585 detector using 1×1 binning. Consequently, the files are relatively large, although the spectral trace occupies only a small fraction of the detector area. In practice, it is often preferable to crop the images during acquisition so that only the spectral region is recorded, thereby reducing the data volume. Full-frame images nevertheless offer one important advantage: the same DOF corrections remain valid even if the spectrum shifts slightly from one observing session to another. This is generally no longer true once cropped images are used, particularly if the size or position of the cropped region changes. The choice ultimately depends on the observer’s preferences.

No flat-field frames. No flat-field images are provided, as they are unnecessary in this example. The method used to determine the instrumental response does not require them, and the excellent cosmetic quality of the IMX585 sensor—which exhibits neither significant low-frequency nor high-frequency fixed-pattern artefacts—makes flat-field correction largely superfluous. This considerably simplifies the acquisition procedure.

Reduced data set. The data supplied here represent only a subset of the original observations. The complete sequence consisted of 14 stellar spectra, 15 bias frames, and 12 dark frames (the latter acquired in daylight inside a refrigerator). Such a data volume maximises the signal-to-noise ratio and improves the robustness of the instrumental response curve. The present data set has been intentionally reduced to keep the download size reasonable.

File naming. One final recommendation deserves particular emphasis: adopt simple and consistent file names. For astronomical targets, the root name should correspond to the object designation listed in SIMBAD. This is essential for efficient use of **specINTI**. File names such as `s_h185261_210521_beautiful_star-A0-1.fits` should be avoided entirely. Likewise, directory paths should remain short, and spaces should never be used in either file or directory names. The underscore character (`_`) should instead be used as a separator.

4.1 Low-resolution calibration

The first step is to determine the coefficients of the wavelength-calibration polynomial together with the instrumental response. This is the instrument **calibration** stage. Version 4 of **specINTI** considerably simplifies this process, which is often regarded as one of the most demanding aspects of spectroscopy. Once determined, these calibration data can be reused for all objects observed during the same session, and often during subsequent observing sessions.

Launch **specINTI**. In the “Processing” tab, under the “Parameters” section, click the “LR” (*Low Resolution*) button, then click the “< Calibration” button located at the bottom of the window.

This opens the calibration interface, from which the wavelength calibration and instrumental response of your spectra will be computed.

Create a working directory. In this example we shall use

d:\antibes_tse_20260521

The directory name indicates that the observations were carried out from Antibes, using a TSE (*TinyStar'Ex*) spectrograph, on 21 May 2026.

Next, enter the object name in the “Object name” field, here HD158261. Clicking “Autofill” automatically fills in both the raw-image prefix (HD158261-) and the number of images (8). This illustrates the importance of adopting a rigorous file-naming convention during data acquisition, a task made straightforward by software such as *Prism* or *CCDCiel*. The resulting interface is shown in Fig. 3.

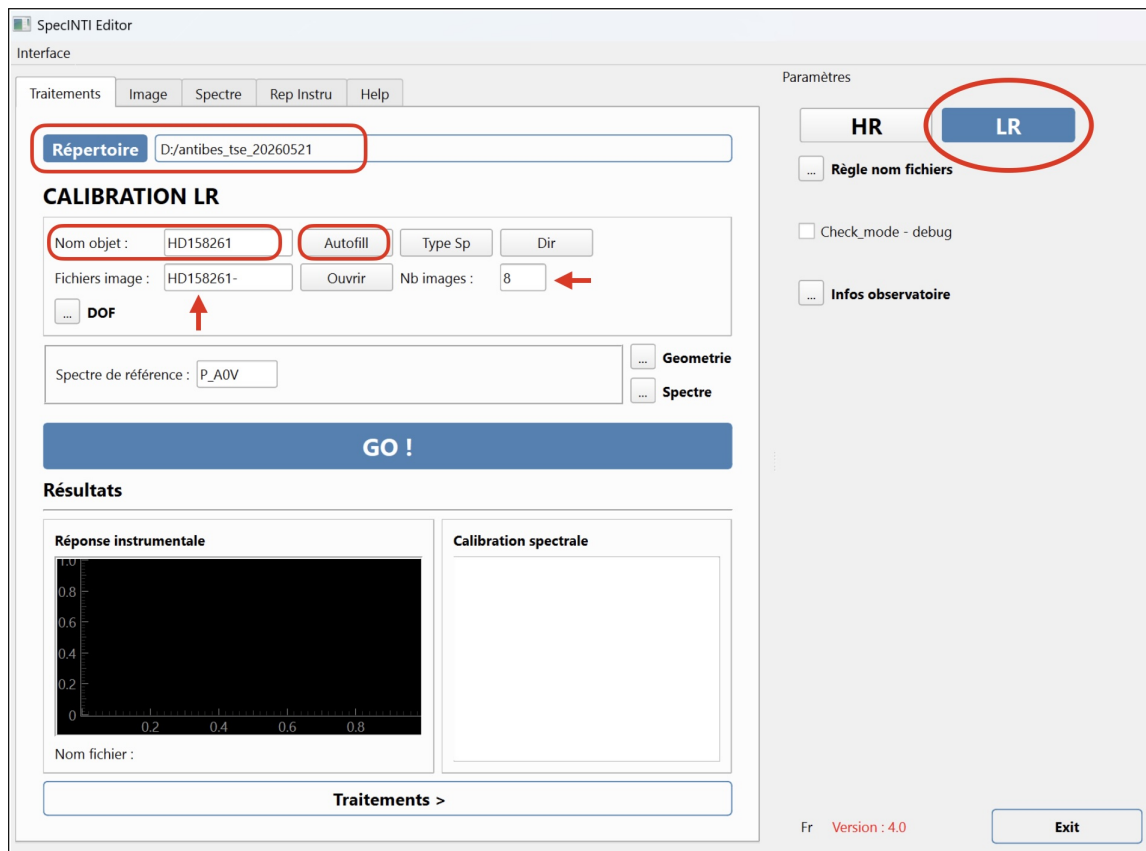


Figure 3 – Calibration interface for low-resolution spectra.

From this interface, the instrumental response will be derived automatically from the observed spectrum of the target, in this case HD 158261. This star serves as the “reference” because its intrinsic spectrum is well known, being that of a typical A0V star.

The software must nevertheless be informed where to find the intrinsic spectrum, $S_{\text{true}}(\lambda)$, unaffected by either the instrument or the Earth’s atmosphere (see Section 3.3). For low-resolution (“LR”) spectroscopy, *specINTI* relies on the Pickles spectral library², which is installed automatically in the “_calib” subdirectory of the *specINTI* installation.

The software can automatically identify the closest matching spectral type in the Pickles library. Click the “Type Sp” button: the selected spectral type together with the corresponding Pickles

2. Pickles, A. J. (1998). A Stellar Spectral Flux Library: 1150–25000 Å. *Publications of the Astronomical Society of the Pacific* (PASP), 110(749), 863–878.

reference file is displayed immediately (Fig. 4).

Figure 4 – Selection of the reference spectral profile.

The available DOF calibration frames must also be specified. Click the “DOF” button and complete the various fields (bias, dark and flat), as shown in Fig. 5. If no calibration frames are available (although this is generally discouraged, except in some cases for flat-field frames), simply enter “None” together with a negative image count.

Figure 5 – Specification of the DOF calibration frames.

In this example, note how the DOF files are specified: only the generic file name (or “root name”) is entered, including the trailing hyphen if this is used as the filename separator. Clicking the “Nb” button allows the number of available images to be verified. No flat-field frames are used in the present example, hence the corresponding field is set to “None”. An alternative approach is described in the boxed section entitled “Simplified use of DOF frames”.

Simplified use of DOF frames

During the first data reduction (or during instrument calibration), the software automatically stores a processed version of the elementary DOF frames in the working directory. These constitute a new category of compact calibration files, eliminating the need to manage long image sequences. More specifically:

- The individual bias frames (**o-1**, **o-2**, **o-3**, ...) are combined into a single file named **_offset**, containing the median of all the individual bias frames. Note that the leading underscore (**_**) is mandatory.
- The individual dark frames (**n-1**, **n-2**, ...) are similarly combined into a single median dark frame named **_dark**. The bias signal is removed beforehand, so that the mean level of the resulting **_dark** image is typically close to zero.
- The flat-field frames (**f-1**, **f-2**, **f-3**, ...) are likewise combined into a single file. Both the bias and dark signals are subtracted beforehand, with the appropriate scaling for the different exposure times of the dark and flat-field frames. Thus, whereas **f-1** is a raw flat-field image, **_flat** represents the true flat-field signal, i.e. the relative gain distribution of the system.

These combined DOF frames can subsequently be reused for processing additional spectra, including data acquired during different observing sessions (see Fig. 6).

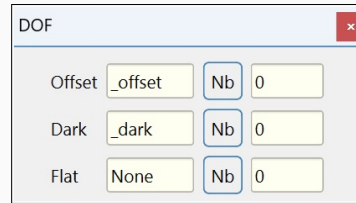


Figure 6 – Entering DOF information in simplified mode. Setting the image count to 0 does not indicate that the file is absent; in that case enter **-1** or **None**. A value of 0 simply indicates that the file name is not indexed.

The advantages are twofold: processing is faster, and only a small number of combined DOF files need to be copied from one working directory to another, rather than transferring large collections of individual calibration frames.

specINTI also uses the Balmer lines of the reference star to derive the wavelength solution. These lines are detected automatically, allowing the dispersion polynomial to be computed in only a few seconds. Before doing so, however, the software must be informed of the spectral region within which the Balmer lines are to be searched. Their wavelengths are indicated in Fig. 140.

Click the “Geometry” button. A dialog box opens (Fig. 7), allowing the geometrical parameters of the spectrum to be adjusted. Feel free to zoom in with the mouse and adjust the display contrast on the first image of the sequence. Refer to the user manual for further details on image visualisation.

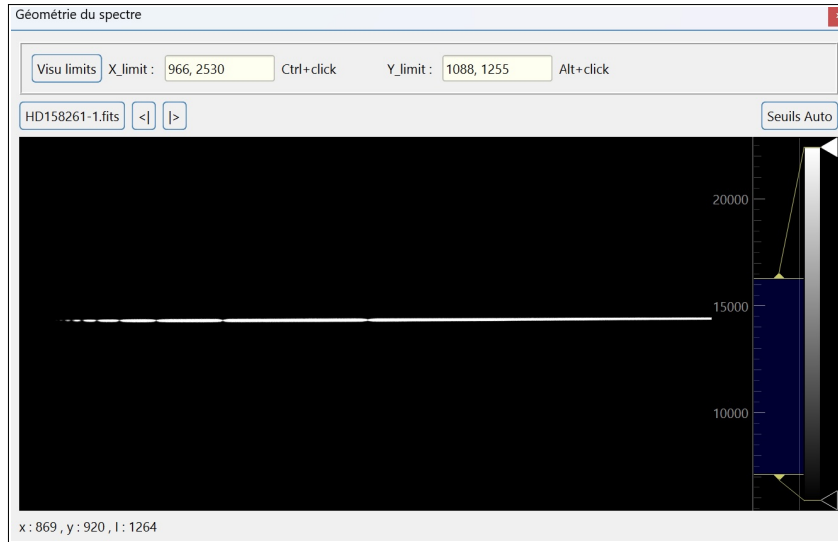


Figure 7 – Dialog box for adjusting the geometrical parameters of the spectrum.

Define two limits along the horizontal (X) axis, within which the software will search for the Balmer lines. To do so, perform two “Ctrl+click” operations with the mouse: one on the blue side of the spectrum and the other on the red side. Try to include as many Balmer lines as possible within this interval, while ensuring that they remain well defined and exhibit good contrast. Figure 8 illustrates the placement of the blue-side limit (a blue vertical line appears). The click should be made on the spectral continuum, away from any absorption line.

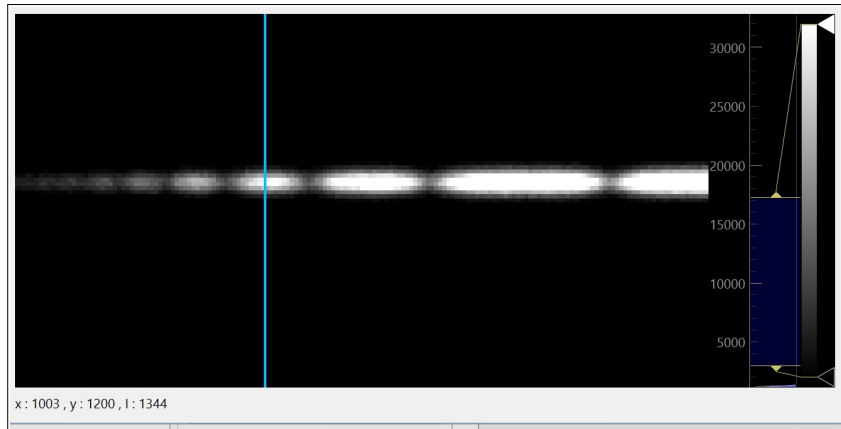


Figure 8 – Blue-side limit (placed on the spectral continuum). We have deliberately not extended the interval to the extreme blue end of the spectrum, preferring the region where the Balmer lines remain well contrasted.

Repeat the procedure on the red side, making sure that the $H\alpha$ line is included within the selected interval (it is worth becoming familiar with its appearance in the spectrum). A red vertical line is then displayed (Fig. 9).

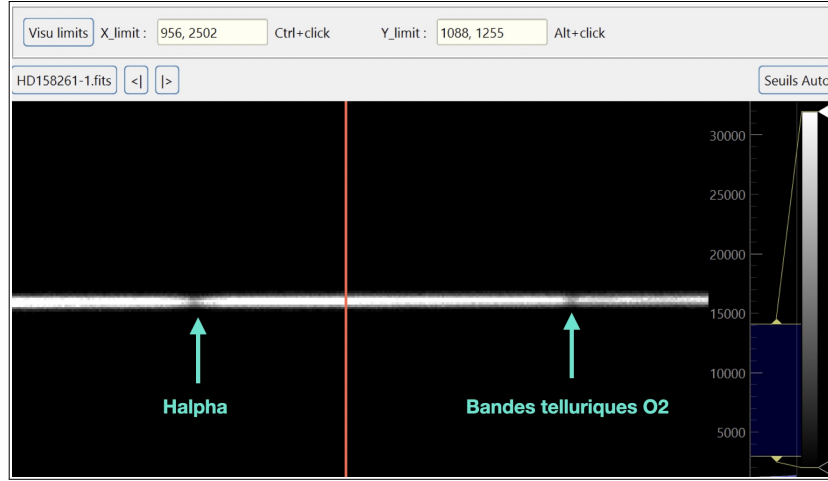


Figure 9 – Red-side limit. The H α line is included within the selected interval.

The “X limit” field at the top of the dialog box displays the coordinates of the two selected limits. When using the advanced interface, these values can instead be entered directly through the `xlimit` parameter in the instrument configuration file. In the present example:

```
xlimit: [956, 2502]
```

While this dialog box is open, it is also advisable to indicate the approximate vertical position of the spectrum within the images. Although `specINTI` usually identifies the trace of a bright star automatically, specifying this region is useful in two situations: when the stellar spectrum may be confused with the characteristic photometric region produced by the GEN2 Star’Ex slit (an example is given later), or when the faint spectrum under study lies close to a brighter one (see Section 9).

The procedure is straightforward. Perform a first “Alt+click” below the spectral trace, followed by a second above it. Two green horizontal lines then appear (Fig. 10). You have now defined the image regions (“Y limit”) that are excluded from the automatic search for the spectral trace. These limits correspond to the `exclude_zone` parameter in the configuration file.

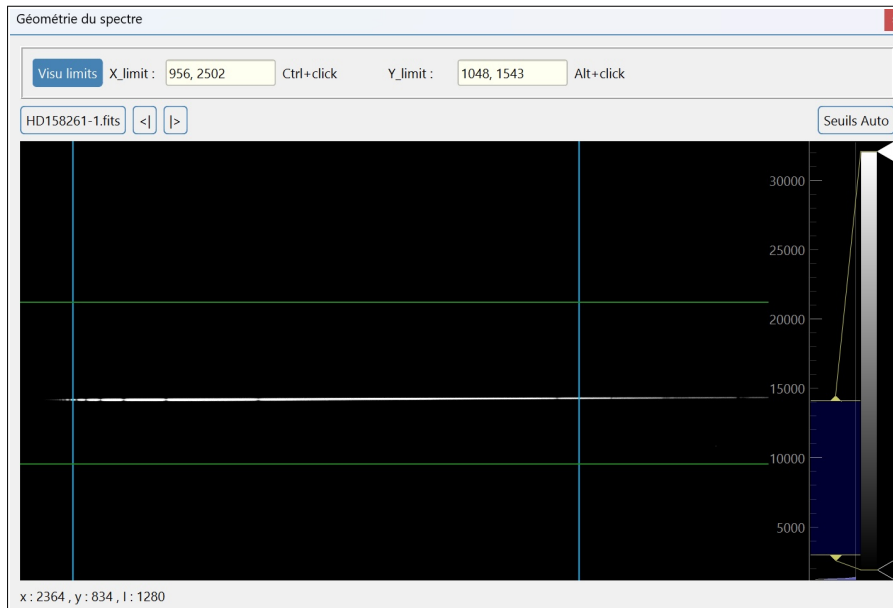


Figure 10 – Vertical limits defining the search region for the spectral trace.

The “Geometry” dialog box can now be closed.

Next, click the “Spectrum” button. A new dialog box opens, which can be closed immediately. For the time being, simply retain the default settings (Fig. 11). At this stage, it is sufficient to note that the “Gaussian sigma” parameter controls the amount of smoothing applied to the computed instrumental response curve.

Figure 11 – Dialog box for adjusting the spectrum extraction parameters.

Finally, click the large “Go!” button to start the instrument calibration.

Information is displayed in the output **console** on the right-hand side of the window. The various processing steps are listed together with several intermediate results, such as the vertical (Y) position determined for each spectrum in the sequence.

At the end of the process, the instrumental response is displayed as a curve (Fig. 12) and is also saved in the working directory under the name `_rep_20260521_hd158261.fits`. The file name records both the observation date and the reference star used for the calibration. The coefficients of the wavelength-calibration polynomial are displayed to the right of the response curve (see the boxed note entitled “Understanding the wavelength-calibration polynomial”).

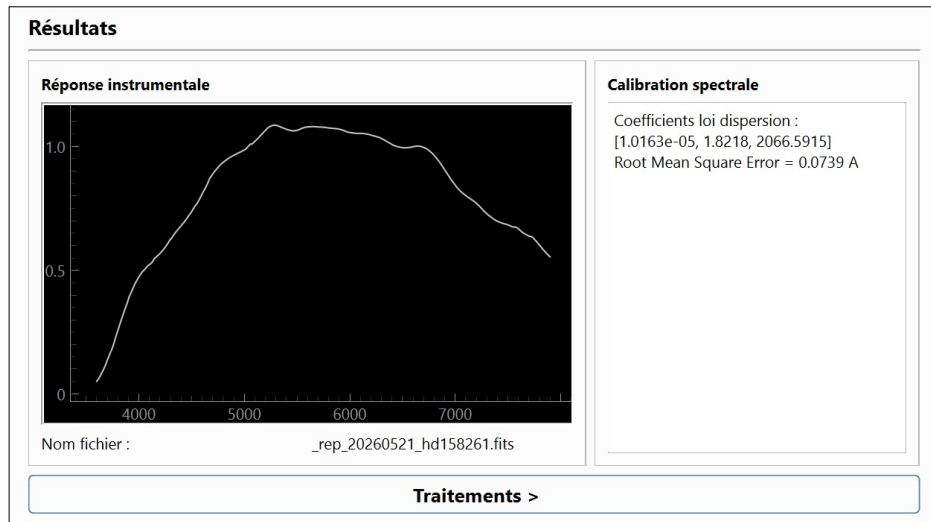


Figure 12 – Result of the calibration procedure.

Operations that are often regarded as tedious in conventional spectroscopic workflows have thus been completed with relatively little effort.

Understanding the wavelength-calibration polynomial

The first coefficient of the polynomial (reading from left to right) is the **constant term**. Its value (c) represents a simple wavelength offset. In the present example, the leftmost pixel corresponds to a wavelength of 2066 Å (a separate polynomial is determined for each image). The second coefficient is the dispersion term, which provides the first-order approximation to the wavelength scale. Here, the mean spectral dispersion in the raw images is found to be $1.82 \text{ Å pixel}^{-1}$. The final coefficient describes the departure from linear dispersion. For Star'Ex and TinyStar'Ex spectrographs, this term always remains small.

Note. The quantity referred to here as the “response” is not the true instrumental response, i.e. the intrinsic throughput of the instrument, since atmospheric transmission has neither been measured nor removed. This point will be revisited when discussing the advanced processing mode.

4.2 Processing at low resolution

We can now move on to the actual reduction of our reference star, and indeed of any other object, using the instrument’s own calibration parameters, which we computed in the previous section.

Click the “Processing >” button, found on the bar at the bottom of the interface. This takes you to a panel that already contains familiar elements, such as the name of the working directory and the reference to the star being processed, but also the instrumental response and the dispersion polynomial computed earlier, both now taken into account (see Figure 13).

Figure 13 – The low-resolution processing configuration.

What remains to be done before the spectrum can be reduced?

First, make sure the DOF frames are correctly specified (“DOF” button).

Go to the “Geometry” dialogue box. You will not necessarily find the same values as those set during calibration. This is not unusual: they may change depending on, for instance, the nature of the object under study. Moreover, the “X limit” parameter has a different meaning here: it is no longer a matter of isolating lines to determine the dispersion law, but simply of defining

two points on the spectral trace, used to automatically find the correction factors for geometric defects, such as an inclination of the trace, which we call `tilt`. Here you are free to choose points anywhere in the strong part of the trace, provided they are sufficiently far apart.

The “Y limit” parameter, for its part, still delimits the vertical search zone for the trace.

In the end, you should obtain something similar to what is shown in Figure 14. You will probably not need to revisit this often if you always observe in the same way (which is recommended!), since in that case you can work "blind" from a set of parameters established once and for all, so to speak (click the “Visu limits” button if you are unsure about the positioning of the zones).

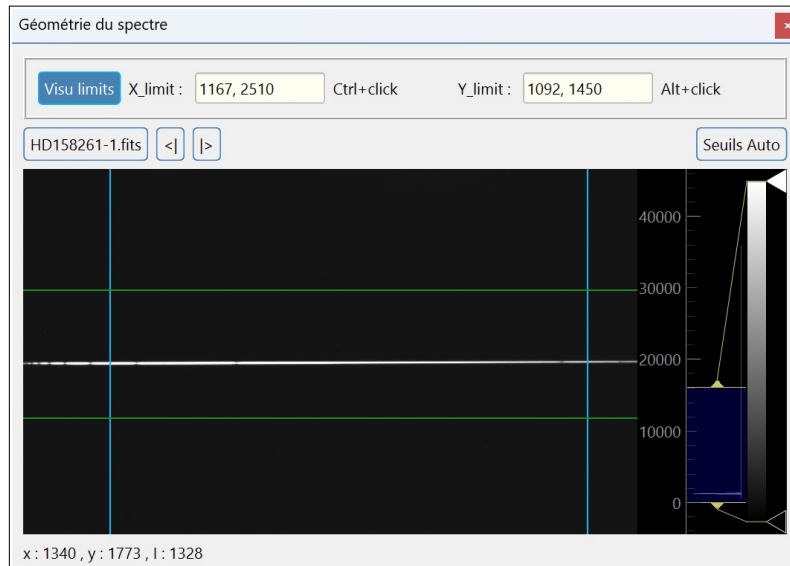


Figure 14 – The geometric parameters adopted for the reduction.

Let us continue by entering the parameters that relate more specifically to the spectrum. To do this, click the “Spectrum” button. A dialogue box then opens (Figure 15).

Figure 15 – The parameters affecting the spectral profile.

You can define a wavelength interval over which the mean intensity is normalised to unity (in the example, between 6620 Å and 6640 Å). You can also restrict the wavelength range of the final profile — in the example, between 3600 Å and 7900 Å.

The “Filtering” box gives detailed information on filtering the spectrum to reduce noise.

Filtering

The “Kernel” parameter defines an $x \times x$ pixel matrix (or kernel) that is moved across the image, within which a median filter is applied (the central value of a distribution of intensities). Since the characteristic kernel size is 3×3 , the value entered is 3 (the length of one side). It is very rare to go as far as a side length of 5 pixels (note that even values are not allowed; an error message is returned during processing if one is entered).

This type of (non-linear) filter is effective at removing certain defects (such as cosmic-ray hits) or the impulsive (non-Gaussian) noise that affects CMOS sensors. It can, however, also destroy genuine information when used in its traditional form. This is why a negative value is given in the example (-3). This distinction is very important: a median filter is still applied, but here with a weighting that ensures noise is correctly removed while preserving the integrity of the image. So do not forget the “ $-$ ” sign.

You may of course choose not to apply any median filtering at all, for example if the object is bright enough, or if the spectrum is severely undersampled — which can itself be a source of artefacts, so be wary of it (a spectrum is undersampled if you are using a very short-focal-length imaging instrument, or if you acquire your images with 2×2 binning, which serves no purpose: let specINTI handle the noise for you, which will be far more effective). To disable median filtering, enter the value 0.

The “sigma gauss” parameter sets the strength of a Gaussian filter applied to the spectral image. A value of 0 or 0.1 corresponds to no filtering, or almost none. A value of 0.7 begins to produce a noticeable effect. A value of 2 corresponds to very heavy Gaussian-noise filtering, and so on. Note that the result naturally affects the spectral resolution.

When reducing the spectrum of a bright object, where the aim is to reach the spectrograph’s maximum resolving power, Gaussian filtering is not called for, hence the use of a zero or very small value (0.1).

Use the default values shown here, then click the large “Go!” button.

The pipeline then runs on the 8 raw spectra and, once processing is complete, the reduced spectrum is displayed on screen in the “Spectrum” tab of the **specINTI Editor** interface; see Figure 16.

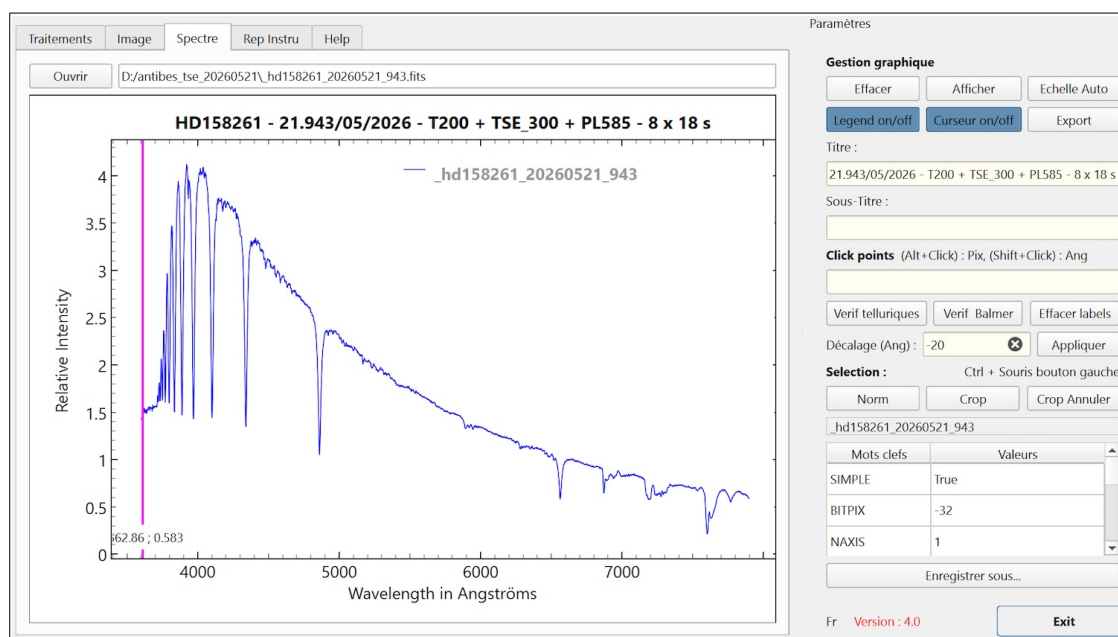


Figure 16 – Display of the reduced spectrum.

The spectral profile is automatically saved in the working folder, as a FITS file whose name takes the form `_object_name_date`. It is corrected for the instrumental response and calibrated in wavelength. This file can, for example, be sent to professional astronomers for analysis; it is also compatible with the BeSS database of Be stars.

Other files are produced as well. One of them, whose name ends in “2D”, is the sum of your 8 spectral images, precisely aligned, with the sky background subtracted and the noise filtered according to your settings (Figure 17).

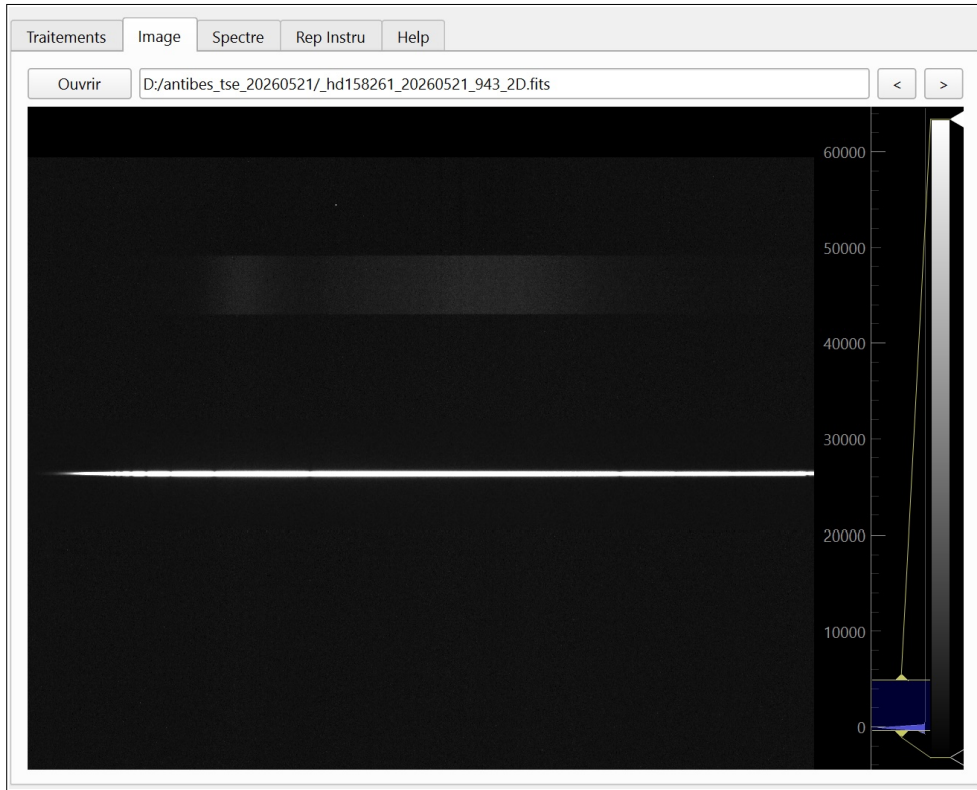


Figure 17 – Display of the processed 2D image, from the “Image” tab.

Many options are available for manipulating the data (zoom, contrast adjustment, editing of titles, and so on). See the user manual for further details.

Let us return to the spectral profile. It amounts to a form of self-calibration, since we have reduced the spectrum of the star HD 158261, which is also the very star used to extract the instrument’s calibration data.

Here is a useful check to confirm that the reduced spectrum genuinely resembles the Pickles reference spectrum used: look in the working directory for the file whose name ends in “_op2.fits” — this is none other than our Pickles spectrum, renamed by the software and copied into this working folder. Display it from the “Profile” tab. It overlays our computed spectrum satisfactorily, as can be seen in Figure 18 — which is only to be expected if everything has gone well.

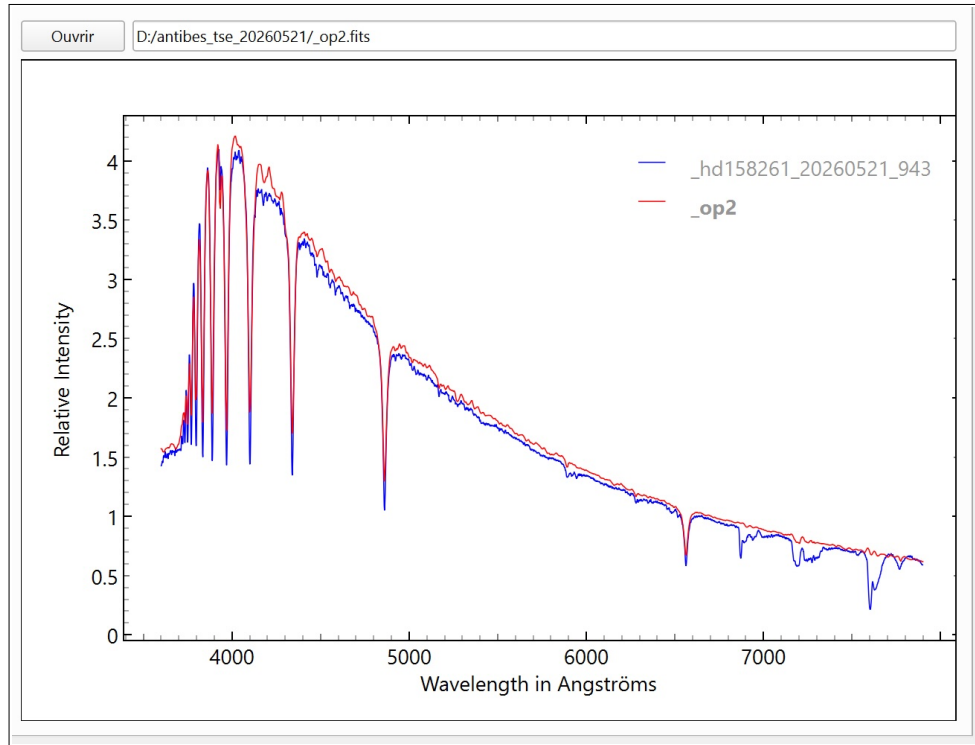


Figure 18 – In blue, the observed spectrum; in red, the Pickles spectrum.

Any other object can be reduced in exactly the same way: simply enter its name, then click “Autofill” and finally “Go!”.

Let us, however, stay with the star HD 158261 in order to explore an important feature of the software. Here is an observation worth making: there is no need to recompute the dispersion law for every object; indeed, the one you are currently using may well date back several sessions. However, while the terms (a) and (b) of the dispersion polynomial can be treated as constants for a given instrument, the same is not true of the constant term (c). Its value is in fact sensitive to shifts of the spectrum along the spectral axis, caused by instrumental deformations of mechanical or thermal origin.

If this risk applies to the constant term — which depends on your equipment and how you use it — you need to update its value. There is, however, no need to carry out a full instrument calibration to do so: **specINTI** offers several ways of performing this operation more lightly.

Suppose you are faced with a new stellar object showing well-marked absorption lines, as is the case for the star HD 158261 (though it could equally well be an object of an entirely different spectral type). In the LR processing interface, open the “Shift method” drop-down list and select the “Object line in absorption” option, then click the “Realignment line” button.

The image of the star’s spectrum is then displayed (Figure 19). We are going to use the observed position of the $H\alpha$ line to update the constant term (c) — which, as we can see, is not quite so constant after all. Shift+click on the position of the $H\alpha$ line (do not hesitate to zoom in for greater accuracy, bearing in mind that you have a tolerance of a good ten pixels or so). In the example, the horizontal coordinate of the $H\alpha$ line is $x = 2435$.

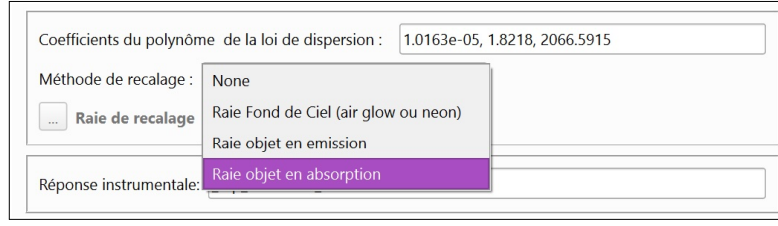


Figure 19 – The tool for realigning the spectrum in wavelength.

Next, enter the wavelength of this line. To help you, a predefined list of notable lines is available in the top-right corner of the dialogue box; select the value 6562.81 Å. Then click the “Add” button. The pair (x, λ) is then entered into a list. You can add several reference lines in this way to further improve the accuracy of the wavelength realignment.

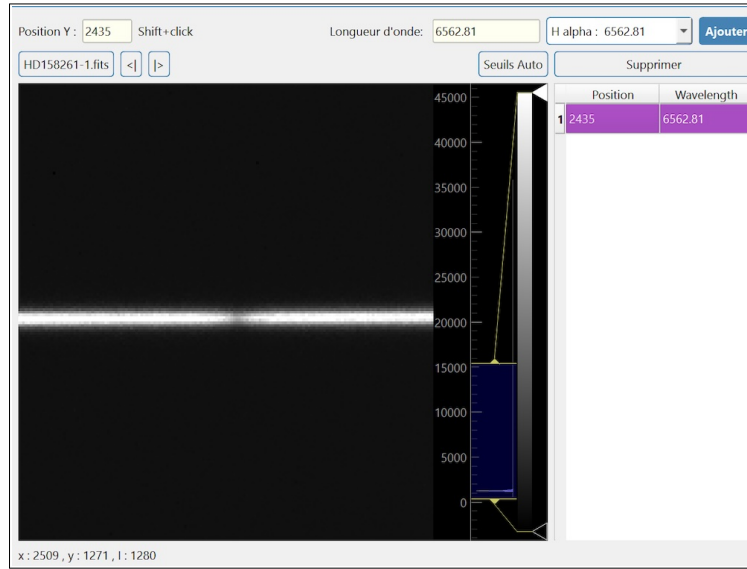


Figure 20 – The position/wavelength pair has been formed. It concerns the H α line, visible at the centre in a highly magnified view of the image.

Close the dialogue box, then run the reduction again. You will probably not notice much difference, since realigning the spectrum was not actually necessary in this particular case: as already noted, this is precisely the star that was used to establish the polynomial in the first place. This will not, however, always be the case (we had selected the “None” option during the first computation).

4.3 Processing the spectrum of an extended object

In the downloaded folder corresponding to the data for the example just processed, you will find four raw images of the spectrum of the famous planetary nebula Messier 57 (the Ring Nebula in Lyra). Since the instrumentation is identical to that used for the star HD 158261, we can reuse the same calibration for M57.

The exposure time of each image is 900 seconds, giving a total of one hour.

Begin by telling `specINTI` that you wish to process an extended object (Figure 21).



Figure 21 – Preparing to process the Messier 57 nebula.

A new “Binning zones” button then appears in the interface. In the dialogue box that opens when you click this button, press “Show zones” (Figure 22).

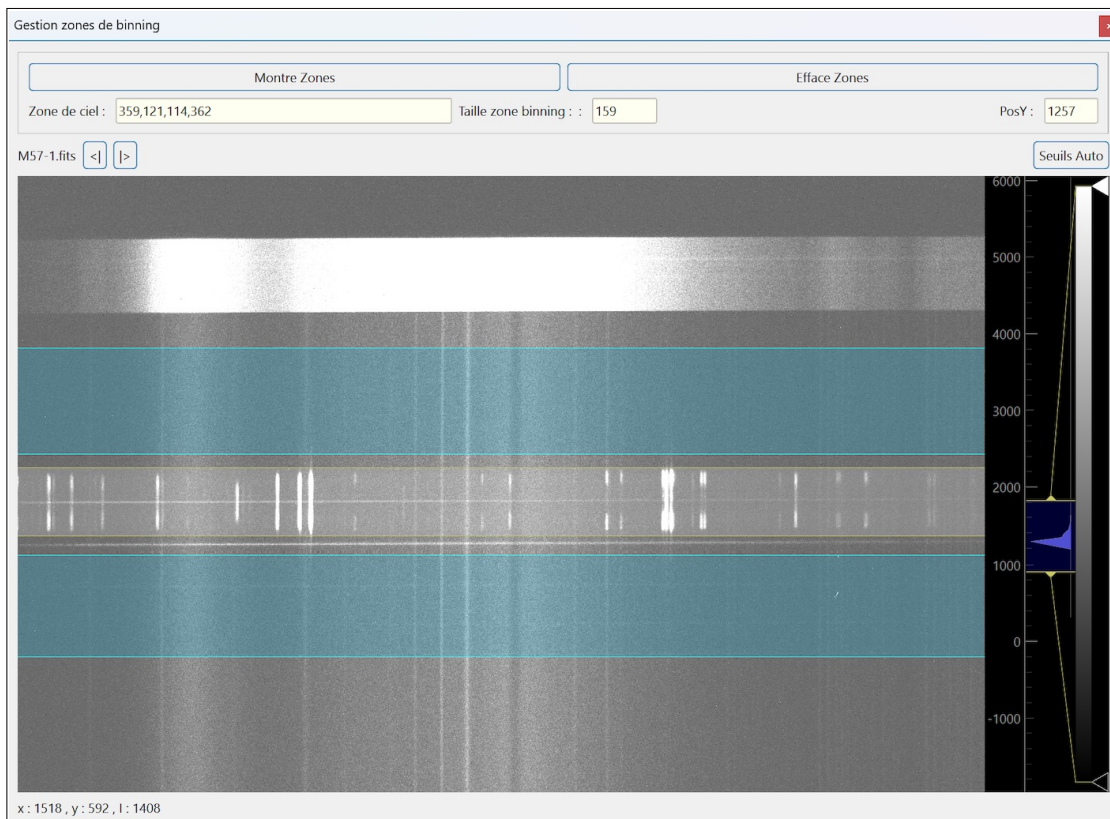


Figure 22 – Adjusting the binning zones.

The grey zone corresponds to the actual binning region, which should be centred on the object under study. You can move it and resize it freely; centre it on the spectral image of the Messier 57 nebula. Fine adjustment is possible: in the screenshot, note that we have carefully excluded the star adjoining the nebula near the bottom of the image.

Also position the sky-background calculation zones on either side of the nebula (they appear in light blue). As shown in the screenshot, take care not to include the photometric zone of the GEN2 slit fitted to the **Star'Ex** or **TinyStar'Ex**.

Finally, it is important to check the vertical centring of the binning zones (object and sky) by checking the value of the **PosY** parameter. This is updated automatically whenever you move the main binning window, but it can be adjusted manually if necessary.

Then close the window. You could at this point click the “Go!” button, but let us first carry out one further step. The nebula was in fact observed in a different part of the sky from the reference star, or at a different time (sometimes several nights apart): the constant term (c) of the dispersion polynomial may therefore no longer be up to date. It is advisable to update it

before launching the reduction.

Several options are available. The first is to use an emission line from the nebula itself: select the “Object line in emission” option as the realignment method, then click the “Realignment line” button. In the dialogue box that opens (Figure 23), locate the H β line in the nebula’s spectrum. This line, being well isolated, is particularly well suited to this purpose; its wavelength is $\lambda = 4861.34 \text{ \AA}$. The software will use it to realign the dispersion polynomial independently on each of the four images. Remember to click the “Add” button to confirm your choice of line (you can select more than one).

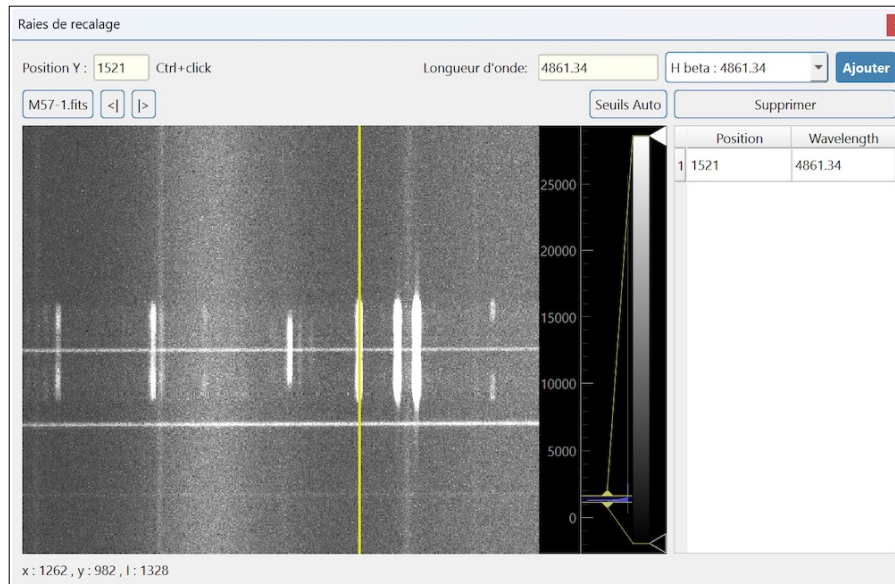


Figure 23 – Realignment on an emission line of the nebula.

The second option is to use a line present in the sky-background signal, whether due to light pollution (particularly marked in the example treated here) or of natural origin (airglow). In that case, select “Sky background line” as the realignment method, and choose as the reference line the airglow [O I] line at $\lambda = 5577.35 \text{ \AA}$.

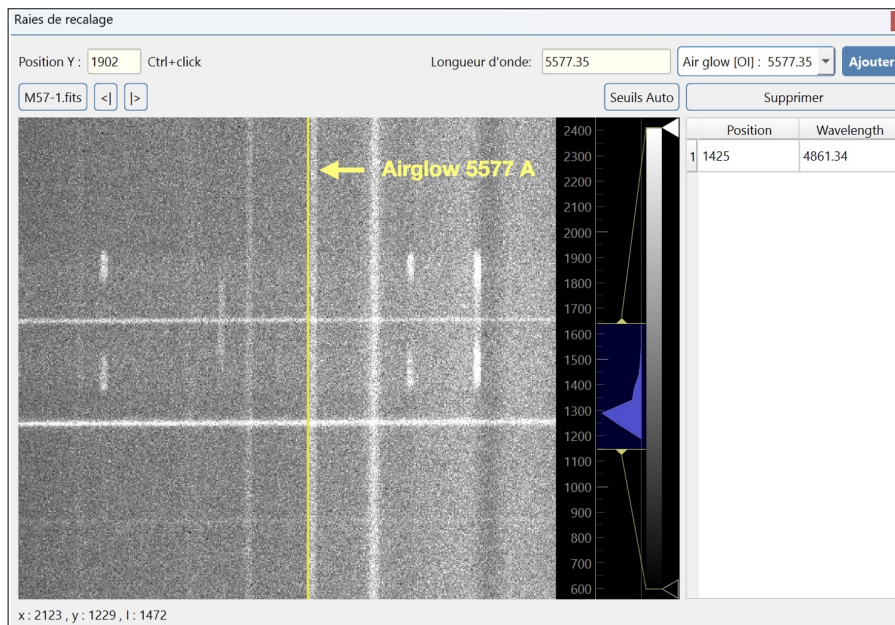


Figure 24 – Realignment on a sky-background emission line.

You can now launch the reduction: the spectra of the nebula will be correctly calibrated in wavelength (Figure 25).

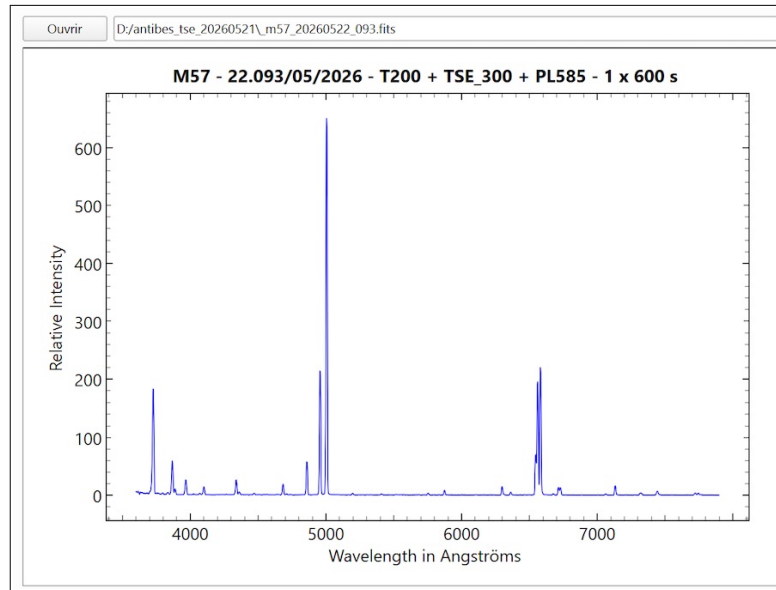


Figure 25 – The spectral profile of the Messier 57 nebula.

Everything appears to have gone well: a fine planetary-nebula spectrum has indeed been obtained. This is true to first order, the spectrograph used being **well aligned**; but if perfection is the goal, a few checks on the geometry of the spectrum in the 2D images are called for.

Once your spectrum has been reduced, take the trouble to inspect the 2D image produced by **specINTI**, alongside the spectral profile. This image is, in principle, corrected for the geometric imperfections that often affect 2D images (depending on how the spectrograph is aligned). It is also the sum of all the individual images in the sequence, with the DOF corrections and noise reduction applied. In a sense, this document is the equivalent of the images one aims to obtain in deep-sky imaging. This is a good opportunity to note that traditional imaging and spectroscopy are, in the end, not such distant disciplines: what one learns in one readily applies to the other.

Let us examine this famous 2D image from the “Image” tab of **specINTI Editor**, with high contrast and a strong zoom, in order to bring out any defects (Figure 26).

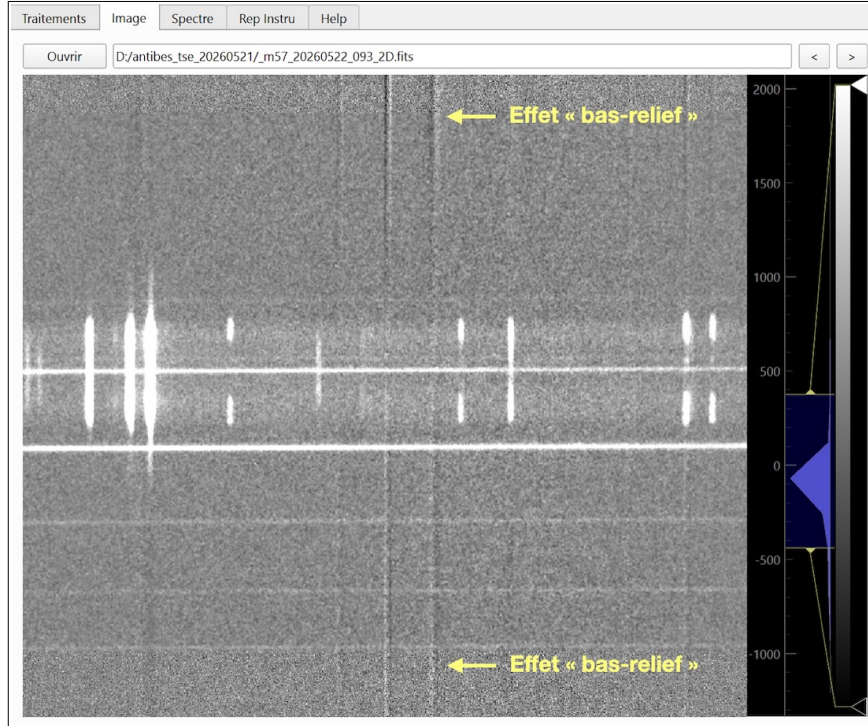


Figure 26 – High-contrast display of a detail from the processed image of the Messier 57 spectrum.

Normally, the sky background (here heavily light-polluted) is subtracted. But on closer inspection, one can see, at the top and bottom of the image, residuals of the strongest sky-background lines, appearing as a gradient (or “relief” effect) that reverses direction between the top and the bottom. The cause is simple: the lines are slightly tilted with respect to the vertical. This is the defect known as **slant**, quantified as the angular deviation from the vertical. Here it amounts to only $0.15''$, but that is enough to make it visible. In certain situations, **specINTI** can automatically determine this angular value and correct for it. The ability to correct this defect is provided from the standard interface of **specINTI Editor**; finding the correct numerical value, however, has to be done manually, by iterating between the application and inspection of the 2D image, until the “relief” effect has completely disappeared.

Similarly, one can see that the trace of the star visible at the bottom of the nebula is tilted by -0.18° with respect to the horizontal. This is the **tilt** defect. Here again the value is small, but you may wish to correct it in pursuit of the art of spectroscopy. Note that the tilt angle was computed when the calibration parameters were determined, and is displayed in the console; you can also estimate it by iteration, by measuring the Y coordinate of the trace at its two ends, which should ultimately be identical. In this way, the binning region from which the spectral profile is extracted will always correspond to the same physical parts of the spectrum, whatever the wavelength, which is a mark of rigour.

Figure 27 shows how to apply these corrections from the “Geometry” button and the window that opens. Then run the reduction again with these new parameters.

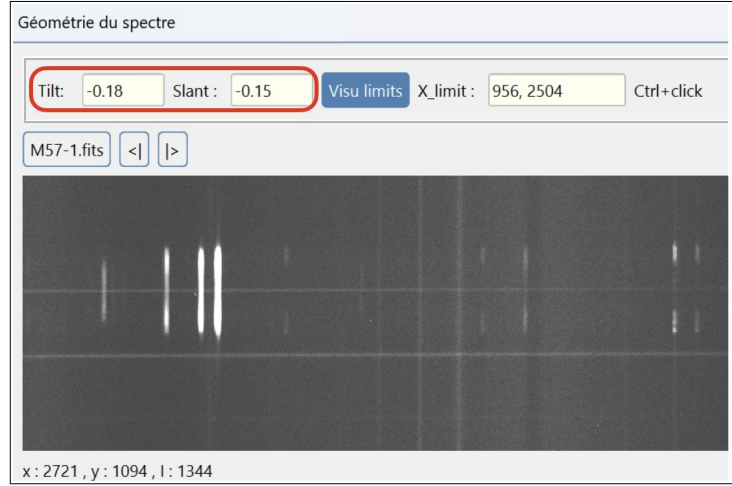


Figure 27 – Entering the angular geometric-correction parameters (**tilt** and **slant**). Pay attention to the sign of the numerical values. As long as you do not alter your spectrograph, you will not need to re-evaluate them.

The result can be seen in Figure 28: the sky-background line residuals have disappeared and the spectrum is perfectly horizontal. Note: it always remains desirable to properly orient the lines and the spectral trace through the instrument’s mechanical and optical alignment, but it is reassuring to know that residual imperfections can be corrected *a posteriori*.

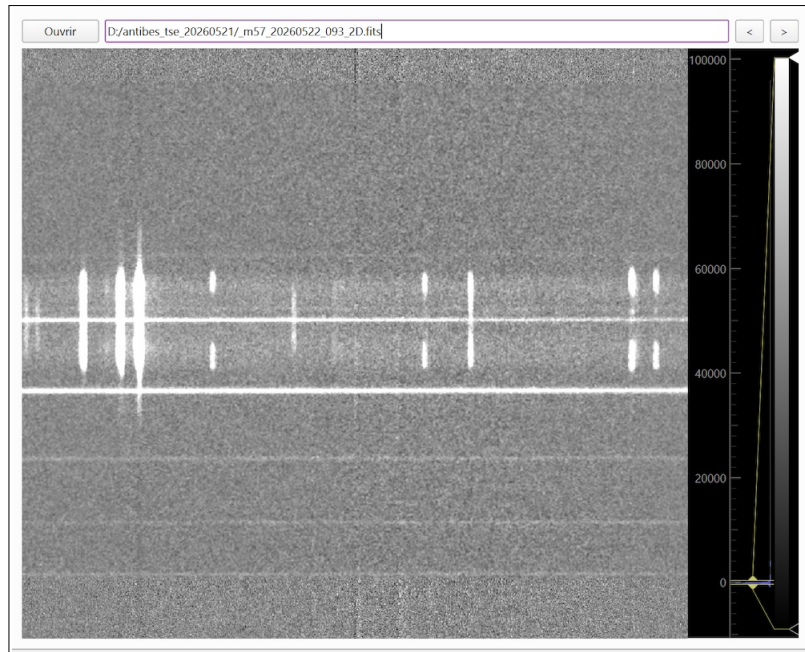


Figure 28 – The processed 2D image after geometric correction. The sky background is perfectly subtracted, which will make it possible to detect the faintest extensions of the object.

To conclude this part devoted to reducing low-resolution spectra from the standard interface of **specINTI**, let us stress the importance of correctly filling in the FITS header of your output files. The software takes care of most of this information itself, but you must also supply the data specific to your identity as an observer and to your instrumentation, via the “Observatory info” button, as shown in Figure 29.

The screenshot shows a window titled 'Infos observatoire' with a sub-window 'Informations Entête FITS'. The sub-window contains several input fields with the following values:

Field	Value
Observer :	cbuil
Instrument :	T200 + TSE_300 + PL585
Site :	Antibes Saint-Jean
Latitude :	43.5969
Longitude :	7.1928
Altitude :	50

Figure 29 – Your observatory’s parameters, entered into the headers of the files produced.

5 Standard interface: use in high spectral resolution

So-called “HR” spectra are produced by spectrographs such as the **Lhires III** (Shelyak Instruments), a suitably configured **Star’Ex**, and even, as we shall see, the small **TinyStar’Ex** instrument, even though the latter was originally designed for low resolution.

The data for this section can be downloaded from this link: https://buil.astrosurf.com/specinti4/demo/antibes_tse_20260522.zip.

This is an observation made at high spectral resolution ($R = 15000$), on the night following the one used to illustrate low resolution in the previous section, with the same telescope (a 200 mm f/5 Newtonian) and a **TinyStar’Ex**, but with the latter reconfigured to reach high spectral resolution (in particular, fitting a 2400 lines/mm grating in place of the 300 lines/mm grating). The slit is 20 microns wide, and the camera, used with 1×1 binning, is again a model fitted with the Sony IMX585 CMOS sensor.

The object concerned is Regulus (α Leo), a hot star of spectral type B8 IV. It is a fast rotator (very fast indeed: a projected velocity of over 350 km/s, the star completing a full rotation in only 16 hours!), a characteristic that blurs and broadens the $H\alpha$ line, making the star particularly convenient for estimating the instrumental response at high resolution, which is what we are about to do.

The DOF images supplied (`_offset`, `_dark`, `_flat`) are composites of the elementary DOF frames, as explained earlier in the “Simplified use of DOF frames” box.

The spectral region targeted is that of the $H\alpha$ line (the red hydrogen line).

5.1 Calibration at high resolution

We have available a sequence of 6 images of the spectrum of the star Regulus (`regulus-1`, `regulus-2`, ...), together with:

- a separate image of the spectrum of a neon lamp (`regulus_neon-1`), taken immediately after the star’s spectra, by manually waving the lamp in front of the telescope aperture for an exposure of about 20 seconds (see Figure 30) — a manual procedure that is admittedly rudimentary, but simple, effective, and always useful;
- an image of the spectrum of a household LED panel, held in front of the telescope aperture at the end of the night and valid for the whole session (see Figure 31). This item can be bought at any hardware store; it provides the `_flat` component of our DOF frames. The spectral content of LEDs is far from uniform, even over the narrow spectral range covered at

high resolution, but it remains usable in this situation (there is a noticeable deficit of light on the red side compared with the blue side, which is its main drawback).



Figure 30 – Exposure of the spectrum of the reference spectral lamp (a neon night-light).



Figure 31 – Exposure of the spectrum of the flat lamp (an LED panel).

From the standard interface, choose the **high resolution** “HR” option, then the **calibration** section (button at the bottom). Set the working directory — here, for example, “d:/antibes_tse_20260522” — then identify your spectral data as shown in Figure 32. Remember to select the “Neon before or after exposure” option, since we have a reference spectrum that is separate from the science data (we are not working in the so-called “lateral” mode).

Figure 32 – Entering the data for the observation of the star Regulus in preparation for calibrating the instrument. You can be helped by the handy “Autofill” button, provided you were wise enough to name your files properly.

Fill in the DOF section in the usual way, as shown in Figure 33.

Figure 33 – Using the “compressed” version of the DOF frames.

Set the approximate limits of your spectral trace from the “Geometry” button, as shown in Figure 34.

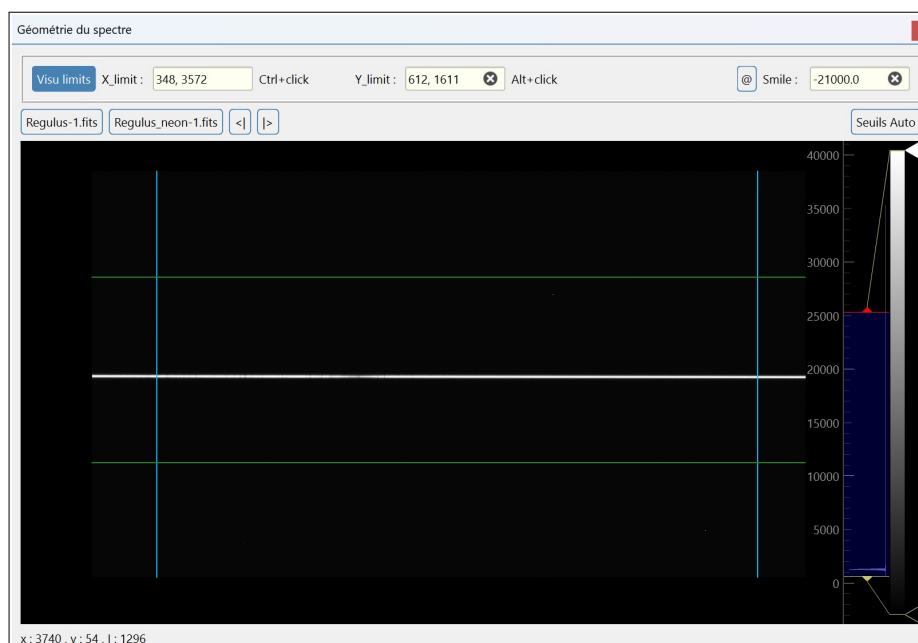


Figure 34 – Defining the limits of the Regulus spectrum in X and Y. There are few constraints on the values chosen, other than keeping the `X_limit` bounds well apart.

Notice the presence of a “smile” button, which is used to compute the radius of curvature of

the spectral lines, a characteristic effect when working at high resolution (a geometric distortion arising from variations in the angle of incidence of light rays on the grating, which becomes more pronounced the more dispersive the grating is). **specINTI Editor** has an interactive tool for determining this radius (here close to $-20\,000$ pixels); enter this characteristic value (do not forget the minus sign, which specifies the direction of the concavity). See also the “Measuring the smile radius” box.

Measuring the smile radius

specINTI Editor makes it possible to measure the value of the **smile** (line curvature, reminiscent of a smile) directly on the images. The difficulty with the data in this example (the “Regulus” session) is that the neon-lamp lines, on which this measurement must be made, are not present in the images of the star’s spectrum, but in a separate file (we are not in “lateral” mode, as noted above — we shall come back to this later in this guide).

A simple trick, however, gets around the problem. Temporarily enter “regulus_neon” as the object name. In the “Geometry” dialogue box, which then displays the neon-spectrum image, click the “@” button (Figure 35). A red frame appears on the neon-spectrum image. Position it with the mouse over a line: the radius of curvature is displayed in real time. Besides moving this window, you can also resize it using the handle located in its top-right corner.

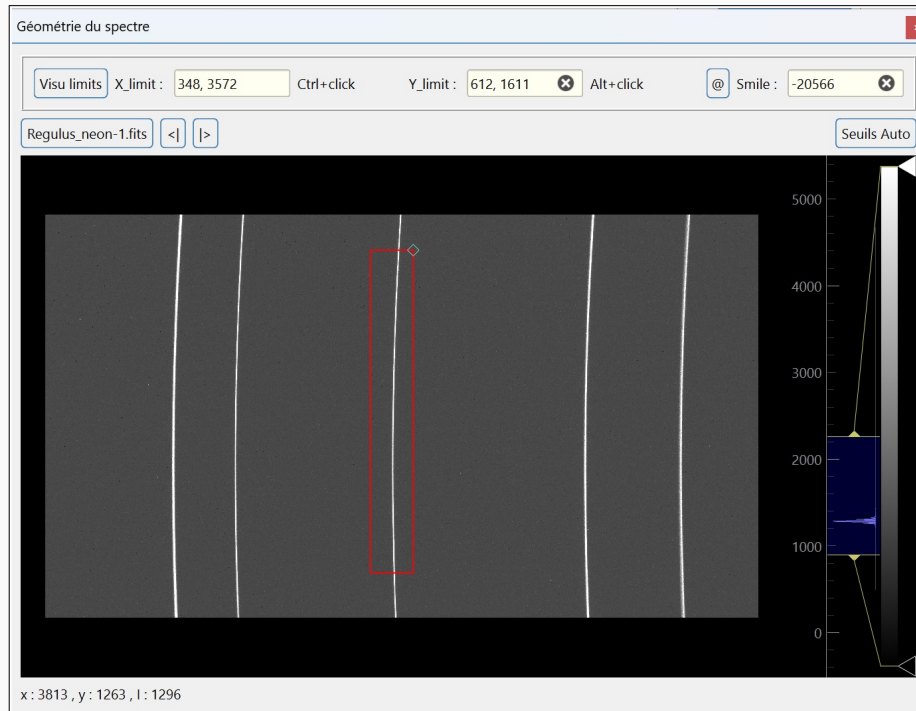


Figure 35 – Measuring the smile radius.

The “Spectrum” button opens a dialogue box containing, among other things, the parameters for trimming the resulting spectral profile and for normalisation (Figure 36). Keep the default values when working in the region of the $H\alpha$ line at high resolution. On this point, also make sure the “Automatic H-alpha region calibration” option is selected from the main interface: **specINTI** will then automatically search for the spectral calibration lines in order to establish the dispersion polynomial in the neon image you supply. This is a particularly useful and powerful convenience.

Paramètres spectre

Effacement raies résiduelles du Néon

Largeur en angströms : 0

Décalage spectral

Décalage en angströms : 0.0

Filtrage

Kernel : -3

Sigma gauss : 0.2

Spectre

Zone crop : 6500, 6710

Zone de normalisation : 6640, 6660

Figure 36 – Setting the spectrum parameters.

Click the “Go!” button to launch the computation of the calibration parameters. Once the process is finished, a curve is displayed in the lower part of the interface, as shown in Figure 37. This is the processed spectral profile, calibrated in wavelength but not yet corrected for the instrumental response.

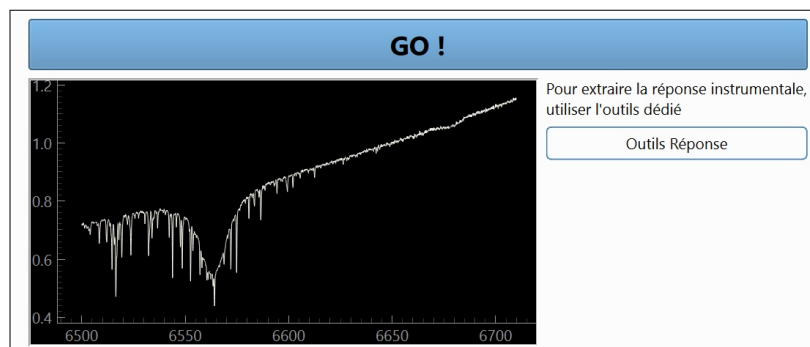


Figure 37 – First step in computing the instrumental response.

Click the “Response tools” button. The spectral profile of the star is then displayed at a larger scale in the “Instr. resp.” tab (Figure 38).

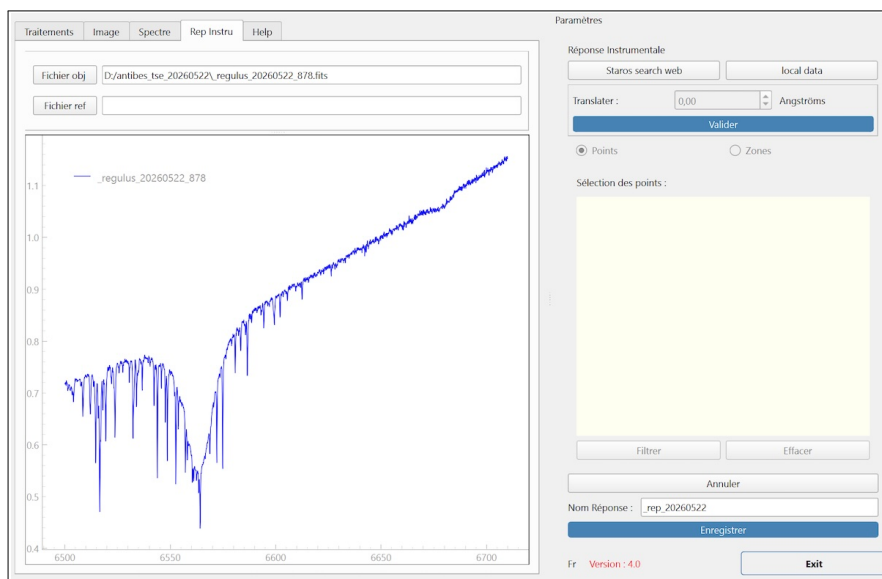


Figure 38 – The interactive tool for computing the instrumental response.

The instrumental response is obtained, as usual, by dividing the observed spectrum by the true (extra-atmospheric) spectrum of the same star, or of a star of the same spectral type. We shall look for this reference profile in the Melchior's high-resolution spectral database, made accessible through the tools developed by the STAROS projects association (<https://staros-projects.org>). The spectrum of Regulus does appear in this library, but it is indexed there only by its HD number. On SIMBAD (for example), you will find that Regulus corresponds to HD 87901.

Click the “Staros search web” button at the top right of the interface. You are then taken to the Melchior's search web page (Figure 39). Enter: **Melchior's HD 87901**.

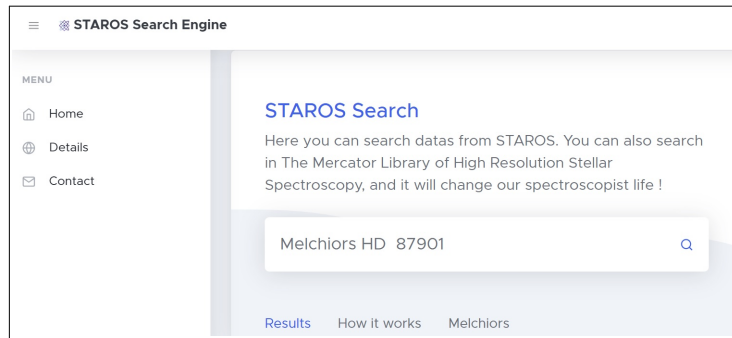


Figure 39 – Querying the Melchior's database from the Staros Projects website (<https://staros-projects.org>).

Press “Return” and, after a short while, the star's spectrum appears on screen (Figure 40). You can download it (“Download FITS”), then save it in the working directory (it is a spectrum in FITS format), for example under the name `_ref_regulus.fits` (you are entirely free to choose the name).



Figure 40 – The reference spectrum of the star Regulus.

From the “Instr. resp.” tab of **specINTI Editor**, load this file using the “Ref file” button. The software then displays, on the same graph, the observed spectrum of Regulus in blue, the Melchior's spectrum in purple, and the ratio of the two in green (Figure 41).

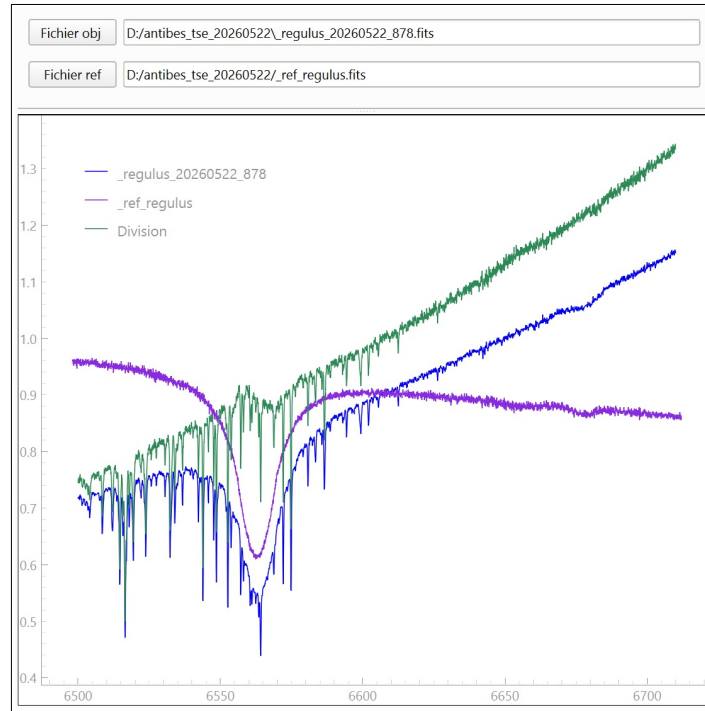


Figure 41 – The observed spectrum of Regulus, the expected spectrum, and the ratio of the two.

Focus in particular on the ratio, the green curve, which will become the future instrumental response. A bump can be seen at the $H\alpha$ line, caused by a spectral shift between the two operands of the division. The observed spectrum is indeed shifted in wavelength owing to the Earth's radial velocity in the direction of the star (more precisely, the projection, along that direction, of the Earth's motion around the Sun), an effect that is noticeable at the spectral resolutions involved here.

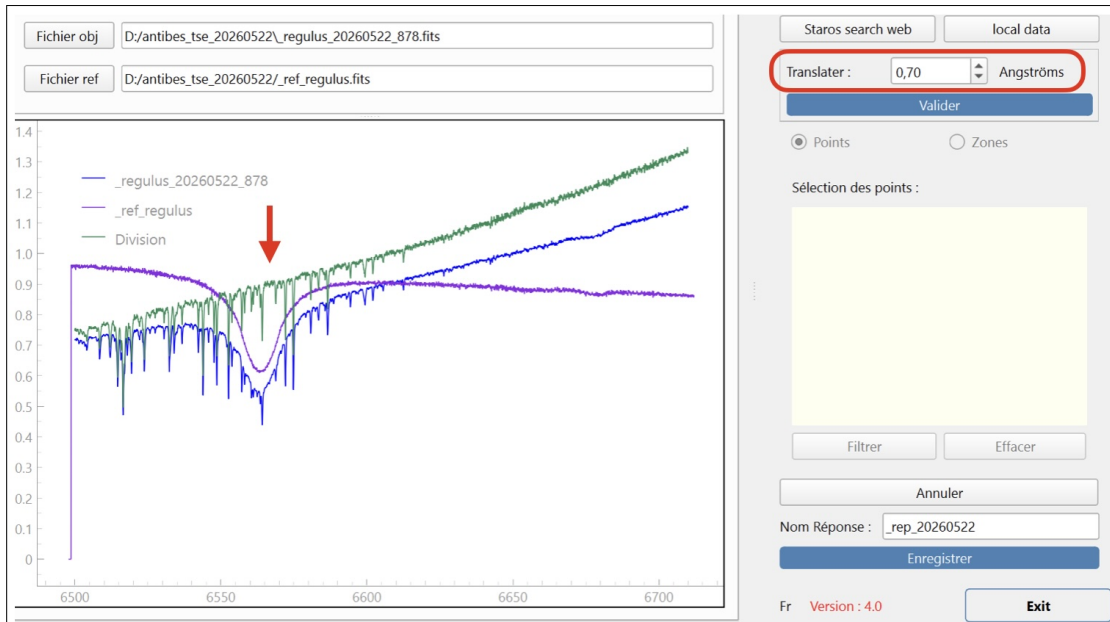


Figure 42 – Shifting the observed spectrum by $0.7 \text{ \AA}$, which restores the correct continuum level at the $H\alpha$ line. You can see the effect of the shift immediately by clicking the up/down arrows of the “Shift” item. Note also the presence of telluric lines in the observed spectrum, whereas they have been removed from the Melchior's spectrum.

Click “Validate”. The response is then isolated, but still contaminated by telluric lines and by noise. The most effective way to clean it up is to define a series of points within the continuum

(away from any lines): place 10 to 20 well-distributed points from one end of the graph to the other, as shown in Figure 43.

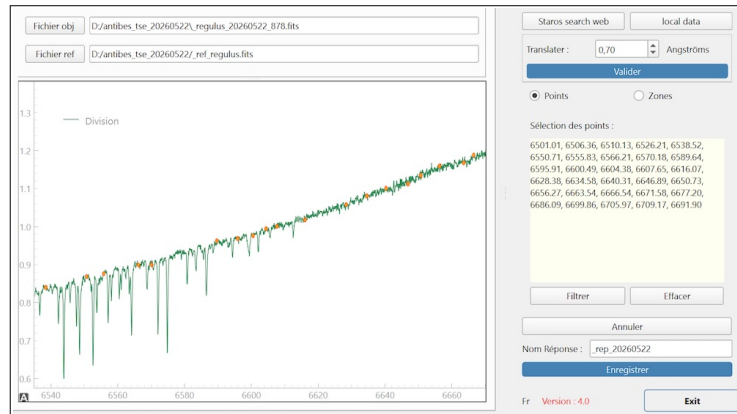


Figure 43 – The red points on the profile were placed with a left mouse click. Do not hesitate to zoom in for greater accuracy and to avoid the telluric lines, which are numerous around the $H\alpha$ line.

Then click the “Filter” button: the software fits the best monotonic curve matching the overall shape of the continuum through these points (Figure 44).

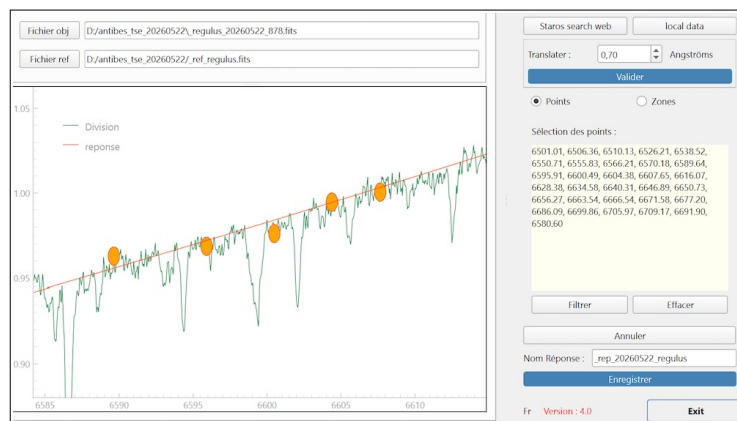


Figure 44 – Fitting the best curve through the selected points (the red line).

Save the result in the working directory. Good practice is to build the file name starting with `_rep`, followed by the date and the object name, so as to more easily identify the origin of your response files later on.

The particular shape of this response curve, with its characteristic slope (Figure 45), should be sought not in the instrument itself, but in the spectral distribution of the source used to evaluate it, which also serves as the flat-field during processing: the light of an LED lamp. If you examine the `_led.fits` image among the DOF frames of the example, you will notice a deficit of light in the red compared with the blue, compensated for in the response profile by an upward slope in the red. The lesson here is this: if the light used to produce the flat-field is not white but coloured (contrary to appearances), the response curve reflects not only the response of the instrument, but also that of this source.

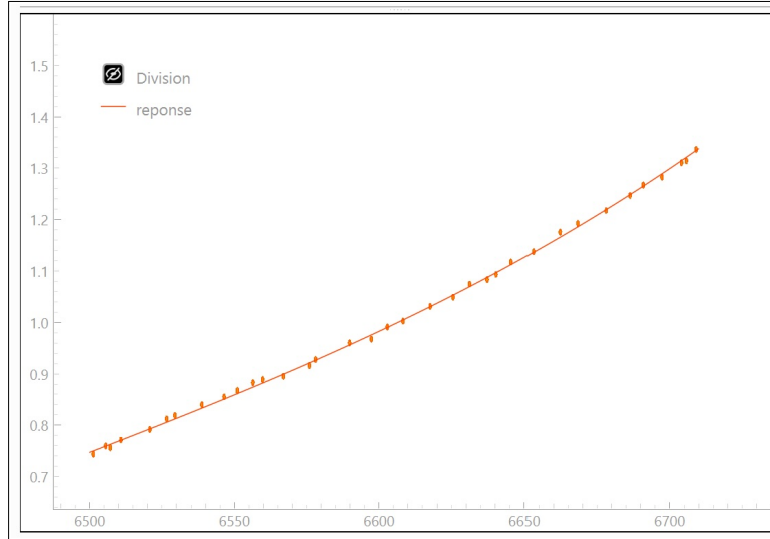


Figure 45 – Appearance of the response curve computed using an LED lamp as the source for producing the flat-field.

5.2 Processing at high resolution

Go back to the “Processing” tab and the HR processing section. Make sure everything is correctly pre-filled to process the sequence of spectra of the star Regulus (Figure 46), in particular that the spectral-response file name is indeed that of the curve computed earlier (`specINTI Editor` fills in this field automatically) and that the DOF names are correct (here `_offset`, `_dark`, `_flat`, see also Figure 33).

Figure 46 – The full set of parameters set up to process the “Regulus” spectrum sequence.

All that remains is to click “Go!” to process the spectrum. The result is shown in Figure 47.

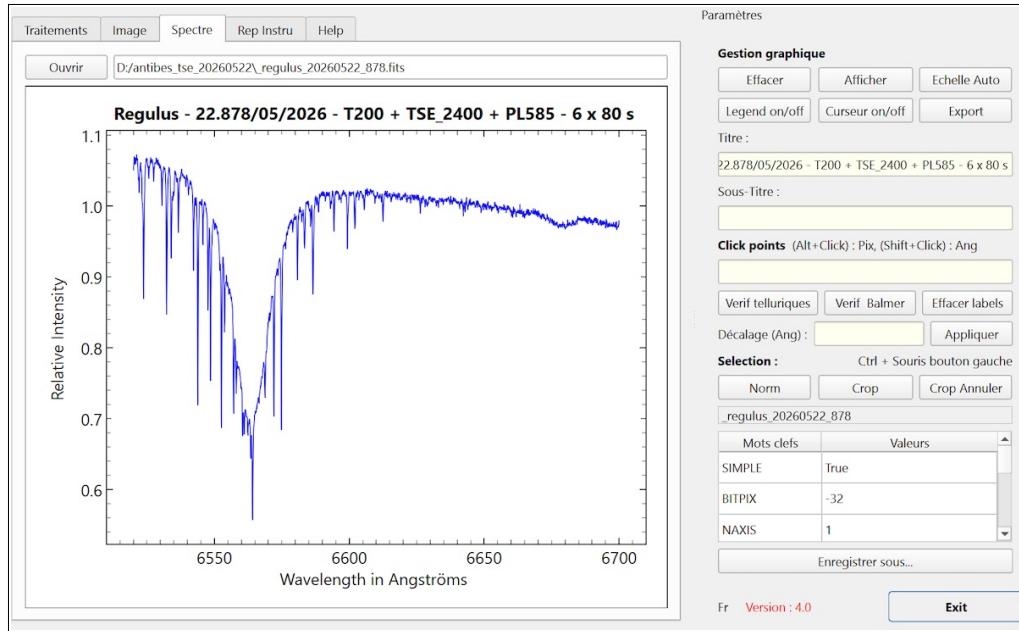


Figure 47 – The final spectrum of the star Regulus.

You can now process your spectra in a very small number of clicks, whether they come from the night in question or from subsequent sessions. To illustrate this, we have added to the downloadable data a short sequence on the Be star β Lyr, acquired under the same conditions. Simply enter the object’s name, click “Autofill”, then “Go!” to obtain the reduced spectrum (Figure 48).

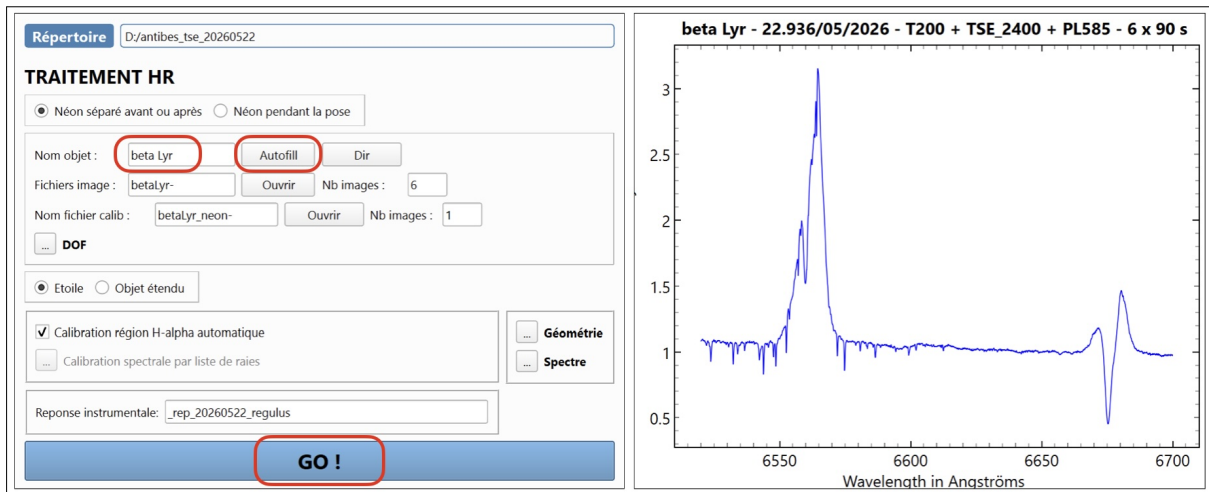


Figure 48 – Processing a sequence of spectra of the star β Lyr.

5.3 Revisiting the instrumental response

The question of how to evaluate the instrumental response is frequently debated and a common source of confusion. The approach followed so far, for reducing both low- and high-resolution spectra, has been to assume that the star used to evaluate this response is angularly close to the target star and that the two observations are made close together in time. In this way, the problem of atmospheric transmission is sidestepped, since it is assumed to be identical in the numerator and denominator when computing the response. It is, incidentally, somewhat misleading to speak of a “instrumental response” in this context, since the result in fact also depends on atmospheric transmission. Thus, you will not obtain the same result depending on

whether you observe high in the sky or close to the horizon, or depending on the level of aerosols suspended in the atmosphere — and yet the instrument itself has not changed.

In practice, the observer has two broad strategies available, summarised in Figure 49.

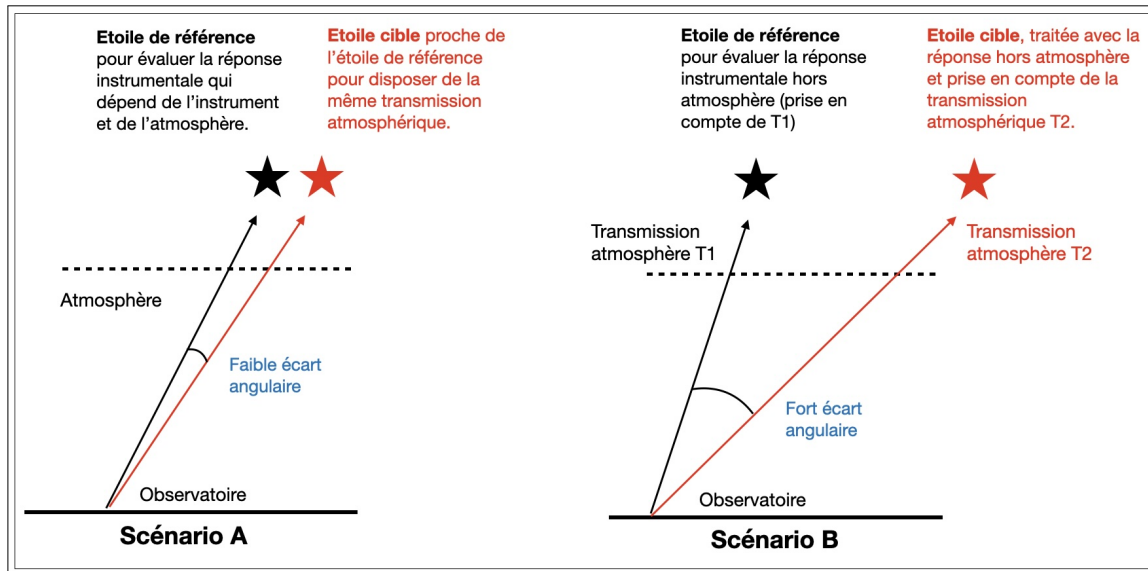


Figure 49 – Techniques for evaluating the “instrumental response” (see text).

Scenario A corresponds to the conventional approach. The observation of the target star is complemented by that of a nearby reference star selected from standard spectral libraries (e.g. Melchior, Pickles, or MILES). This approach has two disadvantages. First, additional observing time is required for telescope pointing and the acquisition of the reference star. Second, a suitable reference star must be identified in the vicinity of the target³.

Scenario B consists in determining, once and for all, the intrinsic instrumental response, corrected for the atmospheric transmission prevailing at the time of its determination, that is, the response that would be obtained outside the Earth’s atmosphere. For each subsequent observation, the inverse procedure is applied: the atmospheric transmission corresponding to the observing conditions (mainly determined by the target altitude and sky transparency) is first computed and removed before applying the intrinsic instrumental response, which is independent of the atmosphere. This approach also presents two limitations. First, atmospheric dispersion produces a secondary spectrum whose extent increases towards low elevations and whose orientation varies with the parallactic angle. Second, when a refracting telescope is used, the derived response becomes sensitive to the quality of the focus, even for an apochromatic instrument, especially at ultraviolet wavelengths. Both effects may bias the derived instrumental response. In practice, however, these biases remain small provided that observations are not acquired at low elevations (preferably at an azimuth similar to that of the reference-star observation) and that a reflecting telescope is used instead of a refractor.

The choice between these two strategies is left to the observer. In our experience, however, **Scenario B** is generally preferable for three reasons. It is both simpler and faster to implement, since a single instrumental response can usually be applied over an entire observing night, or even across several observing sessions. In addition, uncertainties associated with the atmospheric transmission model rapidly become negligible compared with the other sources of error. Finally, **Scenario A** is itself not immune to focus-related effects or to more subtle biases, such as the

3. Several convenient online tools are now available for this purpose, for instance <https://spectro-starfinder.net/>. The specINTI documentation also provides a list of bright A-type stars that can be used as reference stars.

position of the stellar image on the slit, which depends on the stellar spectral type and therefore reintroduces sensitivity to atmospheric dispersion.

Implementing **Scenario B** requires adding a few parameters to the configuration files automatically generated by **specINTI** in standard mode. These are advanced features, which nevertheless remain transparent to the user. Making use of them requires the advanced interface of **specINTI**, the use of which is described in the sections that follow.

6 Advanced interface: use in low spectral resolution

Figure 50 shows how to switch from the standard interface to the advanced interface.

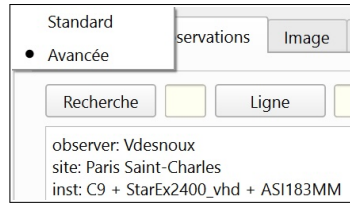


Figure 50 – Switching to the advanced interface of **specINTI** Editor.

The advanced interface relies on the *manual* editing of two files, each in the form of a list of parameters: the **configuration file**, which describes the layout of the spectrum to be reduced, the instrument, and the type of reduction required; and the **observation file**, which describes the data to be processed. The role of these files is detailed in the next section.

Figure 51 shows the general layout of this interface. From left to right: the configuration-file editor, the observation-file editor, the tool for selecting these files, and the output console.

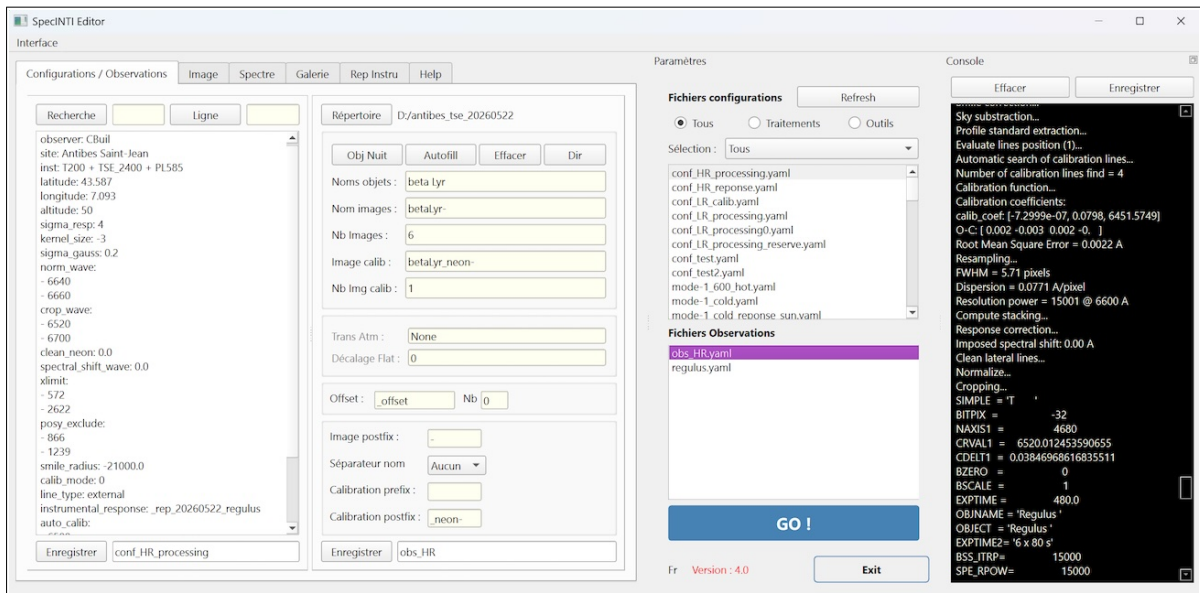


Figure 51 – Layout of the advanced interface of **specINTI** Editor.

To illustrate the use of advanced mode, we shall use a series of images of the spectrum of the star δ UMa, of spectral type A2 V, notable in particular for well-marked Balmer-series lines.

The data were acquired with a 200 mm f/5 Newtonian telescope. The spectrograph is a **TinyStar'Ex** fitted with a 14 μm -wide slit and a CMOS Player One Uranus M Pro camera (Sony IMX585 sensor, used with 1×1 binning).

You can download this data by clicking on the following link: https://buil.astrosurf.com/specinti4/demo/tse_20260415.zip.

This is only an extract from a fuller sequence, deliberately reduced to keep the data volume small — at the cost, however, of the final signal-to-noise ratio.

As always, the first operation is to tell the software where the data is located on your storage system. In this example, the directory path is `c:\tse_20260415` (the spectrograph’s short name, followed by the observation date, here 15 April 2026). The choice of name and location is entirely up to you, but it is recommended to avoid overly long paths and to favour short, readable names, **with no spaces**.

From the “Configuration/Observation” tab, click the “Directory” button, then select your working directory for the session (Figure 52). From the “Image” tab, you can examine the images



Figure 52 – Choosing the working directory.

in the series of the star δ UMa (three 5-second exposures). These images, in FITS format, are named `DeltaUma-1.fits`, `DeltaUma-2.fits` and `DeltaUma-3.fits`:

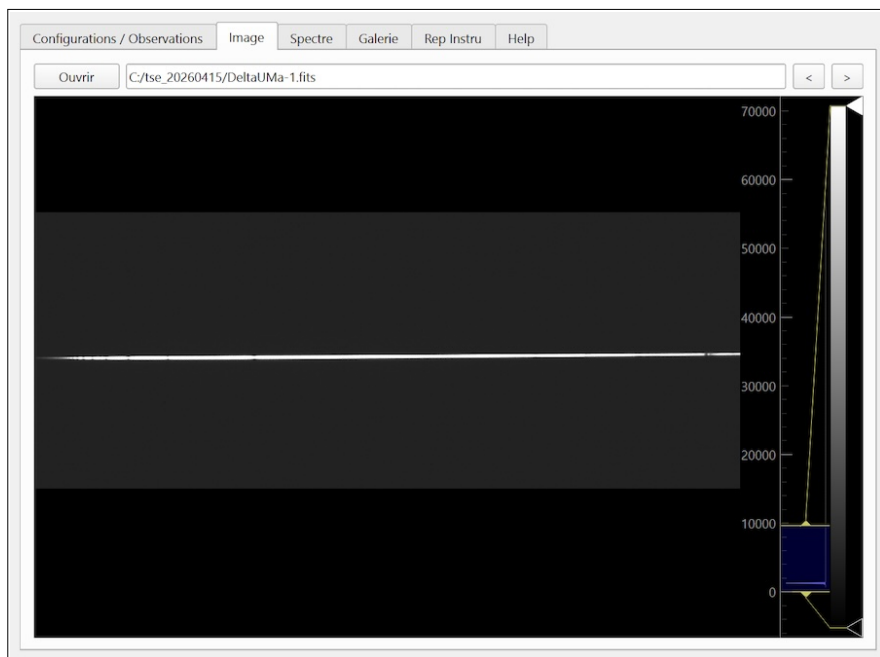


Figure 53 – Display of the spectrum of the star δ UMa.

Figure 54 shows an enlargement of this spectrum, in which the Balmer-series lines appear clearly.



Figure 54 – Detail of the spectrum of the star δ UMa.

Note that, to improve reduction efficiency, the spectral image was windowed at acquisition: it is only 900 pixels high here, centred on the spectral trace.

Let us now turn to what makes advanced mode both distinctive and powerful, by describing the role of the configuration and observation files.

6.1 Understanding the configuration and observation files

The **configuration file** is a file you can write and view with any basic text editor. This editing work is also one of the functions of the “Configuration/Observation” tab in **specINTI Editor** (on the left-hand side of the tab).

When you click the “Save” button beneath this editing area, the text is always saved in a subdirectory of the application’s installation directory bearing a predefined name: `_configuration`. The presence of this directory is required for **specINTI** to run.

The list of configuration files already present in the `_configuration` directory is shown in the “Parameters” part of the “Configurations/Observations” tab; see Figure 55.

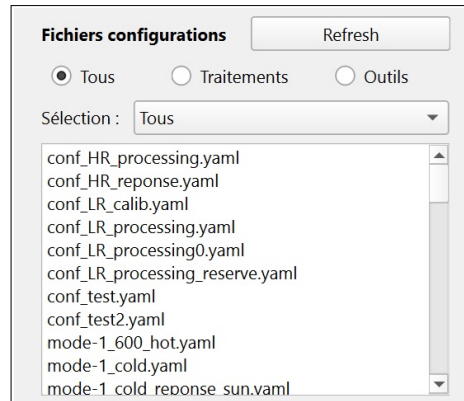


Figure 55 – List of parameter files present in the `_configuration` subdirectory.

To display the contents of one of these files, and possibly edit it, simply click on its name with the mouse. You are, moreover, entirely free to choose the names you give to the configuration files you create, but try to name them logically, according to the type of reduction associated with the configuration described.

But what, exactly, is a configuration file, and what is it for?

Although you can save and edit a configuration file just like a plain text file, its content is nevertheless not arbitrary. It is written in YAML format (pronounced *yamel*), used in computing to store structured data in a highly readable way (its name stands for *YAML Ain’t Markup Language*). These files carry the “.yaml” extension.

A YAML file consists of a set of pairs, each comprising a keyword followed by a value, in the form **key: value**.

These pairs are normally organised by indentation, but in our examples we adopt a simple structure: one parameter per line, that is, one **key: value** pair per line.

The **specINTI** software uses the content of the configuration file passed to it as a parameter to understand what you are asking it to do: perform a calibration, process a nebula’s spectrum, and so on. This file also describes the nature of the spectra being processed, their layout, the operations to be carried out on them, and the observing conditions. The whole challenge for you is to write this content correctly to suit your needs. You are helped in this by the **specINTI Editor** interface.

Once the configuration file has been edited, clicking the “Go!” button saves it in the “_configuration” directory, then launches the **specINTI** program, passing it the name of the file. This operation is transparent to the user.

Splitting the software into two parts — the interface on one side, the computation engine on the other — gives great flexibility in developing the application. You could, for example, build

an entirely new interface from scratch, to your own taste, without this affecting the quality of the reduction. This is how the `specINTI` application finds its way into processing pipelines set up to automatically handle data from remote installations — the software then running even without an interface at all!

We have seen that each line of the configuration file contains a parameter (whose name is reserved) and an argument. It is the values of these parameters — some of which are optional and may therefore be omitted — that govern how `specINTI` behaves. Let us take an example of a compulsory parameter:

```
working_path:  C:/tse_20260415
```

The parameter name (or key) here is `working_path`, and the argument is `C:/tse_20260415`.

Pay close attention to the syntax here: strict care is required. The colon “:” is compulsory and must be attached directly to the parameter name, with no space. A single space, on the other hand, separates the “:” from the value assigned to the parameter. Note also that a space must never be inserted before the key name.

What does this `working_path` parameter do? It tells `specINTI` the path of the folder containing the data for the observing session to be processed, here `C:/tse_20260415`.

Let us take another example:

```
batch_name:  regulus
```

The value of this parameter directs `specINTI` towards processing the spectrum of an object whose observation details (object name, number of images, DOF frames to use, etc.) are specified in an observation file called “regulus” (this is just an example). This file is itself a “.yaml” file, and it is saved in the current working directory using the details entered in the “Configurations/Observations” tab. For example, the following content entered into the interface:

The screenshot shows the 'specINTI Editor' interface. At the top, there's a 'Répertoire' field with the value 'C:/starex_20250405'. Below this are four buttons: 'Obj Nuit', 'Autofill', 'Effacer', and 'Dir'. The main configuration area consists of several input fields: 'Noms objets' (regulus), 'Nom images' (regulus-), 'Nb Images' (7), 'Image calib' (regulus_neon-), and 'Nb Img calib' (-1). Below these are 'Trans Atm' (None) and 'Décalage Flat' (0). A section for offsets and dark frames includes 'Offset' (o-), 'Nb' (4), 'Dark' (n900-), 'Nb' (6), and 'Flat' (led-), 'Nb' (5). The bottom section includes 'Image postfix' (-S), 'Séparateur nom' (Aucun), 'Calibration prefix' (empty), and 'Calibration postfix' (_neon-). At the very bottom, there's an 'Enregistrer' button and a text field containing 'regulus'.

Figure 56 – Preparing the observation file from the **specINTI Editor** interface.

will produce the observation file “regulus.yaml”, whose content is as follows:

```
regulus:
- regulus
- regulus-
- 7
- regulus_neon-
- -1
- led-
- 5
- n900-
- 6
- o-
- 4
- None
- 0
```

It is undeniably easier to write this file from a user-friendly interface than from a text editor, with the risk of syntax errors this carries. Naturally, **specINTI** knows perfectly well how to decode this content in order to process the sequence concerned, here that of the star Regulus.

It is possible to go even further, processing several targets in sequence with no user intervention at all (a *batch* mode). Simply list several object names to be processed from the interface, separated by a comma; see the example in Figure 57.

Obj Nuit	Autofill	Effacer	Dir
Noms objets :	regulus, pollux		
Nom images :	regulus-, pollux-		
Nb Images :	7, 5		
Image calib :	regulus_neon-, pollux_neon-		
Nb Img calib :	1, -1		

Figure 57 – Preparing sequential (*batch-mode*) processing of the spectra of the stars Regulus and Pollux.

In this example, the observation file becomes:

```
regulus:
- regulus
- regulus-
- 7
- regulus_neon-
- 1
- led-
- 5
- n900-
- 6
- o-
- 4
- None
- 0
pollux:
- pollux
- pollux-
- 5
- pollux_neon-
- -1
- led-
- 5
- n900-
- 6
- o-
- 4
- None
- 0
```

The list can contain as many objects as you like. In this way you can process an entire observing session with a single click of the “Go!” button!

The argument of a parameter can be a character string, but it can also be a number, or even a series of numbers, using the following syntax:

```
crop_wave: [3500, 7500]
```

This parameter instructs the software to trim the final spectrum between 3500 Å and 7500 Å.

The order of parameters is entirely unimportant in a configuration file: you can therefore organise it however you like. The use of lower- or upper-case letters is likewise unimportant.

Note that any text beginning with the character # is treated as a comment. Example:

```
# This is a comment
crop_wave: [3500, 7500]    # Windowing
```

The use of a configuration file lies at the heart of how **specINTI** works. This was already true when the **standard interface** was discussed at the beginning of this document. Even in that mode, every time a calibration or a reduction is launched from **specINTI Editor**, a configuration file is created in the **_configuration** directory under a fixed name — for example “**conf_lr_processing.yaml**” when you request low-resolution processing — and is then read by **specINTI** to carry out the reduction. With the standard interface, the concept of a configuration (and observation) file is present underneath, but hidden for the sake of simplicity.

6.2 Evaluating the spectral calibration law and the instrumental response

Let us now go back to the observation of the star δ UMa. To begin with, we shall use it to calibrate our instrument, both in wavelength and in response to the incoming light.

The result obtained at the end of this step can be used to process every object in the session, and indeed in subsequent sessions, for as long as the equipment has not been significantly altered.

We therefore have three spectra of the star, together with four bias images (**o-1**, **o-2**, ...) and three dark-signal images, corresponding here to long 30-minute exposures (**n1800-1**, ...). No flat-field images were acquired: these are considered unnecessary in this context, given the high performance of the IMX585 sensor and the fact that low-spatial-frequency response defects will be handled when computing the instrumental spectral response, as we shall see later. Incidentally, working without flat-field images is a welcome simplification of the observing routine.

Go to the “Configuration/Observation” tab of **specINTI Editor**, then fill in the fields as shown in Figure 58.

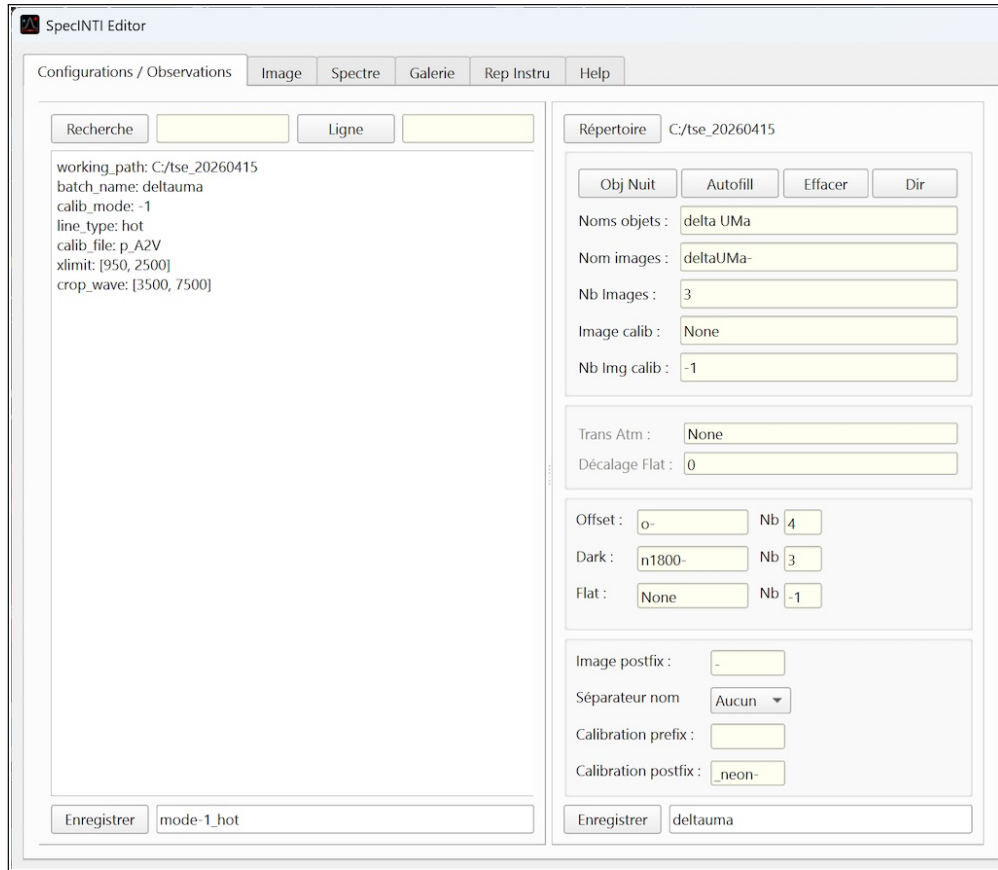


Figure 58 – Configuration for extracting the low-resolution calibration parameters.

The right-hand panel gathers the information relating to the observation data and to the DOF frames, saved here in an *observation file* named `deltauma.yaml` (any name will do). We simply enter the target’s name (“delta Uma”), then click the “Autofill” button. This name will appear in the FITS header of the reduced spectrum; it is therefore important that it be an official name recognised by SIMBAD. If you click the “Save” button, or if you launch the computation, the observation file is automatically saved in your working directory.

The left-hand panel concerns the *configuration file*, which specifies the type of reduction required. We have seen (Section 6.1) that it is saved in the `_configuration` folder within the software’s installation directory, whether because you clicked the “Save” button, or because you launched the computation.

Calibrating the instrument in wavelength and in response is achieved by activating a specific processing mode, identified by a number passed to a parameter called `calib_mode`. This number is `-1` here, the minus sign indicating that this is a **utility mode**. This mode is particularly powerful, since, with a single click and within a few seconds, it carries out operations often perceived as complex or off-putting when first encountering spectroscopy.

This configuration has therefore been named `mode-1_hot`, with `specINTI` automatically appending the `.yaml` extension. This name reflects both the processing mode (`-1`) and the use of the spectrum of a hot star (*hot*) for calibration. Here we find some of the conveniences offered by the standard interface (see Sections 4 and 5), but this time within the advanced interface.

Rather than a star of spectral type A, the software also offers the option of using a solar-type star (type G), whose spectrum is more complex. The choice between a hot star and a cool star is determined by the value of the `line_type` parameter in the configuration file: “hot” or “cold”. Table 2 summarises the options offered by mode `-1`.

Table 2 – Table of special and utility modes

Function	Mode	Line_type	Calibration source	Configuration settings
Automatic computation of the spectral calibration law and the instrumental response	-1	Cold	Fraunhofer lines of a spectral-type-G star (natural source)	A type-G (solar-type) reference spectrum is needed. The name of the corresponding file is given via the <code>calib_file</code> parameter
		Hot	Balmer-series lines of a spectral-type-A star (natural source)	A type-A (hot star) reference spectrum is needed. The name of the corresponding file is given via the <code>calib_file</code> parameter
	-2		Computes a spectral calibration polynomial from the approximate positions of a series of absorption lines and their associated wavelengths. Input: the spectrum of a star	The <code>line_pos</code> parameter holds the pixel coordinates of the lines, while <code>wavelength</code> holds the corresponding wavelengths
	-3		Same function as mode -2, but for emission lines.	Same parameters as mode -2.
	-4		Fits a dispersion polynomial of the chosen degree, from a list of wavelengths and the corresponding pixel coordinates. The polynomial coefficients are shown in the output console	The <code>line_pos</code> parameter holds the pixel coordinates of the lines, while <code>wavelength</code> holds the wavelengths
	-5		Extracts the raw profile of a star's spectrum from its two-dimensional image, with no spectral calibration and no instrumental-response correction. The intensities, in ADU, correspond to the sum of the spectra extracted from the images in the supplied list.	The only compulsory parameter is <code>xlimit</code> (spectrum limits used to compute the geometric parameters)

The content of a configuration file is very flexible: some parameters are compulsory, while a larger number of others are optional. Here is the file we shall use to calibrate the instrument. At this stage it consists of just seven lines (you can copy and paste them into the editing area of the interface):

```
working_path: C:/tse_20260415
batch_name: deltauma
calib_mode: -1
line_type: hot
calib_file: p_A2V
xlimit: [950, 2500]
crop_wave: [3500, 7500]
```

Here is the same content, this time annotated:

Listing 1 – Configuration for automatic extraction of the calibration parameters

```
working_path: C:/tse_20260415 -> location of the data and of the observation file
batch_name: deltauma -> name assigned to the observation file
calib_mode: -1 -> automatic extraction mode for the calibration parameters
line_type: hot -> hot star (spectral type A)
calib_file: p_A2V -> reference spectrum used for calibration
xlimit: [950, 2500] -> limits (in pixels) for the search for the Balmer lines
crop_wave: [3500, 7500] -> limits of the spectral range processed (in Å)
```

Let us look in more detail at the role of these parameters:

- **working_path**: designates the directory containing the data to be processed. This folder also contains the DOF files, as well as the observation file (`xxx.yaml`). This field is updated automatically when the observation data is saved and when the computation is launched (“Go!” button).
- **batch_name**: specifies the name of the observation file describing the sequence of data to be analysed. This file contains, among other things, the list of images, their number, and the names of the associated DOF files.
- **calib_mode**: defines the program’s mode of execution. A negative value (here `-1`) activates the utility responsible for automatically determining the spectral calibration law and the instrumental response. See also Table 2.
- **line_type**: in mode `-1`, two types of star can be used to establish the dispersion law. The *hot* type corresponds to a hot star showing well-marked Balmer lines, while the *cold* type corresponds to a cooler star, typically of spectral type G or K.
- **calib_file**: designates the reference spectrum used to determine the instrumental response. The software automatically looks for this file in the “calib” directory, created during installation. This folder contains, among other things, the spectra of the *Pickles* library, particularly well suited to processing low-resolution spectra (see also Section 4.1).

In the example presented here, we use the file `p_A2V.fits`, since A2 V is precisely the spectral type of the star δ UMa. The available spectral types do not always correspond exactly to those of the stars observed, but the error introduced by using an approximate type⁴ generally remains negligible compared with other sources of uncertainty.

- **xlimit**: the two values given (in pixels) delimit the region of the spectrum within which the software automatically searches for the Balmer lines. On the blue side, it is recommended to select a region containing the $H\eta$ or $H\zeta$ line (see Figure 59). On the red side, the $H\alpha$ line must always be included within the chosen interval.

4. It is mainly the colour temperature that matters here, more than the luminosity class.

These limits should be set by examining the position of the lines in the spectral trace. Once determined, they remain, in principle, constant for as long as the instrument’s mechanical configuration stays unchanged.

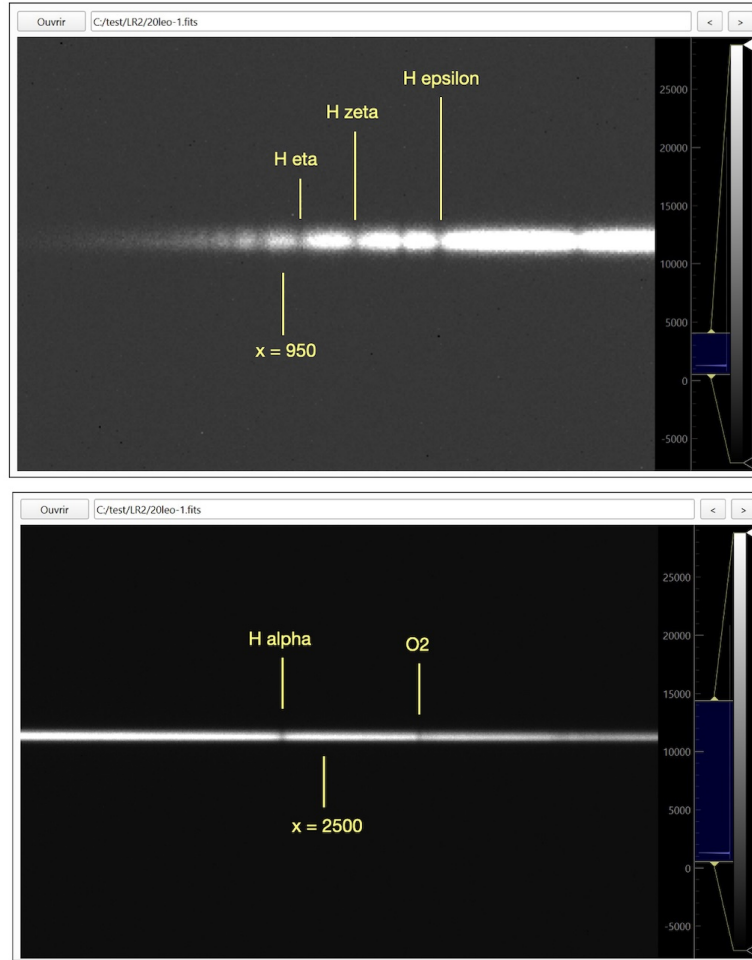


Figure 59 – Example of the zone used for the automatic search for the Balmer lines: at the top, the blue side; at the bottom, the red side. The Balmer-series lines, from $H\alpha$ to $H\eta$, are used here for the spectral calibration. The values shown are specific to the TinyStar’Ex spectrograph; they will differ for another model.

- **crop_wave**: this parameter defines the blue and red limits of the portion of the spectrum considered usable in the computed spectral-response file⁵. The response profile is computed only between these two limits, expressed in angstroms.

By default, **specINTI** automatically determines the height of the binning zone used to extract the spectral profile, as well as the zones used to evaluate the sky background. Automatic detection of the spectral trace in each image rounds off this system: taken together, this constitutes a very welcome simplification for the user.

Once the configuration has been set up, simply click the “Go!” button.

The reduction takes only a few moments. Once it is finished, examine the last lines shown in the output console (Figure 60): you will find there a set of essential information for what follows.

⁵. The red limit comes from grating-order overlap; the blue limit, from the optical transmission of the instrument.

```

Line coordinates find:
[ 959.04  988.23 1032.04 1103.54 1232.83 1514.16 2427.63]
Wavelength:
[3835.38 3889.05 3970.07 4101.74 4340.47 4861.33 6562.85]
Calibration coefficients:
calib_coef: [9.9897e-06, 1.8234, 2079.2709]
Root Mean Square Error = 0.0879 Å
Response file:
C:/tse_20260415\_rep_20260415_deltauma.fits
End.

```

Figure 60 – Content of the output console once the reduction is finished.

Among other things, you will find the list of Balmer lines used to compute the spectral calibration law (their position in the first image of the sequence, together with their corresponding wavelengths).

The two genuinely important pieces of information — the ones this mode `-1` is designed to provide — are the coefficients of the calibration polynomial and the instrumental-response curve.

The coefficients of the calibration polynomial are stored under the `calib_coef` parameter. The characteristic (RMS) error here is of the order of $0.1 \text{ \AA}$, for a spectrum sampled at $1.77 \text{ \AA/pixel}$, which reflects the quality of the calibration. You will need to copy this `calib_coef: ...` line when the spectrum itself is later processed.

As for the instrumental response, it is saved as a spectral profile in the working directory, here under the name `_rep_20260415_deltauma.fits`. You can display it from the “Spectrum” tab (see Figure 61).

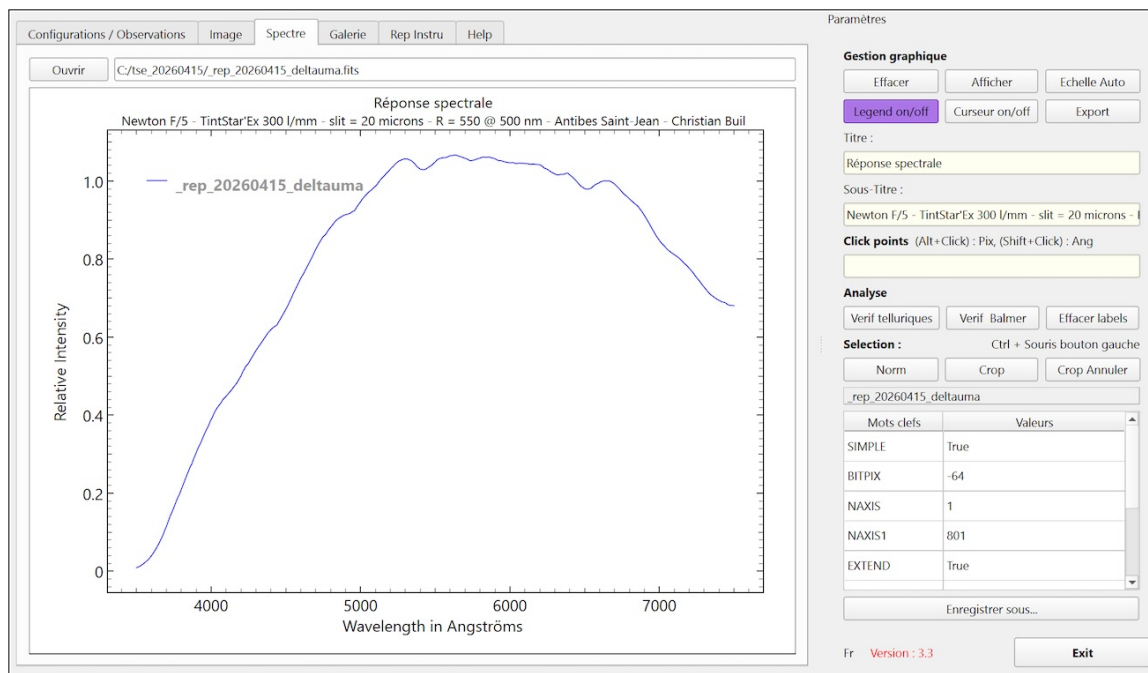


Figure 61 – Spectral response computed from the observation of the star δ UMa. The oscillations seen correspond largely to the instrumental signature rather than to artefacts: they arise from the spectral efficiency of the detector, which varies slightly with wavelength in the case of the IMX585 sensor.

6.3 Variations on computing the calibration parameters

One of the strengths of `specINTI` is that it gives the user great flexibility to adapt to the different situations encountered. The software’s behaviour can be adjusted by editing the parameters in the configuration file: removing some, adding others, or simply changing their values.

In the previous section, we set up a deliberately minimal configuration file, which nevertheless gives entirely satisfactory results in the majority of cases. It will, however, need to be adapted in certain situations.

For example, processing spectra from an Alpy 600 spectrograph requires a degree-3 polynomial (owing to the presence of a “grism”-type dispersive element), whereas `specINTI` fits a degree-2 polynomial by default, which is sufficient for accurate calibration of Star’Ex LR or TinyStar’Ex spectrographs. To fit the spectral dispersion law with a degree-3 polynomial, you must state this explicitly by adding the following line to the configuration file (its position does not matter):

```
poly_order: 3
```

Another example: it may be necessary to adjust the software’s settings in order to calibrate spectra in the infrared range, which the software also allows.

By way of illustration, here is an enriched configuration file, offering more levers for fine-tuning the calibration parameters, which you are free to modify as you see fit:

```
working_path: C:/tse_20260415
batch_name: deltauma
calib_mode: -1
line_type: hot
calib_file: p_A2V
poly_order: 2
xlimit: [950, 2500]
exclude_zones: [(5860, 5975), (6520, 6620), (6840, 7040), (7140, 7370), (7570, 7710)]
sigma_resp: 8
crop_wave: [3500, 7500]
norm_wave: [6620, 6640]
```

The following listing briefly explains the meaning of the optional parameters added. The order of the parameters does not matter: you can organise this file however you like. However, take care to avoid typing errors (the space after the colon, in particular, is important).

Listing 2 – Configuration for automatic extraction of the calibration parameters

```
working_path: C:/tse_20260415
batch_name: deltauma
calib_mode: -1 -> automatic extraction mode for the calibration parameters
line_type: hot -> hot star (spectral type A)
calib_file: p_A2V
poly_order: 2 -> degree of the spectral calibration polynomial
xlimit: [950, 2500]
exclude_zones: [(5860, 5975), (6520, 6620), (6840, 7040), (7140, 7370), (7570, 7710)] ->
zones excluded when computing the instrumental response (in Å)
sigma_resp: 8 -> smoothing coefficient for the instrumental response
crop_wave: [3500, 7500]
norm_wave: [6620, 6640] -> normalisation zone for the instrumental response
```

Let us look at these parameters in detail:

- **exclude_zones**: certain portions of the observed spectrum are affected by telluric absorption bands, mainly due to molecular oxygen (O₂) and water vapour (H₂O). Since these features are absent from the reference spectrum, they must be excluded from the computation of the instrumental response. The **exclude_zones** parameter is used to define these regions as wavelength intervals. The values shown are those used by the application by default, but you can change the limits and add further ones, in particular if you are exploring the infrared part of the spectrum.
- **sigma_resp**: this parameter allows more or less strong smoothing of the instrumental-response curve, so as to reduce artefacts remaining after dividing the observed spectrum

by the reference spectrum (see Section 3.3). The higher the value, the stronger the smoothing. Care should be taken, however, not to smooth out genuine structures in the instrumental response: a value of 20, for example, produces a curve that looks visually smoother, but at the risk of erasing certain significant features of the response. There is inevitably some subjectivity here, but on the one hand a feel for it is quickly acquired with practice, and on the other hand the criticality is not especially high once the overall shape of the response has been computed correctly.

Setting `sigma_resp` is arguably the trickiest step of the process. The default value (6) is generally satisfactory and does not need adjusting in most cases. To fine-tune this setting, note that `specINTI` generates several intermediate files in the working directory: `_op1.fits`, `_op2.fits` and `_op3.fits`. It is recommended to inspect them, especially the first few times you use the software (see Figure 62), using the spectrum viewer available from the “Spectrum” tab.

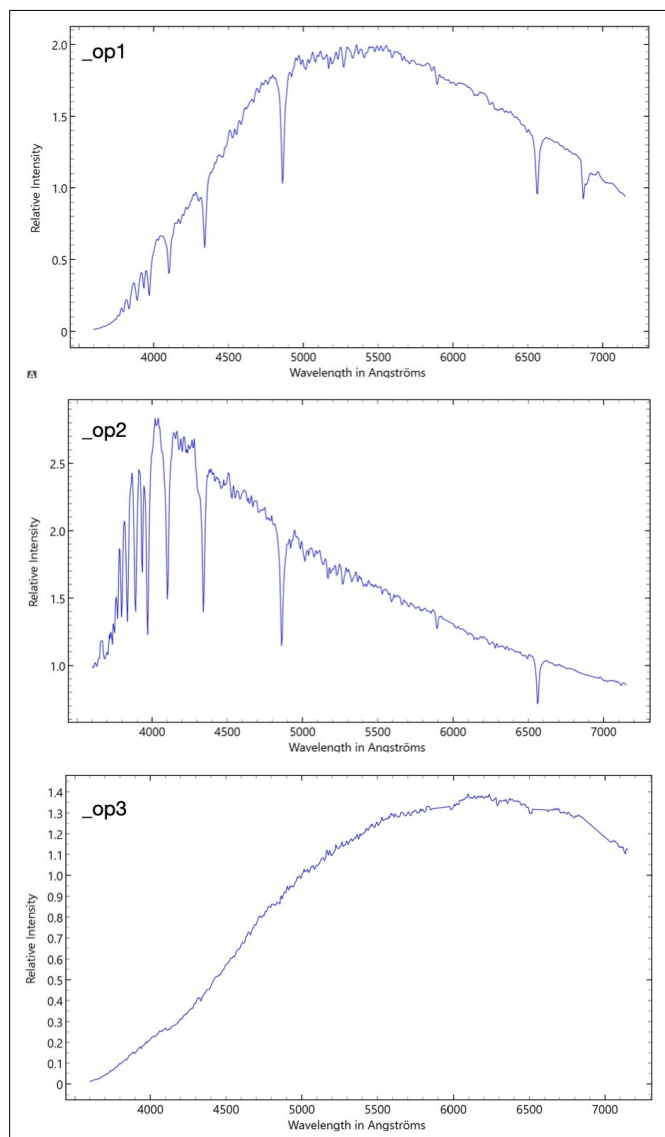


Figure 62 – Content of the `_op1`, `_op2` and `_op3` files.

The `__op1` file is the raw spectral profile of the observed star, after the usual corrections have been applied (offset, dark, geometric rectification, and so on). The `__op2` file contains the reference spectral profile taken from the Pickles library. Finally, `__op3` results from dividing `__op1` by `__op2`: it is therefore a first estimate of the spectrograph’s relative spectral response, before smoothing.

This curve (already calibrated in wavelength) shows local fluctuations, together with the signature of the excluded zones (telluric bands). The smoothing applied via `sigma_resp` is intended to reduce these insignificant irregularities so as to obtain a realistic response curve (see Figure 61). An instrumental response is, indeed, generally smooth, with no rapid variations. Several successive attempts may be needed to adjust this value optimally.

- `poly_order`: defines the degree of the polynomial used to model the spectral dispersion law. A degree of 2 is generally sufficient for Star'Ex or TinyStar'Ex spectrographs at low resolution. Some instruments using a grism as the dispersive element (the Alpy 600, for example) may require a higher degree.
- `norm_wave`: the instrumental response is normalised to unity over a given spectral interval, to make it easier to work with. More precisely, the mean of the response over this interval is set to the value 1.

Let us finally mention the `posy_exclude` parameter. If you used the standard interface of `specINTI` Editor in the earlier sections of this document, you have already set this without realising it. For example, `posy_exclude: [500, 1100]` restricts the software to searching for the spectral trace only between lines 500 and 1100 of your images.

The `sigma_gauss` parameter, for its part, applies a Gaussian smoothing (by convolution) to the spectral images. The value `sigma_gauss: 0` disables all smoothing, while a higher value, for example `sigma_gauss: 3.5`, produces marked smoothing, at the expense of spectral resolution. This parameter can prove useful when computing the instrumental response, to match the resolution of the spectral image to that of the reference spectrum.

Let us also mention a handy feature relating to DOF frames. Very often, these are acquired during a given session, then reused unchanged for subsequent sessions (it is rarely necessary to take DOF frames every night). But you have probably noticed that these “reference” DOF frames then have to be copied manually into each session directory, which can become tedious. This can be avoided by defining a dedicated directory on your hard drive, via the `master_path` parameter. By adding, for example, the following line to the configuration file:

```
master_path: c:\processing\DOF\
```

`specINTI` will look for the DOF frames in this directory, provided they carry the following reserved names: `__offset`, `__dark`, `__flat` (note the double *underscore*). In that case, the corresponding entries in the interface are shown in Figure 63.

Figure 63 – Defining DOF frames shared between several sessions. If a DOF frame is absent, enter “None”.

6.4 Reducing the spectrum at low resolution

Now that we have the calibration parameters (dispersion law and instrumental response), we can move on to the most interesting phase: actually processing the spectra from our observing sessions.

We are going to process the spectrum of the star δ UMa, the very star used to establish these parameters (a self-calibration operation, in other words).

The observation file remains unchanged: we use the same information concerning this star (name, number of images in the sequence, associated DOF frames).

The configuration file, on the other hand, must be changed, since the nature of the processing is different. Several options are possible; let us start with the simplest, shown in Figure 64.

Figure 64 – Configuration for processing a spectrum with no wavelength realignment (`line_type: none`).

Here is this same configuration file, ready to be copied and pasted:

```
working_path: C:/tse_20260415
batch_name: deltauma
calib_mode: 1
line_type: none
calib_coef: [9.9897e-06, 1.8234, 2079.2709]
instrumental_response: _rep_20260415_deltauma
xlimit: [960, 2500]
crop_wave: [3600, 7400]
norm_wave: [6620, 6640]
```

The `working_path` and `batch_name` parameters still point to the data for δ UMa and require no further comment.

The calibration mode, however, changes. Here we are processing data acquired at low resolution; in line with Table 1, we therefore use `calib_mode: 1`. We also choose to apply the coefficients of the calibration polynomial directly, with no adjustment, via `line_type: none`.

These coefficients naturally need to be supplied, via the line:

```
calib_coef: [9.9897e-06, 1.8234, 2079.2709]
```

These are the values obtained during calibration in Section 6.2.

We also give the instrumental-response file used to correct the raw data:

```
instrumental_response: _rep_20260415_deltauma
```

The `xlimit` parameter is carried over unchanged. It is now important to choose two points sufficiently far apart along the spectral trace, to allow the software to accurately estimate its tilt angle — an essential step in correctly restoring the distribution of the true flux coming from the star. A separation of a few hundred pixels between the points, on a well-defined and narrow trace, is satisfactory.

The `crop_wave` parameter defines the useful portion of the spectrum in wavelength. Finally, the optional `norm_wave` parameter normalises the resulting spectral profile to unity over a given interval.

We name the configuration file “`model_lr_none`” here (this is only an example), as a reminder of the broad outline of the computation configuration.

All that remains is to launch the reduction by clicking “Go!”.

After a few moments, the reduced spectrum appears as a thumbnail (Figure 65).

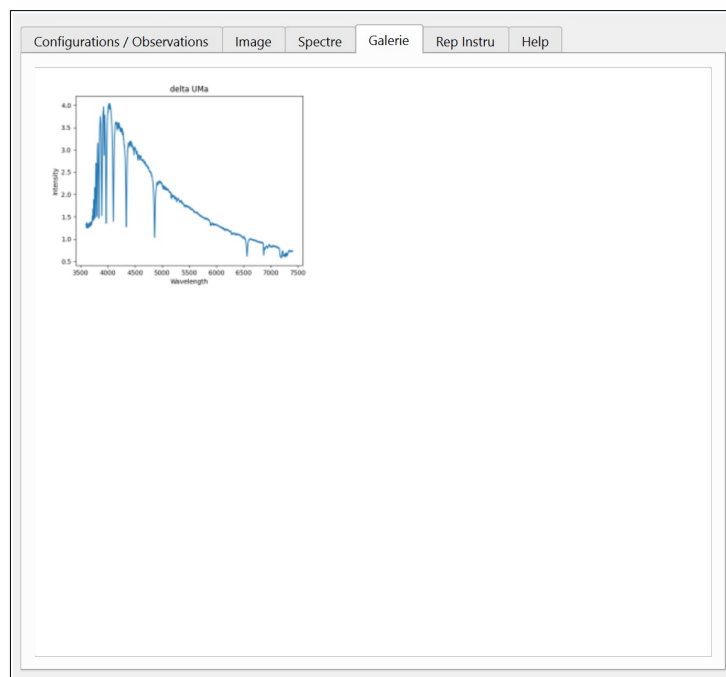


Figure 65 — Preview of the reduced spectrum. When `specINTI` processes several spectra at once, that many thumbnails are shown.

Double-clicking on this thumbnail displays the spectrum at full resolution, as shown in Figure 66.

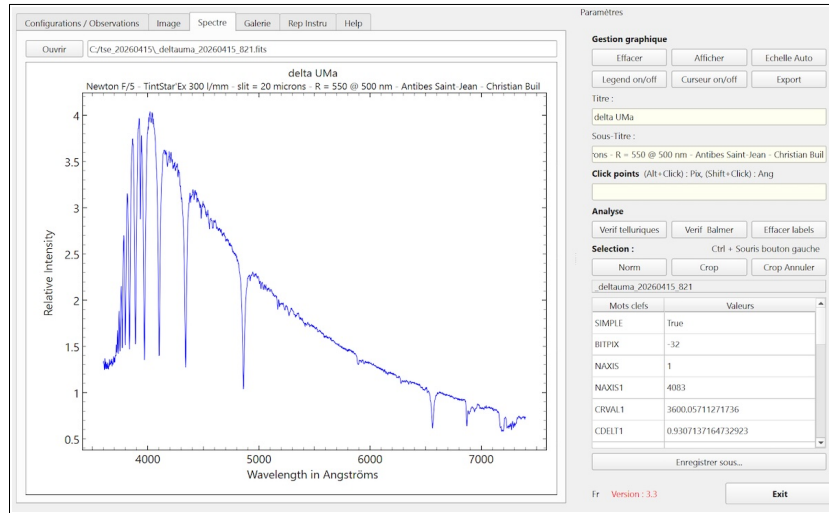


Figure 66 – The reduced spectrum displayed at full resolution.

The spectrum is also saved in the working directory in various forms, in particular in FITS format, under the name `_deltauma_20260415_821`. This name incorporates the object's name together with the precise date of observation (to within a fraction of a day), as shown in Figure 67.

```

SIMPLE = 'T'
BITPIX = -32
NAXIS1 = 4084
CRVAL1 = 3600.05711271736
CDELT1 = 0.9307137164732923
BZERO = 0
BSCALE = 1
EXPTIME = 13.0
OBJNAME = 'delta UMa'
OBJECT = 'delta UMa'
EXPTIME2 = '3 x 5 s'
BSS_ITRP = 0
SPE_RPOW = 0
BSS_VHEL = 0
DATE-OBS = '2026-04-15T19:42:10.056'
BSS_COSM = 'Removed'
BSS_TELL = 'None'
BSS_NORM = 'None'
CRPIX1 = 1
CTYPE1 = 'Wavelength'
CUNIT1 = 'Angstrom'
WAVELENGTH = 'specINII 3.0.0'
JD-OBS = '2461146.32095'
-----
Output files name:
C:/tse_20260415\delta_20260415_821.fits
C:/tse_20260415\delta_20260415_821.png
C:/tse_20260415\delta_20260415_821_2D.fits
C:/tse_20260415\delta_20260415_821.yaml
C:/tse_20260415\delta_20260415_821_log.txt
-----
End of the processing.

```

Figure 67 – Files generated once processing is finished.

6.5 Advanced configurations

The problem most frequently encountered in spectroscopy is drift in the position of the spectral image on the detector over the course of an acquisition sequence.

Temperature variations during the session, together with mechanical deformation linked to gravity and to the telescope's movements, are the most common causes. Even the most robust spectrographs, mounted at the focus of a professional telescope, can be affected by this. The result is degraded spectral resolution, as well as wavelength-calibration errors.

The way to deal with this problem is to compensate for shifts in the spectrum at the processing stage, using information present within the acquired spectra themselves.

The first step to take is to split a very long exposure into relatively short sub-exposures. For

example, a planned 2-hour exposure can be split into 15- to 30-minute sub-exposures, which are then summed at the processing stage. The degree of splitting depends on the intrinsic stability of the spectrograph.

The aim is to recentre *a posteriori*, by processing, the individual spectra of a sequence onto a common reference, generally the first spectrum acquired. Recentring the spectral images along the spatial axis (perpendicular to the spectral axis) is performed automatically by `specINTI`, with no user intervention, unless otherwise specified via the `ypos` parameter.

The situation is trickier along the spectral axis. The software needs reference points to carry out this realignment, and these can be of very different kinds. What is involved is updating the constant term of the dispersion polynomial. The strategy adopted depends on the value assigned to the `line_type` parameter (see Table 1).

Important note: although the processing options presented here are illustrated with so-called low-spectral-resolution spectra, they are more general in nature and can therefore be used to process spectra at intermediate spectral resolution ($R = 2000, 3000\dots$), and even at high resolution. The only requirement is to have the coefficients of the dispersion polynomial, whose constant term is to be adjusted.

6.5.1 `line_type: absorption`

Let us take an example applied to the star δ UMa. Its spectrum shows many well-marked absorption lines, those of the Balmer series. By measuring the position of these lines from one spectrum to the next, it is possible to determine the relative shifts and correct for them, so as to align — also called *registering* — all the data onto a common reference.

In practice, this means telling `specINTI` which lines to use as a reference. This gives the following configuration file, similar to the previous one but with a few differences that we shall detail:

```
working_path: C:/tse_20260415
batch_name: deltauma
calib_mode: 1
line_type: absorption
calib_coef: [9.9897e-06, 1.8234, 2079.2709]
instrumental_response: _rep_20260415_deltauma
wavelength: [4861.34, 6562.81]
line_pos: [1513, 2428]
xlimit: [950, 2500]
slant: 0
crop_wave: [3600, 7500]
norm_wave: [6620, 6640]
```

The `line_type` parameter changes from `none` to `absorption`, indicating that a spectral realignment must be applied based on the absorption lines present in the object’s spectrum.

The `wavelength` parameter takes the list of wavelengths (in angstroms) of the chosen lines. A single line may be enough, but using several lines improves the accuracy of the realignment. In this example, we use the $H\beta$ (4861.34 Å) and $H\alpha$ (6562.81 Å) lines.

The `line_pos` parameter contains the pixel positions of these same lines in the first image of the sequence. These positions can be read off from the “Image” tab using the mouse pointer. A high degree of accuracy is not required: discrepancies of the order of 10 pixels are automatically corrected by the software. Of course, the correspondence between positions and wavelengths must be kept in matching order in the `wavelength` and `line_pos` lists.

The `slant` parameter corresponds to the tilt angle of the spectral lines with respect to the vertical of the image. In the absence of specific information, the spatial axis is assumed to be perfectly vertical, hence the value 0 in this example. In some cases, this angle can be estimated

automatically (in **lateral** or **external** mode, see below).

Once the configuration has been updated, click “Go!”. In this example, the differences compared with the previous processing may seem minor, but the feature described here proves crucial when thermo-elastic deformation of the instrument is significant.

Technically, the software adjusts the constant term (c) of the spectral calibration polynomial (see Section 3.2), which amounts to a translation along the wavelength axis. The higher-order coefficients (a , b) are not very sensitive to thermal and mechanical variations; they can therefore be treated as instrumental constants.

6.5.2 `line_type: emission`

The realignment method described in the previous section applies in a similar way when the spectrum of the object observed shows a well-marked emission line of known wavelength, as in the example shown in Figure 68.

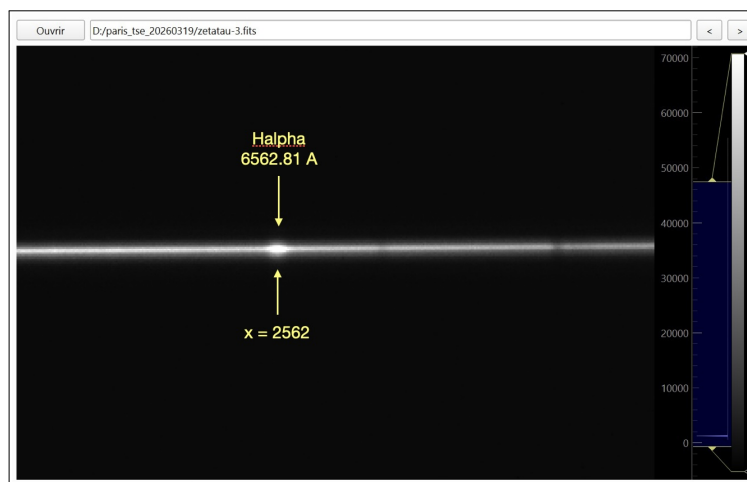


Figure 68 – Example of a low-resolution spectrum (the Be star ζ Tau) showing a marked emission line. This can be used as a reference point for spectral realignment via `line_type: emission`.

The only change needed in the configuration file concerns the `line_type` parameter, which here takes the value `emission`. You can use one or several emission lines whose wavelength(s) you know.

6.5.3 `line_type: external`

It can happen that no usable line is present in the spectrum of the object observed, owing to its nature (a supernova or quasar spectrum, for example). It then becomes necessary to resort to an artificial calibration source, external to the object.

Figure 69 shows the use of a small neon night-light as an auxiliary calibration source. When this gas is excited by an electric current, it emits a spectrum of lines at well-known wavelengths, whose values can be found in tables. See also the “A small atlas of neon spectral lines” box.



Figure 69 – A neon night-light waved in front of the telescope aperture to obtain a spectrum of wavelength-calibration lines in an image separate from the one being calibrated (a stellar spectrum, typically).

In the dataset supplied for the low-resolution processing examples, you will find, in addition to the images `deltauma-1`, `deltauma-2` and `deltauma-3`, an image `deltauma_neon-1`.

This last image was acquired immediately after the observations of the star, by waving a neon night-light in front of the telescope aperture during an exposure of about ten seconds, manually, “the old-fashioned way”, as shown in Figure 69. The star was then moved out of the slit (a small nudge on the telescope’s hand controller), so as not to risk mixing the star’s spectrum with that of the calibration lamp.

Let us emphasise the word “waved”: this means sweeping the lamp across the telescope aperture during the exposure (a few seconds) so as to illuminate it uniformly, which tends to improve the fidelity of the calibration⁶.

You can view the neon spectrum in the usual way, as shown in Figure 70.

6. The light from the calibration source then follows an optical path as close as possible to that of the star’s light, which is ideal for avoiding certain measurement biases.

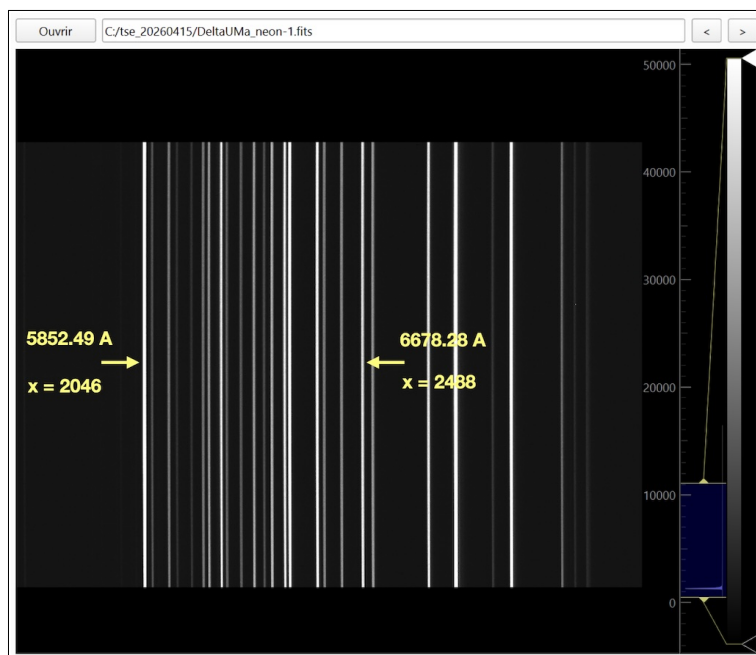


Figure 70 – The characteristic spectrum of a neon lamp. The lines are mainly concentrated in the red part of the spectrum.

In this image, two lines have been identified and their position (the x coordinate) roughly read off using the mouse pointer. This information will be used shortly.

To take this reference spectrum into account, it must be declared in the observation file, as shown in Figure 71.

Figure 71 – Declaring the neon calibration image in the observation panel. Several calibration images can be used (before and/or after the observation). The “Autofill” button fills in the fields automatically if the files are correctly named.

The configuration file then reads as follows:

```
working_path: C:/tse_20260415
batch_name: deltauma
calib_mode: 1
line_type: external
calib_coef: [9.9897e-06, 1.8234, 2079.2709]
```

```

instrumental_response: _rep_20260415_deltauma
xlimit: [1200, 2500]
wavelength: [5852.49, 6678.28]
line_pos: [2046, 2488]
search_wide: 20
crop_wave: [3600, 7500]
norm_wave: [6620, 6640]

```

The **line_type** parameter here takes the value **external**, indicating that calibration relies on an external, artificial source. The **wavelength** and **line_pos** parameters respectively hold the wavelengths (in angstroms) and positions (in pixels) of the selected neon lines. One or several lines can be used as needed.

The optional **search_wide** parameter is used to adjust the search tolerance for the lines within the image (by default ± 10 pixels). Narrowing this zone can help resolve certain ambiguities when several lines lie close to one another.

Note that if you do not specify a value for the **slant** parameter (the tilt angle of the lines), it is automatically computed from the line(s) produced by the spectral lamp and identified as the argument of the **line_pos** parameter.

Processing is then carried out as usual, by clicking “Go!”.

A small neon spectral atlas

Neon-gas night-lights, which are easy to find, produce an emission spectrum that is very convenient for calibrating spectrographs and for helping to align them (see also https://buil.astrosurf.com/specinti/Atlas_Spectral_Lamps.pdf). The following few views should help you find your way around this spectrum.

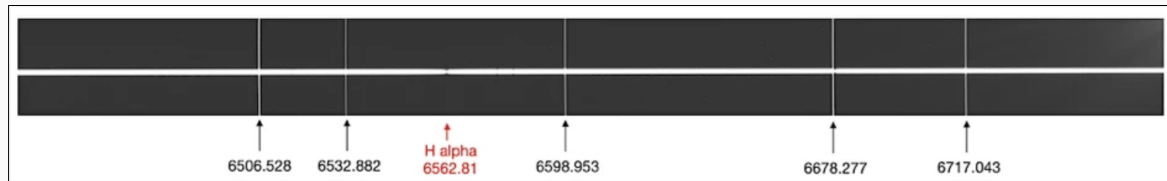


Figure 72 – Zoom on the neon spectrum around the $H\alpha$ line, in a spectrum acquired in **lateral** mode (the science target’s spectrum and that of the calibration source both appear in the same image). Spectrum taken at high resolution with a Star’Ex HR spectrograph.

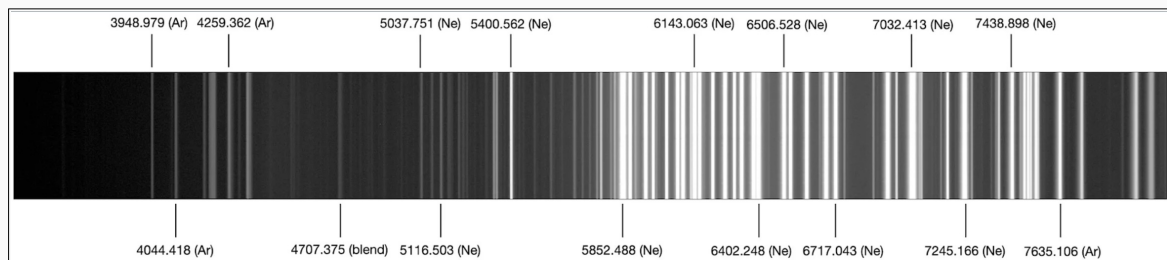


Figure 73 – A wider view of the neon spectrum around the $H\alpha$ line, taken with a Star’Ex LR spectrograph. Note the presence of argon lines, mixed in with those of neon.

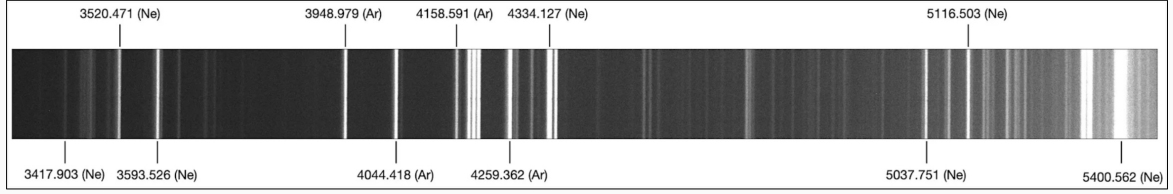


Figure 74 – The neon spectrum covering the blue and ultraviolet region, with argon lines added, this gas also being present in the lamp’s bulb, giving lines useful for calibrating this spectral region. The spectrum was taken with a UVEX4 spectrograph fitted with a 600 lines/mm grating (<https://spectro-uvex.tech/>), a particularly capable instrument for exploring the ultraviolet, owing to a design made almost entirely of non-absorbing mirrors and free of chromatism.

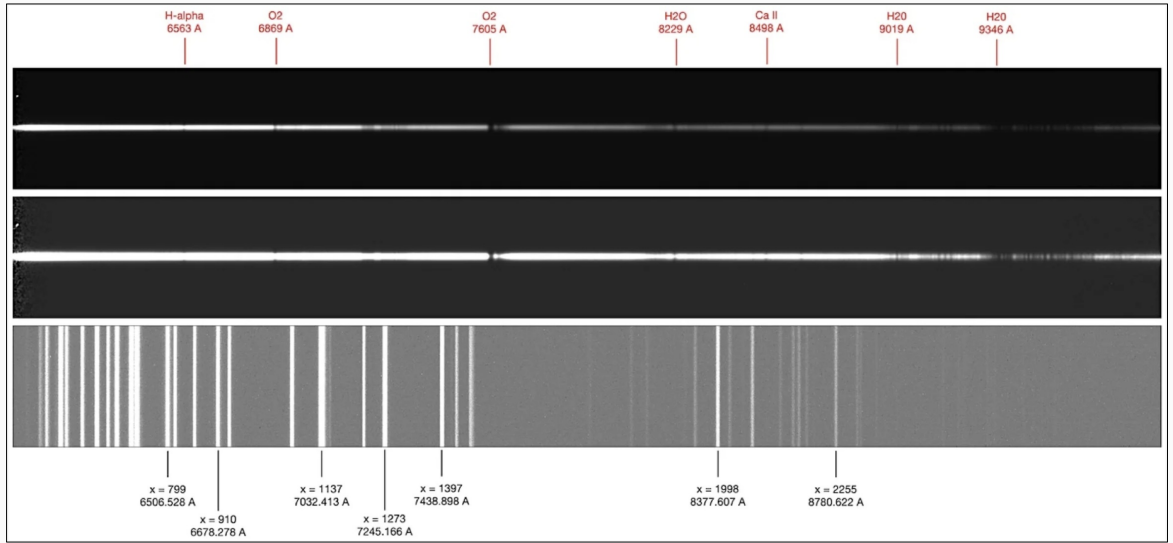


Figure 75 – At top, shown at two different contrast levels, the infrared spectrum of the star HD 4915, close to the solar spectral type. At bottom, the corresponding infrared spectrum of neon. Star’Ex LR spectrograph (fitted with a grating blazed at 1000 nm and a long-pass order filter, used to reject the blue part of the second-order spectrum).

6.5.4 line_type: lateral

Let us end this presentation of advanced calibration features with an example concerning a much fainter object than the star δ UMa: the supernova SN2026fvx in the galaxy NGC 4205, of magnitude 13.5 or so at the time of the observation. The equipment used is identical to that used for δ UMa, the two objects having been observed during the same night (the supernova data were downloaded at the same time as the δ UMa data).

We shall process four images of the spectrum of this supernova, each exposed for 30 minutes, giving a total integration time of two hours, which justifies the use of 1800-second dark frames, whose exposure time is at least equal to that of the individual sub-exposures. Figure 76 shows the content of the observation file.

Répertoire
C:/tse_20260415

Obj Nuit
Autofill
Effacer
Dir

Noms objets : SN2026fvx

Nom images : SN2026fvx-

Nb Images : 4

Image calib : SN2026fvx_neon-

Nb Img calib : -1

Trans Atm : None

Décalage Flat : 0

Offset : o- Nb 4

Dark : n1800- Nb 3

Flat : None Nb -1

Image postfix : -

Séparateur nom : Aucun

Calibration prefix :

Calibration postfix : _neon-

Enregistrer
sn2026fvx

Figure 76 – The observation file for the supernova SN2026fvx.

We reuse the calibration parameters established from the observation of the star δ UMa.

To refine the spectral calibration, we now make use of a natural source whose signature is almost always present in low-resolution spectra acquired with long exposures: the green line of atomic oxygen from the airglow, at 5577 Å. This atmospheric emission is produced at an altitude of about 95 km, in the Earth's upper atmosphere, and is clearly visible in the raw spectra (see Figure 77).

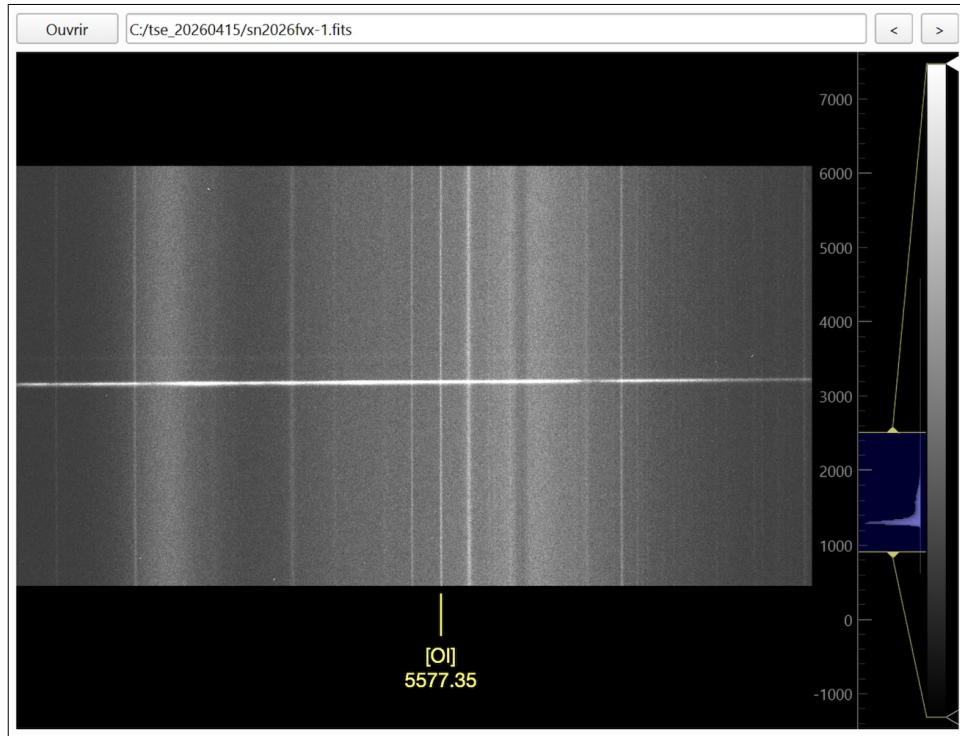


Figure 77 – Spectral trace of the supernova SN2026fvx in a raw 30-minute exposure. The sky background is affected by light pollution from Antibes; note in particular the natural [O I] airglow line at $5577.34 \text{ \AA}$, which we use here (two other, fainter, atmospheric-oxygen lines are present at wavelengths $6300.30 \text{ \AA}$ and $6363.78 \text{ \AA}$).

For this reduction, the `line_type` parameter takes the value `lateral`. This mode makes use of a reference source present simultaneously in the object’s spectral image: `specINTI` uses the information located on either side of the main spectral trace to extract from it a spectrum representative of the sky background and of the calibration source, hence the term “lateral”.

This approach has one important advantage: the sky spectrum (airglow, light pollution, and so on) is acquired at the same time as that of the object, which increases the accuracy of the calibration. This unwanted spectrum is then automatically subtracted during processing.

Here is the corresponding configuration file:

```
working_path: C:/tse_20260415
batch_name: sn2026fvx
calib_mode: 1
line_type: lateral
calib_coef: [9.9897e-06, 1.8234, 2079.2709]
instrumental_response: _rep_20260415_deltauma
wavelength: [5577.34]
line_pos: [1896]
xlimit: [1200, 2500]
search_wide: 20
norm_wave: [6620, 6640]
crop_wave: [3600, 7500]
```

We are now on familiar ground. Few parameters change; the main points to note are:

- the `lateral` value assigned to the `line_type` parameter;
- the new values of the `wavelength` and `line_pos` parameters, corresponding to the green airglow line;
- the use of a single reference line, generally sufficient in this context.

Figure 78 shows the spectral profile obtained, typical of a type-Ia supernova.

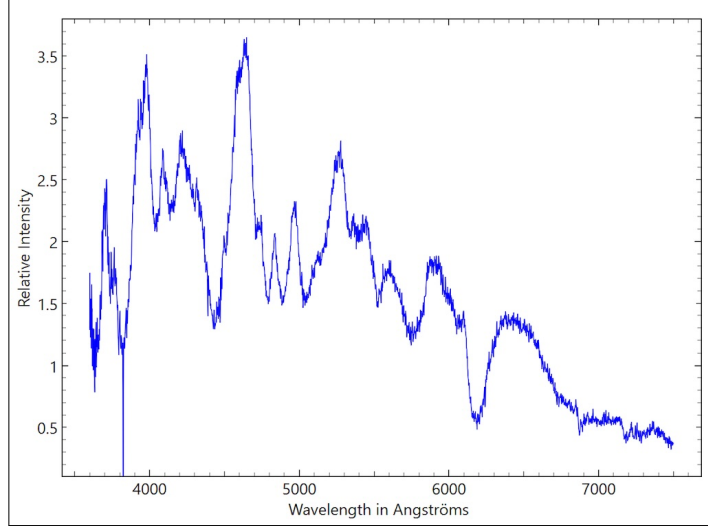


Figure 78 – Spectrum of the supernova SN2026fvx. 200 mm f/5 Newtonian telescope, TinyStar’Ex spectrograph fitted with a 14 μm slit and a Player One Uranus M Pro camera.

It is possible to go still further by enabling several advanced processing options, leading to what might be called “optimal processing” with **specINTI**. The following parameters are then added to those already present in the configuration file:

```
sky_mode: 1
kernel_size: -3
extract_mode: 1
sigma_gauss: 2
```

Thus:

- **sky_mode: 1** improves the estimation of the sky background and effectively reduces residual cosmetic defects (the default value is 0);
- **kernel_size: -3** applies a lightly destructive but effective median filter against impulsive noise;
- **extract_mode: 1** enables optimal extraction of the spectral profile, preferable to a simple summation of the rows of the 2D image;
- **sigma_gauss: 2** applies Gaussian smoothing to the spectral profile so as to reduce noise, at the cost of a slight reduction in spectral resolution, which drops from about $R = 740$ to $R = 480$ — still amply sufficient for studying supernovae.

Figure 79 shows the result obtained after applying these advanced processing options.

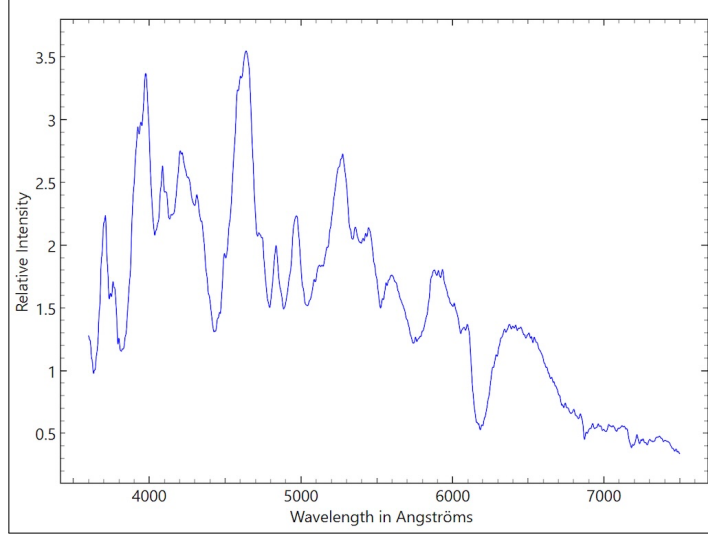


Figure 79 – Spectrum of the supernova SN2026fvx after optimised processing.

Let us finally point out that urban light pollution can, paradoxically, become an ally. Certain lighting sources do in fact produce artificial spectral lines that make excellent calibration markers, making it easy to use lateral mode without the need to build a dedicated calibration system. The urban environment here provides, free of charge, what others must painstakingly set up, as illustrated in Figure 80.

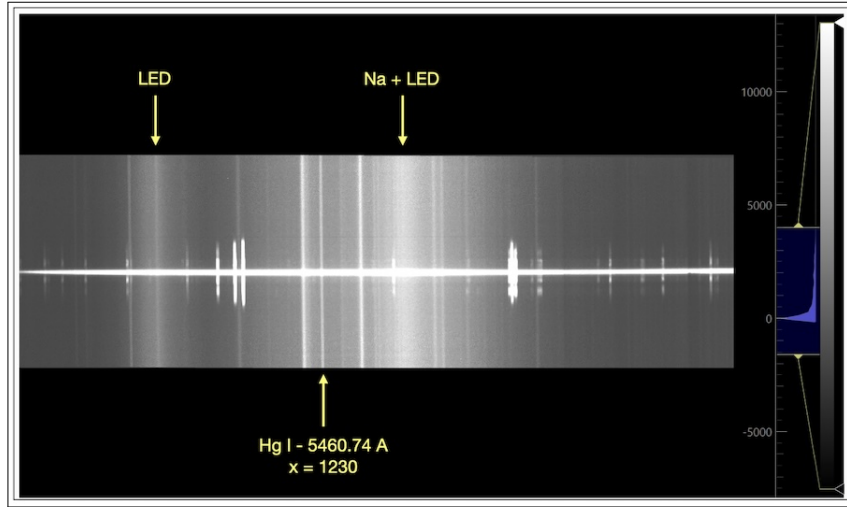


Figure 80 – Low-resolution spectrum of the planetary nebula NGC 2392 acquired in central Paris, heavily affected by urban lighting. This light pollution nevertheless allows accurate spectral realignment in lateral mode (`line_type: lateral`) thanks to a mercury line at 5460.74 Å. The natural airglow is difficult to detect here owing to the high sky brightness, which by no means prevents good-quality spectroscopy from being carried out.

As with “external” mode, if you do not specify a value for the `slant` parameter (the tilt angle of the lines), it is automatically computed from the designated airglow or light-pollution line(s).

Let us recall that it is entirely possible to briefly illuminate the telescope aperture, while observing the target, with a spectral lamp as described above, either for a set duration or simply held by hand, in order to obtain a well-exposed calibration spectrum in lateral mode. Simply take care not to saturate the calibration lines.

The examples above show that the spectrum of the target object can be contaminated by light coming from the sky background, whether natural or artificial. The software carries out two important operations to reduce the impact of this contamination: (1) binning (the summing of the signal by column) is performed over a restricted zone around the trace of the star’s spectrum;

(2) the sky-background signal is measured independently of that of the star, above and below it, before being subtracted from the spectral images. By default, `specINTI` carries out these operations according to its own criteria, based on an analysis of your data: you therefore do not need to worry about it.

In certain circumstances, however, it becomes necessary to regain control of the binning width and of how the sky-background computation zones are defined. The figure 81 shows a typical example: the observation of a comet.



Figure 81 – Image of the spectrum of the comet C/2022 E3 (ZTF), taken on 31 January 2023 with an 80 mm-diameter refractor and a Star’Ex LR fitted with a 600 lines/mm grating. We do not want to include, in the sky-background computation, light coming from the comet’s coma (so the sky-background computation zones must be moved away from the object’s zone of influence). We also want to extract only the spectrum of the comet’s nucleus (so a narrow binning zone must be chosen). This example illustrates the need for manual adjustment of the software’s parameters (the `bin_size` and `sky` parameters, see text).

The binning height, in pixels, is controlled by the parameter `bin_size`, to be added to the configuration file. For example, `bin_size: 20` forces the software to bin over a height of 20 pixels centred on the spectral trace.

The zones used to compute the sky-background signal are controlled by means of the `sky` parameter. The syntax is `sky: [y1, y2, y3, y4]`, where `y1`, `y2`, `y3` and `y4` are pixel coordinates relative to the spectral trace, as defined in the diagram in Figure 82.

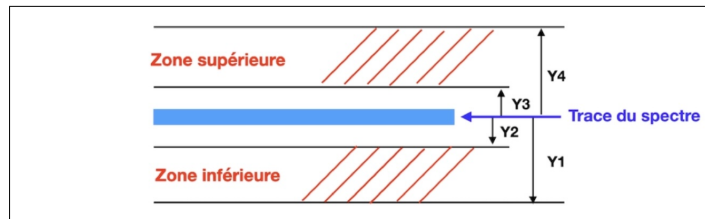


Figure 82 – Defining the sky-background computation zones.

Example: `sky: [220, 80, 80, 220]` defines two symmetric zones on either side of the trace, each $220 - 80 = 140$ pixels wide.

Tip: it can be useful to check that the values of the `bin_size` and `sky` parameters have been chosen correctly. You can get a graphical view of your choices by adding the line `check_mode: 1` to the configuration file, then examining the image “`_step101.fits`”, automatically created in the working directory. Your binning zones and sky-background computation zones are shown there as lines overlaid on the processed 2D image of your spectrum.

6.6 Accounting for atmospheric transmission

In the previous sections, we deliberately ignored a factor of some importance for determining the instrumental response: the spectral attenuation caused by the Earth’s atmosphere, which does in fact appear explicitly in the flux-calibration equation (see Section 3.3).

This has an immediate practical consequence: to obtain a rigorous spectral response, it is normally necessary to observe a reference star close to the science target, so that the two objects are seen through a similar airmass (that is, through a comparable thickness of atmosphere). See Section 5.3 for a detailed discussion of this point.

This well-established method, however, has several drawbacks: it is relatively demanding, sometimes tedious, and mechanically reduces the observing time available for science targets.

In the previous examples, the observations of δ UMa (the reference) and of the supernova SN2026fvx are deliberately decoupled in terms of atmospheric transmission: the airmasses are different, and the data could even have been acquired on separate nights. To make these observations compatible, it becomes necessary to estimate the atmospheric transmission $T_a(\lambda)$ in the direction of each object observed.

The idea is to explicitly separate the contributions of the atmosphere and of the instrument. Recall that the intrinsic instrumental response can then be written as:

$$\mathcal{R}(\lambda) = \frac{S_{\text{raw}}(\lambda)}{S_{\text{true}}(\lambda) T_a(\lambda)} \quad (1)$$

This response thus becomes independent of the Earth’s atmosphere: it is the “true” instrumental response, the one that would be measured from space.

This approach differs fundamentally from the classical method, which consists in systematically observing a pair of objects, a reference star and a science target. Here, once the instrument’s intrinsic response has been determined (outside the atmosphere), it need only be combined with the atmospheric transmission specific to the observing conditions in order to reconstruct the full apparent response of the instrument + atmosphere system.

The whole difficulty lies in correctly evaluating $T_a(\lambda)$. Fortunately, as soon as one is interested only in its slow variations with wavelength — which is sufficient here — the calculation is simple and accurate⁷.

`specINTI` implements this calculation directly. Atmospheric transmission depends mainly on two parameters:

- the altitude of the star above the horizon, which determines the airmass traversed;
- the amount of aerosols present in the atmosphere, quantified by the AOD (*Aerosol Optical Depth*, see the box below).

7. The atmospheric transmission computed by `specINTI` includes the contributions of Rayleigh scattering (molecular), Mie scattering (aerosols), and ozone absorption. Absorption due to water vapour and molecular oxygen is not taken into account: modelling it is more complex, and it is not needed for estimating the overall instrumental response.

AOD (*Aerosol Optical Depth*)

The AOD quantifies the presence of aerosols (dust, droplets, pollution particles) in the atmosphere. These aerosols are responsible for an attenuation of the light coming from celestial objects.

A simple way to estimate the AOD is to relate it to the horizontal visibility V (in kilometres), that is, the maximum distance at which an object can be distinguished with the naked eye. An empirical relation derived from the work of Koschmieder (1924) and Elterman (1968) is written:

$$\text{AOD} = \frac{3}{V} \left(1 - \frac{V}{205} \right) H \quad (2)$$

where H is the aerosol scale height (typically $H \approx 1.5$ km).

The factor of 3 corresponds to the minimum perceptible contrast threshold ($C_{\min} \approx 5\%$), i.e. $\ln(1/C_{\min}) \approx 3$. The value of 205 km represents the theoretical maximum visibility in a purely molecular atmosphere (Rayleigh scattering alone).

A few useful orders of magnitude:

- AOD ≈ 0.05 to 0.10 : a very clear atmosphere;
- AOD ≈ 0.13 : average conditions at a rural site;
- AOD ≈ 0.20 : a slightly hazy atmosphere;
- AOD > 0.5 : degraded conditions (haze, pollution).

In practice, if this correction is enabled in the configuration file, it is applied automatically during processing. The software computes the star’s altitude and the corresponding airmass from the object’s equatorial coordinates. The user only needs to supply an estimate of the AOD via the `corr_atmo` parameter. A value of 0.13 is generally a satisfactory default choice.

It is, of course, essential to remain perfectly consistent: if the intrinsic instrumental response was computed outside the atmosphere from the reference star (δ UMa in our examples), the atmospheric correction must also be applied to the science targets — here, the supernova SN2026fvx.

All of these operations are automated within `specINTI`. It is enough to add the necessary parameters to the configuration files once and for all.

Here, for example, is the file used to compute the intrinsic instrumental response in mode `-1`:

```
working_path: C:/tse_20260415
batch_name: deltauma
calib_mode: -1
line_type: hot
calib_file: p_A2V
xlimit: [950, 2500]
crop_wave: [3500, 7500]
# -----
corr_atmo: 0.13
simbad: 1
Longitude: 7.093
Latitude: 43.587
Altitude: 50
```

The `corr_atmo` parameter specifies the AOD value. As soon as it is defined, atmospheric transmission is included in the computation of the instrumental response.

When `simbad: 1`, the software automatically queries the SIMBAD (CDS) database to obtain the equatorial coordinates of the observed object (hence the importance of naming it correctly). Failing that, these coordinates are looked for in the header of the FITS image files, if they are present there under the standard keywords for this format.

The geographic coordinates of the observing site must also be supplied (in degrees) via the parameters `Longitude` (positive east of Greenwich), `Latitude`, and `Altitude` (in metres).

Here now is an example of a configuration file that makes use of this “true”, atmosphere-free, instrumental response when processing the supernova SN2026fvx:

```

working_path: C:/tse_20260415
batch_name: sn2026fvx
calib_mode: 1
line_type: lateral
calib_coef: [9.9897e-06, 1.8234, 2079.2709]
instrumental_response: _rep_20260415_deltauma
wavelength: [5577.34]
line_pos: [1896]
xlimit: [1200, 2500]
search_wide: 20
norm_wave: [6620, 6640]
crop_wave: [3600, 7500]
# -----
sky_mode: 1
kernel_size: -3
extract_mode: 1
sigma_gauss: 2
# -----
corr_atmo: 0.13
near_star: NGC 4605
simbad: 1
Longitude: 7.093
Latitude: 43.587
Altitude: 50

```

It can happen that the object observed is not listed in SIMBAD (novae, comets, and so on). The `near_star` parameter can then be used to designate a nearby object on the celestial sphere that is listed in SIMBAD, whose coordinates are used for computing the atmospheric transmission.

In the example above, we use:

```
near_star: NGC 4605
```

Since NGC 4605 is the host galaxy of the supernova, its coordinates provide an excellent approximation of those of the object. Be careful, however: if you reuse this configuration file for a different target, remember to remove or comment out this line.

This method of accounting for atmospheric transmission proves effective and accurate enough in the vast majority of practical situations. We use it routinely for our own observations.

Let us close with one final piece of advice. Your observations may eventually contribute to collaborative pro-am scientific work, or feed databases such as BeSS for Be stars. In this context, it is essential to correctly identify the observer and the instrument in the FITS headers of the spectra produced. Simply add the following parameters to the configuration file (adapting them, of course, to your own case):

```

site: Antibes Saint-Jean
inst: T200 + tinystarex + PL585
observer: cbuil

```

In doing so you will produce properly documented data that can be readily used by the scientific community.

6.7 Revisiting the calibration technique: the case of a cool star

In Section 6.2 we showed how to calibrate a spectrograph at low resolution using the spectrum of a hot star alone, typically of spectral type A. This observation makes it possible to derive both the wavelength calibration law and the instrumental response.

specINTI can also achieve an equivalent result from the spectrum of a “cool” star. Everything is relative when one speaks of a cool object: here we mean a solar-type star (more precisely of type G2V), whose surface temperature is nevertheless close to 5,000 K!

We shall illustrate this capability of the software with a very particular observation: that of the closest star of all, the Sun. This will also be an opportunity to present some further features of specINTI.

The exposure is therefore taken in broad daylight, under familiar conditions and without any telescope whatsoever. All that is needed is a sunny day, free of cloud, and pointing the entrance of the spectrograph at a white screen diffusing the sunlight, as shown in Figure 83. The screen faces the Sun.

If the screen is taken to be spectrally neutral, the light entering the spectrograph is indeed that of a star of spectral type G2 V. We shall not go into the nature of the screen here, but a sheet of foam board, commonly used for drawing, makes a perfectly suitable white diffuser (only the deep blue shows a slight distortion). More neutral laboratory reference reflectors exist (Spectralon), but they are considerably more expensive.



Figure 83 – Measuring the calibration parameters of a spectrograph, here a Star’Ex LR.

The spectral distribution of the Sun is well known, and the wavelengths of its many Fraunhofer lines are likewise established with great precision. We therefore have available a natural reference source of high quality.

Figure 84 shows the image of the solar spectrum acquired under these conditions, with a TinyStar’Ex spectrograph fitted with a 14 μm slit, the entrance beam being stopped down to the aperture used when observing stars at the telescope (here f/6).

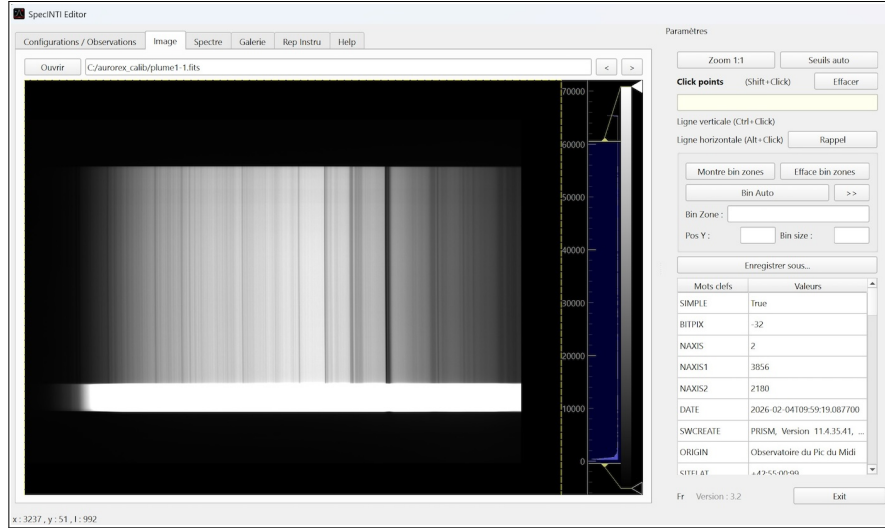


Figure 84 – Image of the solar spectrum acquired with a TSE spectrograph fitted with a Sony IMX585 sensor camera (Player One Uranus M Pro model). The bright band in the lower part corresponds to the spectrum acquired in the photometric zone of a GEN2-type slit ($140\ \mu\text{m}$ wide, as against $14\ \mu\text{m}$ in the part actually used here). The whole height of the slit is uniformly illuminated: in a sense, we are observing an extended object.

We need to account for atmospheric transmission in order to correctly compute the instrumental response outside the atmosphere. Two pieces of information are needed: the approximate altitude of the Sun above the horizon, and the value of the AOD at the time of the observation.

As for the dispersion law, when a cool star is involved (type G or K), **specINTI** proceeds very differently from the method used for a hot star: the technique now relies on a cross-correlation involving a large number of lines.

The reference spectrum used here is a Pickles profile corresponding to a star of spectral type G2 V, the Sun's own type.

The procedure is largely automated, but it nevertheless requires prior identification of two spectral lines whose wavelengths are known. This information allows the software to estimate an initial average spectral dispersion. Figure 85 illustrates this step.

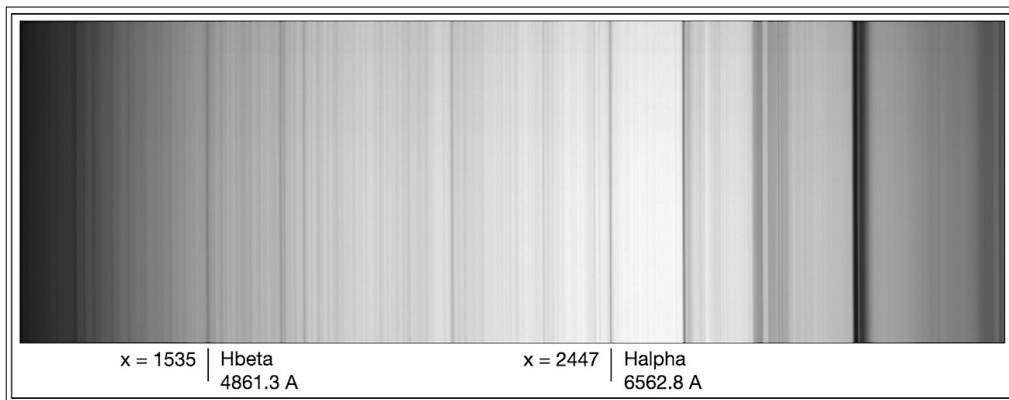


Figure 85 – Identifying the $H\alpha$ and $H\beta$ lines in the image of the solar spectrum observed in reflection off a white screen.

For example, the following coordinates are read off: $x = 1535$ for the $H\beta$ line and $x = 2447$ for the $H\alpha$ line. The altitude of the Sun above the horizon is estimated at 25° (using an ephemeris, or by measuring the shadow cast by a stick planted vertically in the ground — the old-fashioned way!), and the AOD is taken to be 0.13.

Listing 3 details, line by line, the configuration file used to calibrate the spectrograph from a

daytime observation of the Sun.

Listing 3 – Calibration from the solar spectrum.

```
working_path: C:/aurorex_calib -> working directory
batch_name: plume -> name of the observation file
calib_mode: -1 -> automatic calibration mode from a natural spectrum
line_type: cold -> solar-type source, realignment by cross-correlation on absorption lines
wide_source: 1 -> indicates that the spectrum is very extended along the spatial axis
posy: 1100 -> vertical coordinate of the centre of the binning zone
bin_size: 400 -> width of the binning zone
calib_file: p_G2V -> Pickles reference spectrum (G2V star)
corr_atmo2: [0.13, 24.5] -> atmospheric correction: AOD = 0.13, altitude 24.5°
wavelength: [4861.3, 6562.8] -> wavelengths of the reference lines
line_pos: [1535, 2447] -> pixel positions of the reference lines
poly_order: 2 -> degree of the dispersion polynomial
compute_wave: [3900, 7000] -> spectral range used for the correlation
exclude_zones: [(5860, 5975), (6840, 7040), (7140, 7370), (7570, 7710)] -> excluded
zones (telluric lines)
slant: 0.25 -> tilt angle of the lines, in degrees
crop_wave: [3650, 7500] -> spectral range kept for the instrumental response
norm_wave: [6620, 6640] -> normalisation interval for the response
sigma_resp: 5 -> smoothing coefficient for the instrumental response
```

Remarks:

The coordinates of the reference lines are given in Figure 85. The value of the *slant* angle comes from examining how vertical the lines are in the check images `_step9.fits` or `_step10.fits`, produced by the software when the `check_mode` parameter is enabled (`check_mode: 1`).

The binning width used here (400 pixels) is deliberately large, so as to maximise the signal-to-noise ratio.

The `exclude_zones` parameter is used to exclude regions of the spectrum heavily affected by telluric molecular bands (H_2O and O_2). The number of excluded zones is unrestricted; within these intervals, the instrumental response is reconstructed by linear interpolation.

The `corr_atmo2` parameter is a variant of `corr_atmo`: in addition to the AOD value, it allows the altitude of the object above the horizon to be specified.

The spectral calibration coefficients are shown in the output console, as illustrated in Figure 86, while Figure 87 shows the computed spectral response.

The value 1 given to `wide_source` requires `posy` to be set by the user (the software cannot automatically compute the centre of the binning zone for a very extended source, so this becomes your responsibility). Sky-background subtraction is also not performed, since subtracting it would amount to removing the very data one is trying to measure. This behaviour does not occur (standard mode) if this parameter is absent from the configuration file, or if its value is 0.

```

-----
specINTi 2.6.0
Path = C:/aurorex_calib\
-----
Observation file: C:/aurorex_calib\plume.yaml
*****
Target: plume
*****
Offset processing...
Target processing...
Total exposure time = 1.2 seconds
Geometric correction...
Find trace vertical coordinate...
Median filtering...
Tilt correction...
Predefined tilt angle = 0.000
Slant correction...
Predefined slant angle = 0.250
Raw profile: C:/aurorex_calib\plume_raw.fits
Compute calibration functions...
Atmospheric transmission correction (corr_atmo2
parameter)...
AOD = 0.15
Angular elevation = 26.00 deg.
Normalize...
Calibration coefficients:
calib_coef: [6.7585e-06, 1.8370, 2026.9736]
Root Mean Square Error = 0.8438 A
Response file:
C:/aurorex_calib\response_plume.fits
End.
-----

```

Figure 86 – Progress of the calibration from the solar flux.

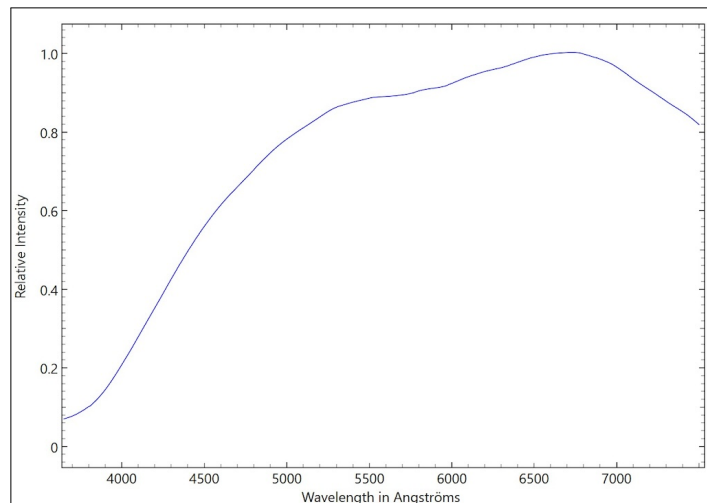


Figure 87 – Spectral response computed from the observation of the white screen.

7 The advanced interface: use in high spectral resolution

7.1 General points

The `calib_mode` parameter now takes the value 0:

```
calib_mode: 0
```

Two approaches are possible for spectral calibration:

- supplying a pair of data files consisting of the spectrum of the observed object and a reference spectrum from a calibration source producing emission lines at known wavelengths. A neon lamp, for example, is particularly well suited to studying the $H\alpha$ line at high resolution. This

calibration spectrum makes it possible to determine the coefficients (a, b, c) of the spectral calibration law. The corresponding procedure is activated via `line_type: external`;

- supplying a single spectrum acquired in *lateral mode*. In this case, the spectrum of the object and that of the calibration source are recorded simultaneously on the same image. The lines of the reference source thus overlap those of the object, which greatly limits the effects of instrumental drift that can arise when the acquisitions are separated in time. This approach is potentially the most accurate. Once extracted by the software, the reference spectrum is used to determine the coefficients (a, b, c) of the dispersion law. The calibration lines are then removed during processing, at the sky-background subtraction stage. This procedure is activated via `line_type: lateral`.

To illustrate the processing principles applied to high-resolution spectra in advanced mode, we shall use a sequence of 7 raw spectral images of the star Regulus (α Leo), of spectral type B8 IV, each exposed for 180 seconds. The data are acquired in lateral mode, unlike the example dealt with in Section 5.1, where the neon-lamp spectrum was separate from the spectra of this same star. The images, in FITS format, are named `regulus-1`, `regulus-2`... They cover the region of the hydrogen $H\alpha$ line.

We have also taken from this session a few DOF images:

- 4 offset images (`o-1`, `o-2`...);
- 6 dark images (`n900-1`, `n900-2`...);
- 5 flat-field images (`led-1`, `led-2`...), acquired with an LED panel (see Section 5.1 and Figure 88).

You can download all this data from this link: https://buil.astrosurf.com/specinti4/demo/starex_20250405.zip

The spectra were acquired with a Star'Ex HR spectrograph fitted with a 2400 lines/mm grating and a $23\ \mu\text{m}$ entrance slit. The camera is a Player One Uranus M Pro (Sony IMX585 sensor), mounted behind an Askar 107PHQ refractor of 107 mm aperture and 740 mm focal length ($F/D = 7$). The observations were made on 31 May 2025, from a suburban site (Antibes).



Figure 88 – Acquiring a *flat-field* using a household LED lighting panel. Although the LED spectrum is not very uniform across the whole visible range, using one remains sensible for calibrating high-resolution spectra around $H\alpha$. The image on the right shows the setup mounted in front of the Askar 107PHQ refractor; on the left, in front of a venerable orange Celestron 8.

Figure 89 shows the spectral-calibration setup used. A plastic optical fibre of 3 mm diameter

continuously injects, at the centre of the refractor's objective, the light from a neon lamp placed at its other end (fibre length: about 400 mm).

Two spectra are thus acquired simultaneously:

- the spectrum of the celestial object;
- the spectrum of the calibration lamp, which illuminates the whole height of the slit.

The calibration system is entirely static (no moving parts, no mechanism).



Figure 89 – Continuous injection of light from a calibration source (here a neon lamp) into the entrance pupil of the instrument in **lateral mode**. A plastic optical fibre of 3 mm diameter is used. It is best to avoid too point-like a source, so as to limit *speckle* effects in the image of the lines — hence the relatively large fibre diameter. The resulting obstruction remains negligible and does not significantly affect the throughput of the instrument.

The image in Figure 90 shows one of the spectra in the Regulus sequence, with the monochromatic lines of the calibration spectrum overlaid.

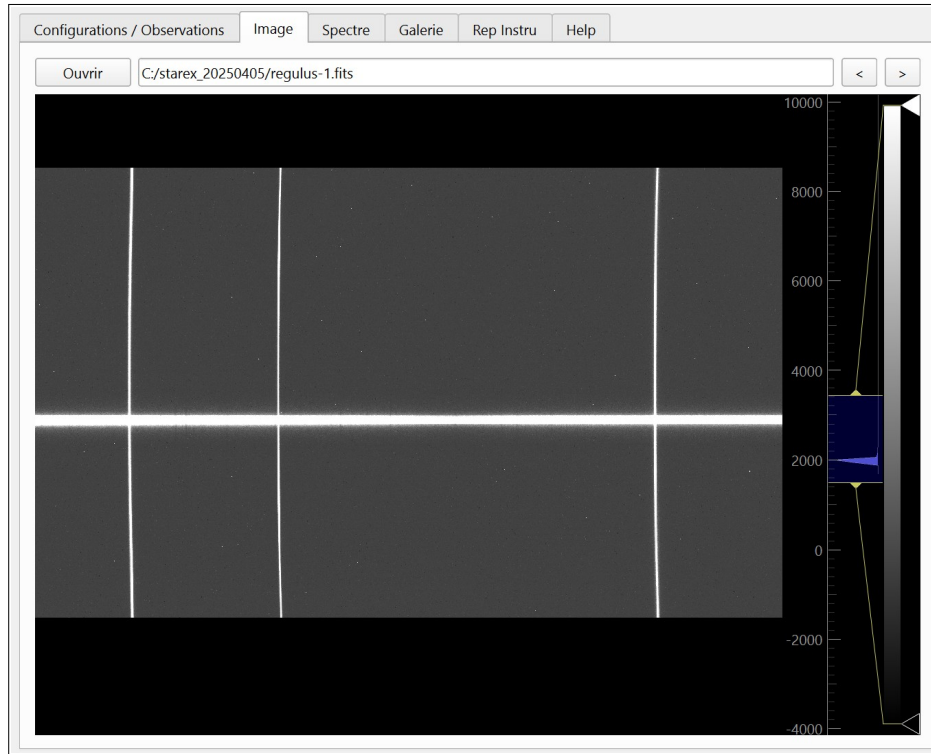


Figure 90 – Horizontal trace of the Regulus spectrum, and, in the perpendicular direction, the lines of the neon source recorded simultaneously (*lateral mode*). Note the slight curvature of the lines, a classic geometric distortion known as the *smile* effect.

Figures 91 and 92 illustrate other possible configurations for acquiring the lateral spectrum.

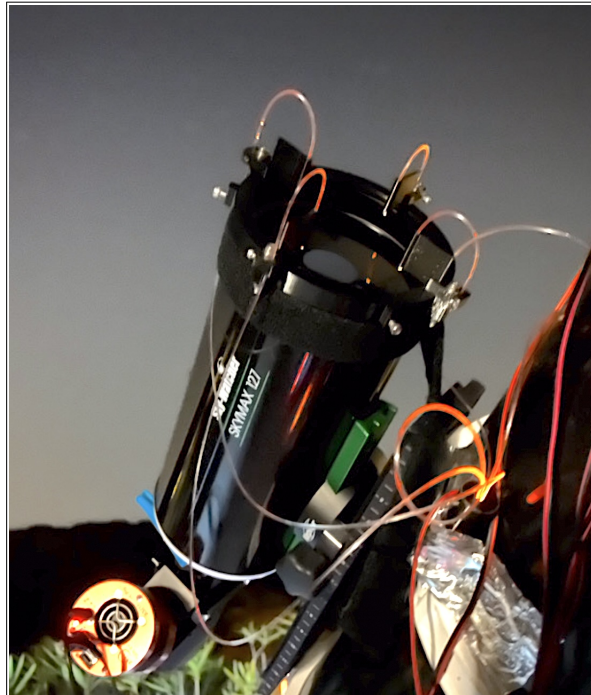


Figure 91 – A variant of lateral mode using several fibres distributed across the pupil. This configuration reduces how point-like the source is and can improve the accuracy of the spectral calibration.



Figure 92 – Another approach is to place the calibration lamp directly in front of the pupil, without using a fibre (the telescope here is a Celestron C9.25).

7.2 Evaluating the instrumental response

Computing the instrumental response is done in three steps:

1. carrying out a “complete” reduction of the spectrum of a reference star, generally of spectral type A, whose theoretical spectrum is known from catalogue data. The word “complete” means that all the usual corrections are applied (DOF, geometric corrections, profile extraction, spectral calibration, and so on), with the exception of the instrumental-response correction, still unknown at this stage;
2. dividing the spectrum obtained in the previous step by the theoretical reference spectrum. After cleaning and smoothing away residual irregularities due to computational imperfections or noise, the result is the instrument’s own spectral response;
3. as a check, the spectrum of the reference star is processed a second time, this time applying the instrumental response computed in the previous step. The resulting profile should be very close to the theoretical reference spectrum, which validates the whole procedure.

Refer to Table 1 to identify the reduction scenario best suited to your case. Here we are working at high spectral resolution, with **lateral mode**. The configuration file must therefore contain the line: `calib_mode: 0`, together with the line: `line_type: lateral`.

Before looking at the configuration file itself, let us look at the observation file and at the specINTI Editor interface, which generates it automatically (Figure 93).

Répertoire C:/starex_20250405

Obj Nuit Autofill Effacer Dir

Noms objets : regulus

Nom images : regulus-

Nb Images : 7

Image calib : regulus_neon-

Nb Img calib : -1

Trans Atm : None

Décalage Flat : 0

Offset : 0- Nb 4

Dark : n900- Nb 6

Flat : led- Nb 5

Image postfix : -

Séparateur nom : Aucun

Calibration prefix :

Calibration postfix : _neon-

Enregistrer regulus

Figure 93 – Observation parameters for the star Regulus.

In its minimal version, the configuration file looks as shown in Figure 94.

Recherche Ligne

```

working_path: C:/starex_20250405
batch_name: regulus
calib_mode: 0
line_type: lateral
auto_calib: [6490, 6722]
poly_order: 2
clean_wave: [6506.5, 6532.8, 6598.9, 6678.3]
clean_wide: [0.9, 0.9, 0.9, 0.9]
xlimit: [400, 1800]
smile_radius: -22000
crop_wave: [6500, 6700]
norm_wave: [6640, 6660]

```

Enregistrer mode0_HR_lateral

Figure 94 – Configuration parameters for a first extraction of the Regulus spectrum.

At this stage, the aim is to produce a first version of the star's spectrum, then to compare it with the expected theoretical spectrum of Regulus in order to derive the instrumental response.

Here is the detail of the configuration file:

Listing 4 – Configuration for high-resolution processing in lateral mode

```

working_path: C:/starex_20250405
batch_name: regulus
calib_mode: 0 -> high-resolution spectrum processing
line_type: lateral -> spectral calibration in lateral mode
auto_calib: [6490, 6722] -> automatic search range for the neon lines (Å)
poly_order: 2 -> degree of the dispersion polynomial
clean_wave: [6506.5, 6532.8, 6598.9, 6678.3] -> wavelengths of the lines to be cleaned (Å)
clean_wide: [0.9, 0.9, 0.9, 0.9] -> width of the cleaned zones (Å)
xlimit: [400, 1800] -> range used for the geometric calculations (pixels)
smile_radius: -16000 -> radius of curvature of the lines (smile effect, pixels)
crop_wave: [6500, 6700]
norm_wave: [6640, 6660]

```

By setting the `auto_calib` parameter, the software is instructed to automatically detect, for each image of the sequence, the neon lines within the given spectral range (here between 6490 and 6722 Å), to precisely measure their position to a fraction of a pixel, and then to compute the corresponding dispersion law.

The `poly_order` parameter sets the degree of the polynomial used to model the dispersion law. A degree of 2 is more than sufficient here, given the characteristics of the spectrograph used.

Despite careful removal of the sky background and of the lateral-spectrum lines, slight artefacts can remain at the points where the calibration lines cross the star's spectral trace. The optional `clean_wave` and `clean_wide` parameters can be used to automatically remove these residuals in the final spectrum, at the specified wavelengths and spectral widths.

The `smile_radius` parameter triggers correction of the curvature of the spectral lines (the *smile effect*), the radius of curvature being given in number of pixels.

With a Star'Ex HR spectrograph, the typical radius of curvature is of the order of 15,000 to 20,000 pixels depending on the spectral region observed (see again, if needed, the box devoted to *smile* in Section 5.1).

If you add the line `check_mode: 1` to the configuration file (the default value of this parameter being 0), the program runs in “check” mode. More information is then produced on the progress of the computation. In particular, examine the image `_step9.fits`, written to the working directory, which shows the appearance of the first image of the sequence after the geometric corrections have been applied. The neon lines should then appear perfectly vertical and straight (see Figure 95).

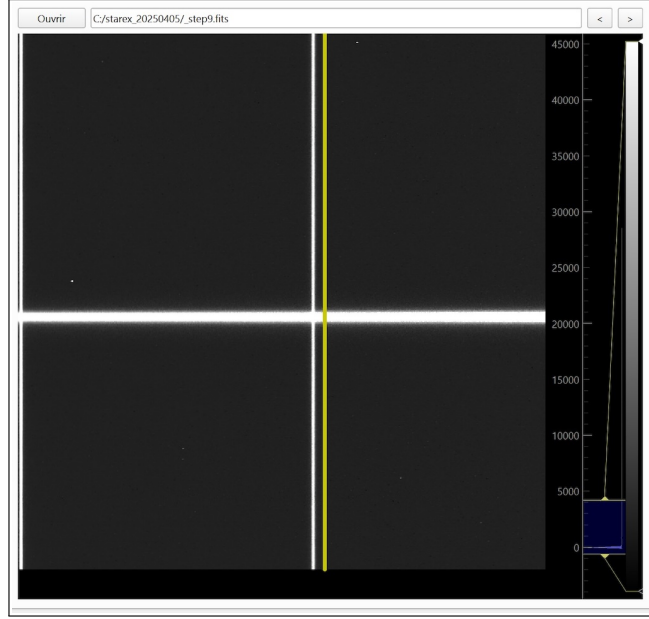


Figure 95 – Content of the `_step9` image. The geometric correction of the lines is nearly perfect here. To make visual checking easier, you can draw a vertical line in the image with a `Ctrl + click`. If the lines still show a slight curvature, adjust the value of `smile_radius` in the configuration file, then run the reduction again.

The spectrum of Regulus obtained with this configuration is computed within a few seconds of clicking “Go!”. The result is already very satisfactory, even though the instrumental response has not yet been applied.

We shall now evaluate this response. We need the reference spectrum of the star α Leo outside the atmosphere. The most convenient approach is to consult the Melchior's spectral library; see Section 5.1 for details.

From the “Instr. resp.” tab, load the reduced spectrum of Regulus, together with the reference spectrum from Melchior's (named `_ref_regulus.fits` once downloaded). The ratio of the two profiles is then displayed automatically (see Figure 96).

Recall that the strong slope visible in the ratio comes mainly from the LED panel used for the flat-field: despite the narrow spectral range studied, this source produces noticeably more signal on the blue side than on the red side. In the absence of a flat-field (which remains perfectly acceptable here), the ratio profile would have been much closer to the final profile.

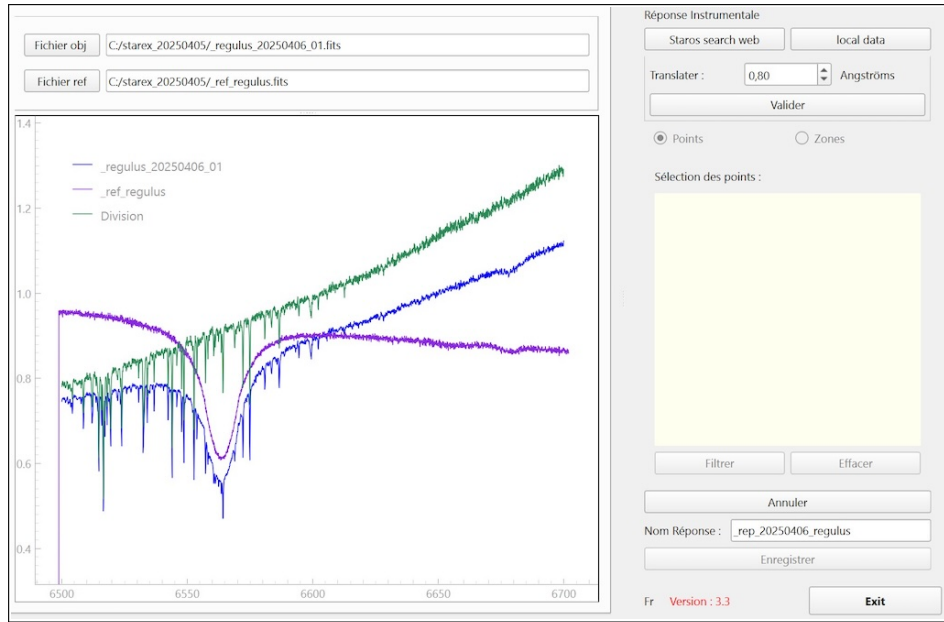


Figure 96 – In blue: the reduced spectrum of Regulus with no instrumental-response correction, showing a strong slope caused by the spectral distribution of the LED panel. In purple: the reference spectrum. In green: the ratio of the two profiles.

Let us go back over the procedure. Apply a spectral shift to compensate for any radial-velocity offset between the observed spectrum and the reference spectrum: this is the case if you see a marked bump at the H α line. In the example, the shift is +0.8 Å (the “Shift” field). Then click “Validate” so that only the ratio profile is displayed.

Then define a few points on this curve with mouse clicks, as shown in Figure 97. These points should be spread evenly along the continuum, avoiding the telluric lines. Ten to twenty points are usually enough.

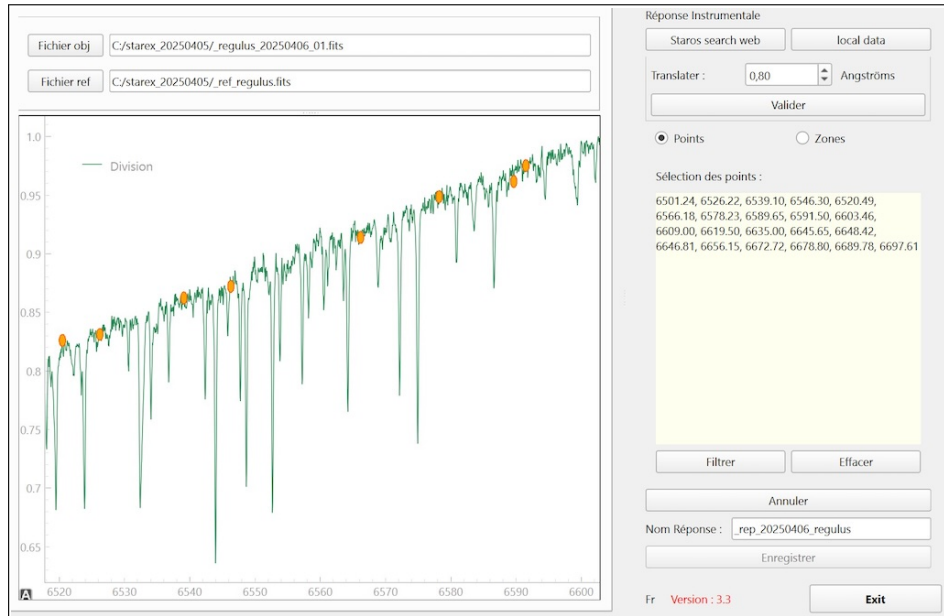


Figure 97 – Selecting the points (in red) used to define the continuum of the instrumental-response profile.

Click “Filter” to smooth the instrumental response and give it its final form (Figure 98). All that remains is to save the result in the working directory, for example under the name `_rep_20250406_regulus`.

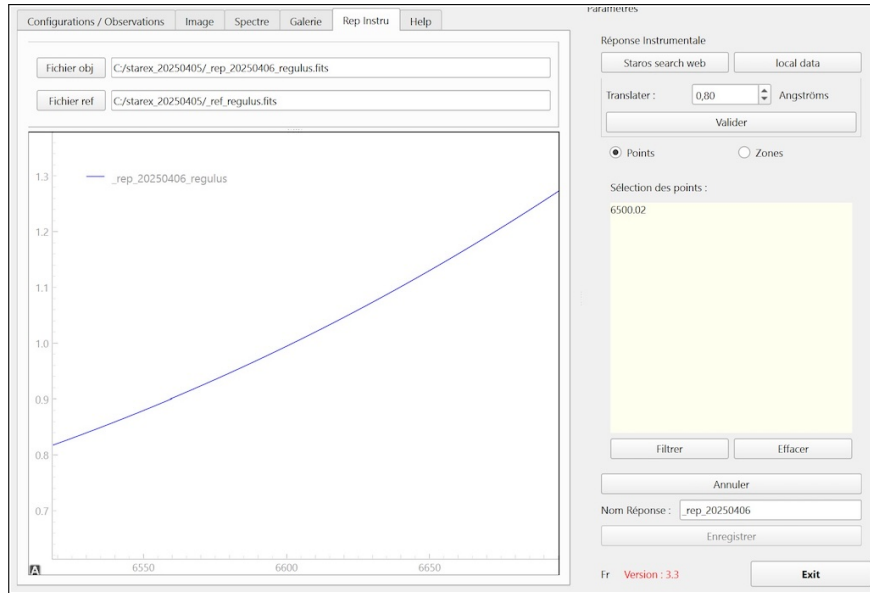


Figure 98 – The final shape of the instrumental response, here strongly influenced by the spectral distribution of the LED source used for the flat-field.

An alternative to LEDs for acquiring flat-field images is to use tungsten-filament bulbs. Their spectrum, close to that of a black body, is much more uniform and gives better coverage of the blue and infrared regions, which are out of reach for LEDs.

Where photometric rigour is the goal, LEDs remain acceptable at high resolution over a narrow spectral range, but filament lamps, of the halogen type, offer a more general solution (see Figure 99).



Figure 99 – Acquiring a *flat-field* using torches fitted with tungsten-filament bulbs (MagLite brand — take care not to use LED models). A diffuser (a sheet of tracing paper) should cover the entrance of the refractor. Recall that, within the processing scheme presented in this guide, obtaining such a flat-field (called “PLU” in French, for *Plage de Lumière Uniforme*, i.e. a uniform-illumination field) remains optional, not compulsory. The purpose of this flat-field is essentially to smooth out pixel-to-pixel sensitivity differences (PRNU = *Pixel Response Non-Uniformity*), which are very slight with modern sensors.

7.3 Routine processing of high-resolution spectra

To properly reduce the spectrum of the star Regulus, or of any other star, simply add the following line to the configuration file:

```
instrumental_response: _rep_20250406_regulus
```

then run the reduction again. You will obtain the final result, as shown in Figure 100.

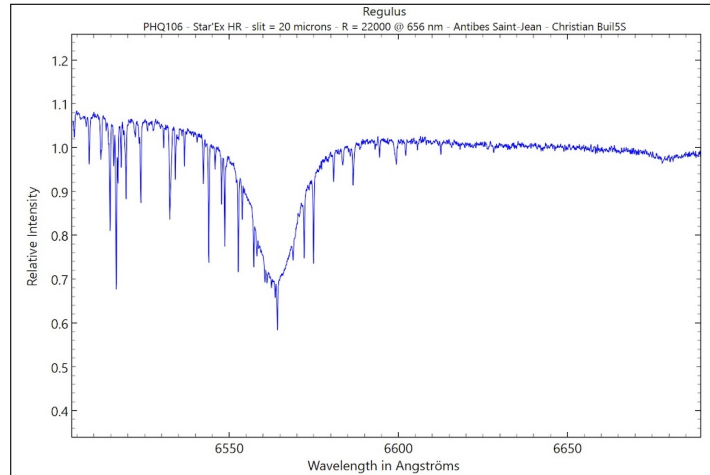


Figure 100 – Final spectrum of the star Regulus.

You can improve the quality of the reduction by adding the filtering parameters described in Section 6.5.4 for low resolution, which are entirely transferable to high resolution (particularly useful for objects of low surface brightness). It is also advisable to document the FITS header of the reduced data. Lastly, the atmospheric-transmission correction can be enabled via the `corr_atmo` parameter, although this is less critical at high resolution.

Here is a complete, representative configuration file for high-resolution processing (you can copy and paste it as a starting point):

```
working_path: C:/starex_20250405
batch_name: regulus
calib_mode: 0
line_type: lateral
auto_calib: [6490, 6722]
poly_order: 2
instrumental_response: _rep_20250406_regulus
clean_wave: [6506.5, 6532.8, 6598.9, 6678.3]
clean_wide: [0.9, 0.9, 0.9, 0.9]
xlimit: [400, 1800]
smile_radius: -16000
crop_wave: [6500, 6700]
norm_wave: [6640, 6660]
# -----
kernel_size: -3
extract_mode: 1
sky_mode: 1
sigma_gauss: 0.5
# -----
Longitude: 7.093
Latitude: 43.587
Altitude: 50
simbad: 1
corr_atmo: 0.13
```

```
# -----
site: Antibes Saint-Jean
inst: 106fsq + starex_hr + PL585
observer: cbuil
```

Let us continue working with this spectrum by checking the quality of the spectral calibration achieved. **specINTI Editor** offers a useful function for this purpose when dealing with high-resolution spectra covering the region of the H α line. Display the reduced spectrum from the “Spectrum” tab, then click the “Check telluric” button (Figure 101).

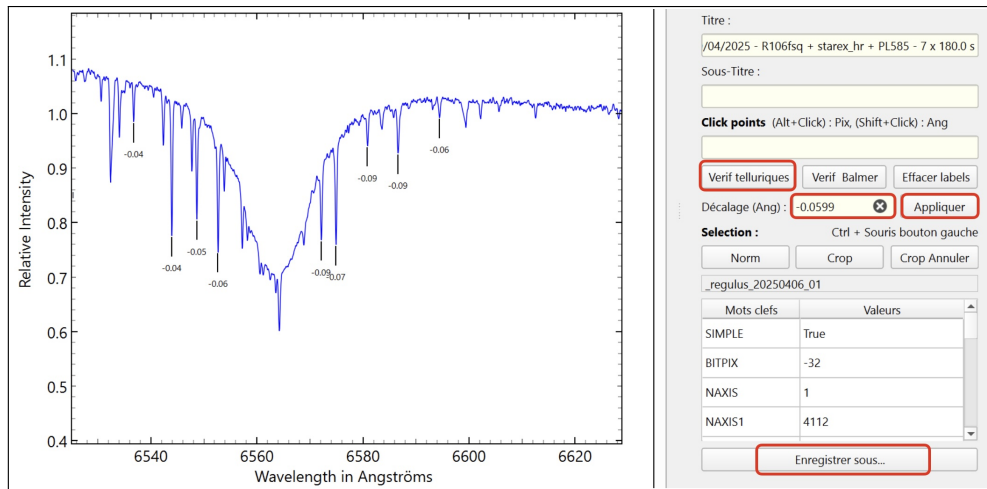


Figure 101 – Checking the spectral calibration based on the position of the telluric lines.

The application evaluates the wavelengths of a number of telluric water-vapour (H₂O) lines, which are prominent in this part of the spectrum, and compares them with the theoretical wavelengths. The mean discrepancy is a measure of the wavelength-calibration error, here -0.06 Å. Clicking the “Apply” button corrects the spectrum for this offset. Save this new version of the profile (“Save as...” button) under the same name or a different one; here we simply choose “p0”.

We shall carry out one final operation on this “p0” spectrum, now precisely calibrated in wavelength: removing the telluric lines, those lines formed when starlight passes through the Earth’s atmosphere. These lines are a form of contamination, and even though they were useful earlier for refining the spectral calibration, they can be removed to give a more accurate picture of the star’s true spectrum. Be careful, however: certain stellar databases, such as BeSS, devoted to Be stars, require the telluric lines to be kept in the spectrum. This operation, fairly delicate and rarely necessary, is above all a good opportunity here to introduce the concept of **functions** in **specINTI**.

Functions are used to perform more or less basic operations, either on images or on spectral profiles (a list is given in Section 13, coming up later). They fit on a single line in a simplified configuration file, in the form of the function’s name followed by a list of arguments.

Here we use the `_pro_telluric` function (note the “_” character at the start of the name, which is characteristic of functions), used as follows:

```
_pro_telluric: [molecular_spectrum, input_spectrum, output_spectrum]
```

where `molecular_spectrum` is the name of a spectral profile of the telluric lines, in the form of a FITS file that must be located in the “atmo_molecular” subfolder of the **specINTI** installation directory.

We use precomputed synthetic molecular spectra, available in a library that can be downloaded from the ESO *Sky Model Calculator* website: <https://www.eso.org/observing/etc/bin/gen/form?INS.MODE=swspectr+INS.NAME=SKYCALC>. They are formatted for use with specINTI and correspond to various amounts of precipitable water vapour and spectral resolutions. You can download this library of spectra from this link (a file to be unzipped into the `atmo_molecular` directory): http://www.astrosurf.com/buil/specinti2/atmo_molecular.zip.

For our example, we use the atmospheric profile “`_molecular_30mm.fits`” (Figure 102) and write the configuration file as follows (just two lines):

```
working_path: C:/starex_20250405
_pro_telluric: [_molecular_30mm_HR, p0, p1]
```

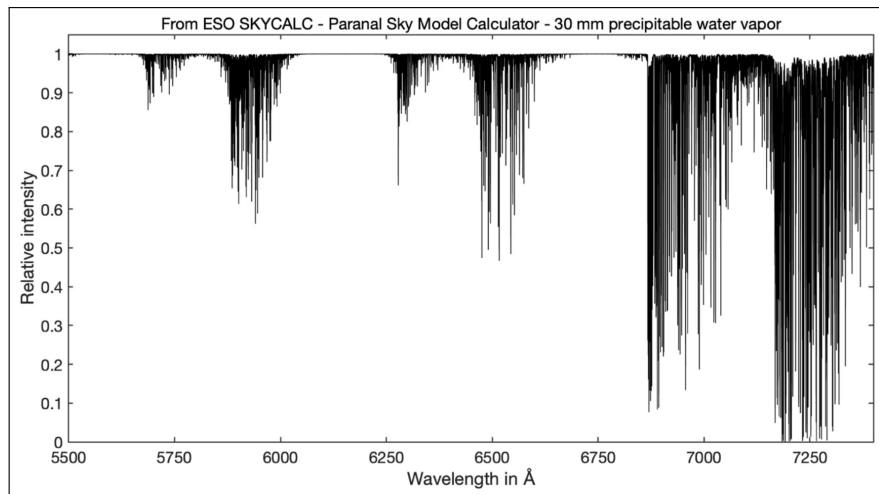


Figure 102 – Extract from the spectrum “`_molecular_30mm.fits`”, which you can view from the “Spectrum” tab.

Clicking “Go!” turns the starting profile “`p0`” into an output profile “`p1`”, in which the telluric lines have been largely erased. This is the spectrum of the star Regulus as it would be observed from space (Figure 103).

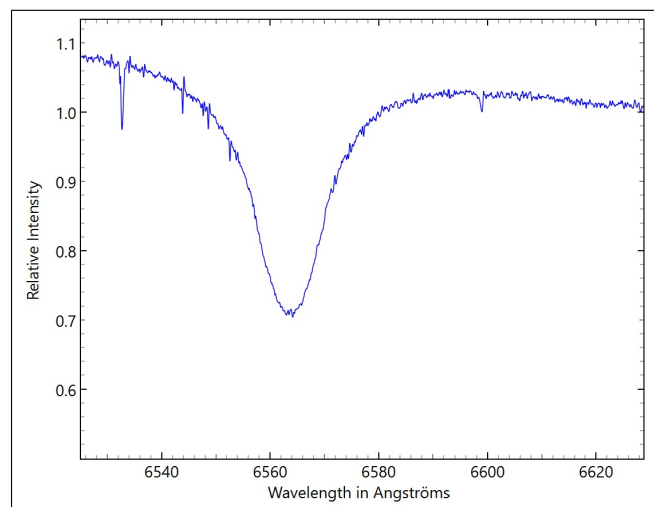


Figure 103 – Spectrum of the star Regulus with the water-vapour telluric lines removed.

8 Summary

By way of a summary, we offer one final session, all the more instructive for being strewn with pitfalls — five of them, at least — which we shall learn how to work around. We first use the standard interface of `specINTI Editor` to calibrate the instrument, then to process the spectra.

This session concerns a sequence of 10 images of the spectrum of the star epsilon Cas, of spectral type B3V, acquired with an Askar 107PHQ refractor and a Star’Ex LR spectrograph fitted with a 300 lines/mm grating, a 26-micron entrance slit, and a ZWO ASI533MM camera. The individual exposure time is only 5 seconds per image. The exposures date from 3 September 2025. The images are named `epscas-1`, `epscas-2`, `epscas-3`...

The images delivered by the ASI533MM camera’s sensor are natively 3008×3008 pixels. For our purposes, they were windowed (*cropped*) at acquisition to a 3008×801 format, sufficient to capture the spectral trace with a good margin in every direction.

All of this comes with the usual DOF frames, in composite, pre-processed form, named `_offset`, `_dark` and `_flat`.

You can download the data from this link: https://buil.astrosurf.com/specinti4/demo/antibes_starex_20240903.zip

For this example, we assume they are located in the working folder:

“antibes_starex_20240903”.

8.1 First pitfall: the indispensable flat-field

Let us begin by looking at the flat-field image supplied with this dataset. Figure 104 shows how it was taken: by illuminating a sheet of tracing paper stretched over the objective’s aperture with a set of 5 miniature xenon-bulb torches of the MagLite brand. This setup produces a continuous spectrum with a colour temperature of about 3,000 K. Because of this, the resulting light-energy distribution clearly favours the red part of the spectrum over the blue part. This is why one should not hesitate to stack a very large number of flat-field images, in order to maintain a good signal-to-noise ratio in the blue part of the spectrum (the flat-field supplied is an average of 40 images correctly exposed in the red).



Figure 104 – Setup used to acquire the low-resolution flat-field, using incandescent-filament lamps producing a “black-body”-type spectrum.

The screenshot in Figure 105 shows the appearance of the flat-field image (the `_flat` image, displayed from the “Image” tab). It is characteristic of a “flat” taken with a CMOS sensor (IMX533), which produces a fairly marked fringing pattern, particularly visible in the red and infrared part of the spectrum. This is the main reason for using a flat-field to process this session (with the new generation of Sony CMOS sensors, such as the IMX565, these fringes are much fainter, so taking a flat-field is no longer always necessary).

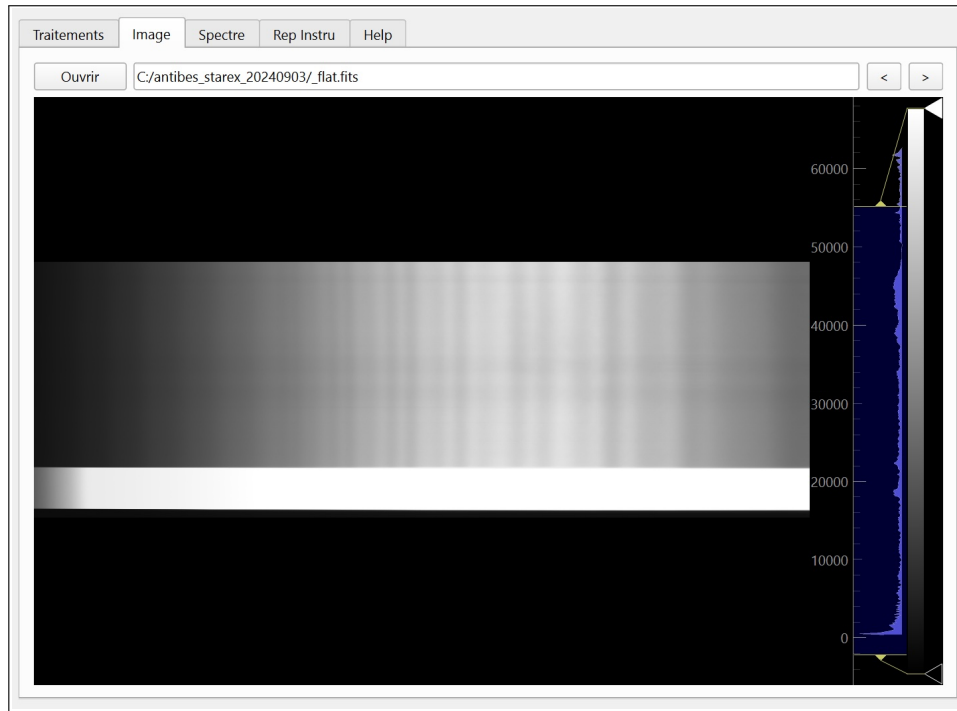


Figure 105 – Appearance of the flat-field image for the epsilon Cas session, showing the interference fringes characteristic of the IMX533 sensor.

Note also, at the bottom of the image, an overly bright band corresponding to the photometric zone of a GEN2 slit, which will need to be carefully avoided during processing.

8.2 Second pitfall: the calibration star is not of spectral type A

Fill in the “LR Calibration” interface of standard mode as shown in Figure 106. You are on familiar ground: use “eps_cas” as the object’s root name, then click “Autofill”.

Figure 106 – Entering the parameters to calibrate the instrument from the observation of the star epsilon Cas. The spectral-type field is filled in manually (see text). Remember also to fill in the DOF information.

By default, the standard interface of **specINTI Editor** requires the use of a type-A star to calibrate the instrument. However, epsilon Cas is a star of spectral type B3V. Is it possible, despite this, to calibrate the instrument with this object? The answer is yes, even though a spectral-type-A star remains preferable.

The catch is that, if you click the “Type Sp” button from the “LR Calibration” interface, the software returns an error message stating that the spectral type is not valid. Simply force your way past this by manually entering `p_B3V` in the “Reference spectrum” field: this profile does indeed exist in the Pickles library, as can be checked by looking in the “_calib” subfolder of the software’s installation directory (or, if needed, by choosing an approximate type). Everything then works correctly.

Third pitfall: dealing with chromatism The refractor used (despite being “APO”) shows a chromatism defect in the ultraviolet part of the spectrum, which appears as a local widening of the trace. This is known as a “fish-tail” effect (Figure 107). This is a classic phenomenon with refractors, more or less pronounced depending on the model (in the present example, it is in fact moderate, despite appearances).

This does not pose any particular difficulty, except that the extent of the search zone for the Balmer lines must be reduced on the blue side, as shown in Figure 107.

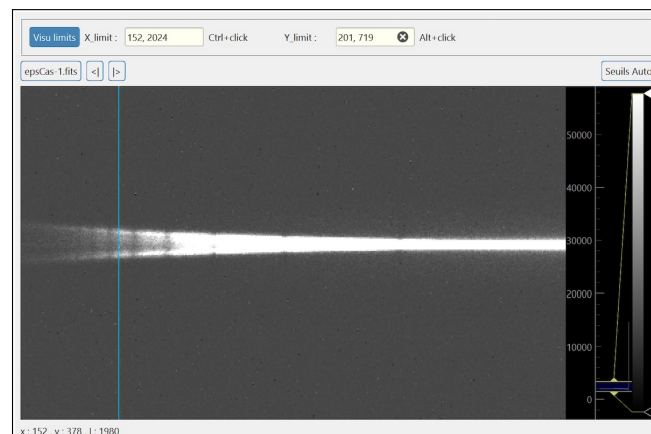


Figure 107 – Positioning the left-hand limit of the Balmer-line search zone for spectral calibration. The widening of the trace corresponds to a defocusing of the spectrum caused by chromatic aberration. Clicking within a fairly wide zone does not trouble the application, as long as the lines to the right of the limit are well contrasted — which is the case here.

On the red side, the problem is different: the spectrum is so narrow that it is difficult to distinguish the $H\alpha$ line without confusing it with the telluric lines further into the red (see Figure 108). Make sure to position the red limit slightly to the right of the $H\alpha$ line.

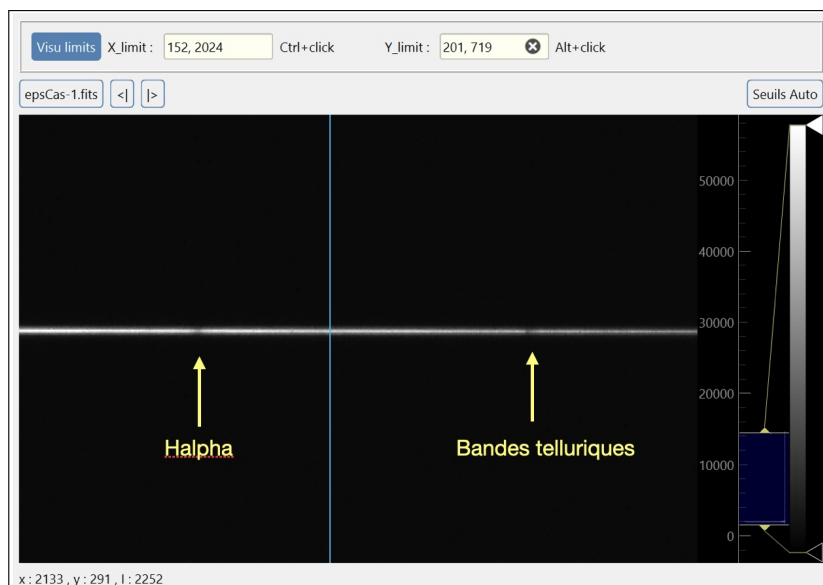


Figure 108 – Positioning the right-hand limit of the Balmer-line search zone for spectral calibration. Adjust the display contrast to clearly distinguish the $H\alpha$ line, very narrow here.

Also remember that the photometric zone of the GEN2 slit intrudes into the acquired image. In anticipation of observing objects much fainter than our reference star, remember to raise the lower limit of the exclusion zone used when searching for the spectral trace (here to above approximately $y = 200$); see Figure 109.

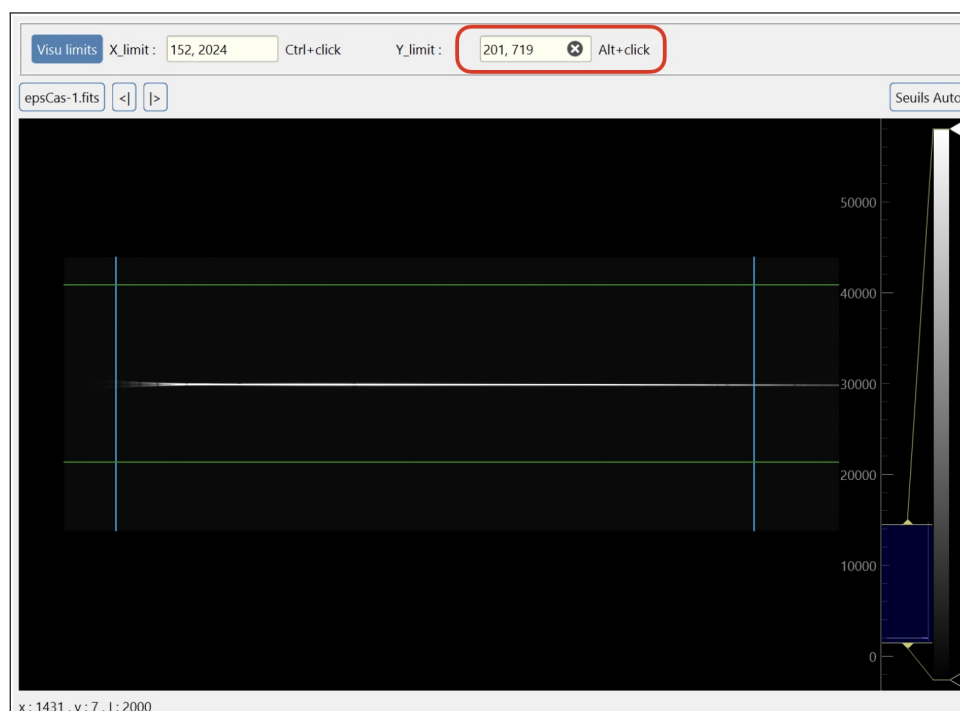


Figure 109 – Defining the exclusion zone for our example. Note the high position of the lower coordinate, chosen so as to avoid the photometric zone (not essential here, since the object is very bright, but this will not always be the case).

8.3 Fourth pitfall: watch out for the narrowness of the spectrum

Let us examine the trace of one of the raw spectra with a very high zoom (Figure 110). Except in the region of the “fish-tail” in the blue, it turns out to be very narrow. Several factors con-

tribute to this: (1) a refractor of relatively modest aperture, not very susceptible to atmospheric turbulence; (2) a short focal length (740 mm); (3) optics delivering very sharp images over a large part of the visible spectrum; (4) the use of a camera with “large” pixels (3.76 microns). All of this leads to severe undersampling of the spectrum. As a result, the width of the trace (FWHM) is barely 1 pixel.

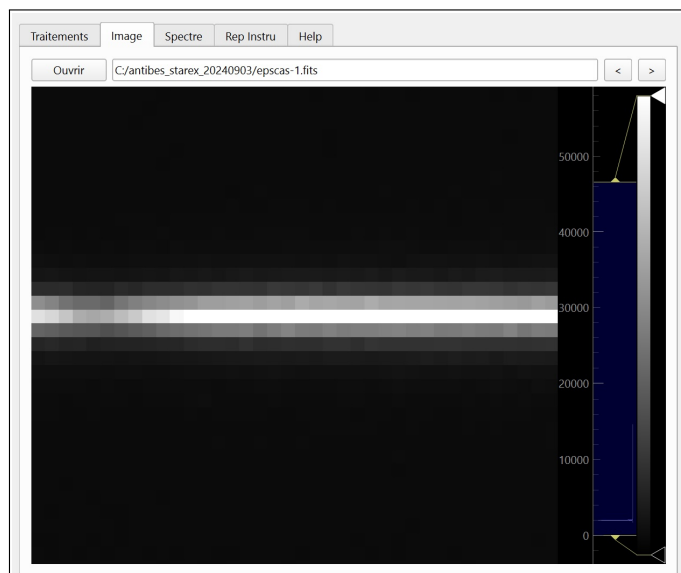


Figure 110 – Enlargement of a portion of the star’s spectrum. The full width at half maximum (FWHM) is about 1 pixel, corresponding to an undersampling regime for the spectral image.

In such a situation, it is essential not to use a median-type statistical filter, such as that offered in the **specINTI Editor** interface, since this can potentially generate artefacts in the image (to use such filtering safely, the finest details of the spectrum would ideally need to be sampled over at least 2.8 pixels). The “Spectrum” dialogue box should be set as shown in Figure 111, both when calibrating the spectrograph and when carrying out the actual reduction (the filtering-kernel value, or “kernel”, set to 0).

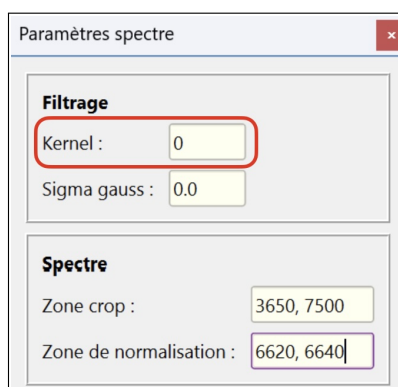


Figure 111 – Setting the “Spectrum” dialogue box. Median filtering of impulsive noise is disabled here. You remain free, however, to apply Gaussian filtering by giving a non-zero value to “Sigma gauss”, but here we are aiming for maximum spectral resolution.

Processing is carried out in the usual way from the “LR Processing” panel. The calibration parameters are transferred automatically (Figure 112). Simply click “Go!” (taking care to first check that your DOF frames are correctly referenced, since these are different from those used during calibration).

Répertoire

C:/antibes_starex_20240903

TRAITEMENT LR

Nom objet :

eps Cas

Autofill

Dir

Fichiers Image :

epsCas-

Ouvrir

Nb images :

10

...

DOF

☒ Etoile

☐ Objet étendu

Coefficients du polynôme de la loi de dispersion :

4.142e-06, 1.5429, 3546.6263

Méthode de recalage :

None

...

Raie de recalage

Réponse instrumentale:

_rep_20240903_epscas

Géométrie

Spectre

GO !

Figure 112 – Launching the reduction of the epsilon Cas spectrum from the “LR Processing” panel.

The result is shown in Figure 113, depending on whether the median-filter parameter is set to -3 or to 0 . In the first case, the appearance of the computed profile (in blue) departs, in places, from the expected spectrum (the red profile corresponds to the “_op2” check file of the Pickles star, created automatically during calibration). In the second case, the computed spectrum matches the reference-library spectrum very closely.

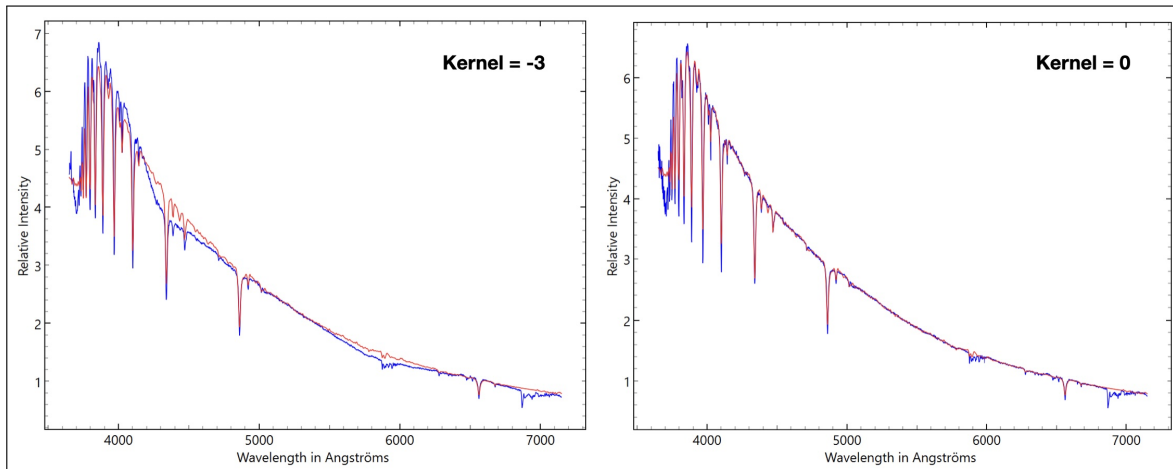


Figure 113 – Comparison of the computed spectrum (in blue) with the Pickles spectrum of a type-B3V star (in red), with and without median filtering of impulsive noise applied. The pronounced undersampling of the spectral trace by the detector’s pixels severely affects the integrity of the spectrum (erasing useful detail). In this situation, median filtering clearly has to be abandoned.

Figure 114 shows the spectral response computed when calibrating the instrument, from the observation of the star δ Cas. The steady slope, from red to blue, comes from the flux distribution of the flat-field lamp (an inverted black-body curve), which “colours” the response. The drop in response in the ultraviolet, meanwhile, comes from a normal loss of sensitivity of the equipment in this part of the spectrum.

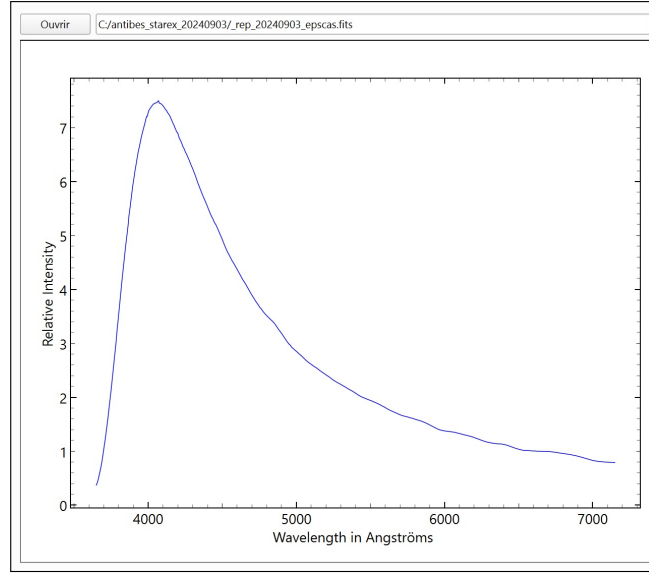


Figure 114 – Appearance of the computed spectral response, strongly shaped by the spectral distribution of the flat-field lamp.

8.4 Fifth pitfall: watch out for the binning width

Binning, as understood here, refers to the operation of summing the signal in the two-dimensional images, column of pixels by column of pixels, to produce the spectral profile. The width of this binning zone is constant from one end of the spectrum to the other, and is adjusted automatically by **specINTI**, with no way to change it when using the standard interface.

The question that arises is this: given that the width of the trace varies from blue to red — particularly in the ultraviolet, owing to the refractor’s chromatism — does the computed binning width remain appropriate over the whole spectrum? If a significant fraction of the spectral trace spills, in places, outside the binning zone, this in fact introduces a spectrophotometric bias into the computed spectrum.

In the example of the star ϵ Cas, **specINTI** fortunately computes a binning width suited to the need. To check this, tick the “Check mode” box from the LR processing panel, run the reduction again, then examine the content of the image “_step101.fits”, automatically saved in the working folder (Figure 115).

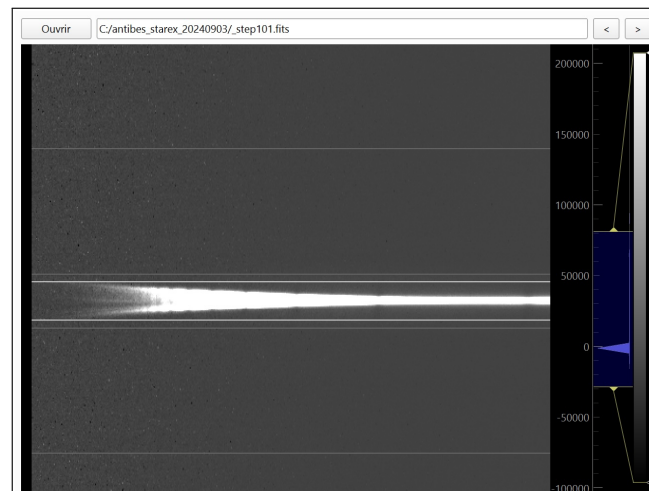


Figure 115 – Display of the content of the image `_step101.fits` at high contrast. The two brightest lines, on either side of the trace, mark out the binning width adopted by **specINTI** for the reduction. Even where the spectrum is widest, this dimension remains sufficient to capture the whole of the recorded signal.

But what should be done if, for one reason or another, it becomes necessary to adjust the binning width? The solution lies in the **advanced interface**. From its “Configuration files” area, click `conf_LR_processing.yaml` (Figure 116). This is the name of the configuration file generated automatically by **specINTI Editor**, then passed to **specINTI** when working from the **standard interface** and clicking “Go!”. “`conf_LR_processing`” is a reserved file name, just like “`conf_LR_calib.yaml`” for calibration (or “`conf_HR_processing.yaml`” for high spectral resolution). These files are the true crossing point between the **standard interface** and the **advanced interface**: the wrapper changes, but the way **specINTI** operates underneath remains the same.

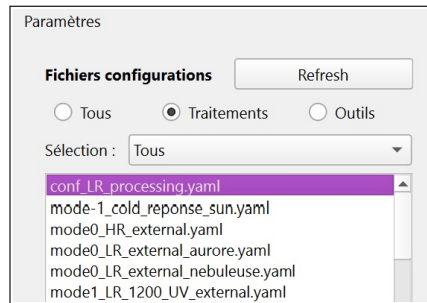


Figure 116 – Selecting the “`conf_LR_processing`” file to inspect its content.

The content of the “`conf_LR_processing`” file is then displayed in the editing area, as shown in Figure 117. While the way the **key: value** pairs are organised differs from what was used earlier (this is a different form of YAML file), the content remains readable. You will recognise here the values of all the parameters edited from the standard interface. For example, `posy_exclude: [200, 720]` corresponds to the exclusion values defined in the “Geometry” part of the standard interface. You will find the full list of **specINTI**’s parameters, together with their meaning, in Section 12. Try, as an exercise, decoding the content of this file purely from the parameter names: it is an excellent exercise.

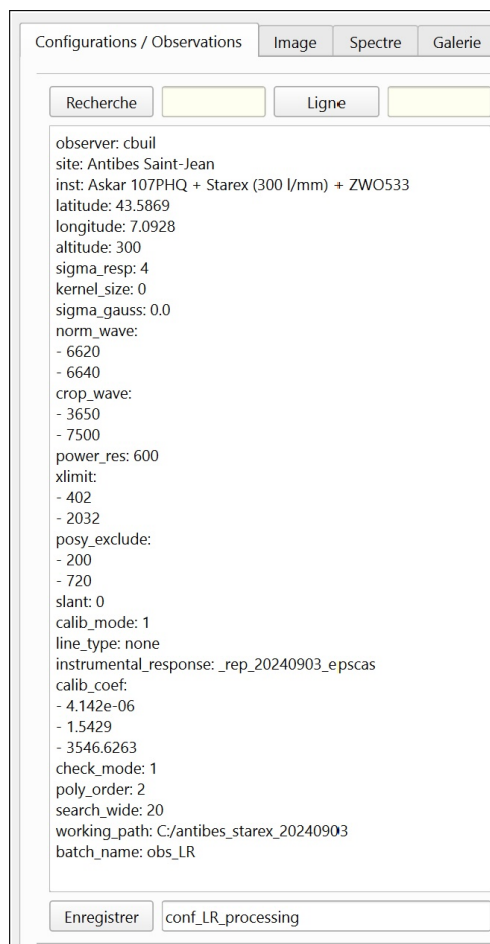


Figure 117 – Content of the configuration file “conf_LR_processing.yaml”.

What we have here is a translation, in list form, of all the data edited from the standard interface. Since this is the configuration file currently active in the “Advanced” interface, if you click the “Go!” button, you carry out exactly the same reduction as before, since the content of the configuration file has not been changed.

One of the benefits of the advanced interface is the ability to make fine-grained changes to the configuration file by editing its content directly. For example, if you add the following line (at the end of the file, say):

```
bin_size: 30
```

and run the reduction again (from either the standard or the advanced interface, it makes no difference), the binning width changes from the value automatically computed by **specINTI** to the value you impose manually, namely 30 pixels. You can see the result by viewing the image “_check_bin.fits” (or “_step101.fits”).

You now have the full power of **specINTI** at your disposal. Another example to illustrate this: simply add the line **seq_mode: 1** to the configuration file to produce, in the working folder, as many reduced spectra (whose names begin with “@”) as there are images in your sequence — rather than a single summed spectrum.

This is a powerful feature for studying the variable phenomena frequently observed in spectra, which make spectroscopy one of the most fascinating disciplines that can be practised in astronomy.

9 The case of the Alpy 600 spectrograph

The Alpy 600 spectrograph (Shelyak Instruments) is a popular instrument, and, as one would expect, data from it can be processed with `specINTI V4`.

The Alpy 600 belongs to the category of low-resolution spectrographs ($R = 550$ to 750 typically). It has, however, another characteristic that can make it tricky to extract the calibration parameters using the automatic modes offered from the standard interface of `specINTI Editor`, in particular for the spectral calibration law — the formula that, as a polynomial, links the pixel numbers in the image to wavelengths. This is the main reason for this section.

A difficulty inherent to the Alpy 600 lies in the existence of a strong, native non-linearity in its dispersion, related to the use of a *grism* component as the dispersive element. This element results from combining a grating with a prism: while the angular dispersion of the grating is indeed linear, the same is not true of the prism. The automated tools offered from the standard interface of `specINTI Editor` (see Section 4) or by mode `-1` used from the advanced interface (see Section 6) can sometimes be defeated by this particular feature of the Alpy 600. These automatic modes assume, in fact, a sufficient degree of linearity in the spectral dispersion — such as that offered by the `Star'Ex` and `TinyStar'Ex` spectrographs — in order to operate with complete reliability.

This section, devoted more specifically to the Alpy 600, is also an opportunity to mention the existence of calibration tools usable in particular situations, which we refer to here as “utility modes” (see the full list in Section 12). You may encounter these situations when using other instruments too, so reading this section is ultimately worthwhile for everyone.

To keep things concrete, we shall show how to process a dataset acquired by the 2SPOT team (<https://2spot.org/FR/>), who remotely operate a dual telescope located in Chile (Figure 118). The images were acquired with no human intervention on 24 April 2026.

You can download this data by clicking on the following link: https://buil.astrosurf.com/specinti4/demo/2spot_alpy_20260424.zip



Figure 118 – The equipment located in Chile and operated from France by the 2SPOT team. The optical tube of interest here is the one on the left: a 305 mm f/4 Newtonian telescope with the Alpy 600 spectrograph at the focus.

The batch of images obtained with the Alpy 600 comprises:

- 5 images of the spectrum of the star HD 103883, of spectral type B4V and magnitude 5.6, each exposed for 15 seconds at the focus of the 305 mm f/4 Newtonian telescope. The camera is an Atik 414EX CCD model. These images are named HD103883-1, HD103883-2, and so on.
- 1 “neon” calibration image (HD103883_neon-1).
- 6 images of the spectrum of the Wolf-Rayet star WR31, of magnitude 10.5, each exposed for 600 seconds with the same telescope. These images are named WR31a-1, WR31a-2, and so on.
- 1 “neon” calibration image associated with this last sequence (WR31a_neon-1).
- 1 composite offset image named M10_BIN1_offset.
- 1 composite thermal-signal image corresponding to a 600-second exposure time, named M10_BIN1_Dark600.

Note: We do not make use of the *flat-field* lighting system built into the Alpy 600. Indeed, its built-in lamp has too low a colour temperature and does not, in our view, provide enough light to correctly calibrate the bluest part of the spectrum. This risks excessively amplifying noise in that region. We prefer to use a much hotter natural source, in this case a type-A or type-B star. Although this is not, strictly speaking, a true *flat-field* source, it ultimately serves a very similar purpose.

We assume that this data is stored in a working folder whose path is “d:/wr31a” (you can, of course, adapt this location to suit your own setup).

We shall first make use of the spectra of the star HD 103883, which will serve both to compute the spectral dispersion law and to evaluate the instrumental response of the 2SPOT system.

From the advanced interface, begin by telling the software the path to the working folder, then fill in the various fields needed to build the observation file (Figure 119).

Répertoire: D:/WR31a

Obj Nuit: Autofill Effacer Dir

Noms objets: HD103884

Nom images: HD103884-

Nb Images: 5

Image calib: None

Nb Img calib: 0

Trans Atm: None

Décalage Flat: 0

Offset: M10_BIN1_offset Nb 0

Dark: 0_BIN1_Dark600 Nb 0

Flat: None Nb -1

Image postfix: -

Séparateur nom: Aucun

Calibration prefix:

Calibration postfix: _neon-

Enregistrer: obs_HD103884

Figure 119 – Observation data for the star HD 103883. Note that we do not use the “neon”-lamp calibration image here, nor any flat-field images, for this reduction.

The first task to carry out is to determine the spectrograph’s wavelength calibration law. We shall rely for this on the Balmer-series lines visible in the star’s spectrum. Here they are relatively narrow, the object being a very hot main-sequence star (type B4V), but they remain excellent spectral markers all the same (Figure 120).



Figure 120 – Detail of one of the images of the star HD 103883 (blue part of the spectrum). The Balmer lines appear as fine notches cut into the spectral continuum.

specINTI's utility mode –2 is precisely designed for this kind of task: making use of absorption lines present in the star's spectrum, of known wavelength, to evaluate the dispersion polynomial. Here we shall use the first 6 lines of the hydrogen Balmer series, whose wavelengths in angstroms, going from blue to red, are:

3889.05 3970.08 4101.75 4340.48 4861.34 6562.81

You will find a complete list of the Balmer-series lines in Section 14 (Figure 140).

The aim is to associate each of these lines with a pixel coordinate along the spectral axis (x) in the images. This work is done by hand, but rest assured: it only needs to be done once, provided your equipment stays unchanged (no dismantling, for example).

We therefore need to read off the pixel coordinates of the lines corresponding to our list of wavelengths. **SpecINTI Editor** offers a handy tool for this. From the “Image” tab, Shift+click with the mouse on each line, in increasing order of wavelength (that is, going from left to right). Take care not to miss any! The x coordinates of the clicked points are shown in the parameters area of the “Image” tab (Figure 121).

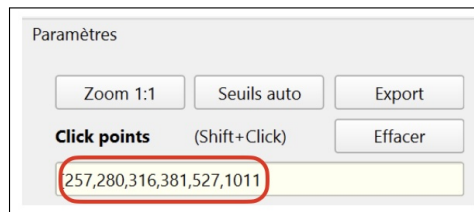


Figure 121 – x coordinates resulting from mouse clicks on the selected lines of the Balmer series.

Tip: if you prefer to start with the $H\alpha$ line, you are entirely free to do so, but you will then need to reverse the order of the wavelength list in the configuration file to keep the correspondence correct (the list then starting with the wavelength of $H\alpha$, i.e. 6563 Å).

Now go to the configuration-file editing area of the “Configurations/Observations” tab and enter the following parameters:

```
working_path: D:/WR31a
batch_name: obs_HD103883
calib_mode: -2
wavelength: [3889.05, 3970.08, 4101.75, 4340.48, 4861.34, 6562.81]
line_pos: [257, 280, 316, 381, 527, 1011]
poly_order: 3
search_wide: 20
bin_size: 40
sky: [160, 30, 30, 160]
xlimit: [300, 1050]
check_mode: 1
```

Tip: in the “_configuration” directory created automatically when the software is installed, you will find a ready-to-use file named “mode-2.yaml”, which you need only tweak slightly.

If you have followed the guide up to this point, you are on familiar ground.

The `working_path` and `batch_name` parameters identify your working directory and the name of your observation file.

We select utility mode `-2` via the `calib_mode` parameter.

Next come the list of wavelengths and their pixel coordinates, passed respectively to the `wavelength` and `line_pos` parameters. The wavelength list stays essentially the same every time. For the position list, simply copying and pasting the sequence obtained from the earlier Shift+clicks, between the brackets of the `line_pos` parameter, will reduce the risk of error and be quicker.

The degree of the dispersion polynomial is set to 3 instead of the usual 2; we shall see why below.

We also fix the values of the binning-zone width (`bin_size` parameter) and of the sky-background computation zones on either side of the spectral trace (`sky` parameter).

Finally, we use a wide lever arm between pixels $x = 300$ and $x = 1050$ to evaluate certain geometric characteristics of the spectrum (`xlimit` parameter).

Figure 122 shows the appearance of the editing area at this stage in a screenshot; you will find all our parameters and values there.

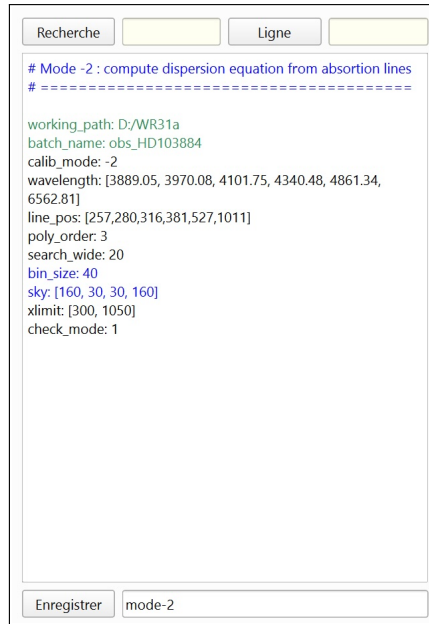


Figure 122 – Content of the configuration file for computing the dispersion law.

All that remains is to click “Go!”. After a few moments, the result is shown in the console (Figure 123): the dispersion law of the Alpy 600 combined with an Atik414EX camera.

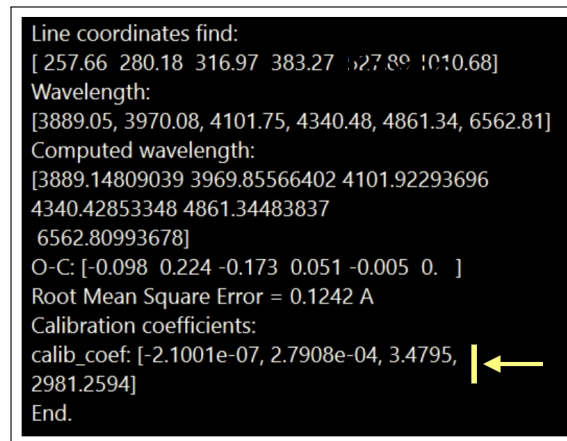


Figure 123 – Result of the spectral calibration: values assigned to the `calib_coef` parameter.

Let us note the smallness of the RMS fitting error, only $0.12 \text{ \AA}$ here, whereas the mean dispersion is about $3.5 \text{ \AA/pixel}$. If, on the other hand, you choose a degree-2 polynomial and run the computation again, the statistical error rises to $0.87 \text{ \AA}$, which is markedly less satisfactory.

Here we are confronted with the problem of the Alpy 600’s native non-linearity, which demands greater rigour when calibrating the instrument’s spectrum.

Note, however, that this RMS error corresponds to the numerical fitting error of the dispersion, and does not necessarily reflect the true calibration error: the input data itself carries its own uncertainties, linked in particular to the quality of the software’s algorithms, to noise, to optical distortions, or to the inherent limitations of the spectrograph model (residual non-linearity in particular).

The second step of the calibration is to determine the instrument’s spectral response to the incoming light flux.

The technique is the usual one:

- first, a complete reduction of the reference star’s spectrum is carried out using the spectral calibration law just established, without correcting for the instrumental response, since it is not yet known;
- the result is then divided by the expected profile of the same star, taken from a library of reference spectra; this gives the instrumental-response curve, reusable for all your data;
- as a check, our star is then processed once more, this time applying the computed instrumental response.

Let us carry out all of these operations; it only takes a minute or two.

Edit a new configuration file, this time intended for the actual reduction of the spectrum. Here is its content (you will also find it in the “_configuration” directory under the name “model_LR_none”, saving you from having to type it all in again — the idea being always to start from an existing file to save effort):

```
working_path: D:/WR31a
batch_name: obs_HD103883
calib_mode: 1
line_type: none
calib_coef: [-2.1001e-07, 2.7908e-04, 3.4795, 2981.2594]
bin_size: 40
sky: [160, 30, 30, 160]
xlimit: [300, 1050]
norm_wave: [6620, 6640]
crop_wave: [3600, 7400]
kernel_size: 0
extract_mode: 1
sigma_gauss: 0.2
# instrumental_response: _rep_20260424
```

Here we enable calibration mode 1 (**calib_mode** parameter), using the polynomial as it stands, with no realignment of the constant term, hence the “none” value given to the **line_type** parameter.

You will no doubt recognise the calibration coefficients computed during calibration: they have simply been copied and pasted from the console into the **calib_coef** line.

We also find here parameters already seen during calibration, to which a few options for reducing noise have been added.

It should be noted that the **kernel_size** parameter is given the value 0, which disables median filtering (the usual choice with CMOS sensors is -3). There are two reasons for this: (1) the Atik 414EX camera is fitted with a CCD sensor rather than a CMOS one, so filtering impulsive noise is not really necessary here; (2) the spectrum is undersampled owing to the large pixel size, which can introduce distortion when applying a non-linear filter such as the median operator.

We do, however, enable optimal extraction of the spectral profile by giving the **extract_mode** parameter the value 1. Finally, we apply no Gaussian filtering, so as to preserve spectral resolution and because the object is bright; the token value of 0.2 assigned to **sigma_gauss** has no effect on the image (a value above 0.5 is needed before the effect of smoothing becomes noticeable).

The most important point is that the **instrumental_response** parameter is commented out (or absent), since we do not yet have the instrumental response.

Launch the reduction (“Go!” button). Figure 124 shows the profile obtained: it is calibrated in wavelength, but not yet corrected for the response.

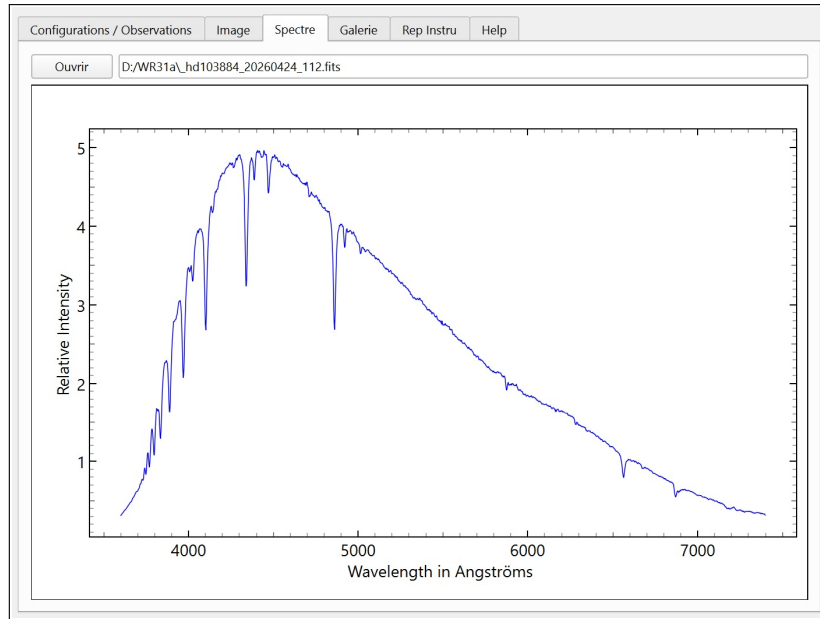


Figure 124 – The processed spectrum with no instrumental-response correction.

We shall now compute the instrumental response by opening the “Instr. resp.” tab and specifying the name of the spectral profile just extracted (Figure 125).

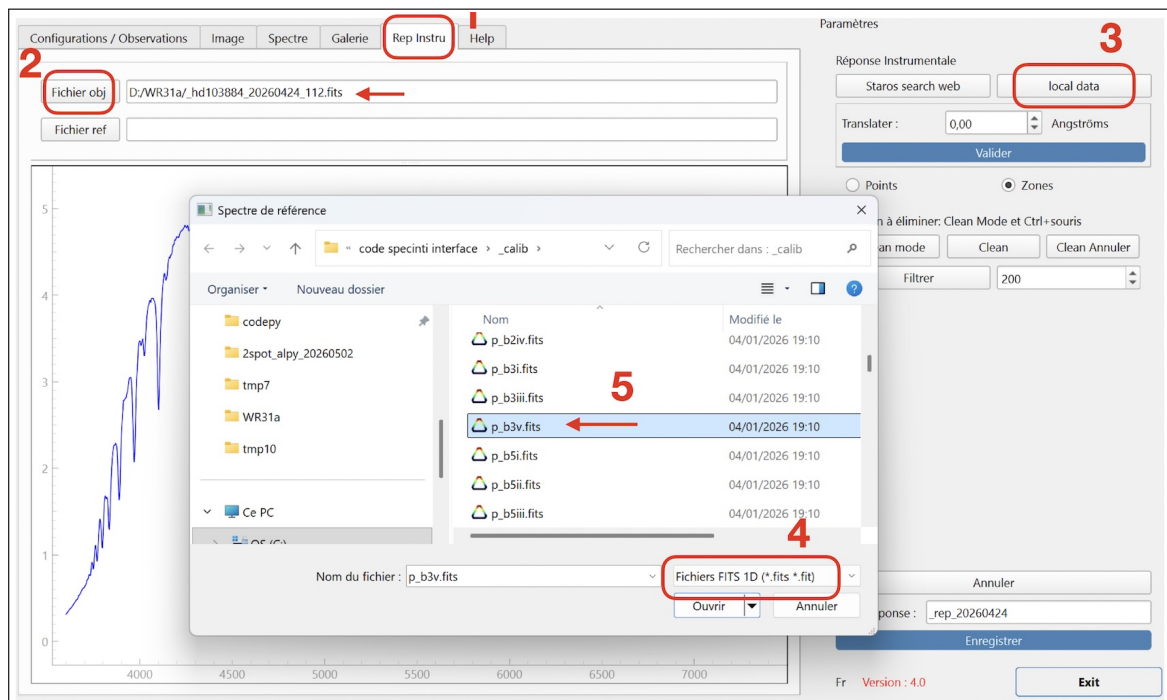


Figure 125 – Display in the interactive response-computation tool: the observed star’s profile and the reference file.

The star’s profile is then displayed automatically in blue. We now need to load the theoretical profile of a type-B4V star. We use the Pickles spectral library, installed together with the software in the “_calib” folder. Use the “Local data” button to load the Pickles file closest to type B4V (the file name indicates the spectral type). The one chosen here corresponds to spectral type B3V (file “p_b3v.fits”).

Select it: **specINTI Editor** immediately displays the ratio of the two profiles, which already provides a very good approximation of the instrumental response we are looking for (Figure 126).

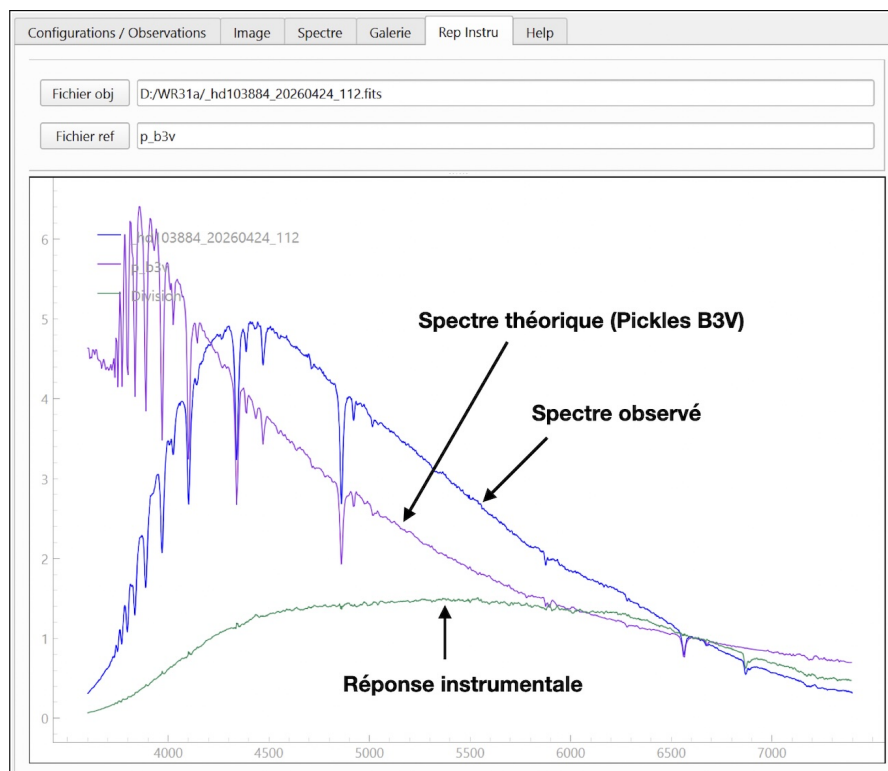


Figure 126 – Automatic computation of the instrumental response.

Confirm the computed profile by clicking the “Validate” button (there is no spectral shift between the observed spectrum and the theoretical spectrum, so this step is not strictly necessary). The provisional response profile is then displayed on its own.

This profile is disturbed by the presence of telluric lines, which need to be removed, as well as by noise, which it is best to filter.

To erase the lines (or molecular bands), enable the erasing mode by selecting “Zones” mode rather than “Points” mode, then click the “Clean mode” button. Then position the blue band with the mouse over the region of the spectrum to be erased; you can also change the width of this zone by dragging its edges. When you are satisfied, click “Clean”. This operation can be repeated as many times as necessary, including on a small scale to deal with a localised artefact, such as a residual line.

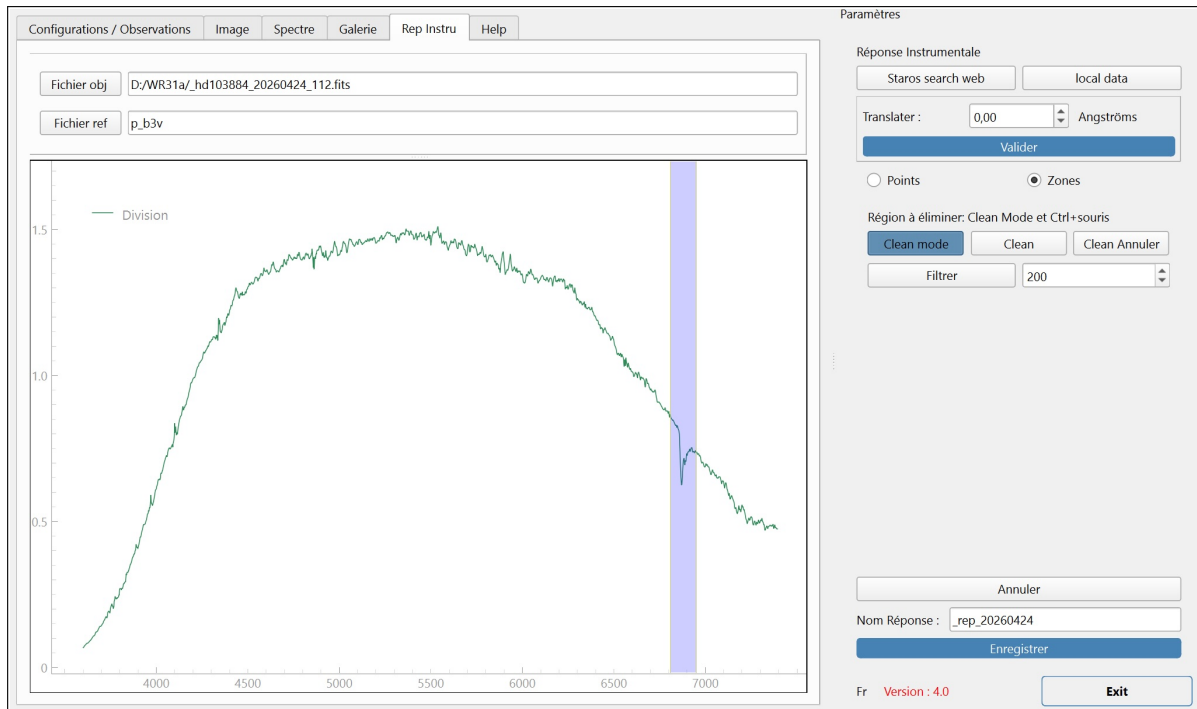


Figure 127 – Erasing the telluric molecular bands. Since the observatory is located at altitude in the Chilean desert, these bands are only weakly marked here; they will be more pronounced at more humid observing sites.

Now click the “Filter” button, keeping the default smoothing value of 200, which works very well in most cases. The red curve that appears is your final spectral-response profile (Figure 128). All that remains is to save it in the working directory under the suggested name, by clicking “Save”.

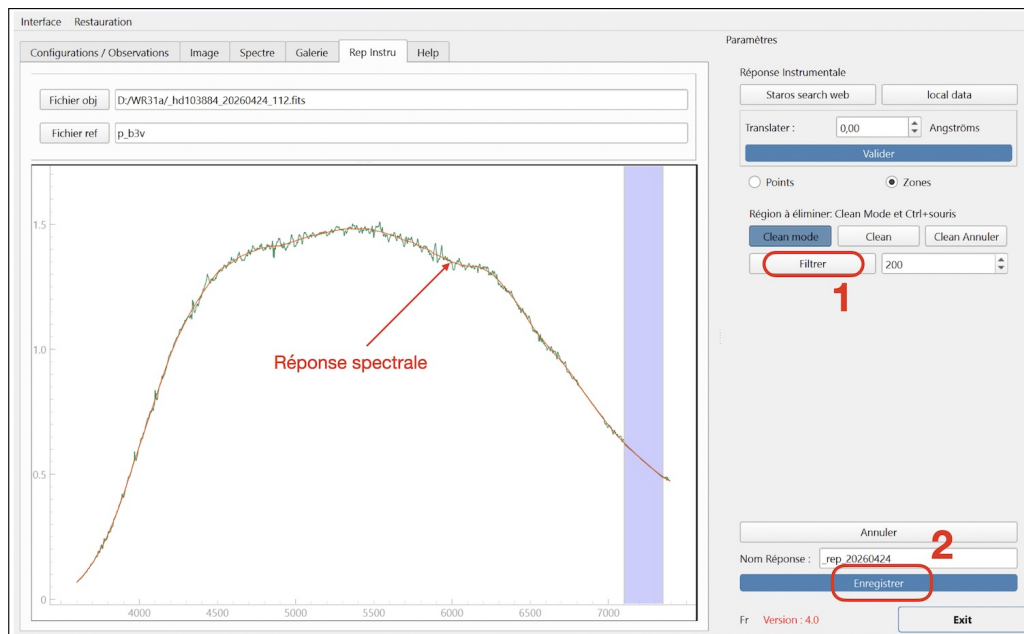


Figure 128 – Finalising the computation of the instrumental response: noise filtering and smoothing of artefacts not representative of the true response.

Go back to the “Configurations/Observations” tab and edit the configuration file to indicate that we now have an instrumental response available, by activating the line: `instrumental_response: _rep_20260424`.

Run the reduction again via the “Go!” button with no other change: you obtain the fully reduced

spectrum of the star HD 103883, as shown in Figure 129 — a fine spectral profile of a very hot star.

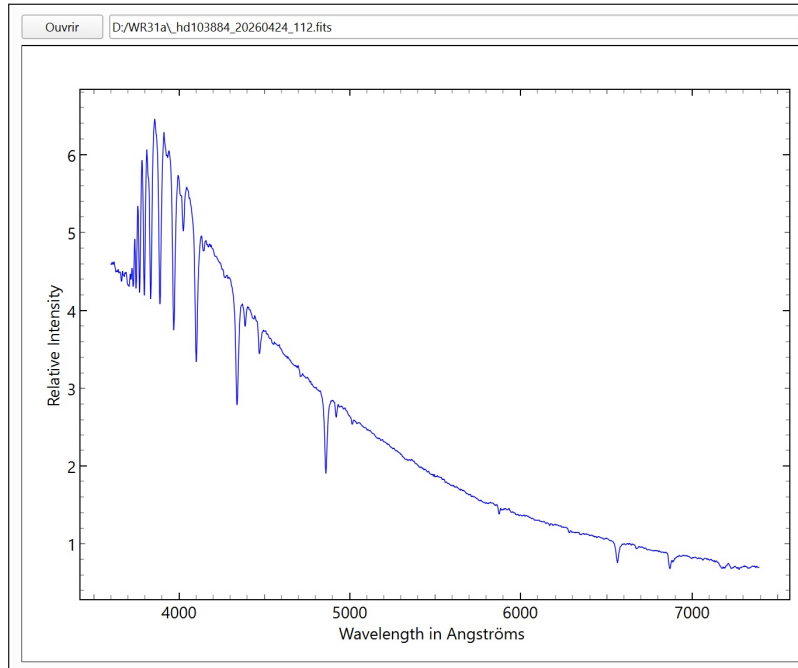


Figure 129 – The final spectrum of the star HD 103883.

Everything now becomes very simple. We shall, for example, process the spectrum of the star WR 31a by reusing the calibration parameters we have just computed — they will no doubt serve you for a long time to come.

The observation data is configured as shown in the screenshot in Figure 130.

Figure 130 – Observation configuration for the Wolf-Rayet star WR 31a.

The image of this object's spectrum, together with its surroundings, is shown in Figure 131.

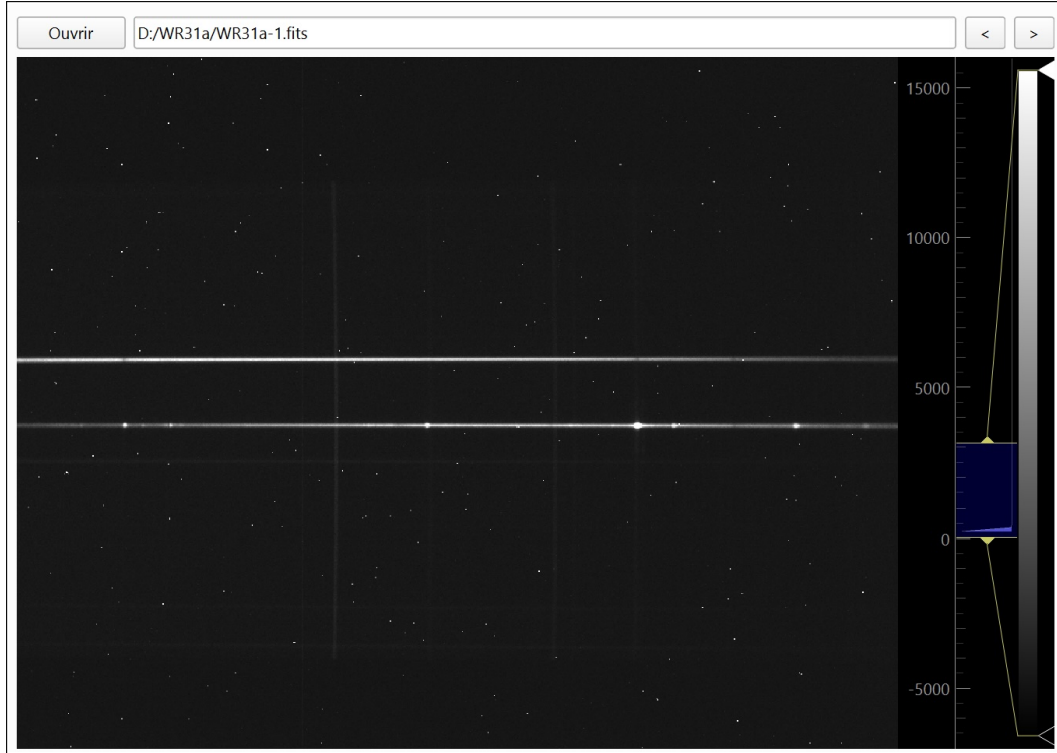


Figure 131 – The spectrum of the Wolf-Rayet star WR31a is the lower one (with its characteristic emission lines). The upper spectrum is the unwanted spectrum of a star that happened to lie on the slit at the same time.

A difficulty arises here: the software must not confuse the trace of the Wolf-Rayet star with that of the interloping star above it, which is, moreover, more intense.

One strategy for dealing with this is to force `specINTI` to process the spectral trace located at the height y of the target star. Measuring this coordinate with the mouse gives $y = 533$. To process this spectrum safely, it is enough to add the parameter `posy: 533` to the configuration file, and nothing else⁸.

We shall, however, adopt a more appropriate approach, making use of `posy_exclude`, a parameter that ignores any part of the image lying outside the two given limits. By examining the position of the interloping star, we choose $y_1 = 490$ and $y_2 = 580$, and write in the configuration file: `posy_exclude: [490, 580]`.

Another point deserves close attention: the binning width and the placement of the sky-background computation zones, which must not include the trace of the interloping star. The only way to find the right parameters is by iteration: adjust the values, then examine the “`_check_bin`” image, created automatically for this kind of check. This image shows the binning zone and the sky-background zones overlaid on the spectrum; it is often more informative than analysing a curve.

And that is not quite all! Examining the “neon” images readily reveals another particular feature of spectra from the Alpy 600: a *smile*-type distortion, which curves the spectral lines, with a radius of curvature of about $+13,000$ pixels⁹. It is important to correct this defect in

8. A less drastic solution, useful when processing a very faint spectrum in a crowded, noisy environment, is to write `posy: -533`. The minus sign tells `specINTI` to position itself around the given coordinate, while still looking for the best centring within the binning zone to vertically realign each image of the sequence. Since the binning zone is always narrow, there is less risk of the software latching onto an artefact rather than the star’s true trace.

9. Tip: to check how straight the lines are, display the image “`_check_smile_sum.fits`”, generated automatically in the working directory.

order to correctly subtract the sky-background signal, hence the addition of the `smile_radius` parameter to the configuration file.

Here, finally, is the form found to be satisfactory for processing the spectrum of WR31a:

```
working_path: D:/WR31a
batch_name: obs_WR31a
calib_mode: 1
line_type: none
calib_coef: [-2.1001e-07, 2.7908e-04, 3.4795, 2981.2594]
bin_size: 40
sky: [160, 30, 30, 160]
xlimit: [300, 1050]
norm_wave: [6620, 6640]
crop_wave: [3600, 7400]
kernel_size: 0
extract_mode: 1
sigma_gauss: 0.2
instrumental_response: _rep_20260424
posy_exclude: [490, 580]
smile_radius: 13000
```

Running the computation with these parameters gives the profile shown in Figure 132.

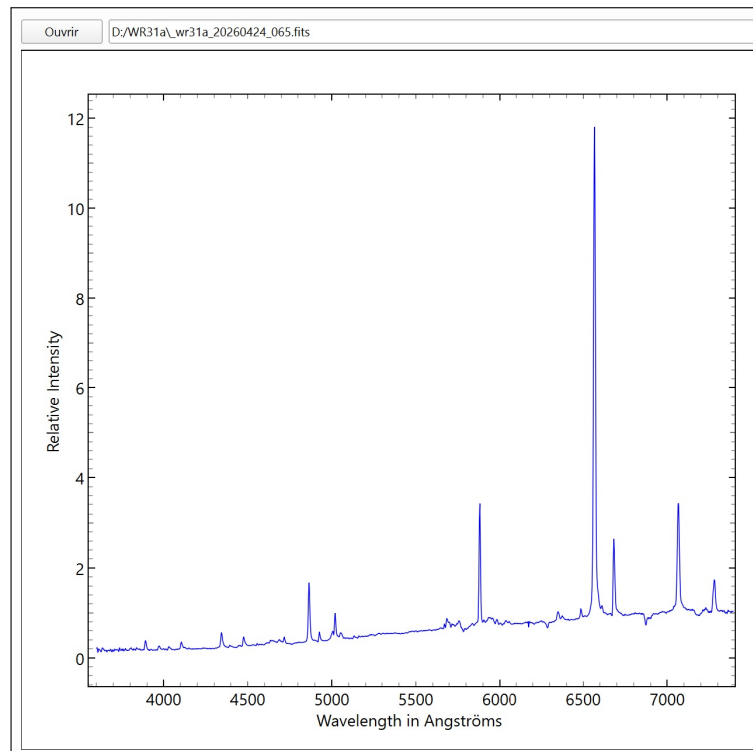


Figure 132 – The processed spectrum of the Wolf-Rayet star WR 31a.

In an environment as crowded as the one surrounding WR31a, it is always very useful to examine the image “`_check_bin.fits`” (or “`_step101.fits`”, an equivalent image, produced when `check_mode` is set to 1, for a finer analysis of the software’s behaviour), see Figure 133.

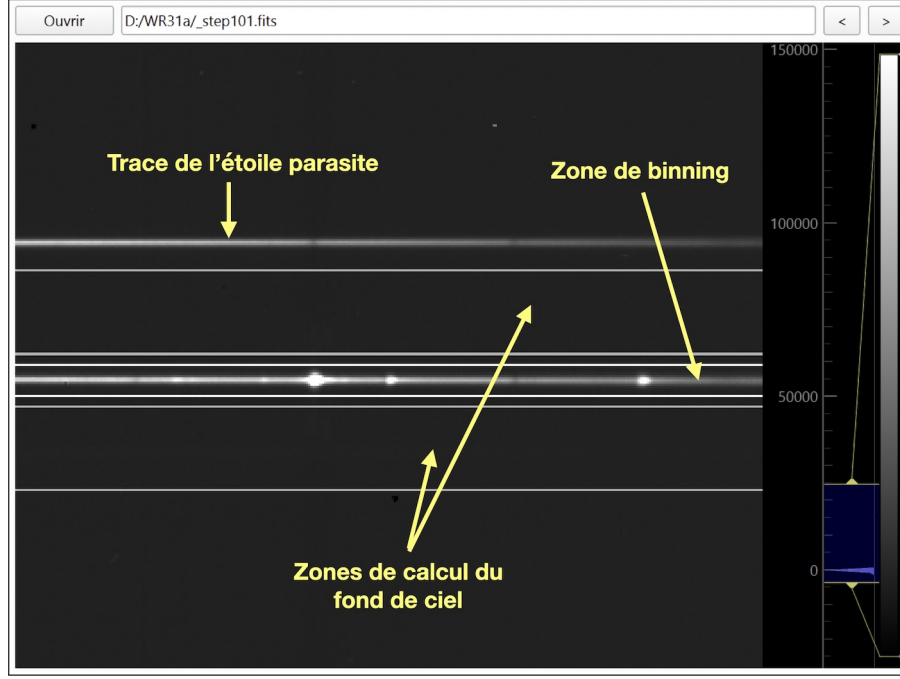


Figure 133 – Content of the images “_check_bin.fits” and “_step101.fits”. Note that the interloping star does not fall within the sky-background computation zone, thanks to the correct choice of the `sky` parameter values. In addition, the binning width is narrowed around the spectral trace to minimise noise and the presence of poorly corrected cosmic rays or hot pixels. The sky background is nice and clean, and the sky-background lines are correctly removed, thanks to the correct choice of the `smile_radius` parameter, determined by trial and error.

Let us stress, finally, that we place complete trust in the spectral calibration performed on HD 103883 to calibrate the spectrum of WR 31a, by giving the `line_type` parameter the value “none”. The Alpy 600 has a reputation for being mechanically stable, and the short time interval separating the two observations suggests that this choice is a sound one.

We can, however, test other calibration scenarios, more dynamic in that they operate in time closer to the observation of WR 31a itself. The aim is to realign the constant term of the polynomial to compensate for any deformation of the spectrograph. We can first make use of the neon image taken immediately after the observation of the target (the file “WR31a_neon-1.fits”, whose existence we noted in the observation file; see Figure 130 and the “Calib. image” field).

Examining the image “WR31a_neon-1.fits” reveals many emission lines¹⁰; we shall keep only two of them, bright and well isolated: one at $x = 806$, corresponding to a neon line at $5852.49 \text{ \AA}$, and the other at $x = 1010$, corresponding to the hydrogen $H\alpha$ line at $6562.81 \text{ \AA}$ (Figure 134).

10. Most Alpy 600 units have a built-in spectral-calibration system, in the form of a small bulb used as a starter (generally the RELCO SC480 model), the kind used in domestic “neon” lighting tubes. This source produces a weak light whose spectrum, fairly complex to decode (sometimes variable over time), consists of lines of neon gas, argon gas, and even, in some series, hydrogen.

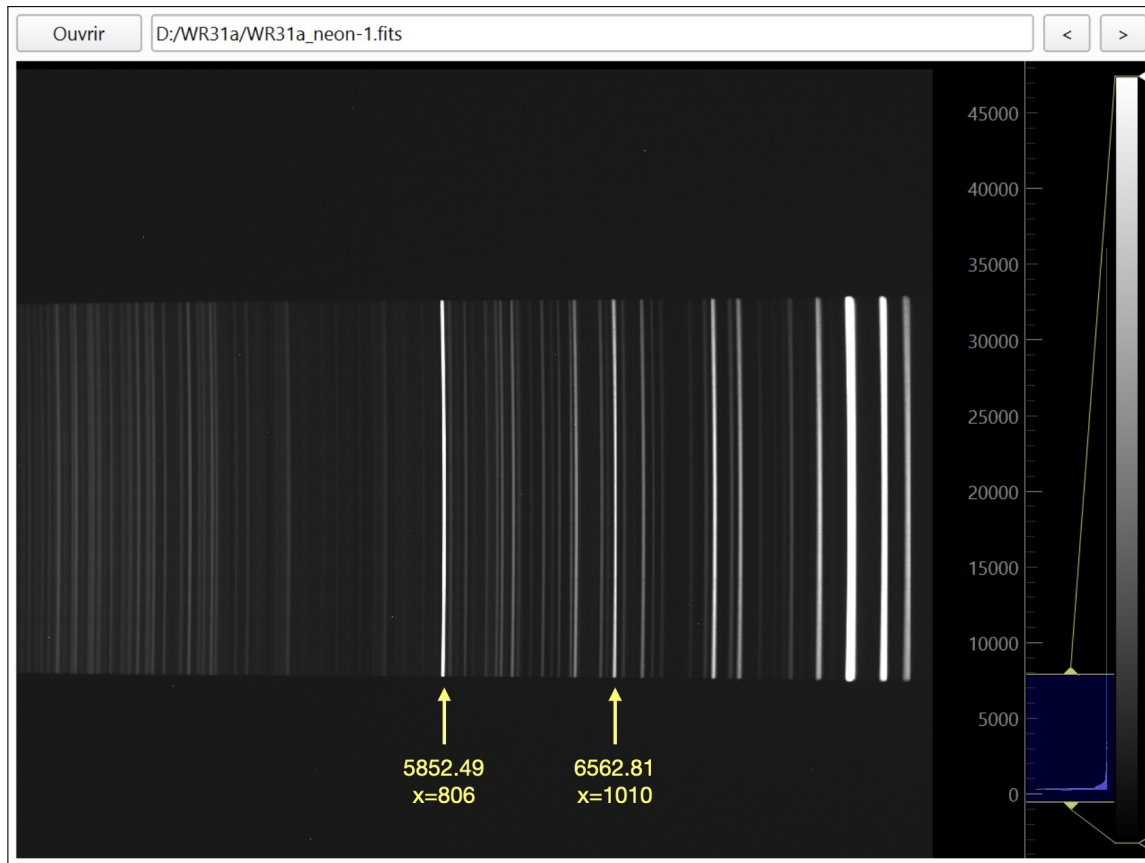


Figure 134 – Identifying two calibration lines in the spectrum of the “neon” lamp.

The configuration file is easy to modify. Via the `line_type` parameter, we specify that an external calibration source is to be used (the value is then the word “external”), and we add the `wavelength` and `line_pos` parameters to give the wavelengths and pixel positions of the corresponding lines. Here are the changes to be made to the previous configuration file:

```
...
line_type: external
wavelength: [5852.49, 6562.81]
line_pos: [806, 1010]
...
```

Run the reduction: you will find that the spectral calibration is almost identical to that obtained without realigning the constant term of the polynomial, which is a good sign.

We can also make use of lines present in the sky-background signal to update the dispersion polynomial. There is no question of artificial light pollution in the clear sky of the Chilean desert, but rather of the [O I] airglow emission, always present in the Earth’s upper atmosphere, which offers an excellent opportunity to calibrate any spectrograph operating at low resolution with long exposures (since airglow lines are relatively faint). These lines become visible in the raw images of WR31a when the display contrast is increased (Figure 135).

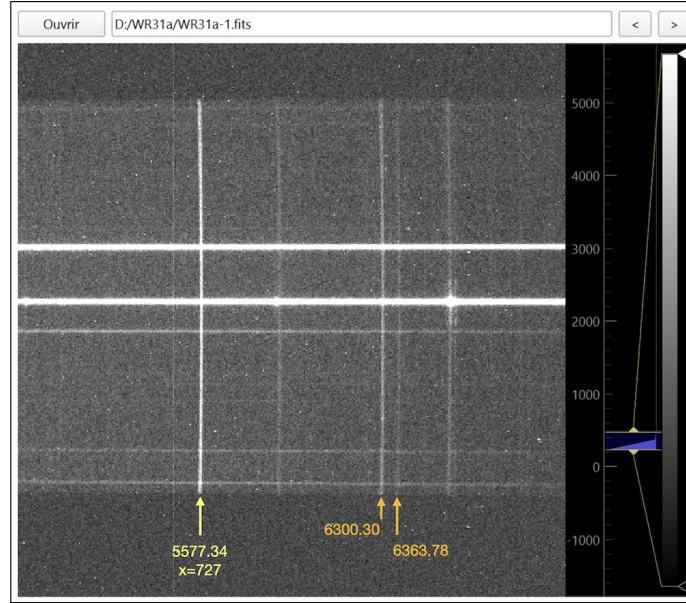


Figure 135 – Identifying the [O I] airglow lines, at the precise wavelengths of 5577.339 Å, 6300.304 Å and 6363.776 Å. Only the first is used for the present calibration in lateral mode; it lies at coordinate $x = 727$. Note also, in passing, the presence of a faint nebula emitting in the red hydrogen line, around the Wolf-Rayet star.

Here we apply the principle of calibration in lateral mode, that is, using one or more lines present within the target’s own image. The `line_type` parameter then takes the value “lateral”, and the configuration file is modified as follows (only the relevant lines are shown):

```
...
line_type: lateral
wavelength: [5577.34]
line_pos: [727]
slant: 0
...
```

The addition of the `slant` parameter deserves a comment. *Slant* corresponds to an angular tilt of the lines relative to the vertical axis. It is important that this tilt be removed by the end of the reduction. This is, in fact, one of the geometric corrections to the spectral images carried out automatically by `specINTI`. The algorithm uses the outer parts of the sky-background computation zone to determine the tilt angle of the lines. In our example, we are relying on a single line (the green airglow line), which is moreover faint, with only a small gap between the extremes of the sky-background computation zones. It is therefore wise, here, not to leave the software free to compute the value of the *slant* itself. This is why this parameter is present, forced, for safety, to the value 0.

Tip: to find the precise value of the *slant* angle, examine the check image “`_check_bin.fits`”, found in the working directory, tightening the contrast well when displaying it. If you observe poorly subtracted sky-background lines, with a relief-like effect that reverses between the top and bottom of the spectral trace, it is probably worth adjusting the *slant* angle value. After a few iterations between processing and inspecting the check image, you will find that the precise value to adopt in our example is an angle of $+0.1^\circ$. This is then written as `slant: 0.1`. Two lessons can be drawn from this: first, that the Alpy 600 spectrograph used by the 2SPOT team is very well aligned — note also that the tilt angle (i.e. the inclination of the spectral trace relative to the horizontal) is very close to 0° ; and second, that the software allows one to work with great precision on the data, the whole secret lying in careful examination of that data, a skill acquired with experience.

In the end, processing in this “lateral” configuration gives an identical result to the previous

ones, while illustrating the flexibility offered by **specINTI**'s advanced mode.

10 The case of infrared observation

A spectrograph's spectral sensitivity range is not limited solely to the part visible to the naked eye. Beyond the red lies the infrared. Most CMOS sensors are sensitive to this part of the spectrum, up to a wavelength of about 1.1 micron, which is a very interesting range to explore with spectrographs such as, for example, the **Lhires III** (by adapting the grating), the **Star'Ex** (in its low-resolution version), or the **TinyStar'Ex**.

Two changes need to be made, however, to everything covered so far: one concerns the hardware, the other the software.

- On the hardware side, it is essential to add a red long-pass filter (of type OG580 or even RG610), screwed, for example, onto the front of the drawtube (as in Figure 136) or fitted into a dedicated filter-drawer slot. This filter prevents the blue part of the grating's second-order spectrum from overlapping the infrared part¹¹.



Figure 136 – Improvised fitting of an order filter at the front of a TSE spectrograph in order to take spectral images in the infrared.

- On the software side, one can no longer rely on the tools that make use of the Balmer lines for wavelength calibration. You can forget the “standard” interface. An external calibration source is essential, ideally a simple neon lamp (see the box “Building an inexpensive neon calibration source”). The configuration file must be edited by hand. There is, however, nothing new here: the structure of this configuration file has already been described.

We therefore have available the spectrum of the star to be processed, and the infrared spectrum of neon, as shown in the example in Figure 137.

11. Reminder: a diffraction grating produces several spectra associated with (interference) order numbers. Most spectrographs make use only of order 1.

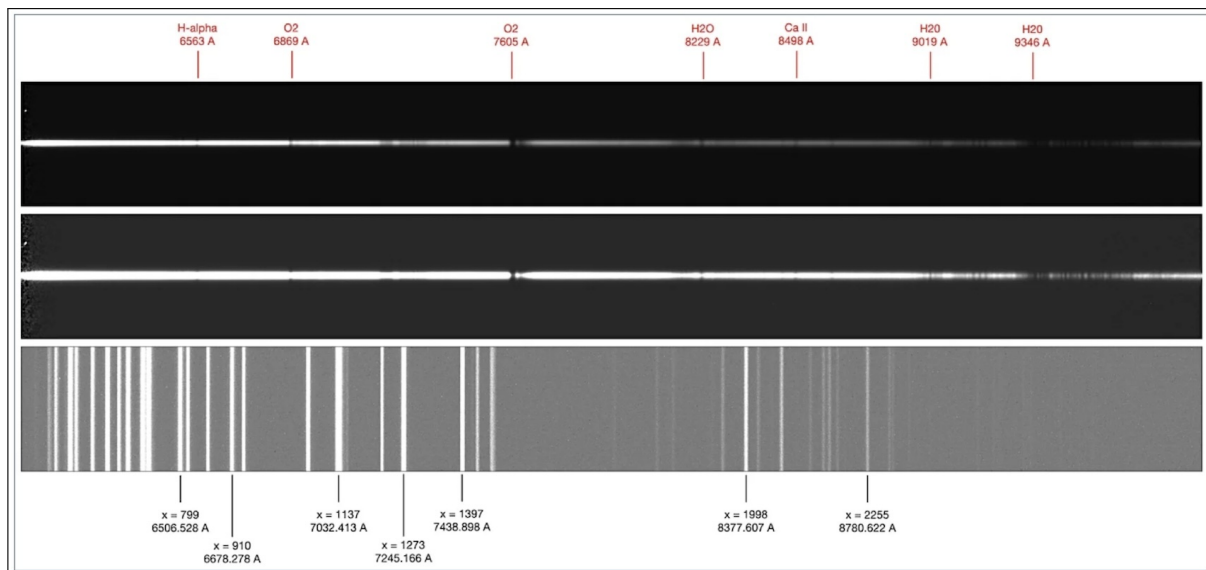


Figure 137 – Example of the infrared spectrum of a star (HD 4915), shown at two contrast levels, together with the neon calibration image in the lower part.

It is important to know how to identify the neon lines, bearing in mind that other gases are sometimes present, in very small quantities, but can still show their spectral signature, such as argon. See Appendix 14 on this point.

The following listing shows a typical example of finding the calibration parameters in the infrared.

```
working_path: D:/antibes_tse_20260709
batch_name: Vega
calib_mode: 0
line_type: external
poly_order: 2

wavelength: [6598.95, 6929.47, 7245.17, 7635.11, 8377.61, 8780.62, 9122.97, 9657.79]
line_pos: [1407, 1583, 1752, 1961, 2357, 2573, 2754, 3037]

# instrumental_response: _rep_vega_ir

bin_size: 40
sky: [210, 30, 30, 210]
posy_exclude: [890, 1080]
xlimit: [1490, 2650]
crop_wave: [6400, 11000]
norm_wave: [6640, 6660]
kernel_size: -3
extract_mode: 1
sky_mode: 0
sigma_gauss: 0.1
Longitude: 7.093
Latitude: 43.587
Altitude: 50
simbad: 1
corr_atmo: 0.13
Site: Antibes Saint-Jean
Inst: T150 + TSE + PL585
Observer: cbuil
```

Here we calibrate in mode 0, which is most often used at high resolution, but which can equally well be used at low spectral resolution. This is the main message of this section.

We proceed in three stages:

- first, computing the dispersion polynomial, working on the assumption that the instrumental response is not yet available. In the listing above, the `instrumental_response` line is disabled. Note that the wavelengths supplied to the parameters do indeed lie in the infrared.
- the resulting profile is carefully cleaned from the “Instr. Resp.” interface using the “Zones” tool. Telluric O₂ and H₂O lines are present in our spectrum but not in the Pickles spectrum. They need to be erased while taking care to preserve the characteristic shape of a response (relatively smooth and monotonic).
- computing the final spectrum by applying the instrumental response computed earlier. The configuration file then becomes a fully-fledged processing file, once the “instrumental_response” parameter is present.

When evaluating the instrumental response at wavelengths as far out as 1 micron, it is not certain that a reference spectrum covering this region can easily be found. Pickles spectra, for example, stop a little short of that, in the red. The spectra of the CALSPEC library (STSci), very widely used in the professional community, resolve this difficulty¹². You will find the spectra of this library in the “calspec_lib” folder of the “_calib” directory installed with the software. Some objects are faint, but others are bright, such as the spectrum of the star Vega, very convenient for broadband calibration, from the UV to the infrared.

12. See: <https://archive.stsci.edu/hlsp/reference-atlases>

Building an inexpensive neon calibration source

The main difficulty in lighting a neon bulb is that it needs to be powered at 220 V. You must NEVER plug such a lamp directly into the mains (even through the essential 10 k Ω resistor), as this is a genuine electrocution risk.

The only really acceptable and convenient solution is to use a small 5 V–12 V to 220 V conversion module (costing only around 10 euros). It is true that there is indeed 220 V on the output line, but at a very low current: if you touch it with your fingers, you will feel something, but you will stay alive!

Figure 138 shows the typical assembly, the components being available from many well-known online suppliers.

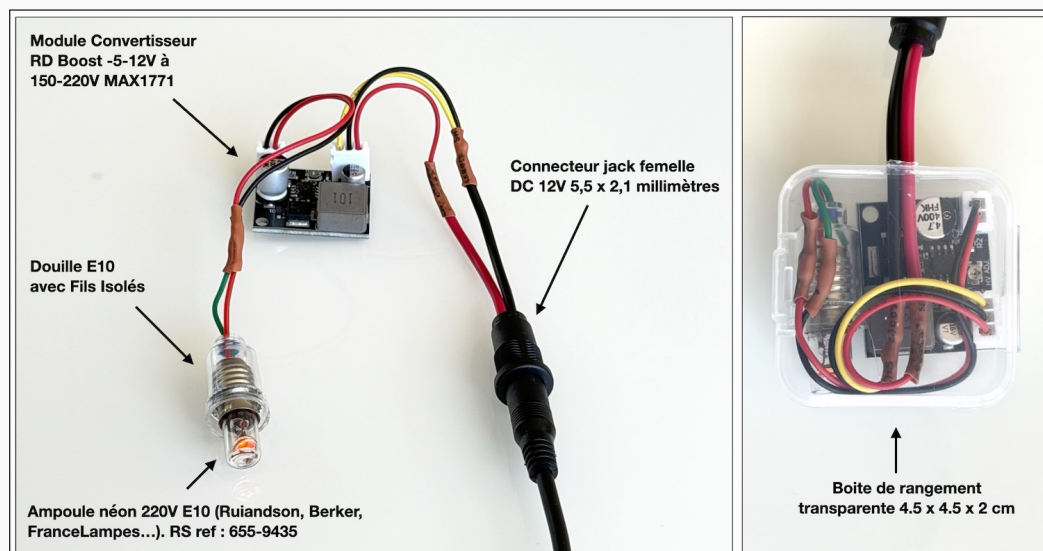


Figure 138 – Components and assembly of a neon-lamp setup. A standard miniature 220 V/12 V power supply feeds the whole assembly. Only four solder joints are needed, and that is all.

The result fits inside a small translucent box (Figure 139), which you can, for example, use to illuminate the telescope aperture by simply holding it by hand.



Figure 139 – The wavelength-calibration lamp, ready for use.

11 Appendix 1: Utility modes

specINTI offers a number of **utility modes**, each a tool making it easier to work with data in various situations. The list of these is given in Table 2.

We have already made extensive use of **mode -1** (activated by writing the line `calib_mode: -1` in the configuration file), which is a powerful way of simply finding the spectral dispersion

law and the instrumental response from a natural source.

Let us now describe the other utility modes available.

11.1 Mode -2

This mode computes the dispersion polynomial from a series of **absorption** lines identified in the spectrum of a star, whose wavelengths are known. Fill in the observation file as usual (star name, number of images, DOF, and so on), then run the program with a configuration file similar to this one:

```
working_path: D:/antibes_tse_20260405
batch_name: test
calib_mode: -2
wavelength: [3889.05, 3970.08, 4101.75, 4340.48, 4861.34, 6562.81]
line_pos: [1003, 1046, 1118, 1250, 1527, 2448]
shift_posx: 0
poly_order: 2
search_wide: 20
bin_size: 30
sky: [160, 20, 20, 120]
xlimit: [1300, 2500]
```

The **wavelength** parameter takes the wavelengths of the lines used (here Balmer-series lines). The **line_pos** parameter takes their approximate X positions in pixels: accuracy to within a few pixels is enough, since **specINTI** determines the centre of each line to sub-pixel accuracy itself. The optional **search_wide** parameter is used to adjust the search tolerance, expressed in pixels.

The **shift_posx** parameter is an option that adds a shift, positive or negative, to the given X coordinates. This makes it possible to reuse an already established list of positions, even if the spectrum currently being processed is slightly shifted.

Do not forget to specify the degree of the polynomial via **poly_order**, then click “Go!”. The calibration-polynomial coefficients and the fitting statistics are shown in the console; you can copy and paste them for use in another configuration file.

Section 9 gives a concrete example of using calibration mode -2.

11.2 Mode -3

This mode is very close to mode -2, except that the lines used are **emission** lines (lines from a gaseous nebula or a spectral lamp, for example).

The example below shows how to determine the dispersion polynomial in the infrared from a set of emission lines of a neon lamp:

```
working_path: D:/antibes_tse_20260523
batch_name: obs_neon
calib_mode: -3
wavelength: [7032.413, 7245.17, 7438.898, 8300.325, 8377.607, 8495.359, 8780.622]
line_pos: [1447, 1565, 1665, 2127, 2168, 2231, 2384]
shift_posx: 0
poly_order: 2
posy: 1220
search_wide: 20
bin_size: 100
sky: [260, 20, 20, 220]
tilt: -0.585
posy_exclude: [1000, 1400]
```

```
sky_remove: 0
```

The centre of the binning zone is given via the `posy` parameter (here $Y = 1220$). The width of this zone is deliberately large (`bin_size: 100`) so as to improve the signal-to-noise ratio of the spectral profile used for line detection.

In this example, the *slant* angle is determined automatically by the software, while the *tilt* angle is fixed at -0.585° .

The `sky_remove` parameter plays an essential role here. By giving it the value 0, `specINTI` is instructed not to subtract the background of the image, an important precaution for accurately measuring the position of the emission lines that sit on top of this background. By default, `sky_remove: 1` results in the estimated sky background being subtracted from either side of the spectral trace.

11.3 Mode -4

This mode performs a least-squares polynomial fit to a series of points defined by the `line_pos` coordinates as a function of the values given in `wavelength`. This is a purely numerical computation, requiring no input spectrum at all. Among other things, this mode can be used to check the consistency of a dataset or to evaluate the coefficients of a fitted polynomial from a known list of points.

Example:

```
working_path: D:/antibes_tse_20260405
batch_name: test
calib_mode: -4
poly_order: 2
wavelength: [3948.979, 4044.418, 4259.362, 5037.751, 5116.590, 5400.562]
line_pos: [307, 369, 508, 1009, 1061, 1244]
```

Once run, the polynomial coefficients and the fitting statistics are shown in the console.

11.4 Mode -5

Mode -5 extracts the raw spectral profile of a star, with intensities expressed in ADU (*Analog-to-Digital Units*) and the spectral axis given in pixels. This is simply a vertical summation of the rows of the image.

The profile is saved in the working directory under the name:

```
star_name_raw.fits
```

In the following example, all the geometric parameters are determined automatically by the software (binning zones, sky-background subtraction), leading to a particularly short configuration file:

```
working_path: D:/antibes_tse_20260405
batch_name: 4Aur
calib_mode: -5
```

12 Appendix 2: Parameter reference manual

This section is the reference manual for the configuration file: every possible parameter is described here.

While not compulsory, you can organise (and write) the configuration file by grouping the parameters into three distinct sections:

- a section devoted to the parameters that change when moving from one observing session to another (typically, the path to the working folder containing the data to be processed);
- a section relating to your preferences for how spectra are processed, preferences that are generally the same from one observing session to another (though they can also be specific to processing a particular object within a single session, though this remains fairly rare);
- a section of parameters relating to how you use your acquisition software or your observatory (FITS file extensions, geographic coordinates). This information can be kept separate, since it has no impact on the quality of your reductions.

It is this three-part organisation that we adopt in this reference manual for describing the parameters.

12.1 Parameters relating to the observing session

BATCH_NAME — Optional parameter. Specifies the name of the observation file (of type `.yaml`) present in the working folder to be processed. It is recommended to set this parameter in the configuration file; if it is absent, `specINTI` assumes the observation file is named `object.yaml`. Example:

```
batch_name:  obs_qrvul
```

WORKING_PATH — Compulsory parameter. Path to the folder containing the images to be processed and the observation file (`.yaml`). This folder is called the “working directory”. Examples:

```
working_path:  /Volumes/Samsung_T5/specINTI_test/300/ (macOS)
```

```
working_path:  c:\observations\nuit_051221\ (Windows)
```

12.2 Parameters relating to spectra

ATMO_BLUR — Optional parameter. If this parameter is set, the molecular atmospheric-transmission file passed to `specINTI` (via the observation file) is subjected to Gaussian filtering (smoothing) with a full width at half maximum (FWHM) equal to the value of this parameter. This option makes it possible to match the spectral resolution of the atmospheric-transmission model (a file of lines) to the resolution of the spectra being processed. Example:

```
atmo_blur:  25
```

AUTO_CALIB — Optional parameter. Performs an automatic search for calibration lines (neon gas only), together with an automatic spectral calibration. This function is suited to high-resolution spectra. It works very well with spectra acquired in lateral mode, or with spectra accompanied by a separate image of the calibration source (external mode). The line search is bounded by the two wavelengths given in the list passed to the parameter. Example:

```
auto_calib:  [6450, 6750]
```

AUTO_CALIB_TH — Optional parameter. Detection threshold for the calibration lines, in ADU, for the `auto_calib` function. Only lines whose peak intensity exceeds this threshold are then used. It is normally unnecessary to add this parameter, since the threshold is in that case

computed automatically by the software; but in rare circumstances it can be useful to set it manually so that the application does not latch onto noise while searching for lines.

BIN_FACTOR — Optional parameter. Binning factor for the points of the final spectrum. For example, if the value of this parameter is 2, points are summed two by two (`specINTI` computes an average value). Spectral binning increases the signal-to-noise ratio when this is low in the initial spectrum, but the resolving power correspondingly decreases (by an amount determined by how the spectrum is sampled by the detector's pixels relative to the spectral finesse — there is not necessarily a direct relationship between the binning factor and the final spectral resolution). Example:

```
bin_factor: 6
```

BIN_SIZE — Optional parameter. Width, in pixels, of the binning zone along the spatial axis of the 2D spectrum, used to build the spectral profile. This width should ideally be large enough to capture essentially all of the object's signal along the spatial (vertical) axis, say 99 % of it, so as to avoid photometric errors. Too large a binning-zone width can add noise (to a lesser extent if the optional `extract_mode` parameter is set to 1). If the `bin_size` parameter is omitted from the configuration file, the application computes a reliable, optimal value for you from an analysis of your spectral trace, and this is very often the best way to proceed. Example:

```
bin_size: 24
```

CALIB_COEF — Compulsory parameter. Terms of the spectral dispersion polynomial, as a list starting with the highest-degree term, for example [A4, A3, A2, A1, A0]. `specINTI` determines the degree of the polynomial from the number of terms supplied. Example:

```
calib_coef: [-4.82275e-10, 4.5412e-6, 0.96946, 3771.874]
```

CALIB_MODE — Compulsory parameter. Adapts the spectral calibration approach to the resolution of the spectra. The value is 0 for high-spectral-resolution spectra and 1 for low-spectral-resolution spectra. Example:

```
calib_mode: 1
```

CLEAN_WAVE — Optional parameter. List of wavelengths around which the software performs a linear interpolation, over a width given by the corresponding `clean_wide` parameter. This function is used to erase certain artefacts, or telluric lines identified by their wavelength, in angstroms. Example:

```
clean_wave: [6543.9, 6552.5]
```

CLEAN_WIDE — Optional parameter. List of spectral widths, in angstroms, at the ends of which the software performs a linear interpolation to erase a detail of the processed spectral profile. This parameter is used together with the `clean_wave` parameter. Example:

```
clean_wide: [2.0, 2.4]
```

CORR_ATMO — Optional parameter. If this parameter is present, the software computes the non-molecular atmospheric transmission and applies it to the processed spectrum in order to obtain an extra-atmospheric profile. The value of the parameter is that of the AOD (aerosol optical depth, reflecting the transparency of the atmosphere). For reference, very dry mountain air corresponds to an AOD of 0.02, and a dry desert to an AOD of 0.04. In France, the AOD is typically around 0.07 in winter, 0.21 in summer, and 0.13 as an annual average. In hot, stormy weather it can reach 0.50.

The equatorial coordinates of the object are essential for computing its altitude above the horizon. Normally, these coordinates are found in the header of the files, under the keys `CRVAL1` and `CRVAL2`. If these keys are not set, the computation does not proceed, unless a lookup of these coordinates is implicitly requested from the SIMBAD database, via an internet query. Three conditions must then be met: (1) the `simbad` parameter with the value 1 is present in the

configuration file; (2) an internet connection is available; (3) the object's name is recognised in the SIMBAD database. The lookup of the equatorial coordinates is then automatic and transparent. Finally, recall that, if absolutely necessary, it is possible to manually set the `CRVAL1` and `CRVAL2` keys using the configuration-file function mechanism (the `_img_add_item_float` function). Example:

```
simbad: 1
corr_atmo: 0.13
```

CORR_ATMO2 — Optional parameter. Function similar to `corr_atmo`, but with two arguments, in order: the AOD value, then the altitude of the object, in degrees. In this case it is no longer necessary to supply the equatorial coordinates of the object or the geographic coordinates, but the angular elevation of the object must have been established some other way. Example:

```
corr_atmo2: [0.13, 42.7]
```

CORR_TOPO — Optional parameter. If this parameter is present, the software computes the observer's topocentric radial velocity relative to the object. The computation takes into account the observer's geographic coordinates on the Earth's surface (the `longitude`, `latitude` and `altitude` parameters then become compulsory), as well as the Earth's motion around the Sun. The internal precision of the computation is of the order of a centimetre per second, and is therefore very accurate.

The value of the parameter is, in addition, the object's own radial velocity, in km/s (information available from databases such as SIMBAD, but which must be supplied explicitly). If this velocity is not known, enter the value 0. Following the SIMBAD convention, the radial velocity is positive if the object is receding, and negative otherwise.

In our definition, the computed overall radial velocity (topocentric velocity added to the object's own radial velocity) is positive if the object is receding from the Earth, and negative if it is approaching. The software then corrects the wavelengths of the spectrum so as to bring the object back to rest (*rest wavelength*).

The equatorial coordinates of the object are essential for computing the topocentric velocity. Normally, these coordinates are found in the header of the files, under the keys `CRVAL1` and `CRVAL2`. If these keys are not set, the computation does not proceed, unless a lookup of these coordinates is implicitly requested from the SIMBAD database, via an internet query. Three conditions must then be met: (1) the `simbad` parameter with the value 1 is present in the configuration file; (2) an internet connection is available; (3) the object's name is recognised in the SIMBAD database. The lookup of the equatorial coordinates is then automatic and transparent. Finally, recall that, if absolutely necessary, it is possible to manually set the `CRVAL1` and `CRVAL2` keys using the configuration-file function mechanism (the `_img_add_item_float` function). Example:

```
simbad: 1
corr_topo: -12.45
```

CROP_WAVE — Optional parameter. Isolates a portion of the final computed spectrum, delimited by two wavelengths $[\lambda_1, \lambda_2]$. This parameter can, for example, be used to extract the part of the spectrum considered meaningful. Example:

```
crop_wave: [3800, 7500]
```

EXPAND — Optional parameter. If this parameter is set, at the end of processing, `specINTI` generates a new 2D image representing the spectral profile expanded along the vertical axis, over the number of rows given as the parameter's value (the profile is thus duplicated x times). Two images are in fact produced, in FITS format and in PNG format. Example:

```
expand: 800
```

EXTRACT_MODE — Optional parameter. If the value is zero (the default), **specINTI** extracts the raw spectral profile from the 2D images by performing a simple column-by-column summation, over the width **bin_size**. This is a quick method, suitable for objects giving a spectral trace with a good or intermediate signal-to-noise ratio. If the spectrum’s signal is very weak relative to the noise, it is preferable to use an optimal extraction method, which significantly reduces the noise in the profile by performing a weighted summation. The algorithm also removes cosmic-ray hits present within the binning zone, by means of an iterative process (Keith Horne, 1986, *An Optimal Extraction Algorithm for CCD Spectroscopy*, PASP vol. 98, p.609). Optimal extraction is enabled by giving this parameter the value 1. Example:

extract_mode: 1

HOT_PIXELS — Optional parameter. Removes hot pixels in the dark-signal image. A pixel is considered hot if its intensity exceeds the value given as the parameter, expressed in ADU. The pixel’s intensity is then replaced by the average intensity of its 8 neighbours. Example:

hot_pixels: 6500

HOT_PIXELS2 — Optional parameter. Removes hot pixels in each of the spectral images of a sequence. A pixel is considered hot if its intensity exceeds the value given as the parameter, expressed in ADU. The pixel’s intensity is then replaced by the average intensity of its 8 neighbours. Example:

hot_pixels2: 50000

INSTRUMENTAL_RESPONSE — Optional parameter. Name of a FITS file, of spectral-profile type, containing the instrumental response of the full instrumental chain (telescope, spectrograph, detector). This file must be located in the working directory of the session being processed. Once this parameter is used, the extracted spectrum is corrected for the instrumental response. Example:

instrumental_response: reponse_starex300

KERNEL_SIZE — Optional parameter. **specINTI** offers the option of applying median-type filtering to the 2D images before extracting the spectral profile. This filter is particularly useful with CMOS sensors, which are subject to impulsive noise (known as *telegraph* noise, or RTS, or “salt and pepper” noise). The value of the parameter is the size of the convolution kernel; it must be odd. For example, the value 3 corresponds to a convolution kernel of 3×3 pixels. It is not recommended (or useful) to exceed a value of 5.

If you supply a negative value, the software also performs median filtering, but with an optimised weighting algorithm, both effective against impulsive noise and more respectful of image detail than classic median filtering. Example:

kernel_size: -3

LINE_POS — Compulsory parameter for certain calibration modes. Specifies the pixel coordinates, in the spectral image along the spectral axis, of the lines whose wavelengths are otherwise given by the **wavelength** parameter. If the calibration lines show a marked *slant* (tilt), make sure to take the position measurements at the point where they cross the spectral trace.

Do not worry too much about the precision of these readings: by default, the tolerated error is ± 25 pixels. You can change this uncertainty range using the **search_wide** parameter (to reduce or increase it). Example:

line_pos: [237, 450, 633]

LINE_TYPE — Compulsory parameter. Tells the software the nature of the line spectrum to use for spectral calibration. Depending on the context (the value of **calib_mode**), the possible options are:

- **none**: no calibration spectrum used.

- **external**: a calibration spectrum separate from the science targets is used for calibration.
- **lateral**: a calibration spectrum embedded within the science spectrum is used.
- **emission**: emission lines present in the science spectrum are used for wavelength calibration.
- **absorption**: absorption lines present in the science spectrum are used for wavelength calibration.

NORM_WAVE — Optional parameter. If this parameter is set in the configuration file, the final spectrum is normalised to unity, using the mean of the intensities within a spectral region determined by two wavelengths given as the parameter $[\lambda_1, \lambda_2]$. Example:

`norm_wave: [6600, 6620]`

PLANCK — Optional parameter. If this parameter is set in the configuration file, the final spectrum is multiplied by the Planck profile computed for the temperature, in kelvins, given as the parameter. Example:

`planck: 2900`

POLY_ORDER — Compulsory parameter when calibrating the instrument in wavelength. Sets the order of the dispersion polynomial determined from a set of calibration lines. If n is the number of lines used, the order of the polynomial cannot exceed $n - 1$.

`poly_order: 3`

POSY — Optional parameter. Under normal circumstances, **specINTI** automatically computes the vertical (Y-axis) coordinate of the spectral trace (a value specific to each image of a sequence). This is what happens if **posy** is absent from the configuration file, which is a simplification for the user. It can nevertheless happen that the trace is almost imperceptible, or that you do not want the software to perform this computation automatically (when processing the spectrum of nebulae, for example). In that case, set the **posy** parameter in the configuration file, specifying the Y coordinate of the spectrum: **specINTI** then uses this coordinate as the basis for every image in the sequence.

If the value of **posy** is negative, the software assumes the spectral trace lies vertically, approximately, at the Y coordinate given (in absolute value), but also recentres every spectrum within the binning limit (the **bin_size** parameter). Using the **posy** parameter with a negative value is a powerful option, allowing you to get out of tricky situations, such as extracting the spectrum of a faint object whose trace lies close to that of a much brighter object, while still allowing vertical recentring of all the traces in a sequence (within the limit of **bin_size**).

Be careful not to overuse **posy**: its use should remain a last resort, since the computation performed automatically by **specINTI** is always correct and accurate in the overwhelming majority of situations. You can therefore omit this parameter in normal processing situations. Example:

`posy: 376`

POSY_EXCLUDE — Optional parameter. Defines an exclusion zone for computing the vertical Y coordinate of a spectral trace. The argument of this parameter is a pair of coordinates (Y1, Y2) delimiting a band within which the search is carried out; parts of the image lying outside this band are not used. This parameter is useful in particular applications where certain parts of the spectrum can disturb the search for the trace, for example when using a photometric slit whose widened part can be a source of trouble. Example:

`posy_exclude: [250, 600]`

RATIO — Optional parameter. If the value of this parameter is 1, once processing is finished, the ratio of the spectra of the first two objects supplied in the observation file's "object list" is computed. The result is a profile file written to the working directory, named after the profile of the first spectrum, with the suffix "**_ratio**" appended.

This computation is ignored if there is only a single object in the “object list”. The default value is 0 (the ratio is not computed). Example:

```
ratio: 1
```

SEARCH_WIDE — Optional parameter. Changes the tolerance on the horizontal coordinates of the lines given via the parameter `line_pos`. By default, the uncertainty is ± 25 pixels, which corresponds to a value of 50 for this parameter. You can, if needed, increase this value if the calibration lines are well separated from one another, or, on the contrary, reduce it if the lines are close together, with a risk that the software might confuse them. Example:

```
search_wide: 30
```

SEQ_MODE — Optional parameter. If the line `seq_mode: 1` is added to the configuration file, when processing a sequence of images of the spectrum of the same object, in addition to producing a spectral profile representing the sum of the individual spectra, `specINTI` saves, in the working directory, the fully processed spectra derived from the individual spectral images (that is, as many spectra as there are images). The individual profiles are distinguished by the @ character prefixed to the file name. This feature is extremely valuable for studying objects whose spectrum changes rapidly over time. This function is disabled by leaving out this parameter, or by writing: `seq_mode: 0` (the default value). Example:

```
seq_mode: 1
```

SIGMA_GAUSS — Optional parameter. Applies Gaussian filtering to the 2D spectral images if the value of the parameter is non-zero. The higher the value, the stronger the filtering. This smoothing reduces noise, but can lead to a reduction in the final resolving power. A value between 0.5 and 1.5 is typical; it should be adjusted to find the best compromise between noise reduction and preservation of spectral fineness. Example:

```
sigma_gauss: 1.0
```

SHIFT_POSX — Optional parameter. Shifts every element of a `POSX` list by the value given as the argument. Example:

```
shift_posx: -10
```

SNR — Optional parameter. Returns, in the output console, the signal-to-noise ratio (SNR) obtained at the end of processing a spectrum. The computation zone, within the processed spectral profile, lies between λ_1 and λ_2 (in angstroms). Ideally, this zone should lie within the continuum of the spectrum, in a part free of spectral lines. Before computing the noise needed to evaluate the signal-to-noise ratio, `specINTI` subtracts the best-fit straight line through all the points in the zone, so as to reduce the bias introduced by a slow variation of the spectral profile within that zone. Example:

```
snr: [6550, 6560]
```

SKY — Optional parameter. Set of pixel coordinates, along the spatial (vertical) axis, delimiting two zones for computing the sky-background signal, on either side of the spectral trace. These coordinates are relative to the vertical coordinate of the spectral trace. The format of the parameter is `[y1, y2, y3, y4]`. The values (y1, y2) are associated with the lower part of the trace, while the values (y3, y4) relate to the upper part. Example:

```
sky: [180, 30, 30, 180]
```

The coordinates given can perfectly well be asymmetric relative to the spectral trace, for example:

```
sky: [110, 25, 30, 180]
```

If the `sky` parameter is absent from the configuration file, the software takes care of finding values that are, most of the time, satisfactory.

But if you do need to define these sky-background computation bands yourself, do so with care.

The width of these bands should typically be at least 100 pixels, in order to obtain an excellent sky-background value. Moreover, these coordinates are used repeatedly within `specINTI`, not just for computing the sky background. The coordinates (y2, y3) define a zone within which optimal filtering is requested (see the `kernel_size` parameter). The coordinates y1 and y4 are also used to estimate the `slant` (tilt) angle of the lines, a geometric defect (see also the `slant` parameter).

SKY_MODE — Optional parameter. Choice of method used to remove the sky background (see also the `sky` parameter). If the value is 0 (the default), for each column, the software independently computes the median intensity within the sky-background evaluation zones and subtracts this value from the column. If the value is 1, the software fits a Legendre polynomial to the sky-background intensity within the evaluation zones, then subtracts the interpolated value from the spectral trace along the column concerned. This latter technique is generally more accurate, but also slower to run. Example:

```
sky_mode: 1
```

SKY_REMOVE — Optional parameter. If the value is 0, the sky background is not removed from the images (useful for certain observations of nebulae, for example). If the value is 1 (the default, and practically always used), the sky background is removed (see the `sky_mode` parameter). Example:

```
sky_remove: 1
```

SLANT — Optional parameter. Normally, the *slant* (tilt) angle of the spectral lines is determined automatically by `specINTI`, which uses it to straighten the lines so that they are properly vertical. You can, however, force the value of this angle, in degrees, via the `slant` parameter. Example:

```
slant: -0.51
```

SMILE_RADIUS — Optional parameter. The *smile* is a geometric distortion of the spectral lines which, instead of being straight, take on a curved shape, with a radius of curvature R in pixels — the value of the parameter. When the radius of curvature of the lines is set via `smile_radius`, `specINTI` corrects the distortion in the 2D images so that the lines are properly straight in the end.

A tip: if the `check_mode` parameter is set to 1, the software produces, in the working directory, an image `_step5.fits` showing the shape of the calibration lines before correction of the smile effect, and an image `_step71.fits` after correction. Comparing these two images makes it possible to adjust the value of the radius of curvature, together with its sign (concave or convex curvature relative to the wavelength axis). Example:

```
smile_radius: -18500
```

SPECTRAL_SHIFT_WAVE — Optional parameter. Shifts the final spectral profile by a given amount, in angstroms. Can, for example, correct a systematic calibration error.

```
spectral_shift_wave: 0.24
```

SPECTRAL_SHIFT_VEL — Optional parameter. Shifts the final spectral profile by a given amount, in km/s. Can, for example, correct a Doppler shift.

```
spectral_shift_vel: -22.3
```

TILT — Optional parameter. Normally, the *tilt* angle of the spectral trace is determined automatically by `specINTI`, which uses it to bring the trace back to horizontal (this is in any case recommended at the acquisition stage). You can, however, force the value of this angle, in degrees, via the parameter `tilt`. Example:

```
tilt: -1.32
```

XLIMIT — Compulsory parameter. Set of two values in pixels delimiting a region of the trace within which the software performs a number of calculations to extract the spectral profile. The format is `[x1, x2]`, `x1` and `x2` being the limits bracketing a part of the trace considered intense enough for the calculations to run correctly. The choice is flexible: the trace simply needs to be perceptible there, and relatively narrow. Typically, the value of $x_2 - x_1$ should exceed 200 pixels, but can easily reach 2000 pixels. Example:

```
xlimit: [452, 1966]
```

WAVELENGTH — Compulsory parameter for wavelength calibration of the instrument. Set of wavelengths of the calibration lines, allowing **specINTI** to compute the dispersion polynomial (the relation between the pixel index along the spectral axis and wavelength). This parameter is closely linked to the **line_pos** parameter, which gives the approximate positions of the corresponding lines in the 2D images.

You can give a single line or a list of many lines. Examples:

```
wavelength: [6532.88]
```

```
wavelength: [6506.53, 6532.88, 6598.95, 6678.28]
```

WIDE_SOURCE — Optional parameter. If the value of this parameter is 1, **specINTI** switches to a simplified computation mode, suited to objects with an extended surface (in particular, the sky-background parameters are then not required). The default value is 0. Example:

```
wide_source: 1
```

12.3 Operational parameters

ALTITUDE — Optional parameter. Altitude of the observing site, in metres. This information is written into the FITS header of the processed spectrum. Example:

```
altitude: 105
```

CHECK_MODE — Optional parameter. Determines the nature of the information displayed on screen (console) and written to the working directory during processing. With the value 0 (default), the information provided is limited to the strict essentials. With the value 1, **specINTI** provides extra information on each step of the processing and writes the corresponding image and spectrum files to the working directory (reserved names `_stepxxx` and `_profilexxx`). This option is particularly useful for checking that processing is running correctly, spotting possible errors, or helping a user who is having trouble (a kind of support tool!). Example:

```
check_mode: 1
```

FILE_EXTENSION — Optional parameter. Choice of extension for FITS files. With the value 0 (default), the extension is “.fits”; with the value 1, it is “.fit”. Example:

```
file_extension: 1
```

INST — Optional parameter. Description of the observing equipment. This information is written into the FITS header of the processed spectrum. Example:

```
inst: T250+Solex(300)+ASI183MM
```

LATITUDE — Optional parameter. Latitude of the observing site, in degrees. This information is written into the FITS header of the processed spectrum. Example:

```
latitude: 34.5701
```

LONGITUDE — Optional parameter. Longitude of the observing site, in degrees (positive east of Greenwich). This information is written into the FITS header of the processed spectrum. Example:

```
longitude: -1.4677
```

MASTER_PATH — Optional parameter. Gives the path of a directory containing DOF frames whose names begin with “_”. The images found there are used whenever they are referenced, within `specINTI`, with this same prefix “_”. Example:

```
master_path:  d:/my_master_directory
```

NEAR_STAR — Optional parameter. Name of an object located close to the target under study. If this parameter is set, `specINTI` uses the equatorial coordinates of this object rather than those associated with the catalogue name of the target itself. This option is useful when the target has a name not recognised by SIMBAD (an asteroid, a nova, and so on): it still allows the atmospheric transmission and the topocentric velocity to be correctly corrected, since the reference object is close to it in the sky. Example:

```
near_star:  SAO 16325
```

POWER_RES — Compulsory parameter when no monochromatic emission lines are available to evaluate the instrument’s resolution. Gives the spectral resolving power. This information is written into the FITS header of the processed spectrum. Example:

```
power_res:  15000
```

SEQ_MODE — Optional parameter. Enables writing, to disk, in the working directory, of the individual processed spectra of a sequence. Files are named following the format `@object-xxx` if the value of the parameter is 1, and `_@object_YYYYMMDD_FFFF` if it is 2. Example:

```
seq_mode:  1
```

OBSERVER — Optional parameter. Name of the observer. This information is written into the FITS header of the processed spectrum. Example:

```
observer:  C. Buil
```

SIMBAD — Optional parameter. With the value 1, the software queries the SIMBAD database to obtain the equatorial coordinates of the object processed, if the keywords `CRVAL1` and `CRVAL2` are not set in the FITS file headers. These coordinates are needed whenever the `corr_atmo` and `corr_topo` parameters are enabled. Example:

```
simbad:  1
```

SITE — Optional parameter. Name of the observing site. This information is written into the FITS header of the processed spectrum. Example:

```
site:  Antibes St Jean
```

VERSION — Optional parameter. With the value 1, returns the version of the `specINTI` software; the program then stops running, after displaying the version number. Example:

```
version:  1
```

13 Appendix 3: Function reference manual

Functions are one-line commands inserted into the configuration file. When such a function is present (its name begins with the character “_”), the associated code is executed, and the program then stops.

_version:

Returns the current version of `specINTI`.

_img_add: [in1, in2, out]

Adds the images `in1` and `in2`. The result is the image `out`.

_img_add_item_float: [in, item, value, out]

Adds an item of type `float` to the header of a FITS image.

_img_add_item_int: [in, item, value, out]

Adds an item of type `int` to the header of a FITS image.

_img_add_item_str: [in, item, value, out]

Adds an item of type `string` to the header of a FITS image.

_img_compute_smile: [x1, y1, x2, y2, x3, y3]

Computes the radius of curvature and the centre of a circle from the coordinates of three points lying on that circle. This function is useful for determining the radius of curvature of a line (whose shape is treated as a circle), so as to correct this distortion and make the line straight (that is, with an infinite radius of curvature). The result is given in the output console. Example:

_img_compute_smile: [1669, 23, 1666, 414, 1669, 678]

_img_fill: [in, x1, x2, out]

Sets to zero the parts of image `in` lying between $x = 0$ and $x = x1$ on one side, and between $x = x2$ and the maximum width of the image on the other. The result is the image `out`.

_img_make_offset: [in, out]

Generates an image `out` whose intensity at every pixel equals the average intensity of image `in`. This function is particularly useful for generating a uniform, noise-free offset-signal image from a single acquired offset image (which is itself already very uniform, apart from noise, with modern CMOS sensors).

_img_merge: [in1, in2, threshold, out]

Merges two images, one briefly exposed (`in1`) and the other long-exposed (`in2`), using the HDR technique. The `threshold` parameter defines the saturation level in image `in2`.

_img_offset: [in, offset, out]

Adds a constant `offset` to image `in`. The result is image `out`.

_img_sub: [in1, in2, out]

Computes the difference of images `in1` and `in2`. The result is image `out`.

_img_uniform: [in, value, out]

Generates image `out`, of the same format as image `in`, in which every pixel has the intensity `value`.

_img_smile: [in, radius, y0, out]

Applies a “smile”-type distortion to an image, with a radius of curvature `radius` (in pixels) and `y0`, the height of the centre of curvature (in pixels).

_pro_blur: [in, coef, out]

Filters a profile using the SAVGOL algorithm, whose smoothing strength is set by the value `coef`.

`_pro_clean: [in, w1, w2, out]`

Performs a linear interpolation between wavelengths `w1` and `w2`. This function can be called several times by writing `_pro_clean1`, `_pro_clean2`, and so on.

`_pro_div: [in1, in2, out]`

Divides profile `in1` by profile `in2`. The result is profile `out`.

`_pro_fit: [in, w1, w2, w3, w4, out]`

Smooths profile `in` using a parabolic function passing as closely as possible through wavelengths `w1`, `w2`, `w3` and `w4` (in Å). The smoothed result is profile `out`.

`_pro_fit2: [in, w1, w2, w3, w4, w5, out]`

Smooths profile `in` using a degree-3 polynomial function passing as closely as possible through the 5 given wavelengths. The smoothed result is profile `out`.

`_pro_fit3: [in, w1, w2, w3, w4, w5, w6, out]`

Smooths profile `in` using a degree-3 polynomial function passing as closely as possible through the 6 given wavelengths. The smoothed result is profile `out`.

`_pro_gauss: [in, sigma, out]`

Convolve profile `in` with a Gaussian function of width `sigma` (in Å). The result is profile `out`.

`_pro_norm: [in, w1, w2, out]`

Normalises input profile `in` to unity, using the mean computed between wavelengths `w1` and `w2`. The result is profile `out`.

`_pro_offset: [in, offset, out]`

Adds a constant `offset` to profile `in`. The result is profile `out`.

`_pro_shift_vel: [in, v, out]`

Applies a velocity-based spectral shift `v` (in km/s) to every point of profile `in` to produce profile `out`.

`_pro_shift_wave: [in, w, out]`

Applies a wavelength spectral shift `w` (in Å) to every point of profile `in` to produce profile `out`.

`_pro_snr: [in, lamb1, lamb2]`

Returns, in the output console, the signal-to-noise ratio (SNR) of spectral profile `in`, computed between the limits `lamb1` and `lamb2` (in Å).

`_pro_telluric: [atmo, in, out]`

Removes the telluric lines from a spectrum. `atmo` is the name of a FITS file containing the molecular spectrum of the Earth's atmosphere — a synthetic spectrum, taken from those contained in the `atmo_molecular` archive (Section 7.3), whose data comes from a reformatted version of the ESO *Sky Model Calculator* (<https://www.eso.org/observing/etc/bin/gen/form?INS.MODE=swspectr+INS.NAME=SKYCALC>). The argument `in` is the name of the starting spectral profile, and `out` is that same profile once the telluric lines have been removed. This function

also saves, in the working directory, the atmospheric spectrum `@atmo`, fitted by least squares to the observed spectrum. See also Section 6.6.

Tip: it is possible to run several functions one after another by enclosing them within the keywords `_begin:` and `_end:`. Here, for example, is a configuration file that runs three functions in sequence in order to evaluate the instrumental response at high spectral resolution without going through a graphical interface (though naturally with some limitations). In order: smoothing a reference spectrum ("`_ref_epscas`"), then dividing an observed spectrum by this reference spectrum, and finally fitting the response obtained with a degree-3 polynomial function (smoothing the response, using wavelengths that avoid those of the spectral lines). Note the use of intermediate images ("`tmp1`", "`tmp2`"), leading to the desired result, the image named here "`_resp412`".

```
# -----
# Working directory
# -----
working_path: D:/starex412

# -----
# Computing the instrumental response
# -----
_begin:

_pro_gauss: [_ref_epscas, 70, tmp1]
_pro_div: [_epscas_20240902_993, tmp1, tmp2]
_pro_fit3: [tmp2, 6505, 6530, 6540, 6584, 6625, 6690, _rep412]

_end:
```

14 Appendix 4: Reference spectra

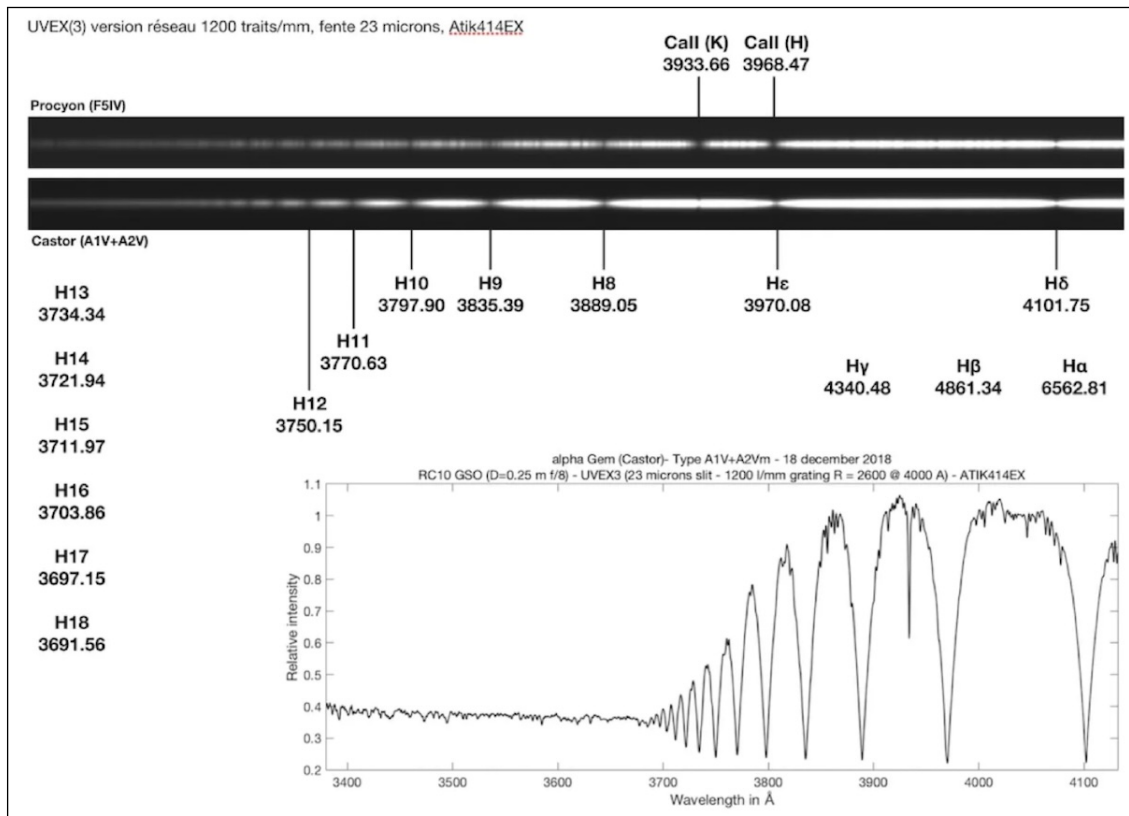


Figure 140 – The hydrogen Balmer series.

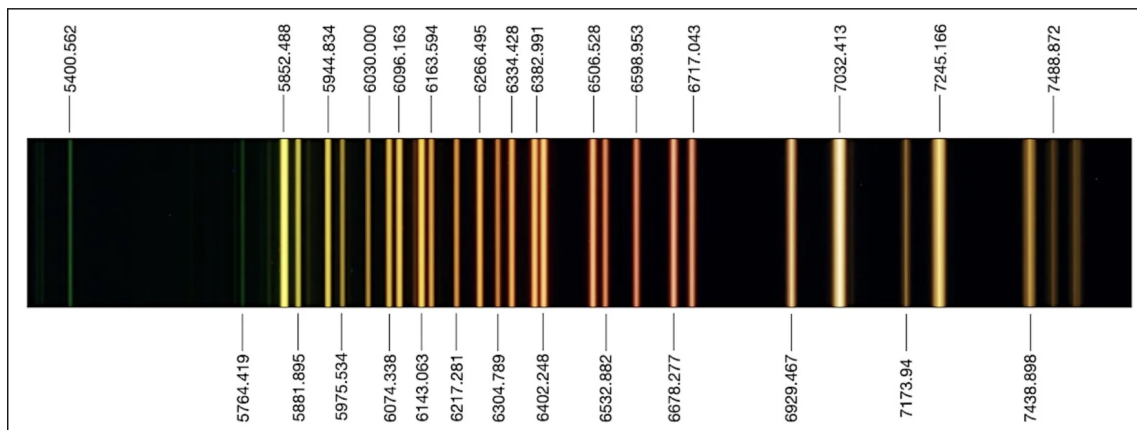


Figure 141 – The neon spectrum (visible part).

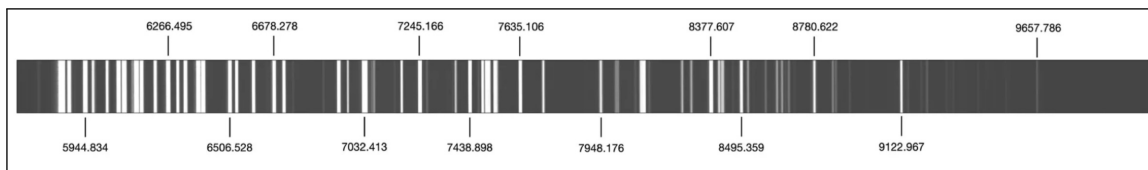


Figure 142 – The neon/argon spectrum (infrared part). The lines at wavelengths 7635 Å, 8781 Å and 9123 Å are associated with argon. Note also the presence of an argon line at 10470.054 Å, which can be useful for calibrating the deep-infrared part of the spectrum. Argon is present here only as a trace element, which makes it very convenient for calibrating the infrared part of the spectrum (some bulbs contain no argon at all, this gas being regarded by the manufacturer as a pollutant, which is a shame).

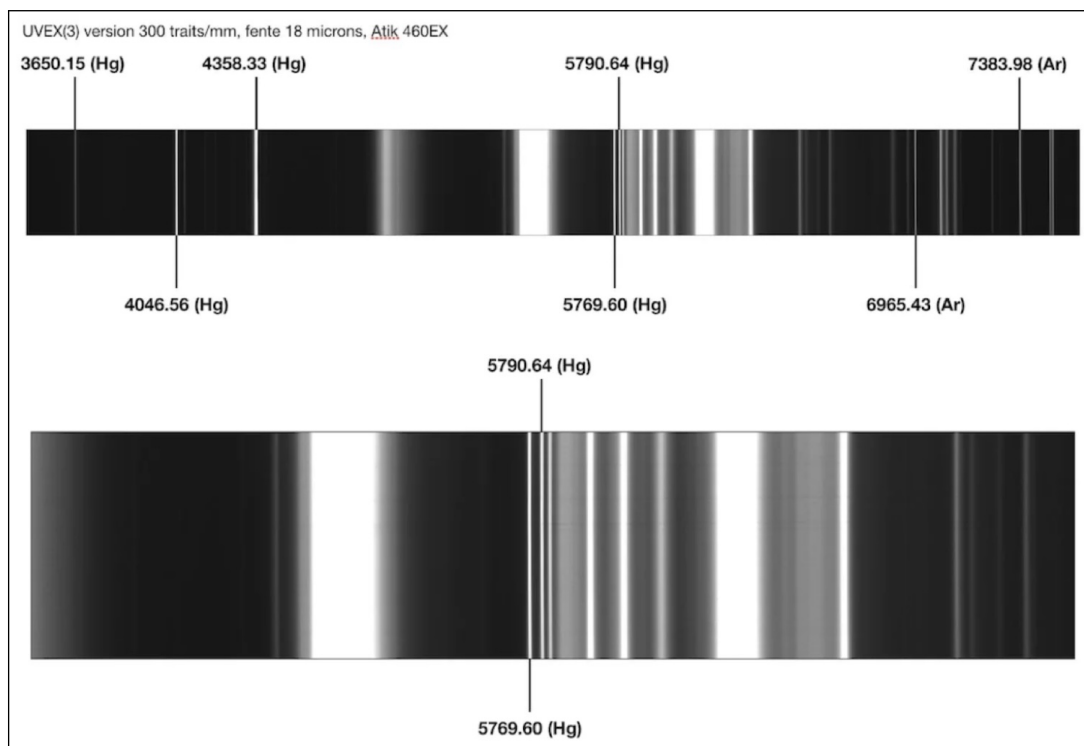


Figure 143 – Spectrum of a compact fluorescent lamp (mercury lines).

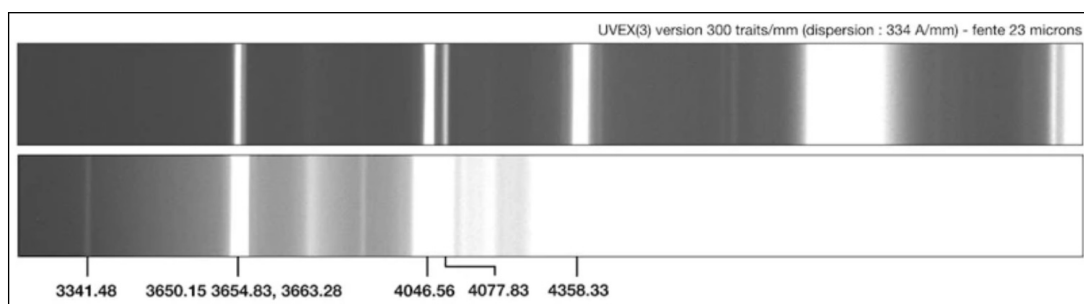


Figure 144 – Spectrum of a compact fluorescent lamp, ultraviolet region (mercury lines).

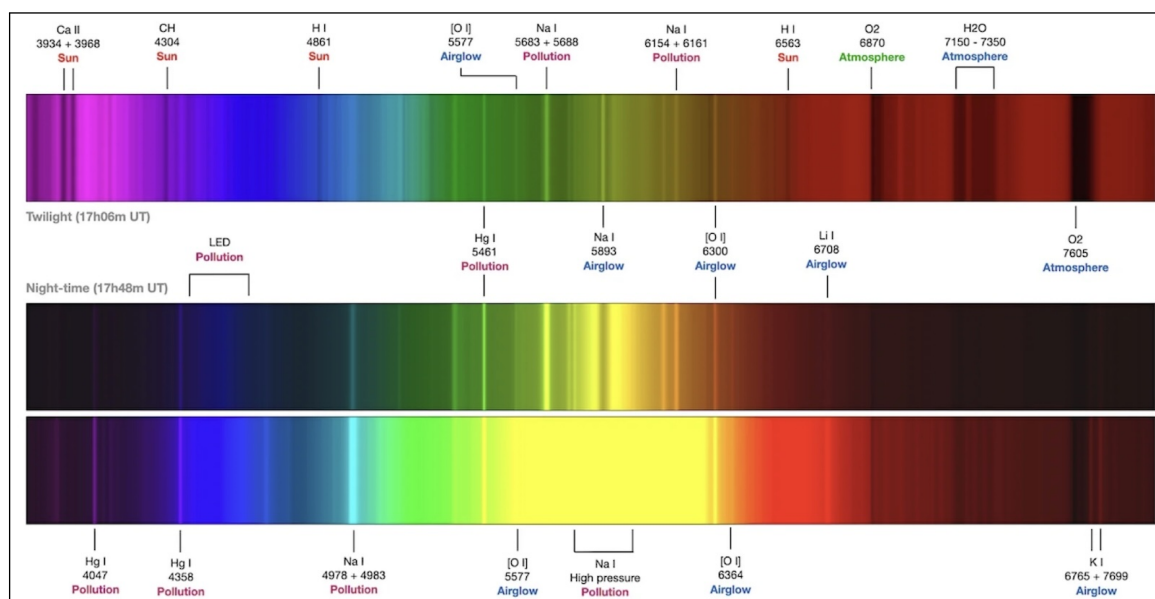


Figure 145 – Solar twilight spectrum and light pollution.

