TripleSpec Focus Calculator

Release 1.0.3.dev3+g4bd224d.d20230803

Simón Torres

Aug 03, 2023

HOW TO INSTALL

ONE

TEST COVERAGE

TWO

OVERVIEW

This is a tool to obtain the best focus value for the Slit Viewer camera of the TripleSpec NIR spectrograph. More information on TripleSpec can be found at NOIRLab Official Website

TripleSpec is a fixed configuration, this means that it has a fixed focus with respect to the telescope, but there is a slit viewing Camera that need focus adjustment.

A slit viewing image looks like this:
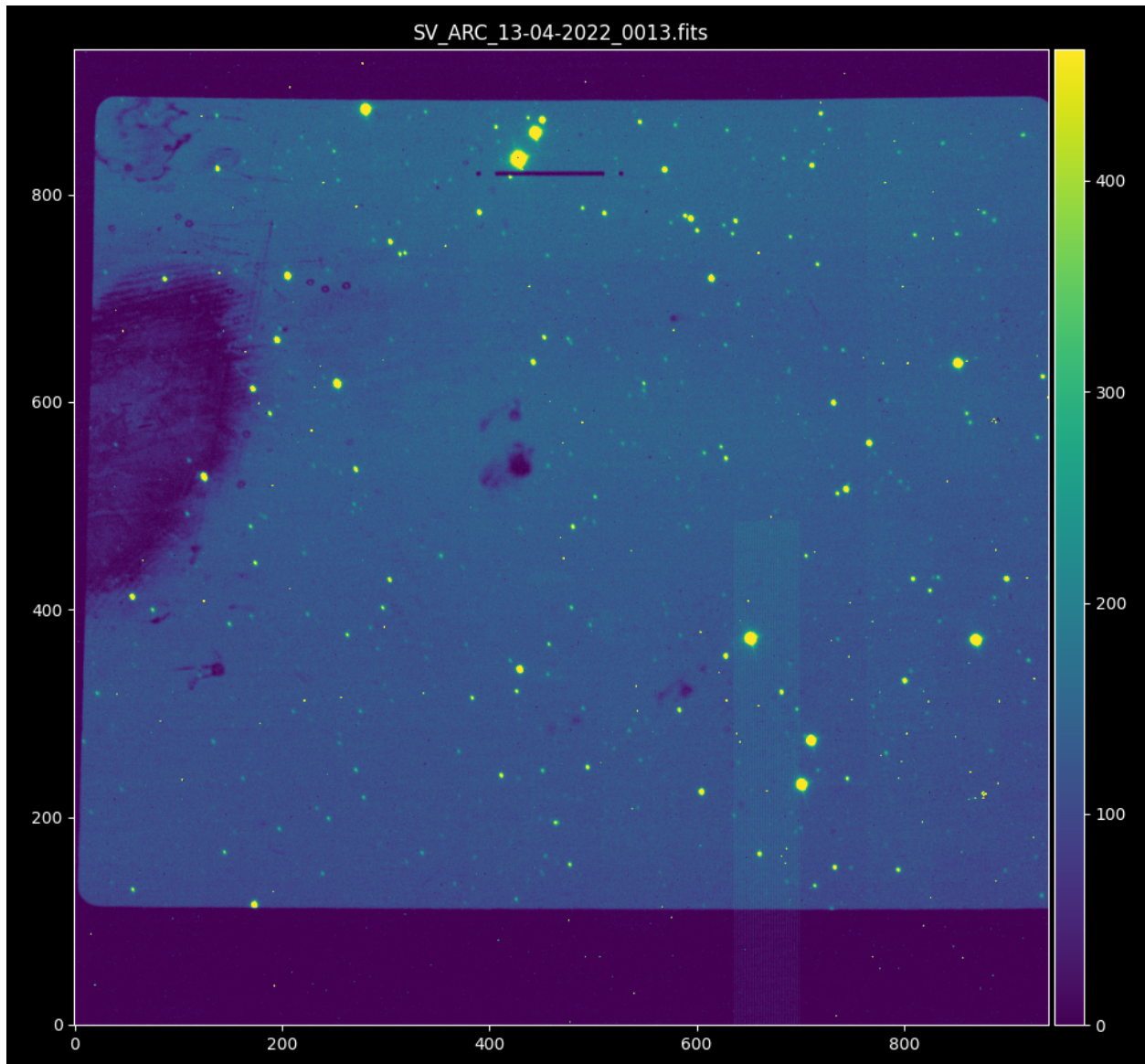
The slit is visible near the top center.

2.1 Install with pip

Note: We recommend using a virtual environment management system such as astroconda

To install you need to have at least python 3.8, it is tested with python 3.9 and 3.10 as well.

Just run the following command in a terminal.

```
pip install triplespec-focus
```

2.2 Data and Process Overview

You have seen already a sample of the data. You will get typically 10 to 15 images taken at different focus values.
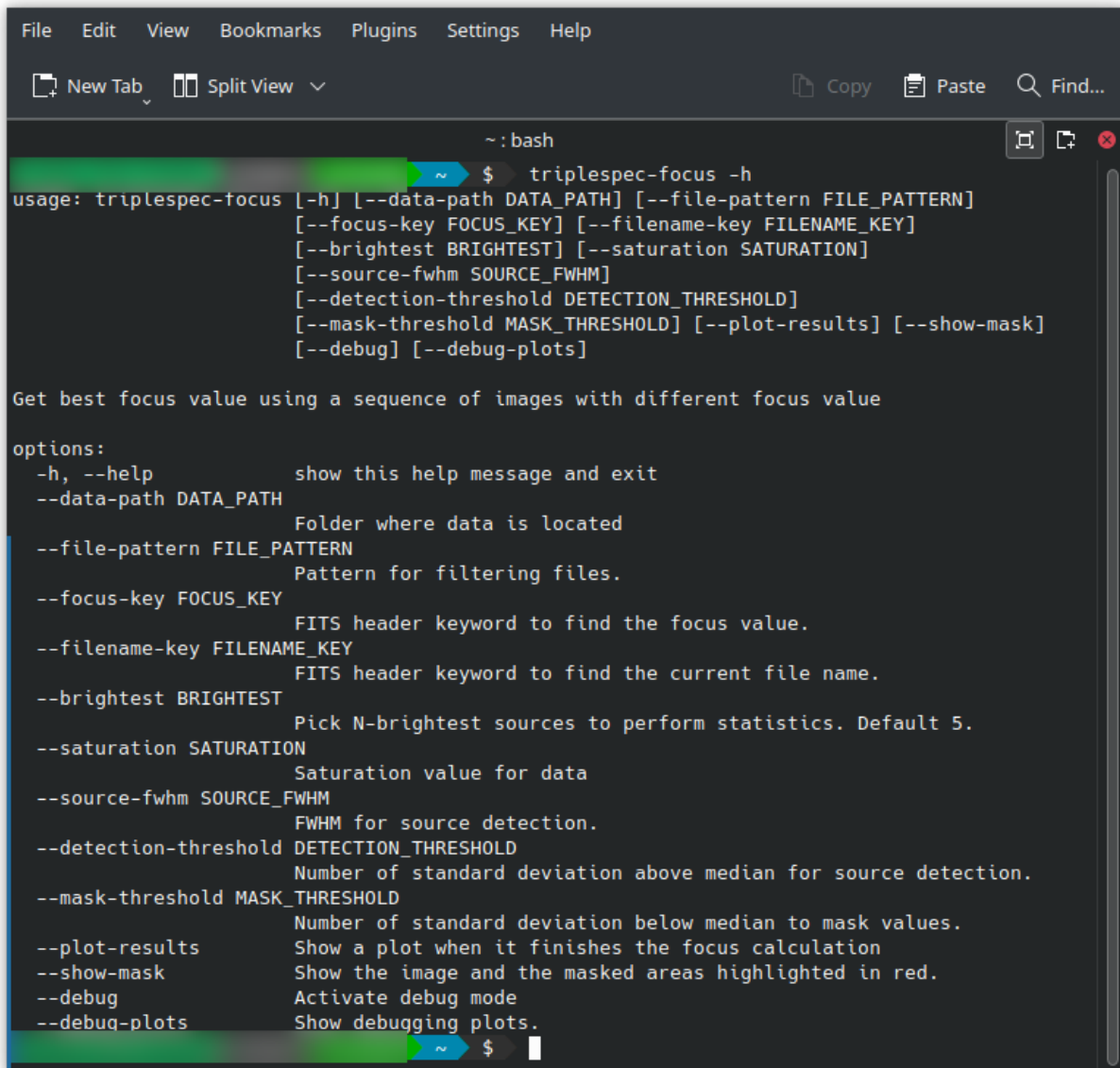
A quick examination routine will select the sharpest image by selecting the most intense image whose peak is also below the saturation level which for TripleSpec Slit Viewer camera is set to $40.000 ADU$

On this selected image the `DAOStarFinder` routing will be used to detect all the sources, the the most intense sources are selected to obtain the $FWHM$ using `ApertureStats` which is then fitted with a `Chebyshev1D` of order 6.

2.3 Use from terminal

The installation will create a terminal entrypoint `triplespec-focus`

Getting help is as easy as running with the `-h` argument:

```
File   Edit   View   Bookmarks   Plugins   Settings   Help

  New Tab    Split View  ∨                              Copy    Paste    Find...

                              ~ : bash

                    ~    $    triplespec-focus -h
usage: triplespec-focus [-h] [--data-path DATA_PATH] [--file-pattern FILE_PATTERN]
                        [--focus-key FOCUS_KEY] [--filename-key FILENAME_KEY]
                        [--brightest BRIGHTEST] [--saturation SATURATION]
                        [--source-fwhm SOURCE_FWHM]
                        [--detection-threshold DETECTION_THRESHOLD]
                        [--mask-threshold MASK_THRESHOLD] [--plot-results] [--show-mask]
                        [--debug] [--debug-plots]

Get best focus value using a sequence of images with different focus value

options:
  -h, --help            show this help message and exit
  --data-path DATA_PATH
                        Folder where data is located
  --file-pattern FILE_PATTERN
                        Pattern for filtering files.
  --focus-key FOCUS_KEY
                        FITS header keyword to find the focus value.
  --filename-key FILENAME_KEY
                        FITS header keyword to find the current file name.
  --brightest BRIGHTEST
                        Pick N-brightest sources to perform statistics. Default 5.
  --saturation SATURATION
                        Saturation value for data
  --source-fwhm SOURCE_FWHM
                        FWHM for source detection.
  --detection-threshold DETECTION_THRESHOLD
                        Number of standard deviation above median for source detection.
  --mask-threshold MASK_THRESHOLD
                        Number of standard deviation below median to mask values.
  --plot-results        Show a plot when it finishes the focus calculation
  --show-mask           Show the image and the masked areas highlighted in red.
  --debug               Activate debug mode
  --debug-plots         Show debugging plots.

                    ~    $   ▮
```

If you run `triplespec-focus` from the same data folder as the location of the data, you don't need to specify any argument.

The arguments, default values and options are listed in the following table:

Table 1: Default values for arguments

Argument	Default Value	Options
`--data-path <input>`	Current Working Directory	Any valid path
`--file-pattern <input>`	`*.fits`	Any pattern
`--focus-key <input>`	`TELFOCUS`	Any valid FITS keyword
`--filename-key <input>`	`FILENAME`	Any valid FITS keyword
`--brightest <input>`	5	Any positive integer
`--saturation <input>`	40000	Any positive float
`--source-fwhm <input>`	7	Any positive float
`--detection-threshold <input>`	5	Any positive float
`--plot-results`	False	True
`--show-mask`	False	True
`--debug`	False	True
`--debug-plots`	False	True

After running the script with all desired parameters the result will be printed in the terminal.

2.4 Using as a Library

This library can be integrated into other software, just import `TripleSpecFocus` create an instance and call it.

```python
from triplespec_focus import TripleSpecFocus

focus = TripleSpecFocus(debug=False,
                        date_key='DATE',
                        date_time_key='DATE-OBS',
                        focus_key='TELFOCUS',
                        filename_key='FILENAME',
                        file_pattern='*.fits',
                        n_brightest=5,
                        saturation=40000,
                        plot_results=False,
                        debug_plots=False)
```

Since in this case all the parameters are the default ones it is equivalent to do:

```python
from triplespec_focus import TripleSpecFocus

focus = TripleSpecFocus()
```

Once the instance is created you can just call it or you can modify certain parameters.

```python
results = focus()
```

or

```python
from pathlib import Path

data_path = Path('./')

results = focus(data_path=data_path,
                source_fwhm=7.0,
                det_threshold=5.0,
                mask_threshold=1,
                n_brightest=5,
                saturation_level=40000,
                show_mask=False,
                plot_results=False,
                debug_plots=False,
                print_all_data=False)
```

Usually you would want to have more control so you can modify any parameters, but the most interesting part is that you can pass a list of files.

Using the same `data_path` variable created in the previous example

```python
file_list = sorted(data_path.glob(pattern='*.fits'))
```

or create the list manually or programatically

```python
file_list = [
'/path/to/file_001.fits',
'/path/to/file_003.fits',
'/path/to/file_007.fits',
'/path/to/file_008.fits',
'/path/to/file_010.fits',
'/path/to/file_012.fits',
]

results = focus(file_list=file_list,
                source_fwhm=7.0,
                det_threshold=5.0,
                mask_threshold=1,
                n_brightest=5,
                saturation_level=40000,
                show_mask=False,
                plot_results=False,
                debug_plots=False,
                print_all_data=False)
```

2.5 Interpreting the Results

Using the option `--plot-results` or set `plot_results=True` you will get the following information at the end.
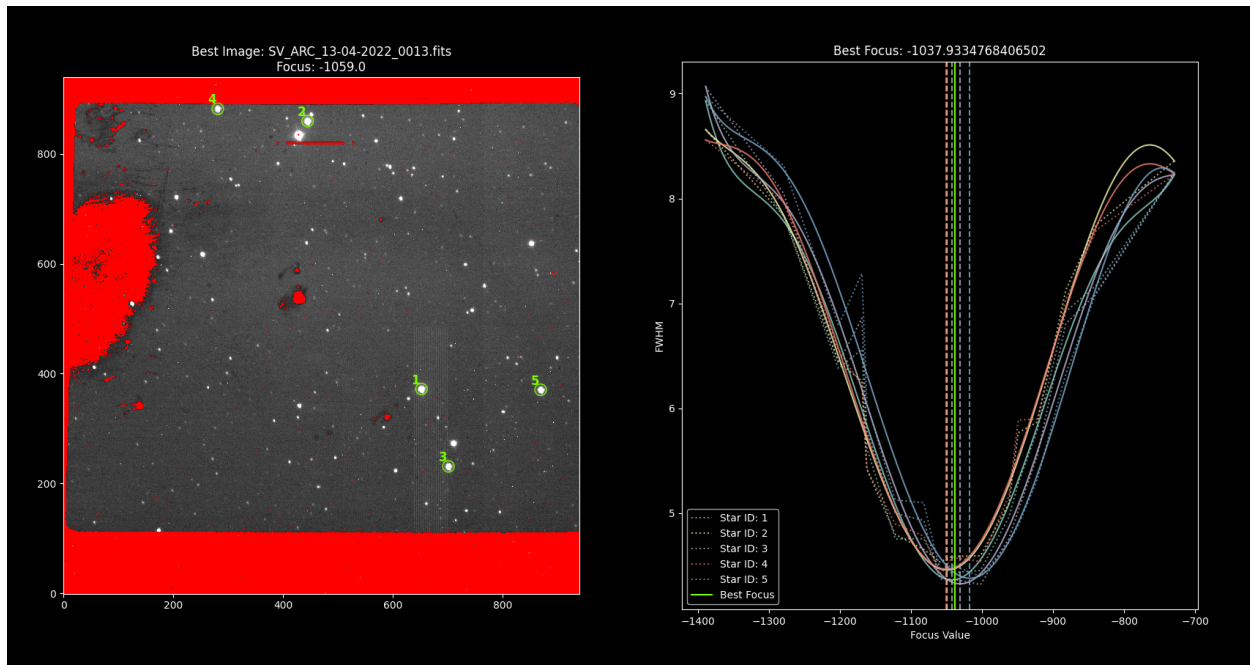
The sources used are plotted to the left and to the right you get the data and fitted models plot. The dashed vertical lines represent the best focus for each individual star. The solid vertical line represent the best focus value.

The result is a dictionary.

```
{
    "date": "2022-04-14",
    "time": "2022-04-14T02:00:16.472",
    "mean_focus": -1037.9334768407,
    "median_focus": -1042.3437309744,
    "focus_std": 12.4463329703,
    "fwhm": 4.400507903,
    "best_image_name": "SV_ARC_13-04-2022_0013.fits",
    "best_image_focus": -1059.0,
    "best_image_fwhm": 4.451495851,
    "focus_data": [
        -1389.0,
        -1279.0,
        -1202.0,
        -1169.0,
        -1162.0,
        -1122.0,
        -1081.99,
        -1059.0,
        -1042.0,
        -1002.0,
        -962.0,
        -949.0,
```

(continues on next page)

```
            -922.0,
            -882.0,
            -839.0,
            -729.0
    ],
    "fwhm_data": [
            8.8346668203,
            7.9968682153,
            6.520501982,
            6.6366621644,
            5.6593294617,
            4.9156293967,
            4.8462257042,
            4.451495851,
            4.4960399387,
            4.4620697186,
            4.9432397845,
            5.3684107482,
            5.6873559877,
            6.8445617244,
            7.3650931863,
            8.2617132835
    ]
}
```

2.6 Change History

2.6.1 1.0.0 07-09-2022

- First functional version
- Uses ApertureStats to obtain sources' statistics
- Fully documented
- Fully tested

2.6.2 0.0.1.dev0 06-05-2022

- Initial release of a non-functional tool but will all the packaging setup.

2.7 License

BSD 3-Clause License

Copyright (c) 2022, SOAR Telescope All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

INDICES AND TABLES

- genindex
- modindex
- search