# hfdfocus

*Release 0.1*

**Michael Fulbright**

**Nov 02, 2020**

# CONTENTS:

CHAPTER

ONE

# USING AUTOFOCUS_HFD_SCRIPT.PY

## 1.1 Introduction

The script autofocus_hfd_script.py handles focusing on a bright star near the center of an image.

## 1.2 Invocation

The invocation of autofocus_auto_star.py is:

```
usage: autofocus_hfd_script.py [-h] [--focus_min FOCUS_MIN]
                               [--focus_max FOCUS_MAX] [--focus_dir FOCUS_DIR]
                               [--focus_start FOCUS_START] [--debugplots]
                               [--debugplotsdelay DEBUGPLOTSDELAY] [--simul]
                               [--stayopen] [--profile PROFILE]
                               [--focuser FOCUSER] [--camera CAMERA]
                               [--exposure_start EXPOSURE_START]
                               [--exposure_min EXPOSURE_MIN]
                               [--exposure_max EXPOSURE_MAX]
                               [--starflux_min STARFLUX_MIN]
                               [--saturation SATURATION]
                               [--framesize FRAMESIZE] [--winsize WINSIZE]
                               [--focusdelay FOCUSDELAY]
                               [--numaverage NUMAVERAGE]

optional arguments:
  -h, --help            show this help message and exit
  --focus_min FOCUS_MIN
                        Lowest focus travel allowed
  --focus_max FOCUS_MAX
                        Highest focus travel allowed
  --focus_dir FOCUS_DIR
                        IN or OUT
  --focus_start FOCUS_START
                        Starting focus pos
  --debugplots          show debug plots
  --debugplotsdelay DEBUGPLOTSDELAY
                        Delay (seconds) showing each plot
  --simul               Simulate star
  --stayopen            stay open when done
  --profile PROFILE     Name of equipment profile
  --focuser FOCUSER     Focuser Driver
  --camera CAMERA       Camera Driver
```

(continues on next page)

(continued from previous page)

```
  --exposure_start EXPOSURE_START
                        Starting exposure value
  --exposure_min EXPOSURE_MIN
                        Minimum exposure value
  --exposure_max EXPOSURE_MAX
                        Maximum exposure value
  --starflux_min STARFLUX_MIN
                        Maximum flux in star
  --saturation SATURATION
                        Saturation level for sensor
  --framesize FRAMESIZE
                        Size of capture frame, 0=full
  --winsize WINSIZE     Size of window used to analyze star
  --focusdelay FOCUSDELAY
                        Delay (seconds) after focus moves
  --numaverage NUMAVERAGE
                        Number of images to average
```

## 1.3 Explanation of arguments

The "–focus_min", "–focus_max", and "–focus_dir" arguments define the allowed region and direction of the focus run.

The hardware drivers can be specified individually with the "–focuser" and "–camera" arguments, or pulled from an "astroprofile" using the "–profile" argument.

The "–simul" argument will run the program using a simulated star instead of connecting to a real camera and focuser. Useful for testing.

CHAPTER

TWO

# USING AUTOFOCUS_AUTO_STAR.PY

## 2.1 Introduction

The script autofocus_auto_star.py handles finding a focus star, slewing to the star, autofocusing and then returning to the original position. It does this by calling other python utilities which handle most of the actual work.

## 2.2 Invocation

The invocation of autofocus_auto_star.py is:

```
usage: autofocus_auto_star.py [-h] [--profile PROFILE] [--lst LST]
                              [--onlyside ONLYSIDE] [--lon LON]
                              [--meridianthres MERIDIANTHRES]
                              [--maxtries MAXTRIES]
                              dist mag
positional arguments:
  dist                  Max distance in degrees
  mag                   Desired mag focus star

optional arguments:
  -h, --help            show this help message and exit
  --profile PROFILE     Name of astro profile
  --lst LST             Local sidereal time
  --onlyside ONLYSIDE   EAST or WEST side only
  --lon LON             Location longitude
  --meridianthres MERIDIANTHRES
                        How close to meridian is allowed (hh:mm:ss)
  --maxtries MAXTRIES   Number of stars to try before giving up
```

## 2.3 Explanation of specifying side of pier

The "–lon" argument allows the specification of the observing latitude. Then script can then compute the local sidereal time. Optionally the local sidereal time can be given with the "–lst" argument.

Once the local sidereal time has been determined then the "–onlyside" parameter can be used to retrict the star to one side of the meridian or the other. It can take a value of "EAST" or "WEST" (capitalized!).

The "–meridianthres" argument can be used to create a "keep out" area near the meridian that excludes choosing a focus star in that area.

## 2.4 Using an astro profile

There are no specific settings covered by an astro profile for the autofocus_auto_star.py script, but several scripts it relies on do.

CHAPTER

THREE

# USING FIND_NEARBY_STARS.PY

## 3.1 Introduction

The "script find_nearby_stars.py" handles finding a star within a specified distance from a RA/DEC position with constraints on brightness. It is normally used by the "autofocus_auto_star.py" script but can also be invoked indepently.

## 3.2 Invocation

The invocation of autofocus_auto_star.py is:

```
usage: find_nearby_stars.py [-h] [--minmag MINMAG] [--maxmag MAXMAG]
                            [--verbose] [--outfile OUTFILE] [--force]
                            [--lst LST] [--onlyside ONLYSIDE]
                            [--meridianthres MERIDIANTHRES] [--lon LON]
                            cat ra2000 dec2000 dist

positional arguments:
  cat                   Catalog to search
  ra2000                RA J2000
  dec2000               DEC J2000
  dist                  Max distance in degrees

optional arguments:
  -h, --help            show this help message and exit
  --minmag MINMAG
  --maxmag MAXMAG
  --verbose
  --outfile OUTFILE     Output file with candidates
  --force               Overwrite output file
  --lst LST             Local sidereal time
  --onlyside ONLYSIDE   EAST or WEST side only
  --meridianthres MERIDIANTHRES
                        How close to meridian is allowed (hh:mm:ss)
  --lon LON             Location longitude
```

## 3.3 Program Output

The program outputs the list of candidate stars to the console and if the argument "–outfile" is given it will also write CSV output to this file. The file includes a header that explains the columns.

## 3.4 Explanation of specifying side of pier

The "cat" argument should reference a binary SAO Catalog created with the utilities in the "find_star" directory. One such file is in the "data" directory and is called "SAO_Catalog_m5_p11_filtered.bin" and has stars down to magnitude 11. It has been filtered of stars that are close to one another to reduce the chance of having a another star interfere with the autofocus routine.

The "–lon" argument allows the specification of the observing latitude. Then script can then compute the local sidereal time. Optionally the local sidereal time can be given with the "–lst" argument.

Once the local sidereal time has been determined then the "–onlyside" parameter can be used to retrict the star to one side of the meridian or the other. It can take a value of "EAST" or "WEST" (capitalized!).

The "–meridianthres" argument can be used to create a "keep out" area near the meridian that excludes choosing a focus star in that area.

CHAPTER

FOUR

# USING V CURVES

## 4.1 Introduction

A "V Curve" is a graph of the size of a star (measured as a half flux radius, or HFD) versus focus position. The name refers to the shape of the graph as the star will shrink as best focus is approached and then grow larger after it is passed.

For a given imaging telescope a V Curve will need to be captured in order to train the autofocus routine. Actually several V Curves are normally captured and then averaged together.

## 4.2 Capturing V Curves With capture_vcurve_script.py

The program "capture_vcurve_script.py" is used for the automated capture of V Curves. The program will run a specified number of V Curve captures.

```
usage: capture_vcurve_script.py [-h] [--debugplots] [--savefits] [--simul]
                                [--profile PROFILE] [--backend BACKEND]
                                [--focuser FOCUSER] [--camera CAMERA]
                                [--exposure_start EXPOSURE_START]
                                [--exposure_min EXPOSURE_MIN]
                                [--exposure_max EXPOSURE_MAX]
                                [--saturation SATURATION]
                                [--starflux_min STARFLUX_MIN]
                                [--framesize FRAMESIZE]
                                [--runoffset RUNOFFSET]
                                [--hfdcutoff HFDCUTOFF] [--bgthres BGTHRES]
                                [--movedelay MOVEDELAY] [--backlash BACKLASH]
                                focus_center focus_range focus_nstep focus_dir
                                nruns

positional arguments:
  focus_center          Center position of focus run
  focus_range           Range of focus run
  focus_nstep           V Curve number of steps
  focus_dir             IN or OUT
  nruns                 Number of vcurve runs

optional arguments:
  -h, --help            show this help message and exit
  --debugplots          show debug plots
  --savefits            Save all images taken
  --simul               Simulate star
  --profile PROFILE     Name of astro profile
```

(continues on next page)

(continued from previous page)

```
  --backend BACKEND     Backend
  --focuser FOCUSER     Focuser Driver
  --camera CAMERA       Camera Driver
  --exposure_start EXPOSURE_START
                        Starting exposure value
  --exposure_min EXPOSURE_MIN
                        Minimum exposure value
  --exposure_max EXPOSURE_MAX
                        Maximum exposure value
  --saturation SATURATION
                        Saturation level for sensor
  --starflux_min STARFLUX_MIN
                        Maximum flux in star
  --framesize FRAMESIZE
                        Size of capture frame, 0=full
  --runoffset RUNOFFSET
                        Shift center of run by this amount
  --hfdcutoff HFDCUTOFF
                        Ignore points with HFD less than this value
  --bgthres BGTHRES     Threshold multiplier for star detection
  --movedelay MOVEDELAY
                        Delay in seconds between moves
  --backlash BACKLASH   Number of steps of backlash for overshoot method
```

When run it will create a directory with a named based on the current data and time and fill it with the images captured at each focus position. It will also create a file called "vcurve_fits.json" which contains the fit parameters for each V Curve captured. This file is the important output we need.

## 4.3 Averaging V Curves With average_curve_runs.py

Once we have run a series of V Curves we can average them to create a better estimate of the autofocus parameters needed. The script "average_vcurve_runs.py" is used for this purpose.

```
usage: average_vcurve_runs.py [-h] [--debugplots] vcurve_file

positional arguments:
  vcurve_file   V Curve filename

optional arguments:
  -h, --help    show this help message and exit
  --debugplots  show debug plots
```

Basically pass it the path and name of the "vcurve_fits.json" file which is to be averaged. It will right out a JSON string containing the best fit parameters.

CHAPTER

FIVE

# HFDFOCUS

## 5.1 hfdfocus package

### 5.1.1 Submodules

### 5.1.2 hfdfocus.MultipleStarFitHFD module

### 5.1.3 hfdfocus.SAOCatalog module

### 5.1.4 hfdfocus.StarFitHFD module

### 5.1.5 hfdfocus.StarFitHFR_RadialProfile module

### 5.1.6 hfdfocus.c8_simul_star module

### 5.1.7 hfdfocus.utilities module

### 5.1.8 Module contents

CHAPTER

SIX

# INDICES AND TABLES

- genindex
- modindex
- search

# PYTHON MODULE INDEX

## h

# INDEX