Task 1

Astronomical observations of δ Scuti stars with the 70 cm telescope of the LSW

Jochen Heidt & Walter Seifert, March 5, 2024

Introduction

The typical duty cycle of an observational astronomer has 5 different phases. It starts I) with a scientific idea or question to be addressed and the successful application for telescope time, followed by II) the preparation and III) the execution of the observations. Once this is done, the scientific data set has IV) to be reduced and analyzed and V) the results be presented at astronomical conferences and finally published in a refereed astronomical journal. Normally an astronomer faces a number of those cycles in parallel, mostly somewhat out of phase with respect to each other.

While the topics I) and V) are ingredients to be learned and practiced during the course of a Phd, the remaining fundamental ones are well suited to be introduced during the astrolab. Thus you will get familiar in this task with:

1.1 Preparing astronomical observations

1.2 Carrying out astronomical observations

1.3 Data reduction and analysis

1.4 And as a concrete application: Investigating a deep field

To do so, you will learn the use of one popular astronomical software package and will also get a somewhat deeper understanding of the use and principles of modern charged coupled devices (CCDs). In practice, you will take time series data of a periodically variable star via relative photometry. This technique is relatively insensitive to ambient observing conditions. In addition, you will learn how this data set can be used for a completely different application.

1.1 Preparing astronomical observations

In order to use the telescope and its instrument (a CCD camera) you need to get familiar with its use. This will be done at the beginning of your observing session, where you will get an introduction by the assistant at the telescope itself.

For your project, you need to plan your observing session accordingly. Complementary to your astronomical data you also need to take several calibration images like biases, darks & flatfields because the astronomical data set is of little use otherwise. Why this is important will become clear to you later on.

To get ready "on sky" you need to find out a couple of things in advance as listed below. Assume the day when your observations are scheduled and the Landessternwarte Heidelberg as location. The coordinates for the Landessternwarte are: $8^o43'28''$ East, $49^o23'53''$ North. Use astronomical resources on the WEB, there is **no** need to do any calculus by hand.

• When does the sun set, when does it rise (in local time)?

• What is the meaning of civil twilight, nautical twilight and astronomical twilight?

• Flatfields are normally taken around the beginning of the nautical twilight in the evening and at the end in the morning. Astronomical observations can normally start after the end of the astronomical twilight in the evening and end after the start in the morning. At which times start the civil, nautical and astronomical twilights in the evening and morning?

• What is the meaning of sidereal time? What is the sidereal time at the end of the astronomical twilight in the evening? What does it mean in practice for an object with a right ascension equal to the sidereal time (in terms of observability)?

• The main target for the observing session will be a star of δ Scuti type. What is so special about these stars? Where are they located in the Hertzsprung-Russell diagram?

• Tab. 1 shows potential δ Scuti stars for your observing sessions. Not all of them are observable at the same time. Which one would you observe during your observing session and why? Hint: Consider the airmass of the target during your observing session. The airmass can be written as AM $\propto 1/cos(zenith\ distance)$. Why is it unfavorable to observe at higher airmasses (≥ 2)?

Target	RA	Dec	V	Period	Amplitude
	(2000)	(2000)	[mag]	[min]	[mag]
V460 And	$02^h34^m14.25^s$	42^o 14' 27.6"	13.5	108	0.60
GD 428	$03^h47^m19.88^s$	63^o 22' 42.1"	13.1	56	0.32
GSC 00191-01282	$07^h37^m58.43^s$	05^o 52' 27.9"	12.9	68	0.47
CY Aqr	$22^h37^m47.82^s$	01^o 32' 03.8"	11.0	88	0.71

Table 1: Sample of δ Scuti stars, which are the main targets for the observations.

• You need also to make sure that the target of your choice is not too close to the moon and that you should rather choose another one. What is the distance of the moon to the target of your choice in degrees during your observing session? Use the coordinates of the moon for 20h local time to check this.

• When you point to your target with the telescope you will first do an acquisition image to verify that you are on sky where you would like to be. For that you need a Finder Chart as is shown in Fig. 7 for CY Aquarius. Prepare a Finder Chart for the δ Scuti star of your choice.

Good resources for creating Finder Charts are the Sloan Digital Sky Survey (sdss.org) or the SIMBAD astronomical database (simbad.u-strasbg.fr/simbad/).
• Find out what relative (or differential) photometry means. Why is this technique so powerful in particular when observing from the Königstuhl?
• You may also observe a Messier or any other celestial object with different filters for fun and create a multi-color image for our own purposes. Select a nice deep-sky object observable either at the beginning of the night or about 4hrs after the end of the astronomical twilight (pls. do not select M27, M31 or M57 they have been observed many times already during the astro-lab). Recently, several trees have been cut here, ie you may observe objects down to -25deg declination at culmination! Take into account that we have not only broad-band filters (UBVRI) but also narrow-band filters (H_α, H_β, [O III]) at our disposal, the latter of which may be used for observations of gaseous nebulae. Recall that we have a large field of view of about 23' × 23'. Prepare also a Finder Chart and verify that the object is observable at air masses ≤ 2 and is at least 30deg off the moon. Choose the filters according to the physical nature (spectrum) of the object.

Once this is done and everything discussed with the assistant you may start with the observations.

1.2 Carrying out astronomical observations

Important: Make sure to take notes properly (logs can be found at the telescope). You need them later on for the data reduction and analysis.

• Opening dome, startup telescope and setup system.
• Check system by taking a bias frame (not saved).
• Take 10 bias frames first (binning = 2 for the entire observations).
• Once it is dark enough, 5 twilight flats in corresponding filters if you plan to observe a deep sky object, signal 10000 - 30000 ADU and 5 twilight flats in the V-filter (for relative photometry).
• Focusing telescope, determination offset telescope
• Start with the observations of the δ Scuti star. In order to have a sufficiently high signal-to-noise, exposure times should be in the range 30-90sec. Make sure that the target is at the center of the CCD and that a similarly bright star is on the frame (like the bright star $\sim$ 6' east of CY Aquarius, see Fig. 7) as it will be used for differential photometry. The peak intensities of the δ Scuti star and the reference star should be below 50000 ADU to be sure to work in the linear regime.
• The observations should go on for at least 2hrs to cover a full period of your main target.
• Do the observations of your Messier object.
• Shutdown telescope and closing dome.
• Once everything is done, start some darks using a script (made together with the assistant) with the different exposure times you have used for flatfields and your science exposures, 5 each. Once the script is started, it is time to rest.

1.3 Data reduction and analysis - to be done during the astrolab

1.3.1 Charge coupled devices

Astrophysical measurements in the optical wavelength regime are nowadays basically be carried out with CCDs (Charged Coupled Devices). Compared to photographic plates they have a higher quantum efficiency, a larger dynamical range, repeatable and nearly linear characteristics and a broad spectral sensibility. The incident stream of photons $S(\omega)$ is detected in each pixel in the form of signal electrons $N_{e^-}(x, y)$ and digitized during the the readout (ADU(x,y)).

Fundamentally, a charge coupled device (CCD) is an integrated circuit etched onto a silicon surface forming light sensitive elements called pixels. Photons incident on this surface generate charge via the photoelectric effect that can be read by electronics and turned into a digital copy of the light patterns falling on the device. CCDs come in a wide variety of sizes and types and are used in many applications from cell phone cameras to high-end scientific applications.

The function of a CCD can be visualized as an array of buckets (pixels) collecting rainwater (photons). Each bucket in the array is exposed for the same amount of time to the rain. The buckets fill up with a varying amount of water, and the CCD is then read one bucket at a time. This process is initiated by pouring water into the adjacent empty column. The buckets in this column transfer their “water” down to a final pixel where the electronics of the camera read-out this pixel (the computer measuring the bucket) and turn it into a number that can be understood and stored by a computer.

Of course, this is an oversimplification - in fact, this “model” described above is actually wrong in some ways, all the pixels in a CCD are actually shifted simultaneously, not one column at a time. Fig. 1 shows how that works in principle.

In practice the charges generated by the photons on the pixel need to be kept for a certain period of time (the integration time) via a potential well. In order to create a potential well, a voltage is applied to each pixel (called “Vorspannung”). This means that each pixel has a priori a number of electrons in its potential well, regardless of the number of photons received from the science target or sky background and is often called “Bias”.

All CCDs benefit from working at lower temperatures. Thermal energy alone is enough to excite extraneous electrons into the image pixels and these cannot be distinguished from the actual image photoelectrons. This process generates an additional component which is called “dark current” and which is dependent on the temperature of the CCD. Modern CCDs are nowadays cooled down to -120^0 C with liquid nitrogen (LN2), which is a compromise between minimum dark current and maximum quantum efficiency. Lower temperatures would result in lower dark current but also lower quantum efficiency. The CCD which is in operation at the 70cm telescope is a Peltier cooled detector at about -30^0 C. This detector has a much higher dark current compared to CCDs used at large telescopes, but is not a concern as the night sky in Heidelberg is very bright. The main advantage of these detectors is that they are much cheaper (still in the $\sim$ 10000,- Euro range) and no liquid LN2 is required to operate them (just turning on the power is enough).

Although not discussed in detail, it is clear from Fig. 1 that each pixel of a CCD is a complex element of it. Naturally, each pixel has not the same “efficiency” to “convert” the incident photons into electrons, also called “quantum efficiency”. This effect is wavelength dependent.

Unavoidably, there are several on-chip sources of noise. The most prominent one is associated with the readout of the detector and the conversion from the analog (electrons) to digital (analog-to-digital Unit = ADU) signal and is called readout noise (RON). This comes from the amplifier(s) involved and adds some noise to the signal. This is a characteristic of the detector and of the read-out-mode used. The RON is normally given in electrons RMS. If the conversion factor from electrons to ADU called Gain is known, the readout noise can be converted accordingly. Knowledge of the amplitude of the readout noise is fundamental for some astronomical observations. The signal from the nightsky and the target causes shot noise due to the random arrival of photons. Since each photon is an independent event, the arrival of any given photon cannot be precisely predicted; instead the probability of its arrival in a given time period is governed by a Poisson distribution. The statistical properties of a Poisson distribution are well understood, while they are not for the distribution underlying the on-chip sources of noise. Thus the observer must make sure that his observations are shot noise or “photon-noise” dominated. This is not a concern for the observations with the 70cm telescope as the observations are photon-noise dominated with broad-band filters in a few hundreds of seconds but are a potential problem when dispersing the light across the detector during high-resolution spectroscopic observations.

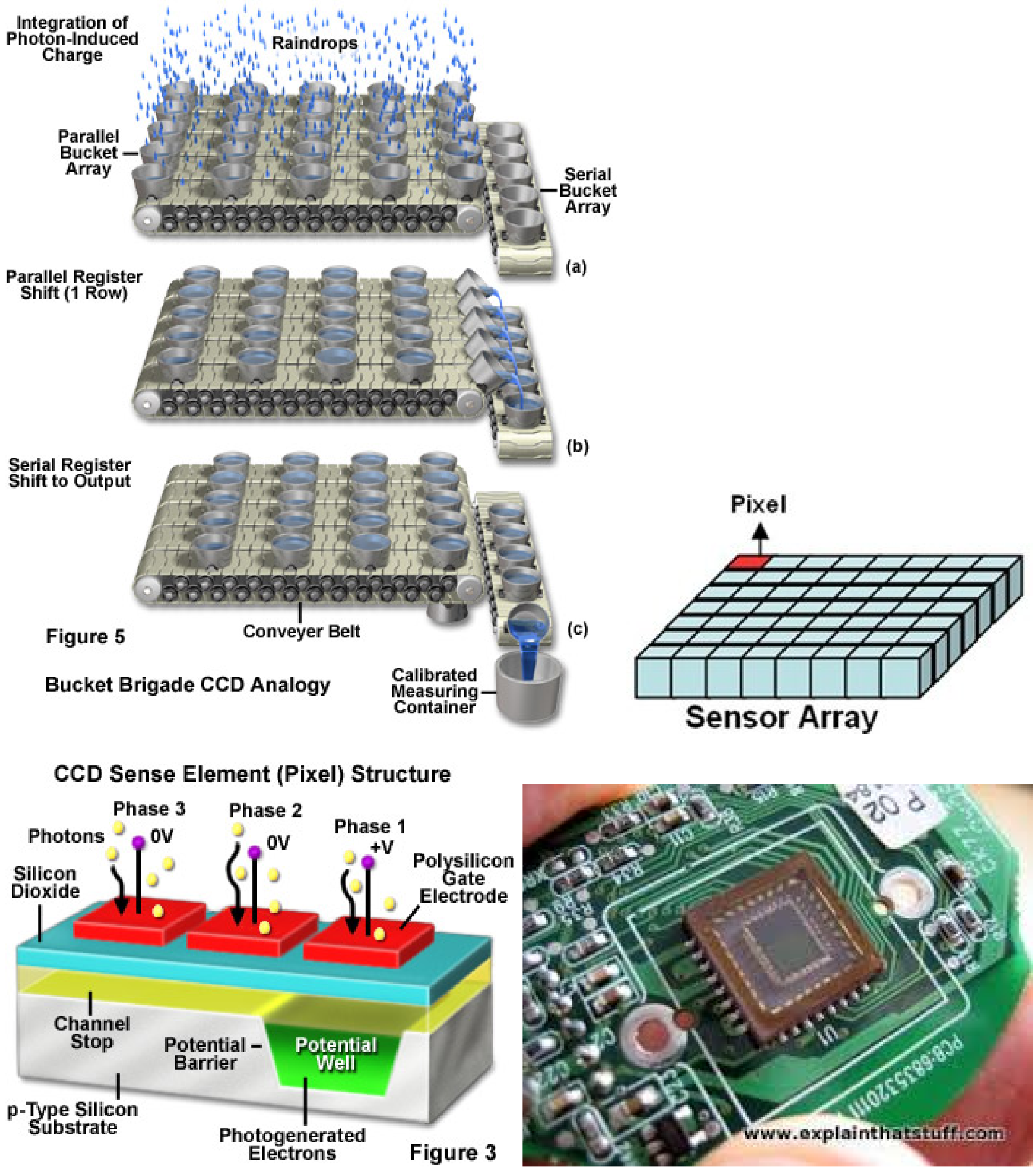


Figure 1: Top left) Working principle of a CCD. Top right) Sketch of an array. Bottom left) Sketch of a pixel. Bottom right) A CCD in reality.

1.3.2 General elements of the reduction of CCD measurements

a) Data reduction

It is obvious from the description of CCDs in the previous chapter that the data collected at the telescope are "contaminated" by various effects. In addition, an imperfect optic and instrument or other effects may led to an inhomogeneous illumination of the detector (eg dust particles on the filter used for the observations). They all need to be corrected for before they can be analyzed and used for scientific purposes. These correction procedure is the first step when working with newly collected data and called "data reduction".

To summarize, the data are impacted and need to corrected for by:

- The bias
- The dark current
- The wavelength-dependent quantum efficiency which varies from pixel-to-pixel
- Potential inhomogeneous illumination of the CCD
- Once this is done one need to make sure that the data are photon-noise dominated

b) Absolute calibration of the measurements

Once the data have been reduced they need to be calibrated on absolute scales. For direct images one has to correct e.g. for possible distortion of the optics to get absolute coordinates (α, δ etc.) from the detector coordinates (x, y). For spectroscopical measurements one has to convert the detector coordinate into a wavelength scale. The last step is the absolute flux calibration which consists of a correction for the Earth's atmosphere and the calibration of the total efficiency of the measuring system (telescope, instrument, detector).

The absolute calibration of the measurements will not be touched here.

1.3.3 Ready to go

The data reduction described above is the main topic of this task. You will learn how one can correct for bias, the dark current, pixel-dependent quantum efficiency, the potential inhomogeneous illumination and make sure that your data are photon-noise dominated. To do so, you need to develop a "magic formula" which takes all this into account when correcting science images and will actively correct a data set taken with the 70cm telescope.

To carry out the data reduction some calibration measurements have been taken before, after or during the night where the scientific data have been collected. These are:

- Bias images, which are exposures with an integration time of 0 sec. Here the CCD is just read out. For your data reduction 10 of them are available and named bias01...bias10.
- Dark current images, which are exposures with a fixed integration time, but the shutter stays closed. No external photons are collected, just the thermal radiation is determined. For your data reduction 2 sets 10 frames each are available. One with an integration time of 5sec which matches the exposure time of the flatfields (see below) and one with an integration time matching the exposure time of the science frames (mostly 40sec). Hence they are named dark5sec01..10 and dark40sec01..10.
- Calibration images taken at twilight called flatfields. These represent images of an homogeneously illuminated sphere without an astronomical signal (here the sky, but can in principle

also be done in the dome with a screen been illuminated by white light) and are used to correct for quantum efficiency variations and other effects. They are named flatV01..10 or flatR01..10.

You will or have received a laptop with a LINUX-distribution containing your data set in a specific folder which named after the date of the observations (eg 20171012). There you will find two subfolders, one named “work” and one named after your target (eg cy_aqui). In the folder work you will find all calibration measurements (bias*, dark*, flat*) and one or two science images (named after target, eg cy_aqui_066 or gd_0428_019). This is the folder where you start, develop your images required for a proper data reduction and reduce an science image to verify that your strategy works. Once this is done, you have to copy your images required for a proper data reduction to the folder named after your target, where all science frames are located. There you will than carry out the data reduction of all images semi-automatically and analyze the data. The deep field image is also created and analyzed in this folder.

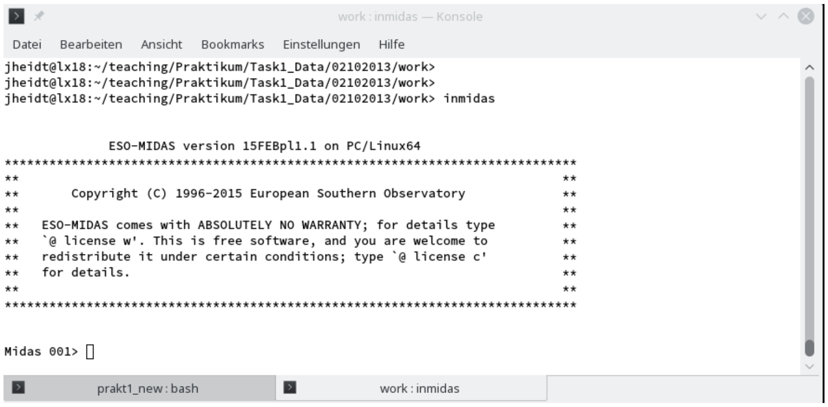
CCD images are normally processed and analyzed with software packages specifically developed for astronomical applications. You will use a software package called *MIDAS*. In the next two chapters, you will find a short introductory guide to *MIDAS* and a short compendium on statistical properties, both of which are relevant for your work.

1.3.4 A short beginners guide to MIDAS

In the parts 1.3 and 1.4 of this task you will work with astronomical data taken with the 70cm telescope. To do so, you need a dedicated software package for that purpose. There are 3 popular packages on the market: IDL (Interactive Data Language), which is an (excellent but expensive) commercial product, IRAF (image reduction and analysis facility) which is provided by NOAO (National Optical Astronomy Observatory of the US) and MIDAS (Munich Imaging and Data Analysis Software), which is commonly used in Europe and which was developed by the European Southern Observatory (ESO). Both, IRAF and MIDAS are freely available (they are eg delivered as part of the scisoft-package) and can be found at http://iraf.noao.edu/ and http://www.eso.org/esomidas, respectively. Here you will use MIDAS.

In the following a short introduction to MIDAS will be given. You will learn how to use the most important 20 or so commands you need for the parts 1.3 and 1.4 of Task 1. Just follow them as they are described below using data on your directory and practice. There are many more commands available, of course, but to get familiar with the entire package requires quite some time...

Starting MIDAS is quite simple. Navigate to the directory with your data (you will start in a directory named "work") and type "**inmidas** in the window and hit return (a hit of the return key is required for all commands and will not be repeated anymore). The window should than look like in Fig. 2. To leave MIDAS type **bye**.

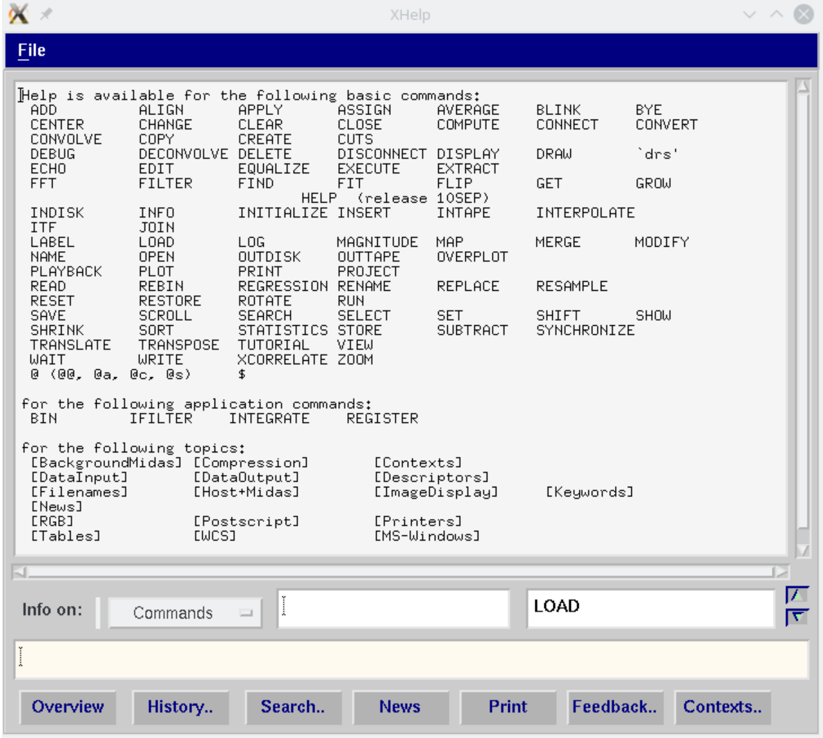


```
jheidt@lx18:~/teaching/Praktikum/Task1_Data/02102013/work>
jheidt@lx18:~/teaching/Praktikum/Task1_Data/02102013/work>
jheidt@lx18:~/teaching/Praktikum/Task1_Data/02102013/work> inmidas


              ESO-MIDAS version 15FEBpl1.1 on PC/Linux64
*****************************************************************************
**                                                                         **
**       Copyright (C) 1996-2015 European Southern Observatory             **
**                                                                         **
**   ESO-MIDAS comes with ABSOLUTELY NO WARRANTY; for details type         **
**   `@ license w'. This is free software, and you are welcome to          **
**   redistribute it under certain conditions; type `@ license c'          **
**   for details.                                                          **
**                                                                         **
*****************************************************************************


Midas 001> 
```

Figure 2: Window appearance once MIDAS has started. Note the "Midas 001>" prompt.

.

To get an overview to the plethora of MIDAS commands, you best use a GUI specifically developed for that. **crea/gui help** will open a separate GUI (Fig. 3). Move the window to any place on your screen and leave the GUI open all time. You can see that MIDAS has numerous so-called "basic commands". Just click on one of them, eg **load**. You will than learn that for each "basic command" MIDAS has some "individual commands" like **load/image**, **load/itt**, **load/lut** and **load/table**. Exactly these "individual commands" are the ones you will execute. On the screen (Fig. 4) you will now see the syntax for each of the individual commands and a short description of what it does. **load/image** eg allows you to "load an image into a display device" (you will see in a second what that means). As for the syntax, you have the command itself and some parameters you can set. Some *MUST* be set and are displayed without brackets and some *can* be set and are displayed in brackets. As for the latter, those parameters are predefined within MIDAS (sometimes in a useful, sometimes in a less optimal way). If you now click on one of those commands, you will than get a full description of what it does, how the parameters can be set etc. and some examples at the end. Check it for **load/image** and you will see that this is quite lengthy. The best way to learn about a command is to go through the examples at the end of the description (and to practice as you should do).



```
Help is available for the following basic commands:
 ADD         ALIGN       APPLY       ASSIGN      AVERAGE     BLINK       BYE
 CENTER      CHANGE      CLEAR       CLOSE       COMPUTE     CONNECT     CONVERT
 CONVOLVE    COPY        CREATE      CUTS
 DEBUG       DECONVOLVE  DELETE      DISCONNECT  DISPLAY     DRAW        `drs'
 ECHO        EDIT        EQUALIZE    EXECUTE     EXTRACT
 FFT         FILTER      FIND        FIT         FLIP        GET         GROW
                               HELP   (release 10SEP)
 INDISK      INFO        INITIALIZE  INSERT      INTAPE      INTERPOLATE
 ITF         JOIN
 LABEL       LOAD        LOG         MAGNITUDE   MAP         MERGE       MODIFY
 NAME        OPEN        OUTDISK     OUTTAPE     OVERPLOT
 PLAYBACK    PLOT        PRINT       PROJECT
 READ        REBIN       REGRESSION  RENAME      REPLACE     RESAMPLE
 RESET       RESTORE     ROTATE      RUN
 SAVE        SCROLL      SEARCH      SELECT      SET         SHIFT       SHOW
 SHRINK      SORT        STATISTICS  STORE       SUBTRACT    SYNCHRONIZE
 TRANSLATE   TRANSPOSE   TUTORIAL    VIEW
 WAIT        WRITE       XCORRELATE  ZOOM
 @ (@@, @a, @c, @s)      $

for the following application commands:
 BIN        IFILTER    INTEGRATE    REGISTER

for the following topics:
 [BackgroundMidas] [Compression]       [Contexts]
 [DataInput]       [DataOutput]        [Descriptors]
 [Filenames]       [Host+Midas]        [ImageDisplay]      [Keywords]
 [News]
 [RGB]             [Postscript]        [Printers]
 [Tables]          [WCS]               [MS-Windows]
```

Info on: Commands | LOAD

Figure 3: The MIDAS help GUI.

.

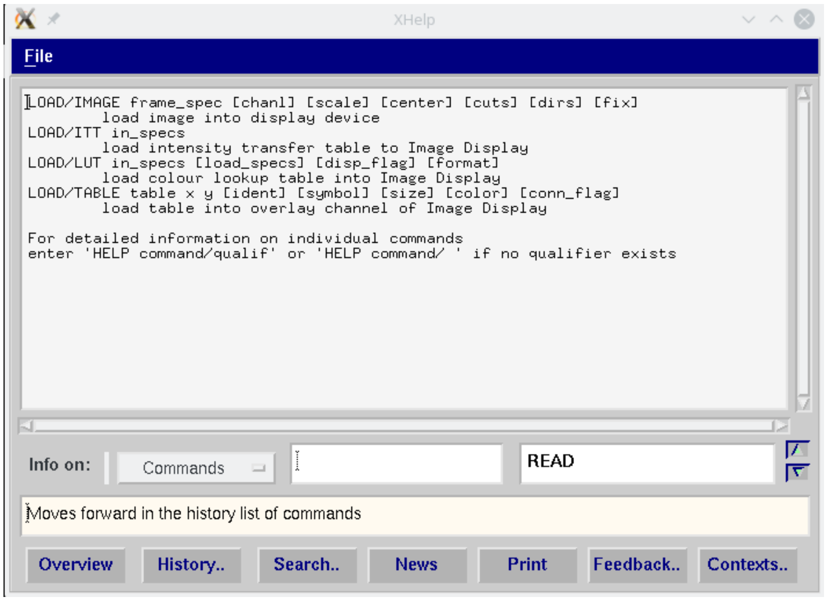
Of course, you need to display your images 2-dimensionally and also need to create plots. **crea/disp** and **crea/grap** opens a display and a graphics window for you, respectively. Rearrange these two windows on your screen as you like, but do ***never*** modify their size using the mouse!

Now its time to do something with your data and we will show you in the following how the commands you need to use work. Some important tips:

• Do never overwrite original files unless you have a good reason for it. In case you do eg arithmetics with one of them, subtracting two images from each other, for example, give the resulting image a **new** name!

• Below you will find examples of how to execute commands. **inputimage** refers always to a placeholder for a real image (which could eg be dark40sec05). **newimage** refers always to a placeholder for a newly created image, which need to be named (eg finalbias).

• There are no restriction on how you can name your newly created images based on arithmetics. You can even name it *test+*. Thus, in case you perform a mathematical operation, you *must always* use an empty string (space), before/after the operator!!!



```
LOAD/IMAGE frame_spec [chanl] [scale] [center] [cuts] [dirs] [fix]
        load image into display device
LOAD/ITT in_specs
        load intensity transfer table to Image Display
LOAD/LUT in_specs [load_specs] [disp_flag] [format]
        load colour lookup table into Image Display
LOAD/TABLE table x y [ident] [symbol] [size] [color] [conn_flag]
        load table into overlay channel of Image Display

For detailed information on individual commands
enter 'HELP command/qualif' or 'HELP command/ ' if no qualifier exists
```

Moves forward in the history list of commands

Figure 4: “Individual commands” related to the “basic command” **load**. Note the (non)bracketed parameters.

.

. Any Linux-command can be executed within MIDAS by a $-sign preceding the command. Type **$ls** to get a listing of your directory.

LOAD/IMAGE allows you to display an image in the MIDAS-display. The parameter, which you have to set is *frame_spec*, which is the name of the frame you would like to display. In addition you should specify the *scale* and *cuts* parameter. The former allows you to display the full image (2048 × 2048 pixel) in the display which has 512 × 512 pixel only, while the latter allows you to select an appropriate dynamic range. For *scale* you should use *-4* (only every 4th pixel is displayed), while you need to find out by yourself which dynamic range to use. If you don't choose a dynamic range, MIDAS will check for the min,max values on the image and set the dynamic range accordingly (the pixel with the lowest value will be black and the one with the highest value will be white while the rest is displayed grayish depending on their intensities). This is bad in most cases as each CCD has bad pixels with unusual values leading to a poorly selected dynamical range. You are normally interested to see either your object on the image nicely or - in the majority of cases - the *background* properly displayed as you than can inspect of inhomogeneities or other effects.

Now just load an image in your display, eg **load/image dark40sec05 scale=-4** and check what you see. Its probably black. Look at the information provided at the bottom of the display. *Start* and *End* give you the x,y-pixel coordinates displayed on the lower left and upper right corner of the display, *Cuts* and *Min,Max* the dynamic range used and the min,max values of the entire image. If *Cuts* and *Min,Max* agree its probably a wise idea to change the dynamic range.

GET/CURSOR allows you to find that out. Move the large cross with your mouse across the display and click with the left hand side on the mouse. On the MIDAS-terminal the pixel-coordinates and intensities of your clicks will be displayed, the positions of your clicks will be shown on the display as small crosses. Do this a number of times to have a feeling about "typical" intensities. Once you are done, click with the central part of the mouse to get a skull. Now you can continue to type commands in the MIDAS-terminal.

The little crosses will stay forever, if you don't remove them. **CL/CH O** takes care of that.

Try now **load/image** again but this time set the *cuts* parameter in addition based on the numbers you just got, eg **load/image dark40sec05 scale=-4 cuts=1100,1250**. Does the background change? If it is grayish and if it looks "noisy", your dynamic range is well selected.

The standard coloring of the frame is black/white. With **LOAD/LUT** you can modify your lookup-table. There are lots of different ones available. You may try **TUTORIAL/LUT** for a freak-show by MIDAS. My favorite is **LOAD/LUT heat** (you need to reolad the image than to make it working). If you would like to return to grayish, use **LOAD/LUT ramp**.

Sometimes it might be useful to plot a *row* or a *column* of an image. This can be done via the **PLOT/ROW** and **PLOT/COL** commands.
PLOT/ROW dark40sec05 1000, for example, will plot the row number 1000 of the image "dark40sec05". Try it and inspect the graphics window. On the x-axis the pixel-number in row 1000 vs. the intensity on the y-axis is given. For the y-axis MIDAS uses normally the *cuts* values from the image. Of course, you can change that. You either reload the image using different *cuts* values or set the graphics parameters directly before redoing the **PLOT/ROW** command.

SET/GRAPHICS allows you to change the appearance of the graphics window. Again, there are many many options, but what is of interest only is the scaling of the *xaxis* and the *yaxis*. **SET/GRAPHICS xa=0,1000 ya=50,5000** eg will set the scaling of the x/y-axis accordingly (you can set both axis at the same time or just one). With **SET/GRAPHICS xa=auto ya=auto** you ask MIDAS to auto-set the scaling of the x/yaxis on the graphics display.

Sometimes it might be of interest to learn a bit more about the contents of an image. Once it is read out at the telescope and saved to disk a ASCII-file is appended, which is called "descriptor" and which contains a lot of information like exposure time, position on sky, weather conditions etc (on large telescopes several hundred entries can be found). Do a **READ/DESCRIPTOR dark40sec05** for example. It shows you than some standard descriptors of the image.
A **READ/DESCRIPTOR dark40sec05 *** shows all of them.

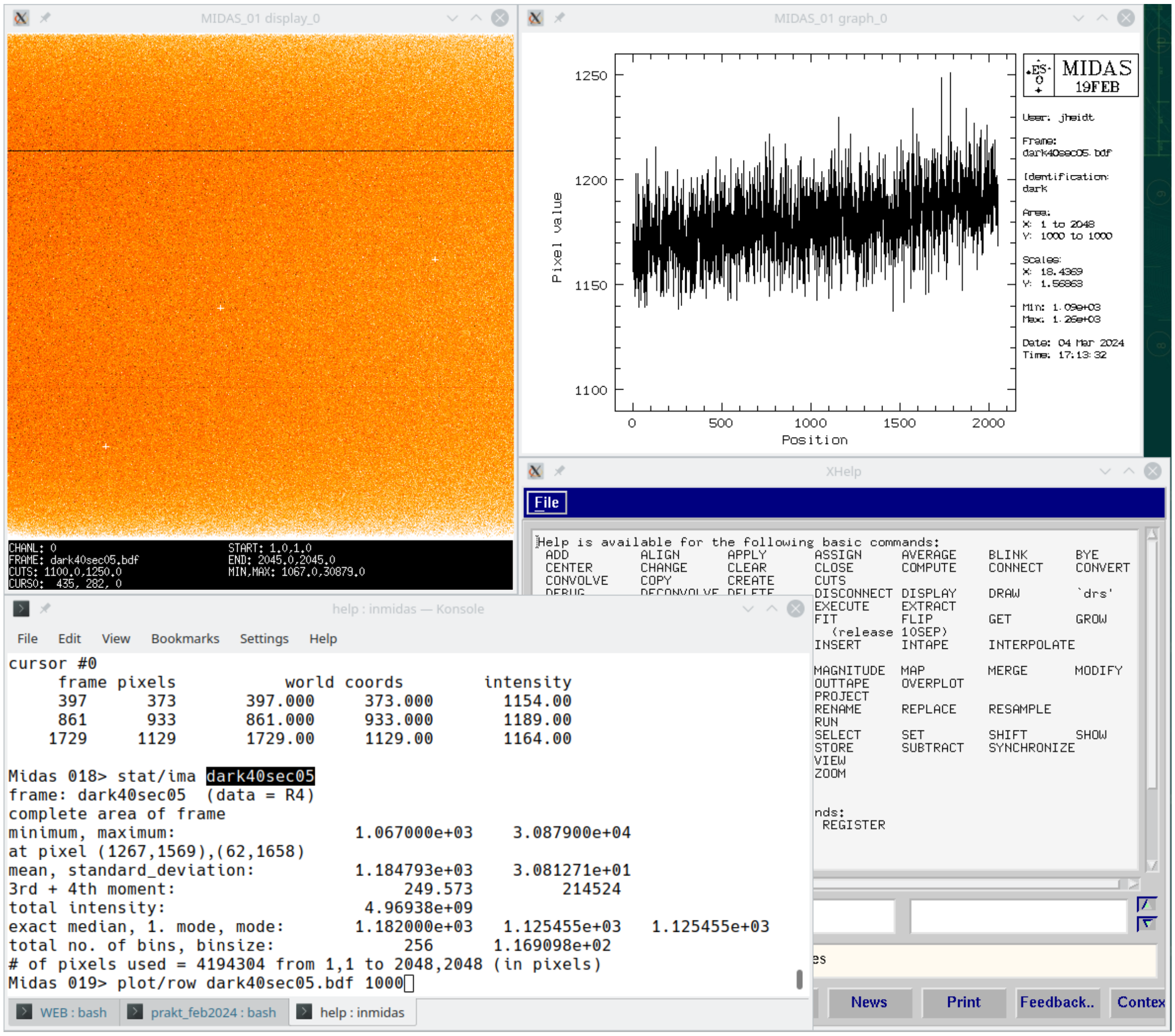
As you will learn throughout this task, the scientific images taken at the telescope need to be corrected for various detector properties and other artefacts to make them ready for science. This can be done via the **COMPUTE/IMAGE out = in** command. All kind of maths can be done via the command, eg
COMPUTE/IMAGE newimage = inputimage - 1000 subtracts from your inputimage 1000 counts and creates a new resulting image, while
COMPUTE/IMAGE inputimage = inputimage / (inputimage-1000) modifies the the contents of the "inputimage" itself (what you should not do). Play a bit with the command and load the resulting images to see how it works.

It is necessary from time to time to inspect an image for large-scale gradients. This may be hard or even impossible, if the gradient is hidden in the noise of the image itself. In that case suppressing the noise by averaging *columns* or *rows* may help. The **AVERAGE/ROW** and **AVERAGE/COLUMN** commands allow you to do that. **AVERAGE/ROW out = in-**

putimage 400,600 for example, averages the rows 400-600 of the “inputimage” and creates the resulting image “out”. Note that the resulting image is 1-dim (you will see just a line in the display when loading it).

STATISTICS/IMAGE is one of the most important MIDAS commands. It allows you to derive various statistical parameters of an image, like mean or median (whats the difference and why should one rather use the median instead of the mean?), min, max etc. By typing **STATISTICS/IMAGE inputimage** the full contents of the image will be used and that is what you shall use. In some cases it is better (similarly to loading an image) to predefine a useful dynamic range for your statistics. This can be done with the parameter *excess*.

If you have performed a **STATISTICS/IMAGE** command and if you would like to inspect the distribution of the pixel intensities than **PLOT/HISTOGRAMM inputimage** is a good idea. If you do an **PLOT/HISTOGRAMM inputimage** without a **STATISTICS/IMAGE** prior to that it will be done automatically by MIDAS (but using the full contents of the image).



```
CHANL: 0                              START: 1.0,1.0
FRAME: dark40sec05.bdf                END: 2045.0,2045.0
CUTS: 1100.0,1250.0                   MIN,MAX: 1067.0,30879.0
CURSO:  435, 282, 0
```

```
User: jheidt

Frame:
dark40sec05.bdf

Identification:
dark

Area:
X:  1 to 2048
Y:  1000 to 1000

Scales:
X:  18.4369
Y:  1.56863

Min: 1.09e+03
Max: 1.26e+03

Date: 04 Mar 2024
Time: 17:13:32
```

```
Help is available for the following basic commands:
 ADD          ALIGN        APPLY        ASSIGN       AVERAGE      BLINK        BYE
 CENTER       CHANGE       CLEAR        CLOSE        COMPUTE      CONNECT      CONVERT
 CONVOLVE     COPY         CREATE       CUTS
                                        DISCONNECT   DISPLAY      DRAW         `drs'
                                        EXECUTE      EXTRACT
                                        FIT          FLIP         GET          GROW
                                          (release 10SEP)
                                        INSERT       INTAPE       INTERPOLATE

                                        MAGNITUDE    MAP          MERGE        MODIFY
                                        OUTTAPE      OVERPLOT
                                        PROJECT
                                        RENAME       REPLACE      RESAMPLE
                                        RUN
                                        SELECT       SET          SHIFT        SHOW
                                        STORE        SUBTRACT     SYNCHRONIZE
                                        VIEW
                                        ZOOM

                                        nds:
                                         REGISTER
```

```
cursor #0
     frame pixels             world coords        intensity
     397       373        397.000     373.000      1154.00
     861       933        861.000     933.000      1189.00
    1729      1129        1729.00     1129.00      1164.00

Midas 018> stat/ima dark40sec05
frame: dark40sec05  (data = R4)
complete area of frame
minimum, maximum:                 1.067000e+03    3.087900e+04
at pixel (1267,1569),(62,1658)
mean, standard_deviation:         1.184793e+03    3.081271e+01
3rd + 4th moment:                      249.573          214524
total intensity:                   4.96938e+09
exact median, 1. mode, mode:      1.182000e+03   1.125455e+03   1.125455e+03
total no. of bins, binsize:                256    1.169098e+02
# of pixels used = 4194304 from 1,1 to 2048,2048 (in pixels)
Midas 019> plot/row dark40sec05.bdf 1000
```

Figure 5: MIDAS in action

1.3.5 Signal, signal-to-noise, mean, median and mode - what does that mean?

In the following a compact summary of some widely used statistical properties in Astronomy is given. Some of it follows the discussion presented in more detail by Olivier Hainault at http://www.eso.org/ ohainaut/ccd/sn.html.

The **Signal (S)** is the number of photons detected from a source. In practice, not all photons that arrive on the detector are detected; only a fraction of them are converted to electrons (the ratio electron/photon is the quantum efficiency, which is typically 50-90% for a CCD). To make things not too confusing we will work in ADU (electrons, which are converted to digital units during the readout process).
The total signal S is related to the number of counts s we get per seconds t, ie

$$S = s \times t$$

The **Noise (N)** is the total random contribution from various sources that affect the measurement of the signal. There are various sources of noise which behave differently. The noise is basically the error on the flux measured S ± N.
The **Signal-to-Noise Ratio, SNR, S/N**, tells how well an object is measured. Some typical values:

- S/N = 2-3: object barely detected
- S/N = 5: object detected, one can really start to believe what one sees
- S/N = 10: one can start to do measurements
- S/N = 100: excellent measurement.

As N is the error on the measurement, 1 / (S/N) is the relative error on the measurement:

- S/N = 100: measurement at 1%
- S/N = 10: at 10%
- S/N = 5 : at 20% (+/- 20%)
- S/N = 2: +/- 50% !

In addition to the various sources of noise, there are different types of noise. On is the so-called **shot noise** and the other one the **readout-out noise, RON**.
If dealing with sources of photons arriving at random, the noise associated with that randomness is

$$N = \sqrt{n}$$

with n the number of photons and which can be from the sky and from the target itself and are written as

$$N_{target} = \sqrt{n_{target}} \text{ and } N_{sky} = \sqrt{npix \times n_{sky}}$$

Note that the noise for the target is derived from just the integrated number of counts, while the noise from the sky is derived from the sky counts per pixel times the number of pixel across which the counts from the target are determined. Its is obvious that observing conditions can have an big impact on the S/N. If the moon is up, the sky is brighter and hence the noise larger, if the seeing is bad and the flux from the target is spread across many pixel, the contribution from the sky becomes stronger. There is also shot noise from the dark current, which adds in the same way as the sky.

When reading the detector, the amplifier(s) involved add some noise to the signal. This is a characteristic of the chip and of the read-out-mode used. If the read-out noise per pixel is ron, than the total RON is than

$$N_{RON} = \sqrt{npix \times RON^2} = \sqrt{npix} \times RON$$

note that all individual read-out noises add quadratically.

Random, uncorrelated noises (such as those described) add quadratically, so, the generic formula for the noise is

$$N_{total} = \sqrt{N^2_{target} + N^2_{sky} + N^2_{dark} + N^2_{RON}}$$

and finally the generic formula for the S/N as a function of signal s and time t:

$$S/N = s \times t \,/\sqrt{s \times t + n_{pix} \times sky \times t + n_{pix} \times dark \times t + n_{pix} \times RON^2}$$

To fully understand what that means, three special cases:

Bright source, faint background: Here the signal from the star is so high that its noise dominates all other noise contributions, the formula reduces than to:

$$S/N = s \times t \,/\sqrt{s \times t} \propto \sqrt{t}$$

In other words, to increase the SN by a factor of 2, one needs to expose **at least 4 times as long** !!!

Faint source, bright background: Here the noise contribution from the sky dominates

$$S/N = s \times t \,/\sqrt{n_{pix} \times sky \times t} \propto \sqrt{t} \,/\sqrt{n_{pix}}$$

If n_{pix} is smaller, S/N becomes larger, ie good seeing (flux of target spread across less pixel) is the key !!!

Very faint source, very faint background, very low dark current: Here the readout-noise dominates, which happens when eg performing high-resolution Echelle spectroscopy. In this case the formula reduces to:

$$S/N = s \times t \,/\sqrt{n_{pix} \times RON^2}$$

Obviously, good seeing helps again. In addition, try to expose as long as possible and minimize the number of readouts (they would go in quadratically).

During the data reduction, it is often necessary to combine images, eg to reduce the read-out noise in the case of biases, or shot-noise in the case of flats to properly determine pixel-to-pixel variations. There are many ways to do that, here only the three most important ones are briefly described, namely **mean, median and mode**. In all cases a log-normal distribution of the data points involved is assumed.

• Mean: This is simply the arithmetic average of a set of images or measurements. It can always been used, but does basically no filtering of data. Extreme outliers enter the averaging. Example: 1,3,4,2,1000 → mean = 202

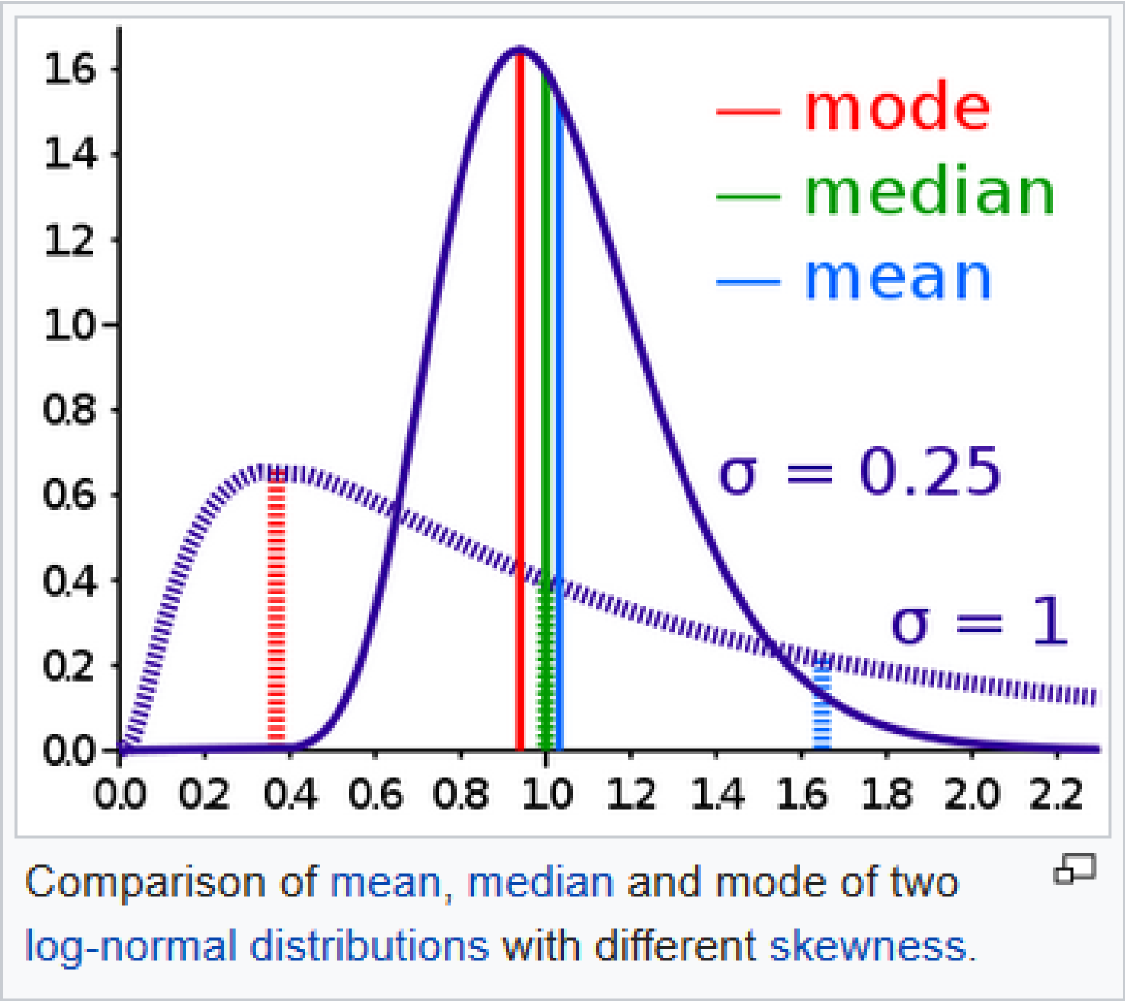
• κ-σ-clipping: Filtering method when averaging all numbers. Use in a first pass all numbers to determine mean and standard deviation σ and determine in a final pass the mean from all numbers κ (typically 1.5-2) $\times$ σ around the mean from the first pass.

• Median: Defined as the middle number in a sequence of uneven numbers, otherwise the average of the two middle numbers in a sequence of even numbers. This is an excellent tool for filtering when combining data as outliers are either ignored or have very little impact. Example: 1,3,4,2,1000 → median = 3

- Mode: Most often number in a large! set of numbers. Its use is particularly useful if the distribution of measurements is skewed.

Note: Median/mode are non-linear operations and should **not** be used for eg combining images for scientific purposes as they don't conserve flux.

Fig. 6 illustrates the difference between mean, median and mode in graphical form.



Comparison of mean, median and mode of two log-normal distributions with different skewness.

Figure 6: Difference between mean, median and mode.

.

1.3.6 The data reduction: step by step

1) You start by analyzing the bias images. Find out if the average bias level is stable as a function of time for the 10 images provided. What is its typical intensity across all frames? Is the median or the mean value a better measure and why?
2) Is the bias value homogeneous across the field of view or does it show a 2-dimensional structure? Why is this information from 1) and 2) important for your data reduction?
3) How can you determine the read-out-noise? The answer depends on whether there is a 2-dimensional structure. Assume that the bias itself is noise-free. Determine the read-out noise.
4) Have a look at the dark current images with the longest integration time available. Is there any difference to the bias images? What is the value of the dark current? Can it be distinguished from the bias level?
5) Inspect now the flatfields. How do they look alike. Compare their appearance to one of the science images.
6) Plan a best possible strategy for correcting the additive parts of the bias and dark current of raw images and the normalization of sensibility (”bias”-, “dark”- and ”normalization”-correction). Once this is done discuss with the assistant your “magic formula”
7) Construct now ’master’ frames for the different types of calibration discussed in 6).
8) Correct one of your object images based on this and compare your reduced and unreduced science image. What is the main difference (there must be one).
9) Compare now the read-out noise you have determined with the background noise in your reduced science image. How long would you at least need to integrate to make sure that your science frame is photon-noise dominated?

All questions raised above can be answered by cleverly using MIDAS-commands described in the introductory guide and considering the facts in the statistics compendium. A few hints on how to speed up the process for question 7): you may create an image-catalog to work on several images in one shot. **An image catalog can be created** via the MIDAS-command: “creat/icat Katalog Inputbilder”, e.g. “creat/icat bias bias*.bdf”. **You can average or median-combine images** via “aver/ima outputbild = Katalog ? ? median/average”, e.g. “aver/ima biasmedian = bias.cat ? ? median”.

1.3.7 Photometry and lightcurve of your δ Scuti star

Since you have many images of your δ Scuti star they will be reduced and analyzed via scripts with catalogs as input.

Copy first your median-combined Dark- and Flatfield-images to the directory, where your full set of target images is located. In this directory you will find the scripts "photo.prg" and "copy.prg". There you will also find a catalog "in.cat" (with images of your target, named "cy_aqui_*" or "gd_0428_*") and a catalog "out.cat" with the images of your target and prefix "r", i.e. "rcy_aqui_*" or "rgd_0428_*"). Now you are ready to reduce and analyze your images in the following order:

• Correction of all images for Dark:
"exec/cata comp/ima out.cat = in.cat - Darkframe (for your images)"
• Flatfielding of all your images:
"exec/cata comp/ima out.cat = out.cat / flatfield"
The basic data reduction is now done on all images, the reduced images have the prefix "r", eg "rcy_aqui_066".
• Now you need to determine the positions of the target and your reference star on all images and to perform aperture photometry of the target and reference star on all images via the script "photo.prg". The script basically uses two MIDAS-commands. One is "center/gauss" which you will use interactively to determine the positions, the other is "magnitude/rectangle", which performs automatically the aperture photometry on the stars whose centers you just determined. What you have to do is:
• To start the script "@@ photo out.cat"
• The script will load the first image and you get a box in the image display, which you need to center on yozur target first, than click left on mouse, move a bright reference star, click again left mouse and finally exit with click on center mouse. For the reference star in the field of CY Aquarius use the bright star about 6' to the left, for GD 428 use one of the brighter stars nearby. Once the centering is done, the photometry is carried out and the next image loaded. Continue until you have worked on all your images. Now you have created n Tables (corresponding to your n Images, with names "grcy_aqui_n" or "grgd_0428_n") which contain the positions and magnitudes of your target and reference star.
• In case you need to redo the centering/photometry on one individual image, there is no need to go through the entire catalog again. Just start the script with
"@@ photo NameOfRedcuedImage" where your centering/photometry has failed and the script will run on just that image.
• Now you have produced the tables with the positions and magnitudes of your target and reference star. The next step is to merge this information in just one table. Run the script "@@ copy.prg", which creates a table "photo", copies magnitudes and their errors from the tables you just made and copies from the headers of the images the exposure time and the start of the observations into it. It finally calculates the differential magnitude between your target and reference star and the error. Finally, it shows you the first 5 entries of the table "photo" in order to judge whether the script went well.

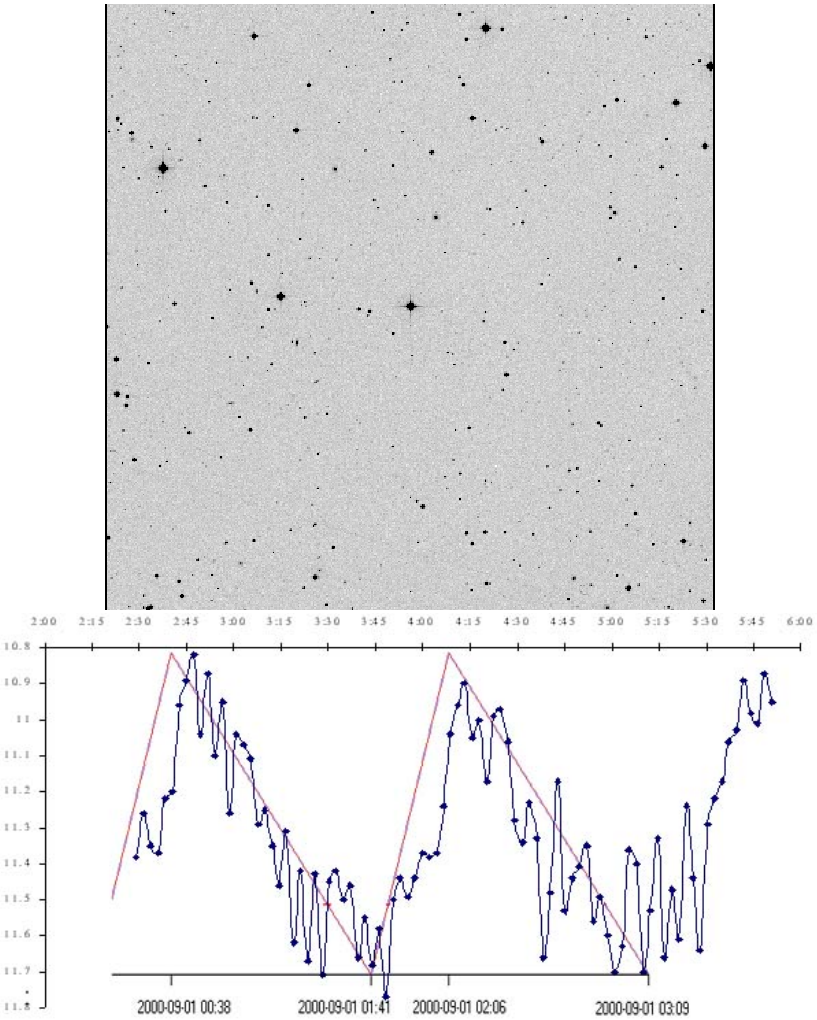


Figure 7: Top: Finding chart for CY Aquarius, field of view 23’$\times 23'$, north is up, east to the right, object is in the center, the comparison star to the left.
Position: $\alpha = 22^h\ 37^m\ 47.82^s$; $\delta = 1^o\ 32'\ 5.1''$ (2000.0)
Bottom: Lightcurve of CY Aquarius from an amateur astronomer.

.

• Finally, the lightcurve will be displayed. If you have not already done, open a graphics window and plot the contents of the table via:
“plot/tab photo :start_time :diff” (makes the lightcurve)
“over/error photo :start_time :diff :error” (overplots the errors).
• You can now inspect the lightcurve, estimate the timescale of the variation and its amplitude. Once this is done a final discussion on the lightcurve will follow.

1.4 A concrete application: The CY Aquarius / GD 428 Deep Field

Deep fields are nowadays a standard tool for a wide range of astrophysical applications. They allow e.g. unprecedented views back in time, allow to constrain e.g. number densities as a function of stellar type and position in galaxies to determine their star formation history or to peer deep into dense stellar fields like e.g. globular clusters.

The Hubble Deep Field is probably the most popular Deep Field of its kind. Although small in its field of view (only a few square arcminutes) it revolutionized our understanding of galaxy formation at early epochs. It is not only the sharpness of the images which allowed to determine the morphology of galaxies when the Universe was just a few Gigayears old but also to determine their distances (and to an lesser amount their dominant stellar population) by means of so-called photometric redshifts.

On the other hand they illustrate nicely the limits of a telescope/instrument which one has been used and that is what we will do with our data set of your δ Scuti star. The idea is simple. One observes a field as long as possible by taking a big set of images, reduces them, combines them and finally counts the number of stars/galaxies in the field as a function of magnitude or other parameters. Here we will just use the number counts to derive a limiting magnitude (often called "completeness limit") at a given exposure time out of your data set.

To derive a Deep Field out of your data, the images reduced in the previous section must first be cleaned of cosmic ray hits (to avoid false identifications), subtracted for their background (for averaging purposes) and finally be stacked, ie summed up. Since the stars are not on every frame on the same position, the images must be shifted in order to overlap. This will be done using the information derived from centroiding the images.

To create the final, summed image, the script "backsubtr" shall be used. Within this script each image is corrected for cosmic ray hits, the background subtracted and all of them aligned wrt to each other. To execute the script simple type: "@@ backsubtr out.cat". If everything works as expected the outcome should be a file called "final".
Once this is done, inspect the image in the image display. Compare it with one of the individual images ("frcy_aqui_*" or "frgd_0428_*"). What is the main difference?

Before you can start to analyze the images you must first flux-calibrate them to an absolute scale. This is normally done via the so-called *zero point*. The zero point is the magnitude of an object, from which you receive 1 count / sec. This stems from the classical definition for the magnitude of an object observed with any system:

$$\mathrm{m_{object}} = -2.5 \times \mathrm{log_{fluxfromobject}} + \mathrm{C}$$

with C a constant dependent on the combination telescope, instrument and atmosphere.

There are various ways to determine the zero point depending on the accuracy of the calibration required. Normally, one observes a set of so-called standard stars throughout the night, whose brightness is known to determine the zero point (and also color-corrections if relevant). This can only be done during so-called "photometric" observing conditions (clear skies all night), however. Alternatively, if the field of view is large enough, one can use the calibration of large surveys eg from GAIA or the Sloan Digital Sky Survey (SDSS). These surveys are well calibrated and brightnesses for a large number of stars up to a certain brightness limit can be used. If it happens that one of the stars in your field is included in such a survey you can use the published brightness of that star to derive the zero-point on your frame. The big advantage of this procedure is that it is independent of the observing conditions - and they vary in Heidelberg pretty strongly.

Determination of the zero point for the CY Aquarius field

For the determination of the zero point for the CY Aquarius field, we will use the SDSS. Open a browser in the laptop and go to sdss.org. Then click on “data”, “imaging data”, “explore sky server” and finally on “navigate” (at the “data access” menu). In the upper left corner at “name” type in “CY Aquarius” and click on “resolve”. Now an image should be loaded with a bright star in its center, which is CY Aquarius. On the right hand side you see information about your object which is the RA, Dec, type and its ugriz-magnitudes. This example shows you already the pitfalls of the automatization of surveys because this star is classified as a galaxy!!! Unfortunately, we can not use the listed r-magnitude (which is the one relevant here) for calibration of our 70cm images, but will use the comparison star which you have used instead. Why?

On the left hand side click on the +/- symbol until the field of view is large enough such that you can unambiguously identify the comparison star. Move the comparison star to the center and zoom in. If you now click on “photometric object” some of the objects in the field are encircled. Now click on some of them to watch hown their brightnesses and types change. Finally click on the comparison star and note down its r-magnitude.

Note: Sometimes you may encounter an error like “No data found”. In that case choose Firefox/settings/history in your browser and delete cache and site settings.

Determination of the zero point for the GD 428 field

Unfortunately, GD 428 is not in the footprint of the SDSS. We thus have to rely on the average brightness of GD 428 published to set the zero point, which will be accurate to 0.15 magnitudes only. This can be found on a huge stellar database named “Simbad” (Set of Identifications, Measurements and Bibliography for Astronomical Data). Open a browser and go to simbad.u-strasbg.fr/simbad/. Then click on queries “by identifier” and type in the identifier field “gd 428” and “submit id”. Simbad will post a number of basic data on GD 428 including its average R-magnitude, which we will use.

Now the image can be analyzed. This is done by using the standalone program called ”SExtractor” (Source Extractor, http://www.astromatic.net/software/sextractor). It requires some parameter files (default.*) where you a) define the way an object is detected on images (several aperture diameter for photometry, the amount of pixel connected above a certain background threshold to define an object etc.) in the files default.sex and b) its output (default.param). For our purposes the parameter files are already predefined. Using those the SExtractor tries to detect/identify stars/galaxies on an input frame, performs a characterization (magnitude, morphological appearance etc.) and write everything in a table. This is simply done by starting the script sexy.prg by ”@@ sexy”. It will than automatically search and identify stars/galaxies in your Deep Field, loads the image, automatically overplot the position of the objects in the field found as well as its number entry in the resulting table ”stars”.

Have now a close look at what is displayed. How well are the objects detected? How many have been found?

Now you need to set the zero point of your image. Check the number listed next to your comparison star for CY Aquarius or the target itself for GD 428. If you have problems to read it properly because the field is too packed with stars, execute the script “starfinder” by typing ”@@ starfinder”. This will display the full image without any overlay and starts a cursor. Use move the cursor in the field to the comparison star for CY Aquarius or GD 428 itself, click on it and exit. Starfinder will then display a zoomed portion of the image where you clicked and display the number in the table “stars” next to it, which you should now be able to read. Now display the contents for this entry in the table stars by ”read/table stars @i” with i the number of the comparison star in the table. Compare the difference to the calibtrated magnitude and the one listed in the table at ”mag_best” and correct for the difference by
”comp/table stars :mag_best = :mag_best - ”difference mag_best - calibrated magnitude”.

To give you an example: if mag_best = 13.15 and your calibrated magnitude is 10.36, then you have to type
”comp/table stars :mag_best = :mag_best - 2.79.

This will automatically correct the magnitudes of all objects in the table stars to the same zero point. Finally, what is the limiting magnitude for the column ”mag_best” in the table ”stars” (think about how one can determine the limiting magnitude of an image)?

MIDAS-commands which you may find useful to explore the SExtractor output (try to understand them using “crea/gui help” within MIDAS, you have already quite some experience with it):

READ/TABLE
LOAD/TABLE
PLOT/TABLE
PLOT/HISTOGRAM (of a table)

Hint: There are various ways of defining a “limiting magnitude”. One is that one refers to a completeness limit of 100%, ie that all objects at or brighter than a given magnitude are detected by SExtractor.

Once this is done compare your result with one of the individual images you just have reduced. You need first to convert the image in fits-format by ”outdis/fits frcy_aqui_*.bdf (or frgd_0428_*.bdf) final.fits” and redo ”@@ sexy”. What is the difference in limiting magnitudes? Would you expect the difference based on the depth of the images? Consider the following: by which amount do you need to change the exposure time to increase your completeness by 1 magnitude? Assume that your signal-to-noise is dominated by the photon noise from the objects.

The Hubble Deep Field has a limiting magnitude of about 26.5mag. How long would you need to integrate with the 70cm telescope to get a as deep CY Aquarius or GD 428 Deep Field based on the experience with your recent data set? Assume again that your signal-to-noise is dominated by the photon noise from the objects (which is true in practice for the HST data only).