Subaru Data Reduction CookBook:
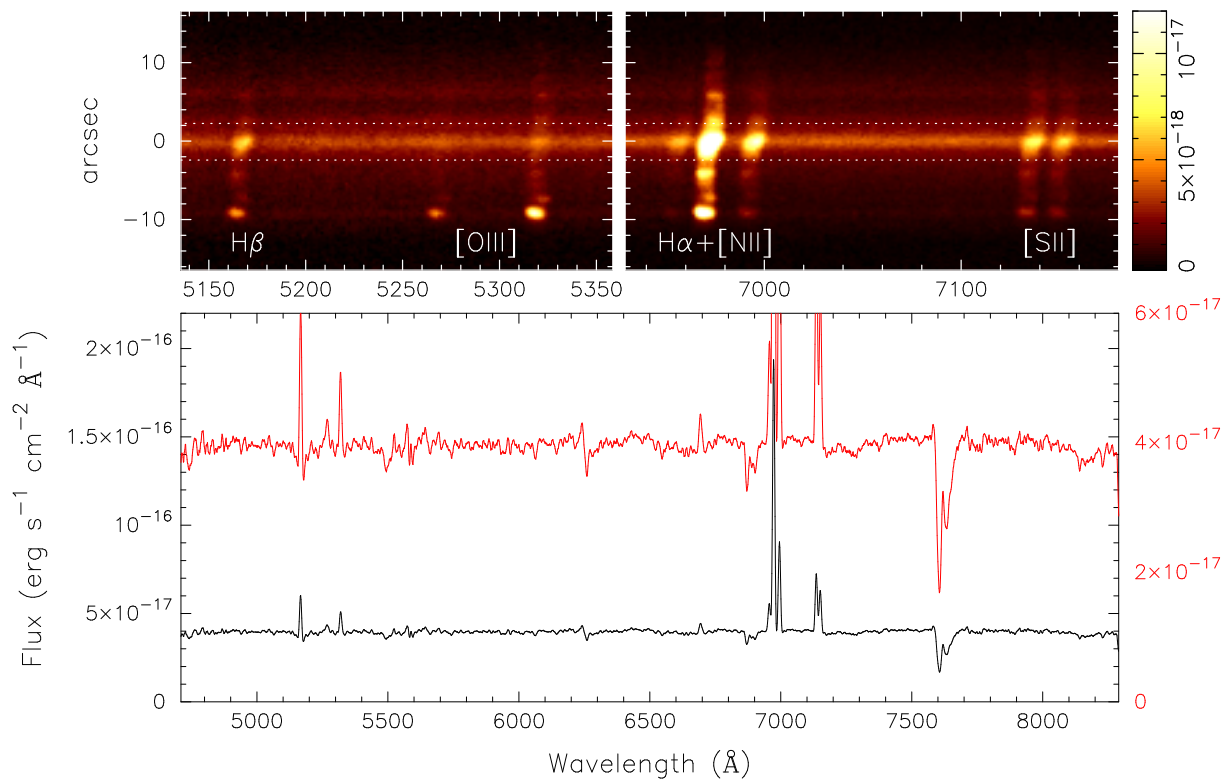# Long-Slit Spectroscopic Observations with FOCAS

— VERSION. 1.0.2E (DECEMBER 8, 2008)—

1 Forward

This CookBook describes a standard procedure to reduce long-slit spectroscopic data taken with the Faint Object Camera and Spectrograph (FOCAS) at Subaru Telescope using Image Reduction and Analysis Facility (IRAF)[1]. We attempt to limit using tasks implemented in IRAF, except for a few process required specifically for FOCAS, such as bias subtraction using the overscan region and distortion correction. We believe that the readers can pursue their knowledge acquired from this CookBook toward data analysis of other instruments than those FOCAS. Although FOCAS offers a large variety of observing capabilities, we focus on describing on how to reduce long-slit spectroscopic data. As for handling data taken with other modes, please see the appropriate documents. Last but not least, we appreciate reader feedback to help improve this CookBook.

2008 October 30

T. Hattori (author of the original version in Japanese)
E-mail : hattori "at" subaru.naoj.org

R. S. Furuya (current editor of English version)
E-mail : rsf "at" subaru.naoj.org

---

[1]See `http://iraf.noao.edu/` or `http://iraf.net` for details on IRAF.

Contents

Revision History

Version	Date	
1.0.0e	2008 October 31	First release based on the Japanese version (2008 October 9) by R.S.F. & T.H.
1.0.1e	2008 November 30	Language correction for *Subaru Asia Winter School* and added Appendix B by R.S.F., T.H., & A.H.
1.0.2e	2008 December 8	Minor corrections after the winter school by R.S.F.

2 Spectroscopic Data Acquired with FOCAS

2.1 FOCAS; *First Fact*

FOCAS — Faint Object Camera and Spectrograph — is one of the cutting edge instruments at Subaru Telescope, and has the following capabilities: imaging, spectroscopy, polarimetric imaging, and polarimetric spectroscopy in the optical regime. In addition, the instruments allows multi-object spectroscopy (MOS) observations. For further information about the instrument, please refer to `http://www.naoj.org/Observing/Instruments/FOCAS/` where you can also obtain the most up to date information.

2.2 A Standard Procedure of Spectroscopic Observations with FOCAS

FOCAS is equipped with two CCDs (hereafter designated chip1 and chip2) whose combined field of view is shown in Figure 1. An exposure creates two image files. The odd and even file numbers correspond to the images from the chip1 and chip2, respectively. As shown in the right-hand panel of Figure 1, there are two slit positions, "center" and "offset", for long-slit observations; these positions are used accordingly for the desired observing wavelengths.
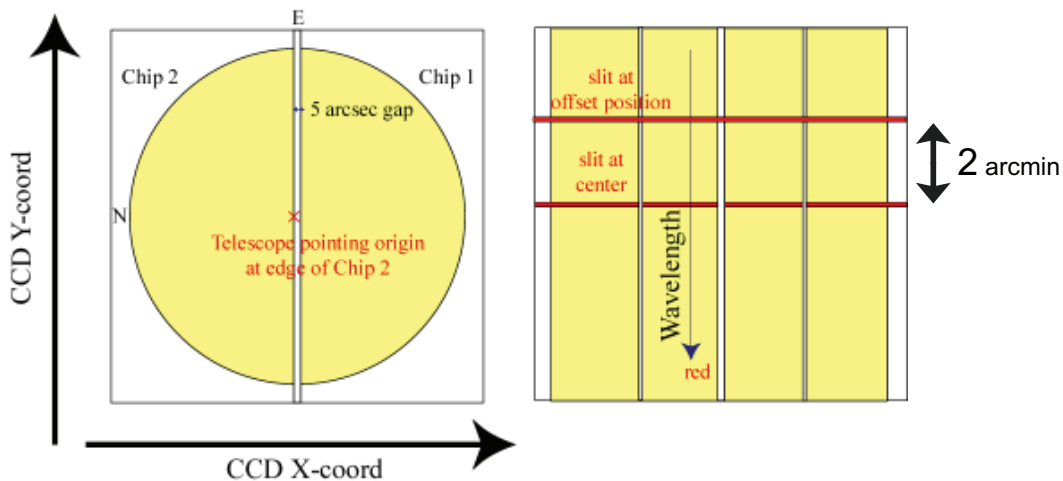
Figure 1: FOCAS field of view; the diameter is 6.0 arcmin.

The left panel of Figure 2 represents an R-band image taken with the chip 2 (the image was taken with the imaging mode) of the spiral galaxy, SDSS J000347.01-000350.3, which is located at the image center and is the object that we are going to work on. In the acquisition

procedure for FOCAS, we take several small field images (Figure 2) to get the target into slit position. Subsequently, a grism is inserted to disperse the incident light along the direction perpendicular to the slit. This results in a long-slit spectrum.
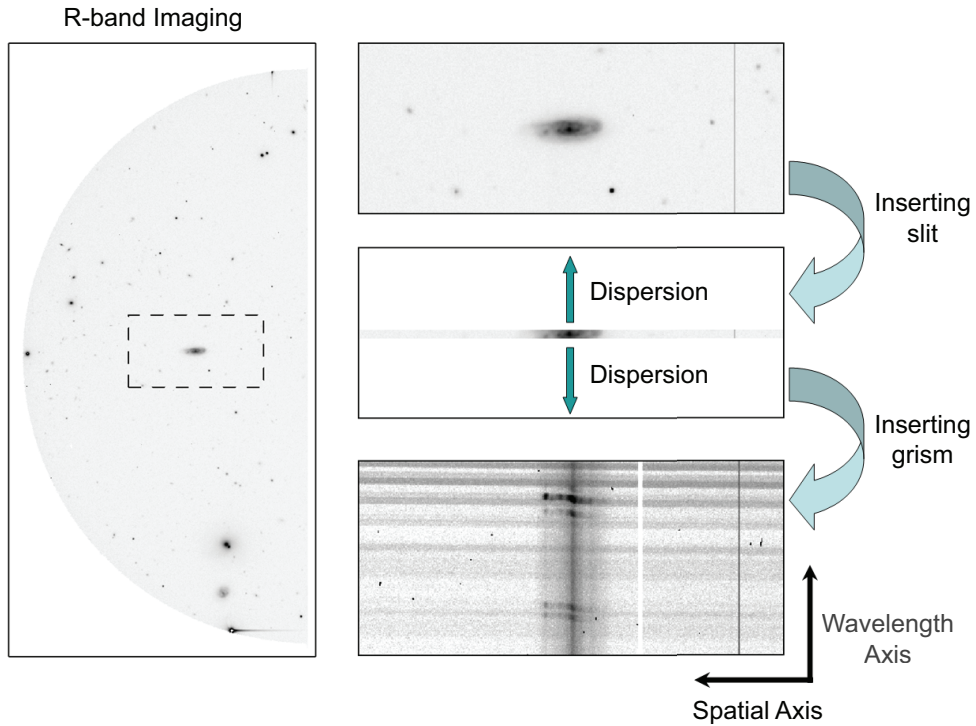
Figure 2: Concepts of long-slit observations. The R-band image shown in the left-hand panel contains the target spiral galaxy in the dashed-rectangular box.

2.3   Strategy for Acquiring Calibration Data

Below is a suggested guideline for planning your calibration observations. Keep in mind that the calibration strategy below is an example, and should be optimized accordingly to your scientific goals [e.g., observing mode(s), required accuracy etc...].

bias — Bias frames are taken before or after scientific observations. For many observations, you can utilize the overscan regions for subtracting the bias level. If you want to use a rather wide area with respect to the field-of-view, we suggest taking several bias frames.

dark — Dark frames are not taken for the majority of FOCAS observations because the current level is negligible.

flat — Dome flats are taken either before (i.e. twilights) or after (i.e. dawn) scientific observations. If working on archival data and if dome flat data do not exist for the source night that the archival data were taken, check whether dome flat data exist on other observing nights within the same observing program.

comparison — This is calibration data used to associate the pixel coordinate with the absolute wavelength, which is sometimes referred to as "wavelength calibration". To calibrate wavelength short-ward of 5000 Å, we strongly suggest taking a comparison at the telescope position (i.e., elevation angle) identical to the target source(s). The reason behind this is that sky emission lines cannot be used as a "comparison" at this wavelength regime. If you don't have such a comparison and if you have to use comparison(s) that were taken at different elevation angle(s), you will probably have to accept the degraded calibration accuracy on the order of a few pixels.

standard star — We take standard star data either before (i.e. twilights) or after (i.e. dawn) observations. Given your scientific goals, consider the following two possibilities. One is to use the 2.″0 width slit for obtaining bulk of the photons, and the other is to use the same narrow slit that was used to obtain the scientific exposure.

3 Data Reduction; *Overview*

Although there are various methods to reduce long-slit spectroscopy data, we favor keeping the spectroscopic data in 2D format (i.e., without extracting spectrum) up to the final reduction step. The reasons why we favor this strategy are:

- It is generally difficult to extract spectrum of objects which cannot be detected in a single exposure, but can be seen after combining many frames.

- If the emission is more extended than the seeing size, one may want to keep the spatial information.
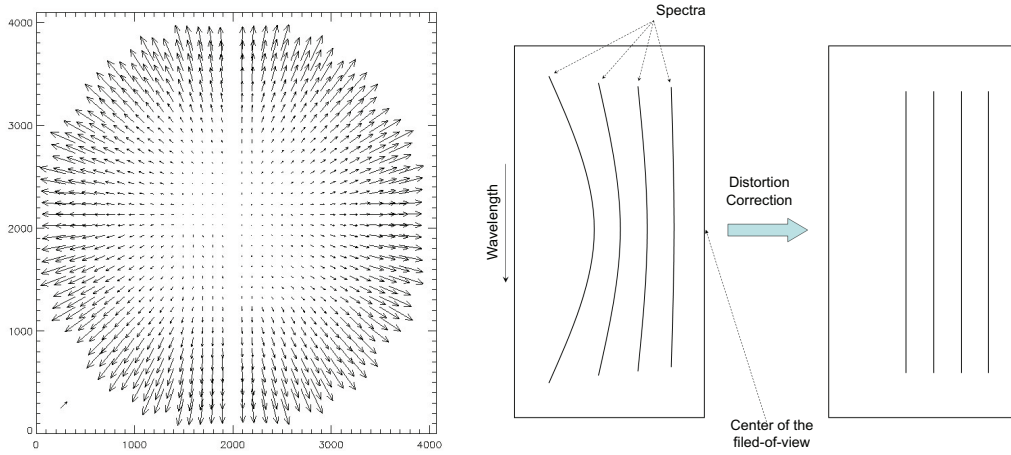
Figure 3: (left) — Distortion pattern of FOCAS in imaging mode (Kashikawa et al. 2002[2]). The left panel illustrates the distortion pattern map that represents the position offsets with respect to a regular grid pattern. (right) — A sketch illustrating of distortion correction for spectroscopic data.

The heart of data reduction can be summarized as,

- Associating the x-axis of the data with the spatial axis, and the y-axis as the wavelength axis

- Converting the count value of the data into a physically sensible unit such as flux density ($\mathrm{erg\ s^{-1}\ cm^{-2}\ \text{Å}^{-1}}$)

The below is the outline of the reduction steps.

1. Bias-subtraction (§5.2)
   Subtracting count offset that was generated during the read-out process of CCDs.
   A bias image taken without exposing the chip (i.e, read-out only) is used to subtract such an offset from all the images.

2. Flat-fielding (§5.3)
   Correcting inequity of the pixel-sensitivity due to the intrinsic performance of the CCDs and/or the optics. This correction can be done by dividing all the images with a flat-field image. Here, the flat field image is obtained through observing uniform light (in this case, dome flat lamps) keeping the instrument configuration identical to that used for scientific observations.

8

3. Distortion Correction (§5.4)

It is generally impossible to guide incident light into an instrument without aberration. Among several effects of aberration, distortion is the most significant and recognizable effect in images. FOCAS data also have such an artifact; the "raw" spectra seen in the image always show a series of "curved spectral features" (see Figure 3). To correct such a distortion pattern, we offer a script that specifically handles FOCAS data.

4. Wavelength Calibration (§5.5)

Associating the pixel coordinates in the CCD with the absolute wavelength by comparison with emission data whose absolute wavelength(s) are well-known.

5. Subtracting the background emission (the so-called sky-subtraction; §5.6)

For slit-spectroscopy observations at optical wavelengths, the slit positions which are free from the source-emission (usually existing on both sides of the source-emission) can be used to calculate the background level.

6. Flux calibration (§5.7)

By using the data of standard star(s) whose absolute flux(es) is/are well-known, we convert unit of the image from count values to flux density.

7. Combining multiple exposures (§6.1.1)

8. Extracting desired spectrum from the image (§6.3)

The readers should bear in mind that the data reduction strategy described in this COOKBOOK is one of many possible approaches.

For instance, one may extract spectrum at an early step (§5.5.2)

of the data reduction process, if observing a bright star.

The readers are advised to use caution when analyzing data taken with the instruments and/or observing modes with which he/she is not familiar. Check the resultant images in each procedure. In some cases, you MUST optimize the methods and parameters shown in this COOKBOOK.

4 Getting Started

4.1 The Sample Data

The sample data of the spiral galaxy SDSS J000347.01-000350.3 (Figure 2) was acquired in 2006 October as part of an educational program for the Graduate University for Advanced Studies. The data are now open to the public. You can retrieve them from the archive system. Table 1 summarizes the parameters characterizing the observations.

Source	:	SDSS J000347.01-000350.3
Slit	:	$0''\!.8$ center
Grism	:	300 B (SCFCGRMB01)
Filter	:	Y47 (SCFCFLSY47)
Exposure Time	:	$600\,\mathrm{sec} \times 3$ ($\pm 3''\!.0$ dither)
Binning	:	3 pixels along spatial axis, no-bining along wavelength axis

Frame No.	Object Category
FCSA00079163–172	Dome flat (5 frames constitute a set)
FCSA00079183, 184	Standard star (BD+28d4211)
FCSA00079261–268	SDSS J00347.01: Acquisition images
FCSA00079269–274	SDSS J00347.01: Spectroscopic data
FCSA00079567–570	Comparisons
FCSA00079961–970	Bias frames (5 frames constitute a set)

Table 1: A Summary of the Observations and the Key Parameters of the Data

The sample data can be obtained from

        http://optik2.mtk.nao.ac.jp/subaru_red/FOCAS/testdata_20061204.tar.gz

After getting the gzipp'ed tar archive, extract the data by,

```
ana03{hattriak}: mkdir /mfs01i/hattriak
ana03{hattriak}: cd /mfs01i/hattriak
```

Bring the downloaded tar archive under the directory made above, then extract the file by,

```
ana03{hattriak}: tar zxvf testdata_20061204.tar.gz
  ./testdata
  ./testdata/FCSA00079273.fits
  ./testdata/FCSA00079261.fits
```

10

.

.

.

4.2 IRAF

If you have never launched IRAF on the machine that you are going to use, you will have
to customize your environment by issuing `mkiraf` as shown below. In this CookBook we
assume a user environment with IRAF version 2.12.2a or later.

```
ana03{hattriak}: mkiraf
-- creating a new uparm directory
Terminal types: xgterm,xterm,gterm,vt640,vt100,etc.
Enter terminal type: xgterm
A new LOGIN.CL file has been created in the current directory.
You may wish to review and edit this file to change the defaults.
```

We suggest selecting `xgterm`. `mkiraf` will create `login.cl` file and an `uparm` directory under
the current directory. Add the following line to the `login.cl` file using your favorite editor
(e.g., emacs). The line will call some tasks optimized for data taken with FOCAS.

```
task $focasred.pkg = "/home/hattriak/FOCASRED/IRAF/focasred.cl"
```

Instead of typing the above, you can copy the above script under the directory where you
launched `mkiraf`.

```
ana03{hattriak}: cp /home/hattriak/temp/loginuser.cl ./
```

Start `xgterm` as,

```
ana03{hattriak}: xgterm -sb &
```

where the option `-sb` gives a scroll bar in the xgterm. Type `cl` from the command line of
`xgterm`.

```
ana03{hattriak}: cl
 NOAO PC-IRAF Revision 2.12.2a-EXPORT Wed Jul 14 20:45:34 MST 2004
 This is the EXPORT version of PC-IRAF V2.12 supporting most PC systems.
```

```
Welcome to IRAF.  To list the available commands, type ? or ??.  To get
detailed information about a command, type 'help command'.  To  run  a
command  or  load  a  package, type  its name.  Type 'bye' to exit a
package, or 'logout' to get out of the CL.  Type 'news'  to  find  out
what is new in the version of the system you are using.  The following
commands or packages are currently defined:

   apropos     dbms.       language.   obsolete.   softtools.   tables.
   color.      focasred.   lists.      plot.       stsdas.      utilities.
   dataio.     images.     noao.       proto.      system.
```

We also should read in the `focasred` package by typing:

```
cl> focasred
        bigimage        fsenscalib      fwavecalib_sky   ovsub
        distcalib       fskysub         lacos_im         resume_wcs
        flatnorm        fwavecalib      lacos_spec
    fo>
```

Here, in this CookBook, we present commands that should be issued in the IRAF environment by e.g., `fo>` and the Unix/Linux commands by `ana03{hattriak}:`.

We strongly encourage the readers to record all the executed commands as well as the messages into a log-file by copying and pasting them there. This will certainly help you to recall what data reduction was performed for future references[2].

For your knowledge, IRAF has a history function which memorizes commands and allows re-use of them. To use this function, type

```
    fo> e
```

where "e" is the first letter of `ehistory`. You can select one of the commands which was executed from the terminal by ↑ or ↓ keys to re-use as another command by editing them with e.g., ←→ or/and "Delete" keys accordingly.

---

[2]If you want to record automatically all the commands that you issue, do the following:
```
fo> cl.logfile = "logfile.cl"
fo> cl.keeplog = yes
```

4.3   Image Viewer

All the data taken with Subaru Telescope (and the majority of the other telescopes as well) are stored in the universal file format system of FITS — Flexible Image Transport System. To examine your images in FITS format or/and to analyze the data interactively with IRAF, it is convenient to open the widely used image viewer, `ds9`. To launch `ds9`, go to the directory where you extracted the tar-ball,

```
ana03{hattriak}: cd testdata
ana03{hattriak}: ds9 &
```

To open an image file, select the image by the GUI menu ("File"→"Open") or use the IRAF command `display` as follows[3]:

```
fo> display FCSA00079184.fits
```

Although the above two methods can display an image similarly, please note that some functions in `ds9` will not function for the latter case.

5   Data Reduction

5.1   Preparation; Making a List of Files

Before starting practice of the data reduction, we have to prepare a list file that contains the names of the data files in ASCII text format. Such a list of files will be used as input to parameters that handle multi-files. This avoids having to repeat the same operation on each file.

For FOCAS data, it is convenient to split the data from chip 1 and chip 2, as follows:

```
ana03{hattriak}: mkdir chip1
ana03{hattriak}: mkdir chip2
ana03{hattriak}: mv *[13579].fits chip1/
ana03{hattriak}: mv *.fits chip2/
```

Here, the command in the third line moves all the files having odd numbers to the `chip1` directory.

Since we will be reducing only chip 2 data at this time, make a list of files for chip 2 as follows (see Table 1):

---

[3]As a default image size, the display command only shows the central 512×512 pixels. To display the entire image, specify the size of the image first, i.e., `fo> reset stdimage= imt1x4`

```
ana03{hattriak}: ls chip2/FCSA000799*.fits > list.bias
ana03{hattriak}: ls chip2/FCSA000791[67]*.fits > list.flat
ana03{hattriak}: ls chip2/FCSA0007918*.fits > list.obj
ana03{hattriak}: ls chip2/FCSA0007927*.fits >> list.obj
ana03{hattriak}: ls chip2/FCSA000795*.fits >> list.obj
```

While the above list has a name of "list.obj", it contains not only the target object but also standard star, and the other calibration frames (e.g., comparisons).

If you want to check the images one by one, use the IRAF command hselect as follows:

```
fo> hselect chip2/*.fits $I,DATA-TYP,OBJECT yes
FCSA00079164.fits        DOMEFLAT    DOMEFLAT
FCSA00079166.fits        DOMEFLAT    DOMEFLAT
FCSA00079168.fits        DOMEFLAT    DOMEFLAT
  .
  .
  .

fo> hselect chip2/*.fits $I 'OBJECT=="DOMEFLAT"' > list.flat
fo> hselect chip2/*.fits $I '@"DATA-TYP"=="OBJECT"' > list.obj
fo> hselect chip2/*.fits $I '@"DATA-TYP"=="COMPARISON"' >> list.obj
```

You may also utilize noao.imred.ccdred.ccdgroups:

```
fo> imred
im> ccdred
cc> ccdgroups chip2/*.fits output="list." group=title
```

5.2   Bias Subtraction

A "combined bias image", or simply "bias image", can be made by combining each "bias frame" as follows:

```
fo> imcombine @list.bias stdbias combine=median
```

where imcombine is an IRAF task to combine images.

```
list.bias          :   A list file containing names of the images to be combined
stdbias            :   Output image name
combine = median   :   Option to combine using "median-filtering"
```

14

The `imcombine` task has many parameters other than those listed the above. All the parameters can be edited by `epar` accordingly.

```
fo> epar imcombine
```

```
                            I R A F
                  Image Reduction and Analysis Facility
        PACKAGE = immatch
           TASK = imcombine

        input   =             @list.bias  List of images to combine
        output  =                stdbias  List of output images
        (headers=                       ) List of header files (optional)
        (bpmasks=                       ) List of bad pixel masks (optional)
        (rejmask=                       ) List of rejection masks (optional)
        (nrejmas=                       ) List of number rejected masks (optional)
        (expmask=                       ) List of exposure masks (optional)
        (sigmas =                       ) List of sigma images (optional)
        (logfile=               STDOUT) Log file

        (combine=             median) Type of combine operation
        (reject =               none) Type of rejection
        (project=                 no) Project highest dimension of input images?
        (outtype=               real) Output image pixel datatype
        (outlimi=                   ) Output limits (x1 x2 y1 y2 ...)
        (offsets=               none) Input image offsets
        (masktyp=               none) Mask type
        More
                                                          ESC-? for HELP
```

After editing parameters, quit and save by ":q" or type ":go" to run the program. In this CookBook, we will show options for an IRAF task executed from the command line. However, it would probably be convenient to use `epar` for parameter editing.

Verify the resultant "bias image", `stdbias.fits`, with `ds9`. You will see that the counts of the "bias image" are mostly distributed around 11,700 counts, except for the dead column

at $x = 530$. In addition, we advise checking the mean and the standard deviation of the pixel counts in the central region by,

```
fo> imstat stdbias[201:500,1001:2000]
```

First, we make a file list using the names of the bias-subtracted images:

```
ana03{hattriak}: sed -e 's/chip2/bs/' -e 's/fits/bs.fits/' list.flat > list.flat.bs
ana03{hattriak}: sed -e 's/chip2/bs/' -e 's/fits/bs.fits/' list.obj > list.obj.bs
ana03{hattriak}: mkdir bs
ana03{hattriak}: cat list.flat.bs
        bs/FCSA00079164.bs.fits
        bs/FCSA00079166.bs.fits
        .
        .
        .
```

Here, we used sed (Stream EDitor) on Unix/Linux to replace the string "chip2" with "bs", and "fits" with "bs.fits". If you are unfamiliar with sed, you can directly edit the file (but don't forget to copy the original file to another file that will be edited). The third line makes a directory, bs where the bias-subtracted files will be stored then have its contents examined using the cat command.

Second, we subtract stdbias.fits using the IRAF task imarith, and save the results as bs/*.bs.fits.

```
fo> imarith @list.flat - stdbias @list.flat.bs
fo> imarith @list.obj - stdbias @list.obj.bs
```

Verify the resultant images stored in the directory.

Third, we subtract the overscan region which are columns along the edge of the image that are not exposed to the light. This region is used to monitor the variation of the bias level in each readout. The sample data consists of a rectangular overscan'ed region having a 20-pixel width to the right-hand side of the image. Before subtracting the overscan region, let's check the counts of bias-subtracted images using task implot.

```
fo> implot bs/FCSA00079184.bs.fits
```

16

Once you run `implot`, a window named `irafterm` will pop up. The window with the label "Line 2048" shows a plot of the data points at $y = 2048$, and the horizontal axis of the plot is the x-coordinate. On the window, type ":`c 665 683`" to show the pixel values for the overscan'ed region as a function of y. Although we have subtracted the "bias image", some of the pixels still show non-zero counts because bias levels between the bias frame and the object frame (`FCSA00079184.fits`) are different. In order to correct for such a time variation, we should use an offset so that the overscan'ed pixels have the zero counts.
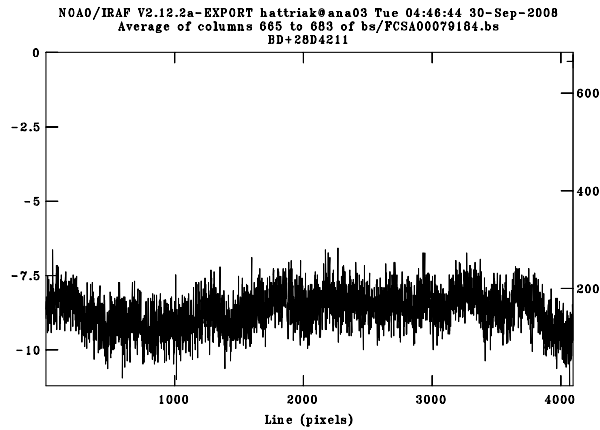
Figure 4: The overscan'ed region after subtraction of `stdbias.fits`

```
fo> wcsreset @list.flat.bs world
fo> wcsreset @list.obj.bs world
fo> ovsub @list.flat.bs @list.flat.bs
fo> ovsub @list.obj.bs @list.obj.bs
```

Above, we have initialized[4] the image coordinates with `wcsreset`, and subtracted the overscan'ed regions with a task `ovsub` implemented in `focasred`. It should also be noted that we have overwritten the overscan-subtracted image onto existing files.

---

[4]It should be noted that the above initialization should **NOT** be applied to imaging data. The initialization is aimed to clear a known bug in the FITS header of spectroscopy data.

5.3 Flat Fielding

The pixel-to-pixel variations (i.e., inequity) in sensitivity must be removed using a "flat field image" that is obtained by observing dome flat lamps. We make such a flat field image with the same procedure as we did for the "bias image", as follows:

```
fo> imcombine @list.flat.bs flat combine=median
```

Our experience suggests that "median filtering" usually gives a reasonable "flat field image". After displaying the image with `ds9`, verify the count distribution along the y-axis by:

```
fo> implot flat
```

then, type ":c 350" on the window. You will find the representative count value is $\sim 10,000$[5]

Next, we divide the image by 10,000 so that the mean of the count becomes unity, i.e., to normalize the flat image:

```
fo> imarith flat / 10000 flat.nr
```

Using the normalized flat image, we divide the object frames,

```
ana03{hattriak}: sed 's/bs/ff/g' list.obj.bs > list.obj.ff
ana03{hattriak}: mkdir ff
fo> imarith @list.obj.bs / flat.nr @list.obj.ff
```

Here, we made another list `list.obj.ff` that was created by replacing the string "`bs`" in `list.obj.bs` with "ff". The `list.obj.ff` file is used as a list of the output files, and the resultant images will be written as `ff/*.ff.fits`.

5.4 Distortion Correction

All the FOCAS images will show a radial pattern expanding outward, as shown in Figure 3, an artifact due to the distortion. This pattern is usually recognized as curved pattern(s) in the case of the spectroscopic observations. Open "`ff/FCSA00079270.ff.fits`" with `ds9` to understand how such a distortion pattern appears. Here, you maybe need to adjust the intensity scale of the opened image. From the top menu, select "Zoom" → "Pan Zoom Rotate

---

[5]You may have realized that the count values gradually increase from right to left on the image. This pattern is caused by the intrinsic color, red, of the lamp whose intensity becomes bright toward longer wavelength. A method to correct such a pattern is described in Appendix B.

Parameters...". Change `Zoom` to `2 0.05`, then press the `Apply` button to see the "curved" spectrum.

Using `distcalib` in `focasred`, we correct for the curve.

```
ana03{hattriak}: sed 's/bs/dc/g' list.obj.bs > list.obj.dc
ana03{hattriak}: mkdir dc
fo> distcalib @list.obj.ff @list.obj.dc obsmode=spec
```

Specify `obsmode = spectroscopy` for our case (if you are reducing imaging data, select `obsmode = imaging`). Verify the distortion-corrected images named `dc/*.dc.fits` using `ds9`.

5.5 Wavelength Calibration

To perform wavelength calibration, we will use the night sky lines for the target (SDSS J00347.01), and the ThAr lamp light for the standard star.

5.5.1 Wavelength Calibration of the Target with Night Sky Lines

Although we have dealt with the whole image so far, at this point, it is recommended that the image is trimmed. The most important reason is that CCD frames usually have a number of pixels that are not exposed to light. After flat-fielding, these pixels often have noisy values which can cause discontinuities, which in turn result in calculations that diverge. It is, therefore, prudent to trim these unexposed pixels to avoide this problem.

Check the images "`dc/FCSA0007927[024].dc.fits`" with `ds9` to determine which region should be extracted. Taking the sample data as an example, we suggest extracting the central portion of the image between pixel numbers 201 and 500 along the x-axis, and between 41 and 2620 along the y-axis.

```
fo> imcopy dc/FCSA00079270.dc.fits[201:500,41:2620] obj1
fo> imcopy dc/FCSA00079272.dc.fits[201:500,41:2620] obj2
fo> imcopy dc/FCSA00079274.dc.fits[201:500,41:2620] obj3
```

Here, we repeatedly used the task three times. Alternatively you can supply a list of files as input and output.
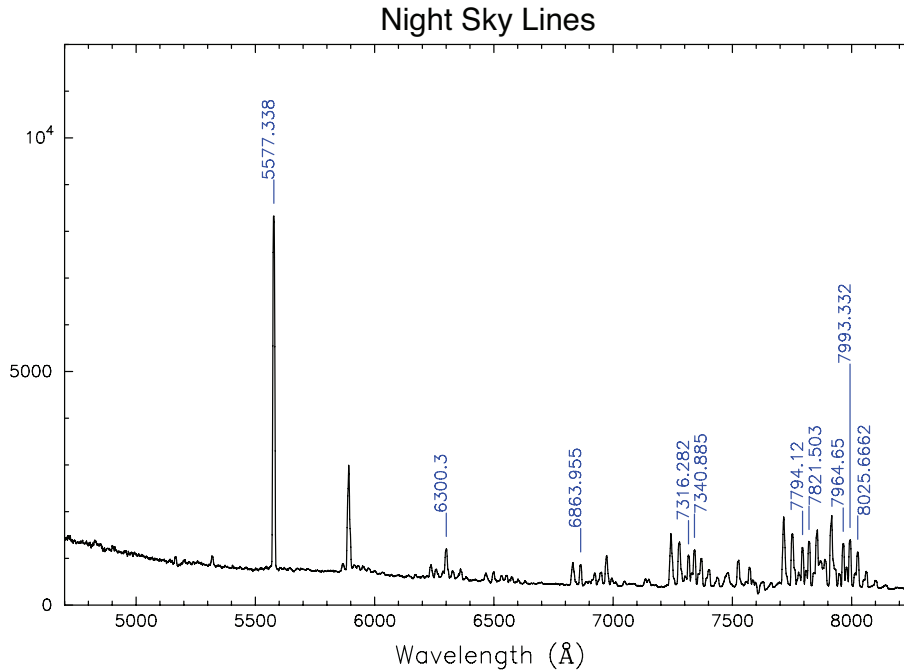
19

Figure 5: Identification of Night Sky Lines

In order to improve the S/N of the night sky lines, we combine the 3 images[6]. Next, we run the "identify" IRAF task to identify the lines

```
fo> imcombine obj?.fits sky combine=median
fo> identify sky section="middle column"
   coordli="FOCASREDHOME$/IRAF/skyline_midres.dat" fwidth=8 order=3
   function=chebyshev
```

---

[6]For many cases, this procedure is not required. The reason why we combine the 3 images is that each image does not have adequate integration time (integration times are 10 min). If exposure time were 20 – 30 min, these would have been adequate S/N ratio without combining. We strongly suggest not combining images that have been taken at different observing times (namely, different elevation angles) because observing at different telescope positions does not guarantee that the target will always falls on the same detector pixels.

Table 2: Cursor Commands for task `identify`

Key	Operations
m	Search for a spectral feature around cursor position
d	Delete the closest spectral feature to cursor position
f	Entering a fitting window with a pixel vs. wavelength plane
X	Magnify plot horizontally centered on cursor position
Y	Magnify plot vertically centered on cursor position
Z	Magnify plot both vertically and horizontally centered on cursor position
>	Set the cursor position as the maximum of y-coordinate
<	Set the cursor position as the minimum of y-coordinate
r	Redraw (canceling all the magnifications set in the before)
c	Display x, y-coordinates of the nearest feature from the cursor position
q	Quit identification
C	Display x, y-coordinates of the current cursor position
?	Showing help menu
	Operations in Fitting Window
f	Retry fitting
d	Deleting the feature
c	Display coordinate of the nearest feature from the cursor position
q	Quit from fitting window
?	Showing help menu
:order 4	Changing fitting order to e.g., 4
:func chebyshev	Changing fitting function

`section`	:	A segment to identify the lines
`coordli`	:	Wavelengths of night sky lines are taken from the file given here.
`fwidth`	:	Initial guess of the line width in pixel unit
`order`	:	Initial value of the order of the fitting function
		For the sample data, we intentionally selected a low number
		to see how it works.
`function`	:	Name of fitting function

Notice that the resultant accuracy to detect line positions will be significantly degraded if an inappropriate `fwidth` value is used. Although we have written all the options for the task on the command line as shown above, you can edit them with `epar` as well.

If you run `identify`, you will get a plot that appears to have inverted Figure 5 horizontally. Bring the mouse to the line whose wavelength is well-known, then press `m` (mark) to select it. The task will ask you the wavelength, give the number. For your knowledge, you can provide just the integer part of the wavelength because accurate wavelength values are already listed in the file specified by the `coordli` parameter.

Completely identify all the lines as shown in Figure 5, and fit them accordingly. To fit, press "f". We suggest identifying the intense lines first, then identifying the weak lines. Once you have identified and fit a couple of strong lines, the program will readily find the most plausible candidate line wavelength for the weak emission that you are trying to identify because the results from the latest fittings are being used. In the above example, we have specified the order of the fitting function to be 3 by `order = 3`. However, if you attempt to fit a wide range of the wavelength axis, the low fitting order being used would be inadequate. This can be checked by inspecting whether or not the residual has a systematic pattern. If this is the case, try a high order by typing e.g., ":`order 4`". It is important to experiment with different fitting order to understand what order gives the best-fit spectra both globally and locally. If you are succeed in fitting, you will have an RMS of ~ 0.2 Å. All the fitting results are stored in `database/idsky`.

If you want detailed information about night sky lines, please see the FOCAS web page, and/or Osterbrock et al. (1996[3]; 1997[4]).

Since we have identified night sky lines only at the image center ($x = 150$), we should identify the lines at the other x-coordinates as well. This will be done with `reidentify`.

```
fo> reidentify sky sky nlost=1 ver+ section="middle column"


REIDENTIFY: NOAO/IRAF V2.12.2a-EXPORT hattriak@ana03 Tue 02:02:47 30-Sep-2008
  Reference image = sky, New image = sky, Refit = yes
        Image Data    Found    Fit Pix Shift  User Shift  Z Shift     RMS
sky[140,*]             10/10  10/10    0.249     -0.353   -4.8E-5    0.155
sky[130,*]             10/10  10/10    0.409     -0.576   -7.9E-5    0.155
sky[120,*]             10/10  10/10    0.361     -0.508   -6.9E-5    0.155
sky[110,*]             10/10  10/10    0.389     -0.548   -7.5E-5    0.186
        .
        .
        .
```

Above shows that the task is sequentially identifying night sky lines at $x = 140, 130, 120, 110, \ldots$ on the basis of the results at $x = 150$. The `ver+` option causes the task to display the fitting results at each line in the terminal, as the task progresses. You should keep eye on whether or not the task works completely till the end, and all the RMS values are comparable to each other.

Table 3: Cursor Commands for task `fitcoords`

Key	Operations
x,y	Changing x and y coordinates that will be plotted
	x : x-coordinate of the sky, y : y-coordinate of the sky
	z : Wavelength of the identified feature
	s : the best-fit z-value given by $z = f(x, y)$
	r : Residual, i.e., $s - z$
	Example 1: Press x twice, horizontal axis becomes x-coordinate of sky
	Example 2: Press y followed by z, vertical axis becomes wavelength
d	Press d followed by p, the nearest data to cursor position will be deleted
	(not to be used in fitting)
r	Redrawing
f	Fitting and re-fitting
q	Quit from `fitcoords`
:xorder 4	Changing the fitting order along x-axis to e.g., 4
:yorder 4	Changing the fitting order along y-axis to e.g., 4
:func chebyshev	Changing function name to be used in fitting

The next step is to perform a 2D fit of the wavelength as a function of x and y using the coordinates obtained from `identify/reidentify`.

        fo> fitcoords sky func=chebyshev

This will ask you "`Fit sky interactively (yes):`". Reply `yes` (or just type the return key).

First, draw the plot by setting the horizontal direction of the image to x-axis and the vertical direction to y-axis; this can be done by typing "`xxyy`". Re-drawing the plot using "`r`" will result in a map of identified night sky lines in xy-coordinates of `sky.fits`. Verify whether or not the identified lines cover the entire x-axis ($x \sim 0 - 300$) of `sky.fits`. Second, try to reduce the `xorder` value by looking at a plot whose vertical axis is the residual; you can obtain such a window by typing "`yr`". Check the fitting quality along the x-axis. Don't forget to re-draw your plot every time you change any parameters. Moreover, verify the updated plot carefully. Third, change the horizontal axis to the y-coordinate by typing "`xy`". Again, inspect the results in the same fashion as before. We suggest setting the fitting order for `yorder` identically to `order` in identify. If you find data points that are obviously far from

23

the best-fit curve, try re-fitting after removing these bad points by pressing "d". To quit the fitting typing "q", and answer "yes" to save all the results.
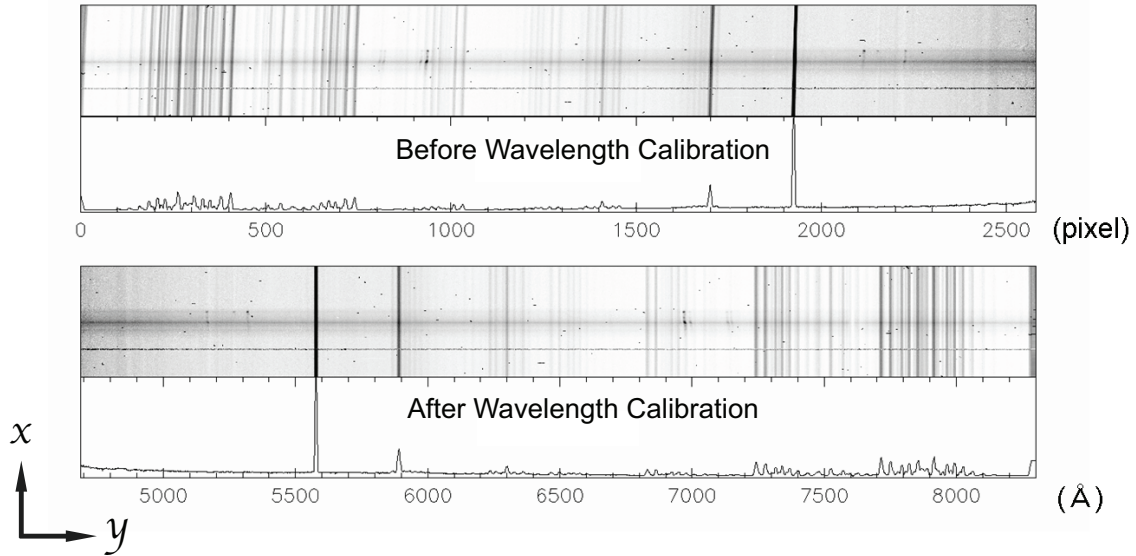

Figure 6: The object image before (the upper panel) and after (the lower) wavelength calibration

Using the results from `fitcoords`, we transform the target image to linear wavelength scales,

```
fo> transform obj1 obj1.wc sky
fo> transform obj2 obj2.wc sky
fo> transform obj3 obj3.wc sky

Transform obj1 to obj1.wc.
Conserve flux per pixel.
User coordinate transformations:
    sky
Interpolation is linear.
Using edge extension for out of bounds pixel values.
Output coordinate parameters are:
    x1 =           1., x2 =       300., dx =           1., nx =  300, xlog = no
    y1 =        4687., y2 =       8301., dy =       1.401, ny = 2580, ylog = no
```

We hope that you have successfully finished the wavelength calibration step.

Figure 6 compares the target images before and after the wavelength calibration where the inclined sky emission in the upper panel became the vertical in the lower panel. The output of the `transform` task describes the relationship between the y-coordinate and wavelength by the following equation[7]:

$$\lambda \ = \ (y - 1) \ \times \ 1.401 \ + \ 4687.$$

5.5.2   Wavelength Calibration of the Standard Star with Lamp

The physics behind the wavelength calibration for the (bright) standard star is essentially the same as (faint) targets. Since the standard star is generally very bright, one can readily trace (then extract) its emission, even if the spectrum is slightly curved. Because of this characteristic, we introduce another approach in this subsection. This method, of course, can be applied not only to standard stars but also any type of bright objects.

We can utilize the standard star's frame that has been flat-fielded (i.e., `ff/FCSA00079184.ff.fits`).

```
fo> imcopy ff/FCSA00079184.ff[501:650,6:2650] std
fo> imcopy ff/FCSA00079568.ff[501:650,6:2650] arc
```

To extract spectrum of the standard star, we use task `apall`; check all the parameters with `epar` as the task has many parameters.

```
fo> epar apall
```

Set "`yes`" for the parameters listed between `interac` and `review`, but set `no` for `resize` parameter. Set `backgro = fit` at EXTRACTION PARAMETERS. Type "`:q`" to quit editing the parameters.

```
fo> apall std format=oned nfind=1
Find apertures for std? (yes):
Edit apertures for std? (yes):
```

As long as you are getting the above 3 lines, just continue to hit return key. If not, correct the field in the parenthesis to the desired values.

Once you obtained the plot as shown in Figure 7a which is a cross-section along the slit, you have successfully started extracting a spectrum by detecting the standard star automatically.

---

[7]The obtained coefficients have more accuracy than they appear; the precise values of the best-fit coefficients are stored as CRVAL2 and CDELT2 in the FITS header.
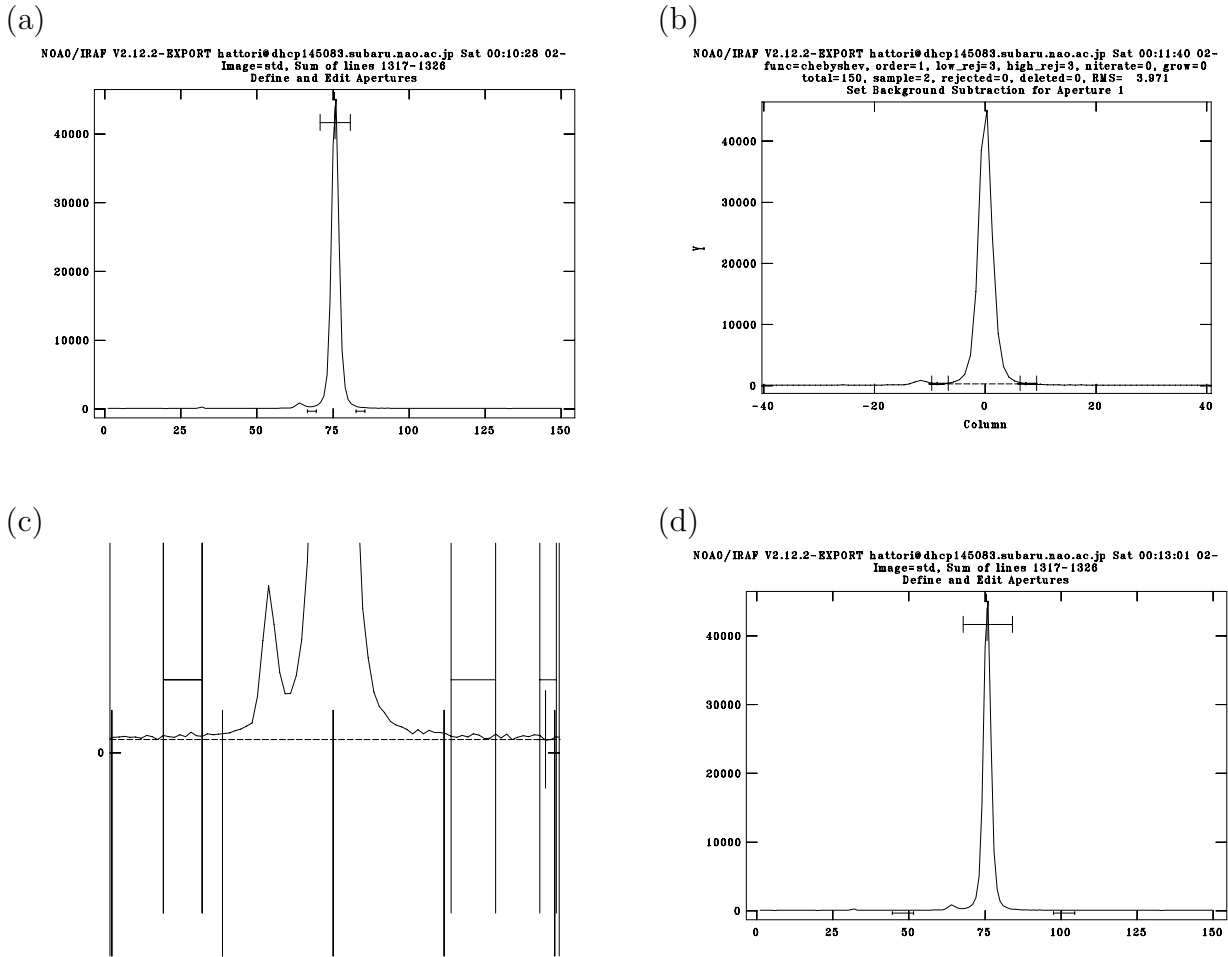
Figure 7: `apall` windows, see the text for details

All you have to do manually is to optimize (i) the aperture size used to extract the spectrum, and (ii) the emission-free region where the background level will be calculated for subtraction. You may notice that there exists an obvious peak very close to the left of the standard star (see the emission around $x = 65$), which is a nearby star. Let us select an aperture as large as possible without including the star. As you did with `identify`, try to zoom in on the image to get the best view when specifying an aperture. Once you have determined an aperture size, bring the cursor to the left edge, then type "l" (i.e., the lower case of L). Similarly, bring cursor to the right, then type "u". Here, the characters "l" (lower) and "u" (upper), which could be misleading to some readers, correspond to the left and right along the wavelength axis. After selecting an aperture, type "b" to optimize the background level (see Figure 7b). In Figure 7(b), you may have realized that the selected regions for defining the background

level are not free from emission of a nearby star. Obviously, this is bad. So, we should initialize the specified ranges by typing "t" before re-setting. After initializing the window, type "Y" a few times until the low level emission around $y = 0$ becomes visible (see Figure 7c). Specify an emission-free region to the left of the standard star by typing "s" twice, and repeat to set another region to the right. After finishing, type "f" to fit the data once more, then "q" to return to the extraction page.

Once you produced a plot like Figure 7d, quit with "q", and answer yes to all the questions for extracting a spectrum. As the extraction progresses, results of the spectral tracing as well as the polynomial function curve fitted to the data will appear in the window. Verify the results carefully, and optimize the fitting order by ":order ", if necessary. Subsequently, extract a comparison spectrum for the standard star as follows:

```
fo> apall arc format=oned reference=std find- recen- resize- edit-
        trace- backgro-
```

The results will be written into a FITS file, std.0001.fits.

Since we want to extract a spectrum in an identical fashion to the standard star, set reference = std, and leave all the options unchanged. The results will be written into arc.0001.fits.
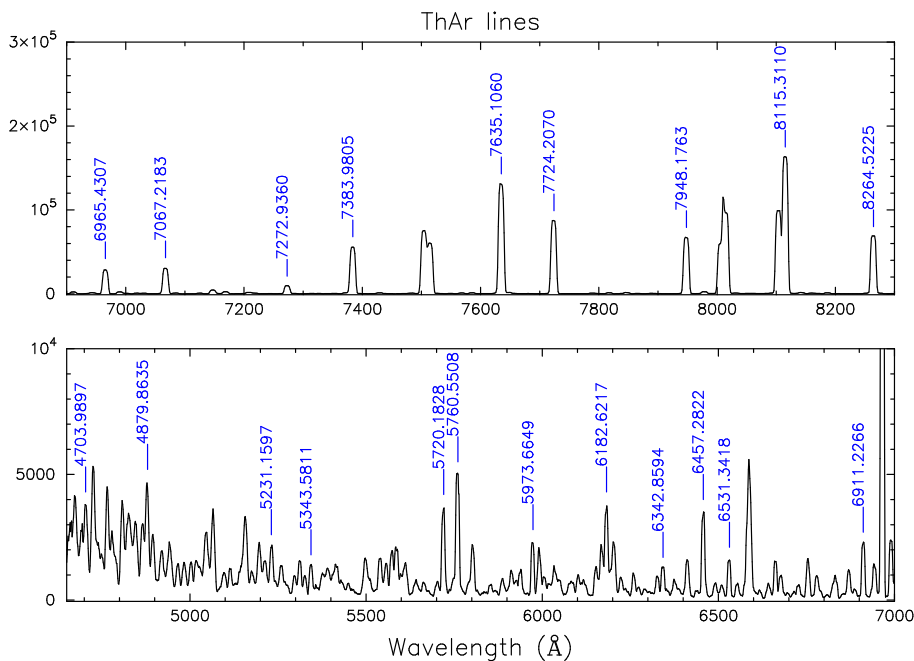

Figure 8: Line identification of the ThAr lamp emission

27

```
fo> identify arc.0001 coordli="FOCASREDHOME$/IRAF/thar.300b_y47.dat"
        fwidth=8 order=3 function=chebyshev
```

The usage of `identify` is the same as in the previous subsection. Fit lines by referring to Figure 8. Notice that the intensity level of the lines increase significantly for wavelength greater than $\lambda \sim 6900$Å in this figure.

To calibrate wavelength (i.e., to associate the pixel coordinate with the absolute wavelength), we use an IRAF task `dispcor` implemented in the `noao.onedspec` package. We calibrate the wavelength through a comparison with a lamp spectrum, which can be done as follows: after specifying the reference lamp spectrum with an IRAF task `refspec` (the task will append REFSPEC1='`arc.0001`' where `arc.0001` is name of the reference spectrum). Calibrate the wavelength using the `dispcor` task. The extracted spectrum will be stored in `std.0001[*,1,1]`, and the sky-spectrum in `std.0001[*,1,2]`. The former is the desirable one that we are going to calibrate.

```
fo> refspec std.0001 reference=arc.0001 select=match sort="" group=""
fo> dispcor std.0001[*,1,1] std.wc
```

Check the wavelength-calibrated spectrum by displaying it with `splot`. The usage of the task is described in §6.2.

```
fo> splot std.wc
```

5.6 Sky-Subtraction

Using the IRAF task `background`, we subtract the sky background emission from the spectrum of the target, SDSS J000347.

```
fo> background obj1.wc obj1.bg axis=1 order=2 low_rej=3 high_rej=3 nit=3
```

obj1.wc	:	Input image
obj1.bg	:	Output image
axis	:	Specify the slit direction, which must be 1 for FOCAS (i.e., the x-direction)
		(All the options below can be changed during the fitting process)
order	:	Order of fitting function plus 1 (i.e., numbers of parameters)
low_rej	:	If counts are smaller than `fit-low_rej`$\times\sigma$, fitting without the point(s)
high_rej	:	If counts are larger than `fit+high_rej`$\times\sigma$, fitting without the point(s)
niterate	:	Numbers of iterations for the data rejection specified above

The `background` task inquiries for "`Fit line`": enter the y-coordinates which define the fitting area. In the case of the sample data, give the coordinates which correspond to the central region along the y-axis which has 2600 pixels, by typing e.g., "`1000 1500`" or "`500 2000`".

The plot that pops up is a cross-section along the slit of data averaged over the y-range specified above (Figure 9).

Specify the sky-region (i.e., source-emission free positions) by using the "`s`" option in a similar manner to that used to define the background region for the `apall` task. After setting the regions, fit the lines with "`f`"; quit with "`q`"; and hit the return key if you are asked for a line number once again.

Keep in mind that quality of the final spectrum is very SENSITIVE to this sky-subtraction process. Check the resultant images using `ds9` with a great caution. We strongly encourage the readers use "trial and error" with various combinations of the parameters, such as the fitting order and the fitting range(s). If you want to re-try fitting some portions of the y-axis, try the following:

```
fo> background obj1.bg[*,1000:1100] obj1.bg[*,1000:1100] axis=1 order=2
        low_rej=3 high_rej=3 nit=3
```

where we re-fit in the range between $y = 1000$ and 1100 as well as all the range in x. After completing the above, finish the sky background subtraction procedure for the obj 2 and obj 3 in a similar manner.

```
NOAO/IRAF V2.14EXPORT hattori@athelas.subaru.nao.ac.jp Tue 21:37:23 28-Oct
func=chebyshev, order=2, low_rej=3, high_rej=3, niterate=3, grow=0
total=300, sample=95, rejected=4, deleted=0, RMS= 0.4039
obj1.wc.fits: Fit line = 500 - 2000
SDSSJ000347
```
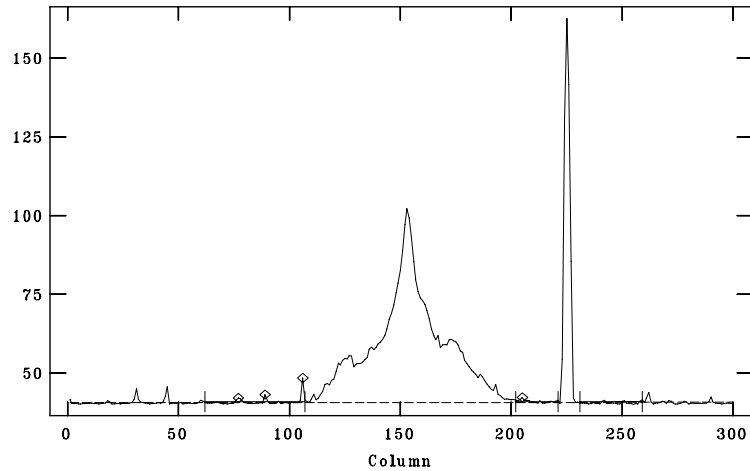
Figure 9: A window showing a spectrum where we defined the column ranges to calculate source-emission free regions for subtracting the sky background emission. In the above example, column ranges between 60 and 110, 205 and 220, and 230 and 260 are selected.

5.7   Flux calibration

In this subsection, we describe a method for flux calibration — finding a correspondence between the count value and the (absolute) flux density scale at each bin of the wavelength axis. Such conversion factors at the individual spectral bins should be estimated using the wavelength-calibrated data. Configure `extinct` and `caldir` in `standard` as follows, then execute the task by ":go".

```
fo> epar standard

PACKAGE = onedspec
   TASK = standard

input   =                 std.wc  Input image file root name
output  =                 std.dat  Output flux file (used by SENSFUNC)
(samesta=                    yes) Same star in all apertures?
(beam_sw=                     no) Beam switch spectra?
(apertur=                       ) Aperture selection list
```

30

```
(bandwid=                    INDEF) Bandpass widths
(bandsep=                    INDEF) Bandpass separation
(fnuzero= 3.6800000000000E-20) Absolute flux zero point
(extinct= FOCASREDHOME$/IRAF/mkoextinct.dat) Extinction file
(caldir = onedstds$spec50cal/) Directory containing calibration data
(observa=         )_.observatory) Observatory for data
(interac=                     yes) Graphic interaction to define new bandpasses
(graphic=              stdgraph) Graphics output device
(cursor =                     ) Graphics cursor input
star_nam=                      Star name in calibration list
airmass =                      Airmass
exptime =                      Exposure time (seconds)
```

Although you will be asked to supply the name of a star, ignore the inquiry and just hit the return key. Then, a list of stars, which is stored in the directory specified by `caldir`, appears in the terminal. Select `bd284211`, and answer `yes` to the inquiry of "`Edit bandpasses?`".

```
std.wc(1): BD+28D4211
Star name in calibration list:
Standard stars in onedstds$spec50cal/  (3200A - 8100 A)


bd284211   eg247      feige34    hd192281   pg0205134  pg0934554  wolf1346
cygob2no9  eg42       feige66    hd217086   pg0216032  pg0939262
eg139      eg71       feige67    hilt600    pg0310149  pg1121145
eg158      eg81       g191b2b    hz14       pg0823546  pg1545035
eg20       feige110   gd140      hz44       pg0846249  pg1708602


Standard stars in onedstds$spec50cal/  (3200A - 10200A)


bd284211   eg247      feige34    g191b2b    hz44
eg139      eg71       feige66    gd140      pg0823546
eg158      feige110   feige67    hilt600    wolf1346
Star name in calibration list: bd284211
std.wc[1]: Edit bandpasses? (no|yes|NO|YES|NO!|YES!) (no): yes
```

At this stage, you should have a window similar to Figure 10 where the data points from the observations are shown by the thick curve. The rectangular boxes on the spectrum indicate the
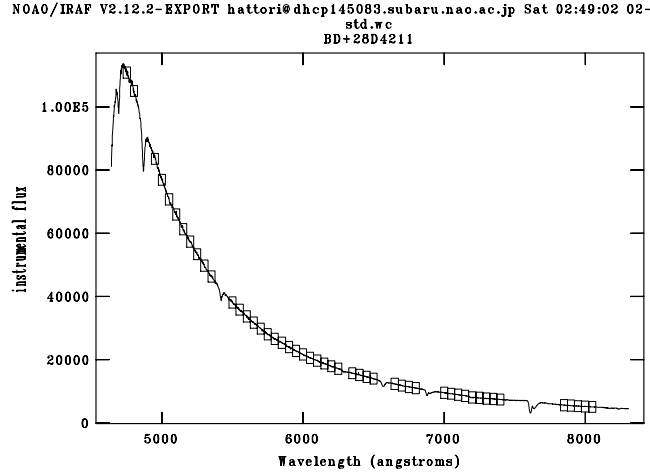
Figure 10: The results of the IRAF task `standard`. The vertical axis is count unit.

wavelengths where flux values are available in the library. Here, the size of each box indicates the bandwidth where a catalog value has been measured. In the case of the sample data, we should eliminate the data points around $\lambda = 6900\,\text{Å}$ and 7600Å where absorption features due to the stellar atmosphere exist. In addition to stellar atmospheric absorption features, i.e., Hα 6563Å, and Hβ 4861Å, absorption features due to the earth's atmosphere also should be eliminated. Bring cursor on the point, and press "`d`" to delete them (see Figure 10).

The task `standard` integrates the emission (i.e., the counts) over the given wavelength range, and writes the results into an ASCII file, `std.dat`. Using the results, we subsequently fit the data with the task `sensfunc` to find an appropriate function that will be utilized when converting the target spectrum into the flux density scale.

```
fo> sensfunc std.dat sens extinct="FOCASREDHOME$/IRAF/mkoextinct.dat"
```

Answer `yes` to the inquiry "`Fit aperture 1 interactively?`". You should obtain a plot like Figure 11.

Verify smoothness of the obtained curve which should not be bumpy. The resultant spectrum, shown in the upper panel of Figure 11, should have a smooth and gradual spectral profile. Since we divided the standard star's spectrum by the dome flat spectrum which has a smooth and gradual spectral profile over the wavelength range, the resultant spectrum also should have a smooth and gradual profile. Otherwise, it is very likely that an error has occurred at some point in the data reduction process. One can recognize a distinct peak around $\lambda \sim 7200 - 7300$ Å in Figure 11. However, we believe that this is an artifact attributable to
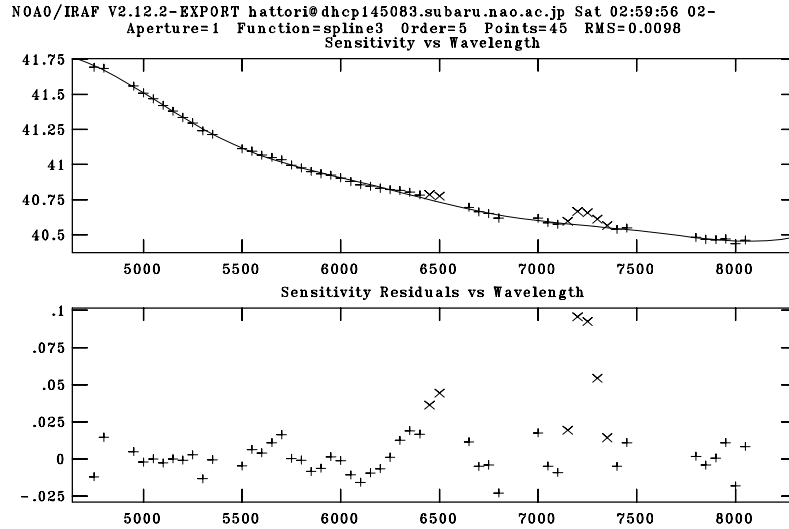
32

Figure 11: The results of the IRAF task `sensfunc`

an existing in the library, rather than an error in the data reduction. Use "d" to delete the unnecessary points and "f" to re-fit.

Using the above results (`sens.fits`), we convert unit of the object frames into the flux density scale (the first line shown below). In addition, we should correct for the extinction by the terrestrial atmosphere measured at the summit of Mauna Kea (the second line shown below).

```
fo> fluxcalib obj1.bg obj1.fc sens exposur="EXPTIME"
fo> extinct obj1.fc obj1.ex extinct="FOCASREDHOME$/IRAF/mkoextinct.dat"
```

6 Going Further: *Additional Recipes*

6.1 Combining Images, Corrections for Extinction and Helio Centric Velocity

6.1.1 Combining Images

It is tempting to combine all the frames, if you have multiple ones. In fact, the sample data have three sets of frames. Keep in mind that there are several methods to combine (i.e., registering each image to a common reference image) multiple images. Below, we show a straightforward method in the sense that we utilize the peak position of the continuum emission — a clear signpost of the central region of the galaxy since it has a simple structure and is less sensitive to the seeing variations.

```
fo> implot obj1.ex
:l 1300 1400
```

This gives a spatial, cross-sectioned, plot of the continuum emission around $6500\,\text{Å}$. Magnify the plot by pressing "X", and measure the x-coordinate value of the peak by pressing "p" once on the left- and right side of the peak (see Figure 12).

You probably will measure the representative position offset for "obj2.ex" and "obj3.ex" with respect to "obj1.ex" to be $\pm\,9.7\,$pixel. Since a single pixel size is $0\rlap{.}''1038$ for FOCAS, the position offset corresponds to $0\rlap{.}''1038\,\text{pixel}^{-1} \times 3^{\text{pixel binning}} \times 9.7^{\text{pixel}} \sim 3''$, which is reasonable compared to the expected position shift for a $3\rlap{.}''0$ dithering observation. After shifting the images (correcting for the positional offset to the reference image) with `imshift`, we combine the three images using `imcombine`.

```
fo> imshift obj2.ex obj2.sh -9.7 0
fo> imshift obj3.ex obj3.sh 9.7 0
fo> imcombine obj1.ex,obj2.sh,obj3.sh obj.comb combine = median
```

It should be noted that we combined the images with the option `combine = median` to eliminate bad pixel counts (i.e., extraordinary large pixel counts) due to incident cosmic rays.

6.1.2 Correction for Interstellar Extinction

We can estimate how much the observed emission has been absorbed by the interstellar medium in our galaxy from the total amount of gas-plus-dust existing in a given column pointing toward the object. This information, obtained from extensive observations of the HI 21 cm
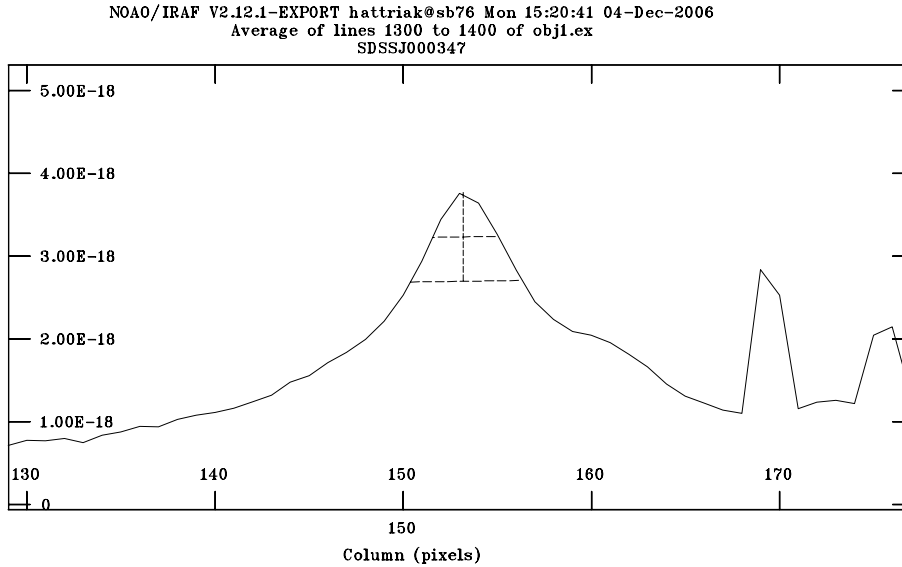
34

Figure 12: An example of measurements of the center position of the emission (galaxy). The above window can be obtained by pressing "p" at the left end of the bottom dashed line, followed by pressing "p" at the right end of the bottom dashed line.

lines and FIR/submm continuum emission, is complied at e.g., the NED (NASA/IPAC Extragalactic Database, `http://nedwww.ipac.caltech.edu`). The NED database tells us that the interstellar extinction toward SDSS J000347.01-000350.3 is estimated to be $A_V = 0.106$. Let us correct for this extinction using the IRAF task `deredden`,

```
fo> deredden obj.comb obj.comb.dred 0.106 type="A(V)"
```

Here, the task implements an empirical relationship of the extinction curve reported in Cardelli et al. (1989)[1].

6.1.3 Correction for Helio Centric Velocity

Those who wish to attain a very high accuracy for the wavelength calibration may try to correct for the Doppler effect of the Earth's rotation. Use the task **rvcorrect** implemented in the **astutil** package to estimate the possible shift of the radial velocity due to the Earth's rotation. This task calculates a velocity shift based on the date and time of the observations (in UT), and the observatory coordinates given in the FITS header.

```
fo> astutil
```

```
as> rvcorrect observa=subaru vobs=0.
# RVCORRECT: Observatory parameters for Subaru Telescope, NAOJ
#        latitude = 19:49.5302375
#        longitude = 155:28.5611198
#        altitude = 4163
Year of observation: 2006
Month of observation (1-12) (1:12): 10
Day of observation (1:31): 14
UT of observation (hours) (0.:24.): 7.6
Right ascension of observation (hours) (0.:24.): 0.063
Declination of observation (degrees) (-90.:90.): -0.064
##   HJD           VOBS    VHELIO     VLSR   VDIURNAL   VLUNAR  VANNUAL   VSOLAR
2454022.82208     0.00   -10.27    -10.26      0.381    0.012  -10.667    0.013
```

The above example suggests shifting the spectrum blue-ward by 10.27 km s^{-1} for the observed velocity (V_{obs}) of 0.0 km s^{-1}. This is because the radial velocity seen from the Sun is calculated to be -10.27 km s^{-1}. The correction can be done using the dopcor task as follows:

```
as> dopcor obj.comb.dred obj.fin 10.27 isveloc+ add- disp+ flux-
```
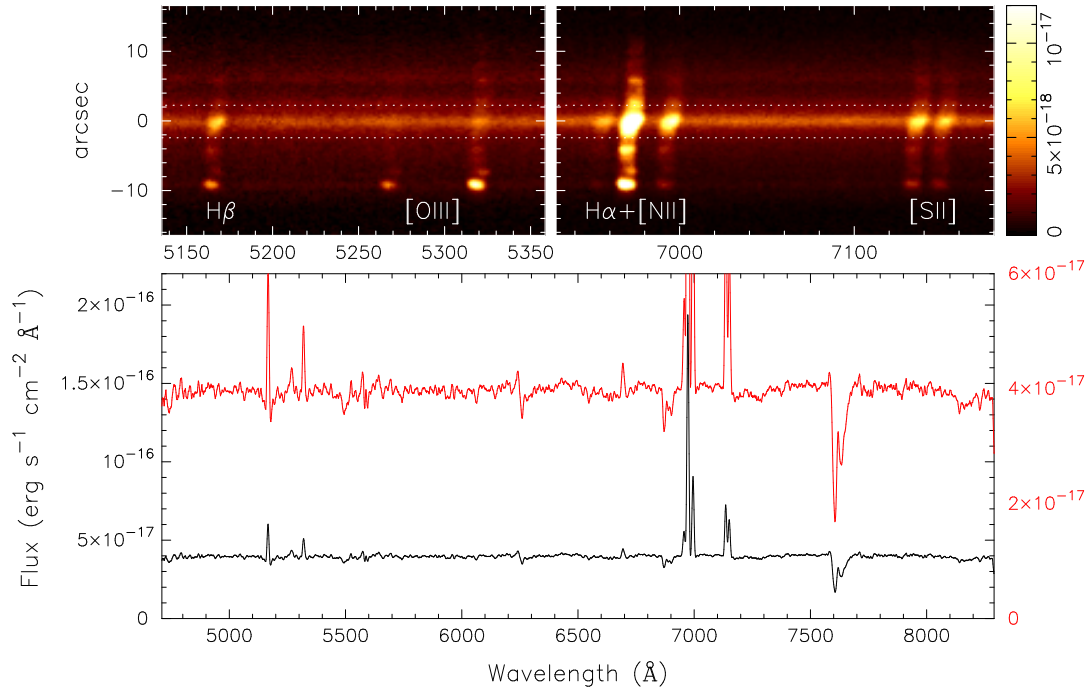
Figure 13: (Upper) — The fully reduced 2D spectra taken with FOCAS. The upper left- and right-hand panels show the magnifications around the strong emission lines. The horizontal axis is the wavelength unit in Å. Notice that the slightly curved appearances of these spectra represent the velocity field of the galaxy. (Lower) — The spectrum extracted (i.e., one-dimensionalized) from the central $\sim 4''.5$ portion of the galaxy, which are the regions encompassed by the two horizontal dashed lines in the upper panels. To show the weak spectral features seen in the low level, the magnified spectrum is shown in red: use the red-coded vertical axis on the right-hand side for the red spectrum.

6.2 Measurements of Redshift

We have obtained the desired fully calibrated spectrum of the galaxy as a result of the previous subsection. Now, we should be ready for the fun part — scientific analysis. Let's measure redshift of the galaxy using the Hα line as a probe. We use task `splot`:

```
fo> splot obj.fin
Image line/aperture to plot (0:) (1): 153
:nsum 15
```

Once you run the task, you will be asked to give a column number to display. Let's take the very center of the galaxy ($x = 153$). After getting the spectrum, it would be prudent of us to integrate the emission over 15 columns centered on $x = 153$ to improve the S/N.

Did you find the Hα line around 7000 Å? Magnify the plot to see the line better by pressing "`a`" on the left-end that you want to display and another "`a`" on the right. After getting a better view, specify the velocity range where a Gaussian function is fitted to the emission; press "`k`" at the bottom-left, and another "`k`" at the bottom-right. Using the obtained center wavelength, calculate its redshift with

$$z = (\lambda - 6562.8)/6562.8$$

where 6562.8 Å is the wavelength of Hα emission measured at the rest-frame.

Cross-check your results with the already measured one listed in e.g., SDSS (see Figure 14). Are you happy with the consistency? You may see a slight difference within 1 pixel, corresponding to ≈ 1.4Å, namely, ≈ 60 km s^{-1} between your measurement and the one in the catalog. This can be attributed to the difference in e.g., aperture sizes, slit-effect, and the overall accuracy of the calibrations. If the difference is significantly larger than ~ 1 pixel, we strongly suggest diagnosing the possible causes for this large difference. Last, `splot` has a lot of functions (see Table 4) that are beyond our description here. See the help page to explore its capabilities.
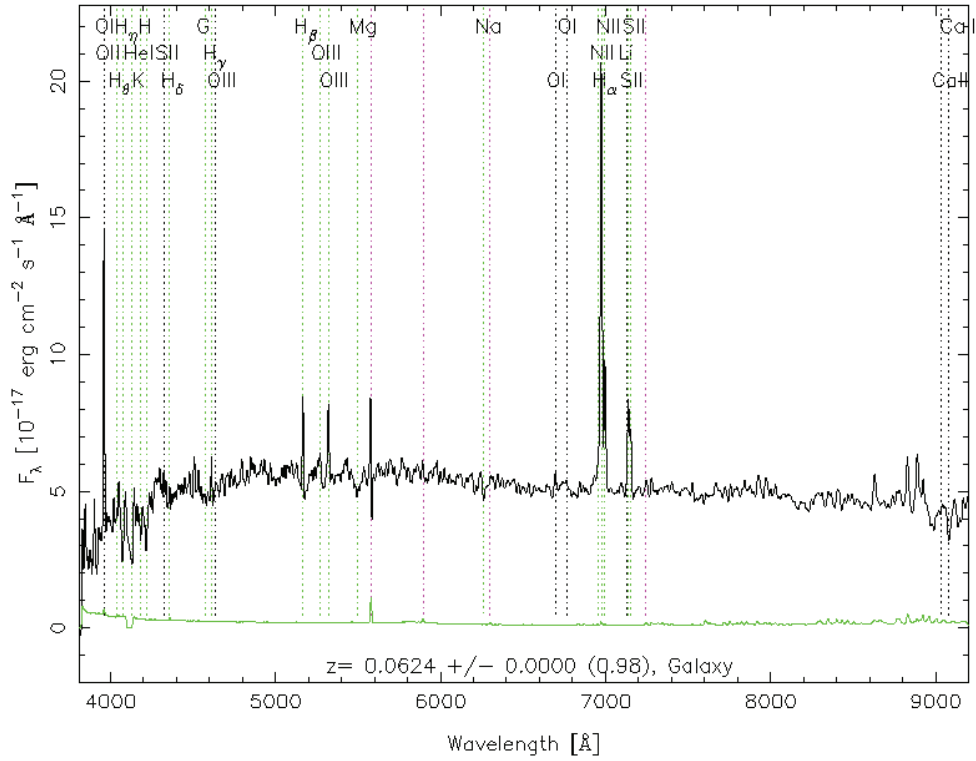
Figure 14: SDSS spectrum (black) of the spiral galaxy SDSS J000347.01-000350.3; the green represents the error spectrum. Wavelengths of the atomic lines are shown by the dashed vertical lines by shifting the spectrum to the galaxy's frame using the derived redshift in §6.2. Clearly, the obtained redshift gives fairly reasonable results. Compare with the FOCAS spectrum shown in Figure 13.

Table 4: Cursor Commands for task `splot`

Key	Operations
a	Auto-expand horizontal scale of the plot between the two locations specified by cursor position. The vertical scale is automatically adjusted. To initialize, either press "c" once or press "a" twice without any moving the cursor.
b	Set the base level of the current plot to 0.0.
c	Initialize the display ranges.
d	Enter "de-blending" mode, allowing simultaneous fitting to more than two lines.
e	Calculate an equivalent width over the region between cursors, press "e" twice.
f	Arithmetic operations to the y-coordinate value, e.g., log, sqrt...
g	Get new image and plot
k	Profile fit to single line. Press "k" by moving cursor on the line followed by either "l" or "v" where the second parameter selects a fitting function, i.e., Lorentzian or Voigt profiles, respectively. Otherwise, Gaussian is used.
l	Convert the vertical scale from energy per frequency to energy per wavelength.
m	Calculate the mean and RMS between the two cursors (i.e., press "m" twice).
n	The opposite of "l".
o	Toggle to overlay the next plot on the current one.
q	Quit from splot
r	Redraw
s	Smooth the spectrum currently displayed using the boxcar window function.
u	Convert the horizontal scale, e.g., after pressing "u", move cursor onto the Hα line, then press "d" and enter 6562.8, the wavelength measured at the rest-frame will be given.
t	Enter continuum fitting mode. There are a few further options that e.g., subtract the fitted continuum level, and normalize with the fitting results etc.
z	Expand (magnify) the data horizontally centered on the cursor position. The vertical scale will be adjusted automatically. To clear the magnification, press "c" or type "a" twice while keeping the cursor at the current position.
)	Go to the next spectrum in the image (e.g., going to the spectrum at $x = 154$ from that at $x = 153$ in the case of the text).
(	The opposite of ")" above.
#	The column "x" of the spectrum to display, give a number.
-	Subtract the fitting results obtained from "d".
,	Down slide spectrum (shifting the direction to shorter wavelengths).
.	Up slide spectrum (the same as the above, but to longer wavelengths)
?	Show help menu
:nsum 3	Set how many pixels are integrated; this option is 2D spectral image only. This example integrates 3 pixels.

6.3 Extracting a Spectrum and Writing into Textfile

To pursue scientific analysis using your favorite software, you may want to extract a spectrum (i.e., one-dimensionalize the image) into an ASCII text file. Below shows an example of how to extract a spectrum over a 5 pixel-width, corresponding to $1''.5$, centered on $x = 153$.

```
fo> blkavg obj.fin[151:155,*] obj.nucleus 5 1 option=sum
fo> imcopy obj.nucleus[1,*] obj.nucleus
```

To write an ASCII text file, use task wspectext as follows,

```
fo> wspectext obj.nucleus.fits obj.nucleus.dat header-
```

With the above command, the spectrum is written into a text file, obj.nucleus.dat, whose first and second columns, respectively, represent the wavelength and the intensity of the data.

References

[1] Cardelli et al. (1989), ApJ, **345**, 245

[2] Kashikawa et al. (2002), PASJ, **54**, 819

[3] Osterbrock et al. (1996), PASP, **108**,277

[4] Osterbrock et al. (1997), PASP, **109**,614

A FITS Header Key Words for Grisms, Filters, and Slits Equipped in FOCAS

ID	Name of Grism
SCFCGREL01	75/mm
SCFCGRLD01	150/mm
SCFCGRMB01	300B
SCFCGRMR01	300R
SCFCGRHDEC	Echelle
SCFCGRHD45	VPH450
SCFCGRHD52	VPH520
SCFCGRHD65	VPH650
SCFCGRHD68	VPH680
SCFCGRHD80	VPH800
SCFCGRHD95	VPH950

Table 5: Correspondence between grisms and their identification number in the FITS header. See the FOCAS web page for details.

ID	Slit
SCFCSLLC02	$0\overset{''}{.}2$ center
SCFCSLLC04	$0\overset{''}{.}4$ center
SCFCSLLC06	$0\overset{''}{.}6$ center
SCFCSLLC08	$0\overset{''}{.}8$ center
SCFCSLLC10	$1\overset{''}{.}0$ center
SCFCSLLC20	$2\overset{''}{.}0$ center
SCFCSLLO02	$0\overset{''}{.}2$ offset
SCFCSLLO04	$0\overset{''}{.}4$ offset
SCFCSLLO06	$0\overset{''}{.}6$ offset
SCFCSLLO08	$0\overset{''}{.}8$ offset
SCFCSLLO10	$1\overset{''}{.}0$ offset
SCFCSLLO20	$2\overset{''}{.}0$ offset
SCFCMS0???	MOS mask

Table 6: The same as for Table 5, but for the slits. The "?" marks for the MOS mask are the 3 digit of the observation numbers.

ID	Filter
SCFCFLBU01	U-band
SCFCFLBB01	B-band
SCFCFLBV01	V-band
SCFCFLBR01	R-band
SCFCFLBI01	I-band
SCFCFLN373	373 nm narrow-band
SCFCFLN386	386 nm narrow-band
SCFCFLN487	487 nm narrow-band
SCFCFLN502	502 nm narrow-band
SCFCFLN512	512 nm narrow-band
SCFCFLN642	642 nm narrow-band
SCFCFLN670	670 nm narrow-band
SCFCFLBSZ1	SDSS z$'$
SCFCFLSO58	O58
SCFCFLSY47	Y47
SCFCFLL600	L600
SCFCFLL550	L550
SCFCFLLC50	C50

Table 7: The same as for Table 5, but for filters.

B Normalization of Flat Field Spectrum

In §5.3, we normalized the domeflat spectrum by dividing by a constant. However, one may want to remove spectral features before flat fielding because the lamp spectrum may not be uniform.

Figure 15 shows an example spectrum of the domeflat taken with the FOCAS 300 R grism and O58 filter; there are two absorption dips due to the dome flat screen. Clearly, one should not divide object data by such a dome flat spectrum because the absorption dips generate excess emission to the object spectrum. Although these excess emission may be removed in the later data reduction step (e.g., flux calibration), it would be prudent to remove the overall pattern of the flat spectrum in advance. We present two methods of normalization in Appendices B.1 and B.2. We subsequently discuss how these methods should be applied to FOCAS data in Appendix B.3 in conjunction with §5.3.
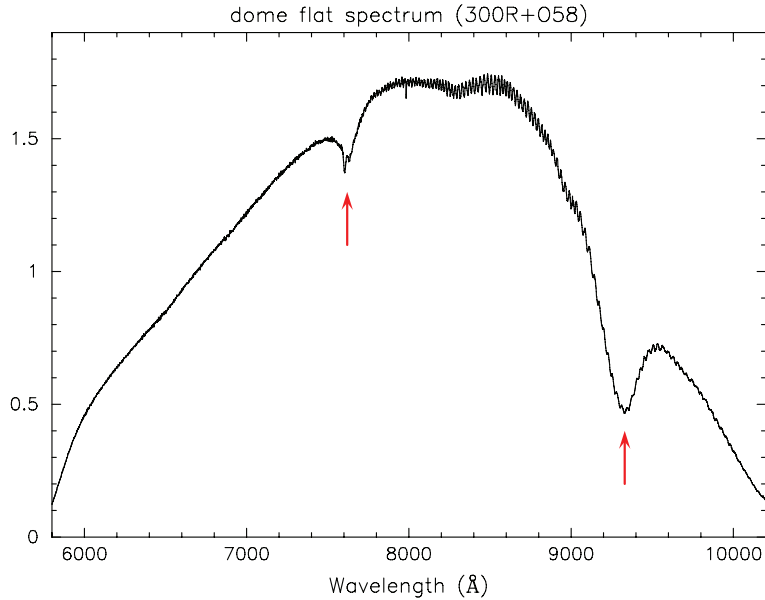
Figure 15: An example of dome flat spectrum taken with 300 R grism and O58 filter. The vertical scale is dimensionless arbitrary unit. The red arrows indicate the absorption pattern due to the dome flat screen.

B.1 Normalization with the IRAF task `response`

In this subsection, we describe a method to normalize the dome flat spectrum with the IRAF task `response`.

```
fo> epar response


PACKAGE = longslit
   TASK = response

calibrat=                              Longslit calibration images
normaliz=                              Normalization spectrum images
response=                              Response function images
(interac=                       yes)  Fit normalization spectrum interactively?
(thresho=                     INDEF)  Response threshold
(sample =                         *)  Sample of points to use in fit
(naverag=                         1)  Number of points in sample averaging
(functio=                   spline3)  Fitting function
(order  =                         1)  Order of fitting function
```

```
        (low_rej=                      0.) Low rejection in sigma of fit
        (high_re=                      0.) High rejection in sigma of fit
        (niterat=                       1) Number of rejection iterations
        (grow   =                      0.) Rejection growing radius
        (graphic=                stdgraph) Graphics output device
        (cursor =                        ) Graphics cursor input
        (mode   =                       q)
```

Using this task, we normalize the `flat.fits` created in §5.3:

```
fo> response flat[201:500,5:2700] flat[431:500,5:2700] flat.nr2
          low_rej=3 high_rej=3 niter=3
```

`flat[201:500,5:2700]`	:	Input image
		This example uses a part of `flat.fits` image as an input. The output
		image has a pixel value of 1.0 for the region(s) not-specified here.
`flat[431:500,5:2700]`	:	Image for which a normalized spectrum is extracted.
		Notice that we selected the region free from the vignetting
		that can be seen to the bottom-left of `flat.fits` (see Figure 17).
`flat.nr2`	:	Output image name

Above command displays the averaged `flat.fits` over $x = 431 - 500$ region as shown in Figure 16a. Try the "`k`" key to look at the ratio of the data and fit (Figure 16b). Verify the fitting result with the plot. If you are not satisfied with the results, try to increase/decrease the fitting order. You can change the fitting order by typing "`:order`" in the same fashion as we repeatedly used in the other tasks. If you want to initialize the plot, type "`h`".

Our experience suggests that it is almost impossible to obtain reasonable fitting results to the dip seen $y \sim 500$; we thus eliminate it from the fitting region using one of the task options of "`s`" (see Figure 16c). Here, we used a rather large fitting order of 50 to eliminate the overall pattern of the flat spectrum.

Type "`q`" to quit the task `response`. The task generates an output file, `flat.nr2.fits`, where each column between $x = 201$ and 500 of the input file has been divided by the best-fit function. The resultant image (Figure 17), whose pixel values are around 1.0, should retain information about inequity of the pixel sensitivity (i.e., pixel-to-pixel sensitivity variation) and those along the slit direction.
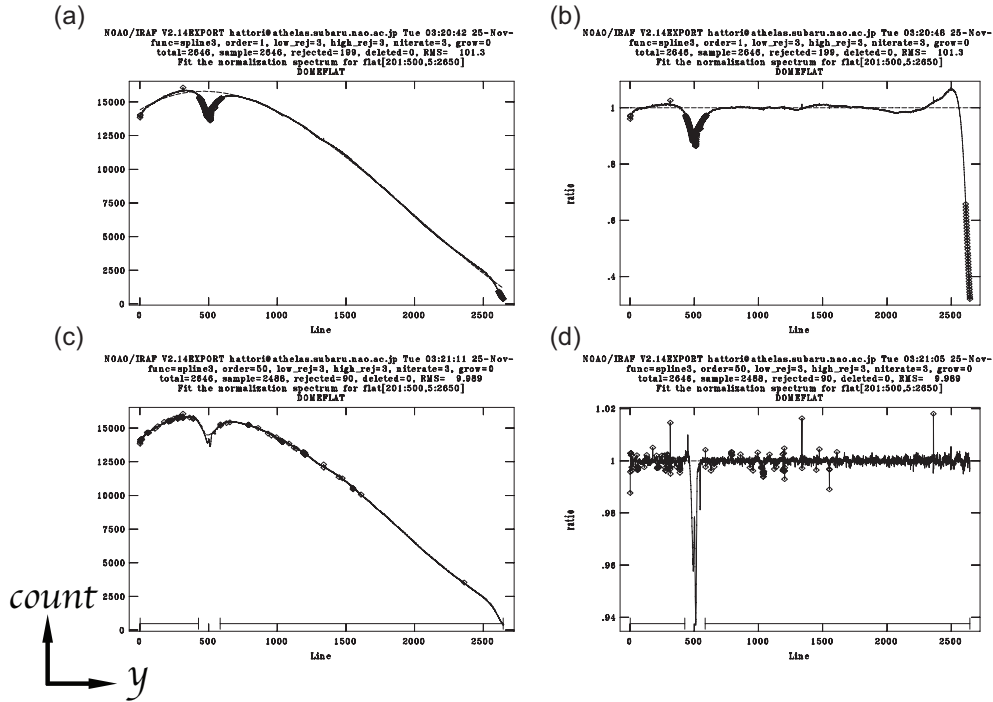
Figure 16: (a) The initial window of the IRAF task "`response`", (b) An example of the updated window whose vertical scale is the ratio of the data and fit; the window is obtained by typing "`k`". (c) and (d) show results with the different fitting range and/or the order. The horizontal and vertical axes indicate Line (i.e., y-axis in `flat.fits`) and pixel count, respectively.
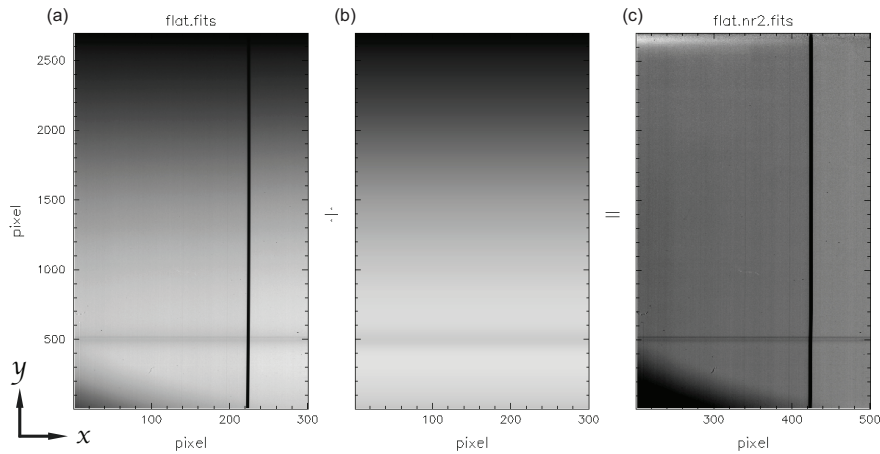

Figure 17: Normalized flat field image for spectroscopic data. The right-hand panel (`flat.nr2.fits`) is obtained by dividing the left-hand image (`flat.fits`) with the central image.

46

B.2 Normalization with the `focasred` task `flatnorm`

Alternative method for normalization of spectroscopic flat field image is to use the task `flatnorm` in `focasred` package. The `flatnorm` task normalizes the input spectrum using an averaged image that is obtained over a specified region and does not perform any fitting, which differs from the method described in Appendix B.1.

The advantage of this method is that we do not need to adjust order of fitting. Moreover, this method can readily correct for the abrupt change(s) and/or discontinued point(s) in spectra, such as the "absorption" dip seen at $x = 500$ (see Figure 16a). On the other hand, the method has disadvantage that it corrects all the artifacts, like fringe pattern, which should be left in the spectrum.

```
fo> imcopy flat[201:500,5:2700] flat.tr
fo> flatnorm flat.tr flat.nr3 x1=231 x2=300
```

The `imcopy` commands extracts the region needed to normalizes, and `flatnorm` normalize the spectrum in the given area. Compare the results with those from §B.1.

B.3 The Suggested Approach

In this subsection, we compare the two normalization methods described in Appendices B.1 and B.2 together with §5.3 for helping readers to select the most suitable method in the case of long-slit observations at FOCAS.

Figure 18 represents the bandpass characteristics (i.e., transmission curve) of the order-sorting filters used at FOCAS [8]. The L550 and L600 filters, which are used for the short-wavelength observations, are interference filters and have the ripple patterns less than 3% in their transmission curves. When observing with these filters, the resultant spectra of target(s) clearly contain such a ripple pattern which must be removed in the flat fielding process. Therefore, we believe that one should not remove such a pattern by the normalization process in the previous subsections. On the other hand, the dome flat spectrum of the Subaru Telescope has a "simple" spectrum profile at $\lambda \lesssim 7000$ Å (see Figure 15).

In summary, one must accordingly correct the intrinsic pattern (i.e., the response function) of the instruments in the case of the short-wavelength spectroscopy observations with L550 and L600, whereas, the dome flat spectrum in the short-wavelength does not have intrinsic

---

[8]http://www.naoj.org/Observing/Instruments/FOCAS/Detail/UsersGuide/PartsDetail/Filters/ FilterList.html
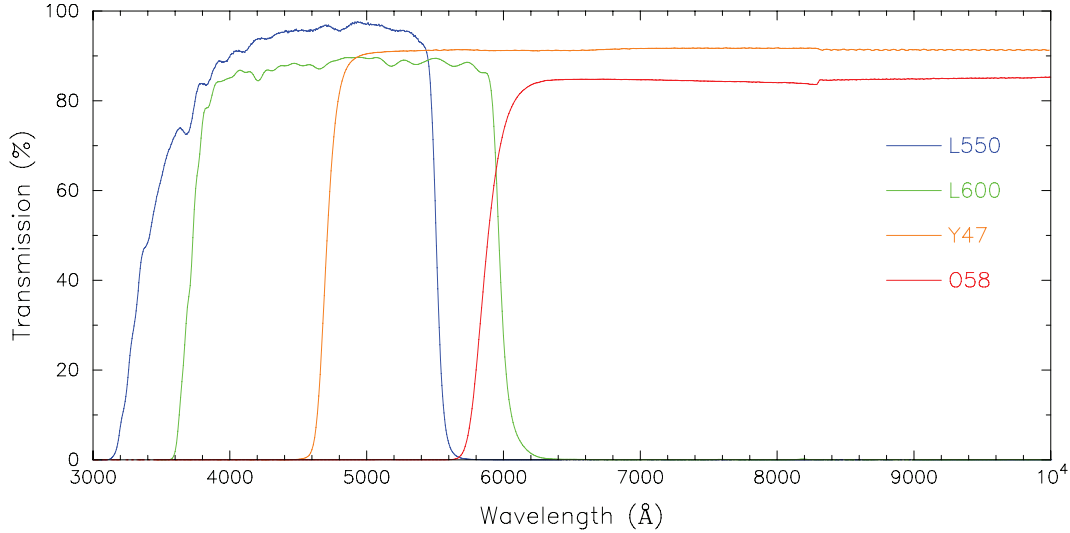
Figure 18: The bandpass characteristics (i.e., transmission curve) of the order-sorting filters used at FOCAS.

artificial pattern needed to correct (recall the simple profile of $\lambda \lesssim 7000\,\text{Å}$ in Figure 15). In this case, we suggest normalizing flat filed images only by dividing by a constant as we done in §5.3. If you want to attain high accuracy in flux calibration in $\lambda < 4900\,\text{Å}$ with Y47 or $\lambda < 6100\,\text{Å}$ with O58 where the transmissions of these filters significantly drop, we suggest adopting the same approach to obtain decent results.

The Y47 and O58 filters are used for the long-wavelength observations and have almost flat filter bandpass shapes, except for the upper band edges ($\lambda \sim 4700$ Å for Y47 and 5800 Å for O58). Recall that the dome flat spectrum have spectral features at $\lambda \gtrsim 7500\,\text{Å}$ (see Figure 15). If observing $\lambda \gtrsim 7500\,\text{Å}$, we suggest normalizing the flat fielding image by the methods described in Appendices B.1 and B.2.