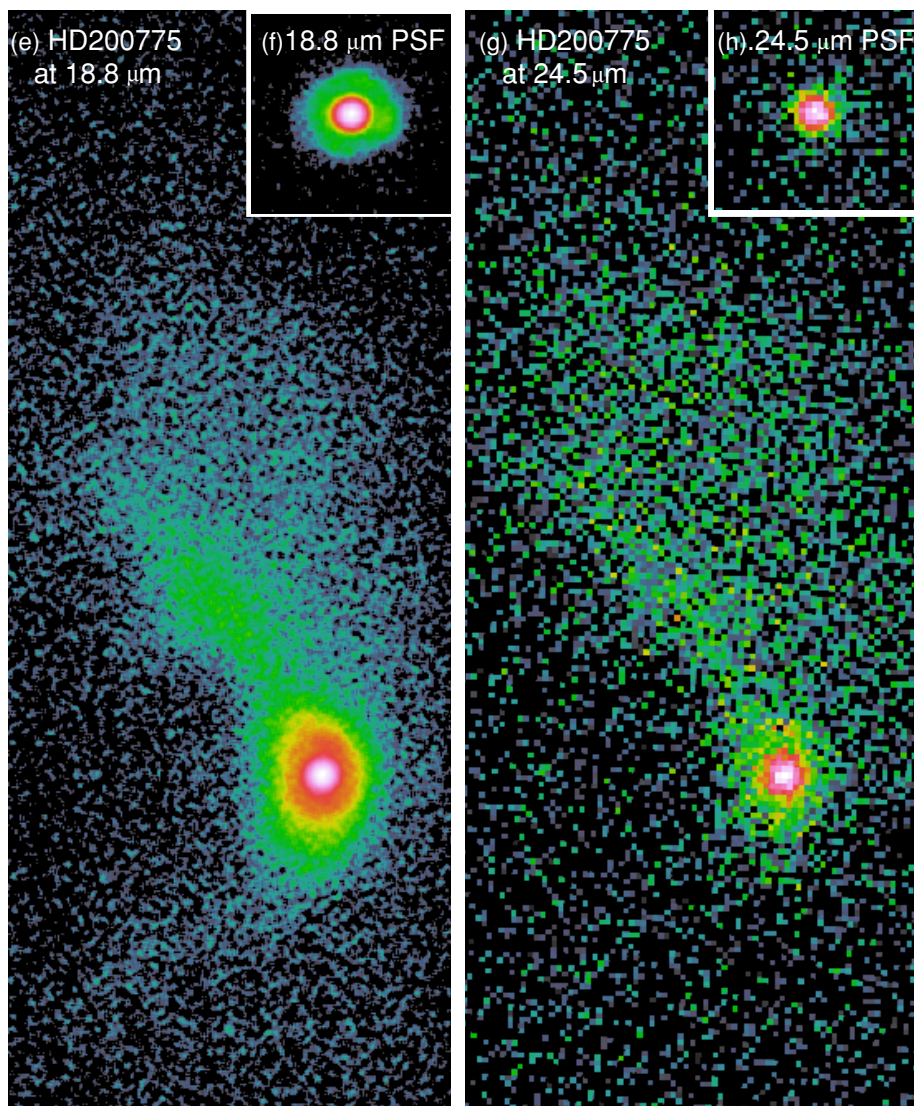


Subaru Data Reduction CookBook: COMICS

— VERSION. 2.1.2E (NOVEMBER 21, 2012)—



Based on the textbook in Japanese by Y. Okamoto

Combined effort of the past and current members of the COMICS team and Subaru Telescope

Current Version Editor: Ray S. Furuya

Contents

1	INTRODUCTION OF COMICS	3
1.1	Mid-infrared Astronomy	3
1.2	An Overview of COMICS	4
2	COMICS Data Set	5
2.1	Chopping and Nodding, and Structure of Raw Data	5
2.2	Summary of COMICS Data Reduction	9
3	FITS FILE AND HEADER	14
3.1	From the 1st to 8th Lines	14
3.2	Header Information Common in All Subaru Data	15
3.2.1	Header Parameters for the Status of the Telescope	15
3.2.2	Header Parameters for Setting Optics	16
3.3	Header for COMICS	17
4	Overview of Data Reduction: A Standard Procedure	19
4.1	Getting Started: the Reduction Software	20
4.1.1	q_series	20
4.1.2	IRAF	20
4.2	Which Data Should I Work On?	20
5	Data Reduction: The Details	23
5.1	Imaging at N and Q Bands	23
5.1.1	Making Dark Frames	23
5.1.2	Making Flat Images	24
5.1.3	Making a Bad-Pixel Mask	26
5.1.4	Reduction of Science Data	27
5.1.5	Adding the Images	28
5.1.6	Miscellaneous	36
5.2	Analyzing Spectroscopy (N-band Low Resolution; NL) Data	38
5.2.1	Preparing for a Dark Frame	38
5.2.2	Making a Flat Frame	38
5.2.3	Reduction of the Read-out Pattern Noise	40

5.2.4	Data Reduction of Scientific Images	41
5.2.5	Extracting the Spectra	48
5.2.6	Adding the Spectra	49
5.2.7	Wavelength Corrections	49
5.2.8	Correction of Atmospheric Absorption and Efficiency	49
5.2.9	Diffuse Emission	51
5.3	Frequently Asked Questions	52
5.3.1	Contamination of the Ozone Line at 9.6 μm	52
5.3.2	Should I perform nodding for my observations?	52
5.3.3	I have a "ghost" in my low-resolution image at the N band, what can I do?	52
5.3.4	I get errors when I transform my image with <code>geotran</code>	53
5.3.5	<code>geomap</code> relevant questions	53
5.3.6	It takes a loooooooooooooooooong time for task <code>q_startrace</code> to complete	53

Revision History

2001/05/24	Ver. 0.0j	Yoshiko K. Okamoto	The original version
2004/11/13	Ver. 1.1 β -j	Mitsuhiko Honda /Yoko Okada	Revision
2007/09/24	Ver. 2.0j	Takashi Miyata	Revised for Subaru Data Reduction School
2007/10/09	Ver. 2.1j	Takafumi Kamizuka, Takashi Miyata, and Ray S. Furuya	Overall revision, particularly for §5 Deleted description about s_series as it has been merged with q_series
2008/04/01	Ver.2.1.0e	Ray S. Furuya, discussed w/ T.M., T.U., & T.K., et al.	English version
2008/05/15	Ver.2.1.1e	Ray S. Furuya	Corrected errors
2012/11/21	Ver.2.1.2e	R.S.F. & C.C.P.	Language corrections

Chapter 1

INTRODUCTION OF COMICS

1.1 Mid-infrared Astronomy

The mid-infrared (MIR) regime spans between ~ 5 and $30 \mu\text{m}$ in wavelength (λ); there exist two windows of the atmosphere surrounding Earth in this MIR regime. These are referred to as the N and Q bands, which have been long-unexplored because of limitations imposed by high background noise and technical difficulties in the MIR detectors. However, this situation has changed dramatically due to recent progress in the development of next-generation 2-dimensional (2D) array detectors, as well as the construction of large-diameter optical/infrared telescopes. In order to make MIR observations possible at Subaru Telescope, Cooled Mid-Infrared Camera and Spectrometer (COMICS) has been developed as one of the first-generation instruments by the team led by Professors H. Kataza and T. Yamashita.

The MIR wavelength regime stands unique due to a number of special characteristics. First, these longer wavelengths can penetrate interstellar clouds than the shorter ones, e.g., optical/near infrared (NIR) do. This allows us to perform deeper searches for deeply embedded and newly found objects identified in optical wavelengths, although MIR emissions suffer from absorptions due to interstellar silicates. Second, the MIR regime is the most efficient band to detect blackbody (thermal) radiation at temperatures of $100 - 400 \text{ K}$, making COMICS one of the best instruments to observe warm thermal emission from circumstellar disks around young stellar objects and warm gas and dust around late-type stars. Third, the MIR regime is a key band to trace the role and evolution of interstellar dust because there are numerous emission and absorption features spread across it. The best-known of these features can be seen around $9.7 \mu\text{m}$ and $18 \mu\text{m}$, and are believed to be of silicate origin and detected over rather wide ranges in the two atmospheric windows. In addition, it is known that there exist unidentified IR (UIR) bands peaked at $\lambda = 6.2, 7.7, 8.65, 11.25, \text{ and } 12.8 \mu\text{m}$, which can be detected towards various type of objects.

1.2 An Overview of COMICS

COMICS is an instrument at the 8-meter Subaru Telescope which is also optimized to observe MIR wavelengths. It is expected to explore various astronomical objects such as those in the solar system, newly formed stars and planets, interstellar dust, as well as nearby and even distant galaxies. COMICS offers two modes of observations: imaging and long-slit spectroscopy. COMICS is mounted at the Cassegrain focus to minimize thermal emission from the telescope. In general, one must cool down the instruments for the MIR bands since the black body radiation peaks around $10 \mu\text{m}$ at room temperature. In fact, all the optics and the associated parts (such as motors) are stored in the dewar and cooled down to less than 30 K. In addition, you need to cool the detectors (mentioned below) further, down to 4 K. This reduces dark currents. In order to have diffraction-limited image with the 8 m telescope, COMICS is designed with a pixel field of $\sim 0.13^{\text{arcsec/pix}}$ for imaging mode. For spectroscopic observations at $10 \mu\text{m}$ (N band), COMICS offers low (L)-, medium (M)-, and high (H) resolution modes with $R \sim 250$, 2500, and 10000, which are designated as NL, NM, and NH, respectively. At $20 \mu\text{m}$ (Q band), COMICS provides medium- and high-resolution modes with $R \sim 250$ (QM) and 5300 (QH), respectively. COMICS equips four gratings for all the modes except for the high-resolution mode at Q-band. We adopted Si:As 320×240 IBC array detector which is sensitive between 8 and $28 \mu\text{m}$, provided by the Raytheon. The IBC type detectors has a fairly high quantum efficiency which enables us to improve detection limits significantly. In the imaging mode, COMICS uses one of the IBC-type detector. Users can obtain a relationship between the slit and the observing target(s) with a high accuracy as the detector at the imaging side works as the slit viewer. For spectroscopic observations, we configure all five detectors in parallel to achieve high observing efficiency. However, one must slightly rotate the grating to detect photons that fall into the gap between the detectors. Most of the spectroscopic observations have configured two sets of such rotation angles to detect all the photons without gaps along the wavelength, which makes it possible to observe the entire atmospheric window in the medium resolution mode. In addition, we offer a total of four slits — available in both the 10 and $20 \mu\text{m}$ bands — having different widths for possible variation in seeing. Table 1.1 summarizes the key parameters of the instrument.

	Imaging	Spectroscopy
Pixel Field of View	$0.130''/\text{pix}$	$0.165''/\text{pix}$
Detector	320×240 Si:As $\times 1$	320×240 Si:As $\times 5$
Field of View	$42'' \times 31''$	$40''$ for long-slit
Wavelengths	N: $8.8\text{--}12.4 \mu\text{m}$ (5 bands) Narrow bands for UIR 8.6, 11.24, [ArIII], [SIV], and [NeII] Q: $17.7\text{--}24.5 \mu\text{m}$ (4 bands)	N: $7.8\text{--}13.3 \mu\text{m}$ Q: $16.7\text{--}20.7 \mu\text{m}$
Spectral Resolution	$R \sim 10\text{--}20$ $R \sim 60\text{--}70$ (narrow band filters)	N band: $R \sim 250, 2500, 10000$ Q band: $R \sim 2500, 5300$

Table 1.1: An Overview of COMICS

Chapter 2

COMICS Data Set

This chapter provides an introduction to COMICS data and FITS file structure, and gives a rough idea of how to reduce observing data.

2.1 Chopping and Nodding, and Structure of Raw Data

For most of observations with COMICS, high-speed read-out is used to compensate for the highly time-variable background level in the MIR bands. For this purpose, we employ "chopping" of the secondary mirror and "nodding" of the primary mirror. Most of the atmospheric background variations can be removed by the chopping method. However, since chopping is being done through tilting of the secondary, a subtracted image obtained (by taking one image from another) leaves a low-level residual pattern in many cases. This is because optics for the two chopped beams are slightly different from each other ¹. Such a residual pattern can readily decrease detection limits, which severely affects the ability to detect faint objects. To avoid this effect, it is common to "nod" the telescope so that the source appears alternately in the signal and reference beams. Nodding scans observe an offset position with a chopping sequence that is the same as for the target position. This will cause a similar residual pattern to that seen in the target position. In the data reduction process, we subtract the offset images(s) to remove this low-level residual pattern. Given this purpose, it is sufficient to "nod" with a lower throw-frequency than "chopping" does. A nodding cycle is typically several minutes to obtain a pair of images. Lastly, one should keep in mind that the decision to use nodding depends on the source brightness and/or scientific goals.

Figures 2.1 and 2.2 illustrate the rough idea of how COMICS obtains a set of images and shows structures of the image FITS files, respectively. A standard observation will perform chopping and nodding scans towards slightly different pointings so that the target will fall onto different pixel positions of the images (see Figure 2.1). Since chopping will be done at each nodding beam, the user should obtain a set of data consisting of the four images labeled

¹Although telescope efficiency is suggested to be responsible for causing such a residual pattern, amplitude level of the residual pattern may be ignored at Subaru Telescope depending on the scientific cases.

A to D (see Figure 2.1). The direction, as well as throws for chopping and nodding, must be selected by considering not only source structure but also its surrounding environment. For instance, the user may select a chopping throw of $\sim 10''$ toward an arbitrary direction² in the case of a point-like source located in a region without diffuse emission. On the other hand, the user should carefully select these parameters in the case of extended sources and/or deeply embedded object(s) in diffuse emission. The user should take into account several constraints such as source size, clumpiness of the surrounding clouds, and so on. For many cases, such a selection gives reasonable atmospheric cancellation and good mechanical performance.

As can be seen in Figure 2.2, each of the four images (labeled A, B, C and D) contains multiple images. Assuming that the user has repeated n times of chopping scans (i.e., $\text{ChopNum} = 2n$), the image data taken with beam A should have the number of n images with $A_1 \sim A_n$. Each of the $A_1, A_2, \dots,$ and A_n th images contains either a summation image over the m times of read-outs (= exposures), i.e., $\Sigma_i A_j^i$ (ADD mode), or sub-images without making the summation image (RAW mode). In the latter case, A_j th image should include the number of m raw images of $A_j^1, A_j^2, A_j^3, \dots,$ and A_j^m . Another set of the chopping data, i.e., the beam C images, has the identical data structure. Each nodding image has a substructure with the two chopped beam images. The nodding images from beams A and C constitute a pair together with those from the nodding beams of B: $B_1 \sim B_n$ and D: $D_1 \sim D_n$. In conclusion, a standard set of COMICS data that are ready to reduce contain the other nodded beam data of B: $B_1 \sim B_n$ and D: $D_1 \sim D_n$.

We define names of images in each substructure as follows.

Name of image	Description	Notation
nodding beam	Nodding images having an identical pointing center	All the images in the A and C
chopping beam	The same as above, but for chopping	All the images in A
beam	Taken at different positions due to chopping and nodding	A, B, C, D
file	Each FITS file for the obtained image	pairs of A and C, B and D
frame	An image set taken within a chop beam	Subscript of A
exposure	Each readout	Superscript of A

In practice, COMICS creates three types of data sets.

1) RAW mode: keeps all the exposure and all the frames. Images in each file are A_i^j and C_i^j .

2) ADD mode: adds all the exposures within a frame (i.e., collapses all the images into an image), but keeps all the frames. This is the data set for a typical COMICS observation. Images in each file are $\Sigma_j A_i^j$ and $\Sigma_j C_i^j$.

²We suggest that the user select a chop direction that keeps the secondary mirror at the most stable position.

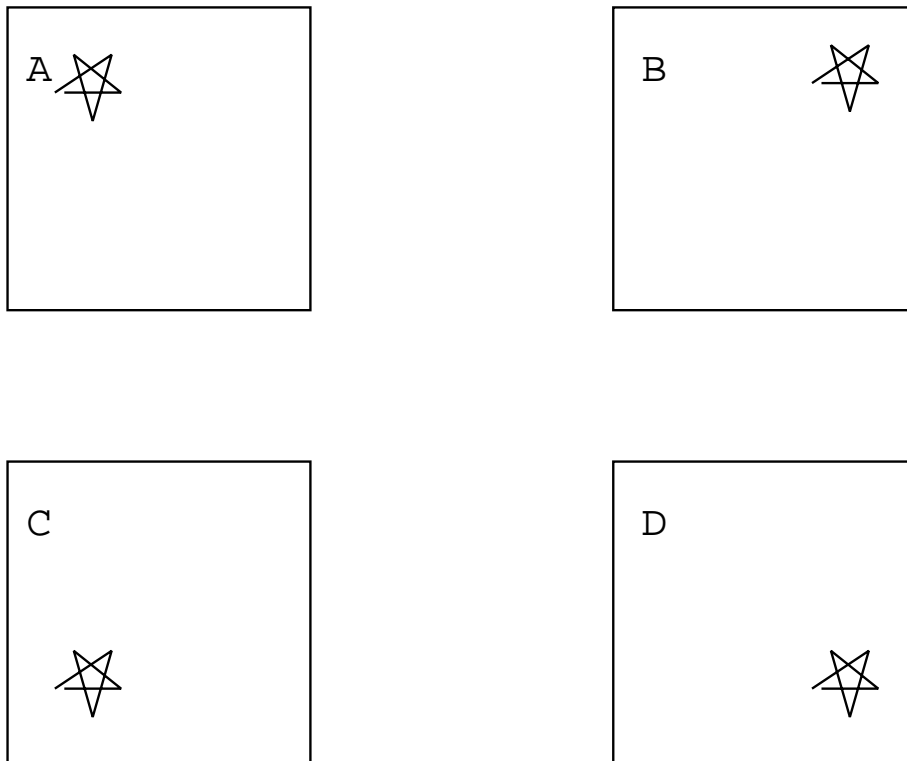
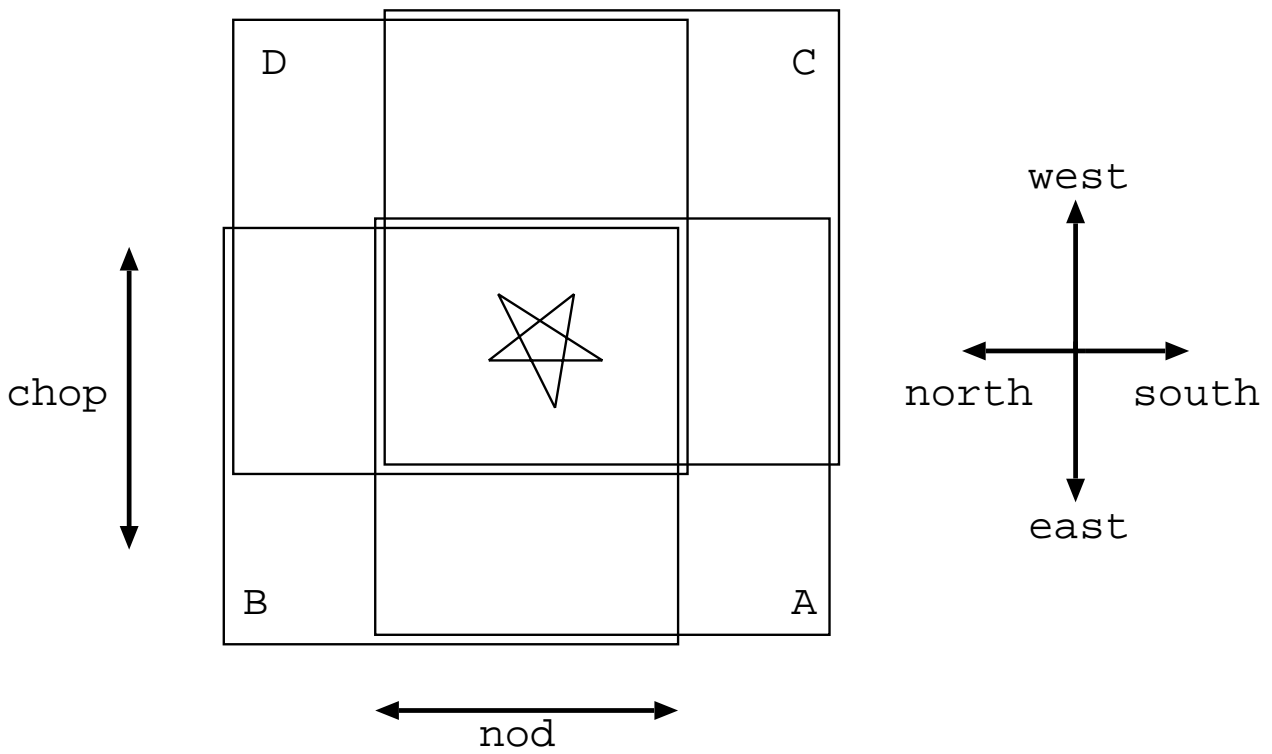


Figure 2.1: An illustration of a typical data set taken with COMICS. The user will obtain a data set consisting of four different images labeled A – D. These 4 images have been taken at slightly different pointing centers in the plane of the sky through chopping and nodding scans. The user can select an arbitrary direction for secondary mirror chopping and primarily mirror nodding. Note that each of the images contains multiple images, as shown in Figure 2.2.

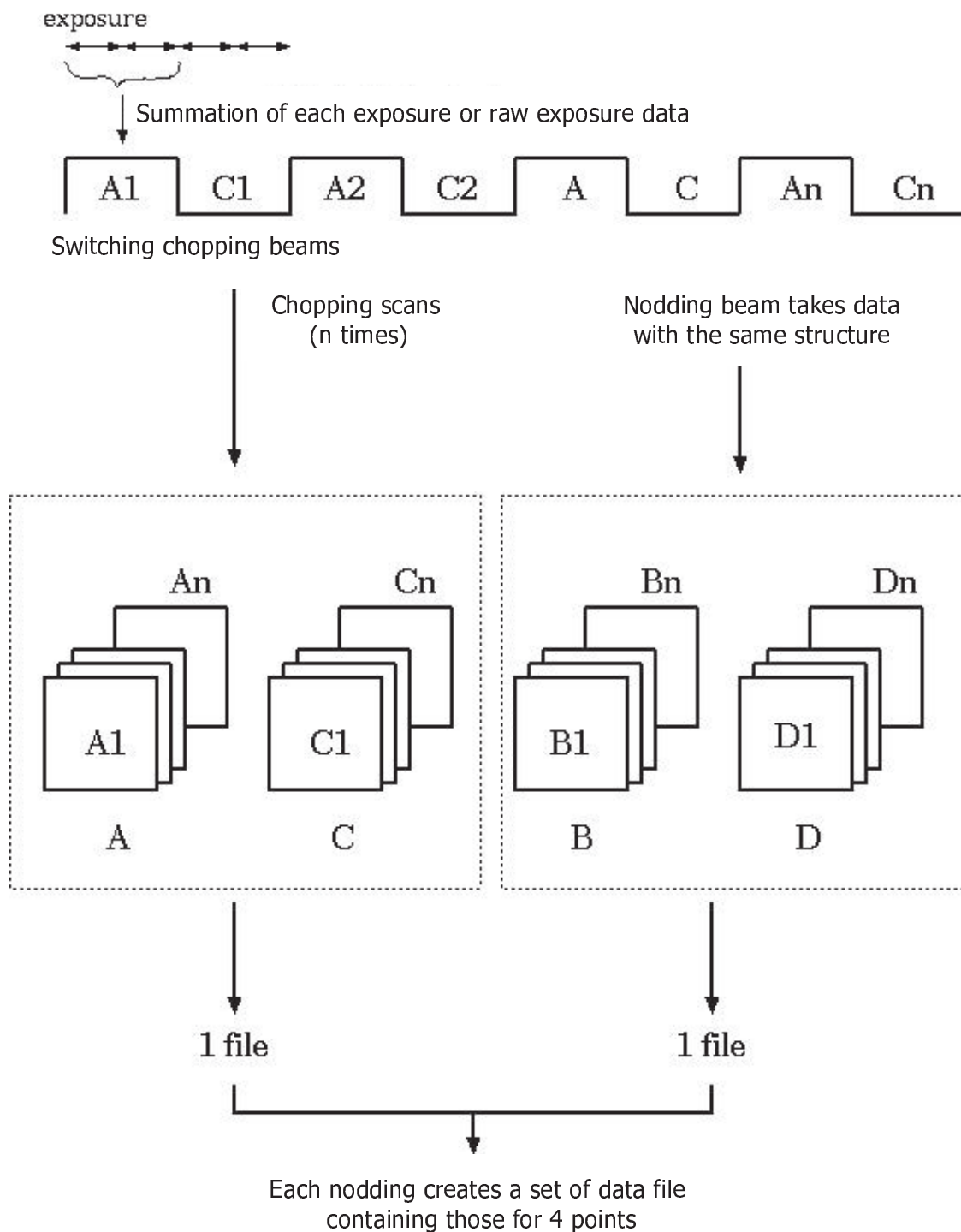


Figure 2.2: A schematic image of a typical a COMICS data file. The chopped and noded beam images set illustrated in Figure 2.1 individually contain sub-images. This is because almost all the data will be taken through multiple chopping scans for each nodding beam. Each chopping beam image contains either the further multiple sub-images, (i.e., the raw data) or a summed image of them.

2) ECO mode: Since this mode collapses all the images into an image on-line, and makes a further summation for all the frames at each beam, the user will obtain a map for the chopping and nodding at each. Images in each file are $\Sigma_{i,j}A_i^j$ and $\Sigma_{i,j}C_i^j$.

In practice, some observers may have acquired images without chopping and nodding for obtaining dome-flat data (see §5.2.3). In this case, `Chop=0` appears in their headers. Users should see multiple frames whose numbers have been specified with the parameter of `ChopNum`. Notice that the number of exposures has been repeated with that specified in the individual frames as well. As explained above, all the exposure frames are kept for `CoAdd = 0`, while the number of `ChopNum` images are saved with `CoAdd = 1` because all the exposure images in each frame are summed. Lastly, the user obtains two images for `CoAdd = 2` because all the images have been summed for each chopped beam.

2.2 Summary of COMICS Data Reduction

This section describes an overview of the COMICS data reduction to obtain fully calibrated images. The details of each step of data reduction will be described in §5.

1. Subtraction of Chopping and Nodding Data We assume that the users has obtained a non-calibrated (i.e., “raw”) data set containing four images (§2.1) where the object of interest can be seen at different pixel positions. The user begins by subtracting chopped images taken sequentially with the different chopped beams (note that they must be taken with the same nodding beam). We refer to it as chopping subtraction. Namely, as shown in Figure 2.2, the user needs to perform the subtraction as follows,

$$(A_1 - C_1) + (A_2 - C_2) + \cdots + (A_n - C_n) = \alpha$$

$$(B_1 - D_1) + (B_2 - D_2) + \cdots + (B_n - D_n) = \beta$$

which gives two images after the chopping subtraction (see Figure 2.3). This operation usually makes it possible to subtract almost all the background pattern as well as the residual one originating from the dark current. However, the user may find a very low-level residual pattern which is caused by the use of different optics. To eliminate such a pattern, the user would need to subtract an image generated by the chopping subtraction (hereafter, nodding subtraction),

$$\alpha - \beta = I$$

to obtain a nodding-subtracted image I . For many cases, the above operation will give you images with sky levels close to zero. If you have made multiple nodding observations, you need to make an image, I , for each nodding.

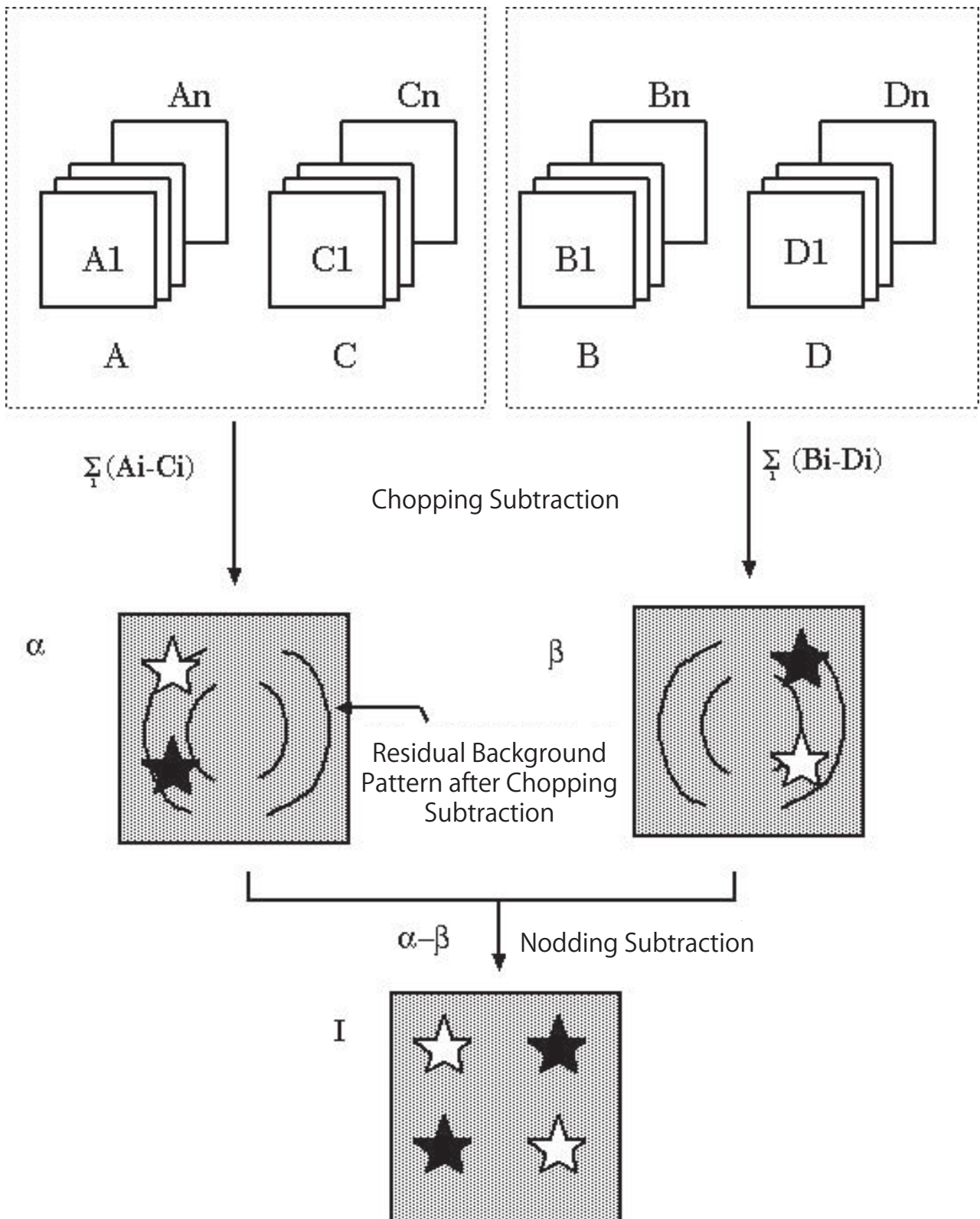


Figure 2.3: Subtraction of chopped and nodded images: The first step in the COMICS data reduction process is subtracting images between the same nodding beams and between the different chopped beams; the resulting four images are stored in a data set. The subtraction between the images A and C in the upper panel gives the image α in the middle (the same for the B and D to make the subtracted image β). Subsequently, the user needs to subtract the two images created by the chopping subtraction to make the desired image shown in the bottom panel ($\alpha, \beta \rightarrow I$); this operation will eliminate residual patterns caused by the use of slightly different optics. Notice that the resultant image of I should contain the positive (white) and the negative (black) portion for the source of interest as a result of the subtraction processes shown above.

2. Flat Fielding The next step is to divide each of the chopping- and nodding-subtracted images by a flat image to correct any inequity in the gain of the detector pixels. Flat images are usually made through sky-flat and dome-flat observations. The former is used for imaging and the latter for spectroscopy. For most COMICS observations, dome-flat data are taken by observing thermal emission from the dome, windscreen, or mirror covers etc, instead of from observing lamps.

3. Image Transformation (Distortion Correction) and Wavelength Calibration for Spectroscopy Observations For spectroscopic observations, the user will need to correct distortion of the spectrum that shows a curved feature in the images because COMICS guides the incident wave obliquely with respect to the grating. Figure 2.4 illustrates a standard procedure to correct such a distortion. First, one has to derive a "spatial axis" from a point-source image such as the standard stars. Here, a "spatial axis" is the axis free from distortion in the fully calibrated image (see Figure 2.4). On the other hand, one should make a spectrum image for the sky that is created from, for example, summation or/and mean images using all the chopped beams for the common nodding beam. Keep in mind that all the images used for calibration must be made with the images found in the data-set file that contains scientific images(s) to be calibrated. The second step is to derive "dispersion axis" from the sky spectral image, then transform the image so that the spatial and dispersion axes are perpendicular to each other. Finally, it is important to calibrate the dispersion axis using a comparison with a standard model of terrestrial atmospheric emission.

4. Adding the Calibrated Images The next step is to add the calibrated images to increase the signal-to-noise ratio (S/N). This operation must be done only for the images taken at the same image center. (Check the telescope pointing accuracy during your observations as well!) In some cases, in order to reduce the number of FITS files, users may want to add certain numbers of images taken at the same position with a long observing time. (this requirement comes from a practical limitation of image numbers that can be stored in a file). Finally, we suggest that users double-check whether or not the target source has shifted by one or more pixels, even for images taken at the same position.

5. Flux Calibration with Standard Star(s) The final step is to convert unit of the image, namely, from count into brightness. It is not easy to find (a) well-test catalog(s) of standard stars at MIR wavelengths. To our knowledge, the catalog by Cohen and colleagues (Cohen et al. 1999, AJ 117, 1864) deals with a rather large number of standard stars based on the IRAS and LRS observations. In addition, the user may use some catalogs available online, such as the UKIRT list. However, that lists only about 20 bright stars.

Cohen et al. (1999) listed flux densities at N and Q bands measured at low-resolution. To apply their results to your data, you must properly handle differences in the slit efficiencies between theirs and that of COMICS. Notice that this effect is particularly important when you analyze diffuse emission because COMICS employs long-slit spectroscopy. For photometry observations, the user is required to calculate flux of the desired standard sources by integrating

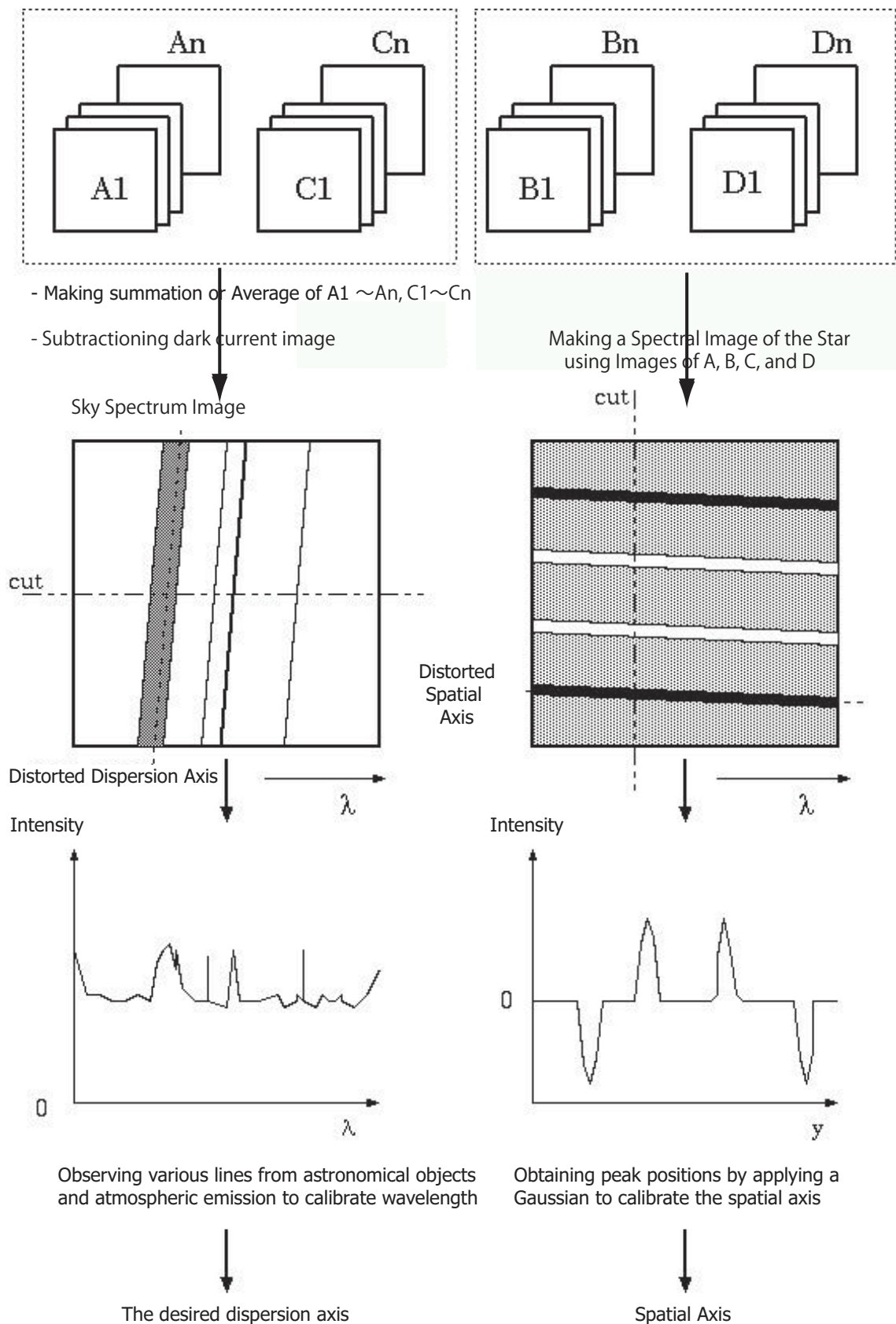


Figure 2.4: Determination of wavelength and spatial axes in spectroscopic observations. The absolute coordinates of the wavelength axis can be identified from a comparison of the spectral image obtained from a summation of the raw data with an atmospheric transmission model. As for the spatial constant axis, the users should determine the axis by applying a Gaussian profile to the obtained spatial cutting line which passes the point-source along the wavelength axis.

the spectra shown in the paper for each N band filter. This is because almost no systematic measurements have been done for individual filters, due to the very high background noise.

Finally, our experience strongly suggests that the user should pay attention not only to differences in airmass but also the presence or absence of clouds in the sky because they heavily affect the quality of the images. Given that sky noise level is intrinsically high at MIR bands regardless of the clouds, some observers may continue to observe with a small amount of cloud cover. However, it should be noted that even small amount of clouds can easily degrade your image quality. In many cases, the user is required to correct the effect of the atmospheric absorption even for the standard star data obtained with the same airmass. In conclusion, we strongly suggest making sure that the spectral shape of the standard star(s) has a fairly good consistency with that of the atmosphere during the given observing time.

Chapter 3

FITS FILE AND HEADER

Your COMICS data are stored in FITS format. COMICS FITS data usually have two types: one is the series of image file names starting with "COMA", and the other begins with "COMQ". For both files, 8-digit images numbers follow the strings of COMA or COMQ. The COMA files are images taken with CoAdd mode, as explained in §2, while COMQ are those of the online (partially) reduced images using COMA files. Here, the COMQ data are created by collapsing all the images into an image taken with the same chopping-beam image, and subtracting another collapsed chopping-image created in the same fashion. In other word, COMQ images are the summation images obtained by adding on-beam images and off-beam images that are multiplied with -1 .

The discussion below gives an introduction on how to read FITS header information of your COMICS images. You may see a header with a different version that can be found with the `Q.GETVER` parameter at the bottom. As for the details of your FITS header and correspondence with different versions, please ask your support scientist.

3.1 From the 1st to 8th Lines

<code>SIMPLE</code>	<code>=</code>	<code>T</code>	/ Standard FITS format
<code>BITPIX</code>	<code>=</code>	<code>32</code>	/ # of bits per pixel
<code>NAXIS</code>	<code>=</code>	<code>4</code>	/ of axis in frame
<code>NAXIS1</code>	<code>=</code>	<code>320</code>	/ # of pixels/row
<code>NAXIS2</code>	<code>=</code>	<code>240</code>	/ # of pixels/column
<code>NAXIS3</code>	<code>=</code>	<code>50</code>	/ # of pixels/time
<code>NAXIS4</code>	<code>=</code>	<code>2</code>	/ # of pixels/detector
<code>EXTEND</code>	<code>=</code>	<code>F</code>	/ ASCII Extension Table

The first eight lines in COMICS header follow the FITS format standard defined by the IAU. The second parameter — `BITPIX` — shows bit number; `BITPIX` with minus ($-$) indicates the numbers of floating decimal points, while one without minus sign indicates the integer,

which is the case for the current COMA and COMQ images. You will see BITPIX= -64 for all the FITS files that are created by the COMICS data reduction package of the `q_series`¹.

The third parameter, `NAXIS = 4`, indicates the dimension number of the image files. Note that COMICS data have a four-dimensional (4D) structure; `NAXIS1` and `NAXIS2`, respectively, represent the x- and y axes of the detector, `NAXIS3` does time-axis (`NAXIS3`), and `NAXIS4` indicates the detectors. This means you obtain one single file even if you observed with more than one spectrometer. If you observed using the imaging or low-resolution spectroscopy mode, you see `NAXIS4 = 1` for `NAXIS = 4`.

Since COMICS has a detector array consisting of 320×240 pixels, you should see `NAXIS1=320` and `NAXIS2=240`. The value found in `NAXIS3` would depend on the data format, i.e., clock settings in COMICS jargon, you gave during the observations. `NAXIS4 = 1` should be stored for imaging mode, while a different value other than 1 would be stored for each spectroscopy mode.

If you have taken both the imaging and spectroscopy data at the same time, these data are stored in two paired FITS files that have adjacent file numbers; a pair of odd- and even-numbered FITS images correspond to the imaging and spectroscopy data, respectively. In most of cases, the file number for the imaging must be smaller by one than that for the spectroscopy by one. If you observed only with imaging mode, the paired spectroscopy FITS file has only the header part.

3.2 Header Information Common in All Subaru Data

The header information after

```
COMMENT = ' ++++++ SUBARU COMMON'
```

in the COMICS header parameter is common for all the data taken with the Subaru Telescope where some basic parameters such as observing time, coordinates, configurations for COMICS can be seen. Those who need to have the more detailed information for the common part, please see "*FITS no Tebiki*" ver. 3.2. (in Japanese) and so on.

3.2.1 Header Parameters for the Status of the Telescope

We advise users to check the following parameters particularly for COMICS data.

```
INSROT = -72.158 / Instrument Rotator angle (deg)
INST-PA = 9.566 / Instrument Rotator P.A. (deg)
AUTOGUID= 'ON' / Auto Guider on/off
M2-TYPE = 'CS_IR' / Type of the Secondary Mirror (Opt/IR)
M2-TIP = 'CHOPPING' / 2nd Mirror tip-tilt on/off
```

¹The only exception is the output files from task `q_2val`, which quantizes with 2-levels, where the resultant variable type becomes integer.

```

M2-ANG1 =          -0.978 / 2nd Mirror Angle-1
M2-ANG2 =           0.687 / 2nd Mirror Angle-2
M2-POS1 =          -1.104 / 2nd Mirror Position-1
M2-POS2 =          -1.571 / 2nd Mirror Position-2

```

The first two lines give information about the "instrumental rotator"; the NST-PA parameter shows the rotator angle with respect to the source of interest. AUTOGUID tells you whether the observations have been done with the auto-guider, which may have used for most of the observations. The nine parameters starting with M2- give information about the secondary mirror (however, many users will not need to give these special attention.)

3.2.2 Header Parameters for Setting Optics

You will see the following parameters that describe coordinate information of the images. In many cases, we believe users cannot execute a command to overlay all the images simply by referring to the coordinate information described. This is because nominal tracking accuracy of the telescope is not accurate enough to allow such blind image adding. Even if this is not the case, you cannot add your images solely with the header information. For example, if you adopted a chopping throw of more than 30", imaged fields in the plane of the sky do not agree with those expected from the telescope pointing center because of the large throw of the secondary mirror.

The header after

```
COMMENT = ' ----- COMICS Optics'
```

are those for COMICS optics.

```

OBS-MOD = 'spectroscopy' / Observation Mode
FILTER01= 'F01C10.50W6.00' / Filter name/ID (pre-opt filter-1)
FILTER02= 'H21' / Filter name/ID (pre-opt filter-2)
FILTER03= 'F08C11.60W1.10' / Filter name/ID (img-opt filter)
FILTER04= 'L01L10I' / Lens name/ID (img-opt)
DISPERSR= 'G01L10L' / Identifier of the disperser used
SLIT = 'S02W160' / Identifier of the slit
SLT-LEN = 39.600 / Length of the slit used
SLT-PA = 0.0 / Slit Position Angle (degree)
SLT-WID = 0.330 / Width of the slit used
SLTCPIX1= 120.0 / Slit center projected on detector(pix)
SLTCPIX2= 160.0 / Slit center projected on detector(pix)

```

OBS-MOD indicates the observing mode of either "imaging" or "spectroscopy". Notice that OBS-MOD becomes "imaging" if you observed with the slit viewer.

FILTER01, 02, 03 and 04 are the filter names for fore-optics 1 and 2, the filter name for imaging, and the name of the lens, respectively. The first character indicates either filter (F),

lens (L) or grating (G). The next two numbers are IDs for each optical component followed by the strings indicating type of the component. For instance, C... indicates the center wavelengths in μm for F, while H?? means 'opened' has been selected with the filter wheel (the two numbers with H show the ID for the holes). L10 shows lens ID for 10 μm with the flag of I indicating Imaging. Similarly, L20 shows lens ID for 20 μm with the flag of P indicating Photometry. L10 (L20) in the grating mode with G has the same meaning as above, except for the last string of L (Low), M(Medium) and H (High) resolution (i.e., dispersion) modes, respectively.

SLIT shows the name of the slit with the two-digit numbers appended for the slit ID and W... for slit width in μm . Notice that the slit width of 160 μm corresponds to two pixels of 0.33" for spectroscopy observations; these values can be found in SLT-WID. S??W000 indicates sole the mirror without use of any slits. If this is found, the data were taken with the imaging mode without using any configurations for spectroscopy. S??D... indicate diameter of the pin-hole. S?SLT-LEN shows length of the slit, which should be constant if there is no vignetting in the image. SLTCPIX1 and SLTCPIX2 can be ignored because no meaningful parameters have been stored for COMICS observations.

The parameters of WAV-MIN and WAV-MAX store some values only for N-band spectroscopy observations using the current observing system. It should be noted that EXPTIME indicates not integration time for all the observations, but that for each exposure. DET-TMP shows the measured temperature of the mount accommodating the detectors. Last, unfortunately, the conversion factor of GAIN does not help you at all with the current system.

```
COMMENT = '          --- Spectroscopy only'
DISPAXIS=           1 / Dispersion Axis  in frame
WAV-MIN  =           7500.0000 / Shortest wavelen (nm)
WAV-MAX  =           13500.0000 / Longest wavelen (nm)
WAVELEN  =           10500.0000 / Central wavelen (nm)
COMMENT = ' ----- COMICS Detector'
EXPTIME  =           0.301 / 1 exposure integration time per exp(sec)
DET-TMP  =           7.65 / Detector temperature (K)
GAIN     =           350.000 / AD conversion factor (electron/ADU)
```

3.3 Header for COMICS

The following header lines starting "Q_" after the line of

```
COMMENT = ' ++++++ COMICS ORIGINAL'
```

are those for COMICS in particular. All the important information needed for your data reduction such as those relevant to the data type (i.e., clock parameters), observing mode, window function. In the below, we show the clock parameters that are especially helpful for your data reduction.

```

Q_DETST = '100110          ' / Detector Readout Status
Q_CLKFL = '/home/comics/cbin/clk/clkgen/012/c050.00150.001.00' / Macro File
Q_PIXTIM=          150 / Clock duration for a pixel (0.1us)
Q_RRSTRT=          1 / Reset Row Start Width (ND)
Q_CHWB =           3 / Wipe Exporsure Number in a Chop-beam
Q_CHEB =           2 / Exporsure Number in a Chop-beam
Q_CHCN =           50 / Chopping Number in this file
Q_CHAM =           1 / Add Mode 0:RAW 1:ADD 2:ECO
Q_CHOP =           1 / Chopping ON=1 OFF=0
Q_CTYPE =          0 / Clock Type 0-9
Q_YSTRT =          1 / Readout Region Y start
Q_1EXP =           0.301 / Integration time per exp. (sec) = EXPTIME
Q_1FRAME=          0.603 / Integration time per frame(co-added) (sec)

```

Q_DETST shows from which detector the data are being read from among the six detectors (i.e., one for imaging and five for spectroscopy); the parameter should be 1 for the detector being read out while 0 for the remaining ones in not use. For instance, the parameter should be 100000 for imaging observations, 100110 for low-resolution spectroscopy at the N-band, and 111111 for medium-resolution spectroscopy in the Q-band. The user should use caution in the case of using a part of spectroscopy detectors such as the case for low-resolution spectroscopy at N-band. In this particular case of 100110, the 4th dimension of the FITS files becomes (1,2), although the designation number of the detectors are 3 and 4 (one of the detectors is for reference).

Q_CLKFL shows s name for data type (i.e., clock parameter). The other important parameters to know for the clock status are Q_PIXTIM (read-our rate), Q_RRSTRT (electrical optimized ND value), Q_CHWB (number of wipes at each frame), Q_CHEB (numbers of read-out that are recognized as usable data at each frame), Q_CHCN (numbers of chopping $\times 2$), Q_CHAM (CoAdd mode), Q_CHOP (whether or not chopping is used), Q_YSTRT (the starting read-out number of "Y" in the case of a partial read-out), Q_1EXP (integration time per exposure in sec), and Q_1FRAME (integration time per frame).

The final part of the header shows the status of chopping as follows,

```

COMMENT = ' ----- CHOPPING'
Q_CHTHRW=          60.00 / Chopping Throw
Q_CHDEG =          53.34 / Chopping Degree
COMMENT = ' ----- FITS VERSION'
Q_GETVER= '4.11          ' / FITS header VERSION
END

```

Q_CHTHRW indicates chopping throw defined by peak-to-peak in arcsec, and Q_CHDEG shows the position angle for the chopping direction in degrees.

Finally Q_GETVER shows version of FITS header as described in the beginning of this section.

Chapter 4

Overview of Data Reduction: A Standard Procedure

In this chapter, we describe a standard procedure to reduce COMICS data. Since we will deal with the data with observing wavelengths more than one order of magnitude larger than the other IR/optical wavelength observations, users are strongly recommended to use the originally developed data reduction package by us. These are the `q_series` data reduction package (hereafter, `q_series`) which are written in C programs. The user can use these programs "as is" and/or can modify them.

`q_series` can be categorized into two groups. One contains the basic tools that can be used for a "quick look" while at the summit. These are mathematical operations on images, extracting a part of the data along column, row, or time, and statistical calculations for the desired portion of the image. The other group of tasks should be used to prepare for publication-quality "final" images, e.g., listing header information (`q_headlst2`), determination of spatial axis (`q_startrace`), and wavelength calibration (`q_sky_nlow`) and so on. `q_series` is specialized to reduce COMICS data, but some operations such as transformation and shifting of images are better achieved using other packages such as IRAF. We believe that both `q_series` and IRAF can handle almost all the data currently being taken.

To get to the on-line help for `q_series` commands, type the command name from the command line of your terminal (currently this is written in Japanese).

In the following, we show the standard procedure of data reduction with the command names in `q_series`. Each of the commands is described in the subsequent chapter in detail. We suggest preparing for a summary of your FITS files that should be reduced with describing data types, qualities and so on. Such a summary table would be more useful than a list created by reading FITS headers of the target files. This is because most users will have a huge number of FITS files.

4.1 Getting Started: the Reduction Software

We assume that the user will reduce data either on UNIX or LINUX. As described above, you need to have

- q_series
- IRAF

Furthermore, the user should have a basic knowledge of script languages such as `awk` and `perl`.

4.1.1 q_series

Users can obtain the latest version (version 30 as of 2007 October) of q_series from <http://canadia.ir.isas.ac.jp/comics/open/rbin/rbin.html>

After getting the source codes contained in `rbin.tgz`, extract the file. You will see a directory called `rbin.030`. Please refer to the README file under `1_README` of the directory for further information of installation. In this manual, we describe all the commands in the q_series without a path. So you should tell your shell the proper path for the commands.

4.1.2 IRAF

Please refer to the other documents or/and your local experts for installing IRAF.

4.2 Which Data Should I Work On?

Before starting data reduction, the user should properly associate data types (e.g., target, standard star, dark etc) with FITS file numbers. For this purpose, we suggest listing the following information that will help your reduction. In the list, of course, you should exclude FITS numbers for unnecessary data and the overhead of the observations (e.g., test exposure, etc.):

- Object name
- Image number
- Data format (i.e., clock type)
- Type of observations (Photometry or/and Spectroscopy), and filter names
- Status of the telescope (e.g., Azimuth/Zenis Angles) and COMICS (e.g., InstPA)
- Chopping parameters

Table 4.1: Summary of a Standard Procedure: Imaging

0. Data Examination	Checking clock and optics	q_headlst2
1. Making Dark	Dark per exposure	q_list_stat q_fcombine q_arith
2. Making flats for imaging (sky flat)	Subtracting dark from raw sky data Splitting images for each chopping beam Statistical operations between frames and between files Smoothing and normalization	q_arith q_bsep q_list_stat IRAF:gauss, q_arith
3. Making a mask	Finding bad pixels	q_badpix
4. The Target Images Analysis of Standard Star(s)	Subtracting chopping data Removing pattern noises Dividing with flat Shifting & Adding Measurement of counts for standard star Deriving conversion factor from count to flux density	COMQ images q_subch q_arith q_arith, q_fcombine IRAF:imshift q_photo or IRAF:imexamine

If you have difficulty in making such a list for some reason or/and are analyzing data retrieved from the archive system, you will probably need to extract such information from the FITS header of each file. Since observers may have many FITS files, it's easy to use the task `q_headlst2` in `q_series`.

A very useful feature of `q_headlst2` is that it can read an ASCII text files containing a list of FITS file names. For example, assume that your data are located under `/data1/20000720A`, type the following commands at the directory to make such an input list file as follows,

```
ls /data1/20000720A/COMA* | awk '{print substr($1,1,16),substr($1,18,12)}' > 720.lista
ls /data1/20000720A/COMQ* | awk '{print substr($1,1,16),substr($1,18,12)}' > 720.listq
```

Of course, you can make such file lists by hand. The task can read them as follows:

```
q_headlst2 @720.lista 720a.clk 720a.opt 720a.temp 720a.tel1 720a.tel2
q_headlst2 @720.listq 720q.clk 720q.opt 720q.temp 720q.tel1 720q.tel2,
```

This gives all the required header information that will be stored into the files `*clk,opt,temp,tel1`, and `tel2`.

Table 4.2: Summary of a Standard Procedure: Spectroscopy

0. Data Examination	Checking clock and optics	<code>q_headlst2</code>
1. Making a Dark Image	Dark per exposure	<code>q_list_stat</code> <code>q_fcombine</code> <code>q_arith</code>
2. Read-out Pattern Reducing noise	Removing pattern noise	<code>q_subch</code>
3. Making a flat for Spectroscopy	Subtracting (Dark) from (Dorm) Subtraction of detector pattern Smoothing and normalization, if needed	<code>q_arith</code> <code>q_subch</code> <code>IRAF:gauss, q_arith</code>
4. The Target Images Analysis of Standard Star(s)	Subtracting chopping data	COMQ images
	Dividing with Flat	<code>q_arith</code>
(a) Determination of spatial axis	Making Sky Spectra of Target and Standard Star 1) (Object or Standard Star) – (Dark) 2) If multiple images, needs to average 3) Dividing with Flat Auto-Wavelength Calibration Determination of spatial axis (Tracing stellar position within standard star) Axis determination Making a conversion table for image transform	COMA images <code>q_arith</code> <code>q_list_stat</code> <code>q_arith</code> <code>q_sky_nlow</code> (only for NL) <code>q_startrace</code>
(b) Image Transformation	Determination of Transformation Parameters Transformation of target, sky and mask images	<code>IRAF: geomap</code> <code>IRAF: geotran</code>
(c) Calibrating Wavelength and Positions	Applying Auto-Wavelength Cal. on the image(s) from the above (b) Determination of disp. axis w. images from (b) (Tracing Stellar Spectrum w. Standard Star) Verification of the Results	<code>q_sky_nlow</code> <code>q_startrace</code>
(d) Shifting & Adding	Adding images after shifting along the slit	<code>q_arith, q_fcombine</code> <code>IRAF: imshift</code>
5. Atmospheric Attenuation Correction w/Standard Stars	Extraction of spectra (Object)/(Standard Star) \times (Standard StarTemplate) Determination of Conversion Factor to Flux	<code>q_list_stat</code>
6. Correction of Atmospheric Attenuation using Standard Star	Determination of atmospheric spectrum using data and template of standard star Making 2D Images of Atmospheric Spectrum Dividing Object Image with the 2D image	<code>q_list_stat</code> <code>q_mkimg</code> <code>q_arith</code>

Chapter 5

Data Reduction: The Details

5.1 Imaging at N and Q Bands

5.1.1 Making Dark Frames

For many observations, dark data are taken either before or after scientific imaging. These calibration data should be taken with the same exposure parameters (i.e. clock configuration) for the sources of interest, standard star or/and dome flat. However, it is worth checking whether or not they are identical because the current COMICS system allows the use of different frame numbers or/and CoAdd mode. Also, many users prefer to specify CoAdd mode =0 which gives RAW data or =1 (ADD) without using chopping. In other words, making a dark image gives dark counts per exposure for many COMICS observations.

Dark images can be made by (1) averaging images that have been taken with the same clock configuration, (2) obtaining mean images using the averaged files, and (3) dividing the image by the exposure numbers. The following example indicates how to make a dark image from images of #51219 and 51221.

Image No.	Clock Parameters	Resultant Dark (*)
51219,51221	PP=30, ND=1, Nexp=163, CoAdd=1, Ystart=1, chop=yes	imgdark30_1

(*): `imgdark30_1`: Dark image for an exposure at PP=30, ND=1

The first step is to make an averaged image over the observing time, namely, over the frames. This can be done with `q_list_stat`. To use `q_list_stat`, the user must properly specify the following options: the file and detector names, pixel ranges (if you want to use all of them, use "-"), and ":" which selects averaging along time axis.

The second step is to average the resulting files `m51219` and `m51221` using the task `q_fcombine`. Keep in mind that the original COMA00051219 and 51221 images have been created with CoAdd = 1 from the 163 exposure files. We therefore have to divide the image by 163 to obtain a mean image per exposure (so that all the images have the same integration time). This can be done with the task `q_arith`.

```
q_list_stat ../DATA/COMA00051219.fits 1 - - : m51219
q_list_stat ../DATA/COMA00051221.fits 1 - - : m51221
q_fcombine m51219 m51221 ave=imgdark30
q_arith imgdark30 / 163 imgdark30_1
```

The user may have to repeat this procedure for all the types of observations to be reduced. In most cases, these are for the source of interest(s) taken in the same night, standard star(s), dome flat for spectroscopy (if exists) as well as all the data with identical parameter combinations of `img/spc`, `PP`, `ND`, and `Ystart` parameters. At this point, we suggest making a log sheet by hand for your records because you have to make many dark images with different clock configurations.

5.1.2 Making Flat Images

To make flat images, the user may select either dome-flat or sky-flat. For imaging observations, the user must use "self-sky-flat" because dome-flat cannot be used due to saturation. On the other hand, the user should use dome-flat for spectroscopy observations as sky-flat usually has severe contamination from atmospheric emission lines whose wavelengths may have shifted. In the procedure below, we suppose that the user has data available to make "sky-flat".

To make "self-sky-flat", you first have to split the images for each chopping beam. This is because you will use "off-beam sky" for the flat of the "on-beam sky" (or the vice versa) by making averaged sky images for each chopping beam.

Since COMICS focuses the incident light onto the slit, unevenness of the slit surface is being transferred into the flat images in most cases. Therefore, you have to use sky images taken with photometry mode for flat fielding, and those taken with the slit-width for each slit-viewer. In addition, if you have used different photometry filters, we suggest making sky images for each filter using an off-beam sky image that have been taken without changing filter positions. Again, this is because there are unevenness in gain of the detector pixels.

In principal, it should be possible to make flat images taken with the same filter configuration and apply it to all the data. However, in practice, this is not recommended because differences in telescope pointings cause non-negligible differences in optics. Hence, we suggest making flat images using off-beam sky image for each target image.

It is well known that the result sky flat has a residual gradient through the entire the image. The amplitude of the gradient is usually $\sim 10\%$ of the peak. If you use such a flat image, you will realize that the accuracy of the photometry is most likely worse than that of the image made with the above flat image. You can see the difference by comparing two photometry results toward an object. In one case, the object is at the center of the field of view. In the other case, the object is seen close to to the edge of the field of view. Therefore it is required to remove such a ripple by fitting the low-spatial frequency components for the use of flat fielding.

0. The following two files are selected to analyze for imaging observations.

```
../DATA/COMA00050615.fits
../DATA/COMA00050617.fits
```

1. Subtracting dark from the COMA images, then making a sky image

First, you have to know the numbers of Coadd (Nexp) of the data, which can be obtained from the header by using `Q_CHEB`. Since Nexp was 16 in this case, you must use a dark image that has been obtained from an image created by adding 16 exposures. Then, simply multiply a factor of 16 to the dark image already made for an exposure.

```
q_arith imgdark30_1 \* 16 imgdark30_16      (* indicates multiplying, not * in UNIX)
q_arith ../DATA/COMA00050615.fits - imgdark30_16 sky50615.fits
q_arith ../DATA/COMA00050617.fits - imgdark30_16 sky50617.fits
```

The above operation has subtracted dark from all the frames. This is because `q_arith` works for all the Z with the conditions: (1) $Z > 1$ for the images to be subtracted, and (2) $Z = 1$ for the image to subtract. Here we obtained images of "sky50615.fits" and "sky50617.fits" after subtraction of the dark.

2. Splitting images for each beam, then averaging each of them

The sky images at this stage contain the dual chopping beam data, which should be split beam by beam. This can be done using the task of `q_bsep` which creates images for individual beam based on the information given in the header.

```
q_bsep sky50615.fits sky50615_p.fits sky50615_n.fits
q_bsep sky50617.fits sky50617_p.fits sky50617_n.fits
```

Here "sky50615_p.fits" corresponds to the data taken with chopping beam A, and "sky50615_n.fits" with chopping beam B. You can check the results by referring to the header; the value of the parameter of NAXIS3 for "sky50615_p.fits" and "sky50615_n.fits" must be exactly half of that for "sky50615.fits".

Subsequently we have to average the individual beam files with `q_list_stat`.

```
q_list_stat sky50615_p.fits 1 - - : sky50615_pa.fits
q_list_stat sky50615_n.fits 1 - - : sky50615_na.fits
q_list_stat sky50617_p.fits 1 - - : sky50617_pa.fits
q_list_stat sky50617_n.fits 1 - - : sky50617_na.fits
```

2. Imaging smoothing with a Gaussian beam

You may have realized that the flat images that we obtained are most likely to contain a large-scale residual pattern over the image, which should be eliminated for making a better flat. We therefore convolve a Gaussian filter to the above images, and divide them. For this purpose, we use the task `gauss` in IRAF.

```
c1> gauss sky50615_na.fits[, ,1,1] sky50615_naG.fits 6
c1> gauss sky50615_pa.fits[, ,1,1] sky50615_paG.fits 6
c1> gauss sky50617_na.fits[, ,1,1] sky50617_naG.fits 6
c1> gauss sky50617_pa.fits[, ,1,1] sky50617_paG.fits 6
```

This example shows the case where we have convolved with the Gaussian beam having a FWHM of six pixels which is the best-searched value for cases, in our experience. The resultant images are sky50615_nG.fits.

Subsequently, we divide the images before Gaussian convolving by those after convolving, as follows.

```
q_arith sky50615_na.fits / sky50615_naG.fits sky50615_naF.fits
q_arith sky50615_pa.fits / sky50615_paG.fits sky50615_paF.fits
q_arith sky50617_na.fits / sky50617_naG.fits sky50617_naF.fits
q_arith sky50617_pa.fits / sky50617_paG.fits sky50617_paF.fits
```

The files of "sky50615_naF.fits" etc are the desired flat images.

5.1.3 Making a Bad-Pixel Mask

Bad Pixels on the Flat

You may have realized that there are non-negligible numbers of pixels or/and region(s) where the count values are obviously lower than those from the surrounding pixels. They are less sensitive compared with those in the surrounding area, and most likely due to dusts on the detector and/or unevenness of the gold coating of the slit. In addition, you also need to examine the resultant flat to see if there are pixels that have significantly lower counts than those for the surroundings or/and pixels being affected by artificial line(s). The tasks `q_badpix` and/or `q_val` tell you such bad pixels or/and regions to be eliminated. The former task will help you to identify bad pixels by making an image where 0 is stored for the pixels exceeding a threshold value you specify and 1 for the other pixels. For instance, if you want to identify bad pixels with sensitivity less than 80% or more than 120% of the mean of the surrounding pixels, issue the following commands:

```
q_badpix sky50615_naF.fits 0.8 1.2 sky50615_naF_bp.fits
q_badpix sky50615_paF.fits 0.8 1.2 sky50615_paF_bp.fits
q_badpix sky50617_naF.fits 0.8 1.2 sky50617_naF_bp.fits
q_badpix sky50617_paF.fits 0.8 1.2 sky50617_paF_bp.fits
```

Of course, 80% and 120% are for examples, and we advise using trial and error to find the best threshold for your data. Our experience suggests that this method should work for most cases because the flat images tend to have rather uniform values. However, if it does not work, you may try to identify bad pixels by specifying threshold values with respect to the standard deviation for the surrounding pixels rather than using the mean.

Pixels Having No Incident Photons due to Vignetting etc

The task `q_badpix` may fail to identify bad pixels located at the edge of the image area that have no incident photons. Unfortunately, since there is no general method that can work for all type of data, you should consult with your local expert or/and the support astronomer, depending upon the cases.

Making a Mask Image

With the procedure above, we assume that you have successfully obtained an image for bad pixel masking; the image has values of 1 for the identified bad pixel while 0 for the others. You can eliminate all the bad pixels from the target images by multiplying the bad pixel mask image.

5.1.4 Reduction of Science Data

When you are happy with the flat and bad-pixel masking images, the next step is to apply these to the images containing the source(s) of interest.

Subtraction of Chopping Images — COMQ Images

You need to subtract the off-beam images from the on-beam ones for the background subtraction. In practice, however, the user can simply use the COMQ images instead of making such images by hand. If you want to know image statistics for each or/and between frames or/and exposures, you should use the task `q_bsep` to extract the desired images before using `q_list_stat` because COMQ images are the summations for each beam images.

Reduction of Read-out Pattern Noise

Since the COMICS system reads the detector current from all 16 channels at the same time, the readout noise should have the same pattern noise for each of the 16 channels. Recall that each channel has 20 columns. On the other hand, because the target source(s) is usually detected over some portion of the detector, you can reduce the readout noise using the pixels where the source(s) is not detected.

For this purpose, use the task `q_subch` which can subtract the median of the input image. If your source(s) is much smaller than the image size, you can simply use all the channels with the command of

```
q_subch ../DATA/COMQ00050615.fits obj50615.fits
```

On the other hand, if your source is extending over a number of images, the above subtraction with median may fail to remove the pattern. You can avoid such cases by specifying channel number(s). For instance, if you want to use channel 1, namely $X = 1-20$ in the image, try the following,

```
q_subch ../DATA/COMQ00050615.fits 1 obj50615.fits
```

For this case, however, you will see that the "random" readout noise has increased, although the "pattern noise" has been subtracted.

Dividing with the Flat Image

The next step is to divide the chopping-subtracted images, i.e., COMQ images, with the flat. Your COMQ images should contain both the positive and negative images as the chopping has been done within the COMICS field of view for many cases.

Start with flat-fielding for the positive image. First, divide the target images with the flat made from the beam-B (the negative) images. After that, remove bad pixels at this stage, as follows,

```
q_arith obj50615.fits / sky50615_naF.fits obj50615_datP0.fits
q_arith obj50615_datP0.fits \* sky50615_naF_bp.fits obj50615_datP1.fits
```

You can divide the flat images for the negative ones as same as for the positive ones. But, be sure to multiply by -1 for the positive images. If your negative images do not appear in the COMICS field of view, you can skip this procedure.

```
q_arith obj50615.fits / sky50615_paF.fits obj50615_datN0.fits
q_arith obj50615_datN0.fits \* sky50615_naF_bp.fits tmp.fits
q_arith tmp.fits \* -1 obj50615_datN1.fits
```

We assume that you now have completed the calibration processes that are required for each image and should have the final positive images for each image file (the negative as well if both of the beams are inside the field of view).

5.1.5 Adding the Images

Adding the Images without Shifting

In principal, you should be able to add the calibrated images as long as they have been taken with the same filter configuration and their pointing centers agree with each other within the margins of the error. You can probably add the images that have been acquired sequentially within a short time and/or your source(s) cannot be recognized in each images.

Your images can be added using the task `q_arith`. In many cases, however, you must issue the command several times, which is not always useful. Instead, you can use `q_fcombine`, which is originally designed for averaging images. To use the task `q_fcombine`, you have to supply an input ASCII text file where the names of images are written in each line. The following example shows how to utilize the task by supplying the `objlist` file that describes names of the files to be added:

```
q_fcombine @objlist ave=objave.fits med=objmed.fits sig=objsig.fits
```

which gives you the averaged image (`objave.fits`), the median one (`objmed.fits`), and the standard deviation one (`objsig.fits`).

If you want to keep the angular resolution (i.e., sharpness) of the original data, we advise you not to add the image files that have been taken over a long observing time. This is simply because your source positions when measured with the detector coordinate have shifted gradually during the observations. If your source(s) has sufficient S/N (and ideally is compact) enough to measure its positions for each file, we suggest shifting the images before adding.

Shift-and-Add the Images

If your source(s) can be seen in different pixel positions or/and if you wish to increase S/N of the source by concatenating both the positive and negative images, you should consider to shifting the images before adding them. In practice, you may succeed in obtaining the positional offset within an accuracy of ~ 0.1 pixel by comparing the profiles with eye. However, there is a more precise method in the following.

1. Getting an offset for image shifting

In order to add the fully calibrated images, you need to measure positional offsets between images. If you have a bright compact source(s) in your field of view, you can use it for a positional reference to calculate offsets using the task `q_photo`. If you don't have a compact source but only an extended emission in your field of view, you cannot take this approach. Another method is to compare spatially "averaged" profiles of the emission projected along the X and Y axes for the images taken with different filters or/and positions. These profiles should be calculated for a rectangular region which includes the target source in each image (see Figure 5.1.5). Comparing cross-correlations of the spatial profiles produced from the images to be added, one should be able to determine the desired positional shifts for adding. Figure 5.1.5 illustrates how this method works. Keep in mind that you must exclude columns and/or rows that contain bad pixels before calculating the cross-correlation.

You have to repeat this process for all the images that you want to add. For the sample images 50615 and 50617, we have obtained the shifts as follows,

```
obj50615_datP1.fits    0    0  Reference
obj50615_datN1.fits  -106.5 +0.8
obj5061_datP1.fits    -0.2 +0.8
obj5061_datN1.fits    +106.5 +0.8
```

2. Shifting images and obtaining a mean

Once obtained positional shifts for all the files, you can add them by supplying the shifts. To shift images, we use task `imshift`.

```
c1> imshift obj50615_datP1.fits[, ,1,1] obj50615_datP1s.fits 0 0
c1> imshift obj50615_datN1.fits[, ,1,1] obj50615_datN1s.fits -106.5 0.8
c1> imshift obj50617_datP1.fits[, ,1,1] obj50617_datP1s.fits -0.2 0.8
c1> imshift obj50617_datN1.fits[, ,1,1] obj50617_datN1s.fits -106.5 0.8
```

It should be noted that the first line above is to make a template image for the shifted images, although the given values are zero. The purpose for this is to define dimensions (i.e., pixel numbers) for the resulting image.

Subsequently, you can average the images. Again, if your images still have non-negligible numbers of bad pixels, you should use the median averaging instead of the simple averaging. Consider using the task `q_fcombine` for both methods. In practice, it is wise to provide an ASCII text file that describes names of the images to be concatenated in each line. For example, suppose that your list file has name of "objlist", type

```
q_fcombine @objlist ave=objave.fits med=objmed.fits sig=objsig.fits
```

This should give you the averaged image (objave.fits), the median one (objmed.fits) and the standard deviation one (objsig.fits).

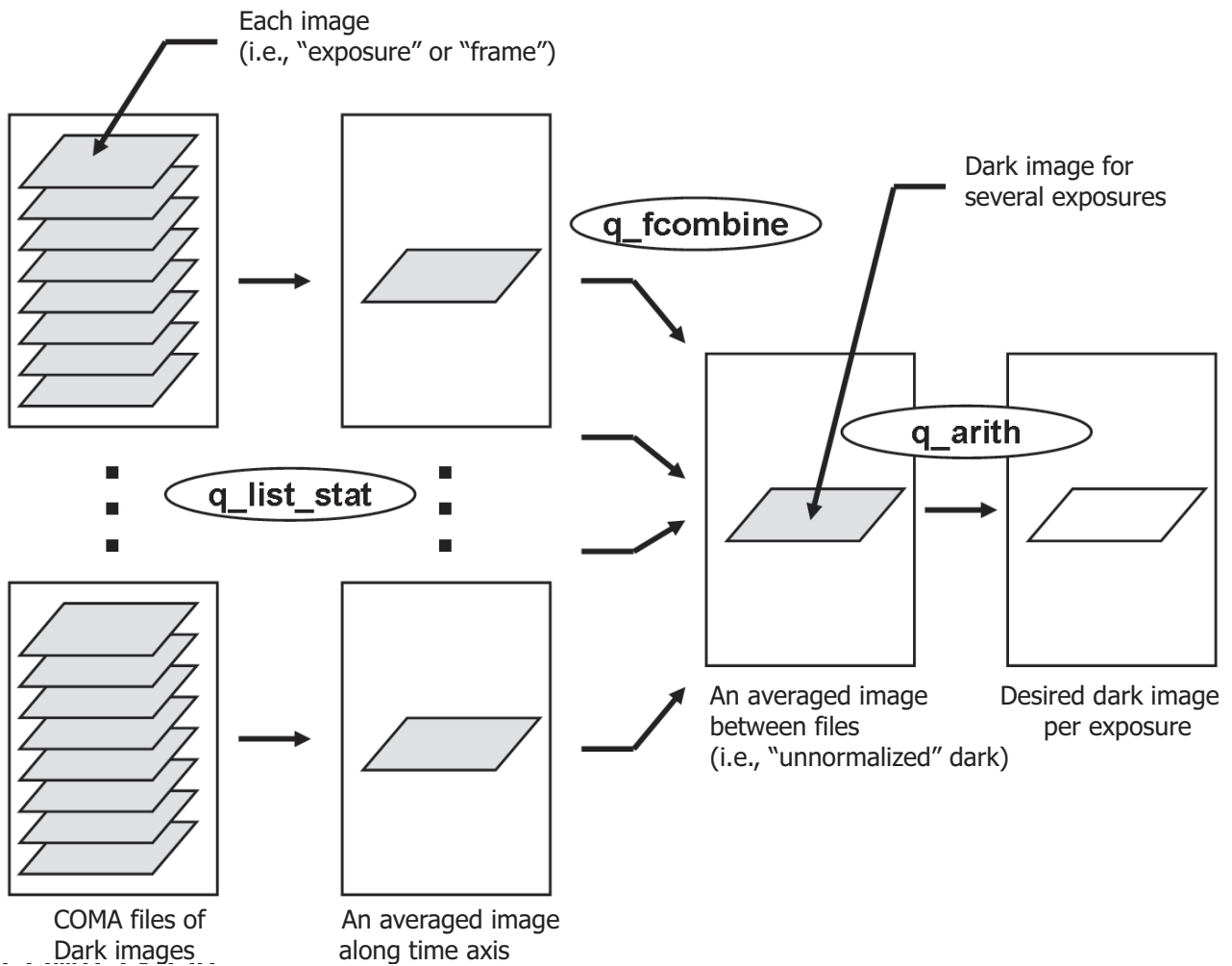


Figure 5.1: A typical procedure for making a dark image requires the user to average all the dark images taken with the same clock parameters, and to convert the averaged images into a dark image per exposure.

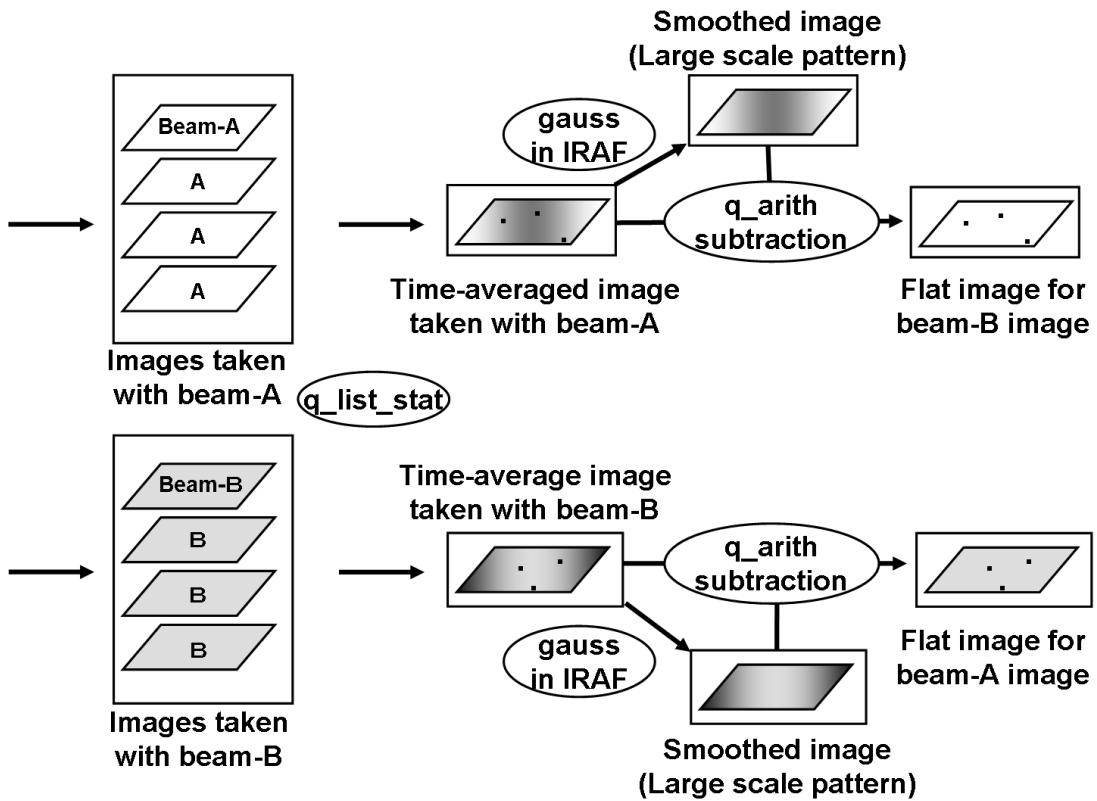
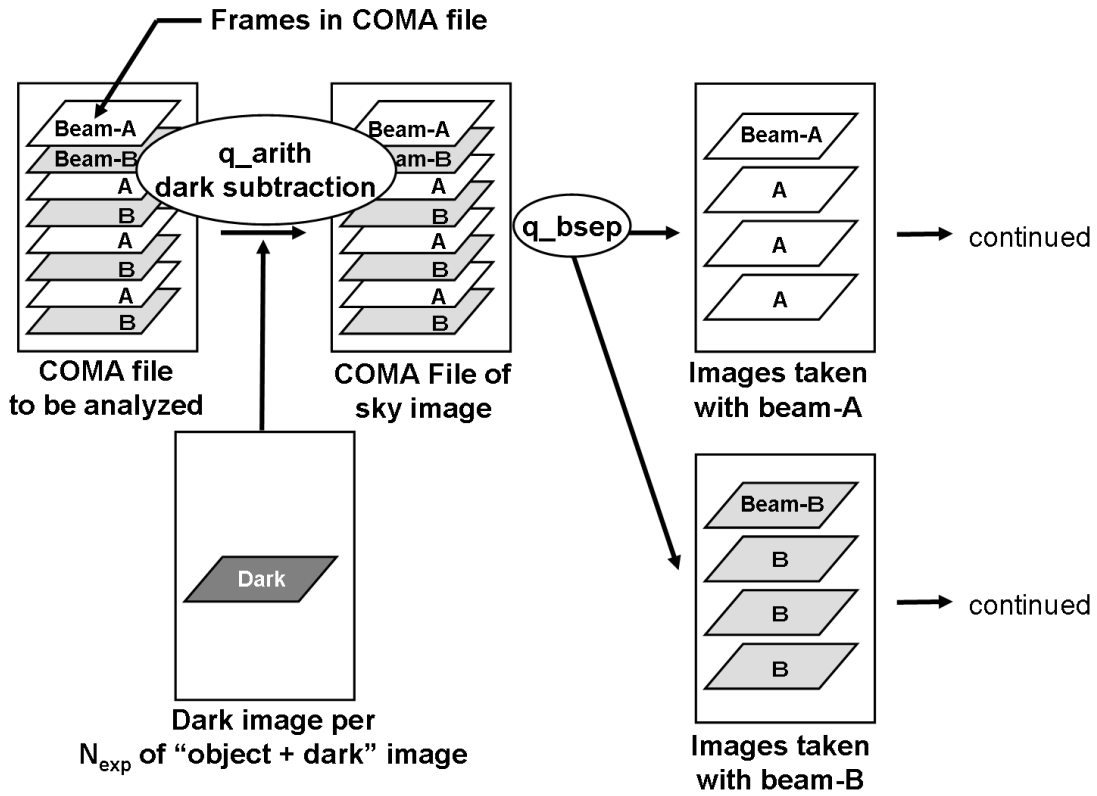


Figure 5.2: Standard procedure for making a sky flat image. In this case, the user can get the desire flat image(s) after subtracting the dark image, beam splitting, averaging, and removing any large-scale residual pattern. Notice that a flat image created from the one-beam data can be used sky flat for the data taken with the other beam.

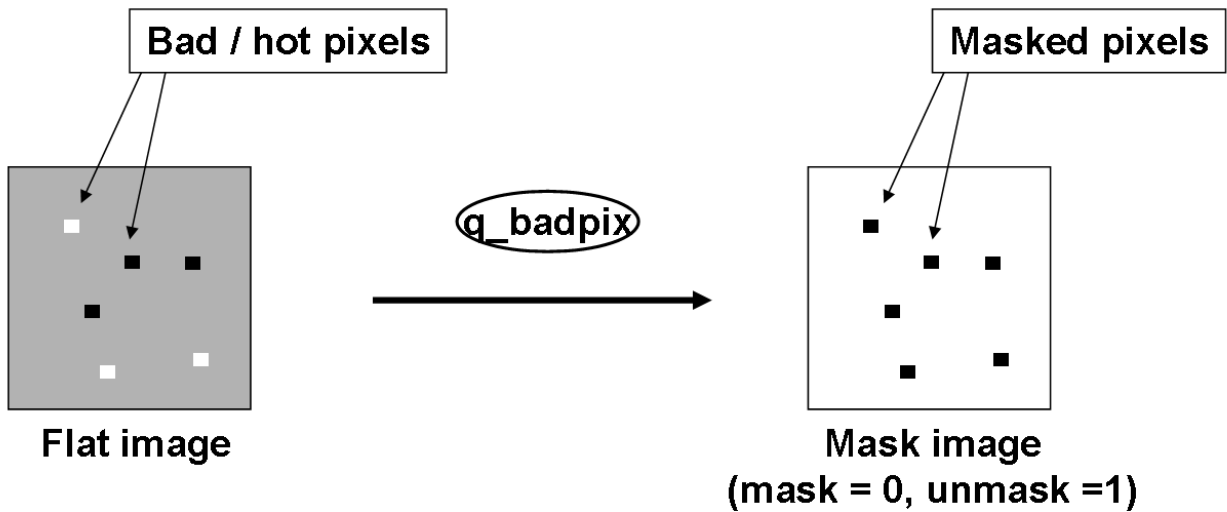


Figure 5.3: Making a mask image with task `q_badpix`: The task produces a mask image where the bad or/and hot pixels whose sensitivity are high or low compared with those for the surroundings. The user can specify threshold values to identify unusable pixels; the resultant mask image has values of 0 for the bad pixels, and 1 for the usable ones.

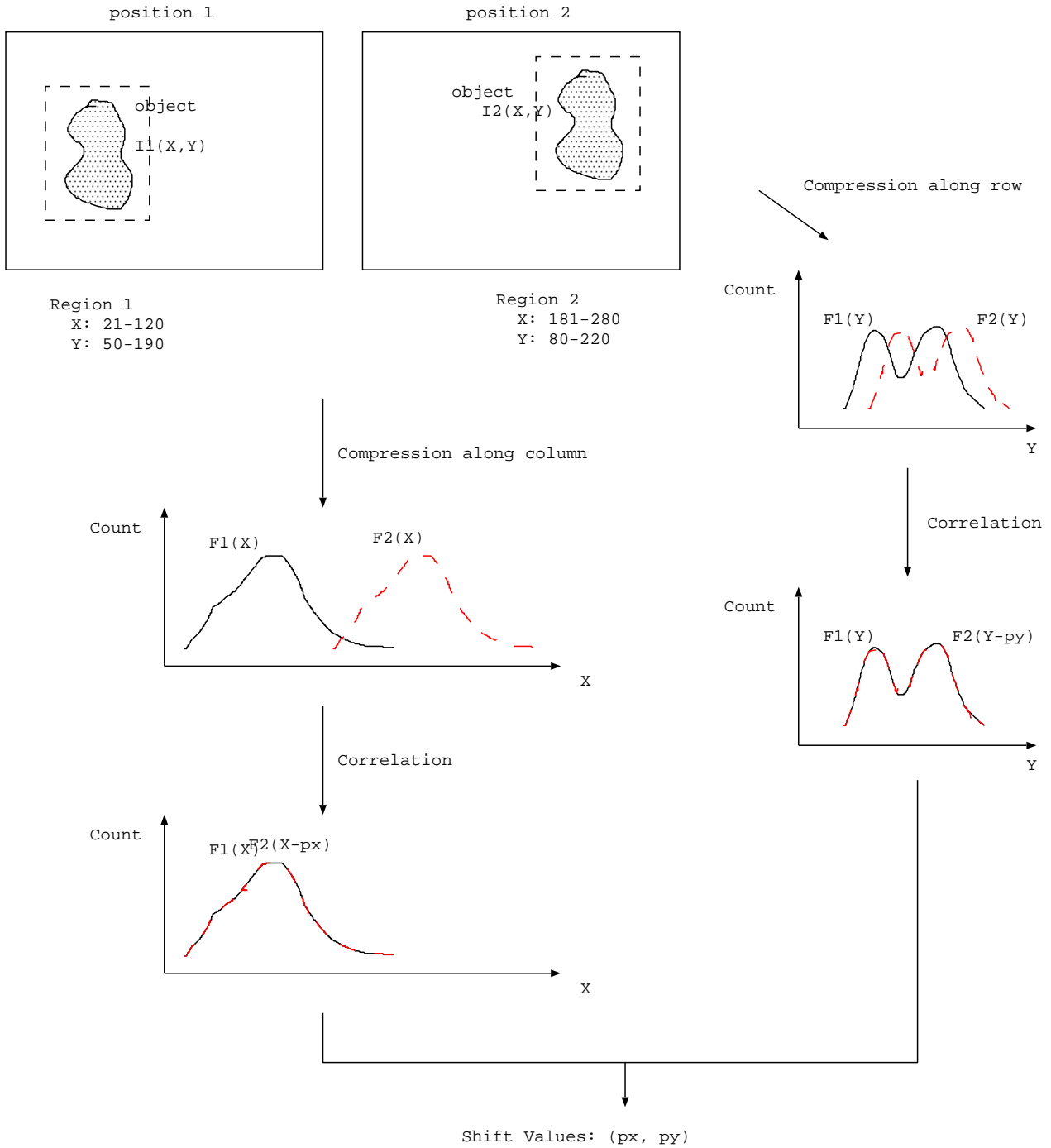


Figure 5.4: A method to add images for the extended emission

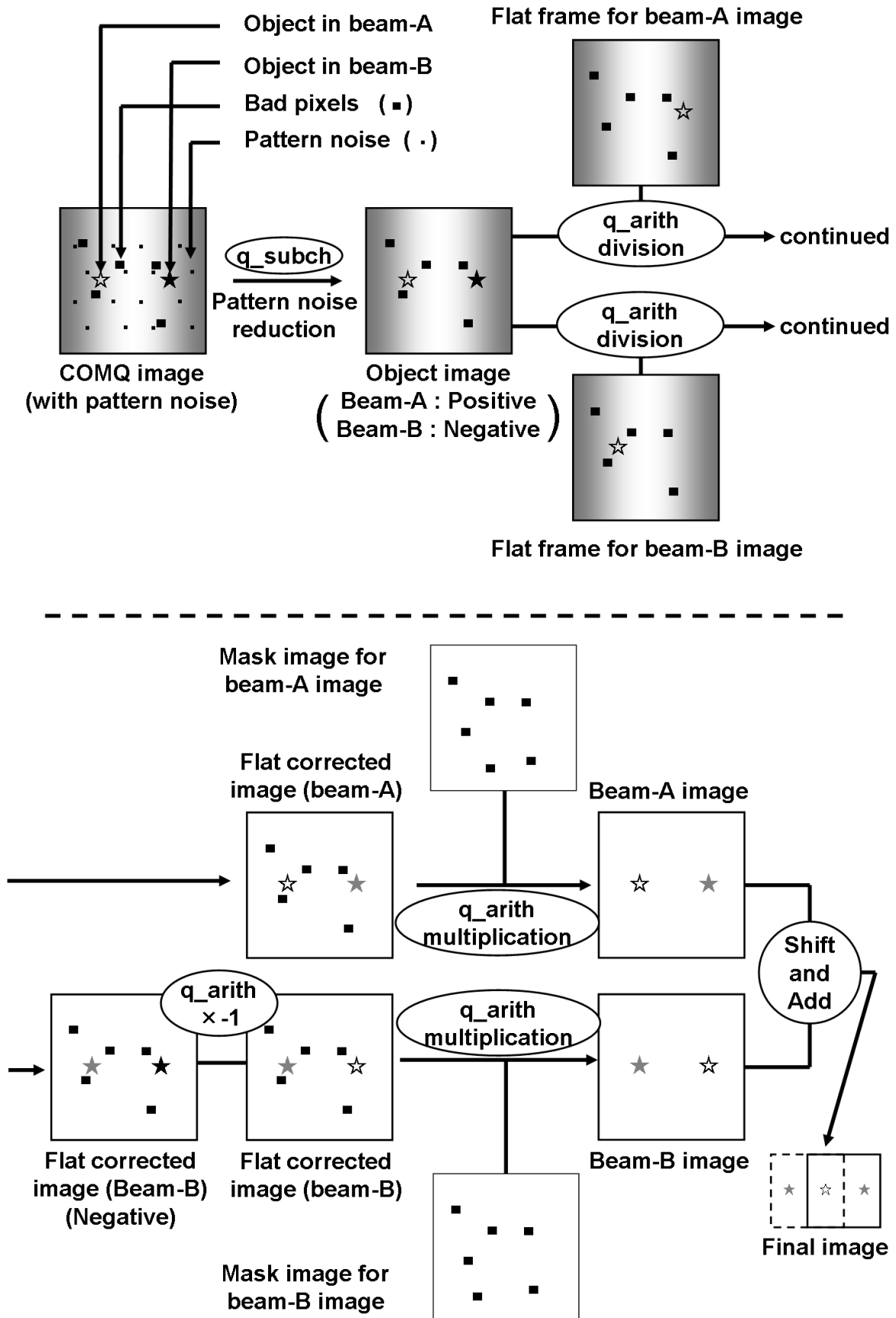


Figure 5.5: A method of photometry data reduction. The final images can be obtained by correcting for the flat to the COMQ images, masking bad pixels, and adding all the images by shifting them (if necessary).

5.1.6 Miscellaneous

Correcting for Sky Levels

We suggest verifying the quality of sky-level subtraction carefully because, in our experience, gradual level changes over the images tend to remain. This would be due to high background emission, for example, and/or the presence of clouds even for a blank sky. If such a residual pattern exists, you should consider applying another sky subtraction by making sky images to remove it.

Conversion of the Image Intensity Unit: Getting the Flux Density Scale

After completing the final image construction, you probably want to convert units of the image from the counts to the brightness using standard star(s).

First, using the data base provided by Dr. Cohen and colleagues, you have to compute the brightness of the standard star at the filter wavelength you observed. Select standard stars which have been observed with an airmass close to that of your target. Since accurate measurements of flux densities of the standard sources have not been generally well-established at MIR bands, except for the N-band, you will need to obtain fluxes for the desired filters by integrating spectral data provided in the Cohen et al. paper over the desired wavelength range. For details of the filter properties, such as the center wavelength, and the passband characteristics, please refer to the COMICS web page at <http://canadia.ir.isas.jaxa.jp/comics/open/guide/index.html> where you can find other useful links such as the data base by Dr. Cohen et al.

For simplicity in your calculations, you may integrate the spectrum over the filter bandpass adopting a rectangular shape, as the first order of approximation. However, you should calculate the flux by considering all the parameters, e.g., the bandpass curve of the filter, that depend on the wavelength.

Second, the user should know the count of the standard star(s). You can obtain counts by calculating summation of the counts over a rectangular area containing the standard star, for example. This can be done with the task `q_photo`, or photometry tasks in IRAF. Now you should be able to obtain a conversion factor from the count unit to flux density through comparison between the obtained counts and calculated flux density. To convert source images into flux density units, you can simply multiply the calculated conversion factor to the fully calibrated images. If your source is extended, you may want to convert the images into brightness temperature (T_b) scale. This conversion can be done by dividing the image with a pixel field of view. Taking photometry observations as an example, you can obtain the images with brightness temperature scale by dividing $0.0166 \text{ arcsec}^2 \text{ pixel}^{-1}$.

Adding Images Taken with the Different Filters

If you want to make a color image using multiple images taken with different filters, you should consider how to properly register these image centers across filters. One method is

to refer to the point source(s) commonly seen in the multi-wavelength images. If this does not help, you may have to employ the shift-and-add method described in §5.1.5. However, you should be careful when using the latter method. Make sure of the validity of the image registrations across the filters by considering the nature of the source(s) (i.e., emission mechanism). Moreover, we suggest making sure possible position shifts across the images taken at different wavelengths and/or different observing times (as telescope tracking accuracy may have changed during the observations).

Estimate of Image Noise Levels

Given that fluctuation of the incident counts at MIR wavelengths is governed by the sky background emission, which is usually brighter than the target sources, the most straightforward and most reliable method to estimate image noise level is measuring the standard deviation for the blank sky in the target images rather than measuring standard deviation of the target sources across the images. We believe that this approach would be valid for photometry observations as well, where high accuracy is generally required. Of course, the above method works only if your final images do not have a residual global pattern of the sky background AND the sky background level is considerably brighter than the source(s) of the interest.

5.2 Analyzing Spectroscopy (N-band Low Resolution; NL) Data

Figure 5.6. shows overall flow of the data reduction process for a point source taken in low-resolution mode. The basic idea of the data reduction can also be applied to the other spectroscopy mode observations.

5.2.1 Preparing for a Dark Frame

For reducing spectroscopic data, we have to use not only the science data but also the dome flat images. For this purpose, the user must prepare for dark images that have been obtained with exposure parameters (e.g., clock configurations) identical to that for scientific observations. This can be done with the same way described for the imaging mode (see § 5.1.1 for the details).

Assume that you have dark image files of COMA00051862.fits (Nexp=98) for the flat, and those of COMA00051872.fits (Nexp=32) for the target sources.

```
q_list_stat ../DATA/COMA00051862.fits 1 - - : m51862
q_arith m51862 / 98 spcdark50_1          dark images for the flat
q_list_stat ../DATA/COMA00051872.fits 1 - - : m51872
q_arith m51872 / 32 spcdark150_1       dark images for the target source(s)
```

It should be again noted that we have used the dark data taken with the same exposure parameters. For many cases, the users can use dark images that were taken for the flux measurements. If you don't have such a data set or/and you have only a dark image obtained from all the detectors, you probably have to analyze data detector by detector (see §2.1 as well).

To see the contents — detector information — of each FITS file, check the Q_DETST parameter in the header. This key word is given by a number consisting of 0 and/or 1. The first digit indicates a flag of the imaging or photometry detectors, and the subsequent ones give status of the spectroscopy detectors. For example, 100110 indicates that you have obtained a photometry image plus spectroscopy taken with the detectors No.3 and 4. Notice that the designation number of the spectroscopy detectors and the 4th axis, i.e., the w-axis, in the FITS files differ from each other. In this example, w=1 corresponds to the spectroscopy detector No.3 and w=2 does No.4.

5.2.2 Making a Flat Frame

For spectroscopy observations, use dome flat images for making a flat. We suppose that you have obtained more than one flat image at each grating position. We strongly suggest using the dome flat image(s) that have been taken immediately before or/and after the source integration. If you don't have such calibration exposure, use those taken as close as possible

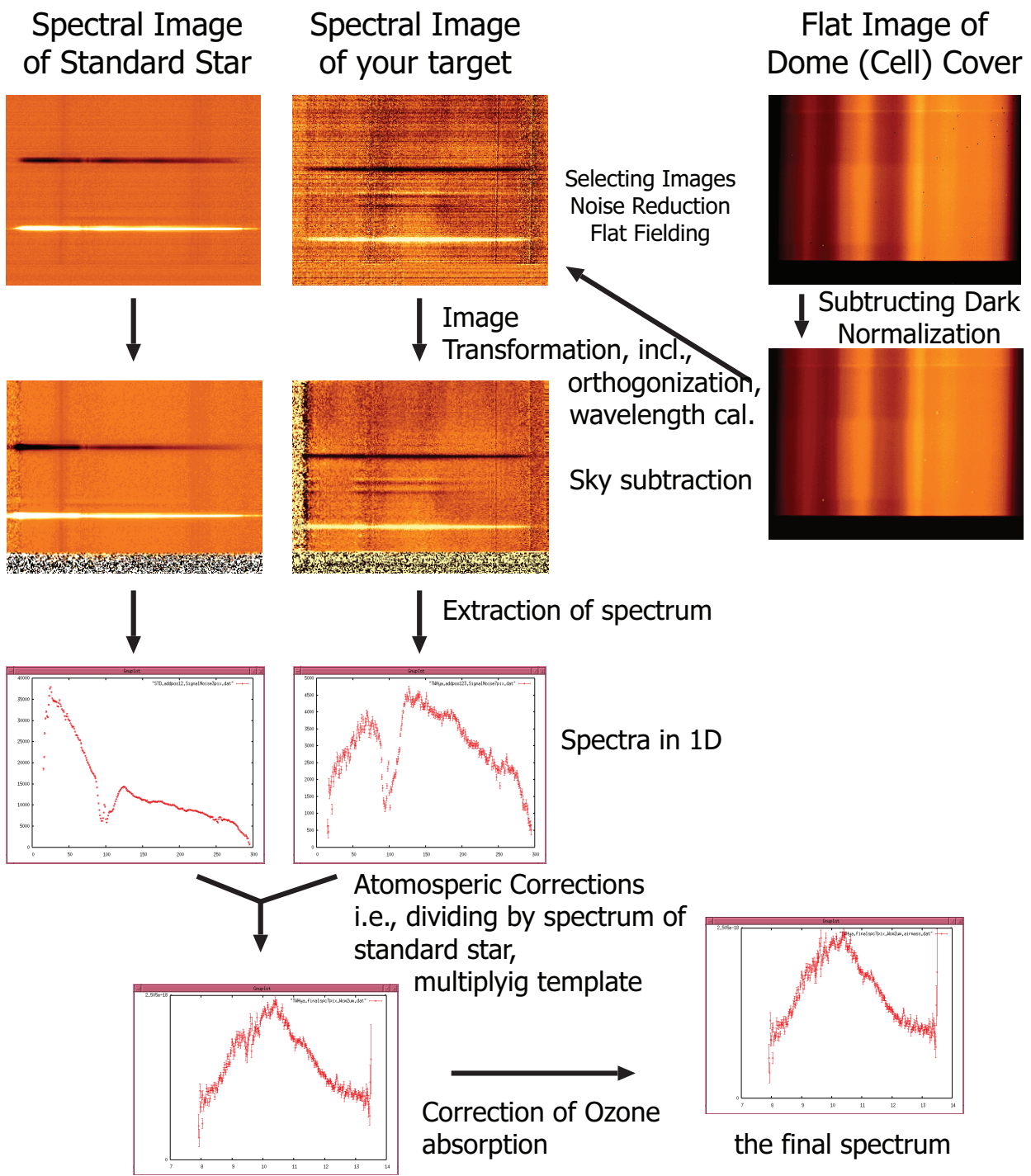


Figure 5.6: A method to reduce COMICS spectroscopy data. An example for a point-like source with the low-resolution mode

to the scientific exposure because flat images usually change even with a slight shift of grating configurations. In this example, first average a COMA image of COMA00051356.fits and subtract the dark. Since the flat image has $N_{exp}=9$, we have to multiply by 9 before the subtraction as shown in the below.

```
q_list_stat ../DATA/COMA00051356.fits 1 - - : flat51356.fits
q_arith spcdark50_1 \* 9 spcdark50_9
q_arith flat51356.fits - spcdark50_9 flat51356_d.fits
```

You will realize that the resulting dome flat contains an un-removed gradient over the field of view, which is similar to those seen in the flat for the photometry. Therefore, you need to divide the image with that is convolved with a Gaussian filter with the same method as we did in §5.1.2.

```
cl> gauss flat51356_d.fits[, ,1,1] flat51356_dG.fits 20
```

This will give you an image convolved with a Gaussian with FWHM of 20 pixels. The width of 20 pixels is recommended, based on our experience.

```
q_arith flat51356_d.fits / flat51356_dG.fits flat51356_dF.fits
```

Now we assume that you have successfully obtained a flat image of at51356_dF.fits.

5.2.3 Reduction of the Read-out Pattern Noise

For some observing modes that do not use all the detectors, such as medium-resolution spectroscopy at the N band, users are allowed to reduce the readout pattern noise by subtracting the readout from the reference detector. This is because one can use read-out from the unused detector. This will allow you to reduce the readout pattern noise by subtracting the readout from the reference detector.

If you wish to attain high accuracy subtraction, you may try to subtract a median pattern image calculated from the reference detector channels that did not contain significant noise. This is because each detector consists of 16 channels with $x=20$ pixels and all the channels are being read at the same time.

To obtain the median for all the channels, you can use the task `q_subch`. For this purpose, you should firstly extract a detector image from the reference image of read-out pattern noise. For instance, if you want to analyze a source spectral image of COMQ00051308.fits (= standard star), use

```
q_list_stat ../DATA/COMQ00051308.fits 2 - - 1 chnoize.fits
```

The next is to calculate median image over the 16 channels from the reference image of the read-out pattern noise, using

```
q_subch chnoize.fits REF chnoize_REF.fits
```

After that, subtract the noise pattern image of `chnoize_REF.fits` from the spectroscopy data as follows

```
q_list_stat ../DATA/COMQ00051308.fits 1 - - 1 obj51308.fits
q_arith obj51308.fits - chnoize_REF.fits obj51308s.fits
```

which will probably help to reduce the noise levels significantly — hopefully down to the 30 – 40% level.

5.2.4 Data Reduction of Scientific Images

You have completed the noise reduction processes across channels to proceed flat fielding. Now you are ready to reduce the scientific images you have completed to make flat image(s). You can subsequently apply these flat images to the source images.

Adding All the Images

If part of your data were acquired at the same position in the sky, you may consider adding all the appropriate images, which makes in much simpler for further data reduction. If this is not the case, we suggest keeping them without adding at this stage. For example, assume that the two spectral images files of `COMQ00051314.fits` and `COMQ00051316.fits` have been taken at the same positions in the plane of the sky and that you have obtained `obj51314s.fits` and `obj51316s.fits` after applying the read-out pattern noise reduction described in §5.2.3. The following command will collapse the two images into an image.

```
q_fcombine obj51314s.fits obj51316s.fits ave=objspc.fits
```

Dividing the Flat

The next step is to divide the added images (i.e., `objspc.fits`) (or the read-out pattern noise-reduced single image, e.g., `obj51308s.fits` in §5.2.3, depending on your reduction process) by the dome flat of `at51356_dF.fits`. Taking the latter case as an example, use

```
q_arith obj51308s.fits / flat51356_dF.fits obj51308s_f.fits
```

Orthogonization of the Spectroscopic Images and Wavelength Calibration

After completing reduction of the read-out pattern noise, you have to transform the images to orthogonize dispersion and spatial axes, which will be used to extract spectra of the emission you desire. Furthermore, you normally calibrate the absolute scale of wavelength at the same time by finding pixel-wavelength correspondences. This subsection describes how such dispersion and spatial axes are determined (see Figure 2.4).

1. Determination of Dispersion Axis

The dispersion axis is usually determined through comparison to an atmospheric sky emission image. The correction equation for the wavelength can be written as $\lambda[\mu\text{m}] = Ax[\text{pix}] + B$ for y . Since the coefficients of A and B vary with y as shown in Figure 2.4, find their y -dependency as $A = a_0y + a_1$ and $B = a_2y + a_3$. Therefore, a set of the parameters of a_0 , a_1 , a_2 , and a_3 represent the desired correction equation.

As the first order of approximation, a wavelength calibration equation can be expressed by $\lambda[\mu\text{m}] = Ax[\text{pix}] + B$ for y . However, since the coefficient of A and B varies with y for many cases, you have to determine their dependencies on y as $A = a_0y + a_1$ and $B = a_2y + a_3$, respectively.

For medium resolution spectroscopy, we suggest splitting data files for each detector to make further reduction processes simpler. In practice, first make a sky image that can be obtained by averaging images of standard star(s) or/and COMA images of scientific images (you have to subtract the appropriate dark image and divide it with the flat one). You can now make a sky image from spectral image of the standard star (COMA00051308.fits).

```
q_list_stat ../DATA/COMA00051308.fits - - - : skyimage.fits
q_arith spcdark150_1 \* 3 spcdark150_3 (Making a dark image
adjusting with Nexp)
q_arith skyimage.fits - spcdark150_3 skyimage_d.fits (Subtracting the dark)
q_list_stat skyimage_d.fits 1 - - - skyimage_d1.fits
q_arith skyimage_d1.fits / flat51356_dF.fits skyimage_d1f.fits (Dividing with flat)
```

This dispersion axis can be applied to the obtained parameters to the other data sets, unless the grating was shifted during observations, because such a calibration curve does not change for many cases. However, users are advised to double-check wavelength calibration parameters obtained from each object or/and those from the mean images (if you have sequentially taken the same configuration images.). On the other hand, if you have grating(s) that may have shifted during observations, we suggest determining calibration parameters for each grating position(s) because repeatability of grating parameters is not completely guaranteed.

As for the low-resolution observations at the N band, the dispersion axis can be obtained using the task `q_sky_nlow`.

```
q_sky_nlow skyimage_d1f.fits 1 - default 1 2 > skyimage_d1f.res
```

This command calculates cross-correlation with the database of the atmospheric emission lines at 20 points in the y axis between $y = 30$ and 220, then guessing a wavelength and pixel correspondence $\lambda[\mu\text{m}] = Ax[\text{pix}] + B$. The identical algorithm may be applied for the other wavelength observations, however, we have not developed such a command for them.

The resulting fitting parameters are stored in `skyimage_d1f.res` and can be checked with standard graphic packages such as GNU PLOT. In `skyimage_d1f.res`, the third column indicates the extracted y -axis value, and sixth and eighth column the best-fit values of A and B ,

respectively. The fitting result at $y = 30$ may have failed because of the photons at $y = 30$ are more likely to be vignetted. If you find such result, eliminate them manually from the fitting data.

Subsequently, apply a linear function to the fitted parameters of A and B by least-square fitting, namely, use fitting equations of $A = a_0y + a_1$ and $B = a_2y + a_3$ (see Figure 5.7). In conclusion, use a parameter set of a_0 , a_1 , a_2 and a_3 to repeat the best-estimated wavelength constant axis.

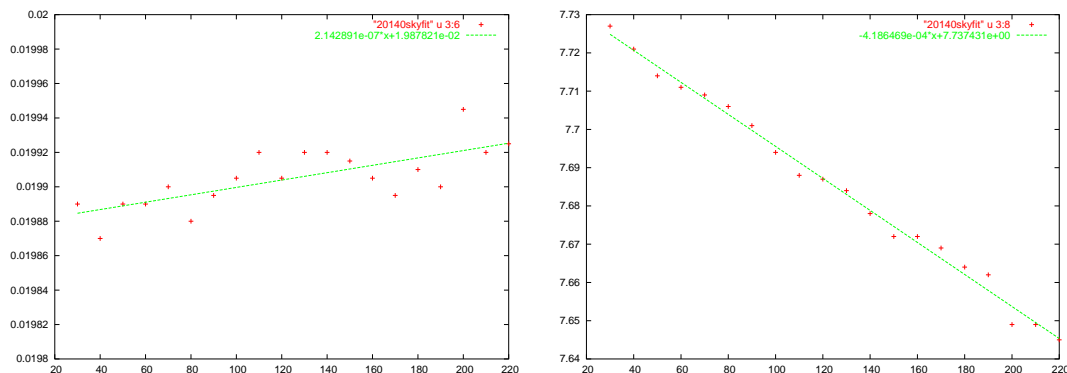


Figure 5.7: A result of wavelength and pixel relationship determination and a plot of linear function fitting

2. Determination of Spatial Axis

To determine the spatial axis, use images of standard star(s) (or any point-source image should work as well) that have had detector response subtracted and divided with a flat. You have to find the peak position of the star in the x -axis, then fit it with equation of $y = b_0(x - b_1)^2 + b_2$ to find b_0 , b_1 , and b_2 . Since the parameter set of b_0 , b_1 and b_2 is thought to be stable during an observation, you can use a point source image that has a high S/N or/and an image that has been obtained by averaging several images without any positional shifts. In this manual, we use the reduced standard star image of `obj51308s_f.fits` using the task `q_startrace`.

Assume that the image contains the standard star at the positions of $y = 62 - 92$, use

```
q_startrace obj51308s_f.fits 1 30-290 62:92 1 > obj51308s_f.str
```

this searches peak positions and stores the results in `obj51308s_f.str`. If the task fails to find peak positions (i.e., failed to converge) or it takes a quite long time to converge, some tips in the FAQ may help you. The second column of `obj51308s_f.str` stores the x -coordinate and the tenth y of the peak, the user should plot them to verify the results (please use GNU PLOT etc); see the example shown in Figure 5.8. In some wavelength ranges such as

the O₃ bands where very intense emission lines are normally seen, the task may fail to make a fit. If this is the case, we suggest excluding such points before fitting. For the medium resolution observations, the procedure above should be repeated for all the detectors.

The fitting can be done with a quadratic expression of $y = b_0(x - b_1)^2 + b_2$ as shown in Figure 5.8. The selection of the quadratic expression is due to the structure of the COMICS optics. In the case of medium resolution spectroscopy, one can approximate distortion as a linear equation because the spectrum spread over the five detectors. If you try to fit with the 2nd-order equation, you would not be able to get a reasonable result for the detectors at the edge, where intensity of the incident wave usually decreases. We therefore suggest fitting with a linear equation for the medium resolution mode. The resultant coefficients of b_0 , b_1 , and b_2 are the desired parameters for describing the spatial axis.

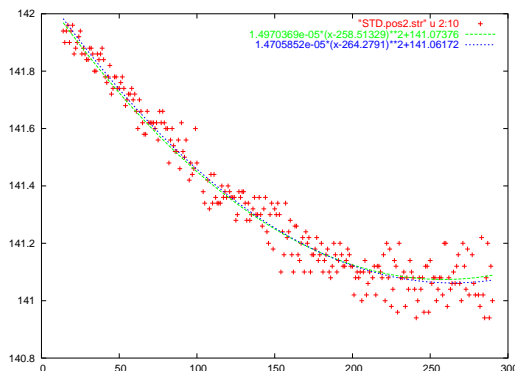


Figure 5.8: An example of a space-constant line and the fitting results

3. Image Transformation

Using the above results, you can transform the images with IRAF. The task `geomap` finds the transformation parameters and the task `geotran` transforms your image. First, produce an image that will be transformed with `geotran` as follows,

```
q_transtable2 a0 a1 a2 a3 b0 b1 trans.dat aa bb
```

where a_0 , a_1 , a_2 , a_3 , b_0 , and b_1 are the parameters obtained in the above, `trans.dat` is a name for the output image, and aa and bb define a wavelength-pixel relationship ($\lambda[\mu\text{m}] = aa \times x[\text{pix}] + bb$) after the transformation. In principal, aa and bb can be arbitrarily defined. However, you should use the same parameters obtained from the λ -pixel relation of the target. This is because one can minimize loss of the information after the transformation by reducing the numbers of pixels that are transformed outside the field of view. In the above example, we obtained the following parameters for the medium resolution at the N band.

You should also set the parameters for geotran as follows

I R A F

Image Reduction and Analysis Facility

PACKAGE = immatch

TASK = geotran

```
input      =      Input data
output     =      Output data
database=      Name of GEOMAP database file
transfor=      Names of coordinate transforms in database file
(geometr=      geometric) Transformation type (linear,geometric)
(xin      =      INDEF) X origin of input frame in pixels
(yin      =      INDEF) Y origin of input frame in pixels
(xshift   =      INDEF) X origin shift in pixels
(yshift   =      INDEF) Y origin shift in pixels
(xout     =      INDEF) X origin of output frame in reference units
(yout     =      INDEF) Y origin of output frame in reference units
(xmag     =      INDEF) X scale of input picture in pixels per reference
(ymag     =      INDEF) Y scale of input picture in pixels per reference
(xrotati=      INDEF) X axis rotation in degrees
(yrotati=      INDEF) Y axis rotation in degrees
(xmin     =      INDEF) Minimum reference x value of output picture
(xmax     =      INDEF) Maximum reference x value of output picture
(ymin     =      INDEF) Minimum reference y value of output picture
(ymax     =      INDEF) Maximum reference y value of output picture
(xscale   =      1.) X scale of output picture in reference units per
(yscale   =      1.) Y scale of output picture in reference units per
(ncols    =      INDEF) Number of columns in the output picture
(nlines   =      INDEF) Number of lines in the output picture
(xsample=      1.) Coordinate surface sampling interval in x
(ysample=      1.) Coordinate surface sampling interval in y
(interpo=      linear) Interpolant
(boundar=      nearest) Boundary extension (nearest,constant,reflect,wra
(constan=      0.) Constant boundary extension
(fluxcon=      yes) Preserve image flux?
(nxblock=      512) X dimension of working block size in pixels
(nyblock=      512) Y dimension of working block size in pixels
(verbose=      yes) Print messages about the progress of the task
(mode     =      ql)
```

geomap can be executed as follows

```
im> geomap trans.dat transpar 1 320 1 240
```


The command above has prepared for the image transformation by setting the parameters. You need the following: image files of the targets, standard star(s) that must be subtracted the detector response and be divided by the flat, sky image (skyimg) that has been used for the wavelength calibration, and the masking image. The sky image will be used to calibrate wavelength after image transformation, and the masking image to eliminate the effect of bad pixels in the transformed image. Since IRAF accepts input files with an extension of ".fits" only, you have to give it explicitly. In the example, we issued the command as follows,

```
im> geotran obj51308s_f.fits[1:320,1:240,1,1] obj51308s_ft.fits transpar trans.dat
```

Here, the first and second input parameters can be also written with the form of *@filelist*.

4. Verification of the results and determination of the wavelength calibration equation

After image transformation, one should verify the results (i.e., the spatial axis should be horizontal and dispersion one vertical) using the tasks `q_startrace` and `q_sky_nlow`. Notice that the output file of `geotran` is a 3D cube, whereas `q_startrace` and `q_sky_nlow` deal with the 4D image only. We therefore convert the `geotran` output files into a 4D image.

Next, check the results on the spatial axis. For this purpose, we should multiply `q_startrace` to the transformed image (`obj51308s_ft.fits`) and plot the results (using e.g., GNU PLOT) as shown in Figure 5.8 where the peak position of the signal should be constant regardless of the *x*-value. In practice, one can do such a procedure using the following commands:

```
q_chgaxis 4 obj51308s_ft.fits obj51308s_ft4.fits
q_startrace obj51308s_ft4.fits 1 30-290 62:92 1 > obj51308s_ft4.res
(Verify the plot of obj51308s_ft4.res)
```

The first line converts `obj51308s_ft.fits` to a 4D image, and the second one is the command to multiply on `q_startrace`.

The next step is to check the results of the dispersion axis, which can be done by transforming the sky image (`skyimage_d1f.fits`) and applying `q_sky_nlow` to the results. Make sure the values of A and B are constant with respect to the *y*-axis.

```
im> geotran skyimage_d1f.fits[1:320,1:240,1,1] skyimage_d1ft.fits transpar trans.dat
q_chgaxis 4 skyimage_d1ft.fits skyimage_d1ft4.fits
q_sky_nlow skyimage_d1ft4.fits 1 - default 1 2 > skyimage_d1ft4.res
(Verify the plots of skyimage_d1ft4.res)
```

Now we can determine the final version of the coefficients for the wavelength-calibration equation. (i.e., the input parameters of *aa* and *bb* to the `q_transtable2` command) using the results from `skyimage_d1ft4.res`.

Further subtraction of the residual pattern

It is best if all the sky level variations could be removed using chopping. However, in practice, chopping frequency tends to be slower than sky level variations. If this is the case, you should see some residual pattern in the chopping-subtracted images. Since you have completed the process of orthogonizing the images (i.e., x -axis: wavelength, and y -axis: space), you should be able to estimate such a low-level residual sky pattern for the emission-free region(s) to apply a further correction. For this purpose, we should remove the median value with respect to the y -axis (i.e., dispersion axis), which can be done with `q_submedrow`, as follows:

```
q_submedrow obj51308s_ft.fits 1 320 150 240 obj51308s_ftc.fits
```

which gives the final chopping image corrected for the residual sky pattern noise.

5.2.5 Extracting the Spectra

Now you can extract the desired spectra from the obtained 2D data. You may want to use `try-and-error` to find an appropriate aperture width (e.g., 3, 5, 7, and 9 pixels) along the spatial axis to maximize the S/N of the spectrum. Here, you can measure the noise level through standard deviation of the emission-free region (i.e., the blank sky region). For instance, suppose that you have an image which accommodates the source at $y = 72$. In this case, you can estimate the noise level for the blank region of e.g., $y=100 - 130$.

```
q_list_stat obj51308s_ftc.fits 1 15-295 110:130 1 >! obj51308s_ftc.noise1pix
```

In this example, the sixth column of `obj51308s_ftc.noise1pix` indicates the noise level for each x -value which corresponds to the wavelength. Next, extract spectra of the source at any y position. Assuming that you want to have a width of three pixels (i.e., $y = 71 - 73$), the following gives the desired spectrum.

```
q_list_stat obj51308s_ftc.fits 1 15-295 71:73 1 >! obj51308s_ftc.signal1pix
```

The 5th column of `obj51308s_ftc.signal1pix` is count value at the x -position (corresponding to the wavelength). Notice that this procedure calculates averaged value over the specified pixel range. With this reasoning, multiply a factor of 3 to obtain the added count value, and $\sqrt{3}$ to estimate the corresponding noise level.

```
awk '{print $2,$5*3}' obj51308s_ftc.signal1pix >! obj51308s_ftc.signalAdd
awk '{print $2,$6*sqrt(3)}' obj51308s_ftc.noise1pix >! obj51308s_ftc.noiseAdd
```

It would be good idea to store these results into a file using the paste command on UNIX like,

```
paste obj51308s_ftc.signalAdd obj51308s_ftc.noiseAdd | awk '{print $1,$2,$4}' >!
obj51308s_ftc.SN
```

You now have a file of `objspc_Ftc.SN` that stores the signal [ADU] and noise [ADU] at the corresponding x [pix].

5.2.6 Adding the Spectra

As explained above, one can simply add any 1D spectra which have been created with the same extracting parameters in the case of point-like sources. For diffuse emission, one has to perform image registration prior to concatenating images. The spatial registration between images is normally done with the other object(s) seen in the slit viewer or measuring peak position on the spectroscopy side. Recall that a pixel corresponds to $0.13''$ for the slit viewer, i.e., the imaging side, and $0.165''$ for the spectroscopy side. Because of this difference, this method is valid only for objects that have shifted along the slit. If your object has shifted along the perpendicular direction, you have to deal with this as another slit position data. In summary, the shifting and adding procedure for the spectroscopy data is essentially the same as for that in the photometry observations; the only difference is not to shift along the wavelength direction, i.e., the x axis.

5.2.7 Wavelength Corrections

The spectral and noise data you have obtained are represented by count with respect to pixel numbers, which should be converted into wavelength units. Since the relationship between wavelength and pixel are already known from the transformed sky image (recall that the best-estimated relationship is written in the `skyimage_d1ft.res` file), use those parameters. This gives $a = 1.9900 \times 10^{-2}$ and $b = 7.5774$, then the wavelength correction equation can be written as,

$$\lambda[\mu\text{m}] = 1.9900 \times 10^{-2} \times x + 7.5774$$

If you are familiar with `awk`, type

```
cat obj51308s_ftc.SN | awk '{print 1.9900e-2*$1+7.5774,$2,$3}' > obj51308s_ftc.SN.um
```

5.2.8 Correction of Atmospheric Absorption and Efficiency

Dividing with the Spectrum of a Standard Star

As shown in Figure 5.9, both the spectra of standard star(s) and the object(s) have been equally attenuated by the atmospheric transmission (T_λ) and affected by the instrumental efficiency. Assuming that T_λ is identical both for the standard star and the object,

$$F_{\text{obsobj}} = F_{\text{trueobj}} \times T_\lambda, \quad F_{\text{obsstd}} = F_{\text{truestd}} \times T_\lambda$$

Assuming that F_{truestd} is known, we can estimate F_{trueobj} from F_{obsstd}

$$F_{\text{trueobj}} = F_{\text{obsobj}}/F_{\text{obsstd}} \times F_{\text{truestd}}$$

In conclusion, you can obtain an "atmosphere-free" spectrum by dividing the source spectrum with that of the standard star and multiplying the "atmosphere-free true" spectrum (also called "true" spectrum) of the standard star (See the spectrum of (C)/(A) \times (B) in Figure 5.9).

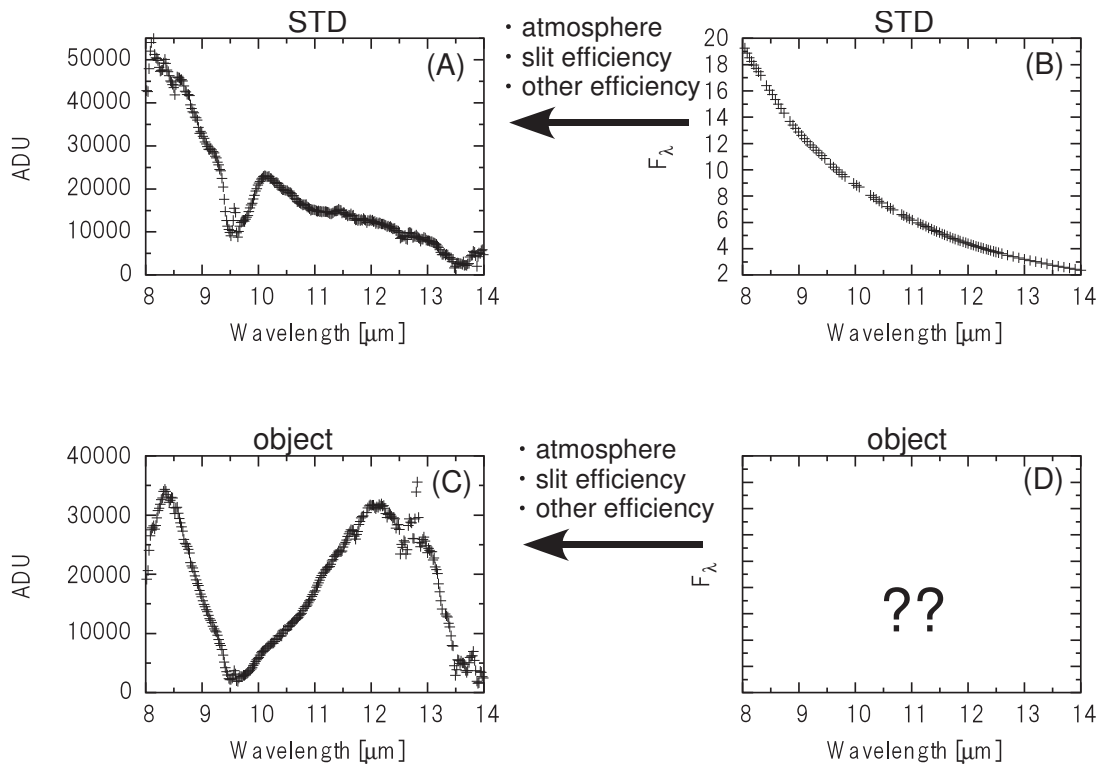


Figure 5.9: A method to obtain spectrum of the object using the standard star. The spectrum obtained by (A)/(B) is a spectrum of the atmosphere including the instrumental response. If you observed a standard star and the object(s) with the identical slit efficiency, (D)=(C)/[(A)/(B)] should give the desired spectrum. However, because the slit efficiencies for a standard star and an object differ from each other in many cases, the user must multiply some scaling factors, see the text for the details.

”True” spectra of the standard stars (F_{truestd}) can be found in Cohen et al. (1999) as templates. They can be obtained from <http://canadia.ir.isas.jaxa.jp/comics/open/guide/index.html>.

Correction of Slit Efficiency

Since COMICS adopts the slit spectroscopy method, all the photons do not fall onto the slit. Therefore, you have to consider slit efficiency. In general, slit efficiency depends not only on the incident angle but also wavelength dependency because the size of stellar images vary with wavelength. Recall that slit width is comparable to diffraction limit. Moreover, one cannot estimate slit efficiency if the stellar image is ”dancing”. With these reasonings, we have to correct slit efficiency not only for diffuse emission but also for point-like sources. Keep in mind that slit efficiencies of the source and standard star are not identical even for the former cases.

In order to correct slit efficiency properly, you need a multi-color data set. Assume that (1) we have two color images, (2) the spectrum obtained so far can be written as $F_{\text{obsobj}}(\lambda)$, and (3) the efficiency equation is defined by $\varepsilon(\lambda) = a \times \lambda + b$ when a and b are constant. In this case, you should find the coefficients of a and b so that $F_{\text{obsobj}}(\lambda)$, the spectrum obtained from the analysis so far keeps a relation of

$$F'_{\text{obsobj}}(\lambda) = 1/\varepsilon(\lambda) \times F_{\text{obsobj}}(\lambda)$$

with those from the photometry observations. Finally, it is worth pointing out that slit efficiency correction works well with a linear correction in many of the cases.

5.2.9 Diffuse Emission

In principal, you can analyze spectral line images of the diffuse emission with the same manner as done for the point-like sources. Namely, you can extract the desired spectrum at arbitrary positions. However, if the emission is widely extended across the images, it would be inefficient and painful to extract spectra at every position for correcting all of them one by one. In this case, it would be wise to apply the correction directly to the 2D image before extracting the spectra. For this purpose, we first have to calculate a correction function (which can be described as

$$F(\lambda) \equiv \frac{F_{\text{truestd}}(\lambda)}{F_{\text{obsstd}}(\lambda) \times \varepsilon}$$

where F_{obsstd} denotes the resulting spectrum of the standard star, F_{truestd} its "true" spectrum, and ε slit efficiency.

Since the correction factors are functions of the wavelength, $F(\lambda)$ can be applied to the desired row by inverting the equation as is done for the wavelength. This allows you to obtain a relationship between pixels and wavelength to make 2D images that will be multiplied with the reduced scientific image (objspc_Ftc.fits). The procedure to make a 2D correction function can be done with task `q_mking`.

For instance, if your correction function is stored in a file called `calib.dat` where the first and second column are X=1 and 2, respectively, such a correction should be done as follows,

```
echo 1 | awk '{for(i=0;i<240;i++){print"cat calib.dat"}}' | sh > tmp
q_mking tmp calib.fits 320 240
q_arith objspc_Ftc.fits \* calib.fits objspc_Ftc_calib.fits
```

5.3 Frequently Asked Questions

5.3.1 Contamination of the Ozone Line at 9.6 μm

You will realize that it is not easy to divide source spectra by that of standard stars around 9.3 – 10 μm caused by the absorption lines of due to the ozone in the upper atmosphere. On the other hand, you may not encounter such a problem if both the spectra have been taken with very close airmasses. Our experience suggests it is likely that airmass mismatching would be responsible for a wavy artifact, or/and bump/dip around 9.3 – 10 μm .

Such an effect may be recovered partially (but not completely) with the following fashion. Using ATRAN (see Lord, S. D. 1992, NASA Technical Memorandum No. 103957), we can calculate the expected atmospheric transparency at the wavelength with the observed zenith angle, then apply it to the spectra. Of course, we have to wisely select the input parameters of ATRAN, such as the representative parameters of the atmospheric conditions such as PWV and number of layers. It may be not trivial to find the appropriate numbers of these parameters, but you can guess them from the weather logs (including from other observatories such as nearby sub-mm telescopes). If you cannot, the standard values would be a good starting point, and you will become known as an expert 10 μm observer from the ground!

5.3.2 Should I perform nodding for my observations?

For a standard MIR imaging, we usually employ not only secondary mirror chopping but also nodding. This observing method will produce a set of data containing four images. The reason to employ chopping and nodding is to reduce the residual pattern in the image after subtraction of the two images taken with the different optics. Such a residual pattern can be removed by subtracting the nodding images. However, since the COMICS system on Subaru telescope has a rather low background noise level originated from system the system, the residual pattern is usually not so obvious. **Therefore, chopping should work pretty well ONLY if the expected intensity of the target is strong enough compared with the residual pattern.** In practice, our experience also suggests that a typical residual pattern at the N11.7 band is 35 mJy arcsec⁻². If your source is brighter than this value, we suggest employing only chopping mode. Of course, if your target is expected to be fainter than the level, you will definitely want to employ nodding.

5.3.3 I have a "ghost" in my low-resolution image at the N band, what can I do?

Unfortunately, the current system often causes such artifacts. As a result, it's almost impossible to use such images for scientific analysis. The only way to reduce such "ghosting" is to configure your imaging filter at through the "hole" position (but please be aware that this does not guarantee perfect removal of the ghost) for future spectroscopic observations. When

you observe with through hole, please keep in mind that you have to decrease the gain so as not to saturate the slit viewer.

5.3.4 I get errors when I transform my image with geotran

For many cases, the error is caused by mismatching the image dimensions or/and the image dimension is not being recognized properly. If this is the case, we suggest explicitly giving the dimensions of the input file, e.g., [1:320,1:240,1].

5.3.5 geomap relevant questions

If you are working on `geomap` several times on the same file, it may cause problems because the task is trying to write the same key words to the created database file. We suggest deleting the output file (or rename it) before repeating the `geomap` procedure.

5.3.6 It takes a loooooooooooooooooong time for task q_startrace to complete ...

This is likely because the initial guess given in the program is far from the flux of your target. Does your source have a rather large signal value? If this is the case, try to divide the image by certain number, e.g., 1000. On the other hand, if the sky background level is high and S/N of the source is low, the program may fail to converge around the x -value. The key is to avoid such x value(s) and try to run `q_startrace` again after removal of such points.