

COMICS DATA REDUCTION MANUAL Ver.2.1.1

Combined effort of the past and current members at the COMICS team and Subaru Telescope
Current Version Editor: Ray S. Furuya (rsf@subaru.naoj.org)

May 14, 2008

Contents

| | | |
|----------|--|-----------|
| 1 | INTRODUCTION OF COMICS | 5 |
| 1.1 | Mid-Infrared Astronomy | 5 |
| 1.2 | An Overview of the COMICS | 5 |
| 2 | COMICS Data Set | 7 |
| 2.1 | Observing Modes — Chopping and Nodding — and Structure of Raw Data | 7 |
| 2.2 | Summary of COMICS Data Reduction | 10 |
| 3 | FITS FILE AND HEADER | 14 |
| 3.1 | From the 1st to 8th Lines | 14 |
| 3.2 | Header Information Common in All the Subaru Data | 15 |
| 3.2.1 | Header Parameters about Status of the Telescope | 15 |
| 3.2.2 | Header Parameters for Setting of Optics | 15 |
| 3.3 | Header for COMICS | 16 |
| 4 | Overview of Data Reduction: A Standard Procedure | 18 |
| 4.1 | Getting started: the Reduction Software | 18 |
| 4.1.1 | q_series | 18 |
| 4.1.2 | IRAF | 18 |
| 4.2 | Which data should I work on? | 21 |
| 5 | Data Reduction: The Details | 22 |
| 5.1 | Imaging at N and Q Bands | 22 |
| 5.1.1 | Making Dark | 22 |
| 5.1.2 | Making Flat Images | 23 |
| 5.1.3 | Making a Bad Pixel Mask | 25 |
| 5.1.4 | Data Reduction of Scientific Data | 27 |
| 5.1.5 | Adding the Images | 28 |
| 5.1.6 | Misc. | 32 |
| 5.2 | Analyzing Spectroscopy (N-band Low Resolution; NL) Data | 33 |
| 5.2.1 | Preparing for a Dark | 33 |
| 5.2.2 | Making a flat image | 33 |
| 5.2.3 | Reduction of the Read-out Pattern Noise | 35 |
| 5.2.4 | Data Reduction of Scientific Images | 35 |
| 5.2.5 | Extracting the Spectra | 40 |
| 5.2.6 | Adding the Spectra | 40 |
| 5.2.7 | Wavelength Corrections | 40 |
| 5.2.8 | Correction of Atmospheric Absorption and Efficiency | 41 |
| 5.2.9 | Diffuse Emission | 42 |
| 5.3 | Frequently Asked Questions | 43 |
| 5.3.1 | Contamination of the Ozone Line at $9.6\mu\text{m}$ | 43 |
| 5.3.2 | Should I perform nodding for my observations? | 43 |
| 5.3.3 | I have a "ghost" in my low-resolution image at the N band, what can I do ? | 43 |
| 5.3.4 | I got only errors when I am transforming my image with <code>geotran</code> | 43 |
| 5.3.5 | <code>geomap</code> relevant questions | 43 |
| 5.3.6 | It takes so loooooooooooooooooong time for task <code>q_startrace</code> to complete | 43 |

List of Tables

| | | |
|-----|---|----|
| 1.1 | An Overview of COMICS | 6 |
| 4.1 | Summary of a Standard procedure: Imaging | 19 |
| 4.2 | Summary of a Standard procedure: Spectroscopy | 20 |

List of Figures

| | | |
|-----|---|----|
| 2.1 | Basic data structure taken with COMICS observation ("chopping" and "nodding") | 8 |
| 2.2 | Structure of a COMICS data file | 9 |
| 2.3 | Subtraction of chopped and nodded images | 11 |
| 2.4 | Determination of wavelength and spatial axes in spectroscopic observations | 13 |
| 5.1 | Standard Procedure to Make a Dark | 23 |
| 5.2 | Standard Procedure making Sky Flat Image | 26 |
| 5.3 | Making a Mask Image with <code>q_badpix</code> | 27 |
| 5.4 | A method to add images for the extended emission | 30 |
| 5.5 | A method of photometry data reduction | 31 |
| 5.6 | A method to reduce COMICS spectroscopy data: An example for a point-source with the low-resolution mode | 34 |
| 5.7 | Wavelength – pixel relationship | 37 |
| 5.8 | An example of space-constant line and the fitting results | 37 |
| 5.9 | A method to obtain spectrum of the object using that of standard star | 41 |

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 |

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 the earth at MIR regime. These are referred to as the N and Q bands which have been long-standing unexplored regime because of limitation from high background noise and technical difficulty in MIR detectors. However, such a situation has been dramatically changed due to the recent progress in development of new generation 2-dimensional (2D) array detectors as well as construction of large diameter optical/infrared telescopes. In order to make possible MIR observations 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 H. Kataza and T. Yamashita.

The MIR waveband stands unique with following characteristics. First, MIR, the longer wavelengths of IR radiation can penetrate interstellar cloud than the shorter ones, e.g., optical/near infrared (NIR) do, hence allowing us to detect deeply embedded objects. In general, MIR observations make it possible to perform the deeper search for newly found objects identified in optical wavelengths than NIR does, although MIR emission suffers from absorptions due to interstellar silicates. Second, MIR regime is the most efficient band to detect blackbody (thermal) radiation at temperature of 100 – 400 K, making COMICS one of the best instruments to observe warm thermal emission emanated from circumstellar disks around young stellar objects as well as warm gas and dust around late-type stars. Third, the MIR regime would be a key band to have a better knowledge of the role and evolution of interstellar dust because there are numerous emission and absorption features spread over the band. One of the well-known such features can be seen around $9.7\ \mu\text{m}$ and $18\ \mu\text{m}$; the feature are believed to be 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 $\lambda = 6.2, 7.7, 8.65, 11.25,$ and $12.8\mu\text{m}$ can be detected towards various type of objects.

1.2 An Overview of the COMICS

COMICS is an instrument at the 8 m Subaru Telescope aimed to observe MIR wavelengths, which is expected to explore various astronomical objects such as those in the solar system, newly formed stars and planets, interstellar dust, nearby and even distant galaxies. COMICS offers two modes of observations: imaging and long-slit spectroscopy. COMICS is mounted at the Cassegrain focus of the telescope to minimize thermal emission from the telescope. In general, one must cool down the instruments for the MIR bands as the black body radiation peaks around $10\ \mu\text{m}$ at the room temperature. In fact, all the optics and the associated parts such as motors are stored in the dewar and are being cooled down to less than 30 K. In addition, we need to accommodate the detectors (mentioned in the below) further down to 4 K to reduce dark currents.

In order to have diffraction limited image with the 8 m telescope, COMICS is designed to have a pixel field of $\sim 0.13^{\text{arcsec/pix}}$ for the imaging mode. For spectroscopic observations at $10\ \mu\text{m}$ (N band), COMICS offers low (L)-, medium (M)-, and high (H) resolution modes with $R \sim 250, 2500, 10000$, 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 4 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. At the imaging mode, COMICS uses one of the IBC type detector. One 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 the spectroscopic observations, we configure all the 5 detectors

in parallel in order 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 whole atmospheric window at the medium resolution mode. In addition, we offer a total of 4 slits — available at both the 10 and 20 μm bands — having different widths for the possible variation of the seeing. Table 1.1 summarizes the key parameters of the instrument.

| | Imaging | Spectroscopy |
|---------------------|--|--|
| Pixel Field of View | 0.130"/pix | 0.165"/pix |
| Detector | 320x240 Si:As $\times 1$ | 320x240 Si:As $\times 5$ |
| Field of View | 42" \times 31" | 40" for long-slit |
| Wavelengths | N: 8.8–12.4 μm (5 bands) Narrow bands for UIR 8.6, 11.24, [ArIII], [SIV], and [NeII] Q: 17.7–24.5 μm (4 bands) | N: 7.8–13.3 μm Q: 16.7–20.7 μm |
| 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, file structure, and illustrates a rough idea on how to reduce your data.

2.1 Observing Modes — Chopping and Nodding — and Structure of Raw Data

For most of observations with the COMICS, high speed read-out is used to compensate highly time variable background level at 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 sole the chopping method. However, since chopping is being done through tilting of the secondary, a subtracted image obtained from the one beam image to the another one has low-level residual pattern for the most cases. This is because optics for the two chopped beams are slightly different from each other¹. Such a residual pattern can readily decrease detection limits, which severely affects in detecting faint objects. To avoid such an 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 chopping sequence that is the same as for the target position; this will cause a similar residual pattern with that seen in the target position. At the data reduction process, we will subtract the offset image(s) from the target image(s) to remove such a low-level residual pattern. Given this purpose, it is sufficient to "nod" with a low throw-frequency than "chopping" does. A nodding cycle is typically several minutes to obtain a pair of images. Lastly, one should keep in mind it would depend on the source brightness and/or scientific goals whether or not to use "nodding".

Figures 2.1 and 2.2 illustrates a rough idea on how COMICS obtains a set of images and structure of the image FITS files, respectively. A standard observation will perform chopping and nodding scans towards slightly different pointing 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 the 4 images labeled with 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 the region without diffuse emission. On the other hand, the user should carefully select these parameters in the case of extended source 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, this gives reasonable atmospheric cancellation and good mechanical performance.

As can be seen in Figure 2.2, each of the 4 images labeled with A – 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 the 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 of 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 exactly identical data structure. Each nodding image has substructure with the two chopped beam images. The nodding images from the beams A and C constitute a pair together with those from another nodding beams of B: $B_1 \sim B_n$ and D: $D_1 \sim D_n$. In conclusion, a standard COMICS data that are ready to reduce also contain the other nodded beam data of B: $B_1 \sim B_n$ and D: $D_1 \sim D_n$.

¹ Although 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.

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

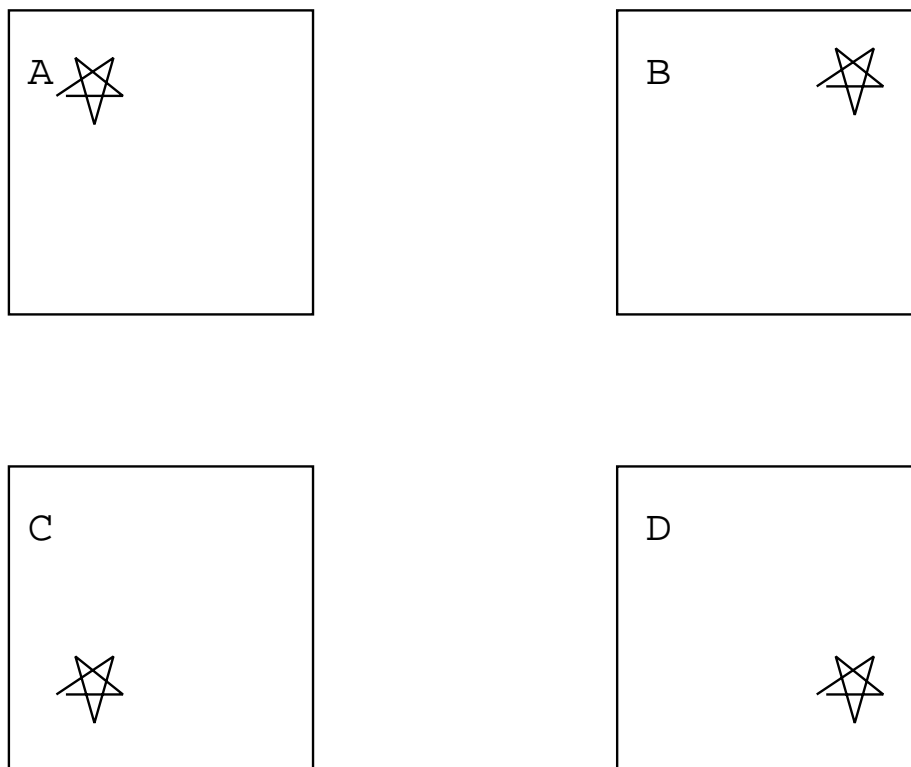
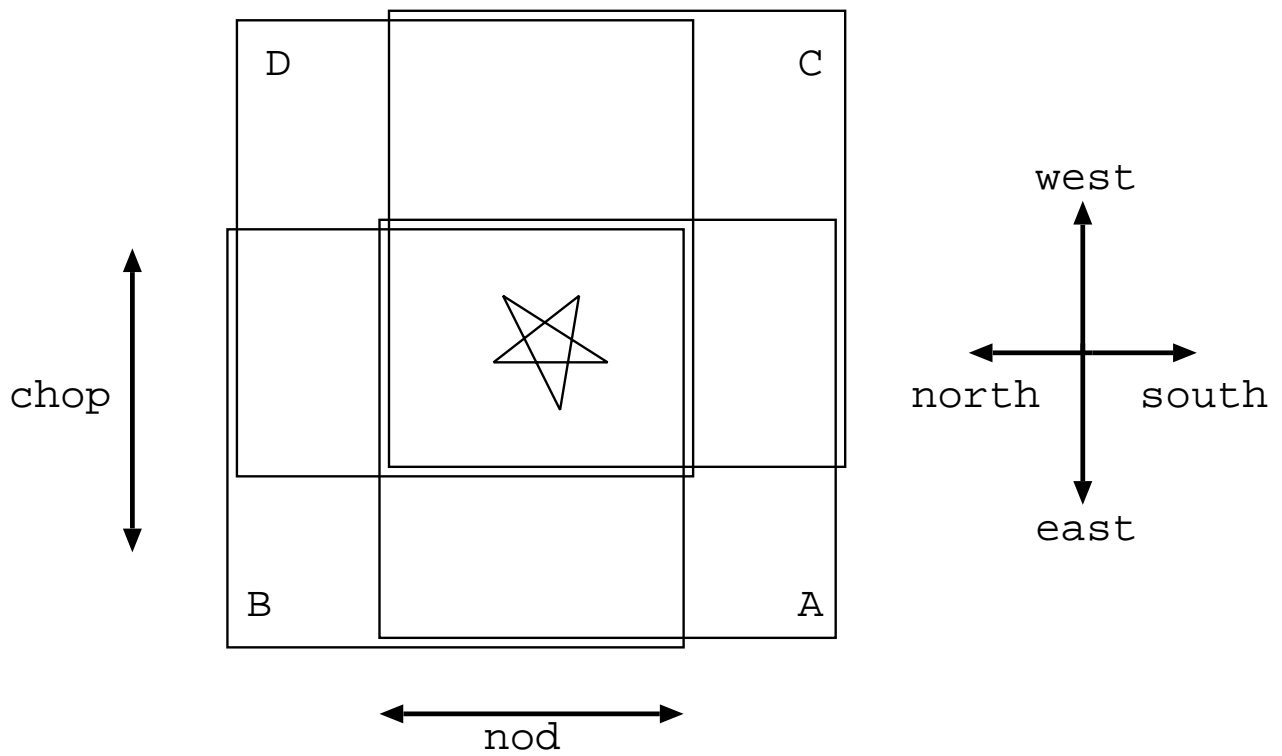


Figure 2.1: An illustration of typical data set taken with COMICS. The user will obtain a data set consisting of 4 different images labeled with A – D. These 4 images have been taken at slightly different pointing centers in the plane-of-sky through chopping and nodding scans. The user can select an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

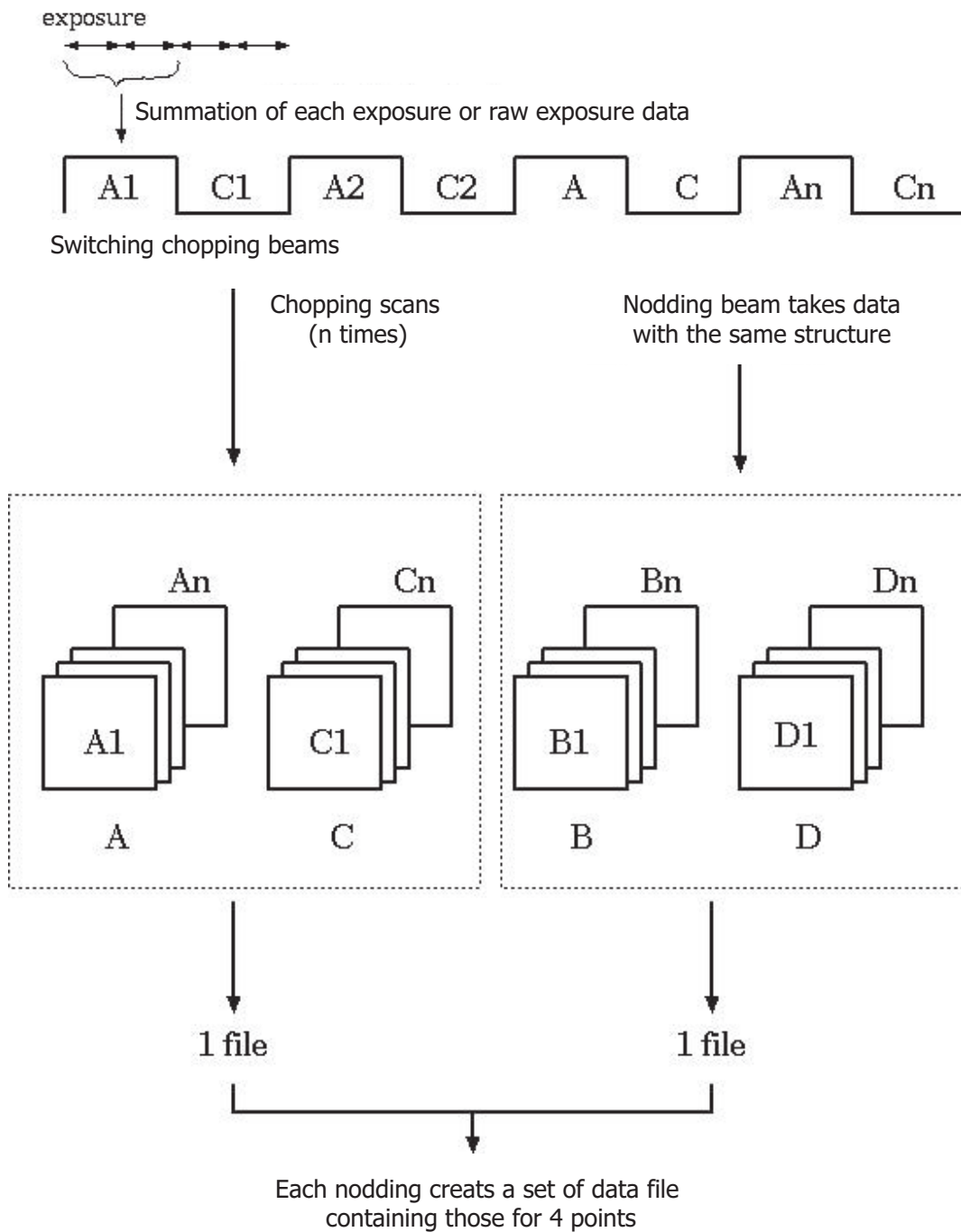


Figure 2.2: Schematic image of a typical a COMICS data file: Chopped and noddied beam images set illustrated in Figure 2.1 individually contain sub-images. This is because almost all the data will be taken through multiple chopping scans for each nodding beam. Each chopping beam images contains either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

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 with the 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 gives the following 3 types of data sets.

- 1) **RAW mode:** This mode keeps all the exposure and all the frames. Images in each file are A_i^j and C_i^j .
- 2) **ADD mode:** This mode adds all the exposure 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 $\Sigma_j C_i^j$.
- 2) **ECO mode:** Since this mode collapses all the images into an image on-line, and makes 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 obtained images without chopping and nodding for obtaining dome-flat data (see §5.2.3). In this case, the observers will notice being `Chop=0` in their headers; they should see multiple frames whose numbers have been specified with the parameter of `ChopNum`. Notice that the number of the exposures have been repeated with that specified in the individual frame for this case as well. As explained in the 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` as 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 a fully calibrated image(s). The details of each step of data reduction will be described in §5.

1. Subtraction of Chopping and Nodding Data We suppose that the user have obtained non-calibrated (i.e., “raw”) data set which contains four images (§2.1) where the source of the interest can be seen at different pixel positions. First, the user needs to subtract chopped images taken in sequentially with the different chopped beams (but, they must have been 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 originated from the dark current. However, the user may find very low-level residual pattern which is caused by the usage of different optics. Second, 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 of I . For many cases, the above operation will give you images whose sky level is close to zero. If you have made multiple nodding observations, you need to make the image of I for each nodding.

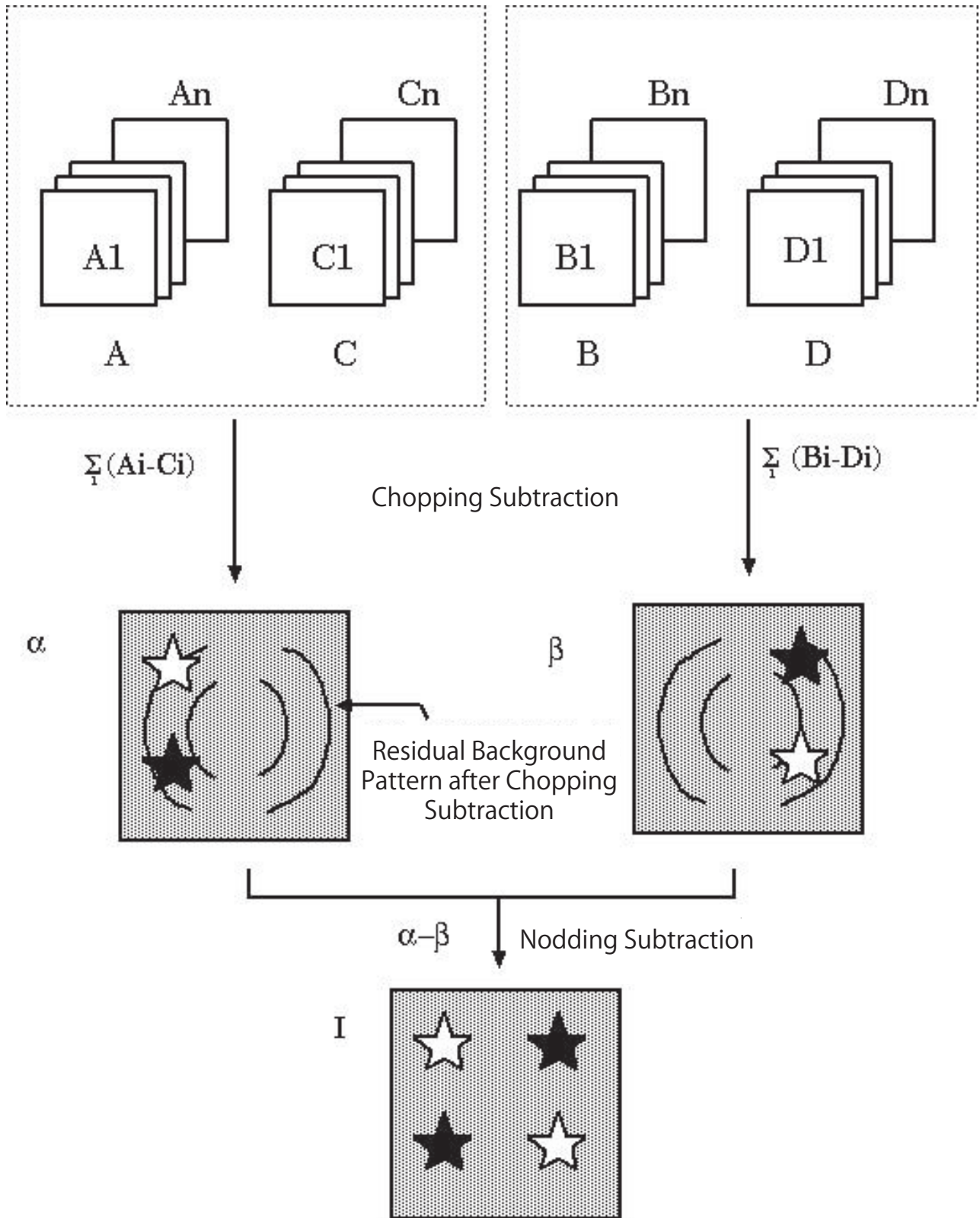


Figure 2.3: Subtraction of chopped and noded images: The first step of the COMICS data reduction is subtracting images between the same nodding beams and between the different chopped beams; the resulting 4 images are stored in a data set. The subtraction between the images of A and C in the upper panel gives the image of α in the middle (the same for the B and D to make the subtracted image of 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 allow to eliminate residual pattern caused by the usage 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 interests due to the above subtraction processes.

2. Flat Fielding The next step is to divide each of the chopping- and nodding-subtracted images by a flat image to correct inequity in 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 of COMICS observations, dome-flat data are taken by observing thermal emission from the dome, windscreen, or mirror covers etc, instead of observing lamps.

3. Image Transformation (Distortion Correction) and Wavelength Calibration for Spectroscopy Observations For spectroscopy 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 “spatial axis” from point-source image such as the standard stars. Here “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 e.g., 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. 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. Last, it is desired to calibrate the dispersion axis through a comparison with a standard model of the terrestrial atmospheric emission.

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

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

Cohen et al. (1999) listed flux densities at N and Q bands measured with low-resolution. To apply their results on your data, you must properly handle difference 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 the spectra shown in the paper for each N band filter. This is because, unfortunately due to the very high background noise, almost no measurement have been done for individual filters.

Last, our experience strongly suggests that the one should pay attention not only difference in airmass but also the presence or absence of clouds in the sky because they heavily affect on the quality of the images. Given that sky noise level is intrinsically high at MIR bands regardless of the amount of clouds, some observers may have continued to observe with a little amount of clouds. However, it should be noted that even small amount of clouds can readily degrade your image quality. For many cases, the user is required to correct effect of the atmospheric absorption even for the standard star data obtained with the same airmass. In conclusion, we strongly suggest making sure whether or not spectral shape of the standard star(s) has a fairly well consistency with that of the atmosphere.

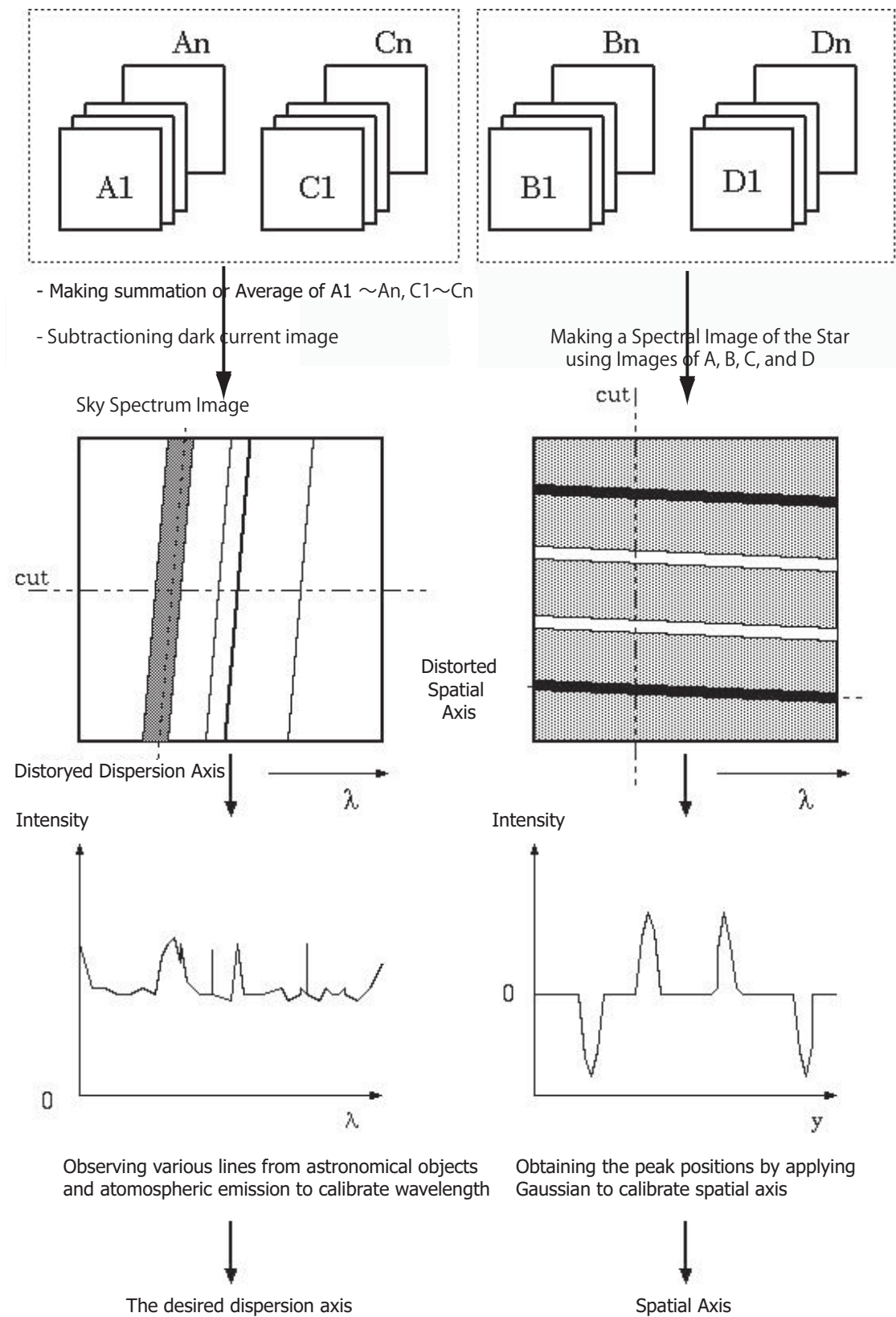


Figure 2.4: Determination of wavelength and spatial axes in spectroscopic observations: absolute coordinate of the wavelength axis can be identified from a comparison of the spectral image obtained from a summation of the raw data with an atmospheric transmission model. As for the spatial constant axis, 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.

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 files starting with “COMA”, and the other 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 images taken with the same chopping beam, 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 below gives an introduction on how to read FITS header information of your COMICS images; you may be looking a header with a different version that can be found with `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> | = | T / Standard FITS format |
| <code>BITPIX</code> | = | 32 / # of bits per pixel |
| <code>NAXIS</code> | = | 4 / of axis in frame |
| <code>NAXIS1</code> | = | 320 / # of pixels/row |
| <code>NAXIS2</code> | = | 240 / # of pixels/column |
| <code>NAXIS3</code> | = | 50 / # of pixels/time |
| <code>NAXIS4</code> | = | 2 / # of pixels/detector |
| <code>EXTEND</code> | = | F / ASCII Extension Table |

The first 8 lines in COMICS header follows the FITS format standard by the IAU. The second parameter of `BITPIX` shows bit number; `BITPIX` with minus ($-$) indicates the numbers of floating decimal points, while without $-$ does 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 of `NAXIS = 4` indicates dimension number of the image files. Note that COMICS data have 4-dimensional (4D) structure; `NAXIS1` and `NAXIS2`, respectively, represent the x- and y axes of the detector, `NAXIS3` does time-axis (`NAXIS3`), and `NAXIS4` the detectors. This means that you obtain one single file even if you observed with more than one spectrometers. If you observed with the imaging or low-resolution spectroscopy mode, you see `NAXIS4 = 1` for `NAXIS = 4`.

Since COMICS has a detector array consisting with 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 jorgen, 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 into 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. For most of the cases, the file number for the imaging must be smaller than that for the spectroscopy by one. If you observed only with imaging mode, the paired spectroscopy FITS file has only the header part.

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

3.2 Header Information Common in All the Subaru Data

The header information after

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

to the COMICS header parameter is those 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 about Status of the Telescope

We advise the 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
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 of "instrumental rotator"; the NST-PA parameter shows the rotator angle with respect to the source of the interests. AUTOGUID tells you whether the observations have been done with the auto-guider which may have used for most of the observations. The 9 parameters starting with M2- gives information about the secondary mirror (however, many users will not need to give special attention.)

3.2.2 Header Parameters for Setting of Optics

You will see the following parameters that describe coordinates information of the images. For many cases, we believe that the users cannot execute a command to overlay all the images simply referring to the coordinate information described in the following. This is because nominal tracking accuracy of the telescope is not so accurate for allowing such a blind image adding. Even if the above is not the case, you cannot add your images sole with the header information. Namely, if you adopted chopping throw of more than 30", imaged fields in the plane-of-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
SLTCP1X1= 120.0 / Slit center projected on detector(pix)
SLTCP1X2= 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 slit viewer.

FILTER01, 02, 03 and 04 describes the filter names for fore-optics 1, 2, filter name for imaging, and name of the lens, respectively; the first character indicates either filter (F), lens (L) or grating (G), the next two

numbers are ID 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 shows the ID for the holes), L10 (L20) shows lense ID for 10 (20) μm with the flag of I(P) indicating Imaging (Photometry). L10 (L20) in the grating mode with G has the same meaning as the 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 following two digit numbers 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 parameters only for N-band spectroscopy observations at the current observing system. It should be noted that EXPTIME indicates not integration time for whole the observations, but that for per exposure. DET-TMP shows the measured temperature of 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 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 is being read among the 6 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 at 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 the reference).

Q_CLKFL shows name for data type (i.e., clock parameter). The other important parameters to know the clock status are Q_PIXTIM (read-out rate), Q_RRSTRT (electrical optimized ND value), Q_CHWB (number of wipe 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" for the case of 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_CHTHR=          60.00 / Chopping Throw
Q_CHDEG =          53.34 / Chopping Degree
COMMENT = ' ----- FITS VERSION'
Q_GETVER= '4.11          ' / FITS header VERSION
END
```

Q_CHTHR indicates chopping throw defined by peak-to-peak in arcsec, and Q_CHDEG does position angle for the chopping direction in degree.

Lastly 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 the following, we describe a standard procedure to reduce COMICS data. Since we will deal with the data whose observing wavelength is more than one order of magnitude larger than the other IR/optical wavelength observations, the users are strongly recommended to use the originally developed data reduction package. 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 as the users' demand.

`q_series` can be categorized into two groups. One is the basic tools that can be used for a "quick-look" purpose at the summit; these are mathematical operations of 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 shift of images are supposed to use widely used other packages, like IRAF. We believe that both `q_series` and IRAF can handle almost all the data currently being taken.

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

In the following, we will show a standard procedure of data reduction with the command names in `q_series`. Each of the commands is described in the subsequent chapter in details. We suggest preparing for a summary of your FITS files that should be reduced with describing data types, qualities and so on. Such a man-made summary table would be pretty useful than a list created by reading FITS headers of the target files simply because most of the 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 in above, you need to have

- `q_series`
- IRAF

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

4.1.1 `q_series`

One 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 of `rbin.tgz`, extract the file. You will see a directory of `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 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.

Table 4.1: Summary of a Standard procedure: Imaging

| | | |
|--|---|---|
| 0. Data Examination | Checking clock and optics | <code>q_headlst2</code> |
| 1. Making Dark | Dark per exposure | <code>q_list_stat</code> <code>q_fcombine</code> <code>q_arith</code> |
| 2. Making flat 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 | <code>q_arith</code> <code>q_bsep</code> <code>q_list_stat</code> <code>IRAF:gauss, q_arith</code> |
| 3. Making mask | Finding bad pixels | <code>q_badpix</code> |
| 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 | <code>COMQ</code> images <code>q_subch</code> <code>q_arith</code> <code>q_arith, q_fcombine</code> <code>IRAF:imshift</code> <code>q_photo</code> or <code>IRAF:imexamine</code> |

Table 4.2: Summary of a Standard procedure: Spectroscopy

| | | |
|---|---|--|
| 0. Data Examination | Checking clock and optics | <code>q_headlst2</code> |
| 1. Making Dark | 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 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 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> |

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

If you have a 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 most of the observers may have numerous numbers of FITS files, it's easy to use task `q_headlst2` in the `q-series`.

One of the useful features of the task `q_headlst2` is that it can read an ASCII text files which stores a list of FITS file names. For instance, let us 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 like,

```
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 files of `*clk,opt,temp,tel1`, and `tel2`.

Chapter 5

Data Reduction: The Details

5.1 Imaging at N and Q Bands

5.1.1 Making Dark

Making Dark Images

For many observations, dark data have been taken either after or before scientific imaging; these calibration data are supposed to have been taken with the same exposure parameters (i.e. clock configuration) for the source of interests, standard star or/and dome flat. However, it is worth checking whether or not they are identical because the current COMICS system allows to use different frame numbers or/and CoAdd mode. Another reason is that 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 is to know dark counts per exposure for many of 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, which can be done with `q_list_stat`. To use `q_list_stat`, the user must specify properly the following options: the file and detector names, pixel ranges (if you want to all of them, use "- -"), and ":" which selects averaging along time axis.

The second step is to average the resultant files of m51219 and m51221 using task `q_fcombine`. We should 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 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 the above procedure for all the types of observations that he/she has to reduce. For most cases, these are for the source of interests taken in the same night, standard star(s), dome flat for spectroscopy (if exists) as well as all the data that have exactly identical parameter combinations of `img/spc`, `PP`, `ND`, and `Ystart` parameters. At this point, we suggest making a logging sheet by hand for your records because you have to make many dark images with different clock configurations.

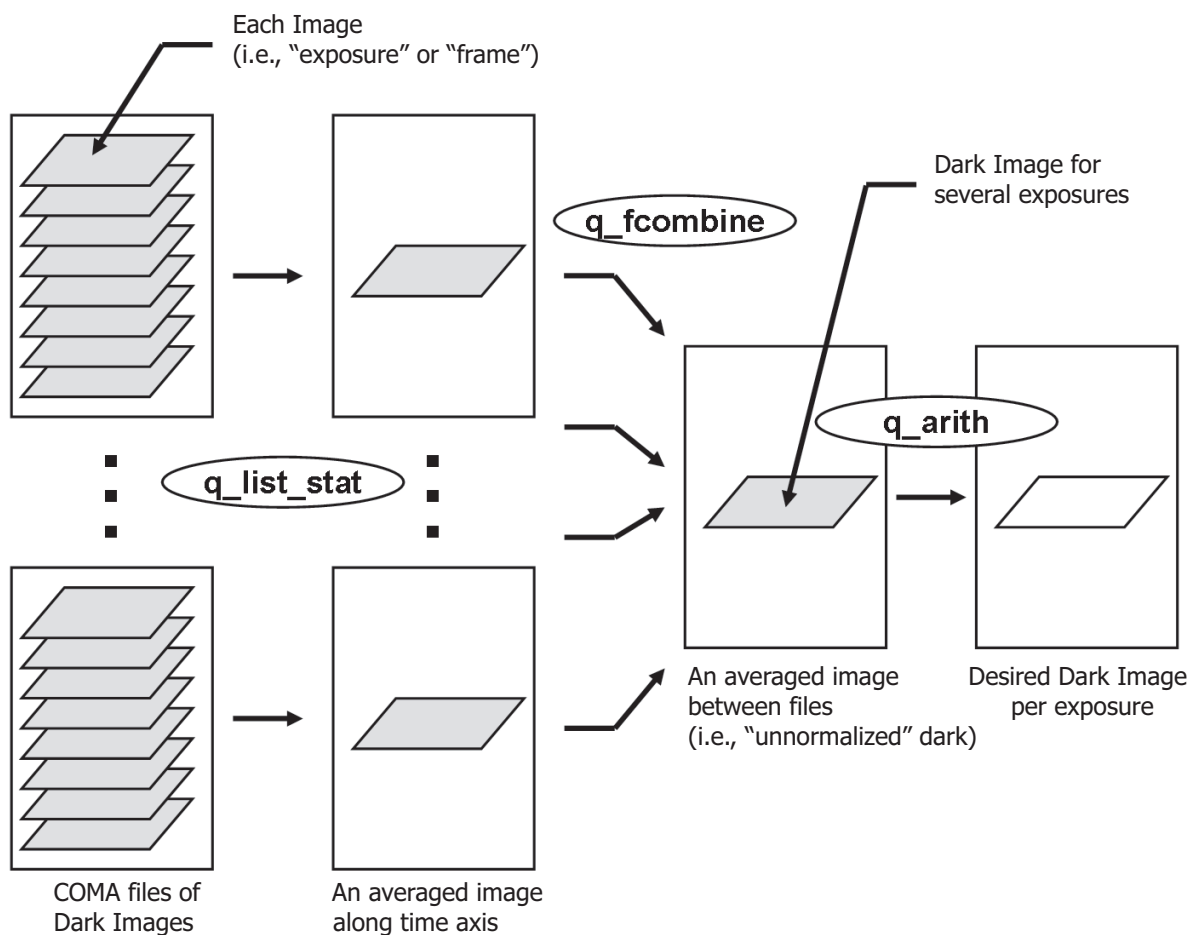


Figure 5.1: A typical procedure making a dark requires to average all the dark images taken with the same clock parameters, and converting the averaged images into a dark image per exposure.

5.1.2 Making Flat Images

To make flat images, the user may select either of 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 have severe contamination from atmospheric emission lines whose wavelengths may have shifted. In the below, we suppose that the user has data available to make "sky-flat".

To make "self-sky-flat", you firstly have to split the images for each chopping beam. This is because we 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 once focus the incident light onto the slit, inequity of the slit surface is being transferred into the flat images at most cases. Therefore we have to use sky images taken with photometry mode for flat fielding, and 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 sky data that have been taken without changing filter positions. This is again because there are inequity 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 difference in telescope pointings causes non-negligible difference in optics, hence we suggest making flat images using off-beam sky image for each target image.

It is well known that the resultant sky flat has a residual gradient entire the image; the amplitude of the gradient is usually $\sim 10\%$ of the peak. If you use such a flat image, you would realize that the accuracy of photometry is most likely worse than that of the image made with the above flat image. You may want to see the difference by comparing two photometry results toward an object; one case is that the object is imaged at the center of field view and the other case is that the object is seen close to the edges 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 assumed to analyze for imaging observations.

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

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

First, we 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 for this case, we must use a dark image which has been obtained from an image created by adding 16 exposures. We can 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 with 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; 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 Gaussian Beam

As 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
```

The above example shows the case that we have convolved with the Gaussian beam having a FWHM of 6 pixels which is the best-searched value for most cases on the basis of 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 for the surroundings, and most likely due to dusts on the detector or/and inequity 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 soundings or/and pixels being suffered by artificial line(s). The tasks of `q_badpix` or/and `q_2val` 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 threshold value you specified and 1 for the other pixels. For instance, if you want to identify bad pixels whose sensitivity is 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, the above 80% and 120% are for an example, and we advise making trial and error to find the best threshold for your data. Our experience suggests that the above method should work for most cases because the flat images tend to have rather uniform values. However, if the above method 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 that have no incident photons. Unfortunately, since there is no general method that can work for all type of data, you may consider to consult with your local expert or/and the support astronomer, depending upon the cases.

Making a Mask Image

With the above procedure, we suppose 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.

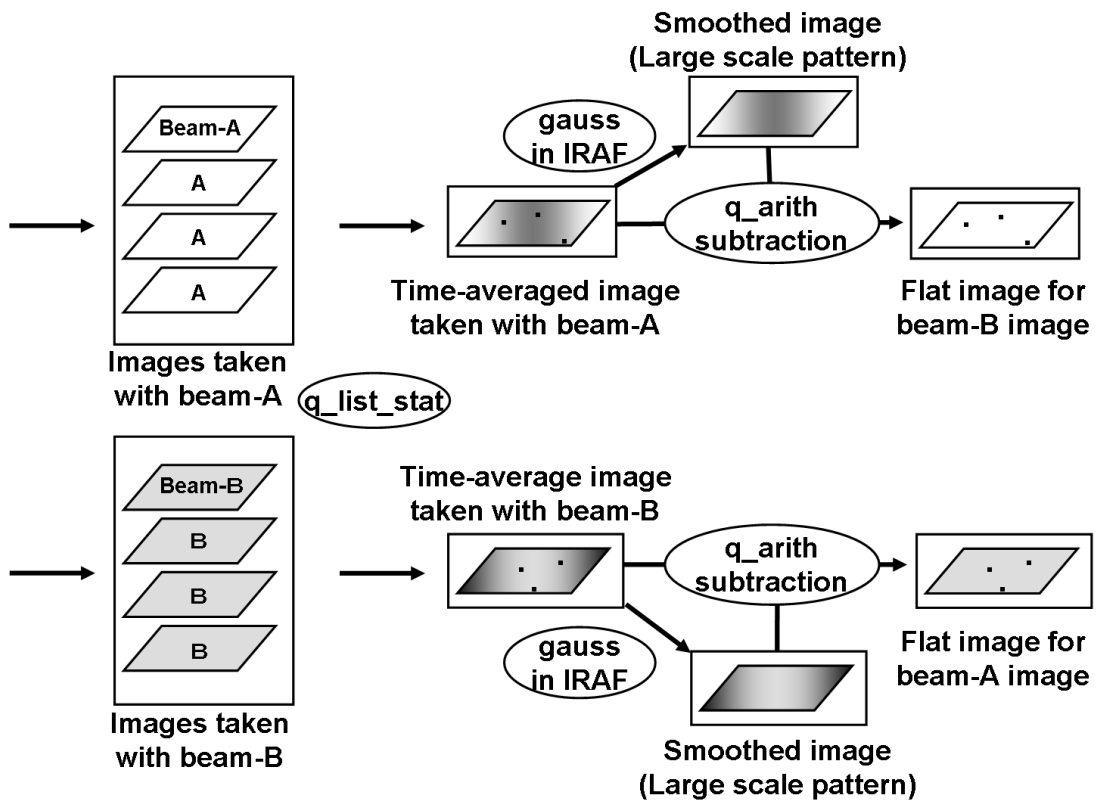
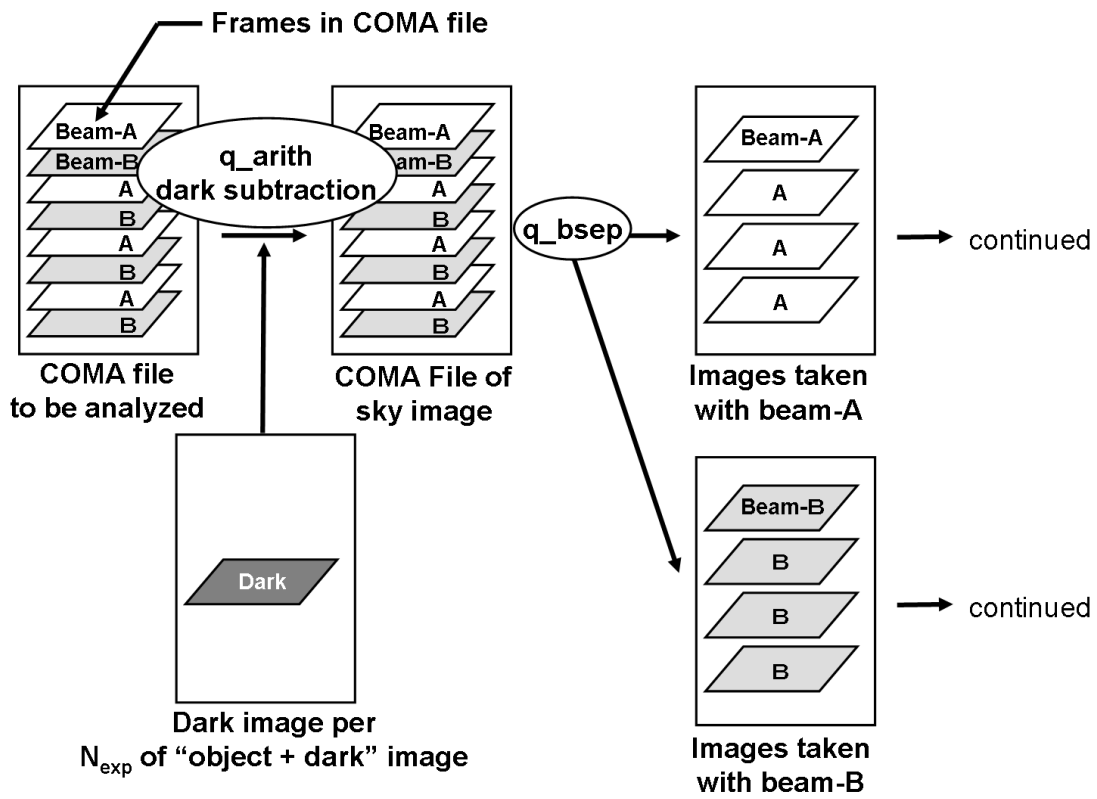


Figure 5.2: Standard Procedure making Sky Flat Image: For a case , the user can get the desired flat image(s) after subtracting dark, beam splitting, averaging, and removing large-scale residual pattern, Notice that a flat image created from the one beam data can be used for sky-flat for the data taken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

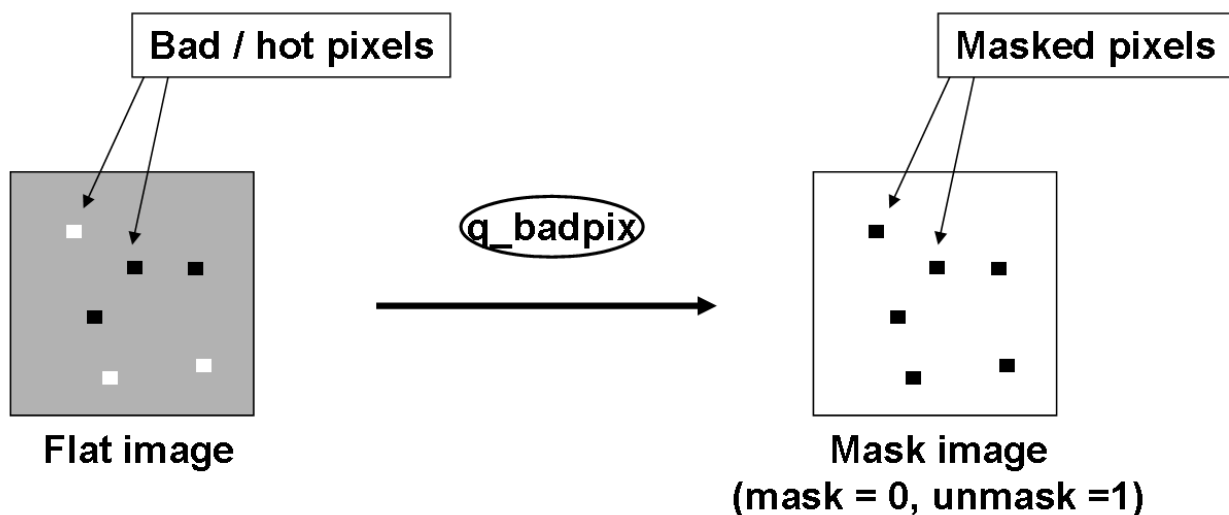


Figure 5.3: Making a mask image with task `q_badpix`: The task produces a mask image where the bad or/and hot pixels whose sensitivity are high or low compared with those for the surroundings. The users can specify threshold values to identify unusable pixels; the resultant mask image has values of 0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

5.1.4 Data Reduction of Scientific Data

We suppose that you are now happy with the flat and bad pixel masking images. The next step is to apply these images on those for the source of the interests.

Subtraction of Chopping Images — COMQ Images

We need to subtract the off-beam images from the on-beam ones for the subtraction of the background. In practice, however, the user can simply use the COMQ images instead of making such images by hand. If you want to know image statics 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 the 16 channels, at the same time, the readout noise should have same pattern noise for the 16 channels. Recall that each of the 16 channels has 20 columns. On the other hand, because the target source(s) is usually detected over some portion of the detector, we can reduce the readout noise using the pixels where the source(s) is not detected.

For this purpose, we use the task of `q_subch` which can subtract the median of the input image. If your source(s) is much smaller than whole 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 the 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 the 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 would recognize that the "random" readout noise has increased, although the "pattern noise" has been subtracted with satisfaction.

Dividing with the Flat Image

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

Let's start from flat-fielding for the positive image at first. In this case, we have to divide the target images with the flat made from the beam-B (the negative) images. After that, we 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 one. But, please be advised to multiply -1 for the positive images. Just in case, 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 suppose that you now have completed the calibration processes that are required for each image and should have the final positive images for each image files (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, one 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 errors. 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 with the task `q_arith`. For many cases, you, however, must issue the command several times, which is not smart enough. 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 `objlist` file which 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 angular resolution (i.e., sharpness) of the original data, we advise you not to add the image files that have been taken over the long observing time. This is simply because your source positions when measured with the detector coordinate have shifted gradually during long observing time. If your source(s) has sufficient S/N (and ideally 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) is 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 shift 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, let us discuss more precise method in the following.

1. Getting Offset for Image Shifting

In order to add the fully calibrated images, we need to measure positional offsets between images. If you have bright compact source(s) in your field of view, you can use it for a positional reference to calculate offsets using task `q_photo`. If you don't have a compact source while only an extended emission in your field of view, you cannot take the above approach. A 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 the adding. Figure 5.1.5 illustrates how this method works. Keep in mind that you must exclude column or/and rows that contain bad pixels before calculating the cross-correlation.

You have to repeat the above process for all the images that you want to add. For the sample images of 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 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 in the above is to make a template image for the shifted images, although the given values are zero. The purpose for this is that we have to define dimension (i.e., pixel numbers) for the resultant image.

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

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

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

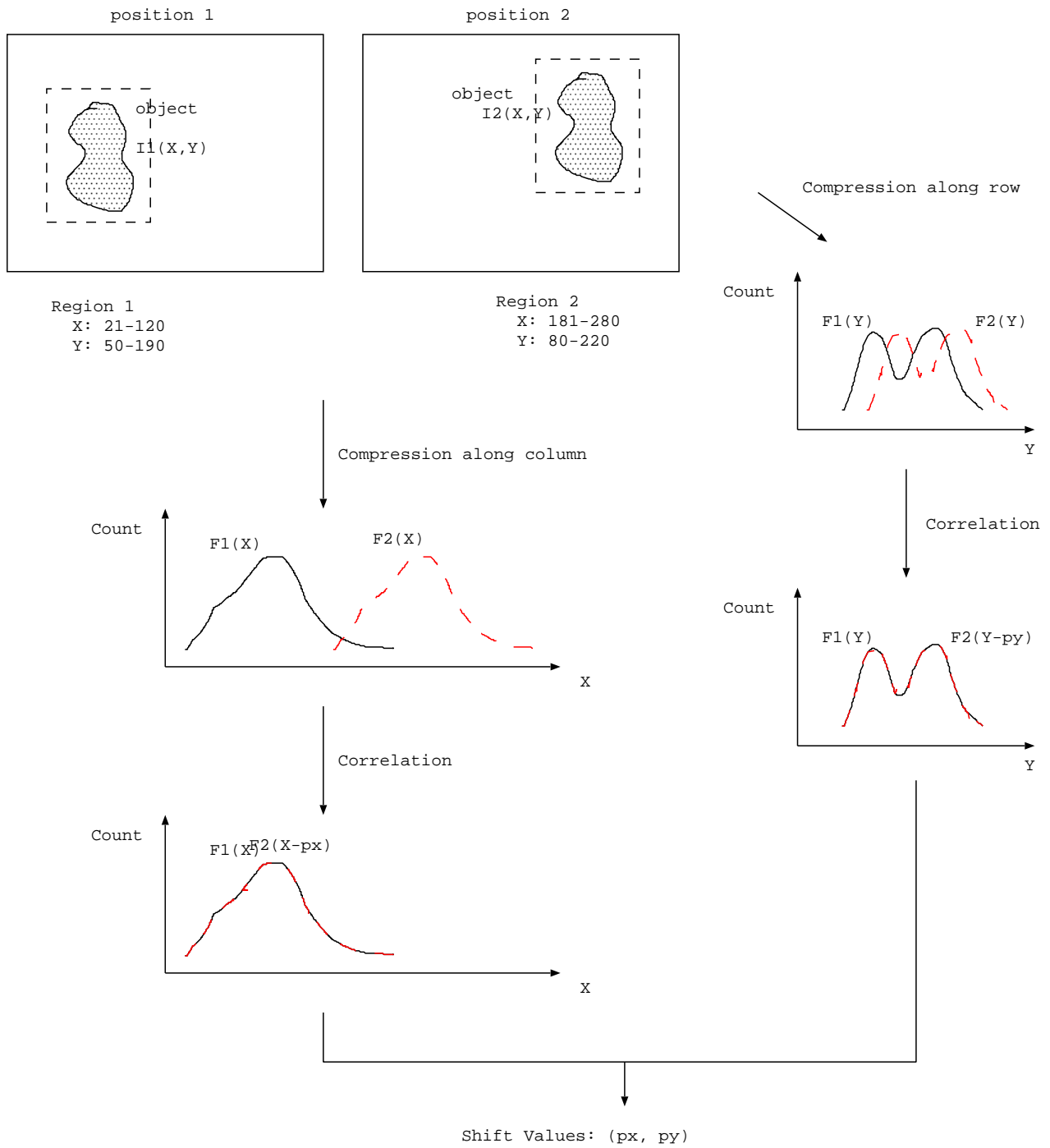


Figure 5.4: A method to add images for the extended emission

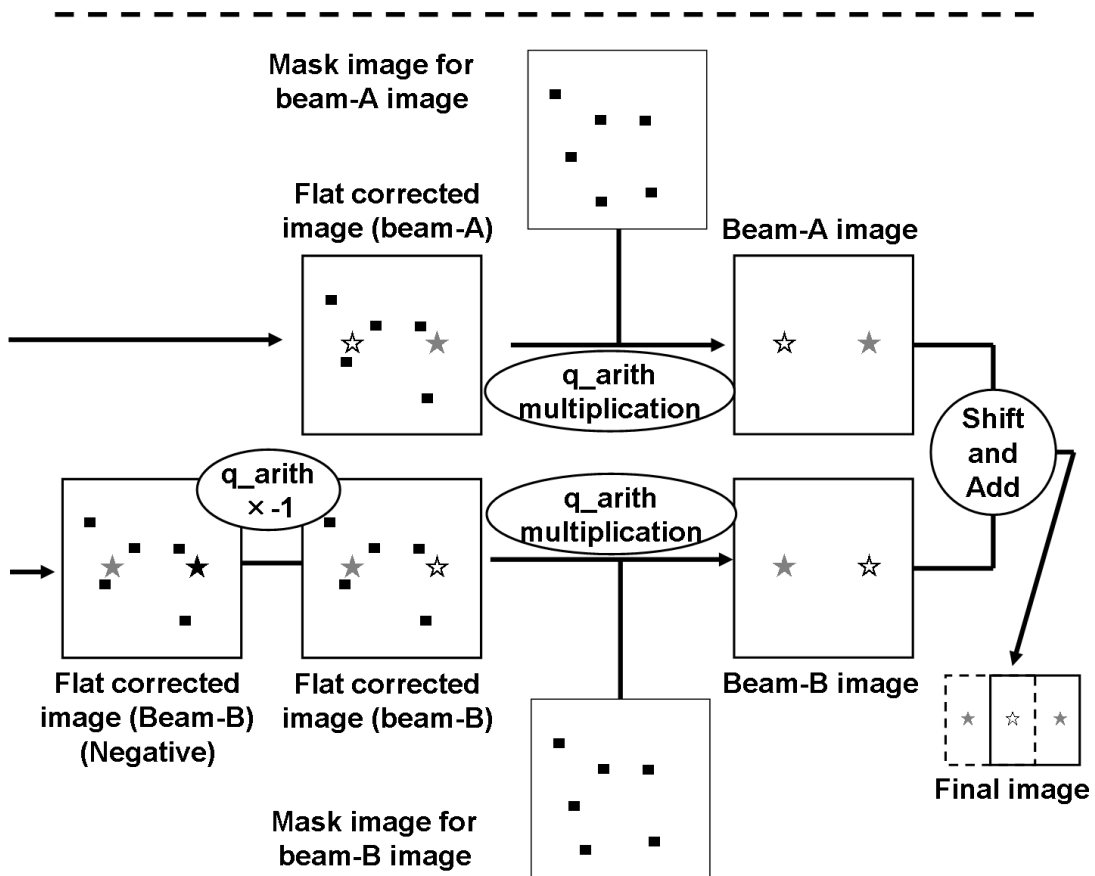
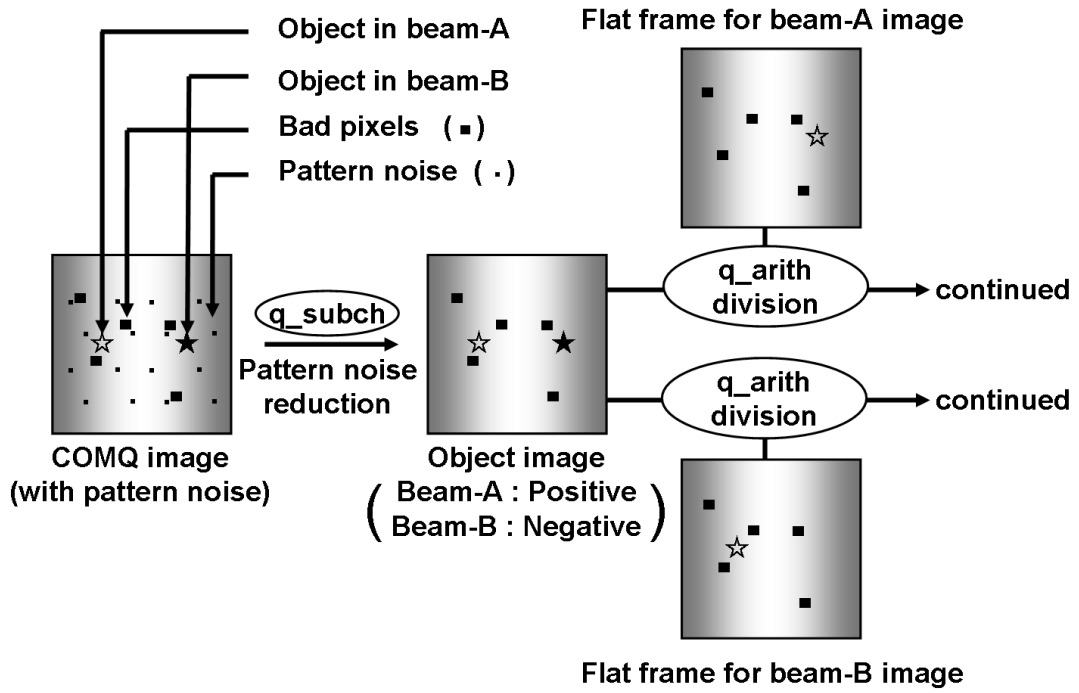


Figure 5.5: A method of photometry data reduction: The final images can be obtained by correcting for the flat to the COMQ images, masking bad pixels, and adding all the images by shifting them (if necessary).

5.1.6 Misc.

Correction for Sky Levels

We suggest verifying quality of sky-level subtraction carefully because our experience warns that gradual level changes over the images tend to remain. This would be due to, e.g., high background emission and/or the presence of clouds even for blank sky you assumed. If such a residual pattern is recognized, you should consider to apply another sky-subtraction by making sky images to remove them.

Conversion of the Image Intensity Unit: Getting Flux Density Scale

After completing the final image construction, you probably want to convert unit 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 brightness of standard star at the filter wavelength you observed. It is highly recommended to select standard stars which have been observed with airmass close to that of the target. Since accurate measurements of flux densities of the standard sources have not been generally well-established at MIR bands, except for whole the N-band, you probably must obtain fluxes for the desired filters by integration spectral data provided in the Cohen et al. paper over the desired wavelength range. As for the details of the filter properties, such as the center wavelength, and the passband characters, please refer to the COMICS web page of <http://canadia.ir.isas.jaxa.jp/comics/open/guide/index.html> where you can find other useful links, e.g., the data base by Dr. Cohen et al.

For simplicity at your calculations, you may integrate spectrum over the filter bandpass adopting a rectangular shape, as the first order of approximation. However, it is desired to 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 e.g., calculating summation of the counts over a rectangular area containing the standard star. This can be done with 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 unit, 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. Conversion to T_b 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. A 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 shift-and-add described in §5.1.5. You, however, should be careful to adopt the latter method; make sure validity of the image registrations across the filters by considering nature of the sources (i.e., emission mechanism). Moreover, we suggest making sure possible position shifts across the images taken at different wavelength and/or different observing time (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 straight-forward and one of the reliable methods 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 residual global pattern of the sky background AND the sky background level is considerably brighter than the source of the interests.

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

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

5.2.1 Preparing for a Dark

For reducing spectroscopic data as well, we have to use not only the science data but also the dome flat images. For this purpose, the users 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.

In this manual, let us suppose that we 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 use 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 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 contents — detector information — of each FITS files, see the Q_DETST parameter in the header. This key word is given by a number that is consisting with 0 and/or 1. The first digit indicates a flag of the imaging or photometry detectors, and the subsequent ones does status of the spectroscopy detectors. For example, 100110 indicates that you have obtained a photometry image plus spectroscopy ones taken with the detectors of 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 differs 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 image

For spectroscopy observations, we use dome flat images for making a flat. We suppose that you have obtained more than one flat images 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 to the scientific exposure because flat images usually change even with a slight shift of grating configurations. In this example, we firstly average a COMA image of COMA00051356.fits and subtract the dark. Since the flat image has Nexp =9, we have to multiply 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 dome flat made in the above contains un-removed gradient over the field of view, which is similar to those seen in the flat for the photometry. We, therefore, need to divide the image with that is convolved with a Gaussian filter with the same method as we did in §5.1.2.

```
c1> 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. Here the width of 20 pixels is recommended from our experience.

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

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

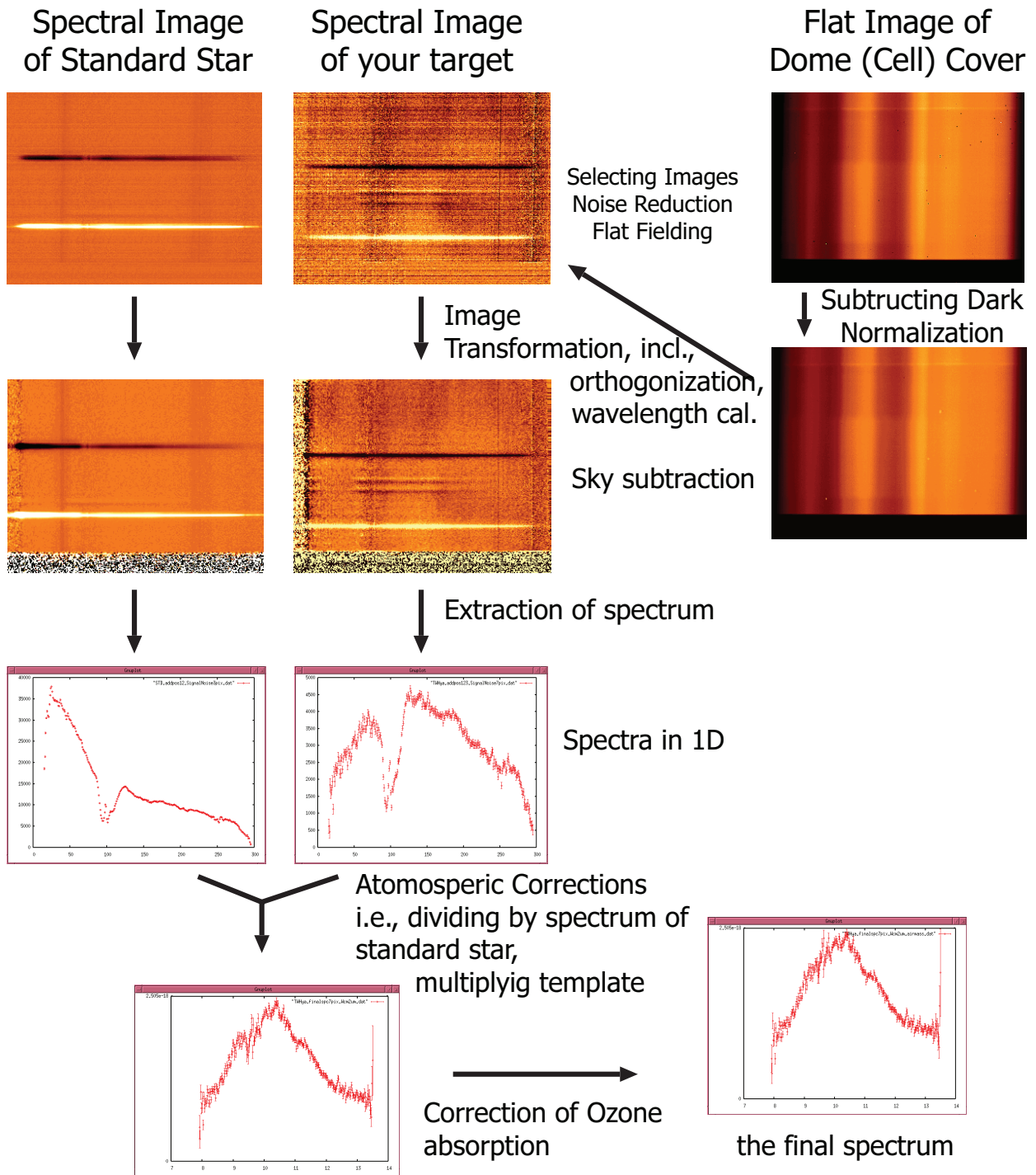


Figure 5.6: A method to reduce COMICS spectroscopy data: An example for a point-source with the low-resolution mode

5.2.3 Reduction of the Read-out Pattern Noise

For some observing modes that do not use all the detectors, e.g., medium-resolution spectroscopy at the N band, the 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 us 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 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),

```
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,

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

After that, we can 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

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

Adding All the Images

If part of your data have been acquired at the same position in the sky, you may consider to add all the appropriate images, which makes it much simpler for further data reduction process. If this is not the case, we suggest keeping them without adding at this stage. For example, let us assume that two spectral images files of `COMQ00051314.fits` and `COMQ00051316.fits` have been taken at the same positions in the plane-of-sky and that we 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 subsequent 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,

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

Orthogonalization of the Spectroscopic Images and Wavelength Calibration

After completing reduction of the read-out pattern noise, we have to transform the images to orthogonalize dispersion and spatial axes which will be used to extract spectra of the emission you desired. Furthermore, we 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 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, we have to 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, we 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 process simpler. In practice, we firstly make a sky image which 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). We now can 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)
```

The obtained 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, the 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 are not completely guaranteed.

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

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

This command calculates cross-correlation with data base 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 resultant fitting parameters are stored in `skyimage_d1f.res` and be checked with standard graphic packages such as GNU PLOT. In `skyimage_d1f.res`, the 3rd column indicates the extracted y -axis value, and 6th and 8th 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, we apply a linear function to the fitted parameters of A and B by least-square fitting, namely, we use fitting equations of $A = a_0y + a_1$ and $B = a_2y + a_3$ (see Figure 5.7). In conclusion, we are going to use parameter set of a_0 , a_1 , a_2 and a_3 to repeat the best-estimated wavelength constant axis.

2. Determination of Spatial Axis

To determine spatial axis, we use images of standard star(s) (or any point-source image should work as well) that have been subtracted detector response and divided with flat. We 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 are thought to be stable during an observation, one 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 a task `q_startrace`.

Let us suppose that the image contains the standard star at the positions of $y = 62 - 92$,

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

this searches peak positions and store 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 FAQ may help you. The 2nd column of `obj51308s_f.str` stores the x -coordinate and the 10th y of the peak, the user should plot them to

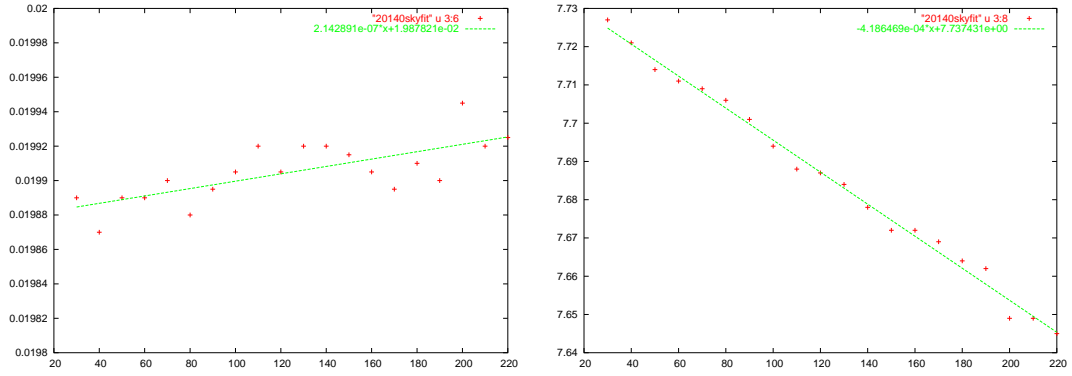


Figure 5.7: A result of wavelength and pixel relationship determination and a plot of linear function fitting

verify the results (please use GNUPLOT etc); see an 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 fit. If this is the case, we suggest excluding such points before fitting. For the medium resolution observations, the above procedure 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 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 5 detectors. If you try to fit with the 2nd order equation, you would not be able to get a reasonable results 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.

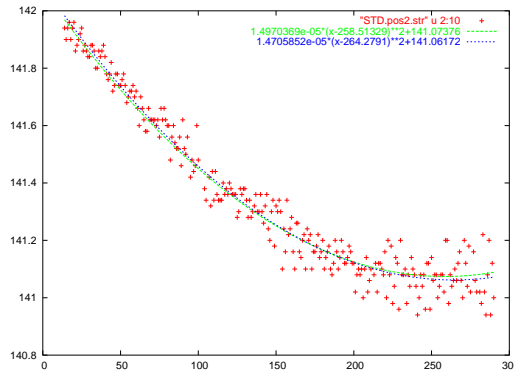


Figure 5.8: An example of space-constant line and the fitting results

3. Image Transformation

Using the above results, we can transform the images with IRAF; the task `geomap` finds the transformation parameters and the task `geotran` transforms your image. First, let us 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 arbitrary defined. However, one 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 which are transformed outside the field of view. In the above example, we obtained the following parameters for the medium resolution at the N band.

```
a0=2.669213e-07
```

```

a1=1.990141e-02
a2=-3.758642e-04
a3=7.578833e+00
b0=1.5874755e-05
b1=195.9228

```

Now we can issue the IRAF `geomap` command at the directory where `login.cl` file exists,

```
cl
```

which starts IRAF, and you should move to the directory for the analysis. Using `epar` command, we set input parameters for `geomap` as shown in the below.

```

                I R A F
      Image Reduction and Analysis Facility

PACKAGE = immatch
      TASK = geomap

input      =          The input coordinate files
database=  transpar  The output database file
xmin      =          1. Minimum x reference coordinate value
xmax      =          320. Maximum x reference coordinate value
ymin      =          1. Minimum y reference coordinate value
ymax      =          240. Maximum y reference coordinate value
(transfo=  )          The output transform records names
(results=  )          The optional results summary files
(fitgeom=  general)  Fitting geometry
(funcutio= polynomial) Surface type
(xxorder=  3)        Order of x fit in x
(xyorder=  3)        Order of x fit in y
(xxterms=  half)    X fit cross terms type
(yxorder=  3)        Order of y fit in x
(yyorder=  3)        Order of y fit in y
(yxterms=  half)    Y fit cross terms type
(reject =  INDEF)    Rejection limit in sigma units
(calctyp=  double)  Computation type
(verbose=  yes)      Print messages about progress of task?
(interac=  no)      Fit transformation interactively ?
(graphic=  stdgraph) Default graphics device
(cursor =  )        Graphics cursor
(mode =  ql)

```

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 above command has prepared for the image transformation by setting the parameters; we have to have image files of the target(s), standard star(s) that must be subtracted the detector response and be divided by the flat, sky image (skyming) 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 effect of bad pixels in the transformed image. Since IRAF accepts input files with an extension of ".fits" only, we 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., spatial axis should be horizontal and dispersion one vertical) using tasks of `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 4D image only. We therefore convert the `geotran` output files into 4D image.

Let us start to 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 4D image, and the second one is the command to multiply on `q_startrace`.

Subsequent step is to check the results of dispersion axis, which can be done through transforming the sky image (`skyimage_d1f.fits`) and apply `q_sky_nlow` on the results. One should make sure whether or not 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 ideal if all the sky level variations could be removed by means of the chopping. However, in practice, chopping frequency tends to be slower than sky level variations. If this is the case, the user should see some residual pattern in the chopping subtracted images. Since we have completed to orthogonize the images (i.e., x-axis: wavelength, and y-axis: space), we 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`.

```
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 we can extract the desired spectra from the obtained 2D data. You may want to 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 we can measure the noise level through standard deviation of the emission free region (i.e., the blank sky region). For instance, let's suppose that we have an image which accommodates the source at $y = 72$. In this case, we 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 6th column of `obj51308s_ftc.noise1pix` indicates noise level for each x-value which corresponds to the wavelength. Next we can extract spectra of the source at any y position. Assuming that we want to have a width of 3 pixels (i.e., $y = 71 - 73$), the following gives you 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 the above procedure calculates averaged value over the specified pixel range. With this reasoning, we have to 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
```

We 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 in the 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 them. The spatial registration between images are normally done with the other object(s) seen in the slit viewer or measuring peak position at the spectroscopy side. Recall that a pixel corresponds to $0.13''$ for the slit viewer, i.e., the imaging side, whereas $0.165''$ for the spectroscopy side. Because of its 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 them 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 we have obtained are represented by count with respect to pixel numbers, which should be converted into wavelength unit. 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), we will use the parameters. Now we have $a = 1.9900 \times 10^{-2}$ 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

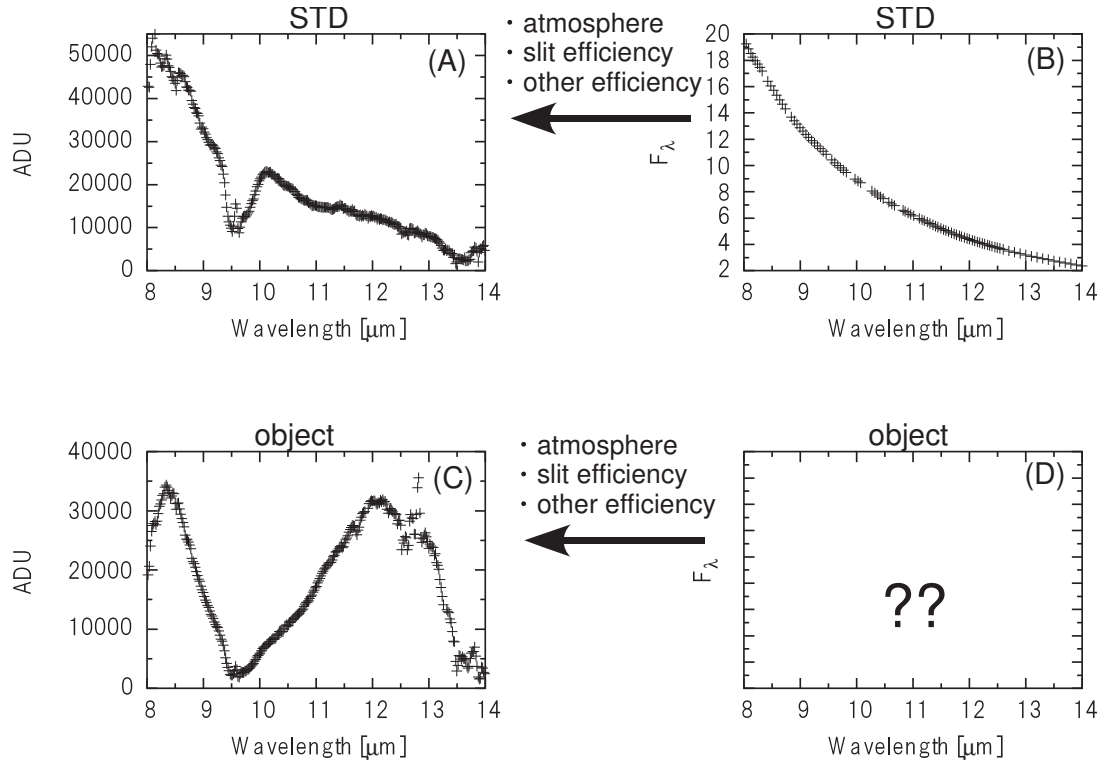


Figure 5.9: A method to obtain spectrum of the object using the standard star. The spectrum obtained by (A) \div (B) is a spectrum of the atmosphere including the instrumental response. If you observed standard star and the object(s) with the identical slit efficiency, (D)=(C) \div [(A) \div (B)] should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

Dividing with the Spectrum of 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, we can obtain an "atmosphere-free" spectrum by dividing the source spectrum with that of the standard star and multiplying the "atmosphere-free true" spectrum (hereafter "true" spectrum) of the standard star (See the spectrum of (C) \div (A) \times (B) in Figure 5.9).

"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 slit spectroscopy method, all the photons do not fall onto the slit. We therefore have to consider a slit efficiency. In general, slit efficiency depends on not only the incident angle but also wavelength dependency because size of stellar images varies with the 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 for not only diffuse emission but also point-like sources. We should also 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, we have to have a multi-color data set. Let us suppose 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, we 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. Lastly, it is worth pointing out that slit efficiency correction works well with a linear correction for many of the cases.

5.2.9 Diffuse Emission

In principal, the user can analyze spectral line images of the diffuse emission with the same manner as done for the point-like sources. Namely, one can extract the desired spectrum at arbitrary positions. However, if the emission is so extended over 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 must be wise to apply the correction directly to the 2D image before extracting the spectra. For this purpose, we firstly 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 function of the wavelength, $F(\lambda)$ can be applied to the desired row by inverting the equation as for the wavelength. Subsequently we obtain a relationship between pixels and wavelength to make 2D images which 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, 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 $9.6\mu\text{m}$

You will realize that it is not easy to divide source spectra by that of standard stars around $9.3 - 10 \mu\text{m}$ 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 at very close airmasses. Our experience suggests it is likely that airmass mismatching would be responsible for wavy artifact, or/and bump/dip around $9.3 - 10 \mu\text{m}$.

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, i.e., 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 the other observatories', e.g., submm telescopes). If you cannot, the standard values would be the good starting point, and you will be an expert of the $10 \mu\text{m}$ 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 4 images. The reason to employ chopping and nodding is to reduce 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 rather low background noise level originated from system itself, 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 typical residual pattern at the N11.7 band is $35 \text{ mJy arcsec}^{-2}$; 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 ?

We must admit that the current system often causes such an artifact. Unfortunately, it's almost impossible to use such images for scientific analysis. The only way to reduce such "ghost" is to configure your imaging filter at through "hole" position (but please be aware that this does not guarantee perfect removal of ghost.) for future spectroscopic observations. When you observe with through hole, please keep in mind that you have to decrease gain so as not to saturate the slit viewer.

5.3.4 I got only errors when I am transforming 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 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 geomap procedure.

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

This is perhaps due to 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 so 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); try to run q_startrace again after removal of such points.