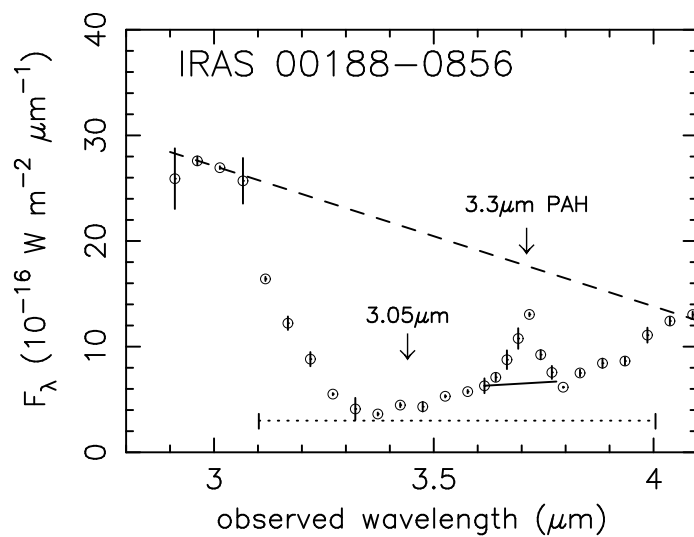
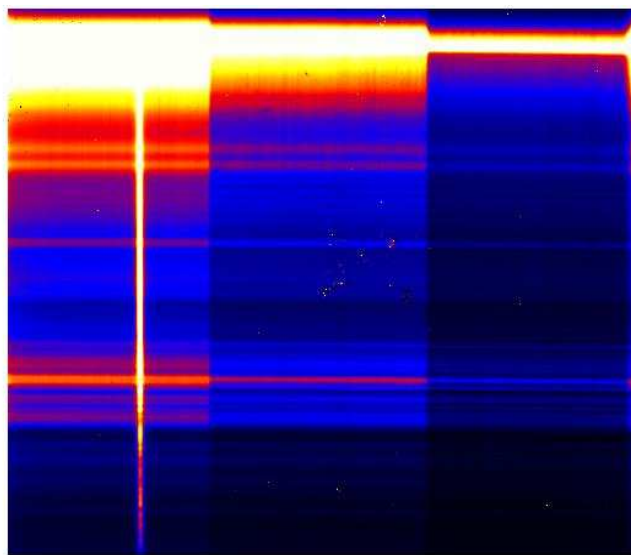


# IRCS グリズム分光データ解析ガイド (第二版)

今西昌俊 (国立天文台・光赤外研究部)

2008年4月10日



# 1 はじめに

本マニュアルは、IRCS の低分散グリズム分光データ解析の初心者が、スペクトルを得るまでの手順の基本的事項をまとめたものである。しかし、single-order の赤外線スペクトルの標準的な解析手法を記述したものであるため、例えば、すばる COMICS、CISCO/OHS、UKIRT CGS4、あかり衛星の IRC といった装置で取られた赤外線分光データにも応用できる。

IRCS グリズムのような赤外線低分散分光は、一般に暗い天体を観測対象とするため、達成される S/N 比は、連続線に対して 10–30 程度になるのが一般である。本マニュアルの説明は、そのようなスペクトルの解析に最適化されている。一方で、赤外線の高分散分光は、比較的明るい天体の、高い S/N 比のスペクトルを得る場合が多い。そのような高い S/N の高分散赤外線スペクトルの解析には、本マニュアルの記述に加えて、さらにすべきことがいくつかある。それらについては簡単にコメントするにとどめる。

IRCS の装置仕様に関しては、他の文献に詳細な記述があるので、ここでは重複して書くことはせず、必要最低限のみを説明するにとどめる。本マニュアルは、実際の解析の際に行なうコマンドの説明が中心である。それに加えて、他のテキストには載っていないような、私自身のこれまでの解析経験からくる注意点、コメントなどを多く記述するようにした。読者が実際の解析の際に疑問に感じた場合、戸惑った場合に、少しでも参考になれば幸いである。また、個人的見解も含まれており、それが正しいかどうかは、解析の際に、読者自身もしっかり考え、理解してから実行に移して欲しい。

本マニュアルの初版は、2006 年度のすばるデータ解析講習のために作成した。参加者からのいくつかのコメントを受けて改訂したのが、本第二版である。口頭のみで説明し、説明を省いた事項もある。わかりにくい点、間違った記述、これを書いて欲しかったといった要望があれば、是非、積極的にフィードバックをかけて欲しい。

2008 年 4 月 10 日

今西昌俊

181-8588 東京都三鷹市大沢 2-21-1

国立天文台 光赤外研究部

E-mail: masa.imanishi "at" nao.ac.jp

# 目次

1	はじめに	2
2	赤外線分光観測の特徴	4
3	IRCS	5
4	赤外線分光観測の流れ	5
5	赤外線分光データの解析の流れ	6
5.1	概要	6
5.2	実際の解析手順の詳細	7
5.2.1	IRAFの準備	7
5.2.2	A-B差し引き	8
5.2.3	分光フラットで割る	12
5.2.4	変なピクセルの補正	15
5.2.5	信号の抽出。AとBの足し合わせ	17
5.2.6	線形性の補正(今回は行なわない)	23
5.2.7	波長較正	25
5.2.8	Airmass補正(今回は行なわない)	27
5.2.9	必要なら binning	28
5.2.10	天体を標準星で割る	29
5.2.11	標準星の元々のスペクトルをかける	29
5.2.12	フラックス補正	30
5.2.13	誤差の評価	32
5.2.14	輝線、吸収線の強さの測定	35
6	最後に	35

## 2 赤外線分光観測の特徴

赤外線分光観測は、可視光の分光観測と基本的に同じである。しかし、

1. 可視で用いられている CCD は赤外線 ( $1\mu\text{m}$  以上) に感度がないため、現在は HgCdTe(水銀カドミウムテルル) や、InSb(インジウムアンチモン) でできた検出器を用いる。
2. 地球大気や望遠鏡からの背景放射が大きく (図 1)、可視光線に比べて、観測が難しい。背景放射から来る大きなポアソンノイズ<sup>1</sup> により、スペクトルの S/N が悪くなる。また、検出器には、一回に測定できる最大の光子数の上限がある (full-well) ため、大きな背景放射は、一回あたりの露出時間を短くすることを要請する。その結果、データのフレーム数が多くなり、観測の効率 (オーバーヘッド) が悪くなる。
3. 解析の際の様々な補正の精度が、可視光に比べて粗い<sup>2</sup>。しかし、赤外低分散分光の場合、補正の不定性よりは、上記のノイズが最終的な S/N を決める場合が多い。

などの違いがある。

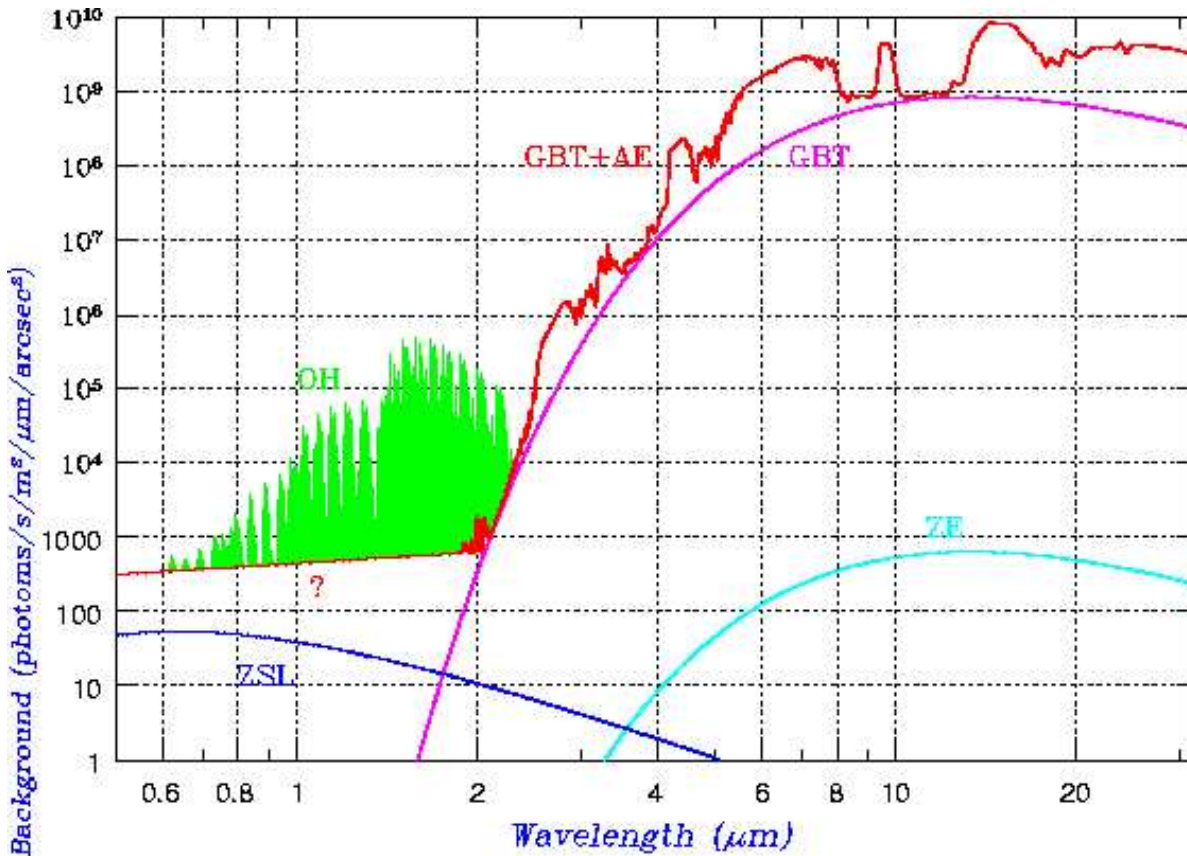


図 1: 可視光から赤外線にかけての背景放射。赤外線の波長が大きくなるにつれて背景放射が急激に増大する。AE: 地球大気の放射 (220-273K)。GBT: 望遠鏡からの放射 (273K)。OH: OH 夜光。ZSL: 太陽光の黄道面の塵による散乱光。ZE: 黄道光 (塵からの熱放射)。

可視光は、ほぼすべての波長で、地上から観測することができるが、赤外線には、地球大気による吸収によって、地上に到達しない波長帯がある。図 2 は、赤外線の地球大気の透過の様子を示したものである。地上からの赤外線分光観測は、透過率の高い波長帯に限定される。

<sup>1</sup> 地上観測の際のノイズ源は、大体以下の 3 個に分類される。可視光で、明るい天体を高分散分光する場合などは、天体の信号のポアソンノイズが主要なノイズ源となる傾向がある (signal-limited)。赤外線での高分散分光の場合は、検出器の読み出しノイズが主要なノイズ源になりやすい (readout-limited)。それに対して、赤外線低分散分光では、一般に背景放射のポアソンノイズが主要なノイズ源となる (background-limited)。これらのノイズの相対的大きさは、天体の明るさ、装置の性能や波長分解能、大気の状態などで変化する。どのノイズが支配的かを各自チェックし、観測パラメーターを最適化する必要がある。

<sup>2</sup> 私は可視、赤外、電波の地上観測経験があるが、電波観測に比べると、精度は良いと思う。

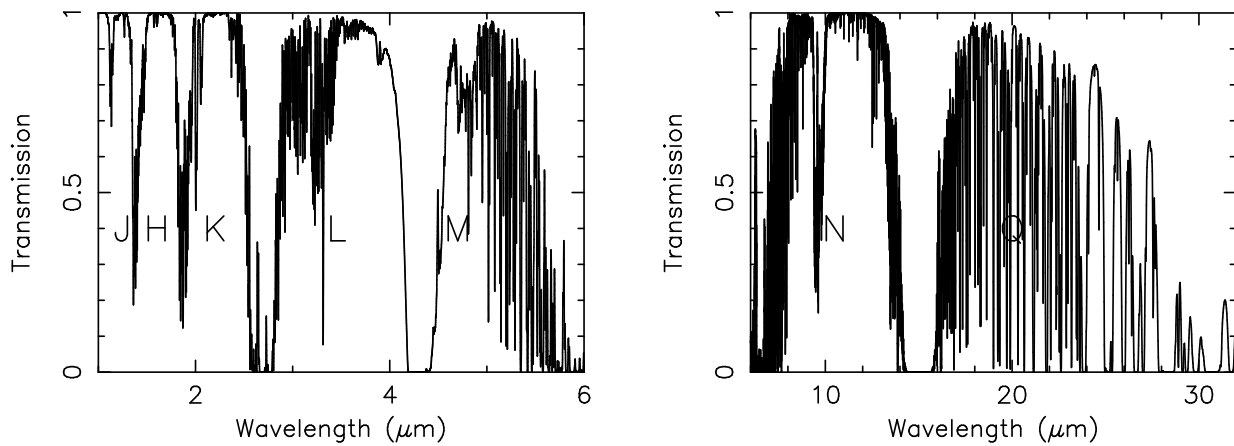


図 2: 地球大気による赤外線透過の様子。地球上から観測する場合、いくつかの大気の窓 (J,H,K,L,M,N,Q バンド) に波長が限定される。<http://www.jach.hawaii.edu/UKIRT/astronomy/utils/atmos-index.html> から引用。この透過率は、観測時の大気の状態、特に、観測地点より上空にある大気中の水蒸気量の総和によって変化する。本図は、水蒸気量が比較的小さい場合である。水蒸気量が増えると、全体的に透過率が下がるが、本図で透過率が悪い波長の下がり方が特に大きい。

### 3 IRCS

すばる IRCS は、赤外線撮像、分光観測のできる装置である。

<http://www.naoj.org/Observing/Instruments/IRCS/index.html> に詳細な記述がある。分光には、single-order の低分散分光と、cross-dispersion の高分散分光モードがあるが、ここでは初心者を対象に、single-order の低分散赤外線分光のデータ解析の説明に重点を置く。cross-dispersion 分光の解析に関しては、例えば、HDS のマニュアルや、<http://iraf.noao.edu/docs/spectra.html> の A User's Guide to Reducing Echelle Spectra With IRAF, Daryl Willmarth and Jeannette Barnes, May 1994 (英語) などが参考になる。

### 4 赤外線分光観測の流れ

single-order の低分散赤外線分光観測は、一般に以下のような流れで行なわれる。空間的にコンパクトな天体の場合を考える。

1. スリット方向の 2 点 (位置 A,B) で天体を観測。赤外線検出器の (非破壊) 読み出し回数 (NDR)<sup>3</sup>、各位置での露出の回数 (COADD) を設定。各位置での総滞在時間は、地球大気の時変動の影響が大きくなりすぎないように抑えられる。
2. ABBA の順で何セットかのデータを取得
3. 目的天体 (一般に暗い) の前後に、明るい標準星 (主に G 型主系列星) を観測。平均 Airmass (図 3) の違いは、0.1 以下にするのが一般。図 2 にあるように、赤外線では、地球大気の透過率が波長によって大きく変化し、かつ、観測時の天候条件によっても異なる。元々のスペクトルの形状がわかっている標準星を観測し、目的天体を標準星で割り算してやることにより、補正できる。

本書では、この観測手順に乗っ取って得られた赤外線低分散スペクトルの解析手法を説明する。

<sup>3</sup> 可視の CCD とは異なり、赤外線検出器の場合は、たまった電荷を非破壊 (NDR; Non Destructive Read) で、複数回読み出すことができ、読み出し回路のノイズを減少させることができる。NDR の最大の回数は、読み出し速度と露出時間によって決まる。なお、IRCS の場合、NDR の回数は最大になるようにデフォルトで設定されている。ただし、本書で扱う L バンド (2.8–4.2 $\mu\text{m}$ ) 低分散分光の場合は、短波長側を除けば、背景放射のポアソンノイズの方が大きくなる。従って、NDR を増やして読み出し回路ノイズを減らしても、全体のノイズはあまり減らず、一方で観測効率の低下をもたらす。NDR は適度な回数に、自ら設定する必要がある。

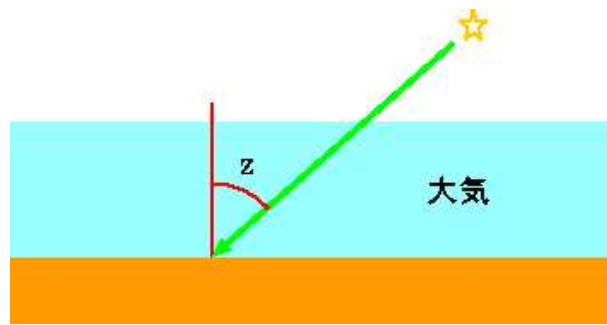


図 3: Airmass の概念。上図で、 $1/\cos(z)$  が Airmass と定義される。天頂を観測すれば、最も Airmass が小さい (=1)。Airmass が大きいと、より多くの地球大気を通過することになり、吸収が大きくなる。と同時に、赤外線では、地球大気の影響放射が増大する。従って、小さな Airmass の状態で観測することが望ましい。天体の座標、観測地点、及び、観測時刻により、実際の天体観測では、Airmass が 1 より大きい状態で観測することになる。赤外線観測では、Airmass が 2 以上で観測することはほとんどない。

## 5 赤外線分光データの解析の流れ

### 5.1 概要

解析の流れは、以下のような感じである。

1. 位置 A – 位置 B のデータの差し引き
2. 変なピクセルの補正
3. 分光フラットで割る
4. 信号の抽出。A と B の足し合わせ
5. 線形性の補正 (今回は行なわない)
6. 波長較正
7. Airmass 補正 (今回は行なわない)
8. 必要なら binning
9. 天体を標準星で割る
10. 標準星の元々のスペクトルをかける
11. フラックス補正
12. 最終スペクトルのできあがり
13. 物理状態の導出、科学的議論

赤外線分光観測では、波長が長くなるほど、背景放射が大きくなり、観測が困難になる。そのため、研究例も少なく、補正の道具もあまり整っていない傾向がある。従って長波長の分光データの解析ほど難しく、それができれば、一般には短波長の赤外線分光データの解析はできるようになると考えられる。今回は、波長  $3\text{--}4\mu\text{m}$  の L バンドの分光データの解析を説明する。天体は、赤外線銀河 IRAS00188–0856 (Imanishi et al. 2006 ApJ 637 114) である。

## 5.2 実際の解析手順の詳細

解析には、IRAF というソフトを用いる。IRAF は、NOAO(米国の光学赤外線天文学観測所) から配付されている、汎用のデータ解析ソフトであり、“<http://iraf.noao.edu/>” がホームページ (英語) である。使い方の詳細に関しては、天文情報処理研究会発行の IRAF クックブックや、“<http://iraf.noao.edu/docs/spectra.html>” の A Beginner’s Guide to Using IRAF (IRAF Version 2.10), Jeannette Barnes, August 1993(英語) を参照するのがよい。

IRAF を用いた single-order の赤外線スペクトルの解析に関しては、“<http://iraf.noao.edu/docs/spectra.html>” の A User’s Guide to Reducing Slit Spectra with IRAF, Phil Massey, Frank Valdes, Jeannette Barnes, April 1992 (英語) に説明がある。日本語文献としては、天文情報処理研究会 可視・近赤外域での天文データ処理手引 (1997 年) の 67 ページ (村山卓著) がある。

以下では、基本的にこれらの文献の記述に沿って、解析をすることにする。実際の細かいコマンドで一部異なる点があるが、データに対する操作は本質的に同じである。

### 5.2.1 IRAF の準備

十分な性能の Linux PC を持っている人は、自分で PC 用の IRAF をインストールして、そこで解析することができる。実際、私はそうしている。<http://iraf.noao.edu/projects/pc-iraf/pc-iraf.html> に情報がある。ここでは、ana 系コンピューター (ana03.adc.nao.ac.jp など) 上の IRAF で解析することにする<sup>4</sup>。ana 系コンピューターで IRAF を使えるようにするには、ホームディレクトリに行って、“mkiraf”, “y”, “xgterm” を入力する。

```
ana03 /home/imanshms> mkiraf
Initialize uparm? (y|n): y
-- initializing uparm
Terminal types: xgterm,xterm,gterm,vt640,vt100,etc.
Enter terminal type: xgterm
A new LOGIN.CL file has been created in the current directory.
You may wish to review and edit this file to change the defaults.
```

これで、login.cl というファイルができる。login.cl を emacs か何かで開いて、

```
#set imtype = "imh"
```

となっていれば、

```
set imtype = "fits"
```

に変換することをお勧めする。これは必須ではないが、この操作により、fits 形式でファイルを取り扱えて便利である。これをしないと、IRAF 形式のファイルが作られ、扱いがやや面倒。ちなみに、ana 系では自動的にこのようになっているので、変更の必要はない。

その後、“xgterm &” で xgterm を開き、そこで “cl” とすれば、IRAF が立ち上がる。

NOAO Sun/IRAF Revision 2.12.1-EXPORT Wed Jul 10 13:33:04 MST 2002

This is the EXPORT version of Sun/IRAF V2.12 for SunOS 4 and Solaris 2.8

---

<sup>4</sup> IRAF のバージョンが異なると、動いていたコマンドが動かなくなったりすることがあるので、注意が必要である。

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

```
apropos      guiapps.    lists.      plot.       system.
color.       guidemo.    mscred.     proto.      tables.
dataio.      images.     noao.       softools.   utilities.
dbms.        language.  obsolete.   stsdas.
```

cl>

のような画面が出てくる。

Unix の別の Window(kterm) で、画像表示ソフト “ds9” (“saoimage” でも良い) を立ち上げる。

### 5.2.2 A-B 差し引き

観測データは、作業ディレクトリの下に”fits” というディレクトリを作り、そこに格納することにする。今回の解析に必要なデータは、ana03.adc.nao.ac.jp の/mfs02a/imanshms/SubaruData/IRCS.tar.gz (あるいは、http://optik2.mtk.nao.ac.jp/subaru\_red/IRCS の下の” サンプルデータ (分光)”) にまとめてある。各自の作業ディレクトリの下に “mkdir fits” で fits というディレクトリを作り、  
“cp /mfs02a/imanshms/SubaruData/IRCS.tar.gz fits/” でそこにファイルを持ってきて、  
“tar zxvf IRCS.tar.gz” を実行すれば、fits ファイルができる。”more IRCA00070351.fits” などとすれば、天体名や観測日時、観測モード、積分時間などが分かる。

```
OBJECT = 'HR72          ' /
```

```
DATE-OBS= '2002-08-19   ' / UT date of Observation (yyyy-mm-dd)
```

“cd ..” で、元の作業ディレクトリに戻しておく。

まず、得られた生画像をしてみる。明るい標準星から見る。IRCA00070351-IRCA00070362 が、標準星 HR72 のデータである。display というコマンドを用いるが、見やすい表示にするために、いくつかのパラメーターを設定する。

cl> epar display

I R A F

Image Reduction and Analysis Facility

```
PACKAGE = tv
```

```
TASK = display
```



```

image = IRCA00070351.fits image to be displayed
frame = 1 frame to be written into
(bpmask = BPM) bad pixel mask
(bpdispl= none) bad pixel display (none|overlay|interpolate)
(bpcolor= red) bad pixel colors
(overlay= ) overlay mask
(ocolors= green) overlay colors
(erase = yes) erase frame
(border_ = no) erase unfilled area of window
(select_ = yes) display frame being loaded
(repeat = no) repeat previous display parameters
(fill = no) scale image to fit display window
(zscale = no) display range of greylevels near median
(contras= 0.25) contrast adjustment for zscale algorithm
(zrange = no) display full image intensity range
(zmask = ) sample mask
(nsampl= 1000) maximum number of sample pixels to use
(xcenter= 0.5) display window horizontal center
(ycenter= 0.5) display window vertical center
(xsize = 1.) display window horizontal size
(ysize = 1.) display window vertical size
(xmag = 0.5) display window horizontal magnification
(ymag = 0.5) display window vertical magnification
(order = 0) spatial interpolator order (0=replicate, 1=linear)
(z1 = ) minimum greylevel to be displayed
(z2 = ) maximum greylevel to be displayed
(ztrans = linear) greylevel transformation (linear|log|none|user)
(lutfile= ) file containing user defined look up table
(mode = ql)

```

xmag = 0.5、ymag = 0.5、zscale = no、zrange = no にする。元々の設定では、違う値になっているはずである。最初の二つは、1024×1024 ピクセルの画像全体を表示させるため、後の二つは、信号のレンジを手動で調整できるようにするためのものである。それ以外は、いじらなくてよい。例えば、xmag のところに行き、“0.5”、“:q” を入力すれば、変更した値が保存される。間違っても変な値を入れてしまった場合は、“(スペースキー)を押すか、cl> で、“unlearn display” とすれば、元の設定に戻る。

その後、

```
cl> display fits/IRCA00070351.fits 1 z1=0 z2=30000
```

で、位置 A での標準星のデータが表示される (図 4 左)。標準星は明るいので、上下方向に分散された信号が見えている。天体は一般に、非常に暗いか見えない。様子の異なる信号が、左右に三分割されている。これは、IRCS では、左から、0.9 秒角、0.6 秒角、0.3 秒角のスリット幅に分割されており、幅の広いスリットほど、より多くの背景放射が入ってくるので、信号が大きくなっている。今、標準星の信号は 0.9 秒角のスリットに乗せているので、一番左のところに信号がある。

参考までに、スリット幅は、観測時の星像の大きさ (シーイング) や、必要な波長分解能に応じて決めるものである。細かいスリットだと、背景放射を減らせるが、シーイングが悪い場合や、望遠鏡のトラッキング誤差が

大きい場合は、多くの信号を失い、S/N比が悪くなる。シーイングが良い場合は、細いスリットを用いるのが良い。

```
cl> display fits/IRCA00070352.fits 1 z1=0 z2=3000
```

位置 B での標準星のデータが表示される (図 4 右)。左右に動いていることがわかる。

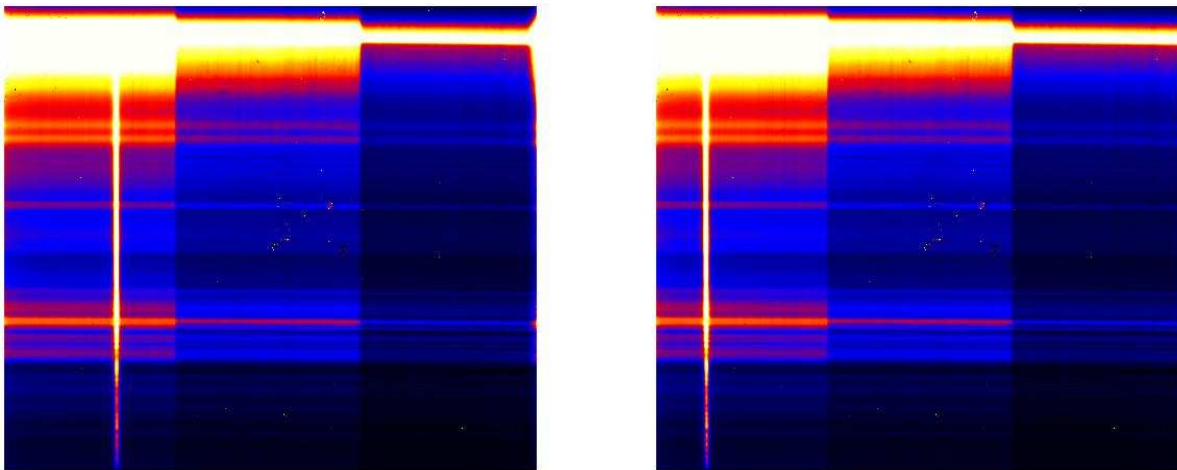


図 4: 左: 位置 A での画像。右: 位置 B での画像。

この画像では、背景放射の上に天体からの信号が乗っている状態である。A-B を行なえば、背景放射は差し引かれ、天体の信号だけが残る。実行するには、Unix プロンプトで、“mkdir ana” で、作業ディレクトリの下に ana を作り、

```
cl> imarith fits/IRCA00070351 - fits/IRCA00070352 ana/HR72_351_352
```

```
cl> display ana/HR72_351_352 1 z1=-1000 z2=1000
```

で、非常にきれいに、標準星の信号が見えていることがわかる (図 5 左)。

上で “ana/HR72\_351\_352” は、出力ファイル名の指定である。好きな名前でもよいが、内容がわかるようなものにすべきである。ここで、正式なファイル名は IRCA00147131.fits などだが、“.fits” を省いても認識してくれる。また参考までに、z1、z2 の値を変えると、見え方が変わるので、試して欲しい。自分が見やすい値に設定するのがよい。

この A-B データを解析していくことになるが、実際に使用する部分は左の 3 分の 1 だけで、それ以外の部分は使用しない。ハードディスクの無駄使いを避けるために、必要な部分のみを切り抜くことにする。それには、IRAF で、

```
cl> imcopy ana/HR72_351_352[1:340,40:900] ana/HR72a
```

を実行する。括弧内の前半が X 座標、後半が Y 座標の切り取り範囲に相当する。データを表示して見よう。

```
cl> display ana/HR72a 1 z1=-1000 z2=1000
```

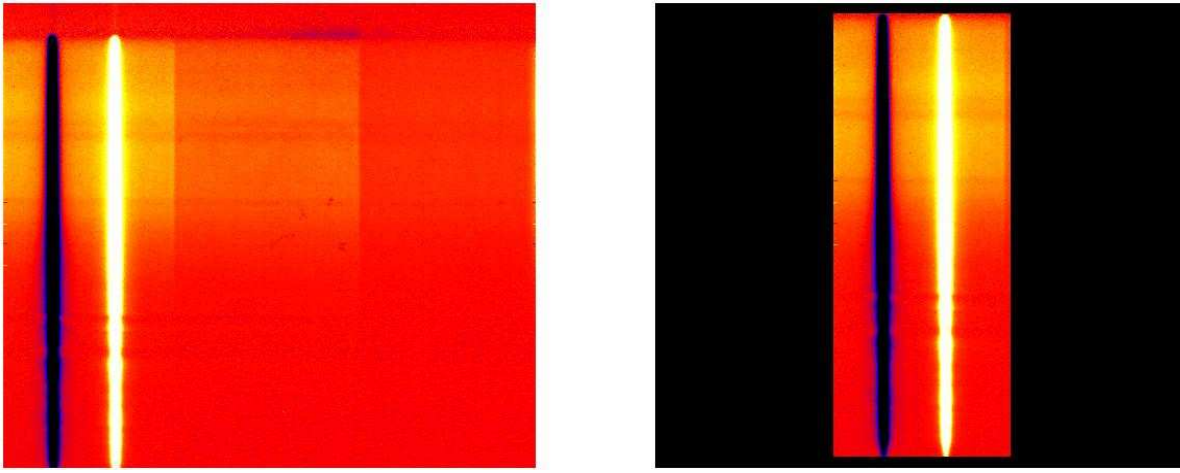


図 5: 左:A-B での標準星の分光データの生画像。標準星の信号以外の部分は、ノイズの範囲内で、ゼロ値になるはずである。右:必要な領域のみを切り出した画像。

必要な部分のみが切り取られていることが分かる (図 5 右)。

これで、解析すべき元となる A-B データを作ることに成功した。ただし、実際の観測では、A-B のセットを、1 回ではなく、数回行なう。従って、すべての A-B フレームを作り、それらを足し合わせることになる。HR72 の場合は、目的の赤外線銀河の前後に観測しており、IRCA00070351 - IRCA00070362 の 12 フレーム (A-B は 6 セット) と、IRCA00070413 - IRCA00070420 の 8 フレーム (A-B は 4 セット) のデータがある。今回は、前半の標準星のデータセットを用いることにする。

手動でする場合は、

```
cl> imarith fits/IRCA00070351 - fits/IRCA00070352 ana/HR72_351_352
cl> imarith fits/IRCA00070354 - fits/IRCA00070353 ana/HR72_354_353
cl> imarith fits/IRCA00070355 - fits/IRCA00070356 ana/HR72_355_356
cl> imarith fits/IRCA00070358 - fits/IRCA00070357 ana/HR72_358_357
cl> imarith fits/IRCA00070359 - fits/IRCA00070360 ana/HR72_359_360
cl> imarith fits/IRCA00070362 - fits/IRCA00070361 ana/HR72_362_361
```

を実行する。これらを組み合わせて S/N の良い標準星のデータを作る。ここでは、和を取ることにする。平均を取ってもよいが、定数の違いがあるだけで、本質的には同じである。

Unix プロンプトで、"emacs HR72.list" を実行し、HR72.list という名のファイルを作る。HR72.list は、以下の 6 行からなる。

```
ana/HR72_351_352.fits
ana/HR72_354_353.fits
ana/HR72_355_356.fits
ana/HR72_358_357.fits
ana/HR72_359_360.fits
ana/HR72_362_361.fits
```

上で、".fits" は省略してもよい。そして、

```
cl> imsum @HR72_list ana/HR72_351_362
```

を実行し、標準星のデータを組み合わせる。そして、

```
cl> imcopy ana/HR72_351_362[1:340,40:900] ana/HR72_1
```

によって、必要な領域だけを切り取る。

コメントであるが、A-B のデータを解析するので、可視のデータ解析でよく出てくるダークやバイアスの差し引きは不要となっている。

また、IRAF スクリプトを作り、これら後半の一連の動作を一気に実行させることも可能である。  
/home/imanshms/gawk の下に、“subsum\_fits.cl” という IRAF スクリプトを作ったので、利用してもよい。このファイルを各自の作業ディレクトリの下に持ってきて、“mkdir sub” で sub ディレクトリを作り、

```
cl> task subsum_fits = subsum_fits.cl
```

```
cl> subsum_fits IRCA0007 351 362 HR72
```

```
cl> imcopy ana/HR72_351_362[1:340,40:900] ana/HR72_1
```

で、同じことができる。上書きできないとかいうエラーが出たら、

```
cl> imdel ana/HR72_1
```

をしてから、同じことを実行する。しかし、最初は、きちんと IRAF のコマンドを実行した方がいいであろう。IRAF に慣れてきたら、IRAF スクリプトを各自で作って、いくつかの動作を自動で行なうようにさせると便利である。

目的天体の赤外線銀河 IRAS00188-0856 についても、全く同様に、組み合わせたデータ<sup>5</sup> を作る。IRCA00070368 - IRCA00070407 のデータであるが、ここでは IRCA00070368 - IRCA00070387 を組み合わせ、IR00188.368.387.fits (A-B を行ったデータ)、IR00188.1.fits(図 6; 必要な領域のみを切り取ったデータ) を作ってみよう。

### 5.2.3 分光フラットで割る

標準星や目的天体は、スリットの A、B の 2 つの位置で観測する。A、B の位置それぞれにおいては、目的天体、標準星ともほとんど同じ位置で観測するが、少しずれてるかもしれない。また、スリットの幅は場所によらず一定になるように設計してあるが、微妙に異なるかもしれない。また、IRCS 内部の光学系の透過率も、スリットの場所によって、微妙に異なるかもしれない。これらを補正するために、スリットの外から、空間的に一様な光を照射させて、スリット方向の透過率の補正を行なう。ランプ ON(ランプからの光+それ以外の物からの熱放射) と OFF(熱放射) のデータを取り、差し引けば、スリットに一樣に照射されたランプからの光を見ることになる。

<sup>5</sup> メモリーが充分大きくないコンピューター上で、“subsum\_fits” を用いて、一度に多くを足し合わせようとすると、反応が異常に遅かったり、変な振る舞いを示すことがある。

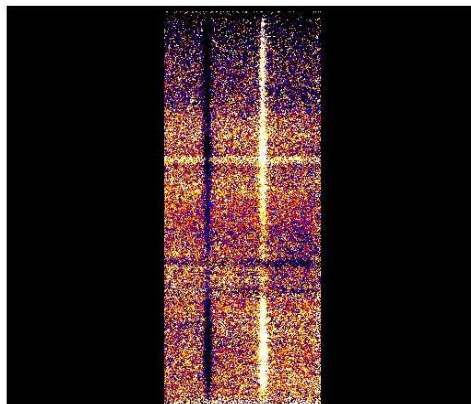


図 6: A–B での目的天体 IRAS00188 の分光データの生画像。必要な領域のみを切り出した。標準星 HR72 に比べて暗く、noisy である。

今の場合、ファイルは、IRCA00070227 – IRCA00070231 (ランプ ON)、IRCA00070247 – IRCA00070251 (ランプ OFF) である。それぞれ 5 フレームずつあり、これらを組み合わせる。

ここで、IRCS データ特有の問題であるが、fits ヘッダーの CTYPE2 が LINEAR になっている。そのままだと、sb 系コンピューターでは IRAF コマンド “imcombine” がうまく動かない。そこで、emacs か何かで、それぞれ 5 個のファイル名が書かれたテキストファイル flaton、flatoff を作成する。参考までに、flaton というファイルの中身は、

```
fits/IRCA00070227.fits
fits/IRCA00070228.fits
fits/IRCA00070229.fits
fits/IRCA00070230.fits
fits/IRCA00070231.fits
```

flatoff というファイルの中身は、

```
fits/IRCA00070247.fits
fits/IRCA00070248.fits
fits/IRCA00070249.fits
fits/IRCA00070250.fits
fits/IRCA00070251.fits
```

である。その後、

```
cl> epar hedit
```

```
images =           @flaton  images to be edited
fields =           CTYPE2  fields to be edited
value   =          DEC---TAN value expression
```

```
cl> hedit
```

```
cl> epar hedit
```

```
images =          @flatoff  images to be edited
fields =          CTYPE2   fields to be edited
value   =          DEC---TAN value expression
```

```
cl> hedit
```

で fits ヘッダーを書き換えておく。この際、動作を実行していいかという質問が次々と表示される。毎回返答するのが面倒で、一気にやりたければ、“epar hedit” で”verify” のところを”yes” から”no” に変えておけばよい。

```
cl> imcombine @flaton ana/flaton_227_231 comb=median
```

```
cl> imcombine @flatoff ana/flatoff_247_251 comb=median
```

```
cl> imarith ana/flaton_227_231 - ana/flatoff_247_251 ana/flat1
```

```
cl> imcopy ana/flat1[1:340,40:900] ana/flat2
```

を行なう。ON から OFF ファイルを差し引き、必要な領域だけを切り抜く。median でなく、average でも良いが、ある pixel に宇宙線ヒットがあって、異常に大きな値を示すフレームがある場合、average だとそれに引きずられる。そのため、median の方がいい。できた分光フラットデータは、規格化しておくのと、割り算しても値があまり変わらないので、何かと便利である。定数をかけるだけの操作なので、必須ではない。

```
cl> epar imstat
```

#### Image Reduction and Analysis Facility

```
PACKAGE = imutil
```

```
TASK = imstatistics
```

```
images =          ana/flat2) List of input images
(fields = image,npix,mean,midpt,stddev,min,max) Fields to be printed
(lower =          INDEF) Lower limit for pixel values
(upper =          INDEF) Upper limit for pixel values
(nclip =          0) Number of clipping iterations
(lsigma =         3.) Lower side clipping factor in sigma
(usigma =         3.) Upper side clipping factor in sigma
(binwidth= 1.000000000000000E-5) Bin width of histogram in sigma
(format =          yes) Format output and print column labels ?
(cache =          no) Cache image in memory ?
```

で、上のように、fields のところに、midpt を加え、image,npix,mean,midpt,stddev,min,max にし、

```
cl> imstat ana/flat2
```

IMAGE	NPIX	MEAN	MIDPT	STDDEV	MIN	MAX
ana/flat2	292740	1440.	1520.	365.3	-58.	2459

```
cl> imarith ana/flat2 / 1520.0 ana/flat
```

のように、フレームの median 値で規格化する。コメントだが、正しい median 値を出すには、上のパラメータ”binwidth” の値 (ヒストグラムの各 bin の幅) を小さめしておくのが良い。

そして、目的天体、標準星のデータを、median 値で規格化された分光フラットで割ってやる。

```
cl> imarith ana/HR72_1 / ana/flat ana/dHR72_1
```

```
cl> imarith ana/IR00188_1 / ana/flat ana/dIR00188_1
```

実際問題としては、すばる IRCS の場合、分光フラットで割っても割らなくても、最終スペクトルはほとんど変わらないが、一応やっておくのがよい。時間があれば、割った場合と割らない場合の結果を、自分で比べて確認して欲しい。

#### 5.2.4 変なピクセルの補正

すばる IRCS の赤外線検出器は 1024×1024 ピクセルからなるが、一部のピクセルはおかしな振る舞いをする。正常なピクセルは、入射した光個の数に比例した信号を出すのが、光子がきても信号を出さない dead pixel や、異常に大きな信号をほぼ恒常的に出す hot pixel が一部存在する。これらをまとめて bad pixel と呼ぶ。また、正常な pixel だけけれども、たまたま宇宙線が当たって大きな信号を示す場合もある。これらの信号をそのまま使うべきではない。例えば、IRCA00070351.fits(図 4 左) をもう一度見てみよう。

```
cl> display fits/IRCA00070351 1 z1=0 z2=30000
```

中央付近にぶつぶつの塊があり、それ以外にも、1 ピクセルだけ周囲に比べて大きく異なる値を示す pixel がある。これらの信号は、周囲の pixel の値を用いて補正する必要がある。一般には、周囲の 8pixel の平均か median 値を取るが、個人の趣味で別の補正の仕方を採用してもよい。

```
cl> display ana/dHR72_1 1 z1=-1000 z2=1000
```

でも、同様に、bad pixel の存在がわかる。

理想的には、bad pixel の補正は、取得した全フレームに対して行うべきである。しかし、赤外線観測の場合、フレーム数が非常に多く大変である。ここでは、何フレームかを組み合わせたデータに対して、補正を行うことにする。

dHR72\_1.fits に対して、cosmicrays というコマンドを用いて、周囲の値に比べて明らかに変な値を示す pixel を見つけ、補正する。

```
cl> noao
```

```
no> imred
```

```
im> crutil
```

```
cr> epar cosmicrays
```

```
(interac=          no) Examine parameters interactively?
```

としておく<sup>6</sup>。

```
cr> cosmicrays ana/dHR72_1 abc
```

この“cosmicrays”というコマンドは、正の値に変な pixel のみを補正するようになっている。今、A-B フレームでは、負の変な値を示すものがあるかもしれない。それも補正するために、

```
cr> imarith abc * -1 abc1
```

```
cr> cosmicrays abc1 abc2
```

```
cr> imarith abc2 * -1 ana/cHR72_1
```

を実行する。display で表示させてみれば、ana/dHR72\_1.fits に比べて ana/cHR72\_1.fits では、bad pixel の数が減少しているのが分かる。以上で、大体補正されるが、補正されない pixel も存在する。それらは手動で補正する必要がある。

```
cr> imcopy ana/cHR72_1 ana/fHR72_1
```

```
cr> fixpix ana/fHR72_1 badHR72_1
```

とする。最初の操作は、fixpix というコマンドは上書きするので、念のために別の名前のファイルにコピーしているだけである。後の操作は、ana/fHR72\_1 の画像の、badHR72\_1 というファイルに書かれてある pixel 座標を、補正するもので、badHR72\_1 は、例えば

---

<sup>6</sup> 一度、cr> に行って cosmicrays を実行すれば、その後は、cl> や no> などでも実行できるようになる。



18 18 97 97  
23 24 67 69

からなるテキストファイルで、 $(X,Y) = (18,97)$ 、 $(23-24,67-69)$  の pixel 座標を、周囲の pixel の値を用いて補正せよという意味である。

補正する値を、周囲の pixel からの外挿ではなく、自分で決めたい場合は、

```
cl> epix fHR72_1 18 97
```

```
          17      18      19
96 -39.0465 -7.45007  44.8567
97  2.20168 -51.8033 -30.5948
98 -17.9674 -55.1241 -52.4665
median -30.59483, mean -23.04381, sigma 32.69519, sample 9 pixels
new value for pixel (-51.8033): 5.0
```

などと入力する。今の場合、 $(X,Y) = (18,97)$  の値を 5.0 と設定している。

ana/dIR00188\_1.fits に対しても、同様に bad pixel を除去し、ana/cIR00188\_1.fits や ana/fIR00188\_1.fits を作る。

### 5.2.5 信号の抽出。A と B の足し合わせ

以上で必要な補正は終了したので、標準星、目的天体のスペクトルを抽出することにする。IRAF の “apall” というタスクを用いる。

apall を用いた single-order スペクトルの解析法に関しては、“<http://iraf.noao.edu/docs/spectra.html>” の A User’s Guide to Reducing Slit Spectra with IRAF, Phil Massey, Frank Valdes, Jeannette Barnes, April 1992(英語) に、より詳しい説明がある。

```
cl> noao
```

```
no> twodspec
```

```
tw> apextract
```

```
ap> apextract.dispaxis=1 (縦方向が波長分散ですという意味。念のための操作)
```

```
ap> epar apall
```

I R A F

Image Reduction and Analysis Facility

PACKAGE = apextract

TASK = apall

```
input = ana/fHR72_1 List of input images
```

```

(output =          ) List of output spectra
(apertur=          ) Apertures
(format =          multispec) Extracted spectra format
(referen=          ) List of aperture reference images
(profile=          ) List of aperture profile images

(interac=          yes) Run task interactively?
(find   =          yes) Find apertures?
(recente=          yes) Recenter apertures?
(resize =          yes) Resize apertures?
(edit   =          yes) Edit apertures?
(trace  =          yes) Trace apertures?
(fittrac=         yes) Fit the traced points interactively?
(extract=         yes) Extract spectra?
(extras =          no) Extract sky, sigma, etc.?
(review =         yes) Review extractions?

(line   =          400) Dispersion line
(nsum  =          20) Number of dispersion lines to sum or median

                                # DEFAULT APERTURE PARAMETERS

(lower  =          -5.) Lower aperture limit relative to center
(upper =           5.) Upper aperture limit relative to center
(apidtab=         ) Aperture ID table (optional)

                                # DEFAULT BACKGROUND PARAMETERS

(b_funct=         chebyshev) Background function
(b_order=          1) Background function order
(b_sampl=         -40:-30,30:40) Background sample regions
(b_naver=         -3) Background average or median
(b_niter=         0) Background rejection iterations
(b_low_r=         3.) Background lower rejection sigma
(b_high_=         3.) Background upper rejection sigma

(b_grow =         0.) Background rejection growing radius

                                # APERTURE CENTERING PARAMETERS

(width  =          5.) Profile centering width
(radius =         10.) Profile centering radius
(thresho=         0.) Detection threshold for profile centering

                                # AUTOMATIC FINDING AND ORDERING PARAMETERS

nfind  =          1  Number of apertures to be found automatically

```

```

(minsep =          5.) Minimum separation between spectra
(maxsep =          1000.) Maximum separation between spectra
(order =          increasing) Order of apertures

                                # RECENTERING PARAMETERS

(aprecen=          ) Apertures for recentering calculation
(npeaks =          INDEF) Select brightest peaks
(shift =          yes) Use average shift instead of recentering?

                                # RESIZING PARAMETERS

(llimit =          INDEF) Lower aperture limit relative to center
(ulimit =          INDEF) Upper aperture limit relative to center
(ylevel =          0.1) Fraction of peak or intensity for automatic width
(peak =          yes) Is ylevel a fraction of the peak?
(bkg =          yes) Subtract background in automatic width?
(r_grow =          0.) Grow limits by this factor
(avglimi=          no) Average limits over all apertures?

                                # TRACING PARAMETERS

(t_nsum =          4) Number of dispersion lines to sum
(t_step =          4) Tracing step
(t_nlost=          3) Number of consecutive times profile is lost before
(t_func=          legendre) Trace fitting function
(t_order=          3) Trace fitting function order
(t_sampl=          *) Trace sample regions
(t_naver=          1) Trace average or median
(t_niter=          0) Trace rejection iterations
(t_low_r=          3.) Trace lower rejection sigma
(t_high_=          3.) Trace upper rejection sigma
(t_grow =          0.) Trace rejection growing radius

                                # EXTRACTION PARAMETERS

(backgro=          fit) Background to subtract
(skybox =          1) Box car smoothing length for sky
(weights=          none) Extraction weights (none|variance)
(pfit =          fit1d) Profile fitting type (fit1d|fit2d)
(clean =          no) Detect and replace bad pixels?
(saturat=          INDEF) Saturation level
(readnoi=          0.) Read out noise sigma (photons)
(gain =          1.) Photon gain (photons/data number)
(lsigma =          4.) Lower rejection threshold
(usigma =          4.) Upper rejection threshold
(nsubaps=          1) Number of subapertures per aperture

```

多くのパラメーターがあるが、実際の設定で注意すべきなのは、一部である。各パラメーターのところにカーソルを持っていき、“?” とすれば、選択肢が表示される。

line = 400, nsum = 20, t\_nsum = 4, t\_step = 4, t\_order = 3, b\_sample = -40:-30,30:40, backgro = fit, weights = none, clean = no, extras = no などとする。以下、詳しく説明する。

extras = no は、信号以外の、(今回は) 不要なデータを書き出さないという意味である。

信号の空間的広がり、波長方向によらず一定との仮定の下に、ある波長のところで、空間的プロファイルを見る。line = 400、nsum = 20 は、 $Y = 400 \pm 10$  のピクセルの値から、空間プロファイルを見るという意味。t\_nsum は、波長方向に 4pixel 足し合わせ、t\_step=4pixel ずつずらして、信号のピーク位置 (X 座標) を、波長方向 (Y 座標) にトレースしていくという意味である。

t\_order は、各波長 (Y) での信号のピーク位置 (X) を、ある関数で fit する場合の次数で、標準星の場合は明るいので 2 (一次式:  $Y = AX + B$ ) ではずれが目立ち、3 (二次多項式;  $Y = AX^2 + BX + C$ ) にした方がよくトレースされる場合が多い。目的天体は一般に暗いので、t\_order=2 (一次式:  $Y = AX + B$ ) で充分である。あまり次数を大きくしても、ノイズに引きずられて、変なトレースをする場合がある。

A-B データは、本来なら、信号以外はノイズの範囲内でゼロになるべきである。ところが、A と B の観測時にスカイの状態が変われば、正か負のどちらかの値を系統的に示す可能性があり、その絶対値は波長によって異なる。backgro=fit により、その引き残しを差し引くことができる。b\_sample は、ピーク位置からどれだけ離れたところ (X 座標) をスカイレベルに取るかということで、天体の信号のない範囲を選択する。正負の両側を取れば、スカイの X 方向の 1 次の傾きがたとえ残っていても、取り除くことができる。

weights = none、clean = no は、optimal extraction を用いないという意味である。optimal extraction というのは、信号の和を変えないで、S/N が最大になるように、ノイズに応じて各ピクセルに重みを付けて足すという方法で、うまく行けばスペクトルの S/N 比が向上するとされている

(<http://archive.stsci.edu/imaps/expastro/node29.html>)。可視光線のように、地球大気の透過率がほぼすべての波長で 100% に近く、信号が滑らかな場合は、うまくいくこともあるようで、可視分光の論文では、optimal extraction を用いたという記述を見ることがある。しかし、赤外の場合は、地球大気の透過率の波長依存性が大きく、信号の大きさも波長によって大きく異なっている。私の経験では、optimal extraction を行なうと、地球大気透過率の波長依存性が強い波長で、明らかに変なスペクトルになる場合があった。よって、標準的に、アパーチャー内の信号を単に足すという手法を用いる。興味のある人は、optimal extraction のマニュアルを読み、各自で詳しく調べ、結果を比較して欲しい。ちなみに、weights = none にしても、clean = yes とすると、自動的に optimal extraction が実行されるので、注意して欲しい。

これ以外のパラメータを変えても、あまり結果は変わらない。しかし、天体が非常に暗い場合などは、適当な値に設定しないとうまくトレースしてくれない場合がある。“help apall” で apall のヘルプが出てくるので、各自でチェックし、色々トライして欲しい。

```
ap> apall ana/fHR72_1 out=ana/fHR72_1pos
```

```
Find apertures for ana/fHR72_1? (yes):
Number of apertures to be found automatically (1):
Resize apertures for ana/fHR72_1? (yes):
Edit apertures for ana/fHR72_1? (yes):
```

で以下のテクトロ画面が、別の Window に出てくる。

テクトロ画面 (図 7) の上にカーソルを持っていき、“:low -8”, “:upper 8”, “:center 215”などで、ピーク位置、及び、スペクトルを抽出する範囲を微調整できる。“t”で、fitting をしてくれる。下の縦棒ではさまれた

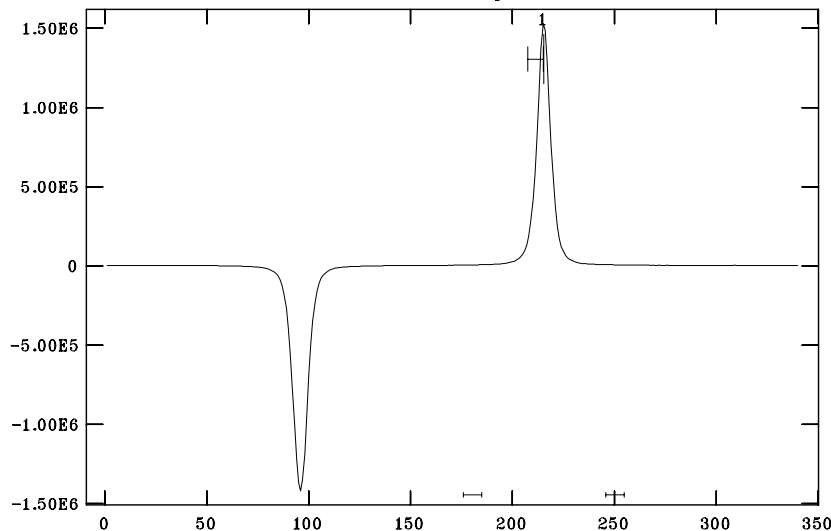


図 7: apall の出力画面。テクトロ画面と呼ぶ。

横線二本が、スカイ差し引きに用いられる範囲である。後は、Yes を押せばよい。図 8 左のように fitting され、図 8 右のような、一次元化されたスペクトルが作られる。“q” で、cl> にカーソルが戻る。ちなみに、一度 apall を実行し、再び同じファイルに対して実行すると、上書きすることになる。最後の方で、“Clobber \*\*\*” の表示がされ、デフォルトでは”No” になっている。何も考えずにリターンを押していると、No が選択されることになり、変更後の結果が上書きされない。パラメーターを変えて行なった 2 回目の fit の方を採用したければ、忘れずに “Yes” と打ち込んでからリターンを押す。

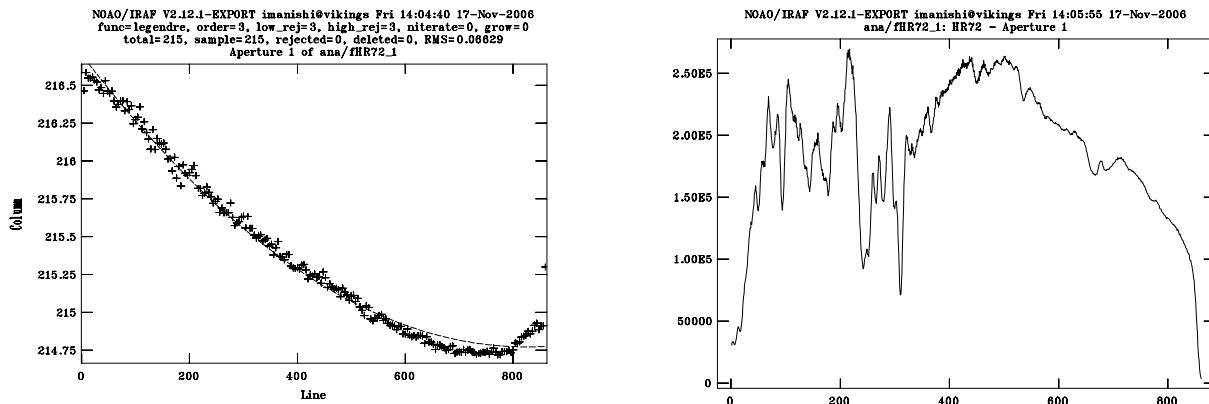


図 8: apall の出力画面の例。(左): トレースの様子。右側で少し (0.25 ピクセルのレベル) ずれている。次数を増やして、もっと良い fit を試すのも良い。(右): 1次元スペクトル。

この操作は、位置 A での正の信号の抽出をただけである。位置 B での信号も抽出し、組み合わせる必要がある。

```
ap> imarith ana/fHR72_1 * -1 ana/fHR72_1n
```

```
ap> apall ana/fHR72_1n out=ana/fHR72_1neg
```

```
ap> imarith ana/fHR72_1pos + ana/fHR72_1neg ana/fHR72_1ana
```

を行なう。目的天体 IRAS00188 に関して、同様のことを行なう。暗いので、line=550, nsum=40, t\_order=2 などとするのがよい。apall で信号をトレースし (図 9)、ana/fIR00188\_1ana.fits を作る。

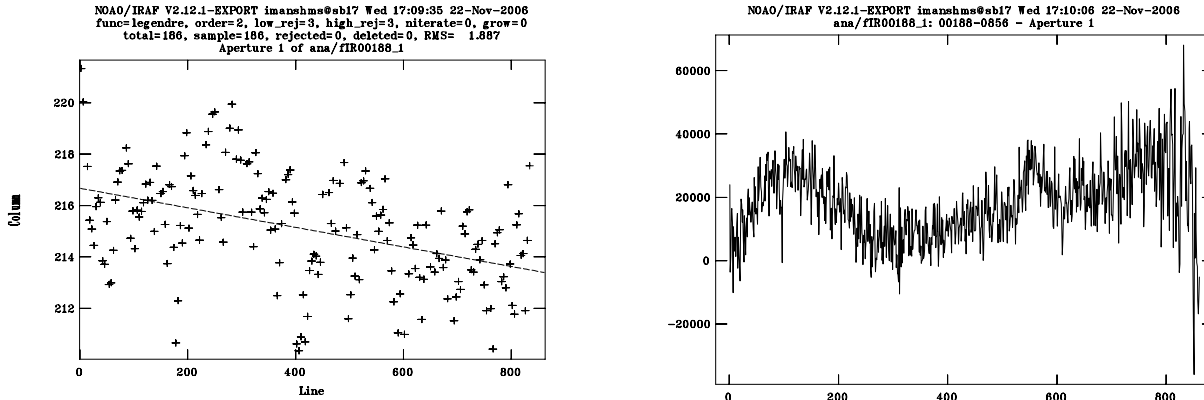


図 9: apall の出力画面の例。(左): 目的天体 IRAS00188 のトレースの様子。(右): IRAS00188 の 1 次元スペクトル。標準星 HR72 に比べて、noisy である。

これで、データは一次元になった。IRAF で操作するという解もあるが、ここでは、これ以降の細かい操作は、テキストファイルに対して、自作のソフト (C 言語、FORTRAN、awk、perl など) で行なうことにする。実際、単に割り算とかをするだけなら IRAF のタスクでも簡単にできるが、binning をしたり、誤差を付けたりするためには、自作のソフトで、テキストファイルをいじる方が何かと便利である。周囲を見ても、1 次元化されたデータに対しては、IRAF 以外のソフトで詳しい解析をしている人が多い。

1 次元の fits ファイルをテキストファイルに変換するには、“wspectext” というタスクを用いる。

```
ap> noao
```

```
no> onedspec
```

```
on> epar wspectext
```

### I R A F

Image Reduction and Analysis Facility

```
PACKAGE = onedspec
```

```
TASK = wspectext
```

```
input = fHR72_1ana Input list of image spectra
```

```
output = Output list of text spectra
```

```
(header = no) Include header?
```

```
(wformat= ) Wavelength format
```

```
(mode = ql)
```

ここで、header のところを “no” にする。これを忘れると、テキストファイルの前半に不要なヘッダー情報が付加されて、煩わしい。“mkdir spec” で、spec ディレクトリを作り、

```
on> wspectext ana/fHR72_1ana spec/fHR72_1.qdp
```

でテキストファイルになる。目的天体 IRAS00188 に対しても同様のことを行なう。qdp という拡張子にしたのは、テキストファイルの表示に qdp というソフトを後で使うからである。

参考までに、天体が非常に暗くて、天体自身の信号を用いて apall でトレースできない場合がある。その場合は、apall を行う際に、“apall ana/fIR00188\_1 out=ana/fIR00188\_1pos refe=ana/fHR72\_1pos” というように、“refe=” で指定する明るい標準星のデータを用いて、天体の信号をトレースすることもできる。ただし、観測時に、天体と標準星を、スリット上の同じ位置で観測していることが条件である。

### 5.2.6 線形性の補正 (今回は行なわない)

理想的な検出器なら、やってきた光子の数に完全に比例した信号が出てくるはず (完全に線形) である。しかしながら、実際の赤外線検出器 (可視の CCD もそう) では、完全な線形性の振る舞いを示さない。ここでは、検出器の原理に戻って、簡単に説明する。実際はそんなに単純ではないが、以下のようなイメージで大体納得できるのではないかと思う。赤外線検出器の原理に関しては、“Electronic Imaging in Astronomy Detectors and Instrumentation” (Ian S. McLean 著; WILEY シリーズ) に、詳細な記述がある。

検出器は、p 型と n 型半導体からなる pn 接合に逆バイアス電圧をかけ (n 側がプラス)、空乏層を作る。そのままでは電流は流れない。空乏層に光子が入射すると、電子が発生し<sup>7</sup>、電子とホールが互いに逆方向に流れていき、逆バイアス電圧が下がっていく。多くの光子が入射し、逆バイアスが充分働かなくなって空乏層で光子を捕えることができなくなれば、光子に対する感度がなくなる。この時点が検出器の full-well に大体相当する。

もう少し例えて言えば、コンデンサー (容量 C) のようなもので、それぞれの側に電荷  $\pm Q$  がたまっていれば、 $V=Q/C$  の電圧が発生する。光子の入射によって、電子が発生し、それがコンデンサーの極に流れていくと、Q の値が変化する。Q の変化は、発生し流れてきた電子の数に比例する。C が一定なら、電圧の変化も電子の数に比例する。我々は、この電圧の変化から、発生した電子の数、つまり、やってきた光子の数を測定しているのである。電圧の変化はアナログであるが、AD コンバーターに通された後は、何 ADU というデジタル値になり、各ピクセルでのその ADU 値が ds9 の画像に表示されている。

さて、実際の検出器の振る舞いとしては、たまった電子の数が増えてくると、やってきた光に対して、発生する電圧が少しずつ減少する (図 10)。コンデンサーの容量 C が増加するかのような振る舞いである。たまった電子の数があるしきい値を超えると、線形性から急激にずれ始めるため、実際の観測では、このしきい値以下に収まるように、露出時間を設定する。そのため、線形性からのずれは小さい。IRCS の場合、4000ADU より小さければ、線形性からのずれは 1-2% 以下と言われており、 $S/N=10-30$  の低分散分光を扱っている限りは、ノイズレベルよりも遥かに小さく、補正の必要はない。多くの人も補正をしていないようである。

電子がたまると C が増加するというのは、以下の説明で大体のイメージがつかめるのではないかと思う。コンデンサーの容量 C は、両極板の表面積を S、間隔を d とすれば、 $C \propto S/d$  で表される。電子がたまると空乏層が小さくなるというのは、両極の間隔が小さくなるというイメージで、従って d が小さくなり、C が大きくなる。従って、実際に観測されている検出器の振る舞いを大体説明できる。

ここで、本来なら解析の前にチェックしておくべきであるが、念のために、IRCS で取得したデータの信号が、4000ADU 以下に収まっているかを確認して見よう。ここで、IRCS のデータは、NDR と COADD の積がかかっているの、それで割ってからの ADU 値を見る必要がある。fits/IRCA00070351.fits を more で見ると、NDR=2, COADD=10 なので、

<sup>7</sup> 光子 1 個が来て、必ず 1 個の電子が発生すれば、量子効率 100% と呼ばれる。現在の赤外線検出器の量子効率は 70-90% とされており、可視の CCD に匹敵するほど感度が良い。

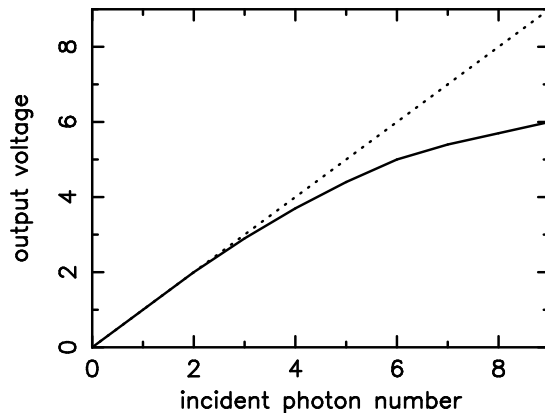


図 10: 検出器の振る舞いが線形からずれる様子の模式図。波線:完全に線形な場合。実線:実際の検出器の振る舞い。

```
cl> imarith fits/IRCA00070351 / 20.0 fits/IRCA00070351d
```

```
cl> display fits/IRCS00070351d 1 z1=0 z2=4000
```

ds9 上でカーソルを動かせば、各ピクセルでの ADU 値が表示される。もっと詳しく調べたければ、

```
cl> imexamine
```

```
cl> z
```

で、調べたい領域の信号が ADU 値で出てくる。“q” で、cl> に戻る。上の一部と hot pixel を除き、4000ADU 以下になっていることが確認できる。目的天体 IRAS00188 に対してもチェックする。NDR=6,COADD=15 であることに注意し、

```
cl> imarith fits/IRCA00070368 / 90.0 fits/IRCA00070368d
```

```
cl> display fits/IRCS00070368d 1 z1=0 z2=4000
```

で、上の一部の信号を除き<sup>8</sup>、4000ADU 以下になっていることを確認する。標準星、目的天体すべてのファイルに対してチェックするのが理想である。

さて、S/N=10-30 程度の低分散グリズム分光は、線形性の補正が不要であった。しかし、明るい天体を高分散分光して、S/N>100 のようなスペクトルの取得を目指す場合は、この線形性からのずれは無視できなくなる。このずれを補正する試みもある。補正の元になるデータは、室内で、(1) フラックス (F) 一定の光を、積

<sup>8</sup> 後述するが、4000ADU を超えているところは、 $\lambda = 4.2\mu\text{m}$  付近に相当する。この波長のデータは私の目的には不要なので、4000ADU を超えるのを承知で観測パラメーターを設定した。



分時間 (t) を変えて検出器に照射させる。積分時間から予想される信号に対する、実際の信号の値を比べ、線形性からのずれを求めるといものである。ただし、実際の観測では、(2) 積分時間一定で、フラックスが異なる場合の線形性からのずれを補正しなければならない。(1) の方法で導出したデータを、(2) の場合 (実際の観測データ) に適用して、どれくらいの精度で正しく線形性からのずれが補正されているかは、各自よく考える必要があると思う。

私はかつて、赤外の高分散分光で、装置の感度表に基付き、3 時間程度積分すれば  $S/N=300$  が達成されるという計算結果を得て、観測提案を出したことがあった。審査員の一人に、たまたま赤外分光装置の開発者で専門家がいて、 $S/N=300$  は現実問題として達成不可能だと言われ、提案が不採択になったことがある。当時は運が悪かったと思ったが、今冷静に考えてみると、これは極めて妥当な評価であったことを思い知る。 $S/N \gg 100$  の赤外スペクトルの取得は、計算上は可能でも、多くの補正の不定性により、実際上は非常に困難であることを心に留めておくべきである。

### 5.2.7 波長較正

波長較正は、波長のわかっている鋭い輝線を数多く放射するランプを用いるのが理想である。可視光や赤外線短波長 ( $2.5\mu\text{m}$  以下) では、適当なランプがある。しかしながら、IRCS の  $3\text{--}4\mu\text{m}$  分光の場合、適当な明るいランプがない。従って、地球大気の透過率の波長依存性を用いて、波長較正する。もっと波長の長い M バンドでも同様である。

図 2 の大気の透過率の波長  $3\text{--}4\mu\text{m}$  の部分を拡大したのが図 11 である。波長によって大きく異なる大気透過率の特性を、ここでは波長較正に利用する。波長  $2.5\mu\text{m}$  以下でも、地球大気からの鋭い OH 夜光輝線を用いて、波長較正する人もいる。

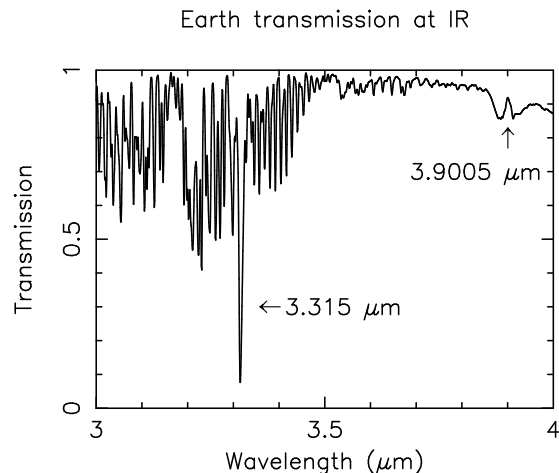


図 11: 地球大気透過率の、波長  $3\text{--}4\mu\text{m}$  での拡大図。UIRT 望遠鏡ホームページ上の <http://www.jach.hawaii.edu/UKIRT/astronomy/utis/atmos-index.html> から引用。

図 11 で、いくつかの波長で、鋭い特徴を示す。実際の数値は、

“<http://www.jach.hawaii.edu/UKIRT/astronomy/utis/atmos-index.html>” の、下の方の “Text version” のところから入手できる。ここでは、簡単のため、 $3.315\mu\text{m}$  のメタン吸収のラインと、 $3.9005$  ミクロンのラインを用い、1 次式で fit することにする。他のラインも含めて用いて最小二乗法で fit したり、高次の関数で fit しても良いが、精度はあまり変わらない。

明るい標準星のスペクトルのテキストデータを、上の地球大気の透過率の波長データと比較する。ここから先は、自作のソフトを用いる。先述のように、C 言語、FORTRAN、gawk、perl といった、自分の得意なもので自作するのを勧める。ここでは、とりあえず、私の作った gawk ソフトを、利用する。

IRCS のスペクトルをテキストファイルに変換した場合、1 列目が、1 から順番に増えていく行番号ではなく、変な値になっている。そこで、行番号に直してやる。”cd spec” の後、

```
gawk -f /home/imanshms/gawk/pix fHR72_1.qdp > fHR72_1c.qdp
```

参考までに、pix というファイルの中身は、

```
{printf("%5.6f %5.6f\n",NR,$2)}
```

となっている。1 列目は、行の番号、2 列目は、信号をそのまま書き出すという操作を行う、単純な gawk ファイルである。詳細は、gawk のマニュアルを参照して欲しい。前半の、“%5.6f” は、小数点より上、小数点以下を何桁まで書き出すかという設定で、本質的なものではない。今の場合、小数点より上を 5 桁、小数点以下を 6 桁まで書き出すようにしている。“%lf” とすれば、全桁数書き出してくれる。C 言語を使ったことのある人にはお馴染みであろう。C 言語や gawk の仕様であり、詳しくはそれらのマニュアルを参照して欲しい。

このデータを表示して、どんな様子かを見る。表示するソフトとしては、qdp、SuperMongo、gnuplot、IDL(ライセンス要) などいくつかあり、各自が慣れたものを使えばよい。私は qdp というソフトに慣れているので、ここではそれを使う。qdp ソフトは、“<http://wwwastro.msfc.nasa.gov/qdp/>” から取得できる。

```
qdp fHR72_1c.qdp
```

```
/xs (あるいは /xw)
```

で、図 12 左の画面が出てくる。“q” で、qdp を終了できる。横軸 X=310 付近の鋭いへこみが、 $3.315\mu\text{m}$  に、横軸 X=680 付近の山が、 $3.9005\mu\text{m}$  に相当すると考えられる。emacs でファイルを開いて、詳しく見てもよし、qdp で “r x 300 400” などと、x=300–400 を拡大して調べても良い。このデータを 1 次式で fit すると、 $\lambda = 1.5975 \times 10^{-3} X + 2.8174$  になるので、1 列目を、行数から、波長に変換する。ここでも、とりあえずは、私が作った gawk ソフトを用いる。

```
gawk -f /home/imanshms/gawk/wavecal fHR72_1c.qdp > fHR72_1cw.qdp
```

参考までに、wavecal というファイルの中身は、以下のようにになっている。

```
{printf("%5.6f %5.6f\n", $1*0.0015975+2.8174, $2)}
```

赤外線 L バンド分光の場合、波長  $2.8\mu\text{m}$  以下は地球大気の透過率が急激に悪くなる。また、波長が  $4.2\mu\text{m}$  近くでは、背景放射が大きく、私の観測パラメーターでは、線形性のいいレンジを超えている<sup>9</sup>。実際、先の画像で 4000ADU を超える Y 座標 (目的天体のフレーム) が、大体  $4.1\text{--}4.15\mu\text{m}$  付近に来ているはずである。従って、地球大気の透過率の小さなデータ (標準星の信号が少ないところ) と、4000ADU をはるかに超えている波長 ( $4.2\mu\text{m}$  付近) のデータは削除することにする。

<sup>9</sup> 興味のあるラインは、波長  $4.1\mu\text{m}$  以下にくるので、科学的には問題ない。  $4.2\mu\text{m}$  まで使えるデータが欲しければ、露出時間を短くすればよいが、観測効率 (オーバーヘッド) が悪くなる。

```
cp fHR72_1cw.qdp fHR72_1ccw.qdp (念のためのコピー)
```

```
emacs fHR72_1ccw.qdp &
```

で不要なデータを削除する。後で出てくる binning のために、ピクセル数を 8 の倍数にしておくとう便利である。

```
qdp fHR72_1ccw.qdp
```

```
/xs
```

で、図 12 右のように、横軸が波長になった信号が見られる。

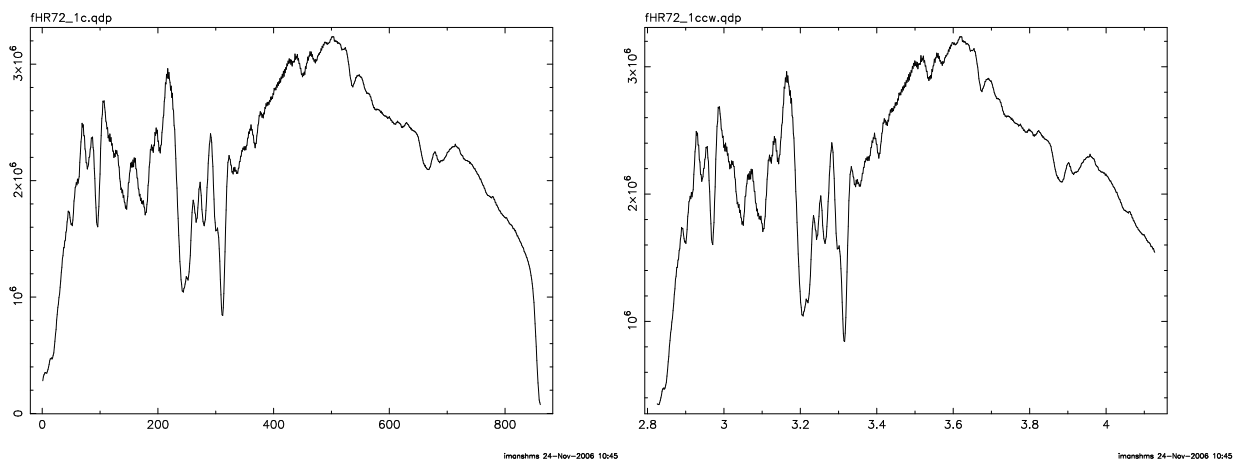


図 12: 標準星 HR72 のデータ。(左): 波長較正前。(右): 波長較正後。

目的天体 IRAS00188 に対しても、全く同様のことを行ない、IR00188\_1ccw.qdp(図 13) を作る。

なお、較正ランプを用いて波長較正できる  $2.5\mu\text{m}$  以下のスペクトルの場合は、IRAF のタスク “identify”, “refs”, “dispcor” などを用いて波長較正ができる。天文情報処理研究会 可視・近赤外域での天文データ処理手引 (1997 年版) の 71 ページ (村山卓著) や HDS のマニュアルに書かれてあるので、そちらを参照して欲しい。

### 5.2.8 Airmass 補正 (今回は行なわない)

Airmass によって、地球大気を通過する長さが異なる。しかし、一般の赤外線分光観測では、目的天体と標準星の Airmass の違いは 0.1 以下に抑えるため、S/N=10-30 の IRCS 低分散グリズム分光データでは、補正の必要はない。私自身、補正した経験はない。

S/N のいいスペクトルに関しては、補正が必要になるかもしれない。IRAF を用いた Airmass の補正に関しては、天文情報処理研究会 可視・近赤外域での天文データ処理手引 (1997 年版) の 72 ページ (村山卓著) に記述がある。

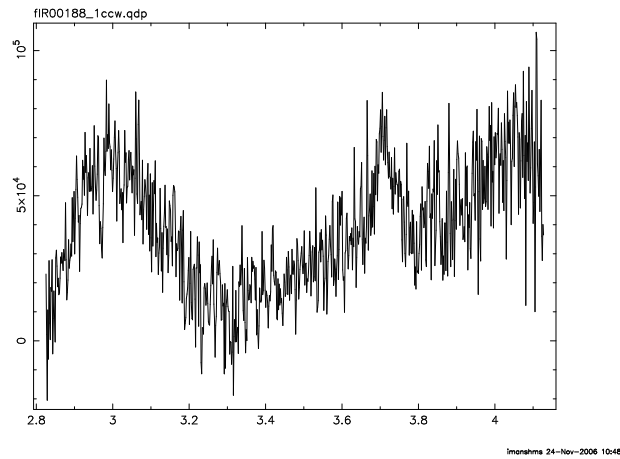


図 13: 目的天体 IRAS00188( $z=0.128$ ) の波長較正後のデータ。波長  $3.7\mu\text{m}$  付近の盛り上がりは、赤方偏移した  $3.3\mu\text{m}$  PAH 放射によるものと考えられる。

### 5.2.9 必要なら binning

IRCS の 0.9 秒角のスリットで分光観測した場合、スリット幅は検出器上では 16 ピクセルに広がり、これが波長分解能 (約 150) に相当する。一方、テキストデータは、検出器の 1 ピクセル毎の信号からなり、実際の波長分解能より 16 倍も密な波長間隔となっている。そこで、16 ピクセルのデータをまとめる (binning) ことにする。理論的には、波長分解能の 2 倍までは意味のある情報が得られるとされている (ナイキストの定理) ので、8 ピクセルの binning でもよい。ここでは、8 ピクセル毎に binning することにする。私が自作した C 言語のソフトを用いる。

```
cd spec (spec ディレクトリに行く)
```

```
/home/imanshms/source/bin2_2col fHR72_1ccw.qdp > fHR72_1ccwb2.qdp
```

```
/home/imanshms/source/bin2_2col fHR72_1ccwb2.qdp > fHR72_1ccwb4.qdp
```

```
/home/imanshms/source/bin2_2col fHR72_1ccwb4.qdp > fHR72_1ccwb8.qdp
```

2 ピクセルずつ binning するソフトなので、3 回繰り返して、8 ピクセルを binning することになる。一気に 8 ピクセル binning するソフトを自作してもよい。私のソフトでは、偶数行を前提としているので、先 (§5.2.7) に不要な波長データを切り取った後のピクセル数が、8 の倍数になっていると便利である。

暗い天体の場合で、8 ピクセル binning でまだ S/N が悪い場合は、さらに binning をして、滑らかなスペクトルにすることになる。信号のレベルに応じて、binning の量を変えるのがよい。また、L バンド分光データは、図 11 にあるように、波長  $3.4\mu\text{m}$  以下で地球大気の透過率が悪く、一般にデータのがたつきも大きくて、S/N が悪い。従って、短波長ほどより多くのピクセルを binning することが多い。

より信号の少ない天体ほど、より多くの binning をしないと、ノイズによるスペクトルのがたつきは小さくならない。しかし、あまりに binning すると、波長方向のデータ点が少なくなり、スペクトルと呼べない程度の波長分解能になってしまう。そのように元々の信号の絶対量が少ない場合は、はっきり言ってどうしようもない。

IRAF では、apall で 1 次元したデータに、"blkavg" を行えば、スペクトルの binning ができる。関心のある人は、IRAF のヘルプを読んで、自分でトライして欲しい。

### 5.2.10 天体を標準星で割る

現時点で、波長毎の信号量のデータができていますが、この信号量は、実際の天体からの信号に、地球大気、すばる望遠鏡、IRCS の透過率 (すべて波長依存) がかったものである。それらを補正する必要がある。

```
paste fIR00188_1ccwb8.qdp fHR72_1ccwb8.qdp > tmp1
```

```
gawk -f /home/imanshms/gawk/div2 tmp1 > fIR00188_fHR72_1b8ccw.qdp
```

単に、awk ファイルを用いて、目的天体、標準星のファイルを組み合わせ、前者を後者で割り算しただけである。

参考までに、div2 というファイルの中身は、以下のような単純な内容である。

```
{printf("%5.5f %5.6f\n",$1,$2/$4)}
```

### 5.2.11 標準星の元々のスペクトルをかける

標準星は、星のタイプが分かれば、温度がわかる。G 型主系列星ならば、元々のスペクトルはその温度の黒体放射で大体 fit できる。B,A,F 型星でも、水素の細いラインを除けば、黒体放射で大体 fit できる。K,M 型といったより晩期の星は、星大気による吸収線が強く、黒体放射から大きくずれる。従って、赤外分光の標準星には使えない。

今、標準星 HR72 は、G0V 型に分類され、従って、5930K と見積もられる (Allen's Astrophysical Quantities, fourth edition, 151 ページ)。従って、5930K の黒体放射をかけてやる。

B,A 型の標準星を用いた場合は、水素の吸収線が強く、L バンドでは、特に、 $Pf\gamma(3.74\mu\text{m})$ 、 $Br\alpha(4.05\mu\text{m})$  の波長に、偽の輝線が見えてしまう。そのような場合は、標準星の生データを見て、周囲の値からの外挿値をあらかじめ手動で入れる。

```
gawk -f /home/imanshms/gawk/bb5930 fIR00188_fHR72_1b8ccw.qdp >
fIR00188_fHR72_1b8ccwbb5930.qdp
```

信号の絶対値があまり大きくなったり小さくなったりすると、桁落ちするので、黒体放射のスペクトルに適当な定数をかけてある。これで、スペクトルの形がわかった。

参考までに、bb5930 というファイルの中身を以下に示す。

```
{printf("%1.6f %1f\n",$1,$2*1.1906*10^5/($1^5)/(exp(14388.3/$1/5930)-1))}
```

## 5.2.12 フラックス補正

最後に、フラックス較正をすれば、できあがりである。小さなアパーチャーによる目的天体のLバンドでの測光観測があれば、時間変動が小さいという仮定の下に、それで補正することができる。しかし、そのようなLバンド測光観測は、残念ながらほとんどない。

そこで、目的天体と標準星の信号の比から、目的天体の明るさを導出する。HR72は $V=6.46\text{mag}$ で、G0V型の星の $V-L=1.46$ (Allen's Astrophysical Qunatities, fourth edition, 151 ページ)なので、 $L=5.00\text{mag}$ と見積もられる。測光でのLバンドの波長範囲は、 $3.547\pm 0.285\ \mu\text{m}$ なので(Allen's Astrophysical Qunatities, fourth edition, 150 ページ)、この波長範囲の信号の総和を、標準星と目的天体で比べる。さらに、総積分時間<sup>10</sup>も考慮して、目的天体の明るさを求める。

実際の操作としては、

1. Lバンドの波長範囲( $3.262\text{--}3.832\ \mu\text{m}$ )で、HR72とIRAS00188の信号は、約80倍の違いがある。(上の波長範囲だけを切り出したファイルに、`/home/imanshms/source/normalize`を使えば、それぞれの信号の和が出る)
2. 積分時間、足したフレーム数、NDRとCOADDの積は、IR00188の方が、18.75倍大きい。なぜなら、fitsヘッダーを見れば、標準星HR72(天体IR00188)のデータは、積分時間が1sec(2.5sec)、足したフレーム数が12(20)、NDRが2(6)、COADDが10(15)であるからである。
3. HR72は $L=5.00\text{mag}$

以上から、IRAS00188のIRCSスリット内の信号の明るさは、 $L=12.9\text{mag}$ であることがわかる<sup>11</sup>。定義により、 $L=0\text{mag}$ では、 $F_\lambda = 6.59 \times 10^{-11} (\text{W m}^{-2} \mu\text{m}^{-1})$ (Allen's Astrophysical Qunatities, fourth edition, 150 ページ)なので、 $L=12.9\text{mag}$ は、Lバンド( $3.55\ \mu\text{m}$ )で $F_\lambda = 4.6 \times 10^{-16} (\text{W m}^{-2} \mu\text{m}^{-1})$ の信号を持つことになる。フラックス補正前のスペクトル(`fIR00188_fHR72_1b8ccwbb5930.qdp`)は、 $3.547\pm 0.285\ \mu\text{m}$ の範囲で、約2.3の信号(平均値)を持っているので、 $F_\lambda = 10^{-16} (\text{W m}^{-2} \mu\text{m}^{-1})$ の単位で表すとすれば、約2.0倍してやればよい。

```
gawk -f /home/imanshms/gawk/multi20E0 fIR00188_fHR72_1b8ccwbb5930.qdp >
fIR00188_fHR72_final.qdp
```

とすれば、フラックス較正された最終スペクトルができる。multi20E0というgawkファイルの中身は、以下である。

```
{printf("%5.6f %5.6f\n", $1, $2*2.0)}
```

その後、

```
qdp fIR00188_fHR72_final.qdp
```

```
/xs
```

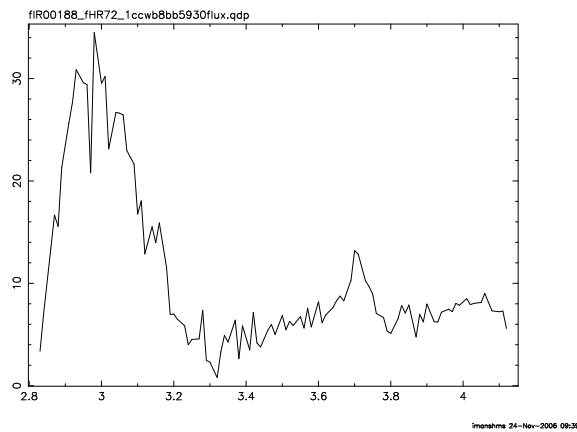


図 14: 得られたスペクトル。

などとすれば、スペクトルが表示される (図 14)。

```
qdp> hardcopy
```

```
qdp> q
```

```
lpr pgplot.ps
```

でスペクトルを印刷できる。

```
qdp fIR00188_fHR72_final.qdp
```

```
/xs
```

```
qdp> la x Wavelength (\u03bcm)
```

```
qdp> la y F\u0304 d\u0304 g l \u0304 u (10\u207b\u2071\u2076 d W m\u207b\u00b2 d \u03bcm\u207b\u00b9 d)
```

などとすれば、縦横軸のラベルが付く。qdp のマニュアルを読めば、色々もっと見やすい表示にすることもできる。

ここでの明るさは、IRCS のスリット内のものである。コンパクトな天体を IRCS で観測した場合、一般にスリット幅 0.9 秒角より L バンドでの seeing (<0.5 秒角) の方が充分小さい。すばる望遠鏡のトラッキング精度がよいので、スリットロス是非常に小さい。仮に 10-20% のレベルであったとしても、コンパクトな目的天体、標準星とも同じようにスリットロスがあるとすれば、割り算することにより相殺され、誤差はもっと小さくなる。

<sup>10</sup> IRCS の場合は、fits ヘッダーを見て、EXPITIME(1 回の露出時間) × NDR(非破壊読み出しの回数) × COADD(各 A、B 位置で何回露出するか) × 足したフレーム数 (§5.2.2) になる。最後の項は、もしいくつかのフレームを組み合わせる場合に、和 (imsum or imcombine, comb=sum) ではなく、平均 (imcombine, comb=average) を取ってれば、省略できる。

<sup>11</sup> 定義により、信号が A 倍暗いと、 $2.5 \log_{10}(A)$  mag だけ大きくなる。例えば、100 倍暗いと 5mag 大きくなる。

参考までに、波長  $2.5\mu\text{m}$  以下では、Lバンドより背景放射が小さいために感度がよく、もっと暗い星を標準星に使える。また、2MASS のデータがあるので、J,H,K バンドでの標準星の明るさを直接求めることができる。しかし、Lバンドでは2MASS のようなカタログがなく、他の波長から、星の色を用いて見積もるしか方法はない。Lバンド分光で使うような明るい標準星は、2MASS では飽和していて、測光誤差が大きく、可視のVバンドから見積もる場合が多い。従って、Lバンドでの標準星の測光誤差は、可視や短波長赤外に比べて大きく、従って目的天体のフラックス誤差も大きいことに注意しなければならない。

小さなアパーチャーでのLバンド測光が行なわれている少数の天体 (AGN) に対して、上記の手法でフラックス較正した結果と、文献の測光値を比較すると、ほとんどの場合 20-30% のレベルで一致し、50% 以上の食い違いを示すことは少ない。つまり、上記の手法によるスペクトルのフラックスも、このレベルで正しいと考えてよいだろう。

### 5.2.13 誤差の評価

これまでは、目的天体の1セットのデータを、標準星の1セットのデータで割っただけなので、スペクトルの各データ点での誤差(エラー)はついていない。しかし、この誤差を評価し、エラーバーを付けるのは、一般に必要である。

エラーの付け方は、研究者によって同じではないようである。ある人は、各波長での背景放射の光子の絶対数を見積もり、そのポアソンノイズをエラーに付けている。また、エラーをつけなくても、ラインが無く滑らかな波長帯のデータのばらつきを見れば、大体のエラーバーは推定できる。

しかしながら、エラーというのは、同じ天体を同じパラメーターで何回か観測し、得られたデータのばらつきから評価すべきものである。私自身は、目的天体、標準星のデータセットを2-3セットにわけ、それぞれに対して、1次元スペクトルを導出する。それぞれを、信号の総和で規格化してスペクトルを出し、そのばらつきからエラーバーを評価している。そのスペクトルに対して、先と同様に、標準星による割り算、黒体放射の掛け算、フラックス較正を行なって、最終スペクトルを導出している。

§5.2.2 で、HR72、IRAS00188 両天体で、データの前半分しか使っていない。後半分も同様に解析し、前半分と組み合わせることにより、エラーバーをつけることができる。私は、自作のソフトを用いて、以下のような操作を行なっている。まず、HR72 の前半と後半のデータから、fHR72.1.qdp、fHR72.2.qdp のテキストデータをそれぞれ作る。IRAS00188 の、前半と後半のデータからもそれぞれ、fIR00188.1.qdp、fIR00188.2.qdp のテキストデータを作る。波長較正までを、まったく同様に行ない、8ピクセル binning を行ない。その後、

```
cd spec
```

```
/home/imanshms/source/normalize fHR72_1ccwb8.qdp > fHR72_1normccwb8.qdp
```

```
/home/imanshms/source/normalize fHR72_2ccwb8.qdp > fHR72_2normccwb8.qdp
```

```
paste fHR72_1normccw.qdp fHR72_2normccwb8.qdp > fHR72_12normccwb8.tmp
```

ここでは、標準星の前半、後半のデータを、それぞれのデータの全ピクセルの信号量の和で割って、規格化している。その後に組み合わせている。一般に、各データ間で、スリットロスの違いによる信号量の絶対レベルが微妙に異なる場合が多い。その信号の絶対値の違いを誤差に含めていたら、不必要にノイズ(エラーバー)が大きくなってしまう。我々の主たる興味は、スペクトルの形なので、規格化した後、各データ間でのスペクトルの形状の違いをエラーバーとして評価するのが適当である。皆さんもじっくり考えて欲しい。

```
nemacs fHR72_12normccwb8.tmp
```



で、まず、最初の3行を削除する。上の normalize という C 言語の自作のソフトは、出力ファイルの最初の3行に、ピクセル数、信号の総和、規格化のために割り算する定数、の情報を書き出すように作ってある。今回は、それらは不要なので、消しているだけである。

その後、

```
gawk -f /home/imanshms/gawk/sigma fHR72_12normccwb8.tmp > fHR72_12normccwb8.qdp
```

で、標準星の前半と後半の規格化されたデータを組み合わせる。その際、ノイズ(標準偏差)も求めている。sigma という gawk ファイルの中身は、以下である。

```
{printf("%1.5f %1f %1f\n", $1, 1/2*($2+$4), sqrt(((($2-($2+$4)/2)^2+($4-($2+$4)/2)^2)/2))}
```

3個のファイルを組み合わせる場合は、3個のデータから標準偏差を求める際の公式に変更すればよい。参考までに書いておくと、以下のような内容になるはずである。

```
{printf("%1.5f %1f %1f\n", $1, 1/3*($2+$4+$6), sqrt((((($2-($2+$4+$6)/3)^2+($3-($2+$4+$6)/3)^2+($4-($2+$4+$6)/3)^2)/3))}
```

以上は、標準星のデータに対する操作である。天体のデータについても、同じことを行う。具体的には、以下の操作を実行する。

```
/home/imanshms/source/normalize fIR00188_1ccwb8.qdp > fIR00188_1normccwb8.qdp
```

```
/home/imanshms/source/normalize fIR00188_2ccwb8.qdp > fIR00188_2normccwb8.qdp
```

```
paste fIR00188_1normccw.qdp fIR00188_2normccwb8.qdp > fIR00188_12normccwb8.tmp
```

emacs fIR00188\_12normccwb8.tmp で、最初の3行を削除

```
gawk -f /home/imanshms/gawk/sigma fIR00188_12normccwb8.tmp > fIR00188_12normccwb8.qdp
```

これで、標準星と天体に関して、エラーバー付きのスペクトルの形が得られた。そこで、天体のスペクトルを、標準星のそれで割り算してやり、標準星の温度に対応する黒体放射のスペクトルを掛け算してやり、フラックス補正をしてやれば、誤差を含んだ、フラックス較正された最終スペクトルができあがる。具体的には、以下の操作を行う。

```
paste fIR00188_12normccw.qdp fHR72_12normccwb8.qdp > fIR00188_fHR72_12normccwb8.tmp
```

Unix の paste というコマンドで、標準星、天体のファイルを結合させる。その後、

```
gawk -f /home/imanshms/gawk/spectrum fIR00188_fHR72_12normccwb8.tmp >
fIR00188_fHR72_12normccwb8.qdp
```

を行う。spectrum という gawk ファイルの中身は、以下のようになっている。

```
{printf("%1.6f %1f %1f\n",$1,$2/$5,sqrt($3*$3/$5/$5+$6*$6*$2/$5/$5/$5/$5))}
```

ここで、標準星、天体ともエラーを持つ。そのデータの割り算をしているため、エラー伝搬の公式を用いている。少し解説すると、天体のデータを  $X \pm \sigma_X$ 、標準星のデータを  $Y \pm \sigma_Y$  とする。天体を標準星で割ったデータ (両方誤差付き)、つまり  $X/Y$  のエラーは、誤差論の参考書 (例えば、新しい誤差論 実験データ解析法 吉澤康和著、158 ページ) によれば、 $(\sigma_X^2 / Y^2 + \sigma_Y^2 \times X^2 / Y^4)$  のルートとなる。

そして、

```
gawk -f /home/imanshms/gawk/bb5930a fIR00188_fHR72_12normccwb8.qdp >
fIR00188_fHR72_12normccwb8bb5930.qdp
```

を行う。bb5930a という gawk ファイルの中身は、以下である。

```
{printf("%1.6f %1f %1f\n",$1,$2*1.1906*10^5/($1^5)/(exp(14388.3/$1/5930)-1),$3*1.1906*10^5/
($1^5)/(exp(14388.3/$1/5930)-1))}
```

データがエラーを含み、3列からなるので、それに対応するように、bb5930 という gawk ファイルから、少し変更を加えただけである。

その後、

```
gawk -f /home/imanshms/gawk/multi530E-2a fIR00188_fHR72_12normccwb8bb5930.qdp >
fIR00188_fHR72_12normccwb8bb5930flux.qdp
```

を行い、5.2.12 節の計算に基付き、フラックス較正を行う。参考までに、multi530E-2a という gawk ファイルの中身は、以下である。

```
{printf("%5.6f %5.6f %5.6f\n",$1,$2*0.05300,$3*0.0530)}
```

エラーなしの場合と定数が異なっているのは、全ピクセルの信号の総和で一度規格化しているからである。エラーなしの場合は、規格化の操作は実施していないことを思い出して欲しい。

emacs fIR00188\_fHR72\_12normccwb8bb5930flux.qdp & で read serr 2 を 1 行目に入れ、

qdp fIR00188\_fHR72\_12normccwb8bb5930flux.qdp

/xs

で、エラーバー付きのスペクトルを表示できる (図 15)。

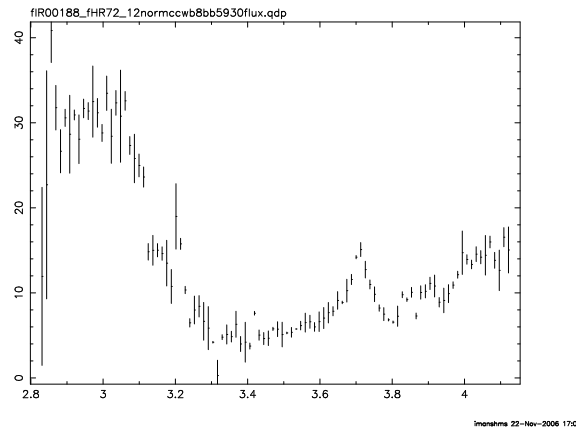


図 15: 得られたスペクトル。

binning=8 のスペクトル (図 15) を見ると、短波長部のがたつきが大きいので、このあたりの binning を増やすことにする。これは、地球大気の透過率が悪いことから来ている。がたつきの大きな波長のデータ部分の binning の数を増やし、その後は上と同様の操作を行ない、スペクトルを得る。

このように、binning は、スペクトルを見ながら適宜調整することになる。なお、binning をする場合は、標準星で割った後の目的天体のスペクトルに対してではなく、標準星で割る前の目的天体、標準星のデータに対して行なう。地球大気の透過率が滑らかな波長帯では、両者の結果はほとんど変わらないが、透過率が大きく変化する赤外線 L バンドの短波長部では、前者の方法は適当ではない (少し考えればわかる)。

適度な binning をし、私が最終的に得たスペクトルは、図 15 である。

皆さんの解析では、どのようなスペクトルになったのでしょうか?

#### 5.2.14 輝線、吸収線の強さの測定

スペクトルが得られたら、輝線を Gaussian fitting して、フラックスや等価幅を求めたり、吸収線の光学的厚さを求めたりする。前者は、qdp で可能である。後者は、自作のソフトで見積もる。それらの値から、物理状態を推定して、科学的議論を展開し、論文にまとめることになる。

## 6 最後に

以上の説明から、読者が、IRCS に限らず、single-order の赤外線分光から、スペクトルを作れるようになれば、非常に幸いである。繰り返しになるが、本マニュアルのわかりにくい点、おかしい点、間違った記述などがあれば、是非、著者までフィードバックをかけて欲しい。

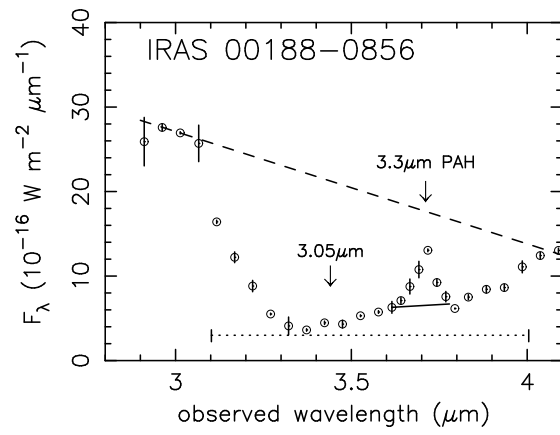


図 16: 最終的なスペクトル。Imanishi et al. (2006 ApJ 637 114) より引用。