

IRAF/DAOPHOT を用いた PSF 測光 簡易マニュアル ver 0.3 ¹

山中 雅之 ²

広島大学 宇宙科学センター 日本学術振興会特別研究員 (PD)

2011 年 10 月 2 日

¹ URL:http://home.hiroshima-u.ac.jp/myamanaka/data/daophot_manual.pdf

² e-mail:myamanaka"at-mark"hiroshima-u.ac.jp

目次

第 1 章	はじめに	2
1.1	本マニュアルについて	2
1.2	PSF 測光とは	2
第 2 章	PSF 測光の全体のフローチャート	4
第 3 章	PSF 測光解析	5
3.1	DAOPHOT の立ち上げ	5
3.2	アパーチャー測光	6
3.2.1	imexamine による星の重心位置の決定	6
3.2.2	phot コマンドによるアパーチャー測光	8
3.3	PSF モデルの作成とチェック方法	12
3.3.1	PSF コマンドによる星の選定とモデル作成	12
3.3.2	作成した PSF モデルのチェック	17
3.4	PSF フィッティング測光	18
3.4.1	allstar コマンド	18
3.4.2	group コマンド	20
3.4.3	nstar コマンド	21
3.4.4	substar コマンド	22
3.4.5	PSF モデルで差し引いた後の画像をチェックする	23
3.5	測光結果の出力	24
3.5.1	pdump コマンドによる情報の出力	24
3.5.2	時刻情報などのヘッダ情報の出力	24
3.5.3	awk コマンドなどを用いた測光結果の整形	25
第 4 章	参照すべきリファレンス	26
第 5 章	Appendix	27
5.1	PSF 星の自動選出	27
5.1.1	pstselect コマンド	27
5.2	IRAF コマンドのシェルスクリプトへの組み込み	31

第1章 はじめに

1.1 本マニュアルについて

本マニュアルでは IRAF に付録されているパッケージソフト DAOPHOT を用いた PSF 測光について私自身の経験を基にその必要最低限の手順を日本語で紹介しております。英語で記述されているマニュアルはいくつかあるのですが、日本語で記述されている文書というのはいくつありません¹。M1 として大学院に入学する院生にとって、英語で記述されたテキストや paper の読解は避けては通れないものですが、英語の読解ばかりに時間を費やしても研究全体として非効率になり兼ねないケースも多々あります² また、解析など技術的な事柄については、細かなニュアンスを国語で理解したいケースが多いでしょう。それらが、僅少ではありますが PSF 測光の経験を持つ私が、DAOPHOT の日本語マニュアルの作成に至った理由です。

本マニュアルの読者対象レベルは「一度でも IRAF を用いて何らかの仕事をした経験がある方、すなわち一次処理、アパーチャー測光、あるいはスペクトル解析のいずれかの解析の経験がある方」とさせていただいております。したがって、IRAF を初歩から勉強したい、あるいは測光とは何ぞや、PSF とは何ぞや、といった基本的な事項を知りたい方については他のリファレンスを参照していただけたらと思います。また反対に、より高度な解析手法についての記述もありません。それらにつきましては、参照すべきリファレンスとして末尾に列挙しておりますのでそちらをご覧ください。

また、本マニュアルは私が独自の視点に基づき記述したテキストであり、その内容の妥当性については読者ご自身で確認していただくよう、お願いします。大いに誤りのある可能性がございます。もし、本マニュアルにおいて、明らかな誤り、改良すべき点などを見出された方は遠慮なく山中までメールをいただけましたら、幸いです。私自身は毎晩観測と解析に従事させていただいている身ではありますが、決して測光解析のプロフェッショナルではありませんので、その道の方にご指導いただけますと幸いです。また、DAOPHOT の使用法の一部を紹介しているに過ぎませんので、さらなる勉強をご希望される方は Lindsey E. Davis 著の“A Reference Guide to the IRAF/DAOPHOT Package”、その他リファレンスを挙げておりますのでこれらを参照してください。

このマニュアルにおいては予め測光したいターゲットが一つあるいは少数に絞られている場合の手法を紹介しております。したがって、daofind などを用いて多くの天体を測光を考えていらっしゃる方は、本マニュアルで imexamine で重心位置を求めるところで、daofind を実行していただきしたら、と思います。daofind の使い方については他のマニュアルなどを参照してください。

1.2 PSF 測光とは

Point Spread Function(PSF) 測光とは、視野内のいくつかの任意の星のプロファイルから PSF モデルを作成し、測光したい天体のピークフラックスにスケールリングして測光する方法です。PSF 測光は、星団や星雲、銀河など込み入った領域に分布する星の測光に有効な手法です。この時、測光したいターゲット天体は星の aperture 領域もしくは sky 領域に他の星などのフラックスによるコンタミネーションが認められます(図 1.1 参照)。このような条件において、いわゆる開口測光 (aperture photometry) を実行すると、ターゲットとなる星のアパーチャー領域のフラックスを過大評価してしまい、結果的に測定される星のフラックスは星本来のそれより明るくなってしまいます(図 1.2 参照)。一方で、PSF 測光では、Gaussian などのモデルを仮定してフィッティングしますので、星本来のプロファイルのフラックスを見積もってくれます。結果として、sky 領域のコンタミネーションを受けにくい測光をすることができます。

¹ 本マニュアルを作成中に、田中幹人氏が記述された DAOPHOT のマニュアル (ネットで公開) や高田唯史氏が記述された「DAOPHOT 簡易マニュアル」の存在を知りました。これらのような良書は参照すべきでしょう。本マニュアルはそれらとまったく独立に作成されたものですので、これらを全て参照されるとより良い「DAOPHOT マイスター」への道が切り開かれることになるかと信じております。

² 私自身はネット上に転がっている英語で書かれた tips やメモなどをいくつか参照して、コマンドを組み合わせで試行錯誤し、結果的に当時は 1 年くらいかかり、このマニュアルで紹介させていただいている手法を見出しました。

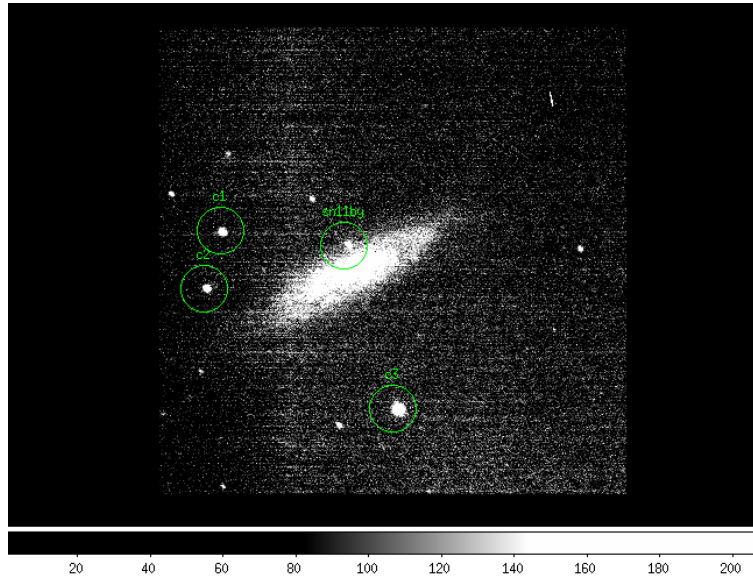


図 1.1: 1.5m かなた望遠鏡および一露出型偏光撮像装置 HOWPol で 2011 年 8 月 4 日 (極大光度の 87 日後) に得られた Ia 型超新星 SN 2011by の視野。超新星は減光しており、母銀河の背景光成分による測光への悪影響が懸念される。

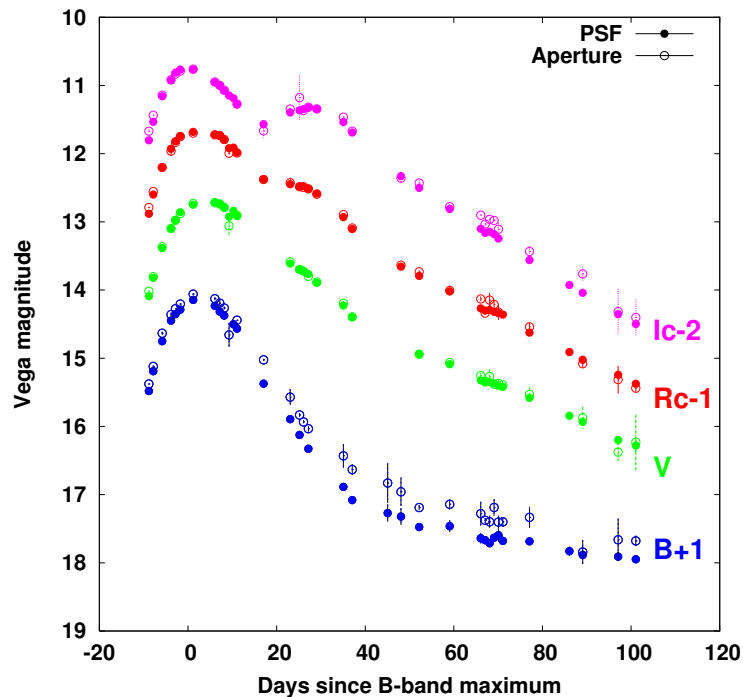


図 1.2: 観測で得られた Ia 型超新星 SN 2011by の B , V , R_C , I_C バンド光度曲線。減光とともに PSF 測光に対してアパーチャー測光の精度が悪化していることが見てとれる。さらに、アパーチャー測光では、明るさを過剰に見積もっており、これは銀河成分によるコンタミネーションのためであると考えられる。B バンドでその差が顕著なのは、CCD では感度が悪く S/N が悪いと考えられる。低 S/N においては、アパーチャー内部にバックグラウンドのバラツキが大きく影響することが考えられ、アパーチャー測光では分が悪い。

第2章 PSF 測光の全体のフローチャート

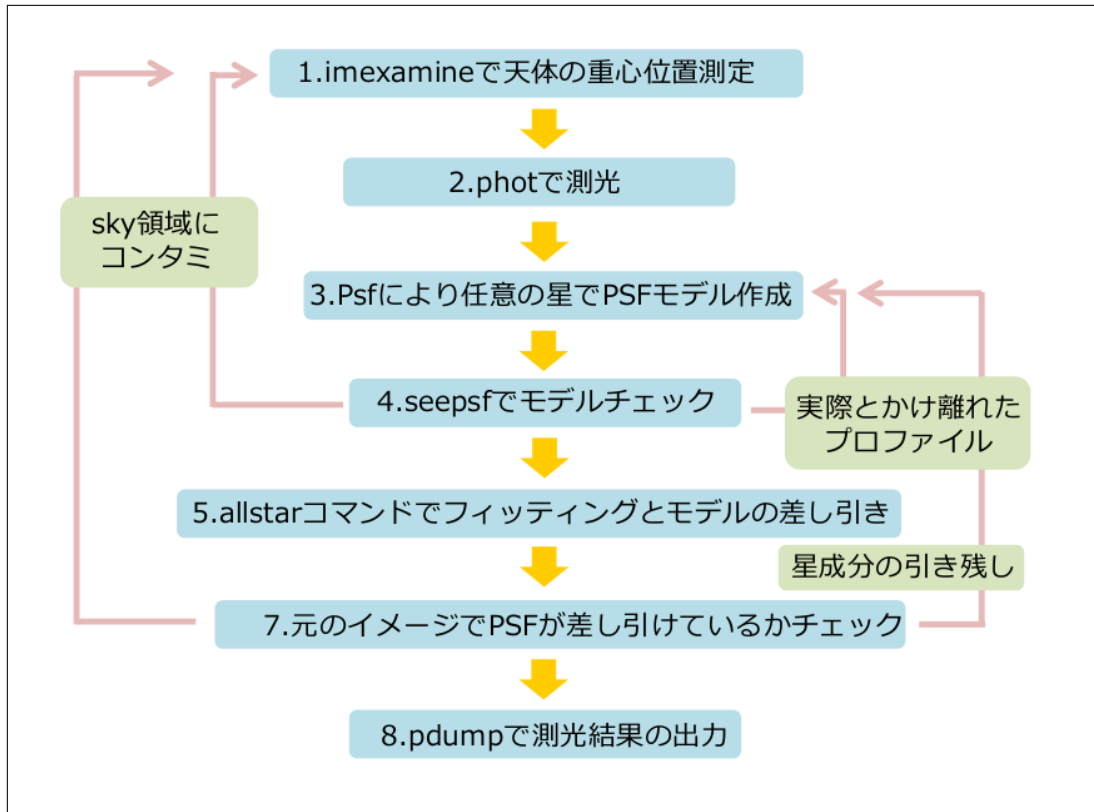


図 2.1: PSF 測光において打たなければならないコマンドの順番。phot、psf コマンドでは観測装置やその夜の観測条件に合わせて、aperture や sky 領域に関するパラメータを適切な値に編集する。なお、phot コマンドの実行は天体の fits ファイルを引数としてエンターキーを押しつづければ良い (3.2 参照)。psf コマンドでは”a”で任意の星を PSF モデルに加え、”w”で psf.1.fits ファイルを保存する (3.3 参照)。allstar コマンドはエンターキーを押しつづければ良い (3.4 参照)。seepsf コマンドや allstar コマンドでは実際に PSF フィッティングが上手くいっているかチェックする。

第3章 PSF 測光解析

3.1 DAOPHOT の立ち上げ

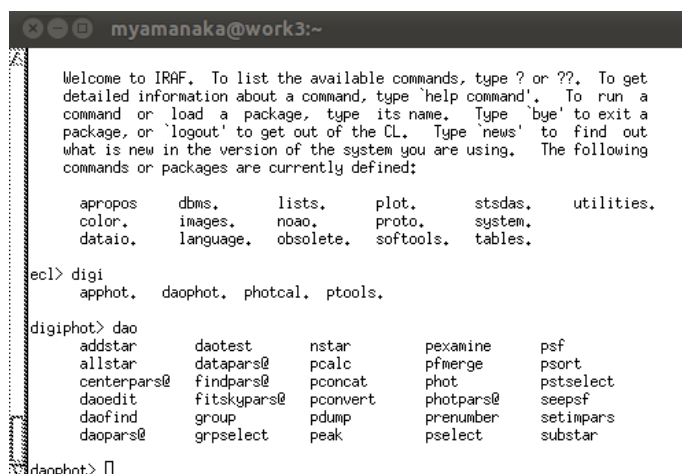
まず IRAF を立ち上げましょう。login.cl をカレントディレクトリに cp してください。作業用ディレクトリに cd で移動した後に例えば、

```
$ cp ~/login.cl .
```

とします。これでカレントディレクトリに login.cl がコピーされます。次に xgterm を立ち上げ、ds9 を立ち上げます。

```
$ xgterm -sb &  
$ ds9 &
```

さらに、xgterm 上で cl(あるいは ecl) と打ち、IRAF を立ち上げます。



```
myamanaka@work3:~  
Welcome to IRAF. To list the available commands, type ? or ??. To get  
detailed information about a command, type `help command'. To run a  
command or load a package, type its name. Type `bye' to exit a  
package, or `logout' to get out of the CL. Type `news' to find out  
what is new in the version of the system you are using. The following  
commands or packages are currently defined:  
  
apropos      dbms.       lists.      plot.       stsdas.    utilities.  
color.       images.    noao.      proto.     system.  
dataio.     language.  obsolete.  softools.  tables.  
  
ecl> digi  
   apphot.  daophot.  photcal.  ptools.  
  
digiphot> dao  
addstar      daotest     nstar      pexamine    psf  
allstar      datapars@  pcalc      pfmerge     psort  
centerpars@  findpars@  pconcat    phot        pstselect  
daoedit      fitskypars@ pconvert   photpars@  seepsf  
daofind      group      pdump      prenumber   setimpars  
daopars@     grpselect  peak       pselect     substar  
  
daophot> |
```

図 3.1: xgterm 上で IRAF、DAOPHOT を起動させたの画面の一例。もちろん IRAF や DAOPHOT のバージョンにより表示される文字列は異なる。

これで IRAF が立ち上がりました。続いて、digiphot と daophot とコマンドを打って、DAOPHOT を立ち上げてください。単に”digi”、”dao”と打つだけでもそれぞれ立ち上げることが可能です。

NOAO PC-IRAF Revision 2.12.2-EXPORT Sun Jan 25 16:09:03 MST 2004

This is the EXPORT version of PC-IRAF V2.12 supporting most PC systems.

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

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

```
ecl> digi
```

```
  apphot.  daophot.  photcal.  ptools.
```

```
digiphot> dao
```

```
addstar      daotest      nstar        pexamine     psf
allstar      datapars@    pcalc        pfmerge      psort
centerpars@  findpars@    pconcat      phot          pstselect
daoedit      fitskypars@  pconvert     photpars@    seepsf
daofind      group        pdump        prenumber    setimpars
daopars@     grpselect    peak         pselect      substar
```

```
daophot>
```

これで DAOPHOT で PSF 測光する準備が整いました。なお、ここで扱うデータはすでにバイアス (ダーク) 引き、フラット割りなどのいわゆる一次処理は済ませてあるものとします。これらについては、他の初学者向けのマニュアルを参照してください。

3.2 アパーチャー測光

DAOPHOT での PSF 測光には .mag ファイルが必要であるため、予備的にアパーチャ測光 (phot) をしておく必要があります。本マニュアルにおいては、PSF モデルを作成するために使う星 (以下、PSF 星と呼ぶことといたします。) は、比較星の中から選ぶことといたします (これは測光作業を単純化させるためのことです。)。今のところ、これで極端に精度が悪化するということは確認されておりませんので、本マニュアルでもそのようにさせていただきます。

3.2.1 imexamine による星の重心位置の決定

アパーチャー測光と PSF 測光に共通することですが、測光する位置はその星のフラックス分布の重心位置であるほうが良い測光精度が得られます。最初に、ds9 に display で測光したいフレーム (例えば PTF11kly_R_36195_1.flb3.fits というファイル) を表示させます。その後、imexamine で測光する対象の星と比較星の重心位置を求めていきます。ピコピコと光るカーソルが出現しますので、目標の星まで持っていく、「a」を押します。この時に測光する順番は、すべての測光作業を終えた後に生成される mag.1 および nst.1 のファイルの中の star ID に反映されますので注意が必要です¹。ここでは、例として、object、比較星 1、比較星 2、比較星 3 の順番で imexamine していきます。

¹ daofind で自動的に星像検出を行うと、測光で得られる ID はランダムな順番となります。ここでは予め測光したいターゲットは決まって折りますので、imexamine を用いた重心位置検出を紹介しております。

```
daophot> display PTF11kly_R_36195_1.flb3.fits
daophot> imexamine > list
```

(測光したい順番に、星の近くで"a"を押して重心位置を求める。)

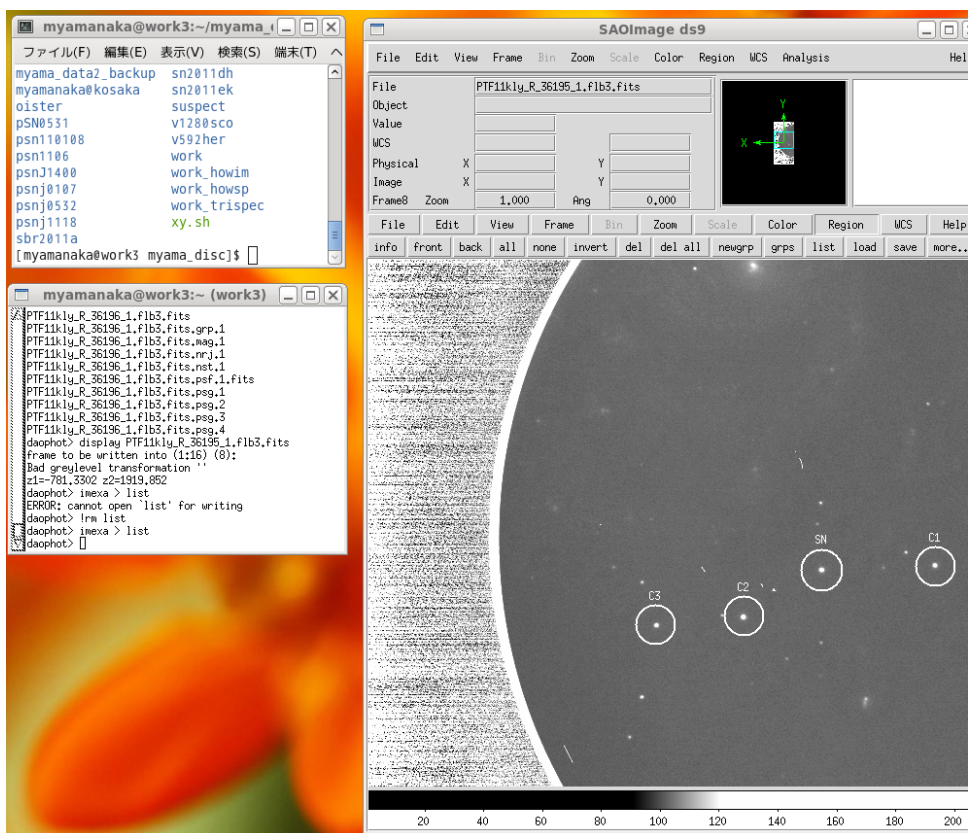


図 3.2: ds9 に測光したい fits ファイルを表示させた時の一例。

imexamine をした結果、list ファイルの中身は例えば以下のように書かれています。

#	COL	LINE	COORDINATES								
#	R	MAG	FLUX	SKY	PEAK	E	PA	BETA	ENCLOSED	MOFFAT	DIRECT
183.20	716.47	183.20	716.47								
6.37	14.40	17457.	554.4	2815.	0.08	8	2.55	2.13	2.01	2.12	
66.46	721.17	66.46	721.17								
6.24	14.71	13061.	552.7	2311.	0.07	-12	1.23	2.10	1.75	2.08	
263.59	667.96	263.59	667.96								
6.35	14.37	17906.	554.8	2796.	0.04	0	1.70	2.13	1.99	2.12	
353.06	659.52	353.06	659.52								
6.73	15.06	9442.	553.9	1289.	0.12	-13	10.1	2.18	2.33	2.25	

ここには重心位置とは異なる情報も書き込まれていますので、vi や emacs など不要な文字列の記述されているラインを削除して保存してください。上記の list ファイルの場合、1,2,4,6,8,10 行目が不要です。なお、vi においては、不要な行に矢印キーでカーソルを持っていき、"d" を二回押すとその行の文字列が削除されます。保存するときは":wq" です。4,6,8,10 行目でかかっている、ENCLOSED、MOFFAT、DIRECT などの値 (例えば 4 行目でいうと、2.13、2.01、2.12) は、それぞれの手法で求められた星の重心位置からの半径方向へのフラックス分布の半値幅 (Full Width of Half Maximum) です。この情報は後ほど phot コマンドで測光する際に必要な値ですので、どこかにメモっておいてください。


```
daophot> vi list
```

(不要な行にカーソルを持っていき、"d"を二回押しして削除する。最終的に":wq"で保存する。)

これで list ファイルの中には、以下のような測光したい星の重心位置の XY 座標のみが残っているはずです。

```
183.20 716.47 183.20 716.47
66.46 721.17 66.46 721.17
263.59 667.96 263.59 667.96
353.06 659.52 353.06 659.52
```

3.2.2 phot コマンドによるアパーチャー測光

DAOPHOT の phot コマンドを実行すると、ある座標が書かれているリストファイルを基に自動測光が実行されます²。ここで編集する必要のあるパラメータは、coords、centerp、fitskyp、photpar です。これらのパラメータは星のアパーチャーサイズやスカイの幅に関するものであり、PSF 測光にとって重要な役割を持っています。coords は先ほど重心位置のみの情報を残したファイル"list"としておきます。

```
daophot> epar phot
```

² これは、デフォルトでは phot のパラメータ interactive が"no"になっているためです。これを"yes"にすると、app タスクにおける phot コマンドと同じようにピコピコ点滅するカーソルが出現し、スペースキーを押すと測光ができます。radplot も"yes"にすると、半径方向のプロファイルとアパーチャーサイズやスカイの幅も表示されます。しかしながら、スペースキーを押す測光では、本来の重心位置とは異なる位置が中心となります(ds9 のピクセルスケールで動かせるカーソルの位置に依存します。)。単純なアパーチャー測光においても、imexamine など重心位置を求めてそれを coords として測光したほうが精度が向上するでしょう。ここでは interactive と radplot は"no"のまま phot タスクを扱います。

```

PACKAGE = daophot
  TASK = phot

image =          Input image(s)
coords =          list Input coordinate list(s) (default: image.coo.?)
output =          default Output photometry file(s) (default: image.mag.?)
skyfile =          Input sky value file(s)
(plotfil=          ) Output plot metacode file
(datapar=          ) Data dependent parameters
(centerp=          ) Centering parameters
(fitskyp=          ) Sky fitting parameters
(photpar=          ) Photometry parameters
(interac=          no) Interactive mode ?
(radplot=          no) Plot the radial profiles?
(icomman=          ) Image cursor: [x y wcs] key [cmd]
(gcomman=          ) Graphics cursor: [x y wcs] key [cmd]
(wcsin =           )_.wcsin) The input coordinate system (logical,tv,physical,world)
(wcsout =           )_.wcsout) The output coordinate system (logical,tv,physical)
(cache =           )_.cache) Cache the input image pixels in memory ?
(verify =          )_.verify) Verify critical phot parameters ?
(update =          )_.update) Update critical phot parameters ?
(verbose=          )_.verbose) Print phot messages ?
(graphic=          )_.graphics) Graphics device
(display=          )_.display) Display device
(mode =           ql)

```

次に、centerp の中身を編集します。centerp までカーソルを運び、":e" を打ちます³。すると以下のようなパラメータが出現します。ここで cbox の値を、おおよそ星像の半値幅の値とします。これが半値幅の値と大きく異なっている場合、phot に失敗することがあります。ここでは、先ほどの imexamine で得られたおおよその半値幅である 2.0 としておきます。centerp の編集を終えて、保存して終了する場合は、":wq" と打って保存して出てください。

³ ちなみに、epar centerp と打つと centerp のパラメータの編集ができます。

```
PACKAGE = daophot
TASK = centerpars
```

```
(calgori=          none) Centering algorithm
(cbox   =          2.) Centering box width in scale units
(cthresh=         0.) Centering threshold in sigma above background
(minsnra=         1.) Minimum signal-to-noise ratio for centering algorithm
(cmaxite=         10) Maximum iterations for centering algorithm
(maxshif=         1.) Maximum center shift in scale units
(clean   =         no) Symmetry clean before centering
(rclean  =         1.) Cleaning radius in scale units
(rclip   =         2.) Clipping radius in scale units
(kclean  =         3.) K-sigma rejection criterion in skysigma
(mkcente=         no) Mark the computed center
(mode    =         ql)
```

次に、fitskyp を編集します。ここでは annulus と dannulus を編集します。星の像が重心位置からどこまで延びているか (annulus) と、そこからの sky の領域の幅 (dannulus) を決めます。sky の領域は星のアーチャーの外側でなければなりません。一般に、星のアーチャーサイズは半値幅の 2 から 3 倍が最も精度が良いと言われています。最適な値は、観測装置や観測条件、星の S/N 比に依って変動しますがここでは間を取って 2.5 倍とします。明るい星では S/N 比が良いので星像の裾のより外側においても背景光のノイズによる影響が少なく、より大きなアーチャーを取るのが良いでしょう。また反対に暗い星では背景光のバラツキが大きくなり、より内側でも寄与が大きくなりますのでアーチャーは小さく取ったほうが精度が上がるようです。sky の幅 (dannulus) は、2 としておきます。この編集作業の終わりにも” :wq” で出てください。

```
PACKAGE = daophot
TASK = fitskypars
```

```
(salgori=          mode) Sky fitting algorithm
(annulus=          5.) Inner radius of sky annulus in scale units
(dannulu=          2.) Width of sky annulus in scale units
(skyvalu=          2.) User sky value
(smaxite=         10) Maximum number of sky fitting iterations
(sloclip=          0.) Lower clipping factor in percent
(shiclip=          0.) Upper clipping factor in percent
(snrejec=         50) Maximum number of sky fitting rejection iterations
(sloreje=          3.) Lower K-sigma rejection limit in sky sigma
(shireje=          3.) Upper K-sigma rejection limit in sky sigma
(khist   =          3.) Half width of histogram in sky sigma
(binsize=         0.1) Binsize of histogram in sky sigma
(smooth  =         no) Boxcar smooth the histogram
(rgrow   =          0.) Region growing radius in scale units
(mksky   =         no) Mark sky annuli on the display
(mode    =         ql)
```

最後に、photpar を編集します。ここで最も重要な aperture の値を設定します。先ほど述べました通り、半値幅の

2.5 倍の 5 としておきます。

```
PACKAGE = daophot
TASK = photpars
```

```
(weighti=          constant) Photometric weighting scheme
(apertur=          5) List of aperture radii in scale units
(zmag   =          17.89) Zero point of magnitude scale
(mkapert=          no) Draw apertures on the display
(mode   =          ql)
```

これで、すべてのパラメータの編集が終わりました。":wq"で編集したパラメータを保存してください。そして phot コマンドを実行します。phot を実行すると、様々な質問が出てきますが必要なパラメータはすべて上記の作業で編集済みですので、すべて無視してエンターキーを押しつづけてください。測光が上手くいくと"ok"という文字が一番右側に出ます。ここでの ID は imexamine で"a"を押した順番が反映されています。もし、上手くいっていないと"err"が表示されます。たとえ"ok"でも等級が記述されるべきところに"INDEF"という文字が表示されることがあります。もし、"err"や"INDEF"が表示された場合は、何らかの原因で測光が上手くいっていませんので、imexamine からやり直してください。

```
daophot> phot PTF11kly_R_36195_1.flb3.fits
Input coordinate list(s) (default: image.coo.?) (list):
Output photometry file(s) (default: image.mag.?) (default):

Centering algorithm (none) (CR or value):
  New centering algorithm: none
Sky fitting algorithm (mode) (CR or value):
  Sky fitting algorithm: mode
Inner radius of sky annulus in scale units (5.) (CR or value):
  New inner radius of sky annulus: 5. scale units 5. pixels
Width of the sky annulus in scale units (2.) (CR or value):
  New width of the sky annulus: 2. scale units 2. pixels
File/list of aperture radii in scale units (5) (CR or value):
  Aperture radius 1: 5. scale units 5. pixels
Standard deviation of background in counts (0.) (CR or value):
  New standard deviation of background: 0. counts
Minimum good data value (INDEF) (CR or value):
  New minimum good data value: INDEF counts
Maximum good data value (INDEF) (CR or value):
  New maximum good data value: INDEF counts

PTF11kly_R_36195_1.flb3.fits   183.20   716.47   561.8608   12.046   ok
PTF11kly_R_36195_1.flb3.fits    66.46   721.17   556.1678   12.344   ok
PTF11kly_R_36195_1.flb3.fits   263.59   667.96   562.659    12.017   ok
PTF11kly_R_36195_1.flb3.fits   353.06   659.52   555.1814   12.683   ok
```

3.3 PSF モデルの作成とチェック方法

理想的な PSF モデルの作成のためには、周辺環境によるコンタミネーションの無い星を選びなければなりません。また、暗い星ではバックグラウンドのバラツキが効いてきますので、これも適しません。程よく明るく⁴ 銀河や星雲、星から離れている星を選びましょう。ここでは、IRAF の長所の一つである、インタラクティブモード (星のプロファイルを `irafterm` に表示させ、その場でパラメータを編集して自分の目で確かめることができるモードです。日本語に訳すと”直観的モード”でしょうか。) を紹介しますが、インタラクティブにせず `imexamine` で選んだ星をそのまま自動的に PSF 星として選ぶ方法もあります。そちらについては、Appendix にてオプションとして紹介させていただきます。

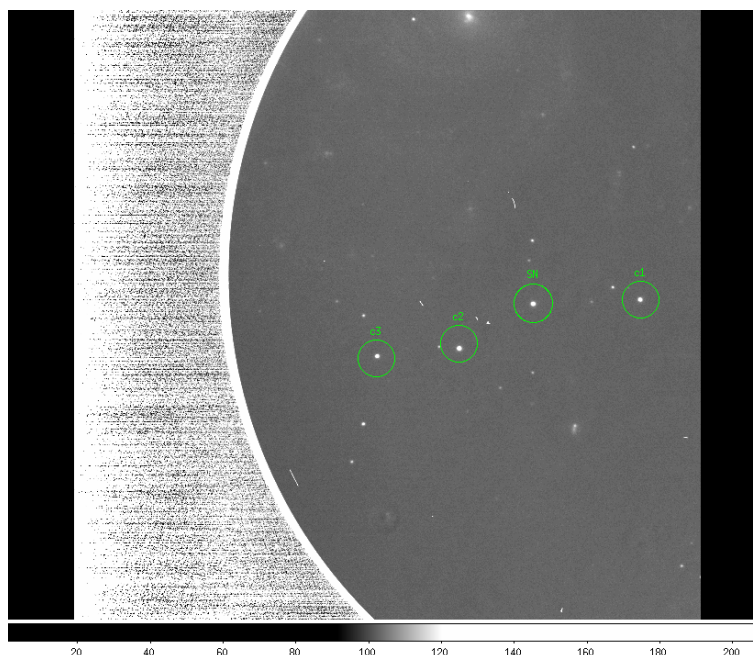


図 3.3: Kanata/HOWPol によって取得された M101 に出現した超新星 PTF11kly (=SN 2011fe) のイメージ。このイメージにおいては、PTF11kly および比較星には明瞭なコンタミネーションが確認されていない。

3.3.1 PSF コマンドによる星の選定とモデル作成

`psf` コマンドのパラメータを編集します。ここでは、PSF 測光において最も肝要なパラメータ `matchrad`、`psfrad` そして `fitrad` を編集します。また、PSF モデルの関数 `function` も IRAF で用意されているモデルから選ぶことができます。

```
daophot> epar psf
```

⁴ 筆者の経験においては、サチらない程度の明るい星を PSF モデルの参照星とすると、やや暗い星のみを選んだ時とモデルの形状が異なることを確認しています。筆者はどちらかというと暗い星の測光経験が多いため、暗い星を PSF モデルに使うことが多く、この時は測光精度が良いのですが明るい星をモデルとすると精度が悪くなる印象を持っております。このようなケースでは、アパーチャー測光を導入したほうが良いでしょう。

```

PACKAGE = daophot
  TASK = psf

image = PTF11kly_R_36195_1.flb3.fits  Input image(s) for which to build PSF
photfile=          default  Input photometry file(s) (default: image.mag.?)
pstfile =          Input psf star list(s) (default: image.pst?)
psfimage=          default  Output PSF image(s) (default: image.psf.?)
opstfile=          default  Output PSF star list(s) (default: image.pst.?)
groupfil=          default  Output PSF star group file(s) (default: image.psg.?)
(plotfil=          ) Output plot metacode file
(datapar=          ) Data dependent parameters
(daopars=          ) Psf fitting parameters
(matchby=          yes) Match psf star list to photometry file(s) by id number ?
(interac=          yes) Compute the psf interactively ?
(mkstars=          no) Mark deleted and accepted psf stars ?
(showplo=          yes) Show plots of PSF stars ?
(plottyp=          mesh) Default plot type (mesh|contour|radial)
(icomman=          ) Image cursor: [x y wcs] key [cmd]
(gcomman=          ) Graphics cursor: [x y wcs] key [cmd]
(wcsin =           )_wcsin) The input coordinate system (logical,tv,physical,world)
(wcsout =          )_wcsout) The output coordinate system (logical,tv,physical)
(cache =           )_.cache) Cache the input image pixels in memory ?
(verify =          )_.verify) Verify critical psf parameters ?
(update =          )_.update) Update critical psf parameters ?
(verbose=          )_.verbose) Print psf messages ?
(graphic=          )_.graphics) Graphics device
(display=          )_.display) Display device
(mode =            ql)

```

daopars で”:e”と打ってください⁵。以下のようにパラメータが表示されます。ここで、matchrad、psfrad そして fitrad を編集してください。一般には、matchrad は半値幅と同程度、fitrad は半値幅の 2-3 倍、psfrad は半値幅の 4-6 倍程度が良いと言われています。特に、matchrad と fitrad は測光精度に影響を及ぼしうる重要なパラメータです。観測装置ごとに最適なパラメータは異なることが予想されますし、色々と試行錯誤⁶ をするようにしてください。フィッティングする関数ですが、Gaussian の他に IRAF 独自のものも用意されております。ここでは、function=auto とし、最も分散の少ないモデルを自動的に選ばせるようにします。

⁵ ここでも、epar daopars でも直接 daopars のパラメータを編集することができます。

⁶ 私個人の経験ではピクセルスケールが小さくオーバーサンプリングである場合より広い fitrad を取り、ピクセルスケールが粗くアンダーサンプリング気味になっている場合 fitrad を小さくしたほうが精度が良くなる感触を得ております。何か情報があればご教授いただけますと幸いです。

```
PACKAGE = daophot
TASK = daopars
```

```
(functio=          auto) Form of analytic component of psf model
(varorde=          0) Order of empirical component of psf model
(nclean =          0) Number of cleaning iterations for computing psf model
(saturat=         no) Use wings of saturated stars in psf model computation ?
(matchra=          2.) Object matching radius in scale units
(psfrad =          8.) Radius of psf model in scale units
(fitrad =          6.) Fitting radius in scale units
(recente=         yes) Recenter stars during fit ?
(fitsky =          no) Recompute group sky value during fit ?
(groupsk=         yes) Use group rather than individual sky values ?
(sannulu=          0.) Inner radius of sky fitting annulus in scale units
(wsannul=         11.) Width of sky fitting annulus in scale units
(flaterr=         0.75) Flat field error in percent
(proferr=          5.) Profile error in percent
(maxiter=         50) Maximum number of fitting iterations
(clipexp=          6) Bad data clipping exponent
(clipran=          2.5) Bad data clipping range in sigma
(mergera=        INDEF) Critical object merging radius in scale units
(critsnr=          1.) Critical S/N ratio for group membership
(maxnsta=        10000) Maximum number of stars to fit
(maxgrou=          60) Maximum number of stars to fit per group
(mode =           ql)
```

続いて psf コマンドを実行して、PSF 星を選びましょう。まずは psf と打ってエンターキーを押してつづけてください。必要なパラメータはやはり全て編集済みとなっているはずです。引数には測光したい fits ファイル名を与えてください。imeximine の時と同じようにピコピコするカーソルが出現します。

```
daophot> psf PTF11kly_R_36195_1.flb3.fits
```

```

Input photometry file(s) (default: image.mag.?) (default):
Input psf star list(s) (default: image.pst?):
Output PSF image(s) (default: image.psf.?) (default):
Output PSF star list(s) (default:image.pst.?) (default):
Output PSF star group file(s) (default: image.psg.?) (default):

Analytic psf function(s) (auto):
    Analytic psf function(s): auto
Order of variable psf (0):
    Order of variable psf: 0
Psf radius in scale units (8.):
    New psf radius: 8. scale units 8. pixels
Fitting radius in scale units (6.):
    New fitting radius: 6. scale units 6. pixels
Minimum good data value (INDEF) (CR or value):
    New minimum good data value: INDEF counts
Maximum good data value (INDEF) (CR or value):
    New maximum good data value: INDEF counts

Warning: Graphics overlay not available for display device.

Computing PSF for image: PTF11kly_R_36195_1.flb3.fits
4 stars read from PTF11kly_R_36195_1.flb3.fits.mag.2

```

ピコピコ点滅しているカーソルを先ほどアパーチャー測光した星に持っていき”a”と打つと、irafterm のグラフに星のプロファイルが表示されます。このグラフでは、XY 軸方向が CCD 上の XY 軸方向に相当し、Z 軸方向に星のカウントが相当しています。この時、”e”や”w”と打つと、視点が横方向に回転していき、”n”や”s”と打つと縦方向に回転します。”r”と打つと半径方向のプロットを表示します。これを見て裾と見なせる位置が”fitting radius”より外側にあると判断された場合、つまり適切でない fitting radius だと思われましたら、”d”でこの画面を抜けてください。epar psf でパラメータを編集してください。

納得いく”fitting radius”と”half-width of half-maximum”が得られましたらば、もう一度”a”と押してください。そうするとこの星は PSF モデルを作成するための参照星として加えられます。このような作業をあといくつかの、スカイ領域にコンタミネーションの見られない星について実行してください。

続いて、”w”(あるいは、”f”を打った後に”w”)を押して保存ください。保存すると以下のようなメッセージが出力されます。ここで出力されている各関数の分散から、最も良い fitting 結果が採用されるようになっております。これは psf コマンドにおけるサブタスク daopars のパラメータ function を”auto”としているためです。function を”auto”以外の何か具体的な関数(例えば”gauss”)にしているとこのような最適なモデルの探索は行われません。最後の 3 行(”writing から始まる文章”)で 3 つのファイルを保存しております。こうして保存された psf.1.fits ファイルが PSF フィッティングに用いられます。

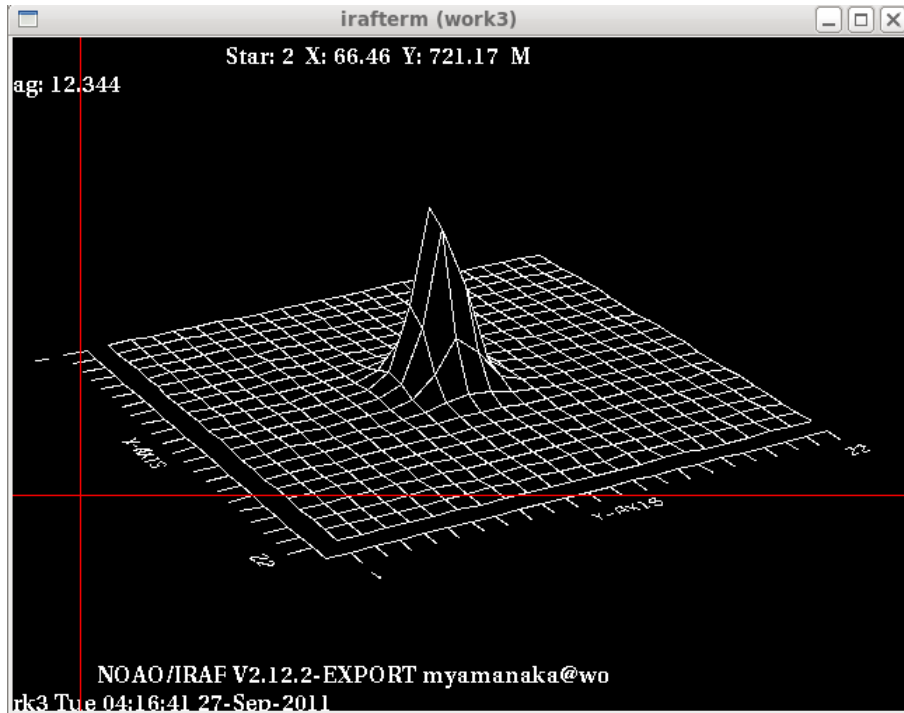


図 3.4: psf コマンドにおいて星の近くで”a”と打ったときに表示される星像の鳥瞰図。

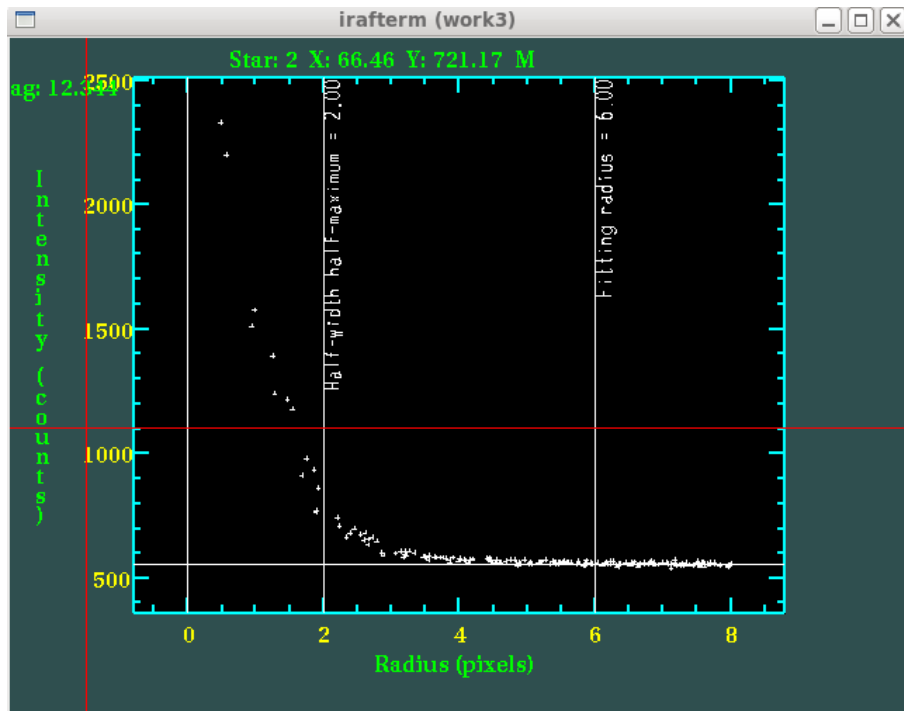


図 3.5: psf コマンドで”a”と打った後に”r”と打ったときに表示される半径方向のカウント分布。Half-width half-maximum は matchrad の半分の値、”fitting radius”は fitrad の値がそれぞれ反映されている。

```

Star 4 has been added to the PSF star list
  X: 353.06 Y: 659.52 Mag: 12.683 Dmin: 539.7638 Dmax: 1724.03
Star 3 has been added to the PSF star list
  X: 263.59 Y: 667.96 Mag: 12.017 Dmin: 542.5501 Dmax: 2992.55
Trying function gauss norm scatter = 0.01329167
Trying function lorentz norm scatter = 0.02876239
Trying function moffat15 norm scatter = 0.01679772
Trying function moffat25 norm scatter = 0.007970733
Trying function penny1 norm scatter = 0.008209451
Trying function penny2 norm scatter = 0.008235213
Best fitting function is moffat25

Analytic PSF fit
  Function: moffat25 X: 341. Y: 703. Height: 965.2758 Psfmag: 12.683
  Par1: 1.002768 Par2: 0.8981706 XYterm: 0.01124858 Moffat: 2.5

Computed 1 lookup table(s)

Writing PSF image PTF11kly_R_36195_1.flb3.fits.psf.1.fits
Writing output PSF star list PTF11kly_R_36195_1.flb3.fits.pst.1
Writing output PSF star group file PTF11kly_R_36195_1.flb3.fits.psg.1

```

また、イメージの上で PSF が異なる場合 (例えば、ある位置では非常に焦点が合っているにも関わらず、他の位置ではやや像がボケてしまっているような場合)、画像中で PSF 星をもう少し多めに 7,8 星ほど選んでおき、daopars 中の varorder の値を”2”としておくと良いようです。

3.3.2 作成した PSF モデルのチェック

得られました PSF モデルのプロファイルを自分の目で見てチェックしたいときに、使われるコマンドが seepsf です。seepsf の使い方は以下の通りです。最初の引数に、psf.1.fits ファイルを与えて、2 つ目に出力したい任意の fits ファイル名を与えます。

```
daophot> seepsf PTF11kly_R_36195_1.flb3.fits.psf.1.fits psf_output.fits
```

この出力したファイルを dipslay コマンドで ds9 上に表示してやると、測光に用いられるモデルプロファイルが表示されます。ここで imexamine などを用いて (“r”、“s”、“e”などを押して色々なプロットでチェックできます。)、sky 領域にコンタミネーションが無いか、あるいはモデルの形状が実際に写っている星と大きくかけ離れていないか、などをチェックすることができます。

3.4 PSF フィッティング測光

得られた PSF モデルを目標天体と比較星にフィッティングしていきます。扱う IRAF タスクは `allstar` コマンドです。また、`allstar` コマンドの代替として、`group`、`nstar`、`substar` の 3 つのコマンドを使っても構いません。`allstar` コマンドで測光と PSF モデルの差し引きが終わったら、3.4.5 の PSF モデルで差し引いた画像のチェックまで読み出しているだけでも何ら支障はございません。これらのタスクの使い方は非常に単純で、要は元の画像ファイルを引数で与えて、エンターキーを押すだけで良いのです。

3.4.1 `allstar` コマンド

`allstar` コマンドは、最初に `phot` コマンドで測光した星をフィッティングしてくれます。また、同時に、元の画像から PSF モデルを差し引いてくれます。元の `fits` ファイルを引数で与えて、エンターキーを押すだけでください。得られた `als.1` ファイルが、測光結果がまとめられているファイルで、`sub.1.fits` が PSF モデルを差し引いた画像です。`sub.1.fits` は PSF モデルが上手く生成され、差し引くことができたかどうかのチェックに使います。最後に出力されている ID が `imexamine` で重心位置を求めた順番に対応しています。最後の `MAG` が IRAF 等級です (ゼロ点の等級は任意に与えられていますので決して実際の `apparent magnitude` ではありません。)

```
daophot> allstar PTF11kly_R_36195_1.flb3.fits
```

```

Input photometry file (default: image.mag.?) (default):
PSF image (default: image.psf.?) (default):
Output photometry file (default: image.als.?) (default):
Subtracted image (default: image.sub.?) (default):
Output rejections file (default: image.arj.?) (default):

Recenter the stars (yes):
    Recenter the stars: yes
Use group sky values (yes):
    Use group sky values: yes
Refit the sky (no):
    Refit the sky: no
Psf radius in scale units (8.):
    New psf radius: 8. scale units 8. pixels
Fitting radius in scale units (6.):
    New fitting radius: 6. scale units 6. pixels
Maximum group size in number of stars (60):
    New maximum group size: 60 stars
Minimum good data value (INDEF) (CR or value):
    New minimum good data value: INDEF counts
Maximum good data value (INDEF) (CR or value):
    New maximum good data value: INDEF counts

NITER = 1
NITER = 2
NITER = 3
NITER = 4
NITER = 5
NITER = 6
FITTING:  ID:      4  XCEN:   353.05  YCEN:   659.52  MAG:    12.66
FITTING:  ID:      3  XCEN:   263.66  YCEN:   667.99  MAG:    11.99
NITER = 7
FITTING:  ID:      1  XCEN:   183.12  YCEN:   716.48  MAG:    12.01
NITER = 8
FITTING:  ID:      2  XCEN:    66.48  YCEN:   721.07  MAG:    12.30

```

3.4.2 group コマンド

group コマンドでは、星団などの込み入った領域の星々を測光する場合に行うグルーピングタスクです。ここで紹介する使用法では、意味をなすタスクではありません。しかし、実際にフィッティングを行う nstar コマンドを実行するときに必要な .grp.1 ファイルを生成しますので、必ず実行してください。以下のように打って聞かれる質問に対してはすべてエンターキーを押してそのまま次に進んでください。

```
daophot> group PTF11kly_R_36195_1.flb3.fits
```

```
Photometry file (default: image.mag.?) (default):
PSF image (default: image.psf.?) (default):
Output group file (default: image.grp.?) (default):

Psf radius in scale units (8.):
    New psf radius: 8. scale units 8. pixels
Fitting radius in scale units (6.):
    New fitting radius: 6. scale units 6. pixels
Critical S/N ratio in stdevs per pixel (1.):
    New critical S/N ratio: 1. stdevs per pixel
Minimum good data value (INDEF) (CR or value):
    New minimum good data value: INDEF counts
Maximum good data value (INDEF) (CR or value):
    New maximum good data value: INDEF counts

Size of      Number of
group        groups

    1          4

Total of 4 stars in 4 groups
```

3.4.3 nstar コマンド

nstar を用いて PSF 測光を行います。これで .nst.1 という PSF 測光結果の値がまとめられているファイルが生成されます。nstar も group 同様に、パラメータを編集する必要はありません。エンターキーを押し続けてください。

```
daophot> nstar PTF11kly_R_36196_1.flb3.fits
```

```
Input group file (image.grp.?) (default):
PSF image (default: image.psf.?) (default):
Output photometry file (default: image.nst.?) (default):
Output rejections file (default: image.nrj.?) (default):

Recenter the stars (yes):
    Recenter the stars: yes
Refit the sky (no):
    Refit the sky: no
Use group sky values (yes):
    Use group sky values: yes
Psf radius in scale units (8.):
    New psf radius: 8. scale units 8. pixels
Fitting radius in scale units (6.):
    New fitting radius: 6. scale units 6. pixels
Maximum group size in number of stars (60):
    New maximum group size: 60 stars
Minimum good data value (INDEF) (CR or value):
    New minimum good data value: INDEF counts
Maximum good data value (INDEF) (CR or value):
    New maximum good data value: INDEF counts

Group:    1 contains 1 stars
          ID:    4  XCEN:   352.84  YCEN:   659.40  MAG:    12.56
Group:    2 contains 1 stars
          ID:    3  XCEN:   263.58  YCEN:   667.97  MAG:    11.81
Group:    3 contains 1 stars
          ID:    1  XCEN:   183.07  YCEN:   716.38  MAG:    11.89
Group:    4 contains 1 stars
          ID:    2  XCEN:    66.13  YCEN:   720.96  MAG:    12.20
```

3.4.4 substar コマンド

最後に、PSF モデルによって実際の星を上手くフィッティングできているかどうかを確認するために、元の画像から PSF モデルを差し引く substar コマンドを実行します。この時得られる等級は nstar コマンドで得られた等級に一致しているはずですが、これも、いくつか質問がありますがひたすらエンターキーを押してください。

```
daophot> substar PTF11kly_R_36196_1.flb3.fits
```

```
Input photometry file (default: image.nst.?) (default):
```

```
Input exclude file (default: image.pst.?):
```

```
PSF image (default: image.psf.?) (default):
```

```
Subtracted image (default: image.sub.?) (default):
```

```
Psf radius in scale units (8.):
```

```
    New psf radius: 8. scale units 8. pixels
```

```
Minimum good data value (INDEF) (CR or value):
```

```
    New minimum good data value: INDEF counts
```

```
Maximum good data value (INDEF) (CR or value):
```

```
    New maximum good data value: INDEF counts
```

```
SUBTRACTING - Star:    4 X = 352.84 Y = 659.40 Mag = 12.56
```

```
SUBTRACTING - Star:    3 X = 263.58 Y = 667.97 Mag = 11.81
```

```
SUBTRACTING - Star:    1 X = 183.07 Y = 716.38 Mag = 11.89
```

```
SUBTRACTING - Star:    2 X = 66.13 Y = 720.96 Mag = 12.20
```

```
A total of 4 stars were subtracted out of a possible 4
```

3.4.5 PSF モデルで差し引いた後の画像をチェックする

最後に、allstar あるいは substar コマンドで差し引いた後の画像 (PTF11kly_R_36195.fb3.sub.1.fits) を元の画像 (PTF11kly_R_36195.fb3.fits) と比較して上手く引けていることを確認してください。綺麗に差し引けて比較的のっぺりした画像が得られていたら、PSF 測光が上手くいっている根拠となります。反対に星のあった位置に大きな穴が開いていたり、取り残しがあったりすると上手くいっていません。この時は、もう一度 psf コマンドで与えたパラメータを編集しなおしたり、imexamine で求めた重心位置が間違っていなかったか、などをチェックし直して測光作業をやり直してください⁷。

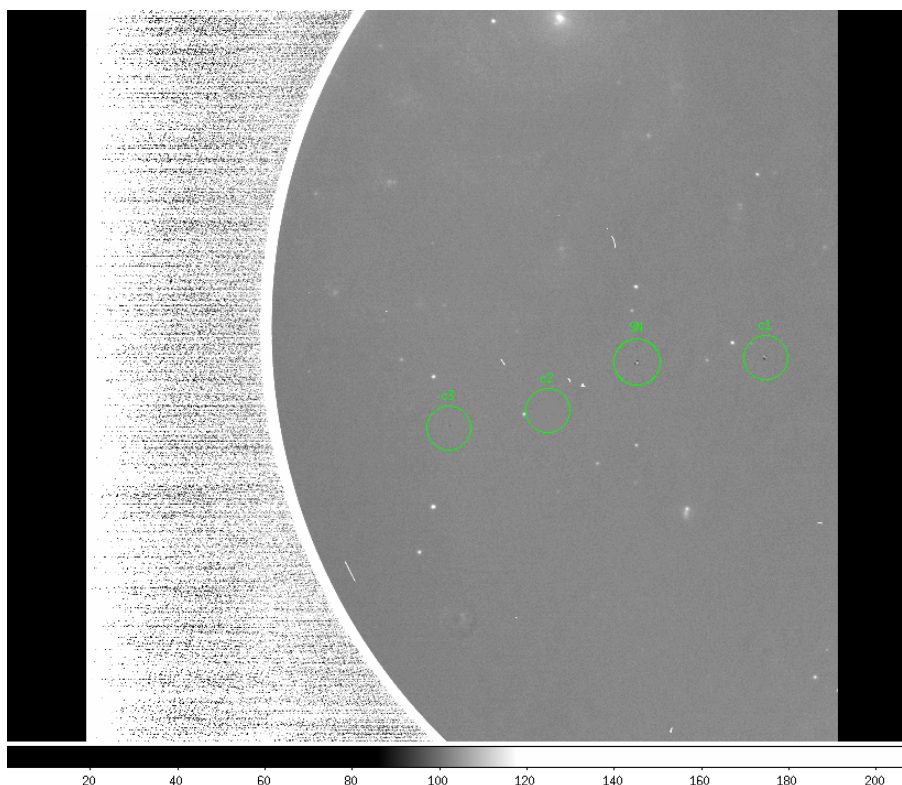


図 3.6: substar コマンドで PSF モデルを差し引いた画像。SN や C1 でわずかに残差が認められるものの比較的上手く差し引けている。

⁷ 差し引いた後に取り残しや穴が残る現象は、多くの場合、オブジェクトや比較星に依存して引き起こされるようです。PSF 星や比較星を変更しても引き残しが見られる場合は、解析で試行錯誤しても解決しないこともあります。そのような場合は、観測をやり直したいですが、突発天体の場合不可能です。緊張感を持って観測に臨みましょう。PSF 測光解析が必要であるようなデータを得る観測になるとあらかじめわかっている場合は、PSF 星に適した星を視野に含めるようにしましょう。

3.5 測光結果の出力

3.5.1 pdump コマンドによる情報の出力

.als.1 ファイルが出力されたならば、pdump コマンドを用いて必要な測光結果を出力しましょう。例えば、image のファイル名 (image)、何番目に測光したか (id)、IRAF 等級 (mag)、等級の統計誤差 (merr)、X 座標位置 (xcenter)、Y 座標位置 (ycenter) の情報を抜き出したいときには以下のようにコマンドを打ちます。

```
daophot> pdump PTF11kly_R_36195_1.flb3.fits.als.1 fields="image id mag merr xcenter ycenter"
```

```
Boolean expression (yes):
```

```
PTF11kly_R_36195_1.flb3  4  12.661  0.006  353.054  659.520
PTF11kly_R_36195_1.flb3  3  11.986  0.004  263.662  667.988
PTF11kly_R_36195_1.flb3  1  12.014  0.011  183.123  716.481
PTF11kly_R_36195_1.flb3  2  12.301  0.012   66.484  721.071
```

pdump コマンドは.mag.1 や.nst.1 ファイルにも適用することができますので、phot コマンドで得られた測光結果 (アパーチャー測光) と比較したい時には同じく pdump コマンドで引き出して置くと良いでしょう。この出力結果をテキストファイルに書き込み保存しておき、IRAF による一連の測光の作業は終わりです。

3.5.2 時刻情報などのヘッダ情報の出力

観測で得られた測光結果から等級の時間変化を調べたい場合には、データ読み出しが行われた日付や時刻、露出時間などを fits ファイルのヘッダから引き出すと良いでしょう。例えば以下のような imheader コマンドが有用です。"lo+" というオプションをつけることによって、ヘッダ情報を羅列することができます。ここでは、linux の grep コマンドを用いて、JST(観測日本時間)、MJD(修正ユリウス日)、DATE(日付) を引き抜いています。ヘッダ情報の種類は観測装置により異なりますので、可能であれば各観測装置の開発担当の責任者の方に問い合わせるのが良いでしょう。

```
daophot> imheader PTF11kly_R_36195_1.flb3.fits lo+ | grep JST
JST      = '21:36:55'           / Typical JST at exposure (hh:m
JSTT-STR= '21:36:55.30'       / JST nearly at exposure start
JSTT-END= '21:37:26.40'       / JST nearly at exposure end (T
```

```
daophot> imheader PTF11kly_R_36195_1.flb3.fits lo+ | grep MJD
MJD      =          55799.525816 / Modified Julian Day at exposu
MJD-STR =          55799.525637 / Modified Julian Day at exposu
MJD-END =          55799.525995 / Modified Julian Day at exposu
MJDT-STR=          55799. / MJD nearly at exposure start
MJDT-END=          55799. / MJD nearly at exposure end (T
```

```
daophot> imheader PTF11kly_R_36195_1.flb3.fits lo+ | grep DATE
DATE     = '2011-08-29T08:59:34' / Date FITS file was generated
DATE-OBS= '2011-08-26'         / Observation start date in UT
```

3.5.3 awk コマンドなどを用いた測光結果の整形

pdump コマンドでは、データの出力は 3.5.1 のような形式となります。ここから目標天体と比較星の等級差 (differential magnitude) を導きたい時に、linux に標準的に実装されている awk コマンドなどを用いれば、非常に簡単に導出することができます。例えば pdump で以下のように情報を hoge.txt に書き込み、

```
daophot> pdump PTF11kly_R_36195_1.flb3.fits.als.1 fields="image id mag merr" > hoge.txt
Boolean expression (yes):
```

以下のように awk、sort、paste コマンドを組み合わせて打てば、目標天体 (ID=1) と比較星 (ID=2) の等級差を求めることが可能です。

```
sort -nk 2 hoge.txt | awk '$3==1 || $3==2 {print $4}' | paste -s | awk '{print $1-$2}'
-0.287
```

この awk コマンドをシェルスクリプトなどに組み込めば、多くの測光結果を一度にまとめることができます。シェルスクリプトについては、インターネット上やクックブックなど多くの資料が存在しておりますので、その詳細についての記述は他に譲ることとします。

第4章 参照すべきリファレンス

ここでは、IRAF/DAOPHOT を使った PSF 測光を習得する上で、参照すべきリファレンスを紹介します。上から順番に難しくなっていきます (本マニュアルは 1.5 番目程度の平易さでしょうか)。IRAF を全く知らない状態で、かつ周囲にいらっしゃいます方がたいへん忙しい時、解析手法を享受することができない時には最初の”IRAF へようこそ”を読まれると良いと思います。これを読んで、IRAF を使った一次処理について習得しておく、本マニュアルの読解も十分に可能となるでしょう。

- ”いらっしゃいませ IRAF へようこそ” http://hamalabo.sakura.ne.jp/Soft/iraf_beginners/
中村隆氏がまとめられた IRAF 初心者に向けた入門書。タスクやパッケージといった言葉まで説明がなされている非常に親切な良書。基本的な Linux コマンドについても言及がある。
- IRAF クックブック 高田唯史著 ”DAOPHOT 簡易マニュアル”
DAOPHOT を用いた PSF 測光の基本的なマニュアル。より高度なユーザーならば、こちらで十分 PSF 測光を習得可能。
- ”IRAF for beginners” http://www.astr.tohoku.ac.jp/mikito/IRAF/task_phot.html
東北大の田中幹人氏がまとめられている DAOPHOT を用いた PSF 測光のマニュアル。Suprime-Cam で得られたデータを想定した、多数の天体検出に特化した記述となっている。
- Phillip Massey & Lindsey E. Davis ”A User’s Guide to Stellar CCD Photometry with IRAF”
DAOPHOT のユーザーズガイド。一部が本マニュアルと同じような内容となっている。ただし、当然英語で記述されている。
- Lindsey E. Davis 1994, ”A Reference Guide to the IRAF/DAOPHOT Package”
DAOPHOT の使い方について記述されているマニュアル。それぞれのタスクやパラメータの意味についての詳細な記述がある。
- Stetson, Peter B. 1987, PASP, 99, 191 ”DAOPHOT - A computer program for crowded-field stellar photometry”
PSF 測光の概念とその有用性について書かれている論文。
- Stetson, Peter B. 1990, PASP, 102, 932 ”ON THE GROWTH-CURVE METHOD FOR CALIBRATING STELLAR PHOTOMETRY WITH CCDs”
CCD で得られる星像曲線について詳細な考察がなされている。aperture correction への示唆もある。
- Harris, William E. 1990, PASP, 102, 949 ”A comment on image detection and the definition of limiting magnitude”
アパーチャー測光における限界等級についての記述。

第5章 Appendix

5.1 PSF 星の自動選出

本マニュアルでは、`phot` コマンドで目標天体と比較星を測光した後に、`psf` コマンドでインタラクティブに PSF 星を選択する方法を紹介しました。しかしながら、一晩で何時間もの連続測光をした場合や、大量のフレームを測光処理しなければならない場合に、一つ一つ手で解析しては膨大な時間を取られてしまいます。そこで、ここでは `pstselect` という便利なコマンドを紹介します。

5.1.1 `pstselect` コマンド

`pstselect` コマンドの使い方は非常に簡単です。引数に天体の `fits` ファイルを与えるだけです。

```
daophot> pstselect PTF11kly_R_36195_1.flb3.fits
```

```
Photometry file (default: image.mag.?) (default):
Output psf star list file (default: image.pst.?) (default):
Maximum number of psf stars (1:) (25):

Psf radius in scale units (8.):
    New psf radius: 8. scale units 8. pixels
Fitting radius in scale units (6.):
    New fitting radius: 6. scale units 6. pixels
Minimum good data value (INDEF) (CR or value):
    New minimum good data value: INDEF counts
Maximum good data value (INDEF) (CR or value):
    New maximum good data value: INDEF counts

Selecting PSF stars for image PTF11kly_R_36195_1.flb3.fits
    4 stars read from file PTF11kly_R_36195_1.flb3.fits.mag.2

Star 3 has been added to the PSF star list
    X: 263.59 Y: 667.96 Mag: 12.017 Dmin: 542.5501 Dmax: 2992.55
Star 1 has been added to the PSF star list
    X: 183.20 Y: 716.47 Mag: 12.046 Dmin: 546.2806 Dmax: 2873.176
Star 2 has been added to the PSF star list
    X: 66.46 Y: 721.17 Mag: 12.344 Dmin: 536.8669 Dmax: 2330.453
Star 4 has been added to the PSF star list
    X: 353.06 Y: 659.52 Mag: 12.683 Dmin: 539.7638 Dmax: 1724.03

Total of 4 PSF stars selected
```

続いて、psf コマンドを以下のように編集いたします。本編で紹介しましたパラメータとは一部異なります。interactive は”no”に、showplot も”no”としてください。また、pstfile は”default”として、先ほど生成された.pst.1 ファイルが引用されるようにしてください。最後に”:wq”で保存してください。

```
daophot> epar psf
```

```
PACKAGE = daophot
TASK = psf

image = PTF11kly_R_36195_1.flb3.fits Input image(s) for which to build PSF
photfile= default Input photometry file(s) (default: image.mag.?)
pstfile = default Input psf star list(s) (default: image.pst?)
psfimage= default Output PSF image(s) (default: image.psf.?)
opstfile= default Output PSF star list(s) (default:image.pst.?)
groupfil= default Output PSF star group file(s) (default: image.psg.?)
(plotfil= ) Output plot metacode file
(datapar= ) Data dependent parameters
(daopars= ) Psf fitting parameters
(matchby= yes) Match psf star list to photometry file(s) by id number ?
(interac= no) Compute the psf interactively ?
(mkstars= no) Mark deleted and accepted psf stars ?
(showplo= no) Show plots of PSF stars ?
(plottyp= mesh) Default plot type (mesh|contour|radial)
(icomman= ) Image cursor: [x y wcs] key [cmd]
(gcomman= ) Graphics cursor: [x y wcs] key [cmd]
(wcsin = )_.wcsin) The input coordinate system (logical,tv,physical,world)
(wcsout = )_.wcsout) The output coordinate system (logical,tv,physical)
(cache = )_.cache) Cache the input image pixels in memory ?
(verify = )_.verify) Verify critical psf parameters ?
(update = )_.update) Update critical psf parameters ?
(verbose= )_.verbose) Print psf messages ?
(graphic= )_.graphics) Graphics device
(display= )_.display) Display device
(mode = ql)
```

あとは、psf コマンドを実行するだけです。

```
daophot> psf PTF11kly_R_36195_1.flb3.fits
```

```

Input photometry file(s) (default: image.mag.?) (default):
Input psf star list(s) (default: image.pst?) (default):
Output PSF image(s) (default: image.psf.?) (default):
Output PSF star list(s) (default:image.pst.?) (default):
Output PSF star group file(s) (default: image.psg.?) (default):

Analytic psf function(s) (auto):
    Analytic psf function(s): auto
Order of variable psf (0):
    Order of variable psf: 0
Psf radius in scale units (8.):
    New psf radius: 8. scale units 8. pixels
Fitting radius in scale units (6.):
    New fitting radius: 6. scale units 6. pixels
Minimum good data value (INDEF) (CR or value):
    New minimum good data value: INDEF counts
Maximum good data value (INDEF) (CR or value):
    New maximum good data value: INDEF counts

Computing PSF for image: PTF11kly_R_36195_1.flb3.fits
4 stars read from PTF11kly_R_36195_1.flb3.fits.mag.2

Star 3 has been added to the PSF star list
    X: 263.59 Y: 667.96 Mag: 12.017 Dmin: 542.5501 Dmax: 2992.55
Star 1 has been added to the PSF star list
    X: 183.20 Y: 716.47 Mag: 12.046 Dmin: 546.2806 Dmax: 2873.176
Star 2 has been added to the PSF star list
    X: 66.46 Y: 721.17 Mag: 12.344 Dmin: 536.8669 Dmax: 2330.453
Star 4 has been added to the PSF star list
    X: 353.06 Y: 659.52 Mag: 12.683 Dmin: 539.7638 Dmax: 1724.03

4 PSF stars read from PTF11kly_R_36195_1.flb3.fits.pst.3

Trying function gauss norm scatter = 0.01388361
Trying function lorentz norm scatter = 0.02623706
Trying function moffat15 norm scatter = 0.01586879
Trying function moffat25 norm scatter = 0.008127699
Trying function penny1 norm scatter = 0.008845821
Trying function penny2 norm scatter = 0.008867996
Best fitting function is moffat25

Analytic PSF fit
    Function: moffat25 X: 341. Y: 703. Height: 1799.797 Psmag: 12.017
    Par1: 0.9804008 Par2: 0.8703144 XYterm: -0.08386066 Moffat: 2.5
Computed 1 lookup table(s)
Writing PSF image PTF11kly_R_36195_1.flb3.fits.psf.2.fits
Writing output PSF star list PTF11kly_R_36195_1.flb3.fits.pst.4
Writing output PSF star group file PTF11kly_R_36195_1.flb3.fits.psg.2

```

これで psf モデルは作成されました。この後、seepsf で PSF モデルをチェックするか、allstar コマンドでフィッティングを行ってください。ちなみに、imexamine で目標天体と比較星を測光した後は、phot、pstselect、psf、allstar とひたすらエンターキーを押すのみですので、これらを一つのスクリプトに組み込めば、ほぼ測光を自動化することができます。

5.2 IRAF コマンドのシェルスクリプトへの組み込み

ここでは、IRAF コマンドをシェルスクリプトに組み込む方法を紹介いたします。測光のタスクをスクリプト化することによって解析を自動化できますので、たいへん有用な手法の一つとなります。まず以下のような文字列を emacs やテキストエディタ (gedit) などを開いて書き込み、適当な名前の.sh ファイル (allstar.sh など) として保存してください。

```
#!/bin/sh

for n in `ls $1*.fits`
do

image=$n

printf "noao \n\
digiphot \n\
daophot \n\
allstar image=$image \n\
logout " | cl

done
```

ここで書かれている文字列の意味については、シェルスクリプトの教本などでしっかり勉強していただきたく思いますが、簡単に紹介します。冒頭の”#!/bin/sh”は#!以降に書かれた命令を実行するという意味です。おまじないだと思って下さい。for n in; do — done は繰り返し (ループ) を意味します。また、ls 以下の命令はカレントディレクトリに例えば hoge1.fits、hoge2.fits、hoge3.fits が存在するとして、これらををリスト化せよ、という意味です。ここでは、最初のループで hoge1.fits を”n”に入れて、”done”まで命令した後に、また次の hoge2.fits を”n”に代入する、ということになります。”\$1”はこのスクリプト (allstar.sh) で与えるべき引数です。ここでは、文字列”hoge”ということになります。ファイル名の冒頭が天体名となっている場合、”hoge”の代わりに、”M51”だとか”M101”と書けるわけです。image=\$n は image という変数を定義し、かつ \$n の文字列を代入しています。

さて、IRAF コマンドをシェルスクリプトに組む上で最も肝要な命令が”printf”です。これは単にダブルコーテーション””で囲まれた命令 (文字列) を呼び出しなさい、という意味ですが、ダブルコーテーションで括った後にパイプ (縦棒のことです。) で cl としてやると、なんとダブルコーテーションで囲まれた内容の IRAF の命令を順次、実行することが可能なのです (最後の logout は必ず入れて下さい)。すなわち、このコマンドによって hoge1.fits、hoge2.fits、hoge3.fits を引数として、allstar コマンドを実行するというのがこの一つのシェルスクリプトで可能となったのです。

この便利なスクリプトを実行するためには、

```
chmod +x allstar.sh
```

などとして実行権を与え、

```
./allstar.sh hoge
```

とすれば、実行完了です。これで自動的にたくさんの als.1 ファイルを生成できるようになりました。これを例にとって、様々な IRAF コマンドをシェルスクリプトに組み込んでやれば、様々な仕事を自動化することが可能となります。