

Astrofísica Observacional

Práctica 2: Calibración y análisis de las imágenes astronómicas

1 Introducción

El objeto de esta práctica es el análisis y la eliminación de la signatura instrumental en imágenes astronómicas. Utilizaremos imágenes obtenidas con la cámara CCD Apogee en el telescopio TROBAR de 60 cm. del Observatorio de Aras de los Olmos. El trabajo lo haremos mediante el uso del paquete de programas IRAF.

Podemos acceder a IRAF en el escritorio virtual linux Ubuntu de la universidad, siguiendo el enlace:

<https://escritorio.uv.es/portal/webclient/index.html>

2 Herramientas de IRAF

Para inicializar IRAF, clicamos sobre el icono correspondiente en las aplicaciones del escritorio virtual Linux Ubuntu. Se abrirá un terminal con el cursor del “command language”

```
ec1>
```

En primer lugar tecleamos la instrucción:

```
set stdimage=imt2048
```

A continuación nos dirigimos al directorio de trabajo, en el cual hemos descargado previamente las imágenes a analizar, y abrimos una ventana gráfica para visualizarlas con la instrucción:

```
!ds9 &
```

Técnicas de análisis y corrección:

- Estadística de la imagen:

```
imstat nombre_de_la_imagen
```

- Análisis de las imágenes: usamos la instrucción de IRAF “imexamine”. Para ello tecleamos

```
display nombre_de_la_imagen  
imexam
```

y a continuación picamos con el botón izquierdo del ratón en la barra superior de la ventana que contiene ds9. Para salir del “imexamine” tecleamos q.

Dentro de imexamine, podemos ejecutar las siguientes operaciones:

c dibuja una columna de la imagen
e mapa de contornos
j ajuste gaussiano unidimensional en la dirección de la línea
k ajuste gaussiano unidimensional en la dirección de la columna
l dibuja una línea de la imagen
r ajuste gaussiano radial
s dibuja la superficie
q sale de imexamine

- Para operaciones aritméticas con las imágenes empleamos la tarea “imarith”, que tiene la sintaxis siguiente:

```
imarith operand1 op operand2 result
```

donde:

- “operand”, “operand2” son imágenes o constantes.
- “op” es el operador a aplicar. Los operadores posibles son: +, -, *, /, min y max.
- “result” es la imagen resultante.

Para finalizar la sesión con IRAF tecleamos:

```
logout
```

3 Desarrollo de la práctica

La primera imagen a visualizar es una imagen de la corriente de polarización, o “bias”. La abrimos con la orden

```
display Bias-001 1
```

En primer lugar hacemos la estadística de la imagen, con la instrucción

```
imstat Bias-001
```

A continuación la analizamos con el programa “imexamine”

```
imexam
```

Una vez ejecutada esta orden, activamos la ventana gráfica con un click del ratón, y ejecutamos cualquiera de los comandos descritos anteriormente. Con la imagen de polarización son útiles los cortes horizontales y verticales, con los comandos “l” y “c”. Salimos de “imexam” con “q”.

Hacemos un estudio similar de las imágenes de oscuridad, cuyo nombre empieza con “dark”, y con las de campo uniformemente iluminado, cuyo nombre empieza con “flat”.

Finalmente, analizamos una imagen astronómica. Cualquiera de las que empieza por “NGC” en los datos de la práctica. En su estudio con imexam usaremos, además de “l” y “c”, las instrucciones “e”, “j”, “k”, “s” y “r”, centrando el cursor sobre diferentes estrellas de la imagen.

4 Calibración de imágenes astronómicas con IRAF

La colección de programas que utilizaremos para la eliminación de la signatura instrumental se encuentra en el paquete “ccdred”. Accedemos a él tecleando:

```
noao
imred
ccdred
```

En primer lugar hemos de comprobar que el sistema es capaz de reconocer los datos principales de nuestras imágenes, tales como el tipo de imagen correspondiente a cada fichero. Esto lo comprobamos tecleando

```
setinstrument direct
<ctrl>+d
<ctrl>+d
```

y a continuación

```
ccdlist
```

y respondiendo “*” al nombre de la imagen. Nos aparecerá un listado como el de la Figura 1. Hemos de comprobar que el sistema reconoce el tipo de imagen, que debe aparecer dentro de los últimos corchetes.

Cuando todo está en orden, procedemos a corregir la signatura instrumental. Lo hacemos en tres pasos:

1. Corrección de la corriente de polarización:

En primer lugar, obtenemos una media de las imágenes de corriente de polarización. Lo hacemos con la instrucción

```
zerocombine
```

que calcula una imagen media de corriente de polarización que llama Zero. Podemos ver con “imstat” que su desviación media es mucho menor que en las imágenes individuales (ver ejemplos en la Figura 2).

A continuación restamos la corriente de polarización media al resto de las imágenes usando “ccdproc”. Para ello editamos los parámetros de “ccdproc”

```
epar ccdproc
```

e indicamos que la única corrección a realizar en este primer paso es la corrección de bias. Configuramos por tanto el fichero de parámetros tal como aparece en la Figura 3.

Una vez configurado salimos del editor con

```
<ctrl>+d
```

y ejecutamos el programa:

```

|ccdred> ccdlist
|CCD images to listed (*):
|Warning: Cannot open image (Documents)
|Flat1.fit[1024,1024][real][flat][][T]:
|Flat2.fit[1024,1024][real][flat][][T]:
|Flat3.fit[1024,1024][real][flat][][T]:
|Flat4.fit[1024,1024][real][flat][][T]:
|Flat5.fit[1024,1024][real][flat][][T]:
|Flat6.fit[1024,1024][real][flat][][T]:
|Flat7.fit[1024,1024][real][flat][][T]:
|Flat8.fit[1024,1024][real][flat][][T]:
|bias001.fit[1024,1024][real][zero][][T]:
|bias002.fit[1024,1024][real][zero][][T]:
|bias003.fit[1024,1024][real][zero][][T]:
|bias004.fit[1024,1024][real][zero][][T]:
|bias005.fit[1024,1024][real][zero][][T]:
|bias006.fit[1024,1024][real][zero][][T]:
|bias007.fit[1024,1024][real][zero][][T]:
|bias008.fit[1024,1024][real][zero][][T]:
|bias009.fit[1024,1024][real][zero][][T]:
|bias010.fit[1024,1024][real][zero][][T]:
|Warning: Cannot open image (bin)
|dark001.fit[1024,1024][real][dark][][T]:
|dark002.fit[1024,1024][real][dark][][T]:
|dark003.fit[1024,1024][real][dark][][T]:
|dark004.fit[1024,1024][real][dark][][T]:
|dark005.fit[1024,1024][real][dark][][T]:
|dark006.fit[1024,1024][real][dark][][T]:
|dark007.fit[1024,1024][real][dark][][T]:
|dark008.fit[1024,1024][real][dark][][T]:
|dark009.fit[1024,1024][real][dark][][T]:
|dark010.fit[1024,1024][real][dark][][T]:
|i01.fit[1024,1024][real][object][][T]:
|i02.fit[1024,1024][real][object][][T]:
|i03.fit[1024,1024][real][object][][T]:
|i04.fit[1024,1024][real][object][][T]:
|i05.fit[1024,1024][real][object][][T]:
|i06.fit[1024,1024][real][object][][T]:
|i07.fit[1024,1024][real][object][][T]:
|i08.fit[1024,1024][real][object][][T]:
|i09.fit[1024,1024][real][object][][T]:
|i10.fit[1024,1024][real][object][][T]:
|i11.fit[1024,1024][real][object][][T]:
|i12.fit[1024,1024][real][object][][T]:

```

Figure 1: Listado de las imágenes con `ccdlist`

```

Oct 1 15:59: IMCOMBINE
  combine = average, scale = none, zero = none, weight = none
  reject = minmax, nlow = 0, nhigh = 1
  blank = 0.
    Images
    bias001.fit
    bias002.fit
    bias003.fit
    bias004.fit
    bias005.fit
    bias006.fit
    bias007.fit
    bias008.fit
    bias009.fit
    bias010.fit

  Output image = Zero, ncombine = 10
ccdred> imstat bias001
#      IMAGE      NPIX      MEAN      STDEV      MIN      MAX
      bias001  1048576    954.6    15.86     886.    4861.
ccdred> imstat bias007
#      IMAGE      NPIX      MEAN      STDEV      MIN      MAX
      bias007  1048576    943.2    14.05     883.    2123.
ccdred> imstat Zero
#      IMAGE      NPIX      MEAN      STDEV      MIN      MAX
      _      Zero  1048576    943.7     5.136    921.2    1502.

```

Figure 2: Corriente de polarización media con zerocombine

ccdproc

La corrección se realiza automáticamente en todas las imágenes. Si volvemos a hacer un listado con

ccdlist

Vemos que en la lista aparece que las imágenes están corregidas de bias.

2. Corrección de la corriente de oscuridad:

El procedimiento es muy parecido. En primer lugar obtenemos una imagen de oscuridad media con

darkcombine

Esta imagen se llama “Dark”. En el proceso de promediado se ha tenido en cuenta que el nivel medio de las imágenes de oscuridad depende del tiempo de exposición. La imagen promedio Dark es la corriente de oscuridad por unidad de tiempo. Al corregir de oscuridad, esa corriente media se multiplicará por el tiempo de exposición de cada imagen.

A continuación corregimos de oscuridad pasando por segunda vez por “ccdproc”. Antes editamos los parámetros, ponemos “no” en la corrección de bias, “yes” en la de oscuridad e introducimos el nombre

I R A F

Image Reduction and Analysis Facility

```
PACKAGE = ccdred
TASK = ccdproc

images = List of CCD images to correct
(output = ) List of output CCD images
(ccdtype= ) CCD image type to correct
(max_cac= 0) Maximum image caching memory (in Mbytes)
(noproc = no) List processing steps only?

(fixpix = no) Fix bad CCD lines and columns?
(oversca= no) Apply overscan strip correction?
(trim = no) Trim the image?
(zero = yes) Apply zero level correction?
(darkcor= no) Apply dark count correction?
(flatcor= no) Apply flat field correction?
(illumco= no) Apply illumination correction?
(fringec= no) Apply fringe correction?
(readcor= no) Convert zero level image to readout correction?
(scancor= no) Convert flat field image to scan correction?

(readaxi= line) Read out axis (columnline)
(fixfile= ) File describing the bad lines and columns
(biassec= image) Overscan strip image section
(trimsec= image) Trim data section
(zero = Zero) Zero level calibration image
(dark = ) Dark count calibration image
(flat = ) Flat field images
(illum = ) Illumination correction images
(fringe = ) Fringe correction images
(minrepl= 1.) Minimum flat field value
(scantyp= shortscan) Scan type (shortscan/longscan)
(nscan = 1) Number of short scan lines

(interac= yes) Fit overscan interactively?
(funcio= chebyshev) Fitting function
(order = 1) Number of polynomial terms or spline pieces
(sample = *) Sample points to fit
(naverag= 1) Number of sample points to combine
(niterat= 1) Number of rejection iterations
(low_rej= 3.) Low sigma rejection factor
(high_re= 3.) High sigma rejection factor
(grow = 0.) Rejection growing radius
(mode = ql)
```

Figure 3: Fichero de parámetros de ccdproc para realizar la corrección de bias

del fichero de oscuridad promedio “Dark”. Ejecutamos

```
ccdproc
```

y luego

```
ccdlist
```

para comprobar que la corrección se ha realizado correctamente.

3. Corrección de variaciones de sensibilidad (“Flat field”)

Procedemos como en los casos anteriores. El programa que calcula los flat medios es

```
flatcombine
```

Antes de ejecutarlo hemos de editar su fichero de parámetros:

```
epar flatcombine
```

e indicar los valores de la ganancia (“gain”) y del ruido de lectura (“readnoise”), que para la CCD empleada (Finger Lakes Instruments) son $4.0 e^-/ADU$ y $10.0 e^-$ respectivamente. También hemos de asegurarnos que el parámetro “process” tenga el valor “no”.

La corrección de flat la hacemos de nuevo con “ccdproc”. Antes editamos los parámetros, ponemos “no” en la corrección de dark, “yes” en la de flat e introducimos el nombre del fichero de flat promedio “Flat.fits”.

En el caso de que estemos haciendo fotometría con varios filtros, “flatcombine” creará varios flat promedio, que denominará “FlatB.fits”, “FlatV.fits”, etc. Si es así, en la configuración de “ccdproc”, como nombre del flat promedio introduciremos “Flat*”. Después de esto, ejecutamos

```
ccdproc
```

y luego

```
ccdlist
```

para comprobar que la corrección se ha realizado correctamente. Finalmente, examinamos las imágenes corregidas, comparándolas con las imágenes brutas.