omdetect

August 16, 2005

Abstract

Program for detecting sources on OM images, parameterising and performing aperture photometry on those detected. Program is also used for grism images to detect zero and first order spectra.

1 Instruments/Modes

Instrument	Mode
OM	FAST
OM	IMAGING

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

This task determines point and extended sources in 2 separate operations. In order to detect sources the program computed the mode of the background, and then any pixel with a value greater than or equal to nsigma * backgound mode is considered to be likely associated with a source. In detecting point sources, each such pixel in the image is analaysed to see if it may be centred on a point-source. If a likely point-source is found, the pixels assigned to the source are set equal to the source number and its parameters calculated and stored. All other pixels above the detection threshold are then assumed to be associated with "extended" sources.

Having obtained a source list, comprising their positions, sizes, counts and background counts, aperture-photometry is then performed on each source to produce a count value for each source that is consistent with the CAL calibration procedure.

The output file is a source list comprising source positions and error (in pixels), parameters measuring the FWHM of the major and minor axes of an ellipse characterising the source shape, source count-rate and error, psf1 correction (point-spread function correction applied to counts to extrapolate from source extraction area to coincidence-loss area), aperture radius, no of counts associated with an extended

source, coincidence-loss correction, significance of the source and flags reflecting whether the source is affected by quality, confusion or extension problems. A count-rate limit is written to the source list file header (keyword CRLIMIT) reflecting the mean nsigma above background detection threshold for the image, where nsigma is a parameter to the task.

4 Parameters

This section documents the parameters recognized by this task (if any).

Parameter	Mand	Type	Default	Constraints
set	yes	string	none	

name of OM OSW FITS image file

outset	yes	string	none	

name of output OM OSW source list file

wdxset	no	string	""	

name of wdx file - needed for images produced by pre 5.4 sas

nsigma	no	real	2.0	1.0–

number of σ above background required for a detection

minsignificance	no	real	1.0	0.0–

minimum significance of sources to be included in OM OSW file. if 0 then no checking done.

levelimage	no	string	none	

name of image of island detections

backgroundimage	no	string	none	

name of background image file

regionfile	no	string		

name of saoimage region file

detectextended	no	bool	true	

Determines whether algorithm for detecting extended sources should be used

5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

Source detection problem- unreliable source list *(warning)*

corrective action: A problem was encountered when trying to deblend a source

image file is not 2D *(fatal)*

different binning in x and y axes *(fatal)*

Could not allocate memory for: *name (fatal)*
Memory allocation for the named allocatable array failed

WrongFilter *(fatal)*
Filter in not one of White, WHITE, V, B, U, UVW1, UVW2, UVM2, GRISM1, GRISM2 or MAGNIF

Failed to release memory for: *name (warning)*

corrective action: Memory deallocation for the named allocatable array failed

6 Input Files

1. Intermediate FITS image file (output from OMMODMAP)

7 Output Files

1. Intermediate OM OSW source list (input to OMWAVELET)
2. Source number image (optional))
3. Saoimage region file of source list (optional)

8 Algorithm

```
subroutine omdetect

read data for selected OSW image

get mode of image by sigma clipping to get average
background level
Loop though all pixels on the image- for each pixel with a value >=nsigma *
background mode analyse surrounding pixels to ascertain if these pixels may
constitute a ‘‘point’’ source. If so, set the pixels to an assigned source
number, compute the source position and shape by moments, and store the
parameters. Repeat until all pixels have been checked.



Recompute the background mode, ignoring those pixels assigned to a point-source.
Check all the remaining pixels and those with a value >=nsigma *
background mode assign a number. Analyse these pixels for individual extended
sources and  assign a number to each one. Compute the source position and shapes of
each extended source by moments, and store the parameters.


Perform aperture photometry on each source.
Examine each point-source and discard any with a significance less than the
minimum allowed value.

Examine each extended source to attempt to see if it is just scattered light
around a point-source- reject this source if it is.

write output sourcelist containing source parameters. Also output
detection count rate threshold.


end subroutine  omdetect
```

9 SourceListFile

- COLUMN NO NAME UNITS DESCRIPTION
- 1 SRCNUM none
- 2 XPOS pixels x-coordinate of centroided source centre
- 3 YPOS pixels y-coordinate of centroided source centre
- 4 POSERR pixels error in x and y centroided position
- 5 RATE counts/sec Total measured source counts (after background subtraction) within default source extraction aperture (r=12 un-binned pixels), divided by exposure time.
- 6 RATE_ERR counts/sec
- 7 BACKGROUND_RATE counts/sec Total estimated background counts (scaled by area ratio, from the background annulus) within default source extraction aperture, divided by exposure time

- 8 SIGNIFICANCE none source counts / sqrt(source + background counts) (i.e. signal-to-noise ratio)
- 9 CORR_RATE counts/sec Background subtracted source count rate after extrapolating (if necessary) for the wings of the PSF where they extend beyond the default extraction aperture, and correcting for coincidence losses, time-degradation sensitivity and dead-time.
- 10 CORR_RATE_ERR counts/sec
- 11 CORR_BACKGROUND_RATE counts/sec BACKGROUND_RATE after correcting for coincidence losses.
- 12 PSF1_CORR percentage Point-spread function correction applied to extrapolate the RATE value from measurement aperture area to the default extraction area of 12 unbinned pixels
- 13 COINCLOSS_CORR percentage Coincidence-loss correction applied - (CORR_RATE - RATE) * 100.0 / RATE
- 14 NOPIXELS none Number of pixels contained by an extended source (EFLAG=1). Point-sources (EFLAG=1) will have this value set to zero.
- 15 APERTURE unbinned pixels For point sources (EFLAG=0), the aperture radius (unbinned pixels) used in the photometry. For extended-sources (EFLAG=1) this value will be set to NULL.
- 16 FWHM_MAJ arcsecs The full-width-at-half-maximum of the semi-major axis of the ellipse describing the shape of the source (computed by moments)
- 17 FWHM_MIN arcsecs The full-width-at-half-maximum of the semi-minor axis of the ellipse describing the shape of the source (computed by moments)
- 18 PA degrees The postion-angle (measured counterclockwise from the x-axis) of the semi-major axis of the ellipse describing the shape of the source (computed from moments)
- 19 QFLAG none Source quality flag - Please see below
- 20 CFLAG none Source confusion flag - Please see below
- 21 EFLAG none (source extension flag) Either 0 point-source (none-extended) or 1 extended.

Quality flag settings

- **Bit set, integer value, reason**
- **0, 1, Source lies above one or more bad pixels**
- **1, 2, Source may be affected by a read-out streak**
- **2, 4, Source may be affected by a smoke-ring**
- **3, 8, Source may be affected by a star diffraction spike**
- **4, 16, Very bright source with a strong modula 8 pattern**
- **5, 32, Source lies within central enhanced region**

For example, a source with a bad pixel and lying on or close to a read-out streak would have bits 0 and 1 set and thus have a quality flag value of 3.

Confusion flag settings

- **Bit set, integer value, reason**
- **0, 1, Point source has one or more sources within a radius of 6-12 unbinned pixels**

- **1, 2, Point source has one or more sources within a radius of 3-6 unbinned pixels**
- **2, 4, Source lies near to an edge of the image**
- **3, 16, Source lies near to a bright source**
- **4, 32, Point source lies partly or totally within an extended region**

  For example, a source lying near to an image edge and having a neighbour at a distance of 10 unbinned pixels would have bits 2 and 1 set and would thus have a confusion flag value of 5.

10 Deficiencies

- If two point-sources are close together so that their images overlap, the detection algorithim for detecting point-sources will either fail to detect one or both. However, the detection algorithm for detecting extended sources will either pick up the missing object or detect both as a single extended source.

11 Photometry

When the source-detection algorithms has run **omdetect** will compute raw count-rates for each detected source. For point-sources (ie those sources having widths consistent with those of the point-spread function) the count-rates are obtained by aperture photometry. The background is subtracted using an annulus region lying just outside the circular aperture region.

These "raw" count-rates have to be corrected for coincidence-loss, dead-fraction and the time-dependent sensitivity degradation before instrumental magnitudes can be computed (using the task **ommag**). The default aperture is 12 unbinned pixels, but for some sources a smaller aperture may be used. For these cases an aperture correction is also applied using the point-spread function. For the filters UVW1, UVM2 and UVW2 a further point-spread function correction is made to extrapolate the count-rates to an unbinned aperture radius of 35 pixels.

For extended sources a different method is used to obtain the count-rates, since the above method is invalid for them. Each pixel of the source region is treated as a point-source and the counts obtained by summing the coincidence-loss corrected pixel counts. Please note that this method is only approximate- a more accurate method would require detailed modelling of the source.

12 Region file

A region file is created if the parameter **regionfile** is set equal to some file name. When the input image is viewed using **ds9** the sources can be overlaid on the image by selecting the region file. The regions will be plotted in different colours, depending on their quality and confusion flag values. Unfortunately, only one flag setting can be depicted - the region colour is selected by the following rules:

- If bit 2 of the source confusion flag is set then the source will be drawn in blue.
- if the source quality flag value is 1 the source will be drawn in red.
- if the source quality flag value is either 2 or 3 the source will be drawn in magenta.

- if the source quality flag value lies between 4 and 7 the source will be drawn in yellow.
- if the source quality flag value lies between 8 and 15 the source will be drawn in white.
- if the source quality flag value lies between 16 and 63 the source will be drawn in black.
- if the source quality flag value lies between 64 and 127 the source will be drawn in red.
- If both the quality and confusion flags are 0 the source will be drawn in blue, if the source is a point-source, or cyan otherwise.

13 Comments

- We have seen images that have been generated by OMFLATFIELD that contain pixel values that have been set to the BLANK keyword value. This occurrs due to the Flatfield being zero for these pixels. The quality array has been flagged to the value of 16 for these cases. The image has been modified to contain the value zero as a temporary fix.

14 Future developments

References