ombackground

February 1, 2016

Abstract

Program for comparing various background-determination methods on OM images.

1 Instruments/Modes

| Instrument	Mode
OM	FAST
OM	IMAGING

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

The main purpose of the program is to compare the default **OM** background-determination algorithm that **omdetect** uses when performing aperture-photometry with the mean, median and mode estimators.

The idea is that if you have an image with no sources and you compute the background-subtracted counts within square apertures placed on the image, the mean of these values should be zero for a good background algorithm. The program produces various plots that are stored in the postscript file specified by the parameter **plotfile**.

1. **Graph 1** A histogram of the distribution of pixel values on the image specified by the parameter **imageset**. Various estimates of the mean background-counts per pixel (using all the good pixels) are computed.

2. **Graph 2** Histograms of the distribution of computed background-counts per pixel for the various background algorithms, using square apertures **that try to avoid source regions**.

3. **Graph 3** Histograms of the distribution of computed background-subtracted counts per pixel for the various background algorithms, using square apertures **that try to avoid source regions**.
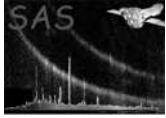
4. **Graph 4** Histograms of the distribution of computed background-counts per pixel for the various background algorithms, using square apertures **that try to avoid source regions**.

5. **Graph 5** Histograms of the distribution of computed background-subtracted counts per pixel for the various background algorithms, using square apertures that **do not try to avoid source regions**.

From SAS 10 onwards it is possible to change the default background-determination method that **omdetect** use through its parameter **backgroundmethod**. It is not advised to change it without very good reasons.

If required, the program will compute a background image and store it in a **FITS** file- the algorithm used is nearly identical to that used by **omdetect**.

3.1 Background algorithms

3.1.1 Current SAS algorithm (SAS 10)

The algorithm is based on the **sextractor** algorithm with a few modifications for low backgrounds (as in the UV).

3.1.2 Mean

Using the mean of the pixel values to obtain an estimate of the background is risky because one or more source pixels can seriously effect its value.

3.1.3 Median

The median can give a good estimate of the background value in moderately crowded fields (ie no more than 50 percent of the pixels used are contaminated by sources).

3.1.4 Mode algorithm

Various photometry packages use the modal value for the background value, but most compute it from the empirical relationship between the mean, median, and mode, which appears to hold for unimodal curves of moderate asymmetry and is given by mean-mode approx 3(mean-median). Bickel, D.R. 2002 ("Robust estimators of the mode and skewness of continuous data", Bickel, D.R., 2002, Computational Statistics and Data analysis Volume 39, Issue 2, pages 153-163) has presented a simple and robust algorithm to compute the mode. Our background-mode algorithm is based on his algorithm with a minor modification for low pixel values.

It can give a good estimate of the background value in crowded fiels (ie even if up to 75 percent of the pixels used are contaminated by sources).

3.2 The Sn value

On top of some of the graphs you will notice that an Sn value is given- this parameter gives a robust estimate of the data spread (standard deviation), and is computed using

$$Sn = cn \times 1.1926 \times LOMED_i HIMED_i \mid x_i - x_j \mid$$

Please see the paper by Rousseeuw, P.J. and Croux, C. (1993), "Alternatives to the Median Absolute Deviation," Journal of the American Statistical Association, Vol. 88, 1273-1283.

For an image with pure poisson noise and few sources the Sn value should be roughly equal to the mean of all the image pixel values.

4 Parameters

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

Parameter	Mand	Type	Default	Constraints
imageset	yes	string		

Name of the OM product image file to process

backgroundset	no	string		

Name of the file to store the computed background image

plotfile	no	string	ombackground.ps	

Name of the postscrip plot file for the graphs

boxwidth	no	integer	40	2:

Size of box size in unbinned pixels to be used for computing background image.

5 Usage

- **ombackground imageset=testimage.fits** - processes the OM image stored in the file testimage.fits. The plots are stored in the file ombackground.ps. No background image is computed.

- **ombackground imageset=testimage.fits backgroundset=backgroundimage.fits** - processes the OM image stored in the file testimage.fits. The plots are stored in the file ombackground.ps. A background image is computed and stored in the file backgroundimage.fits.

- **ombackground imageset=testimage.fits plotfile=plot.ps** - processes the OM image

stored in the file testimage.fits. The plots are stored in the file plot.ps. No background image is computed.

6   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.

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

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

  corrective action: Memory deallocation for the named allocatable array failed

7   Input Files

1. OM product image files (those produced by **ommodmap**)

8   Output Files

1. **P**ostscript plot file.

2. **FITS** 2-d image file of the computed background image (optional)

9   Algorithm

```
1)   Create a list of all the useable (ie non-null) pixels in the image.
     Loop through the differeent background methods and compute the background value (ie
     counts per pixel). Draw Graph 1.


2)   Place square apertures on the image, trying to avoid sources, and for each one
     use the pixels within it to compute a background using all the differenet methods.
     Draw a histogram of these values (Graph 2). Compute the value of the
     background-subtracted counts per pixel by summing the pixel counts in each aperture
     and subtracting the computed background value times
     the number of pixels within it from it. Draw a histogram of these values (Graph 3).
```

3)　　Repeat stage 2 without trying to avoid source regions, to produce graphs 4 and 5.

4)　　If required, compute a background image and store it in the file specified by the
parameter backgroundset. The background image is produced by placing a
box of width specified by the input parameter ''boxwidth'' on each pixel and using
the pixels within it to compute a value for the mean background.

10　Future developments

References