ommosaic

August 16, 2005

Abstract

Mosaics OM images (Derived from emosaic written by Richard West- SSC Leicester)

1 Instruments/Modes

Instrument	Mode
OM	IMAGING

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

This task accepts an arbitrary number of PPS OSW rotated image files and produces a mosaiced image. The task description is basically the same as that for emosaic, from which it has been derived, with the important difference that it does not need an exposure image. The mosaiced image is not intended to be used for photometry or astrometry. The task can process a mixture of both low and high-resolution images- in such case a warning is issued and the pixel-size of the mosiaced image is that of the low-resolution images. The program is mainly intended to combine the five separate images (The Rudi-five) that make up a full-frame image for a given filter. It can also be used to mosaic images from different filters, in which case a warning will be issued.

The output pixel size (in degrees) and world-coordinate system (WCS) reference coordinate for the output image are set to those of the first image with the largest pixel size. The task can read 16-bit, 32-bit or 64-bit input images, the output image is (currently) always of 32-bit real type.

4 Parameters

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

Parameter	Mand	Type	Default	Constraints

imagesets	yes	string	none	

List of input image datasets to be mosaiced

mosaicedset	yes	string	none	

Name of output mosaiced image dataset

exposure	no	real	1000	

Exposure time (secs) for mosaiced image

exposuremap	no	boolean	F	

is the exposure map required to be saved in the output file

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.

badDims *(fatal)*
Only 2-D images are supported

badType *(fatal)*
Image is of unsupported data type

You are combining identical images! *(warning)*

corrective action: file name

You are combining images from different filters! *(warning)*

(warning)

corrective action:
corrective action: List of filters foundYou are combining images from the same exposure!List of overlapping files found

6 Input Files

1. List of image files (image in primary header)

7 Output Files

1. Mosaiced image

8 Algorithm

```
Read parameters

Read each input image

If the input images contain a mixture of low and high resolution windows,
convert the high-resolution windows to low resolution ones.

Check that the filters for each file are identical- if not issue a warning.


Set output projection (reference coordinate, pixel size) from the
   first input image

Project corners of each input image into output frame to find the
   bounds of the output image

Calculate size of output image in pixels from bounds and pixel size

Create output image

Loop over input images
   Normalize each one by its exposure time.
   Loop over pixels in output image
      Project coordinate of output pixel into frame of input image
      Sample input pixel, add sample to output image
      Increment each weighting pixel by one if the corresponding input image
      pixel is to be coadded to the corresponding mosaiced image pixel.
      Increment the corresponding exposure image pixel by the exposure time
      for that image.

   End loop
End loop

Divide the mosaiced image by the exposure image.
Multiply the image by the exposure time (default 1000 secs).
Write output image
If required, write output exposure map image.
```

9 Comments

- none

9.1 Future developments

- none

References