eexpchipmap

August 16, 2005

Abstract

This task calculates an exposure map in CCD coordinates.

1 Instruments/Modes

Instrument	Mode
EPIC MOS:	IMAGING
EPIC PN:	IMAGING

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

eexpchipmap performs essentially the same task as **eexpmap**, with the following differences:

1. Both vignetted and non-vignetted exposure maps are calculated at the same time, thus saving a lot of time.
2. It is possible with **eexpchipmap** to write exposure maps in chip coordinates, which is desirable for correct production of background maps.
3. **eexpchipmap** accesses library routines to perform both its exposure calculations and its coordinate transforms, leading to more maintainable code and easier coodination with other tasks.
4. A cubic spline interpolation of the vignetting function is employed, which is both a much quicker algorithm and one which leads to a smoother result.

4 Parameters

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

Parameter	Mand	Type	Default	Constraints
eventset	yes	dataset		

Name of the event list dataset. This dataset contains information on the instrument, the WCS keywords of the X and Y columns, and exposure information contained in various extensions.

templateset	yes	dataset		

The name of a dataset with a template image in its primary extension should be supplied to this parameter. All keywords (except those pertaining to the DSS) in the primary HDU are copied over to the outputs. If `outputstyle`='sky', the template image is also necessary to specify the pixel dimensions and World Coordinates of the output.

withdss	no	boolean	yes	

Whether to make use of information about event selections stored in the Data SubSpace (DSS) of a FITS extension.

dssstyle	no	string	templateset	templateset—image—eventlist

This parameter is read if `withdss`='yes'. If `dssstyle`='templateset', the task attempts to read a DSS in the first block of the image template set `templateset`. If `dssstyle`='image', the task looks for the DSS in the first block of `dssset`. If `dssstyle`='eventlist', the task looks for the DSS in the `EVENTS` extension of `eventset`.

dssset	yes	dataset		

This parameter is read if `withdss`='yes' and `dssstyle`='image'. In this circumstance the user should supply to `dssset` the name of a dataset which has the desired DSS attached to its first extension.

outputstyle	no	string	sky	sky—raw

Whether to write the exposure map in chip (ie CCD, or 'raw') or sky coordinates. If the latter, a template image is necessary to supply the correct dimensions and WCS keywords. Note that a 'raw' output can always be converted to 'sky' after the fact by use of **eimchip2sky**. The end result should be identical.

flatexpimgset	no	dataset	flat_expmap.ds	

An exposure map without vignetting, in sky coordinates, is written to this file name if `outputstyle`='sky'.

vigexpimgset	no	dataset	vig_expmap.ds	

An exposure map with vignetting, in sky coordinates, is written to this file name if `outputstyle`='sky'.

attstyle	no	string	binnedset	binnedset—template

This parameter is read only if `outputstyle` is set to 'sky'. To convert from chip to sky coordinates it is necessary to know the spacecraft attitude. However the attitude is never completely stable and may vary significantly during an exposure. In this case the nett sky image must be a mosaic of components from different values of the attitude. A time series of attitude values (such as that made either by **attbin** or **evproject**) can be supplied to parameter `binnedattset` if `attstyle` is set to 'binnedset'. If it is judged that the attitude wander during the exposure did not exceed some small fraction of the image pixel dimensions, or if the binned attitude set is not available, then the user may choose to set `attstyle` to 'template' instead. In this case a single fixed value of attitude is read from *_PNT keywords in the template image header.

binnedattset	yes	dataset		

If `attstyle`='binnedset' the user should supply to the present parameter the name of a dataset which

contains a time series of the spacecraft attitude variation during the exposure.

flatexpcubeset	no	dataset	flat_expmapcube.ds	

An exposure cube without vignetting, in chip coordinates, is written to this file name if `outputstyle`='raw'.

vigexpcubeset	no	dataset	vig_expmapcube.ds	

An exposure cube with vignetting, in chip coordinates, is written to this file name if `outputstyle`='raw'.

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.

dummy *(error)*
dummy

6 Input Files

1. (Mandatory) A calibrated event list for the relevant EPIC camera, created by either **emchain** or **epchain**. The header must contain the following keywords:
   - `INSTRUME`
   - `DATE-OBS`
   - `RA_PNT`, `DEC_PNT` and `PA_PNT`.

   The event list must contain the binary table extension `EVENTS`, which must contain the following keywords:
   - WCS keywords for the `X` and `Y` columns.
   - If `withdss`='yes' and `dssstyle`='eventlist', the relevant DataSubSpace (DSS) keywords and extensions.

   The event list must contain at least one binary table extension `EXPOSUnn`, where `nn` indicates the ccd/node. Each of these extensions must contain the following keywords:
   - `WINDOWXO`
   - `WINDOWYO`
   - `WINDOWDX`
   - `WINDOWDY`
   - Any other keywords needed to set the state of the cal via a call to CAL_setState.
   - an intelligible subset of `TIMEZERI`, `TIMEZERF` and `TIMEZERO`

In addition, each EXPOSUnn extension should contain the necessary keywords to allow the state of the cal to be unambiguously set from it.

The EXPOSUnn extensions need not contain any columns.

If the event list contains BADPIXnn extensions (not mandatory), these must contain the following columns:

- RAWX - int16.
- RAWY - int16.
- YEXTENT - int16.

2. (Mandatory) A FITS dataset, which contains an image in its primary extension. The name of this dataset should be supplied to parameter templateset. The output images (flatexpimgset and vigexpimgset) are constructed so as to match templateset's pixel dimensions and World Coordinates.

3. (Only mandatory if withdss='yes' and dssstyle='image') A FITS dataset, the name of which should be supplied to parameter dssset. The primary extension of this dataset should contain Data SubSpace (DSS) information which describes any relevant event selections. Eg if you want to make an exposure map to match the event selections used in the construction of an image, you will probably want to supply this image to parameter dssset (provided that the image contains the selection specification in the form of a DSS).

4. (Only mandatory if outputstyle='sky' and attstyle='binnedset') **attbin** output file, containing a table ATT_BINS with columns TSTOP, RA, DEC, PA and IS_GOOD. The table should also contain a TIMEZERO keyword.

7 Output Files

- If outputstyle='sky':

  1. flatexpimgset: an 2-byte-real-valued exposure map (without vignetting) in sky coordinates is contained in the primary image extension.
  2. vigexpimgset: an 2-byte-real-valued exposure map (with vignetting) in sky coordinates is contained in the primary image extension.

  Each of these datasets contains the same keywords in the primary HDU as the template image, except for DSS-related keywords. Extra extensions in the template image are not propagated.

- If outputstyle='raw':

  1. flatexpcubeset: an exposure cube (without vignetting) is contained in the primary image extension.
  2. vigexpcubeset: an exposure cube (with vignetting) is contained in the primary image extension.

  The format of these cubes is as follows. The cube is a 3-dimensional image which is stored in the primary extension of the relevant dataset. The stored values are 2-byte reals. The order of the indices is such that, if the cube is accessed via the **dal** function real32Array3Data(), the 1st index of the returned pointer cube is the CCD number; the 2nd and 3rd indices respectively correspond to the RAWX and RAWY axes of each CCD.

  ***** plus gti extensions

  ***** plus fracexp kwds??

  Later these datasets will contain some informative keywords: at present they are a bit bare.

8 Algorithm

******

9 Comments

References