JEMX Representations of Shadowgrams

for

JEMX Technical Note 11

Issue 1, Rev. 0

Niels J. Westergaard

September 4, 2003

1 Introduction

This note describes recommended representations of JEM-X shadowgrams based on raw event positions or on corrected event positions in FITS images.

In C the counting starts with 0 (zero) and in the FITS standard counting begins with 1 (one) which might be the cause for some confusion. This difference is treated carefully in this note.

This version describes JMX2, the detector currently in use.

2 The JEM-X detector

The raw event positions are integers falling in the interval [0,255] for both x and y. The detector center defined as the point right below the central crossing of the collimator lamellae is very close to (rawx, rawy) = (128, 124).

The ISSW executable j_cor_position reads the IMOD tables "JMXi-CORX-MOD" and "JMXi-CORY-MOD" and returns corrected positions e.g. RAW(128, 124) becomes COR(0.00, 0.00)[mm].

It is noted that using 'fv' to display e.g. "JMXi-CORX-MOD" then using 'POW' gives the value 0.0 at (128, 124) i.e. when counting from 1 (one).

3 The FITS convention

The translation between physical coordinates and image pixels in FITS is described by three keywords, CRVALn, CRPIXn, and CDELTn, per dimension. In the standard version[1] a linear transformation is assumed:

$$x_c = \mathrm{CRVAL} + (i_F - \mathrm{CRPIX}) \times \mathrm{CDELT} \tag{1}$$

Here x_c is the center of the pixel. It is a common standard that the pixel index i_F refers to the center of the pixel. The subscript F indicates that the representation follows the FITS standard and starts counting with 1 (one).

CDELT is the size of a pixel. CRVAL is the reference value in physical coordinates and CRPIX is the reference value in the pixel coordinates. CRPIX is not necessarily an integer. One way to define these values is: The x-value of the center of the pixel, $\hat{i}_F$, where CRVAL falls is found as $\hat{x}_c$, then

$$\mathrm{CRPIX} = \hat{i}_F + (\mathrm{CRVAL} - \hat{x}_c)/\mathrm{CDELT} \tag{2}$$

The relation to find a pixel index when x is given is

$$i_F = \mathrm{floor}((x - \mathrm{CRVAL})/\mathrm{CDELT} + \mathrm{CRPIX} + 0.5) \tag{3}$$

where "floor" is a function that returns the largest integer less than or equal to the argument.

[1]Non-linear transformations must be explained using "COMMENT" lines in the FITS header.

4 Shadowgram in raw positions

The following standard is proposed:

A one-dimensional array is used and the index calculated in C as

$$\text{index} = 256 \times y_{\text{raw}} + x_{\text{raw}} \tag{4}$$

and then written to disk using a DAL routine.

5 Shadowgram in corrected positions

The most straightforward way is to use pixels of a size of 1 mm. This matches the achievable resolution and a 256×256 array can be used.

The following standard is proposed:

The center of the detector is in the middle of the array i.e. between pixels 128 and 129 (start count with 1).

$$\begin{aligned} \text{CRVALn} &= 0.0 \\ \text{CRPIXn} &= 128.5 \\ \text{CDELTn} &= 1.00 \end{aligned}$$

following section 3.

Equation 3 can be used to get the pixel numbers (i_F,j_F) and

$$\text{index} = 256 \times (j_F - 1) + i_F - 1 \tag{5}$$

is then the index number for e.g. a background shadowgram.

The disadvantage by using this is that some pixels will not be populated due to the irregular position corrections. For nicer presentation this can be overcome by smearing out the event positions in a random way.

6 Programming example using DAL

This example assumes that the `ptr` is a pointer to a DAL data structure and that `image` is an array built as explained above. Note, that the DAL assumes with FITS that counting starts with 1 (one).

```
long startValues[2] = {1,1};
long endValues[2] = {256,256};
long axes[2] = {256,256};
int numAxes = 2;
status = DALarrayModStruct( ptr, DALtype, numAxes, axes, status);
status = DALarrayPutSection( ptr, numAxes,
      startValues, endValues, DALtype, image, status);
```

In the case of a raw shadowgram no extra keywords are required, but when using corrected positions the CRVAL, CRPIX, CDELT keywords must be written specifically by `DALattributePutReal` calls