The XIMAGE User's Guide

Version 2.53

P. Giommi[1], L. Angelini[2] and N. White[2]

1. ASI, SAX Scientific Data Center, Via Corcolle, Rome, Italy.

2. High Energy Astrophysics Science Archive Research Center, Code 668,
Goddard Space Flight Center, NASA, Greenbelt,
Maryland, 20771, USA

Contents

Chapter 1

Introduction

1.1 Overview

XIMAGE is a multi-mission X-ray image display and analysis program. It is designed to be instrument independent and to support the analysis of data from any X-ray imaging detector provided that the proper calibration files are available. Currently XIMAGE supports the detailed analysis of EXOSAT CMA, Einstein HRI and IPC, ROSAT PSPC and HRI, and ASCA GIS and SIS data. It also supports some basic analysis of optical, infrared and radio images. XIMAGE has a built in data simulation program that can simulate images of current and future X-ray missions (e.g. SAX, AXAF and XMM).

The following functions are provided:

- reading in of images and event files
- image rebining, smoothing, and display
- source detect, and statistical analysis
- source removal
- correction for vignetting, exposure, background and point spread function
- image mosaics
- contour plots and overlays
- sky grids and pixel-coordinate conversions
- changing equinox
- x/y image slices
- generation of a point spread function
- circular, annular and box region selection
- an interface to saoimage
- extraction of spectra and lightcurves from event data
- image simulation

The display and graphic capabilities of XIMAGE are based on the `PGPLOT` graphic package, which supports most terminals and workstations. The `saoimage` package can also be spawned to display images and select regions.

While XIMAGE is a multimission package, it must first "know" about the calibration information associated with a mission in order to be able to make a detailed image analysis. Some XIMAGE functions are mission independent (e.g. display), but others are not (e.g. source detect). XIMAGE will read in images from an unknown mission, but the user should beware trying to make a detailed analysis. A warning will be issued if the mission is unknown. A list of known missions can be obtained by giving the command `show detectors`. Adding new missions usually requires either adding new files and/or adding new calls and relinking XIMAGE.

1.2 History

XIMAGE is based on original routines developed as part of the EXOSAT interactive image analysis system. These routines have been restructured and assembled as part of a multimission command driven interactive software package running as part of the XANADU infrastructure.

1.3 Calibration Information

To perform image analysis requires the following calibration items to be known:

- the vignetting function - the radial dependence of the telescope effective area;
- the point spread function (PSF) - the blurring of the image caused by its intrinsic angular resolution;
- the exposure map - the number of seconds each pixel in the image was exposed; and
- a background map - the particle background counting rate in each pixel.

The first two are critical for XIMAGE to produce any useful results. The vignetting can be provided as an image map, or as a function. In the latter case the XIMAGE code must be relinked. The PSF can be derived from within XIMAGE, and read in as a file. The exposure map is optional, and if it is not available then it is assumed the image exposure is uniform. This is usually acceptable for detectors without any obstructing structures (e.g. the EXOSAT CMA), but will lead to errors if there are window supports (e.g. the ROSAT PSPC). The background map is also optional, and is only necessary in the case where there is significant variation in the particle background across the detector.

The vignetting function and the PSF can be energy dependent in many cases (e.g. the ROSAT PSPC), and care must be taken to ensure that they match the energy band of the image.

1.4 Caveat Emptor

XIMAGE is written by astronomers, for astronomers, in fortran 77. This has the advantage that its requirements are driven by real needs, and so it tends to do things in the most natural way and usually very efficiently. While every effort has been made to get things right, the user should always make an attempt to "understand" what ximage is doing to produce the given results.

1.5 Installation

The XIMAGE package can be obtained for VAX/VMS, Dec-ultrix, Dec-OSF and Sun-os. Compressed tar files and VMS backup files can be found in the anonymous FTP on `legacy.gsfc.nasa.gov` under the `/software/xanadu` directory. VMS copies can be obtained from `lheavx::xandis:[savesets]` (copy the `readme.` file first).

1.6 Reporting problems

Any problems in obtaining or installing XIMAGE should be reported to `xanprob@legacy.gsfc.nasa.gov` or `lheavx::oday`. Any questions or problems with XIMAGE or requests to add additional instruments should be directed to `angelini@lheavx.gsfc.nasa.gov`.

Chapter 2

Getting Started

XIMAGE is run directly from the operating system prompt by typing `ximage`. The user will receive the prompt `[1]XIMAGE>`. This is the main command line of XIMAGE. Typing `?` will produce a summary of the available commands. Typing `help` followed by the command name (e.g. `help display`) will provide a description of each command.

2.1 Command

2.1.1 Syntax

XIMAGE is a command-driven program. Each command has associated with it qualifiers and positional parameters which define parameters to be passed to each XIMAGE command. Qualifiers and positional parameters be of three different types, numerical, character and logical. Qualifiers are preceded by a slash `/`. Numerical and character qualifiers must be used as follows `command/qualifier=` the parameter value or the appropriate alphanumeric string must be given after the symbol "=". Logical qualifiers can be used simply by specifying their name (i.e. `command/logical`). Qualifier names can be abbreviated provided that the abbreviation is unique. To get a list of the available qualifiers for each command type `command/?`. Positional parameters follow the qualifiers and are typically used to specify a required parameter e.g. `command pos1`. Up to ten commands can be typed on the same line using the symbol `;` as a separator. The commands are executed in sequence from left to right.

Commands can be given at the `XIMAGE>` prompt or, equivalently, as a command string when XIMAGE is started. The following two examples will produce the same result:

```
$ XIMAGE
[1]XIMAGE>read/fits test.fits
[2]XIMAGE>disp
[3]XIMAGE>exit

$ XIMAGE read/fits test.fits; disp; exit'
```

In both cases the user asks to run XIMAGE, read the image test.fits, display it and then return to the operating system.

2.1.2 Recalling

The command `recall` is used to list previous commands. Any command can be reexecuted by typing `recall n` where n is the command number given in the listing. The `alias` command is used to rename long command strings to a simple name. A log file of the session can be created using `log file`. This will create a text file called file.log.

2.1.3 Scripts

Scripts (Macros) can be created in a text file with the extension `.xco`. They are run by giving the file name preceded by an `@` symbol at either the XIMAGE prompt, or at run time e.g. `ximage @my_analysis [P1] [P2]`. Variables can be passed into the script by using `%N%` in the script, where `N` is the number of the parameter on the command line. E.g. the `my_analysis` script could be

```
read/fits %1%
disp
read/fits %2%
disp
```

Where P1 and P2 are the filenames. If `P1` and `P2` are not given on the command line, then the user will be prompted for them. To prompt for parameters from a script use the `@` symbol.

2.1.4 Spawning

The operating system can be accessed from the XIMAGE command line by preceding the command with a `$` symbol e.g. `$ls`. Typing a `$` symbol on its own will spawn to the operating system command line, without interupting the current ximage session. Type `exit` to return to XIMAGE.

2.2 Data

2.2.1 Images

An image must be read into XIMAGE using either `read_image` or `simulate`. The former is used if the image is taken from a specifed file that can be:

- a FITS image (`read/fits filename`),
- a FITS photon event list (`read/rfits filename`),
- an EXOSAT format image (`read/exo filename`),
- an EXOSAT compressed image (`read/comp filename`),

For FITS image files the image size is dictated by the size of the image. If the image is bigger than the maximum image, then the size will be set equal to the maximum. For other image formats the image will be by default read in at the original resolution into a `256x256` array. These defaults can be changed using the `/size=n` and `/rebin=m` qualifiers. The maximum size, n, is 1024. The rebinning factor, m, can be any value consistent with the size of the original image. After rebinning XIMAGE will continue to work and display in terms of the original unbinned pixel convention. This avoids any problems with interacting with calibration information, or specifying selection regions. If the file is an event file then the image can be accumulated in a particular energy band using the

`/min_pha=n` and `/max_pha=m` qualifiers. To simulate an image, type `simulate` and then give mission and instrument to be simulated.

If a photon event file is available then this can also be read in to create an image using the `read/rfits` command. Currently only the ROSAT FITS format is supported. Various default image sizes have been established for each mission, that can be changed using the `/size` and `/rebin` qualifiers.

2.2.2 Displaying

Once an image has been loaded it is displayed using the `display` command, but first the plot device must be set using the `cpd` command. XIMAGE supports many different plot devices, though the best performace is obtained using x windows. On VMS use `cpd /x11` and on unix `cpd /xw` to set the plot device to X.

The color table is selected automatically. It can be changed using the `cct` command. A number of alternate color tables are available and these can be stepped through using `cct/loop`. For devices with limited colors or without image display capability a contour plot can be made with the `contour` command. The image should be smoothed beforehand using the `smooth` command.

`saoimage` can also be used to display an image. This is useful for panning and zooming of the image. The display is handled by writing an intermediate file. SAOIMAGE is run simply by typing `saoimage` at the ximage command line. It is not possible, however, to overlay grids and other items on the image.

2.2.3 Split screen

Images can be placed side by side for comparison purposes using `disp/left` and `disp/right/overlay`. The currently active window is the last one displayed.

```
read/fits im1.fits          ! read the 1st image
display/left                ! display the 1st image
grid                        ! overlay the sky grid
read/fits im2.fits          ! read the 2nd image
disp/right/overlay          ! display the two side by side
grid                        ! overlay the sky grid
```

2.2.4 Viewports

The exact position and size of the image on the screen can be controlled using the `/v` qualifiers with the `disp` command. This is best done as part of a script `.xco` file, since it usually takes several attempts to get things right. Here is an example of plotting six images in four adjacent panels, with each label specified individually:

```
read/fits/size=64/xpix=440/ypix=410 s0_noflare2.img
title " "
title/low " "
smooth/sig=0.8
DISP/V1=0.1/V2=0.4/V3=0.1/V4=0.4/no/log
info/xs=15/scale=0.75 "Quiescent (smoothed)"

read/fits/size=64/xpix=440/ypix=410 s0_yygem_flare.img
title " "
title/low "25/26 Oct 1993 ASCA SIS0"
```

```
smooth/sig=0.8
DISP/V1=0.317/V2=0.617/V3=0.4/V4=0.7/overlay/no/log/scale=0.9
info/xs=15/scale=0.75  "YY Gem flare (smoothed)"

read/fits/size=64/xpix=440/ypix=410 s0_castor_flare.img
title " "
title/low " YY Gem and Castor "
smooth/sig=0.8
disp/v1=0.1/v2=0.4/v3=0.4/v4=0.7/overlay/scale=0.9/no/log
info/xs=15/scale=0.75  "Castor Flare (smoothed)"

read/fits/size=64/xpix=440/ypix=410 s0_noflare2.img
title " "
title/low " "
disp/v1=0.317/v2=0.617/v3=0.1/v4=0.4/overlay/no/log
info/xs=15/scale=0.75 "Quiescent"
```

2.2.5 Exposure Maps

Current and saved arrays are available to read in an exposure map. An exposure map contains as each element the total number of seconds that each pixel was exposed. Once an exposure map has been read in, then it is assumed that the vignetting correction has been applied to the map. If it has not, then the `vignetting` command can be used to apply the correction (for a known instrument).

The `read/exposure` command will read in an exposure map that is in FITS format. The exposure map must be read in after the corresponding image has been read, and it must have the same binning and size as that image. The map can be displayed using the `disp/exp` command. The exposure corrected image is displayed using `disp/correct`.

2.2.6 Background Maps

These are currently only supported for the EXOSAT CMA. In a future release it will be possible to read in a FITS image containing the background map.

2.2.7 Saved Images

XIMAGE keeps two images in its volatile memory. These are called the "current image" and the "saved image". The current image can be saved using command `save_image` and the saved image can be copied onto the current image with command `restore_image`. The first image loaded by XIMAGE is automatically saved.

2.2.8 Writing

The `write` command allows images to be written out in a number of different formats including FITS. The current exposure map can also be written using the `/exposure` qualifier. The currently displayed image is written using the `/disp` qualifier.

2.2.9 Spectra and Lightcurves

The `extract` command is used to obtain a spectrum and/or a lightcurve for the selected region (the default takes the whole image). The spectra are in the ASCII dump of the XSPEC `sf` format. The

XSPEC `mkpha` program is used to convert it to XSPEC `sf` format. The lightcurve is an ASCII file in QDP format. It can be plotted using either QDP, or XRONOS. In the near future this will be changed so that the output is FITS files.

2.3 Coordinates and Overlays

A sky coordinate grid can be overlayed on the image using `grid`, and a slice in the x and y planes made using the `slice` command. The `skymap` command is used to read in a file of star positions and overlay them on the current image. Circular and box regions can be defined and written to an SAOIMAGE type region file using the `circle` and `box` commands. The HEASARC database can be directly queried using the `skymap/db=`, where `/db=` is used to specify the name of a HEASARC database.

2.3.1 Grid

To draw a skygrid type `grid`. The default grid spacing is usually optimal, but can be changed by giving the degree steps e.g. `grid 0.5 0.5`. If the grid is not optimal use `display` to redisplay, and then try a new grid. The grid will be drawn using the current equinox. The `/gal` qualifier will draw the grid in galactic coordinates.

2.3.2 Pixel Coordinate Conversions

There are two commands to convert pixel coordinates to sky positions and vice-a-versa. These are `pix[el_to_ra_dec]` and `ra[_dec_to_pixel]`. They can, as with all ximage commands, be abbreviated to make them unique e.g. `pix` and `ra`. Use `pix/cur` to input the pixel coordinates using the cursor.

2.3.3 Equinox Year

The default equinox used by ximage will be that of the original image. All coordinates will use this equinox, unless it is changed using the `cey year` command, where `year` is the equinox e.g. 1950. Once the equinox is changed all coordinates will be given referenced to that year. The equinox year can be changed many time during a session.

2.3.4 Slice

To draw a slice along the x or y axis then use `slice/x` or `slice/y`. The cursor is used to select the range of y or x to sum the image over to make the slice (the opposite axis to which the slice is to be made). For am x slice then click the left button to choose two points along the y axis, then click the right hand button to draw the slice. The middle button can be used to undo a selection before the roght button is clicked. The `/log` qualifier will draw the slice on a log scale.

2.3.5 Skymap

The skymap command reads in a list of star positions and overlays them on the current image. The default will read an ascii file with the first line containing the equinox year (e.g. 1950) and on each following line the RA and DEC in decimal degrees. The `/label` qualifier will also overlay the names of the objects and the `/class` will give the classification.

A powerful feature in ximage 2.50 and greater is to directly make overlays of catalogs in the HEASARC database. This is done as follows:

```
[3]XIMAGE> skymap/db=rc3/label
```

This will open a connection to the HEASARC database system at GSFC and will download all objects within the image that are found in the `rc3` database into a local file. This is then automatically overlayed on the image. The `/label` qualifier will cause the objects to be writen. The `/class` qualifier will write the object class as well, if it is known. Useful master databases such as OPTICAL, XRAY, RADIO and ROSID give an overview of the contents of the many individual databases. See the HEASARC online services users guide for further details (to obtain a copy send an e-mail to `request@legacy.gsfc.nasa.gov`).

The SIMBAD source lists provided in FITS format by the US ROSAT data center can be read in using the `/RFITS_SIMBAD` qualifier.

2.3.6 Skyview

It is possible to access the HEASARC skyview facility directly from XIMAGE using the `skyview` command. This will take the coordinates and image size from the current image, and request the image from the specified survey. This is useful for making overlays and identifying X-ray sources. A limited number of surveys are hardwired into XIMAGE and can be accessed using qualifiers (use `skyview/?` to see the qualifiers). Other surveys may have been added to skyview and the full list can be accessed using `skyview/list` and the chosen survey accessed by giving the full name as a parameter e.g. `skyview digitized sky survey`. After the image is retrieved it must be read back in, and displayed in the usual way.

Chapter 3

Source analysis

There are several XIMAGE commands to make a detailed analysis of the image. These include:

- `SOSTA` - calculate the source counting rates and statistics correcting for vignetting, point spread function and exposure,
- `BACKGROUND` - estimate the background.
- `DETECT` - a sliding cell source detection algorithm, using the PSF and vignetting to estimate source significance and statistics.
- `REMOVE` - remove sources from an image.
- `PSF` - generate a point spread function, and compare it to the predicted psf.

These commands are very mission dependent and require the available calibration information.

3.1 Sosta

The SOSTA command can be used to determine the number of counts within a given box (XIMAGE uses boxes, rather than circles, because its quicker, and avoids problems with deciding what to do with pixels that are partly in or outside a given region). SOSTA counts the number of events within a specified box, corrects those counts for vignetting, exposure and the point spread function to give the source intensity and its statistical significance. If the significance is less than `1.e-3` an upper limit is automatically calculated. The simplest way to use SOSTA is with the cursor using the `/CURSOR` qualifier. In its default mode SOSTA will use a box that corresponds to 66% of the encircled energy function (EEF), and will use a background from a region surrounding the source, outside the 98% EEF boundary.

The background can be estimated using several different methods, specified by the user. These are:

- The background is estimated from a region within two boxes centred on the source using the option `/INNER_RADIUS` and `/OUTER_RADIUS` to specify the half-sizes of the two boxes. This is the default.
- The background is taken from any specified part of the image using `/BOX_BACKGROUND`;
- The background intensity can be given using the qualifier `/BACKGROUND_LEVEL`. In this case the background statistical error is assumed to be zero.

The uncertainty in the count rate returned by SOSTA is purely statistical i.e. does not include systematic errors. SOSTA will estimate the optimum box size to maximize the signal to noise ratio. This is given at the end of each run. By using the option `/OPTIMIZE` then SOSTA will go around a second time, and use the optimum box size to calculate the optimum source statistics.

3.2 Background

The BACKGROUND command estimates the background in the image. The method used divides the image into equal boxes to estimate the local background intensity. The event distribution in each box is checked for spatial uniformity and consistency with the expected statistical distribution (Poisson or Gaussian, depending on image type). If inconsistency between the observed and expected distribution is found the box is rejected. The distribution of the average background values in all boxes is then compared to a Gaussian distribution with mean equal to the mean of all the background values found. All boxes where the background level is more than three standard deviations away from the mean are rejected. If more than 70 % of the boxes are rejected the background routine stops and an error message is generated. The value of the background intensity returned is the average of the values measured in all accepted boxes. A P(D) can be obtained using the `background/npd=n/box=m/plot`. The output is written to the file `pd.qdp`.

3.3 Detect

The DETECT command locates point sources in the current image by means of a sliding-cell method. The average background intensity is estimated using the method described previously. The position and intensity of each detected source are calculated in a box whose size maximizes the signal-to-noise ratio. If the source is not pointlike the estimated count rate is in general inaccurate and likely to be under-estimated. A good estimate of the intensity of extended sources can be obtained with command `SOSTA`.

Corrections for telemetry and instrumental dead times, for vignetting effects and for the fraction of the source counts that fall outside the box where the net counts are estimated are applied. Count rate errors include both statistical and systematic uncertainties added quadratically. To minimize the number of spurious sources detected the threshold used by DETECT is somewhat conservative. Consequently, some sources with intensity just above the image background can be missed. In order to allow DETECT to obtain a sufficiently good estimate of the background only images of size 128x128 pixels or larger should be used. Maximum accuracy is obtained running DETECT on full resolution images.

For crowded fields, or for fields with bright sources the qualifier `/bright`. This will make a weighted mean of each excess, and optimizing the psf. Consequently it can take much longer to run.

For busy images, e.g. ROSAT PSPC, it may be necessary to "tweak" the detection parameters to avoid false detections. The `/thr_scal` and `/source_box` qualifiers are particularly useful. The `/thr_scal` scales the threshold for excesses to be detected. This can be increased by modest amounts for ROSAT PSPC deep fields e.g. 1.2 or 1.3. The `/source_box` qualifier determines the size of the box used, in detector coordinates. For the ROSAT PSPC the default is 80, and it can be increased to 160 to reduce confusion.

3.4 Remove

Sources, hot spots or other features can be removed from an image using `REMOVE`. This can be done by:

- using the cursor to mark one or more box regions (/CURSOR)
- entering the pixels coordinates of each box (/X_MIN,/Y_MIN, etc)
- using the sources found in the previous run of DETECT.

The latter mode is useful for checking how good the detect run was. For a crowded region the sequence DETECT;REMOVE/DETECT;DETECT;REMOVE/DETECT can be used to clean the image of all sources.

3.4.1 Regions

Region selection is made using box and circle. The regions are selected using the cursor, and are by default inclusive. The /exclude qualifier will create an exclusive region. These write out a region descriptor file that has a format identical to the region descriptor file used by SAOIMAGE (which can also be used to write the region file). The region file can be used to select a part of the image for further analysis. Multiple regions can be built up, to e.g. create an annulus, or exclude part of the image. Each call to these two commands will add to the existing region file. The /new qualifier is used to start a new region selection. The region file is given by default the name of the image, with the extension .reg. Otherwise it can be specified using /regfile=.

3.5 Extract

The extract command can be used to generate spectra and lightcurves from an input event file. The default is to use the region file generated by the box and circle commands. A new region file name is input using /region=file. Output spectra(/phafile=) and lightcurves(/qdpfile=) are in ASCII format. The former is the old style XSPEC RDPHA output and you must use MKPHA to turn make a file that can be read by XSPEC. The light curve is a QDP file that can be read into xronos. It is in general better to use XSELECT to make lightcurves and spectra. The extract command is most useful for generating color-color diagrams for all the sources detected in an image generated from an events file, or for making an energy dependent PSF for a source, again if the events file is present.

3.6 Color Color Diagrams

If an image has been made from an event file, and one or more sources have been detected when running DETECT, then the command extract/detect/plot will make a color-color diagram. This is useful for identifying very soft or very hard sources in the field. The default channel boundaries can be changed using the /C1, /C2, /C3, /C4 qualifiers. An output QDP format filename containing the results is written with a default of the eventfile.col, where eventfile is the input event filename. A new name is given with the qualifier filecolor=. Note that when /detect is given, the extract command will not pick up the default region file.

3.7 PSF

Calculates differential and integral Point Spread Function, PSF. This accumulates all the counts within annuli of increasing radii. Each annulus increases by one pixel. Rebin the image to change the step size. The results are written to a QDP format file, and are also plotted using QDP. The differential PSF is given as a surface brightness (counts/s/sq arc min) and the integral as the

Encircled Energy Function, EEF. The expected values for a pointlike source are also written to the QDP file and displayed. An accurate estimate of the image background is required, and is obtained using the `background` command. The cursor can be used to specify the source centroid, and the outer ring `psf/cursor`. The `psf` command can be used to extract the surface brightness profile of extended sources, for further analysis outside XIMAGE.

In most cases the PSF is independent of energy, but for the ROSAT PSPC this is not the case. In this case `extract/psf_pha` is used to accumulate a spectrum for the source (use `circle` to define the region). This spectrum is written to a file called `rosat_pspc.psf`, which is then used by `psf` to calculate the PSF weighted for spectrum. If no spectrum is accumulated, then a PSF at 1 keV is used. To speed up `detect` and `sosta` the 1keV average is used.

For a detector that is unknown to XIMAGE, a PSF can be created, using `psf/calfile`. This writes a PSF to the local directory with the name `psf.dat`. This file is then used by `detect` and `sosta` for subsequent analysis.

The following is an example of how to make a PSF for the ROSAT PSPC. This procedure is more complex because the PSF is energy dependent, so a spectrum must first be accumulated.

For ROSAT PSPC only:

```
read/rfits/xpix=5000/ypix=5000 rp500012.fits  ! make an image from the
                                              ! events file.
disp                                          ! display it
circle                                        ! define a source region
extract/psf_pha                               ! extract a spectrum
```

For all instruments (for non-PSPC first read in the image):

```
back/opt                                      ! optimum background
psf/cur/back=1.0e-03                          ! calculate the psf
```

Chapter 4

Mosaics

XIMAGE can be used to take several images from a part of the sky and make a mosaic.

4.1 Without Exposure Maps

The sequence of commands to do this are:

```
read/fits/size=800 file1.fits    ! read the first image into a 800x800 array
save_image                       ! save the image
read/fits/size=800 file2.fits    ! read the second image
sum_image                        ! sum the 1st and 2nd images
save_image                       ! save the summed image
read/fits/size=800 file3.fits    ! read the third image
sum_image                        ! sum the 3rd with the saved image
save_image                       ! save the mosaic
disp                             ! display the mosiaced image
grid 0.5 0.5                     ! draw a skygrid in 0.5 degree steps
disp/exp                         ! display the exposure map
disp/cor                         ! display the exposure corrected image
write/fits mosaic.fits           ! write the mosaic to a fits file
```

To center the image the X and Y of the pixels of the first image where the image is to be centered should be given. Its usually best to start with the central image, if possible.

The exposure for each image is kept track of within XIMAGE, assuming the entire region has been equally exposed. This may not be accurate, especially for those cases where the active area of the image does not match that of the entire image. Where ever possible an exposure map must be read in.

DETECT and SOSTA will not give the correct results if the summed images come from different pointing directions. They will be unable to make any correction for off-axis vignetting and psf changes, since they reference to the detector coordinates of the last image that was read in.

4.2 Overlays

To overlay images from different missions, e.g. ROSAT ontop of ASCA, the images must be rebinned as they are read, so that the scale size of each pixel is the same number of arc sec per pixel. To do a contour overlay then do the following sequence:

```
read/fits/size=800 inst1.fits   ! read the 1st instrument image
save_image                      ! save the image
display                         ! display the 1st image
read/fits/rebin=30 inst2.fits   ! read the 2nd instrument image, rebinning to
                                ! get the same plate scale
smooth                          ! smooth the 2nd image
cont/overlay                    ! make a contour plot and overlay it
                                ! onthe current image.
grid 0.5 0.5                    ! draw a skygrid
```

4.3 With Exposure Maps

When making a mosaic, it is best to read in an exposure map for each image. The exposure map must be read in after the image. The following example repeats the mosaic desribed above, but includes an exposure map.

```
read/fits/size=800 file1.fits    ! read the first image into a 800x800 array
read/fits/exp/size=800 exp1.fits ! read the first exposure map (optional)
save_image                       ! save the exposure map
read/fits file2.fits             ! read the second image
read/fits/exp exp2.fits          ! read the secon exposure map
sum_image                        ! sum the 1st and 2nd images
save_image                       ! save the summed image
read/fits file3.fits             ! read the third image
read/fits/exp exp3.fits          !
sum_image                        ! sum the 3rd with the saved image
save_image                       ! save the mosaic
disp                             ! display the mosiaced image
grid 0.5 0.5                     ! draw a skygrid in 0.5 degree steps
disp/exp                         ! display the exposure map
disp/cor                         ! display the exposure corrected image
write/fits mosaic.fits           ! write the mosaic to a fits file
write/fits/exp mosaic_exp.fits   ! write the exposure
```

Chapter 5

Commands

5.1 alias - rename a command string

`alias/quals [name] [command_string]`

Rename a command string to an alias name. `name` specifies the new name, command_string is that to be renamed. The command string is exactly as it would be given on the command line. `alias/show` will list all of the aliases.

If the command_string contains blanks, it must be enclosed in double quotes.

alias/previous [new_name]

Define the previously given command as an alias with the name new_name.

alias/delete

Delete a previously given alias name.

alias/show string

List the currently defined alias names. The default will give only user-defined names. The qualifier `/system`, when used in combination with `/show`, will list the system-wide aliases. The default lists both the system and user aliases. `string` specifies part of the string to be listed. Wild cards are allowed, e.g., `alias/show c*` will list all aliases beginning with `c`.

alias/system

Used to specify that an alias name is to be system-wide (system managers only). Also, when used in combination with the `/show` qualifier, it will list the system-wide qualifiers (all users).

Examples:

```
alias plot "disp"       ! define the command disp to be plot
```

```
alias/prev my_data      ! redefine the previous command
```

5.2 background - calculate average image background.

`background[/qualifiers]`

The method used consists in dividing the image in n equal small square boxes where the local background intensity is estimated. The event distribution in each box is checked for spatial uniformity and consistency with the expected statistical distribution (Poisson or Gauss, depending on image type). If inconsistency between the observed and expected distribution is found the box is rejected. The distribution of the average background values in all boxes is then compared to a Gaussian distribution with mean equal to the mean of all the background values found. All boxes where the background level is more than three standard deviations away from the mean are rejected. If more than 70 % of the boxes are rejected the background routine stops and an error message is generated. The value of the background intensity returned is the average of the values measured in all accepted boxes.

background/draw_rej_back_boxes

display all background boxes which have NOT been used for the estimation of the background

background/box_size=n

defines the size (in image pixels) of the boxes where the image background is estimated. Allowed background box sizes are powers of two, and values of 16, 32, 64, and 128 are recommended.

background/npd=n

A P(D) distribution is obtained using the `/npd=n` option. `n` is the number of count rate intervals into which the distribution is divided. Use `/plot_pd` to get a plot. The results are written to a file `pd.qdp`, or this a file specified by `/file_pd=`.

background/optimize

Go through a range of background boxes and determine the optimum box size to give a minimum backgorund counting rate.

background/plot_pd

Plot the P(D) distribution written using `background/npd=n`.

background/file_pd

Write the P(D) distribution to a file, otherwise the default `pd.qdp` is used.

Examples:

```
background/opt              ! calculate the optimum background

background/box=32           ! calculate the average image background
                            ! using square boxes of size=32 image pixels.

background/draw             ! use the default box size and draw the
                            ! boxes that have been rejected

background/npd=20/plot      ! do a P(D) distribution into 20 count rate
                            ! ct bins and plot the result
```

5.3 box - select a box region

`box[/qualifiers]`

A box shaped region is selected using the cursor. The results of the selection are written to a region file. This file has by default the same name as the image file, with the extension `.reg`.

A region filename can be specified using the `/file=` qualifier. The regions are by default inclusive. The `/exclusive` qualifier will create an exclusive region. Each new region selection is added to the region file, unless the `/new` qualifier is given. This applies to all region selections, including `circle`. The coordinates written to the file are in the original unbinned pixels, even if the image has been rebinned.

box/excluded

The define region will be excluded.

box/regionfile=filename

Specify the name of the region file. Otherwise the default will be the `imagename.reg`

box/new

Start a new region definition. If `/new` is not specified, then regions will be added to the existing file.

box/displayonly

Draw the contents of a region file on the current image

Examples:

```
box                           ! select a box region using default region
                              ! filename

box/display/region=hello.reg ! draw the contents of the region file hello.reg
```

5.4 cct - change colour table

```
cct [table name]
```

Reads a predefined table of colours and applies it to the previously displayed image. This command must be used after the command DISPLAY since DISPLAY always loads a default colour table. A question mark in place of the table name gives the list of all available tables.

cct/loop

Loop through each of the available color tables

cct/no_break

Go through each color table without stopping.

cct/user_col

Allow to specify user color table from a file using 'cct/user filename'.The colour table is an ascii file with the columns (R, B, G).The values for each colour should range from 0 to 1.

Examples:

```
cct exo15           ! load the colour table exo15

cct ?               ! lists all available colour tables

cct/loop            ! loop through all the available color tables

cct/loop/no         ! loop through all the color tables without a break
```

5.5 circle - select a circular region

```
circle[/qualifiers]
```

A circular region is selected using the cursor. The results of the selection are written to a region file. This file has by default the same name as the image file, with the extension `.reg`.

A region filename can be specified using the `/file=` qualifier. The regions are by default inclusive. The `/exclusive` qualifier will create an exclusive region. Each new region selection is added to the region file, unless the `/new` qualifier is given. This applies to all region selections, including `box`. The coordinates written to the file are in the original unbinned pixels, even if the image has been rebinned.

circle/excluded

The define region will be excluded.

circle/regionfile=filename

Specify the name of the region file. Otherwise the default will be the `imagename.reg`

circle/new

Start a new region definition. If `/new` is not specified, then regions will be added to the existing file.

circle/displayonly

Draw the contents of a region file on the current image

Examples:

```
circle                        ! make a circle region selection
circle/excluded               ! make an excluded circle region selection
circle/filename=fred.reg      ! write the cirle region select to a file
circle/file=fred.reg/disp     ! display the region file fred.reg
```

5.6 cl - change the number of colour levels

```
cl[/qualifiers]
```

The maximum number of levels that can be used depends on the display device used. The default number of levels is set equal to the maximum allowed by the plot device currently defined. If the graphic device used does not allow this, the number of levels is automatically reset to the maximum value allowed.

cl/number_of_levels=nn

defines `nn` as the number of colour levels to be used for the image display.

Example:

```
cl/n=20              ! Sets the number of colour levels to 20.
```

5.7 close_pg_window - close the pgplot graphics window.

```
close_pg_window
```

Examples:

```
close_pg             ! close the current window
```

5.8 contour - display iso-intensity contour levels

```
contour[/qualifiers]
```

A contour plot is drawn. It is recommended that the image be rebinned and smoothed first. The user can specify the number of levels to use, with the qualifier `/NO_OF_CONT_LEVELS`, a default number of 10 levels is used otherwise. The option `/FIRST_CONTOUR_DRAWN` defines the first (lowest) level to be displayed. By default the lowest 5 levels are not displayed.

contour/linear

Conturing image with linear scale histogram

contour/log

Contouring image with logarithmic histogram

contour/histo

Conturing image using image histogram to define levels. This is the default option for command CONTOUR

countor/div_map

Contours calculated on the map ratio obtained dividing the current image by the saved image.

countor/div_saved

Contours calculated on the map ratio obtained dividing the saved image by the current image.

contour/sub_map

Contours calculated on the image obtained from the current minus the saved image. If the result is a negative number, it will reset it to zero.

countor/exposure

Exposure map contours are plotted.

countor/saved_image

Saved image contours are plotted.

countor/correct_exposure

Contours calculated on the current map corrected by the exposure map.

contour/no_frame

The contours are plotted without frame. This qualifier is usefull when the contours plot are overlayed (see /overlay) on a pre-display image.

contour/no_of_cont_levels=nn

This defines the number of contour levels, nn, to be used.

contour/first_contour_drawn=nn

This defines the number of the lower contour used for display.

contour/color=m

The color number, m, to be used for the contours range of allowed values is 1 to the max number of color levels supported by the display used.

contour/overlay

The contour plot is overlayed to the pre-displayed image. The default is that the contour plot is drawn on a new Pgplot page.

contour/left

The qualifier left allows the display to appear on the lefthand side of the screen. Any subsequent interacting command with the Pgplot screen will act on the lefthand side. If a contour plot already is displayed on the righthand side of the screen, a second contour plot can be displayed on the lefthand side. This is done using the '/OVERLAY' qualifier, which allows a second plot to be displayed on the screen without erasing the first. So, if you use the 'CONTOUR/LEFT/OVERLAY' command, the second contour plot will appear on the lefthand side beside the first.

contour/right

The qualifier right allows the display to appear on the righthand side of the screen. If a contour plot already is displayed on the lefthand side of the screen, a second contour plot can be displayed on the righthand side. This is done using the '/OVERLAY ' qualifier, which allows a second plot to be displayed on the screen without erasing the first. So, if you use the 'CONTOUR/RIGHT/OVERLAY' command, the second contour plot will appear on the righthand side beside the first.

cont/v1=x

/v1,/v2,/v3,/v4 control where the image is displayed.

cont/v1=x

/v1,/v2,/v3,/v4 control where the image is displayed.

cont/v2=x

/v1,/v2,/v3,/v4 control where the image is displayed.

cont/v3=x

/v1,/v2,/v3,/v4 control where the image is displayed.

cont/v4=x

/v1,/v2,/v3,/v4 control where the image is displayed.

cont/scale=x

Controls the scale applied to the labels around the image.

cont/file=filename

Allow to input contour levels from an ascii file, which contains a column of level values.

Examples:

```
contour/no_of_cont_levels=20/first_contour=4   ! generate a contour plot with
                                               ! 20 levels, and level 4 to
                                               ! level 20
contour/no_of=20/first=4/overlay/col=5         ! same as before but also
                                               ! overlay the contour on the
                                               ! previous image using color 5
```

5.9 grid - overlay a sky coordinate grid

```
grid[/qual] [dalpha] [ddelta]
```

Draw a grid of sky coordinates over the currently displayed image. The default will plot a preassigned grid spacing. This can be changed by specifying dalpha and ddelta in degrees.

grid/galactic

Draw a the grid in galactic coordinates.

grid/color=n

Specify the color to be used for the grid. The default is white.

grid/tick

This will plot only tick marks on the frame of the image, corresponding to preassigned spacing in dalpha and ddelta. If not specified a full grid across the image is plotted.

Example:

```
grid                          ! draw a grid with the default steps in
                              ! in celestial coordinates

grid 0.5 0.5                  ! draw a grid in 0.5 deg ra and dec steps

grid/gal                      ! draw a grid in galactic coordinates
```

5.10 cpd - change plot device

```
cpd  device-name
```

This command defines the device where all graphic output is directed. cpd ? gives a list of all available devices.

Examples:

```
cpd /X11             ! change the current plot device to X-Windows (vms)

cpd /xw              ! change the current plot device to X-Windows (unix)

cpd test.ps/ps       ! direct the graphic output to a postscript file
```

5.11 detect - source detection

```
detect[/qualifiers]
```

Locates point sources in the current image by means of a sliding-cell method. The average background intensity is estimated in several small square boxes uniformly located within the image (see command BACKGROUND for a more detailed description). The position and intensity of each detected source are calculated in a box whose size maximizes the signal-to-noise ratio. If the source is not pointlike the estimated count rate is in general inaccurate and likely to be under-estimated. A good estimate of the intensity of extended sources can be obtained with command `SOSTA`.

Corrections for telemetry and instrumental dead times, for vignetting effects and for the fraction of the source counts that fall outside the box where the net counts are estimated are applied. Count rate errors include both statistical and systematic uncertainties added quadratically. To minimize the number of spurious sources detected the threshold used by DETECT is somewhat conservative. Consequently, some sources with intensity just above the image background can be missed. In order to allow DETECT to obtain a sufficiently good estimate of the background only images of size 128x128 pixels or larger should be used. Maximum accuracy is obtained running DETECT on full resolution images.

detect/draw_rej_back_boxes

This causes detect to display all background boxes which have NOT been used for the estimation of the background Different colors correspond to different reasons for rejection.

detect/source_box_size=

This defines the size (in image pixels) of the search cell. The default value of the search cell size depends on the instrument PSF and on the image pixel size. Try increasing it to reduce confusion.

detect/back_box_size=m

Define the size, m, (in image pixels) of the boxes where the image background is estimated. Image size must be a multiple of the background box size. Allowed background box sizes are 16, 32, 64, 128, etc...

detect/bright

Do not merge the excesses before optimizing the box centroid. This is slower, but can be better for crowded fields and fields containing bright sources.

detect/plot_excesses

Plot the excesses. This is a useful diagnostic if things go wrong.

detect/prob_limit

Change the background probability limit from the default value (`1E-06`).

detect/snr_threshold

Change the signal to noise acceptance threshold from its default value of 2.

detect/thr_scaling

Scale the threshold for detecting source excesses. Try increasing it to 1.1 or 1.2 or 1.3 to reduce problems with fluctuations from extended emission.

detect/file_out

To specify an output filename which contains the detect results. The default value is the input filename with extension '.det'. The output file from detect is a plain ascii.

Examples:

```
detect              ! Search for all point sources using the default values.

detect/source=10    ! Search for point sources using a search cell size of
```

```
                     ! 10 image pixels

detect/draw_rej      ! Draw the background boxes that have been rejected to
                     ! be drawn onto the image (if this has been displayed).
                     ! All other parameters are set to default values.

detect/back=32       ! Estimate the image background using boxes of size 32
                     ! image pixels. All other parameters are set to default
                     ! values.
```

5.12 display - display the current image

```
display[/qualifiers]
```

Up to 256 colour levels can be used. The actual number of levels used depends on the plot device defined. If the plot device does not support colours a gray scale image is displayed. By default the dispay uses a histogram to determine the color levels. Try the `/lin` or the `/log` qualifiers to get a different view. The `cct/loop` command can be used to go thru the various color tables that are available.

display/linear

Display image with linear scale for colour levels

display/log

Display image with logarithmic scale for colour levels

display/histo

Display image using image histogram to define colour levels. This is the default option for command DISPLAY

display/correct_exposure

Display the image corrected for the exposure map

display/exposure

Display the exposure map

display/div_map

Display the current image divided by the saved image

display/div_saved

Display the saved image divided by the current image

display/saved_image

Diplay the saved image.

display/sub_map

Display the current minus the saved images. If the result is a negative number, it will reset it to zero.

display/user_defined

Display image using user-defined colour levels. The user is prompted for n colour levels. The default value for n is the maximum number of colour levels the graphic device used can handle.

display/levels_from_file

Colour levels are read from a user-supplied file which can be created using command SAVE_LEVELS. The `/file` qualifier specifies the filename.

display/file=filename

A file containing colour levels for image display this qualifier can be used only in conjunction with qualifier `/levels_from_file`

display/table=name

This specifies the colour table to be used for the display. (not implemented yet.)

display/noframe

This suppresses the displaying of image frame. Only image and title are displayed.

display/overlay

The image is overlayed to the pre-displayed image. If an image has been previously displayed on one side of the screen (es. left), specifing this qualifier in combination with the es. '/RIGHT' qualifier allows the second image to appers displayed on the righthand on the screen (see '/LEFT' or '/RIGHT' for usage). If the qualifier is not specified the image is displaied on a new Pgplot page.

display/left

The qualifier left allows the image to appear on the lefthand side of the screen. Any subsequent interacting command with the Pgplot screen will act on the lefthand side. If an image already is displayed on the righthand side of the screen, a second image can be displayed on the lefthand side. This is done using the '/OVERLAY' qualifier, which allows a second plot to be displayed on the screen without erasing the first. So, if you use the 'DISPLAY/LEFT/OVERLAY' command, the second image will appear on the lefthand side beside the first.

display/right

The qualifier right allows the image to appear on the righthand side of the screen. Any subsequent interacting command with the Pgplot screen will act on the lefthand side. If an image already is displayed on the lefthand side of the screen, a second image can be displayed on the righthand side. This is done using the '/OVERLAY' qualifier, which allows a second plot to be displayed on the screen without erasing the first. So, if you use the 'DISPLAY/RIGHT/OVERLAY' command, the second image will appear on the righthand side beside the first.

display/v1=x

/v1,/v2,/v3,/v4 control where the image is displayed.

display/v1=x

/v1,/v2,/v3,/v4 control where the image is displayed.

display/v2=x

/v1,/v2,/v3,/v4 control where the image is displayed.

display/v3=x

/v1,/v2,/v3,/v4 control where the image is displayed.

display/v4=x

/v1,/v2,/v3,/v4 control where the image is displayed.

display/scale=x

Controls the scale applied to the labels around the image.

display/spectrogram

Display images which contain spectra obtained at different time and formatted as an image array. Not fully implemented yet.

display/refresh

This qualifier is disable

Examples:

```
disp              ! display an image with a histogram scaling

disp/log          ! display an image with log scaling

disp/exp          ! display the exposure map

disp/lin          ! display with a linear scaling

disp/left         ! display an image on the left
disp/right/overlay ! display an image on the right, and keep the pervious
                  ! image

disp/v1=0.317/v2=0.617/v3=0.4/v4=0.7 ! display into a specific area of the
                                     ! the screen
```

5.13 cey - change the equinox

`cey epoch`

All commands using celestial coordinates, including `detect`, `ra_dec_to_pixel`, and `pixel_to_ra_dec`, use the equinox epoch set with this command. The default equinox is that of the image. The value actually in use can be checked with command `SHOW`.

Example:

```
cey 2000                    ! Change the equinox year to 2000
```

5.14 exit - quit

Return to the operating system. A `$` can be used to spawn to the operating system without losing the current session.

You can exit from XIMAGE also by typing `quit` or `bye`

5.15 extract - a spectrum and a lightcurve

`extract[/qualifiers] [filename]`

Extract a PHA spectrum or a light curve from a binary FITS file. If no filename is given the routine assumes that the event file from which the data is to be extracted is the same as used to extract the current image (see the `read/rfits` command). This command creates a file with the same name of the FITS file and extension equal to .log and .qdp for the spectrum and lightcurve respectively. These output files are ASCII and can be used as input to programs XSPEC and XRONOS respectively.

The spectrum and lightcurve is extracted only for the region specified in a region file. This region file uses the saoimage region file format. It can be created using saoimage or from ximage using the `circ` and `box` commands. The default region file will have the same name as the name of the image/event file, with the extension `.reg`. If the region file is created within XIMAGE the pixel values are those corresponding to the unrebinned image. If the region file is created within SAOIMAGE the pixel values are those corresponding to a rebinned image. The region given to the 'extract' commands needs to be in original (unrebined) pixels.

If the qualifier `/detect` is used then the output from a previous run of detect is used to generate a set of hardness/softness ratios for the detected sources.

If `/detect` is not given, then extract will automatically generate a spectral file that can be read by the psf command, so as to make an energy dependent psf for the source in question.

extract/c1=n

Specify the channel, n, to be used to generate hardness and softness ratios. The defaults are currently set for ROSAT. To be used together with /c2, /c3, /c4.

extract/c2=n

Specify the channel, n, to be used to generate hardness and softness ratios. The defaults are currently set for ROSAT. To be used together with /c1, /c3, /c4.

extract/c3=n

Specify the channel, n, to be used to generate hardness and softness ratios. The defaults are currently set for ROSAT. To be used together with /c2, /c3, /c4.

extract/c4=n

Specify the channel, n, to be used to generate hardness and softness ratios. The defaults are currently set for ROSAT. To be used together with /c1, /c2, /c3.

extract/detect

Use the positions from a previous run of detect and generate the hardness and softness ratios for each source

extract/detect/colorfile=file

The name of the output file with the HR and SR data. The default is the input filename.col. Use /plot to plot a color-color diagram.

extract/filename=file

Name of the input event file.

extract/phafile=file

Write a spectral file in ASCII format. This file can be converted to xspec format using MKPHA.

extract/plot/detect

Plot a color-color diagram for all the detected sources.

extract/qdpfile=file

Write a timing file in QDP format. This file is written by default; use none to avoid it being written.

extract/region=file

Enter an SAOIMAGE style region file name. If not specified the regionfile name is equal to the events file .reg.

extract/window=file

Enter a XRONOS style timing window file.

extract/phahigh=n

Specifies the minimum of a range of pha channels to be used when constructing a light curve.

extract/phalow=m

Specifies the maximum of a range of pha channels to be used when constructing a light curve.

extract/variability

Calculate a statistical variability test for detected sources. Values greater than 100 are porbably variable.

extract/energy

When making a color-color diagram it will default to use the PI (pulse invariant) channel. The energy qualifier will use the PHA column instead.

extract/psf_pha

It creates an ASCII pha file which name is telescope_detector.psf, used automatically by the command 'psf'. In the case of the ROSAT PSPC the ASCII pha file is rosat_pspc.psf.

Examples:

```
extract/reg=agn.reg rp200123.fits     ! Extract a spectrum from the rosat
                                      ! style fits file

extract/detect/plot                   ! plot a color-color diagram for
                                      ! all the detected sources
```

5.16 flux - Count-rate to flux conversion

```
flux[/qualifiers]
```

Converts instrument count-rate into flux in `erg cm**-2 s**-1`. The flux can be estimated both at the detector or before absorption by the Galaxy. The energy band where flux is calculated depends on the instrument specified. If no qualifiers are given, the user is prompted for all the critical information. Only EXOSAT CMA1 and CMA2 and Einstein HRI are implemented.

flux/instrument=name

This specifies the `name` of the instrument to be considered. Supported instruments are: LE1, LE2 (EXOSAT Low energy imaging telescopes with CMA at focus), and the HRI (Einstein).

flux/nh=x

The Galactic interstellar absorption in units of atoms cm^{-2}

flux/slope=y

The Energy slope for power law model

flux/temperature=z

Temperature in keV for Black Body and Thermal Bremsstrahlung models

flux/model=mod

Spectral model to be used for the conversion. Supported models are:

```
power law              -> model=pl
black body             -> model=bb
thermal bremsstrahlung -> model=tb
```

flux/filter=n

The filter used in conjunction with EXOSAT CMA. The following filter coding should be used:

```
Polypropylene -> filter=2
Thick lexan   -> filter=3
Al/Parylene   -> filter=6
Thin lexan    -> filter=7
Boron         -> filter=8
```

flux/gal_correction

Logical qualifier specifying whether the flux has to be corrected for Galactic absorption. If this qualifiers is not specified the flux is calculated at the detector.

flux/count-rate=x

the instrument count-rate, x, in units of counts/s

Examples:

```
flux/mod=pl/sl=0.7/nh=3.e20/cou=.002/inst=le1/filt=7

Calculate a flux corresponding to a count rate of 2.e-3 ct/s in the CMA1 +
thin lexan filter assuming a power law model with energy index equal to 0.7
and nh=3.e20. No correction for Galactic absorption is applied.
```

5.17 header - display the ximage header

```
header[/qualifiers]
```

List out the header of the current image. This includes:

```
instrument
image title
exposure time
coordinates of image centre
roll angle
image size in pixels
rebin factor
pixel size in arcsecs
min and max of image
```

header/exposure

List out the header of the exposure map associated with the current image.

5.18 help - ximage help

```
help  [command name]
```

Help will describe in detail each XIMAGE command. `Ctrl/Z` to exit from the help routine; `<cr>` to move up the help back to the beginning.

Examples:

```
help detect          ! show detailed information about command ‘detect‘
```

5.19 hotspot - show position of EXOSAT CMA hot spot

```
hotspot
```

Shows the position of the CMA hot spot on a pre-displayed image. This command only works for EXOSAT LE-1 or LE-2 images that have not been rotated.

Example:

```
hotspot            ! A small circle is drawn around the expected position
                   ! of the CMA hot spot.
```

5.20 info - text overlay

```
info[/qualifiers] text
```

Overlay text information onto a pre-displayed image. In its default mode info includes image orientation, instrumentation name, exposure time and image scale. By giving a string it will display that string. In this mode the /xs and /xy qualifiers canbe used to control the position of the text. The /scale qualifier controls the size of the text.

info/xs=x

The x location of the beginning of the text. Zero is the bottom left corner, 512 the top right.

info/ys=x

The y location of the beginning of the text. Zero is the bottom left corner, 512 the top right.

info/scale=x

The scale size of the text.

info/short

Only the The scale size of the text.

Examples:

```
info                          ! write the default info

info "text string"            ! write "text string" in upper left hand corner
```

```
info/xs=256/ys=256/scale=2.0 ! display the text in the middle of the
                             ! image at twice the default size.
```

5.21 label - write labels on the image

```
label[/qualifiers] "text"
```

This will write labels anywhere you want, with any size. Use the `/xpix=` and `/ypix=` qualifiers to determine the location, in original pixel coordinates. The text must be included in double quotes e.g. "write this here". A symbol can also be written using the `/symbol=n` qualifier.

label/color=n

Redefine the color used for the label. The number of each color can be found by doing "scale" and counting the colors from left to right.

label/size_of_symbol=x.x

Redefine the size of the symbol from the default value. Values more than 2.0 will tend to be too large!

label/symbol=n

Plot a symbol on the screen. The value n defines different symbols.

label/width_of_symbol=n

Change the width of the symbol.

label/xpix=x.x

Define the X pixel where the label is to be placed, in original pixels.

label/ypix=y.y

Define the Y pixel where the label is to be placed, in original pixels.

Examples:

```
label/x=128/y=128 "<-- 4U 1822-37"      ! Write this label
```

5.22 levels - list intensity levels

```
levels
```

List out the intensity levels last used for image display.

5.23 log - open a log file

```
log [filename]
```

Write a log file containing the screen output generated by XIMAGE.

The option `filename` specifies name of the file; the file extension is .log. If no filename is specified then the default of ximage.log is used.

A log file will be closed if:

- a new log file name is specified (this will open a new log file);
- the filename is given as "none";
- or XIMAGE is exited.

The log files are in plain ASCII text.

Examples:

```
log results               ! open a log file called results.log
```

5.24 pixel_to_ra_dec - convert pixel to sky coordinates

```
pixel_to_ra_dec[/qualifiers]
```

Converts a specified pixel position in the current image to RA and DEC. If `/cur` is not specified, then XIMAGE will ask for X and Y pixel coordinates.

pixel_to_ra_dec/cursor

The pixel position is specified using cursor.

Examples:

```
pixel_to_ra_dec/cur          ! Input the position using the cursor.
```

5.25 polygon - define a polygon area

```
polygon[/qualifiers]
```

Defines polygonal areas by means of the cursor on a pre-displayed image and returns the image and celestial coordinates of all vertices.

polygon/line

A line is drawn between vertices

polygon/fill

The polygonal area specifies is filled with solid colour.

polygon/symbol

Symbol to be used to mark the vertices. Symbols are specified by means of numbers. The correspondence number-symbols is that defined in the PGPLOT package. The default symbol is number 2 (a cross).

Examples:

```
polygon/line                 ! Define a polygonal area and draw straight lines
                             ! between the vertices.

polygon/line/fill            ! fill the polygon with the last defined colour.

polygon/symbol=3             ! Define a polygonal area marking the vertices
                             ! with symbol number 3 (a star)
```

5.26 psf - estimate the point spread function

```
psf[/qualifiers]
```

Calculates differential and integral Point Spread Function and plots them onto the currently defined graphic device. The expected PSF for a pointlike source is also calculated and displayed. This command needs an accurate estimate of the image background. Use the command `back/opt` before running `psf`.

The default counts all the events within an annulus that increases by one pixel per iteration. Use the `/box` qualifier to use boxes instead.

psf/back=x

Enter the image background in units of `cts/original-pixel`. This can be obtained using the `BACK/OPT` command.

psf/box

Use boxes, not annuli.

psf/calfile

Write the psf to an ascii file called `ximage_psf.dat`, which can then be picked up by detect, sosta, etc....

psf/cursor

Enter the position of the centroid and outer radius using the cursor.

psf/energy

The expected PSFs for different detectors are derived in XIMAGE using the 'theoretical' algorithms usually derived from ground or inflight calibration data. Dependent on the detector the expected PSF can be energy dependent. This qualifier allows to specify the 'interesting' energy value (in keV, the default value is 1.0). The expected PSF is plotted together the measured PSF.

psf/x=n

Specify the x coordinate n for the center of the box position.

psf/y=m

Specify the y coordinate m for the center of the box position.

psf/fileout=name

Write the psf results to an ASCII file called `name`.

psf/print

Write on the terminal the quantities calculated for each step.

psf/radius

Give the outer radius in arc minutes.

Examples:

```
psf/cur/back=2.3e-03         ! calculate the psf for a position specified
                             ! by the cursor and setting the background.
psf/cur/back=2.3e-03/rad=6   ! calculate the psf for a 6 arc min rad
```

5.27 ra_dec_to_pixel - sky to detector coordinates

```
ra_dec_to_pixel[/qualifiers]
```

Converts RA and DEC into (current image) pixel coordinates. A small symbol is drawn at the corresponding position on the image if this has been previously displayed. The user will be prompted for the coordinates in the equinox specified using the `equinox` command.

ra/circle=arcsec

Draw a cricle with a radius in arcsec around the given position. Useful for drawing error circles

ra/ra=hms

Give the ra as part of the command line. The RA can be given as either decimal degrees or as hr, mn, sec. In the latter case it must be included in double quotes (e.g. /ra="12 23 12.1")

ra/dec=dms

Give the dec as part of the command line. The dec can be given as either decimal degrees or as degrees, minutes and seconds. In the latter case it must be included in double quotes (e.g. /dec="12 23 12.1")

Examples:

```
ra                                  ! Enter the RA and Dec from a prompt
ra/circle=600.0                     ! Draw a 600 arc sec radius circle centered
                                    ! at the given ra and dec
ra/dec="-61 12 1.0"/ra="11 45 2.1" ! Give the RA and DEc on the command line
```

5.28 read_image - read an image or events files

```
read_image[/qualifiers] [image-name]
```

The image read becomes the `current image`. The following qualifiers specify the format of the image:

```
/fits   -  a FITS image (in the primary array)
/rfits  -  a photon event file in FITS format (ROSAT RDF format)
/EXO    -  EXOSAT image files
/COMP   -  EXOSAT compressed image files (VMS only)
```

The default assumes `/fits`.

read/exp_map

The `/exp_map` qualifier indicates the image contains an exposure map and loads it into the exposure map arrays. It can be displayed using the `disp/exp` command.

read/exosat

File is in EXOSAT format. This is machine specific.

read/fits

Image is in FITS format. This is machine independent.

read/xpr

File is in Einstein IPC xpr format. This is only supported under vms and is for expert use only.

read/rfits

File is FITS photon event format, using the format of the ROSAT Rev 0, 1 and 2 (RDF) defined by the US ROSAT data center and the HEASARC.

read/compressed

File is in exosat compressed format. This is machine specific and only works under VMS.

read/xpix=n

Load an image centered on x pixel n. The pixel value should be specified in the original 'standard' range used within a detector.

read/ypix=n

Loads an image centered on y pixel n. The pixel value should be specify in the original 'standard' range used within a detector.

read/ra=

Load an image centered on the specified r.a.. The value can be specified either as ra="hh mm ss" (hour minute and second) or ra=deg.ddd (in degrees). It is assumed the current XIMAGE equinox for the specified r.a. (the command SHOW lists the current XIMAGE equinox, and the command CEY changes the current XIMAGE equinox).

read/dec=

Load an image centered on the specified dec.. The value can be specified either as dec="deg mm ss" or ra=ddd.ddd (in degrees). It is assumed the current XIMAGE equinox for the specified r.a. (the command SHOW list the current XIMAGE equinox, and the command CEY change the current XIMAGE equinox).

read/degrees

Used to specify if coordinates are given as degrees value. This qualifier has been temporary inserted to interface with a grafical user interface, and it is not necessary if /RA and /DEC are specify in degress.

read/equinox

To specify the equinox for the requested coordinate image center, given using /RA and /DEC. By default the current XIMAGE equinox is assumed.

read/size=n

The qualifier `/size=n` specifies the image size in pixels. Only square images of size 8,16,32,64,128,256,512, or 1024 pixels are allowed.

If the size is given as a negative value, then it defines the size of the image in arc minutes. This is useful when overlaying or comaparing two images. Be warned that it will round this number to the closest even pixel.

read/rebin=n

As the image is read into memory it is rebinned by n bins. n is 1 or a multiple of 2 (the default is 1). This canbe used in combination with the `/size` qualifier to unzoom an image.

read/detector

Allow to accumulate an image using the content of the DETX, DETY columns. Valid only for /RFITS format.

read/fits/header

Only read the header of a FITS image. Image is not loaded.

read/min_pha=m

The minimum of a range of acceptable values of PHA channels.

read/max_pha=n

The maximum of a range of acceptable values of PHA channels.

read/min_pi=m

The minimum of a range of acceptable values of PI channels.

read/max_pi=n

The maximum of a range of acceptable values of PI channels.

read/t_min=mm.mm

The minimum time accepted. Available only for data in RFITS format. The time value should be given in the same format used in the TIME coumn of the fits file.

read/t_max=nn.nn

The maximun time accepted. Available only for data in RFITS format. The time value should be given in the same format used in the TIME coumn of the fits file.

read/image_number

(Not implemented yet). Allow to specify which image to read, if multiple images are stored in the primary header of a FITS file,

read/file=filename

Allow to specify the filenmae of the image or event list using a qualifier instead of specifing the filename on the command line.

read/regionfile=filename

Allow to specify a spacial region file. The circle or box region file can be created using the command '/CIRCLE' and/or '/BOX'.

read/time

The '/time' qualifier allows to generate a time variability map obtained calculating the time chi-square distribution in a pixel. Chi-square values are stored in each pixel scaled from 0-200, values greater than 100 mean the source is probably variable. Its important to make sure there are a sufficient number of photons per bin, by heavily rebinning the image. To use this the event file must be time ordered.

read/color

The '/color' qualifier allows to generate a color weighted image obtained using the PI or a PHA column in an event list. In each pixel the energy information is summed and weighted by the number of photons. The resulting image is scaled from 0-100.

Examples:

```
read/fits/x=a/y=b/size=512 file-name     ! decompress an image stored in
                                         ! file-name into an image
                                         ! centered on x=a,y=b and of
```

```
                                        ! size=512

read/rfits/min_pha=a/max_pha=b          ! loads an image from a binary FITS
                                        ! events file including photons
                                        ! having pha channels within a and b.
```

5.29 rebin - rebin the current image

```
rebin[/qualifiers] factor
```

The current image is rebinned by the number of bins specified by the `factor` parameter. The original image can be restored using the `restore` command. The `/mode` parameter can be used to apply some slight smoothing, or do rebinning for mosaics.

rebin/mode=n

The mode for rebinning where n is:

```
mode =0  rebinning with slight smoothing
mode =1  normal rebinning, the default
mode =2  rebinning for image mosaics
```

rebin/rebinning_fact=m

Used to specify the rebinning factor. The command 'rebin/rebinnig=2' is equivalent to 'rebin 2'

Example:

```
rebin 8             ! rebin the image by a factor of 8
```

5.30 remove_sources - remove sources

```
remove_sources[/qualifiers]
```

This removes hot spots (or events caused by cosmic rays) from current image. The routine first calculates the average image background then fills the area including the hot spot with background events.

remove/cursor

Specifies that two opposite corners of the rectangular box including the hot spot are input through a cursor.

remove/x_min=xxx

x coordinate (pixels) of lower (or upper) left corner of rectangular box including the hot spot to be removed

remove/x_max=xxx

x coordinate (pixels) of lower (or upper) right corner of rectangular box including the hot spot to be removed

remove/y_min=yyy

y coordinate (pixels) of lower left (or right) corner of rectangular box including the hot spot to be removed

remove/y_max=yyy

y coordinate (pixels) of upper left (or right) corner of rectangular box including the hot spot to be removed

remove/detect

remove all the sources found in the last run of DETECT.

remove/nsources=n

put remove into a loop so that it prompts to remove the number of sources specified by n.

remove/zero

replace the defined region with zeros, rather than the randomized background.

Examples:

```
remove/cursor             ! remove a region defined by the cursor

remove/detect             ! remove the sources found by detect
```

5.31 rescale - rescale the current image

```
rescale[/qualifiers]
```

The current image can be rescaled either by multiplying every pixel value by a constant (the default) or by adding a constant to all pixel values.

rescale/multiplicative

Specifies that the scaling is multiplicative, that is, the value of every pixel in the current image is multiplied by a constant.

rescale/additive

Specifies that the scaling is additive, that is, the value of every pixel in the current image is increased by adding a constant.

rescale/scaling_factor=value

Enter the scaling factor value.

Examples:

```
rescale_image/mult/scal=10     ! every pixel value in the current image is
                               ! multiplied by a factor 10.

rescale_image/add/scal=25      ! 25 is added to every pixel value in the
                               ! current image.
```

5.32 resize - change the pixels size of the current image

```
resize/[qualifiers]
```

This command allows the size of the pixel to be increased up to a maximun of twice the original value. This is useful when two images from different detectors need to be summed but the pixel scale for the two images is different.

resize/pixel_size

Specifies the value of the new pixel size.

Examples:

```
resize/pixel_size=10           ! the pixel size will be change from the
                               ! original value to 10
```

5.33 restore_image - restore the saved image

```
restore_image
```

Copies the image previously read in, or saved with command `save_image` to the current image.

Examples:

```
restore_image                    ! restore the saved image
```

5.34 rotate - rotate the current image

```
rotate[/qualifiers]
```

This routine properly rotates sparse arrays such as X-ray images. In order to preserve the poissonian nature of the background a Monte Carlo redistribution of some events is carried out. The total number of photons is strictly conserved. This operation is however `irreversible`, that is, a rotation of +x degrees, followed by a rotation of -x degrees will produce an image that although closely resembles the original one, it is not identical. For this reason it is advisable to keep to a minimum the number of rotations performed on one image.

rotate/roll_angle=dms

This qualifier can be used to specify the roll angle at which to rotate the image. The default angle is -90 00 00, corresponding to North up and East to the left. Degrees, minutes and seconds must be separated by commas.

Examples:

```
rotate/roll_angle=-20.1             ! Rotates the current image to a roll
                                    ! angle of 20.1 degrees
```

5.35 saoimage - Start the SAOIMAGE display package

```
saoimage
```

SAOIMAGE is a general image display program developed by Michael VanHilst at the Smithsonian Astrophysical Observatory. SAOIMAGE runs only under X-windows therefore it can be started only if the plot device is set to `/X11` (VMS) or `/xw` (unix). When Ximage uses SAOIMAGE it communicates via a binary imagefile. It is important to note that any region files written from saoimage will not correct for any rebinning made within XIMAGE. The XIMAGE `circ` and `box` commands do not suffer from this problem.

5.36 save_image - save the current image in memory

```
save_image
```

Copy the current image and exposure map to a saved area in the volatile memory. They are restored using the `restore_image` command.

Example:

```
save_image                     ! save the current image and exposure map
```

5.37 save_levels - save current color levels

```
save_levels[/qualifiers]
```

Save current color levels in a user specified file. The default extension of the file is .tab, and the file is written in the current directory. It might be used later to display images with the XIMAGE command `DISPLAY/LEVEL_FROM_FILE/FILE=filename`

save/file=name

Define the filename into which write the levels to save. There is no default. If the filename is not given, the user is prompted for it.

Examples:

```
save_levels/file=test     ! save last used colour levels to file test.tab
```

5.38 scale - plot color scale

```
scale[/qualifiers]
```

Plot on the display the color scale used in the horizontal direction of the screen. Each color is associated to an intensity level calculated or input within the display commands. Aside to the color scale will appear the values of the intensity level associated to each color.

scale/label="string"

Write a string to label the color scale.

scale/no_of_ticks

To increase the number of intensity level written aside to the color scale. This number has not excede the number of levels.

scale/vertical

force the scale be plotted in the vertical direction of the screen.

Examples:

```
scale                     ! display the color scale horizontally
scale/vertical            ! display the color scale vertically
```

5.39 script - write a command script

```
script [filename]
```

Open a script file where all XIMAGE commands are written. This command is useful to create macros including a set of commands that have to be executed several times. The option `filename` specifies name of the file; the file extension is .xco. If no filename is specified then the default of ximage.xco is used.

A script file will be closed if:

- a new file name is specified (this will open a new file);
- the filename is given as `none`;
- or ximage is exited.

The script files are in plain ASCII text. `XIMAGE @filename` will re-execute all the commands stored in the file `filename.xco`.

Examples:

```
script filename            ! all commands subsequently executed are stored
                           ! in a file called `filename.xco`
```

5.40 show - current ximage status

```
show[/qualifiers]  [experiments] [all]
```

Show specific information about the current status of XIMAGE including:

```
XIMAGE version number
user address, name and privileges (if any)
current image name
plot device
equinox epoch
image size
currently used colour table
```

If `show experiments` is given the current instruments for which ximage has information is displayed. `show all` will show all the information.

Examples:

```
show                 ! show the current ximage status
show experiments     ! show all the experiments currently supported by XIMAGE


show all             ! show current status and all the experiments.
```

5.41 simulate - simulate an image

```
simulate[/qualifiers]
```

Simulate an image or photon event file. Up to 1000 point sources with random, or specified, position inside the image can be simulated above a background that can be flat or include spatial non-uniformities. Source intensities can follow up to two power-law logN-logS distributions, one for AGN and one for stars.

Additional source positions and intensities can be specified in a file called `catalog file`. This ascii file must contain the right ascension, the declination (in degrees, 1950.0) and the flux (in `erg/cm2/s`) of each source written in free format. All parameters for this command are included in a file called

```
xanadu/image/ximage/files/"satellite-name"_"detector-name".pf
```

Most of the parameters included in the parameter file can also be changed using the corresponding qualifiers. The PSF of each source can be changed and vignetting effects included. The energy spectrum and light curve of each source and background are also simulated if a `.fits` file is requested. Up to ten different spectral shapes can be used for a simulation. The details of each spectral shape are included in files that can be generated with XSPEC. The number of files and their names must be specified in the parameter file. If more than one spectral shape is requested, each simulated source is associated to one of the requested spectral shapes in a random way.

sim/part_background=

This specifies particle background in units of `cts/s/arcmin**2`

sim/cosmic_background

This specifies cosmic background in units of `cts/s/arcmin**2`

sim/catalog

The name of file containing a list of sources to be simulated in addition to those from the logn-logs distributions. The catalog file must include three columns: ra, dec, and flux. The right ascension and declination must be in degrees and for the equinox epoch 1950.0. Each line corresponds to a source. The center of the simulated image will coincide with the position of the first source in the catalog file.

sim/edit_pf

This qualifier is disabled.

sim/size_of_image

The size in pixel of image to be simulated. Only square images are supported.

sim/satelite

This specifies the satellite required. If it is not given ximage will prompt for it, and give a list of the currently supported missions.

sim/detector

This specifies the detector to be simulated. Detector names depend on the satellite requested. For a list of allowed detectors use command `show detectors`.

sim/flux_limit

Flux in `erg/cm2/s` corresponding to the instrument sensitivity limit. If this qualifier is specified the exposure time necessary to achieve the requested sensitivity is automatically calculated unless the `/exposure_time` qualifier is used, in which case the `/flux_limit` qualifier is ignored. Note that, in order to simulate the cosmic X-ray background just below the sensitivity limit, several sources with flux below the flux limit are also generated.

sim/pixel_size

The size of image pixels in units of arcseconds (must be an integer)

sim/zoom_s_f

The rebinning factor between source simulation grid and field grid.

sim/psf

forces XIMAGE to generate a single source positioned at image centre with intensity of about 2000 counts. This image can be used to estimate the size of simulated PSF using command `PSF`.

sim/raster_scan

Simulate a mini raster scan with 9 point sources located at regular positions within the image.

sim/file_psf

The ascii file containing the point spread function to be used for the simulation. The maximum psf size is 64x64 pixels. The ascii file should include n records (where n is `n=psf_size*psf_size/10+1`) and should be structured so that its first record contains only one real number equal to the psf size (e.g. 32) the following records contain 10 values, written in free format, specifying the psf brightness psf(i,j) scaled so that the total psf intensity is 1000. If the psf size is not a exact multiple of 10 one additional record, for each psf row, containing the remaining values must be added after the last full record of each row. For example if the psf size is 32 the ascii file should be made of 1+32*(3+1)=129 records. The first record includes the number 32.0, the following three records include 10 values each (psf(1,1-10), psf(1,11-20), and psf(1,21-30)), the fifth record include two values, psf(1,31),psf(1,32). The sixth to eighth record contain psf(2,1-10),psf(2,11-20), and psf(2,21-30), the ninth record psf(2,31),psf(2,32), and so on.

sim/fits

Make a file containing simulated event data in FITS format.

sim/output_fits_file

The name of the output fits events file to be generated.

sim/exposure_time

The exposure time in seconds. The corresponding flux limit of the exposure is automatiacally calculated on the basis of the background, of the search box, and of the statistical significance (snr) requested. If the exposure time is not specified or is set to 0, the exposure time necessray to achieve the requested sensitivity is automatically calculated.

sim/summary

List the available satellite and detector.

sim/parameter_file

The most of the parameters necessary for the simulation are stored into a parameter file called (VMS):

`xanadu:[image.ximage.files]"satellite-name"_"detector-name".pf`

or (unix)

`xanadu/image/ximage/files/"satellite-name"_"detector-name".pf`

(e.g. `xmm_ccd.pf`). The following is a description of all the parameters included:

```
  parameter  1 : 'output_data_file'
                 file where the positions and intensities of all
                 simulated sources are written.
  parameter  2 : 'flag_seed'
                 flag for radom number generator seed
                 flag_seed  < 0 seed entered interactively at run time
                 flag_seed  = 0 seed is radomly chosen
                 flag_seed  > 0 seed is set equal to flag_seed, must be
                                a large odd number
  parameter  3 : 'flag_fits'
                 flag for the generation of a fits event file
                 flag = 1 -> file is generated; 0-> not generated
  parameter  4 : 'output_fits_file'
                 name of the fits output file
  parameter  5 : 'flag_positions'
                 flag_positions = 1 random positions in FOV
                 flag_positions = 2 positions distrib like M31
                 flag_positions = 0 position are read from from (catalog) file
  parameter  6 : 'catalog'
                 name of the file including positions and fluxes
                 of the sources to be simulated.
  parameter  7 : 'logns_1_slope'
                 (integral) slope of first LogN-LogS (LogN-LogS for AGN)
  parameter  8 : 'logns_1_norm'
```

```
                  normalization of first LogN-LogS (units of sources/sq deg)
  parameter  9 : `logns_2_slope`
                  (integral) slope of second LogN-LogS (LogN-LogS for stars)
  parameter 10 : `logns_2_norm`
                  normalization of second LogN-LogS (units of sources/sq deg)
  parameter 11 : `flux_limit`
                  sensitivity limit of simulated field in ergs/cm2/s
  parameter 12 : `snr`
                  signal-to-noise ratio corresponding to flux limit
  parameter 13 : `search_cell`
                  search cell size in pixels
  parameter 14 : `particle background`
                  particle background in cts/s/arcmin**2
  parameter 15 : `cosmic background`
                  cosmic background in cts/s/arcmin**2
  parameter 16 : `file_psf`
                  file including point-spread-function
  parameter 17 : `zoom_s_f`
                  zoom factor between source and field
                  this allows the user to change the size of the psf
                  (by factors of two) without changing file_psf.
  parameter 18 : `counts_flux`
                  conversion factor between 1.0 ct/s and erg/cm2/s
  parameter 19 : `pixel_size`
                  pixels size in arcseconds (must be an integer)
  parameter 20 : `size_of_image`
                  image size in pixels (max=512)
  parameter 21 : `vign_1_2_3`
                  Coeff. for parabolic approximation for vignetting effect
                  three parameters corresponding to the coefficients of the
                  parabola that approximate vignetting effect as a function of
                  off-axis angle only.
  parameter 22 : ignore_me
                  ignore this parameter
  parameter 23 : ignore_me
                  ignore this parameter
  parameter 24 : `no_of_spectra`
                  no of template spectrum files to be used for spectral
                  simulations
  parameter 25 : `spectrum_1`
                  name of filele including template spectrum 1
  parameter 26 : `spectrum_i`
                  name of filele including template spectrum i
  parameter 27 : spectrum_n
                  name of filele including template spectrum n, n=no_of_spectra
```

Examples:

```
sim/sat=xmm/det=ccd/size=200/flux=5.e-15

Simulate a 200 x 200 pixels XMM CCD image with limiting sensitivity of
5.e-15 erg/cm2/s. Since the specified satellite is XMM the parameters which
were not given through qualifiers are read from the parameter-file.
```

```
sim/sat=sax/size=300/det=legspc/fits/out=sax.fits

Simulate a 300x300 pixel SAX Low-Energy GSPC image. A .fits file containing
spectral and timing information for every simulated photon is created. This
file can be used to extract (see command ‘extract‘) simulated spectra and
light curves.


sim/sat=rosat/det=pspc/fits/out=rosat/cat=deep_survey
         /exposure_time=100000

Simulate a ROSAT PSPC 100000 seconds exposure including all sources
contained in the file ‘deep_survey‘ (which could contain for example the
sources of the Eintein IPC deep survey). Since a ‘.fits‘ file will be
produced apectra and light curve for each source will be simulated as
specified in the parameter file.
```

5.42 skymap - overlay a starchart

```
skymap[/qual] filename
```

The skymap command reads in a list of star positions and overlays them on the current image. The default will open a connection to the HEASARC database and retrieve all the stars in its optical catalog. Using the `/db=` qualifier will restrict the search to one catalog.

Other opitions include reading an ascii file with the first line containing the equinox year (e.g. 1950) and on each following line the RA and DEC in decimal degrees. The SIMBAD source lists provided in FITS format by the US ROSAT data center can be read in using the `/RFITS_SIMBAD` qualifier.

skymap/color=n

Change the color used for the overlay.

skymap/file_input=file

Enter the filename via a qualifier.

skymap/labels

Overlay the source name too (`/rfits_simbad` only).

skymap/browse_dcoord

Read in a list of star positions from a file created within BROWSE (the HEASARC on-line database software), using the command dcoord/deg/full.

skymap/rfits_simbad

Read in a the SIMBAD source list that is provided by the US ROSAT data center. This is an ASCII FITS table.

skymap/size_of_symbol=n

Change the default symbol size used.

skymap/symbol=n

Change the default symbol used.

skymap/width_of_symbol=n

Change the width of the symbol used.

skymap/print_off

Print on the screen for each source in the catalogue, the x pixel y pixel and the offset in arcmin from the center of the detector.

skymap/class

Write on the image the class string if found in the file which contains the catalogue positions.

skymap/dbase=databasename

Connects to the HEASARC database system on legacy.gsfc.nasa.gov and makes a search cone and returns a file containing a list of coordinates of all sources found within the field of view of the current image. Using the /label and /class qualifiers will display respectively the name or class of the object. Use the /old qualifier to avoid repeating the query.

To obtain a list of valid databases logon into the xray account on legacy.gsfc.nasa.gov and type browse. Good databases to start with are OPTICAL, XRAY, ROSID, and RADIO.

skymap/old

Uses the results obtained from previous searches using the /dbase qualifier.

Example:

```
skymap/rfits/file=rosat.fits           ! read in a rosat simabd file

skymap/file=browse.txt                 ! read in a browse file.

skymap/db=optical/label                ! overlay all the sources in
                                      ! the optical catalogue
```

5.43 skyview - retrieve an image from skyview

```
skyview[/qualifiers] [survey name]
```

Open a connection to the HEASARC Skyview system and retrieve a sky survey image with the position and size of the current image. The retrieved image must be loaded in the usual way. The survey can be given as a qualifier, or given directly as the survey name. Use `/?` to list the surveys that are hard wired into XIMAGE. The retrieved image is given a file name that by default the `survey_name.img`, but can be named something else using the `/file` qualifier.

skyview/0408MHZ

Retrieve the Bonn 0408MHz survey

skyview/1420MHZ_BONN

Retrieve the Bonn 1420MHz survey

skyview/1420MHZ_BONN

Retrieve the NRAO 4850 MHz survey

skyview/DSS

Retrieve the digitized sky survey. Usually this should be displayed as a linear scale (`disp/lin`)

skyview/EGRET<100

Retrieve the Egret <100 Mev survey

skyview/EGRET>100

Retrieve the Egret >100 Mev survey

skyview/EUVE_171

Retrieve the EUVE 171 Angstrom band survey.

skyview/EUVE_405

Retrieve the EUVE 405 Angstrom band survey.

skyview/EUVE_555

Retrieve the EUVE 555 Angstrom band survey.

skyview/EUVE_83

Retrieve the EUVE 83 Angstrom band survey.

skyview/file=name

Specify a name for the retrieved file. If this is not given, a default will be chosen that uses the survey name.

skyview/HEAO1_A2

Retrieve the HEAO-1 A2 survey

skyview/IRAS100

Retrieve the IRAS 100 micron survey

skyview/IRAS12

Retrieve the IRAS 12 micron survey

skyview/IRAS25

Retrieve the IRAS 25 micron survey

skyview/IRAS60

Retrieve the IRAS 60 micron survey

skyview/list_surveys

List the available surveys.

skyview/pspc

Retrieve the ROSAT PSPC pointed images.

5.44 slice - draw an x or y slice of the image

```
slice[/qualifiers]
```

Input of slice limits is done using the mouse (the default) by clicking twice on the left button and then once on the right button, or using the `/start_pixel` and `/end_pixel` qualifiers. If the slice is along the x axis the cursor is clicked along the y axis to specify the range of y to be included in the x slice. The middle button can be used to undo the previous click. The middle button ends the selection and the slice is drawn. The default will draw the slice over the image. Use the `/nooverlay` qualifier to draw the slice outside the image.

slice/xslice

The slice is along x-axis. Click the cursor along the y axis to specify the range of y required

slice/yslice

The slice is along y-axis. Click the cursor along the x axis to specify the range of x required

slice/log

The slice data are plotted in a logarithmic scale.

slice/color=m

The color to be used for the slice plot.

slice/start_pixel

The pixel value defining the beginning of the slice (the default uses the cursor)

slice/end_pixel

The pixel value defining the end of the slice (the default uses the cursor)

slice/noooverlay

Draw the slice outside the image area.

slice/filename

Specify the output filename (to be use together with '/output').

slice/output

Write into a file the pixel number, the count per pixel (using only the not empty pixels in the slice), the total count, and the number of non empty pixels.

slice/previous_frame

Allow to draw more than one result slice in the same plotting frame.

slice/mean

slice/spectrogram

If the image contains a serious of spectra, the '/spectrogram" qualifier will plot the results of the slice in a separate window.

Examples:

```
slice/x                          ! do an x slice

slice/y/log                      ! slice along y axis with the result plotted
                                 ! on a log scale.

slice/y/log/start=100/end=200    ! y axis slice with the result plotted on a
                                 ! log scale. Slice limits are set to x
                                 ! start=100, x end=200.
```

5.45 smc - minimum display color level

```
smc
```

Set minimum colour level for display. This is useful when the graphic device is slow Setting the minimum colour level to a value greater than 0 can avoid plotting the image background and save time when the response from the computer (or from the connection line) is slow.

5.46 smooth - gaussian image smoothing

```
smooth[/qualifiers]
```

Gaussian smoothing of current image. The gaussian width can be specified with qualifier `sigma`. If `sigma` is not specified a default value of 1.5 pixels is used.

smooth/exposure_map

Smooth the exposure map.

smooth/sigma=n

To enter the user defined value of sigma for the Gaussian smoothing

smooth/scaling_factor=x

Enter a Scaling factor.

smooth/x_only

The smoothing is applied only in the x direction.

smooth/y_only

The smoothing is applied only in the y direction.

smooth/gaussin

A gaussian function is used in the smoothing. ThIs is the default function used.

smooth/wavelet

A wavelet function is used in the smoothing.

Examples:

```
smooth/sigma=2.         ! Smooth the current image using a gaussian filter with
                        ! sigma equal to 2 image pixels.
```

5.47 sosta - source statistics

```
sosta[/qualifiers]
```

Detailed statistical analysis of sources.

SOSTA counts the number of events within a specified box, corrects those counts for vignetting, exposure and the point spread function to give the source intensity and its statistical significance. If the significance is less than `1.e-3` an upper limit is automatically calculated. The simplest way to use SOSTA is with the cursor using the `/CURSOR` qualifier. In its default mode SOSTA will use a box that corresponds to 66% of the encircled energy function (EEF), and will use a background from a region surrounding the source, outside the 98% EEF boundary. The `/LOOP` qualifier can be used to repeatedly enter values using the cursor, or by typing in the values. To exit the loop click outside the image.

The background can be estimated using several different methods, specified by the user. These are:

- The background is estimated from a region within two boxes centred on the source using the option `/INNER_RADIUS` and `/OUTER_RADIUS` to specify the half-sizes of the two boxes. This is the default.
- The background is taken from any specified part of the image using `/BOX_BACKGROUND`;
- The background intensity can be given using the qualifier `/BACKGROUND_LEVEL`). In this case the background statistical error is assumed to be zero.

The uncertainty in the count rate returned by SOSTA is purely statistical i.e. does not include systematic errors. In general, sosta count rate errors are smaller than those given by DETECT which always adds (quadratically) an estimate of the systematic uncertainties. Also SOSTA will give slightly different count rates from DETECT, and is in most cases is probably more accurate. This is because DETECT uses a global background for the entire image, whereas SOSTA is using a local background.

SOSTA will estimate the optimum box size to maximize the signal to noise ratio. This is given at the end of each run. By using the option `/OPTIMIZE` then SOSTA will go around a second time, and use the optimum box size to calculate the optimum source statistics.

sosta/x_source

The source position in units of elemental x pixels. See also `/y_source`.

sosta/y_source

The source position in units of elemental y pixels. See also `/x_source`.

sosta/source_box_radius

This specifies the half-size of the box where the source intensity is estimated.

sosta/optimize

SOSTA makes an estimate of the optimum box size to maximize the signal to noise ratio. The `/OPTIMIZE` qualifier will make SOSTA go around a second time so that it uses the optimum box size to estimate the source statistics.

sosta/local_background

This is the default, and specifies that the background intensity must be calculated within two boxes centered on the source and of half-size equal to inner_radius and outer_radius respectively. The default starts the inner box at the 98% EEF and includes an area approx twice that of the 98% EEF area.

sosta/inner_radius=size

The radius of inner box (centered on the source) where local background is calculated. The background intensity is calculated within the inner and outer boxes and rescaled to the position of the source. Must be used in combination with `/OUTER_RADIUS`.

sosta/outer_radius=size

The radius of outer box (centered on the source) where local background is calculated. The background intensity is calculated within the inner and outer boxes and rescaled to the position of the source. Must be used in combination with `/INNER_RADIUS`.

sosta/box_background=size

This specifies that the background is estimated in user-defined boxes. The user is then prompted for the position and size of each box. If the `/SIZE_OF_BOXES` qualifier is used, than all the boxes have the same specified size. The `/CURSOR` can be used to select the corners of the boxes using the cursor.

sosta/no_of_boxes=n

The number of square boxes where the background is to be determined. the user is prompted for position and size for each box, or can use the `/CURSOR` option.

sosta/size_of_boxes=m

The half size of the boxes used to determine the background. All the boxes are then defined with the same size. If this qualifier is not used, the user is prompted for a size for each box.

sosta/background_level=counts

The background intensity is provided in units of `counts/elemental pixel`.

sosta/cursor

The image inputs (positions) are given from cursor.

sosta/loop

Enter multiple positions, without exiting sosta. Click outside the the image are to exit.

sosta/detect_sources

Run sosta on the sources found in the previous run of DETECT.

sosta/eef_source=frac

The fractional EEF used to determine the source box size. The default is 0.66.

sosta/eef_background=frac

The fractional EEF used to determine where the inner background box size begins. The default is 0.98.

Examples:

```
sosta/cur                ! The source position is specified using the cursor.
                         ! All the defaults are used. The background is
                         ! determined from a region surrounding the source.
```

```
sosta/loop/cur          ! keep pointing and clicking, over and over.

sosta/cur/opt           ! The optimum source box size is determined.

sosta/cur/box           ! The background is to be determined from a box
                        ! specified by the user.

sosta/cur/in=20/out=40  ! The background is determined from the difference
                        ! between the two boxes with half sizes 20 and 40
                        ! elemental radii.

sosta/cur/source=10     ! The source box size is set to 10 pixels.

sosta/cur/eef_s=0.5     ! The source box size is set to be at the 50% EEF.

sosta/cur/eef_b=0.5     ! The inner background box size is set to be at the
                        ! 50% source EEF.
```

5.48 sum_images - add the saved and current images

```
sum_images[/qualifiers]
```

This sums the current image with the image previously saved with command save_image. The result of the sum becomes the current image. The two images are rotated to a common roll of -90 degrees (north up, east left). The default uses the center of the current image. This command assumes the two images have the same elemental pixel scale size (on the sky).

To make further mosaics using the summed image, the summed image must first be saved.

sum_images/ra_center="hr mn sc"

The right ascension of summed image center. The default is equal to that of the current image.

sum_images/dec_center="deg mm ss"

The declination of summed image center. The default is equal to that of the current image.

sum/equinox

To specify the equinox for the requested coordinate image center, given using /RA and /DEC. By default the current XIMAGE equinox is assumed.

sum/degrees

To specify if coordinates are given as degrees value rather than "hr mn sc" and "deg mm ss".

Example:

```
sum_image              ! sum the current and saved images
```

5.49 title - write an image title

```
title  title-string
```

Defines a title to be used when displaying the current image. If title-string include one or more blank characters it must be written within quotes (")

title/lower

Force the title-string to be written lower case.

Examples:

```
title "This is a test"        ! write the string 'This is a test' at the
                              ! top of the image when displayed.
```

5.50 vignet - correct exposure map for vignetting

```
vignet
```

Multiply the current exposure map by the vignetting funtion. Becareful not to correct an exposure map that might already have been corrected.

Examples:

```
vig                           ! apply the vignetting
```

5.51 write_image - write the current image to a file

```
write_image[/qualifiers] [image-name]
```

Images can be written to a file using FITS or EXOSAT image file formats. The FITS format is portable across computers and is the default format. The EXOSAT format is machine dependant and is not recommended. The name of the file can be specifed as either a qualifier or as a positional parameter.

write_image/exposure

Write the exposure map to a file.

write_image/display_map

Write in a file the current display image.

write_image/exosat

write image in exosat format (not recommended).

write_image/fits

write an image in FITS format (this is the default).

write_image/file=name

The name of the image file to be created.

Examples:

```
write_image filename          ! store the current image to a disk file written
                              ! in FITS format
```

Chapter 6

Data formats

6.1 Input

XIMAGE accepts seven different types of files containing image, spectral and timing data. These files can all be read in by command `read_image`

EXOSAT - EXOSAT image files

These are simply integer*2 binary files with record length equal to the number of image pixels in the X dimension. The image header takes 64 bytes and is stored in the first record of the file if the record length is greater or equal to 64,bytes or in as many initial records as necessary if the record length is less than 64 bytes. As many data records as the number of pixels in the Y direction follow.

Compressed_EXOSAT - Compressed EXOSAT images

EXOSAT compressed image files are compressed to minimize the file size. For a description of the compression algorithm see J. Sternberg (The EXOSAT Express no. 18 page 25)

FITS - FITS image files

FITS files where the image is stored as the primary array. For details see "A User's Guide for the Flexible Image Transport System" by B. Geldzahler (send a note to `request@legacy.gsfc.nasa.gov` for a copy)

RFITS - FITS events files

Binary FITS tables containing photon event data. These are the rev0 data files released by the US ROSAT data center. For details about the FITS standard see "A User's Guide for the Flexible Image Transport System" by B. Geldzahler (send a note to `request@legacy.gsfc.nasa.gov` for a copy)

6.2 Output

EXOSAT Images

EXOSAT image files (.IMG) are produced by command `write/exo filename`. These binary files are operating system dependent (so dont ftp them!).

FITS Images

A FITS image file (.FITS) is produced by the command `write/fits filename`. These files are operating system independent.

Spectral Format

A file containing spectral information written in XSPEC ASCII format are produced by the `extract` command. They have the extension `.log` and can be made into XSPEC `sf` pha files using the `mkpha` program.

QDP lightcurves

File containing timing information written by the `extract` command in a QDP format that can be accepted by XRONOS. They have the extension `.qdp`

RFITS - Photon events file

Photon events files containing a list of photons as a binary table in FITS format. These conform to the Rev0 ROSAT format.