emldetect

January 27, 2025

Abstract

EPIC maximum likelihood multi-source point spread function fitting.

1 Instruments/Modes

Instrument	Mode
EPIC MOS:	IMAGING
EPIC PN:	IMAGING

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

For an input list of **eboxdetect** source locations, simultaneous maximum likelihood PSF fits to the source count distribution are performed in all energy bands of each EPIC instrument. A description of the main properties of the detection algorithm may be found in Cruddace, Hasinger, Schmitt (1988) and an overview of **emldetect** and its use in the XMM-Newton catalogue pipelines at
http://xmmssc-www.star.le.ac.uk/Catalogue/2XMM/UserGuide_xmmcat.html#EmldetFit
and at
https://xmmssc.aip.de/cms/users-guide/data-processing/.
For a concise description, see Traulsen et al. (2019), and for its use in the context of stacked source detection consult Traulsen et al. (2019, 2020).

Input images. Input images per instrument and energy band plus the corresponding exposure images, background images, and detection masks are supplied as lists of file names. The task is designed for a maximum of five energy bands and three EPIC instruments. It is technically possible to run **eboxdetect** for up to six energy bands, but users are encouraged not to use more than five because of the low count numbers in narrow-band images. All input images and accompanying exposure images, background images, and detection masks must have identical orientation and binning and must be supplied in a consistent order.

Fit parameters. Main free fit parameters are: the source location (image coordinates `X_IMA`, `Y_IMA`), source extent (Gaussian sigma or beta model core radius), and source count rates in each energy band for each telescope. The source location and source extent are constrained to the same best-fit value in *all* energy bands per EPIC instrument whereas the source count rates are adjusted to their individual best-fit value in *each* energy band per EPIC instrument. Derived parameters are: total source count rate, likelihood of detection (all-EPIC and in each energy band), likelihood of source extent, and up to four hardness ratios (default: four).

Output source table. A summary of all the columns in the output source list can be found in Table 2 in Section 8 on the output files. For each detected source, the table contains one row for each energy band for each instrument. The individual source rows are identified through the column entries `ID_INST` and `ID_BAND` in the output table. `ID_INST` refers to the EPIC instrument (1: PN, 2: MOS1, 3: MOS2, 0: summary row). `ID_BAND` is the energy band number as defined by the ordering of the energy bands, i.e.: the ordering of the input images given as command-line arguments to **emldetect**. The upper and lower bounds of each energy band are available in the header keywords *aa_n_*`ELO` and *aa_n_*`EHI` where *aa* stands for the EPIC camera (PN, M1, or M2) and *n* stands for the energy band number as given in the table column `ID_BAND`. Additional keywords `N_INST` and *aa*`_BNDS` specify the number of EPIC cameras and the number of energy bands for each EPIC camera. For the definition of the basic energy bands used in the pipeline processing, see
https://xmmssc.aip.de/cms/users-guide/catalogue-construction/catalogue-organisation/#ECFs.
An `ID_BAND` value of 0 refers to the summary rows of each source, which list combined results per instrument and in total. The summary rows over the energy bands for each instrument contain sums of the entries in the individual energy bands where appropriate (counts, count rates, fluxes, and detection likelihoods). In the EPIC summary row per source with `ID_INST`=0, those spatial parameters are repeated that are identical for all energy bands (positions and extent values). The other columns are set to NULL.

PSF fitting. Simultaneous fitting of data from different instruments (i.e., EPIC pn and MOS data) or different exposures is supported. The PSF fitting may either be performed in single-source or in multi-source mode. In multi-source mode, neighbouring sources with overlapping PSFs are fitted simultaneously. Detection likelihoods are optimized for all the overlapping sources simultaneously, and detection likelihoods per source are calculated and written to the column `DET_ML` of the output source table. Selection of sources for simultaneous fitting is controlled by the distance parameter `scut` and by the parameter `nmaxfit` that gives the maximum number of sources to be fit simultaneously ($1 \leq$ `nmaxfit` ≤ 10). Sources fit simultaneously are identified in the output table through the `ID_CLUSTER` table column. It is also possible to fit several PSFs for each input source position by setting the parameter `nmulsou` to the corresponding value ($1 \leq$ `nmulsou` ≤ 3, `nmaxfit`$*$`nmulsou` ≤ 10).

Two parameters determine the image region on which a source fit is performed: The parameter `ecut` determines the size of the subimage around each source used for fitting. The parameter `scut` determines the radius around each source, in which other input sources are considered for multi-PSF fitting, if the parameter `nmulsou` is > 1. Both `ecut` and `scut` are given as encircled energy fractions of the calibration PSF. The actual radii in pixel units therefore change slightly with energy band and source position. Alternatively, `ecut` and `scut` can be given as a fixed value in units of image pixels (if `ecut` or `scut` is > 1). The actual value for the cutout radius of each source is listed in the column `CUTRAD` of the output source list.

Starting with SAS 10.0 and **emldetect** version 5.1, a full 2d parametrization of the EPIC PSF as a function of instrument, energy, and off-axis angle is introduced (Release Notes, gzipped PS). The PSF model can be chosen via the parameter `psfmodel`. Up to version 5.17.1, the medium-accuracy PSF (`psfmodel`=medium) is used by default. The analytical 2d PSF (`psfmodel`=ellbeta) is the default PSF model from version 5.17.2 on. The slew-mode PSF for EPIC/pn (`psfmodel`=slew) has been introduced with **emldetect**-6.0 and **cal**-3.231.

Extent fitting. If the parameter `fitextent` is set to "yes", the point spread function will be convolved

with a source extent model, that can be set to either a Gaussian profile or a β-model profile via the parameter `extentmodel`. In the case of `extentmodel`=beta, the surface brightness is calculated as

$$f(x,y) = \left(1 + \frac{(x-x_0)^2 + (y-y_0)^2}{r_c^2}\right)^{-3/2}$$

The value of the core radius r_c is written to the column `EXT` of the output source list. In the case of a Gaussian extent model, σ is written to the column `EXT`, instead. Note that the source extent can only be determined reliably for relatively bright objects. If the likelihood of the source extent falls below the threshold given via `dmlextmin` (default: 10.0), point source parameters are derived.

From version 4.27 on, the extent-likelihood values (`EXT_ML`) are corrected for the number of input images with the formalism described below in the paragraph on Detection likelihoods.

From version 4.32 on, the maximum value of the extent fit parameter can be given via the task parameter `maxextent`. The unit is image pixels. Large values of `maxextent` can lead to spurious detection of extended sources in some cases. With the parameter `minextent`, the minimum extent can be specified that is still considered to be significant. If the best fit extent is less than `minextent`, a point source model will be adopted for the source.

And from version 7.5 (SAS v18) on, the extent likelihood of all detections is given in the output source list (not only of extended detections). `EXT_ML` is the likelihood difference between a point-like fit and a fit with an extent model. Negative values indicate a clear preference for the point-like fit, thus.

Detection likelihoods. All detection likelihoods are transformed to equivalent likelihoods L_2 (column `DET_ML` of the output source table), corresponding to the case of two free parameters to allow comparison between detection runs with different numbers of free parameters (i.e., when different numbers of input images are used):

$$L_2 = -\ln(1 - P(\frac{\nu}{2}, L')) \quad \text{with} \quad L' = \sum_{i=1}^{n} L_i$$

where P is the incomplete Gamma function, n is the number of energy bands involved, ν is the number of degrees of freedom of the fit ($\nu = 3 + n$ if the task parameter `fitextent` is true and the extent radius of the detection is larger than the user-supplied minimum extent, and $\nu = 2 + n$ otherwise), and $L_i = C_i/2$ with C as defined by Cash (1979). n is 1 for the individual energy-band detection likelihoods of a single observation that are listed in source table rows with `ID_BAND` > 0, and n is equal to the total number of energy bands in the `ID_BAND`=0 summary rows. The equivalent detection likelihoods obey the simple relationship $L_2 = -\ln(p)$, where p is the probability for a random Poissonian fluctuation to have caused the observed source counts. Note that for very small numbers of source counts (less than ≈ 9 counts, Cash 1979), this relation has to be treated with caution. Therefore, it will only give a rough estimate of the number of expected spurious sources.

Hardness ratios. If detection over several energy bands is performed, up to four hardness ratios `HR`i are calculated from the source count rates in the individual bands (default: four). The hardness ratios are defined as follows:

$$\texttt{HR}i = \frac{B_m - B_n}{B_m + B_n}$$

where B denotes the count rates in energy bands n and m, respectively. The energy bands n and m used to calculate the hardness ratios can be specified for each instrument via the parameters `hrpndef`,

Table 1: Default band assignments of hardness ratios HR*i* for the EPIC instruments and default energy intervals during pipeline processing[2].

i	n	m	Pipeline energy bands [keV]	
1	1	2	$0.2-0.5$	$0.5-1.0$
2	2	3	$0.5-1.0$	$1.0-2.0$
3	3	4	$1.0-2.0$	$2.0-4.5$
4	4	5	$2.0-4.5$	$4.5-12.0$

`hrm1def`, and `hrm2def`. The default band assignments (identical for all instruments) are given in Table 1. The band numbers n and m are assigned to the individual bands by numbering the corresponding input images in the order in which they are given on the command line. It is therefore important that the ordering of the input images is consistent with the contents of `hrdef` to obtain meaningful hardness ratios.

Fluxes and energy conversion factors. EPIC count rates and errors are converted to fluxes and flux errors by means of energy conversion factors (ECF = count rate / flux), given in units of 10^{11} cts cm cm / erg via the parameter `ecf` per image, i.e. per camera and per energy band. Fluxes and flux errors per instrument in the output source list are the sum of the values for each energy band. All-EPIC fluxes are error-weighted means. The parameter `ecf` defaults to 1.0, which means that *the default output fluxes of* **emldetect** *are not true source fluxes.* Users need to supply appropriate energy conversion factors for their input images, depending on the instrumental setup – camera, filter, response –, the event selection expression – in particular: patterns and energy band –, and the assumed spectral shape of the sources. New ECFs were derived for the 3XMM catalogue, are listed at
https://xmmssc.aip.de/cms/users-guide/catalogue-construction/catalogue-organisation/#ECFs,
and can be adopted for input images obtained with the same setup as described in the 3XMM documentation. In general, ECFs are estimated from spectral analyses of large source samples (or single sources), fitting or assuming a spectral shape and calculating count rates and fluxes for a given response e.g. within xspec. For the method, see Mateos et al. (2009), Section 3.5 of Rosen et al. (2016), and the technical note
http://xmmssc-www.star.le.ac.uk/Catalogue/2XMM/CAL-TN-0023-v2.0.ps,
and for examples the section "EPIC flux to count rate conversion" of the XMM-Newton Users Handbook at
https://xmm-tools.cosmos.esa.int/external/xmm_user_support/documentation/uhb/epicfluxtocr.html.

OOT correction. All EPIC PN source count rates and fluxes written to the **emldetect** source list are corrected for photons that arrive during readout of the PN CCDs and therefore are not detected on the nominal source position (out-of-time events).
emldetect *v4.5 to 4.27:* Correction factors of 1.0626 for PrimeFullWindow mode and of 1.0223 for the PrimeFullWindowExtended mode are applied by **emldetect**. Data taken in other observing modes are left uncorrected.
emldetect *v4.28 onwards:* From **eexpmap** v3.31 on, the OOT events correction is applied to the exposure maps in all observing modes. **emldetect** reads the keyword OOTCORR from the FITS header of the exposure maps. If OOTCORR is existing and set to "true", no further correction is applied by **emldetect**.

CPU saving. Since both multi-PSF fitting and extent fitting are CPU intensive, two methods exist to reduce the CPU requirements of an **emldetect** run using multi-PSF fitting. With the option `withthreshold`, the user can limit the application of multi-PSF fitting (as specified by `nmulsou`) to sources exceeding a certain threshold. The threshold is set by the parameter `threshold`. The corresponding input column is defined via the parameter `threshcolumn`, which can be `LIKE`, `SCTS`, or `RATE`.

The second method to save CPU time for combined extent and multi-PSF fitting is provided by the option `withtwostage`, which is used in combination with `fitextent`="true" and `nmulsou` > 1. If `withtwostage`

is set to "true", **emldetect** will perform the fit for each source in two stages: In the first stage, *one* extended source is fitted to the source. Only if the extent is significant, the second stage will be performed, and a multi-PSF fit with one extended source and `nmulsou`−1 point sources is applied. The `withtwostage` option avoids misidentification of close pairs of point sources as extended sources in most cases and significantly reduces CPU time.

Position errors. The final stage of the source detection process is done via ML-fitting of the PSF-shape at the given detector position to the observed photon distribution utilizing the C-statistics. The best fitting X-ray position is determined at the minimum value of C, and the 1σ errors in right ascension and declination are derived at $C = C_{\rm min} + 1$. The two-dimensional positional error RADEC_ERR, written to the output source list, is calculated as square root of the quadratic sum of the errors in R.A. and Dec. It translates into a one-dimensional σ =RADEC_ERR/$\sqrt{2}$, if symmetric errors in R.A. and Dec are assumed. Individual position errors in image coordinates can be accessed via the X_IMA_ERR and Y_IMA_ERR columns in the output source list.

XID band. The XID energy band ($0.5 - 4.5$ keV) of the X-ray Follow-up & Identification Programme is marked by `ID_BAND`=9 and will be present if the input parameter `withxidband` has been set to true. The keyword `XID_BND` indicates whether XID band information is present in the source table. Note that the energy bands which constitute the XID band have to be specified for each instrument separately using the parameters `xidpndef`, `xidm1def`, `xidm2def`. Default values are the input bands 2, 3, and 4, as defined in `ID_BAND` column.

From v4.42.5 on, an alternative method to treat the XID band has been introduced: With the parameter `xidfixed` set, **emldetect** can be run on one XID band image per instrument using an **emldetect** output list as input source list. Positions and source extent values will be kept fixed, and only fluxes and detection likelihoods are determined. In this case, the input images (science images, exposure maps, background images) for the desired band (e.g. $0.5 - 4.5$ keV) have to be prepared beforehand. The parameters `xidpndef`, `xidm1def`, `xidm2def` determine which energy bands from the input source list are used to provide the *start values* for the fit. Note that when using this method, the parameter `withxidband` should be set to "no", and the parameter `ecf` is used to set the energy conversion factors (see paragraph on ECFs).

Mosaic images. From version 5.0 onward, the parameter `imagebuffersize` is implemented. The main purpose of this parameter is to make the processing of mosaic-pointings more efficient, where the mosaicked sky image will contain large areas without photon data. The value of `imagebuffersize` characterizes the memory that is allocated for each individual image and is given in image pixels: `imagebuffersize` is (at least) the size of the sub-image per pointing that contains non-zero pixels (Fig. 1). Its default is 640, which is the side length of a typical pipeline-produced EPIC image with a bin size of 4 arcsec.

Pile-up estimate. With SAS v18 (**evselect**-3.67 and **emldetect**-7.5), an estimate of the pile-up level of each detection has been introduced (SSC-CEA-TN-1101). It is based on the count rate of the detection and the instrument's read-out time per frame. For point-like sources with extent zero, it is defined as the source count rate times the frame time, divided by an instrument-specific pile-up threshold. For extended sources, it is derived from the source counts per instrument pixel times the frame time, divided by the number of instrument pixels covered by an image pixel and by a pile-up threshold.
At the time of implementation, the pile-up thresholds used are 0.15 cts/frame for EPIC/pn, 1.3 cts/frame for EPIC/MOS, and 0.005 cts/frame/pixel for extended sources.

4 References

Cash, W., Parameter estimation in astronomy through application of the likelihood ratio, ApJ, 228, p. 939 (1979)

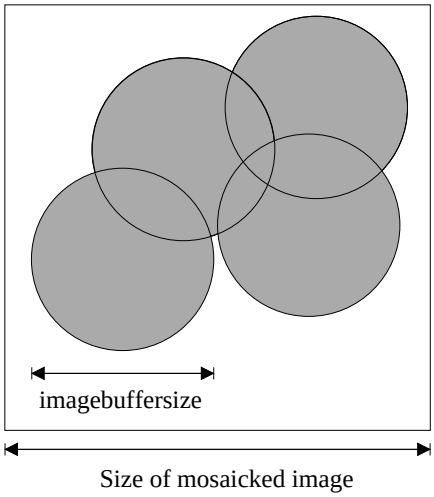

Figure 1: Use of parameter `imagebuffersize`.

Cruddace, R. G., Hasinger, G., Schmitt, J. H., The application of a maximum likelihood analysis to detection of sources in the ROSAT database, in 'Astronomy from large Databases', eds. Murtagh, F. and Heck, A., p. 177 (1988)

Mateos, S., Saxton, R. D., Read, A. M., Sembay, S., Statistical evaluation of the flux cross-calibration of the XMM-Newton EPIC cameras, A&A 496, 879 (2009), e-print arXiv:0901.4026

Rosen, S. R., Webb, N. A., Watson, M. G., et al., The XMM-Newton serendipitous survey. VII. The third XMM-Newton serendipitous source catalogue, A&A 590, A1 (2016), e-print arXiv:1504.07051

Traulsen, I., Schwope, A. D., Lamer, G., et al., The XMM-Newton serendipitous survey. VIII. The first XMM-Newton serendipitous source catalogue from overlapping observations, A&A 624, A77 (2019), e-print arXiv:1807.09178

Traulsen, I., Schwope, A. D., Lamer, G., et al., The XMM-Newton serendipitous survey. X. The second source catalogue from overlapping XMM-Newton observations and its long-term variable content, A&A 641, A137 (2020), e-print arXiv:2007.02932

Webb, N. A., Coriat, M., Traulsen, I., The XMM-Newton serendipitous survey. IX. The fourth XMM-Newton serendipitous source catalogue, A&A 614, A136 (2020), e-print arXiv:2007.02899

5 Parameters

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

Parameter	Mand	Type	Default	Constraints
imagesets	yes	filename list	image.fits	

Names of input EPIC fits images[1,2] or event lists[3] (if `useevents` = true; event-list mode not yet implemented)

boxlistset	yes	filename	eboxlist.fits	

Name of input **eboxdetect** source list

mllistset	yes	filename	emllist.fits	

Name of output **emldetect** source list

withexpimage	no	boolean	true	

Use exposure maps

expimagesets	no	filename list	expimage.fits	

Names of exposure maps[1,2]

withdetmask	no	boolean	false	

If true, only sky pixels inside the detection mask will be used in the PSF fits.

detmasksets	no	filename list	detmask.fits	

Names of detection masks[3]

bkgimagesets	yes	filename list	bkgimage.fits	

Names of background images[1,2]

withsourcemap	no	boolean	false	

Controls whether source maps (one per input image) will be written

sourceimagesets	no	filename list	srcmap.fits	

Names of source maps[3] (one per input image)

mergedlistset	no	filename	mergedlist.fits	

Name of merged source list – obsolete

mlmin	no	float	10.0	[1.0<param<50.0]

Minimum detection likelihood for including a source in the output list

dmlextmin	no	float	10.0	[1.0<param<100.0]

Required likelihood improvement for source extent

scut	no	float	0.9	[0.4<param<100.0]

Source selection radius for multi-source fitting (expressed as fraction of the normalized PSF integrated to the desired radius).[4] Values larger than 1.0 are interpreted as a fixed radius given in units of image pixels.

ecut	no	float	0.68	[0.4<param<100.0]

Source cut-out radius for PSF-fitting (expressed as fraction of the normalized PSF integrated to the desired cut-out radius).[4] Values larger than 1.0 are interpreted as a fixed event cut-out radius given in units of image pixels.

ecf	no	float	1.0	[0.0<param<1000]

Energy conversion factors, given in units of 10^{11} counts cm^2 / erg [1,2]

xidecf	no	float	1.0	[0.0<param<1000]

XID-band energy conversion factors, given in units of 10^{11} counts cm^2 / erg [3]

useevents	no	boolean	false	

Photon mode flag (mode not yet implemented)

fitposition	no	boolean	true	

Fit source positions

fitextent	no	boolean	false	

Fit source extent

fitcounts	no	boolean	true	

Fit source counts (not yet implemented)

fitnegative	no	boolean	false	

Allow fitted count rates to become negative

determineerrors	no	boolean	true	

Determine statistical errors

withoffsets	no	boolean	false	

Flag for reading offsets from eident source list – obsolete

withxidband	no	boolean	false	

Controls whether XID band output will be written

usecalpsf	no	boolean	true	

Read PSF from the calibration database. This parameter is obsolete and has no effect anymore.

extentmodel	no	string	gaussian	gaussian | beta

Model function for source extent

psfmodel	no	string	ellbeta	ellbeta | medium | slew

Model PSF: fully 2d parameterized analytical EPIC PSFs (ellbeta, default from version 5.17.2 on) or medium accuracy PSF for observations in pointing mode; slew for pn observations in slew mode

minextent	no	float	1.5	[0.0<param<300.]

Minimum allowed value for the extent parameter of an extent model in image pixels

maxextent	no	float	20.0	[0.1<param<300.]

Maximum allowed value for the extent parameter of an extent model in image pixels

withhotpixelfilter	no	boolean	false	

If true, the likelihood contribution of the brightest pixel will be ignored (i.e., detections relying on a single pixel will be disregarded).

nmaxfit	no	integer	1	1,10

Maximum number of neighbouring sources to be fit simultaneously (`nmaxfit*nmulsou` ≤ 10)

nmulsou	no	integer	1	1,3

Allow fit to split up one input source in maximum nmulsou sources (`nmaxfit*nmulsou` ≤ 10)

pimin	no	integer	2000	[0<param<30000]

Lower energy boundaries of images; units: eV; one value per input image and up to six energy bands per instrument.

pimax	no	integer	4500	[0<param<30000]

Upper energy boundaries of images; units: eV; one value per input image and up to six energy bands per instrument.

hrpndef	no	integer	1 2 2 3 3 4 4 5	0,10

Array of up to eight indices (integer) specifying the upper and lower energy band for each of the hardness ratios for PN; i.e. two numbers per energy band.

hrm1def	no	integer	1 2 2 3 3 4 4 5	0,10

Array of up to eight indices (integer) specifying the upper and lower energy band for each of the hardness ratios for MOS1; i.e. two numbers per energy band.

hrm2def	no	integer	1 2 2 3 3 4 4 5	0,10

Array of up to eight indices (integer) specifying the upper and lower energy band for each of the hardness ratios for MOS2; i.e. two numbers per energy band.

xidpndef	no	integer	2 3 4	0,10

Index of the energy band(s) from which the images / start values for the XID band are taken for PN

xidm1def	no	integer	2 3 4	0,10

Index of the energy band(s) from which the images / start values for the XID band are taken for MOS1

xidm2def	no	integer	2 3 4	0,10

Index of the energy band(s) from which the images / start values for the XID band are taken for MOS 2

xidfixed	no	boolean	false	

Run **emldetect** on XID-band image with positions and source extent fixed to input values

withthreshold	no	boolean	false	

Allow splitting up into multi-PSF fitting only for sources above threshold

threshold	no	float	20	[param>0.0]

Value of threshold for multi-PSF fitting

threshcolumn	no	string	LIKE	LIKE | SCTS | RATE

Column in input list on which `threshold` will be applied

withtwostage	no	boolean	false	

Use two-stage process for multi PSF (`nmulsou` > 1) fitting

imagebuffersize	no	integer	640	100<param<10000

Parameter that controls memory requirements for mosaic images.

withimagebuffersize	no	boolean	no	

Allow user-defined values of `imagebuffersize`.

[1] Space-separated list, sorted by instrument and energy band. I.e., energy band one to energy band n of instrument one is followed by energy band one to n of instrument two.
[2] One per instrument per energy band; space-separated list, maximum of six energy bands per instrument.
[3] One per instrument; space-separated list.
[4] A parameter value of 1.0 would thus correspond to an infinite radius.

6 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

MissingParameter *(error)*
Missing input file name

WrongInst *(error)*
Unknown instrument

FileMismatch *(error)*
Inconsistent number of input images

FileMismatch *(error)*
Inconsistent instruments or bands

FileMismatch *(error)*
detector masks /= # instruments

FileMismatch *(error)*
Wrong detector mask

WrongType *(error)*
Input image has wrong type

WrongRefPixel *(error)*
Reference pixel is outside FOV

WrongParam *(error)*
ERROR in cut_radius

WrongPSFModel *(error)*
psfmodel=slew is only valid for EPIC/pn.

noPSF *(error)*
point response not valid

notEnoughMemory *(error)*
Not enough memory available to allocate arrays

TooManyBands *(error)*
Images and / or `pimin`, `pimax` parameters for more than six energy bands provided.

TooManyInstruments *(error)*
Number of instruments times number of pointings is limited.

TooManyPhotons *(error)*
Cannot handle so many photons in the source region. Consider decreasing the number of input images.

EmptySourceList *(warning)*
Input source list is empty
corrective action: Exit without output results

FileMismatch *(warning)*
Number of PI boundaries not equal number of images
corrective action: Use default values for missing boundaries

FileMismatch *(warning)*
Number of ECFs not equal number of images
corrective action: Use default values

MissingAttribute *(warning)*
Keyword is missing
corrective action: Keyword is not copied to output

MissingColumn *(warning)*
Column not found in the input source list.
corrective action: Column ignored. In particular: Input source list won't be sorted by decreasing likelihood, if none of the columns `LIKE`, `DET_ML`, `SIGMA`, or `WCORR` are available.

WrongPointDir *(warning)*
Pointing direction is outside of image
corrective action:

BufferOverflow *(warning)*
More than 50000 sources detected
corrective action: Remaining sources will be ignored

7 Input Files

1. PPS product (from task **evselect**): FITS images (one per instrument per energy band if program is run in imaging mode - default)
2. PPS product (from task **evselect**): EPIC IMAGING-mode event lists (one per instrument if program is run in Photon mode – not yet implemented)
3. PPS product (from task **eboxdetect** run in map-detect mode): EPIC **eboxdetect** source list
4. PPS product (from task **eexpmap**, optional): EPIC exposure images (one per instrument)
5. PPS product (from task **esplinemap**, optional): Spline background images (one per instrument per energy band)
6. PPS product (from task **emask**, optional): Detection masks (one per instrument)

8 Output Files

1. PPS product (to be read by task **srcmatch**): EPIC **emldetect** source list
2. Optionally: source maps (one output image per input image)

Table 2: Columns of the output source table

ML_ID_SRC	**emldetect** source number
BOX_ID_SRC	corresponding **eboxdetect** input source number
ID_INST	instrument ID; 1: PN, 2: MOS1, 3: MOS2; 0: summary row
ID_BAND	energy band number (band number 0: summary band)
ID_CLUSTER	cluster id; sources fit simultaneously have same number
SCTS	source counts
SCTS_ERR	source counts error
X_IMA	source image pixel X coordinate
X_IMA_ERR	error of image pixel X coordinate
Y_IMA	source image pixel Y coordinate
Y_IMA_ERR	error of image pixel Y coordinate
EXT	source extent, gaussian sigma or beta model core radius (image pixel)
EXT_ERR	extent error
DET_ML	likelihood of detection
EXT_ML	likelihood of extent
BG_MAP	background at source location (counts/pixel)
EXP_MAP	exposure, PSF-weighted mean of the subimages around the source (seconds, vignetting corrected)
FLUX	source flux (cgs units)
FLUX_ERR	source flux error
RATE	source count rate (counts/sec)
RATE_ERR	count rate error
RA	source right ascension (degrees)
DEC	source declination (degrees)
RADEC_ERR	combined R.A.-Dec. error (arcsec)
LII	source galactic longitude (degrees)
BII	source galactic latitude (degrees)
RAWX	raw X source coordinate
RAWY	raw Y source coordinate
OFFAX	off-axis angle (arcsec)
CCDNR	chip number
HRi ($1 \leq i \leq 4$)	hardness ratios 1..4
HRi_ERR ($1 \leq i \leq 4$)	hardness ratio error
CUTRAD	source cut out radius
MASKFRAC	PSF weighted on-chip fraction
EFF	encircled energy fraction
VIGNETTING	vignetting
ONTIME	Integration time of the CCD, not vignetting corrected. Set to NULL, if CCD no. is not defined (i.e. source center on bad pixels, gaps, damaged/noisy CCDs)
PILEUP	pile-up level of the detection in the active instrument configuration
DIST_NN	distance to nearest neighbour (arcsec)
FLAG	quality flag placeholder (to be set by **dpssflag**)

See also
http://xmmssc.irap.omp.eu/Catalogue/4XMM-DR12/Coordinates.html
http://xmmssc.irap.omp.eu/Catalogue/4XMM-DR12/col_srcpar.html

9 Algorithm

```
subroutine emldetect

Read in EBOXDETECT source list (map detect) and
  sort by source count rate

  Loop over sorted source list (begin with brightest source):

    Selection of sources for simultaneous multi-source fitting:
      1) Search close neighbours within source cut radius of
         current source. Don't consider sources which have been
         processed already; mark selected sources as processed.
      2) Repeat (1) for each close neighbour until the maximum
         number of sources for simultaneous fitting (max. 8; specified
         in parameter file) is reached. I.e., the selection of additional
         sources for the multi-source fitting terminates when either the
         maximum number of sources (parameter max_fit) is reached or
         when no additional sources fulfil the distance criterium
         (parameter scut).

    Determine data area (2d-mask array) to be used for multi-source fitting:
      Loop over mask array: Set to 1 if pixel is (a) within event cut
      radius of selected source and (b) within the area marked in the
      detection mask; set to 0 otherwise.

    Read in data:
    Fill (x, y, count) data records:
       Binned mode:
          x,y:   image pixel coordinates
          count: number of events in pixel
       Single photon mode:
          x,y:   event coordinates
          count:=1

    Feed data records, background maps, and exposure maps
      into Maximum Likelihood PSF fitting algorithm:

    Use Marquarth-algorithm to minimize likelihood function of
      multi-source PSF fit.

    Remove sources from fit which do not significantly improve
      goodness of fit (required improvement in likelihood specified
      by parameter mlmin).

    Set source extent to 0 if extent does not significantly improve
      goodness of fit (required improvement in likelihood specified
      by parameter dmlextmin).

    Add best-fit source models (PSF + source extent) to background
      maps. I.e., sources which have already been fitted by the program
      are treated as background for the remaining sources. Note that
      the sources are processed in the order of decreasing count rate
```

```
      such that all the bright sources will have been modeled into the
      background map once the weak sources are processed.

    If likelihood of detection exceeds threshold THEN
      Write source parameters to EMLDETECT source list.
      Add fluxes and hardness ratios.
    END IF

  END Loop

end subroutine emldetect
```

10 Comments

Due to coding error the likelihood values `DET_ML` and `EXT_ML` computed by **emldetect** versions 4.24 and older were overestimated by a factor 2 or more. From version 4.27 on, the correct values are computed. Note that this change strongly reduces the number of spurious detections at a certain likelihood threshold.

With introducing the analytical 2d PSF, parts of the algorithm for fitting the source positions have been changed, increasing the accuracy of image-pixel and sky-coordinate positions. Therefore, source positions determined by emldetect runs with `psfmodel`=ellbeta and positions determined with `psfmodel`=medium can differ from version 5.15.4 on.

11 Future developments

Photon mode still needs to be implemented.

References