XRISM Xtend Anomalous Pixels Clipper (xtdpixclip)
Special Topic Guide STG002
Version 1.0

XRISM Science Data Center

Last updated: January 6, 2026

Contents

1 Introduction

This guide is designed to get you started with using the software tool Xtend Anomalous Pixels Clipper (`xtdpixclip`). You should read the help for `xtdpixclip` for more information. If you get stuck, e-mail the SDC help desk (`XRISM-SDC-help@lists.nasa.gov`). Reading the *XRISM Quick-Start Guide* is also recommended.

Xtend data, even after screening may suffer from pixels that have anomalously high counts due to non-source events. These events are most often due to flickering pixels or cosmic ray echo events. These anomalous pixels should not be included for data analysis purposes. Instead, their information should be passed along to the exposure map tool (`xaexpmap`), which in turn informs corrections to the effective area (using the `xaarfgen` task). The `xtdpixclip` tool enables the use of multiple simple counts-per-pixel thresholds in different, arbitrary user-defined regions, to flag pixels as anomalous. The `xtdpixclip` tool should be used to handle Xtend flickering pixels instead of the `searchflickpix` tool, as it avoids the problems that `searchflickpix` suffers from.

Using the `xtdpixclip` tool is a three-step process which involves running the tool twice. In the first step the tool is run in histogram mode (`pmode=histo`). This mode will generate 1D histograms of the number of pixels vs. the counts per pixel for all of the pixels inside of user-specified inclusion regions. The tool generates histogram data in FITS file format, as well as plots of the histograms. The inclusion regions, and the optional exclusion regions, should be specified in the form of SAO region files. A separate histogram will be generated for each inclusion region file specified. Multiple region files can be specified and each file may contain multiple regions. The inclusion region files may contain lines for exclusion regions in order to construct complex shapes. There is an exclusion region file provided in the HEASoft package that will exclude the Xtend calibration sources. This file ($HEADAS/refdata/calsrc_XTD_det.reg) can optionally be added to your list of exclusion region files.

The next step is to use these histograms to select counts per pixel thresholds to be used in step three. Because each inclusion region file gets its own histogram, it is possible to select different count thresholds for each set of inclusion regions. In some instances there may be an obvious gap in the histogram between a large number of pixels with lower counts/pixel and a small group of pixels with a much higher counts/pixel. In instances like this it is likely that the small number of pixels with a very high counts/pixel value are anomalous. Selecting a threshold value inside of this gap would be a reasonable starting point. In other instances there may not be an obvious place to set a threshold. In cases like this, it's best to select a conservative threshold value and iteratively lower the value while checking the output to ensure that source counts are not being removed. Remember that the goal of this tool is to flag pixels with anomalously high counts not due to the source. This threshold is the boundary where pixels with higher counts are flagged as anomalous.

In step three the tool is run again, but this time in apply mode (`pmode=apply`). In this mode, each selected threshold will be applied to the corresponding inclusion region. Any pixel with counts exceeding the threshold for the region it is inside of will be flagged as anomalous. If a single inclusion region file contains multiple inclusion regions the same threshold will be applied to all of them. This mode will output a flagged pixel file (.fpix file), and will optionally output a separate cleaned event file that has events in flagged pixels removed.

Additionally, the energy range for making histograms and selecting and applying cut-off thresholds can be adjusted using the `emin` and `emax` parameters. The whole energy range goes from 0.0 keV to 24.575 keV, which corresponds to the 4096 PI channels with 0.006 keV per PI channel. The default energy range is from 0.0 keV to 12.0 keV. Events with energy outside of the energy range defined by `emin` and `emax` will not be included in the histograms. Additionally, they will not be counted when checking the pixel counts against the threshold value. A narrower energy range might be useful for testing if the anomalous pixels have a strong energy dependence. Even though the threshold selection and anomalous pixel flagging is done with the restricted energy range (emin to emax), once a pixel is flagged as anomalous, the output cleaned event file will not contain events of any energy that originated in an anomalous pixel. Likewise, events of all energies will appear in the output cleaned events file if they originate in pixels that are not flagged as anomalous.

2 What You Will Need

Here, we describe the set-up & files that you will need. Documentation on how to set up the items below can be found elsewhere.

1. Installation of the HEASOFT software, version 6.35 or later.

The `xtdpixclip` tool does not use the CalDB.

2.1 Important Information

- The `xtdpixclip` parameter, 'gtifile', if not set to 'NONE', is the name of any standard FITS GTI file and extension number. Time interval files made by `xselect` (e.g., with cursor selection) are accepted. However, note that the purpose of this GTI file is simply a convenient method to ignore data in the input event file by only selecting events that lie inside one of the GTI. The rejected data are not used to construct histograms, nor are the rejected data passed onto any cleaned event file. The input event file is simply filtered by the GTI, and the rejected events are discarded. The filtered events are saved in a new file for reference or for other usage (see help file for the naming conventions used for `xtdpixclip` output files).
- The input region files to `xtdpixclip` are required to have standard SAO region file format. That format requires that lines in a region file that describe exclusion shapes begin with a "minus" symbol

("-"). This is true for *all* region files that are input into the `xtdpixclip` tool, even if a region file is explicitly marked as an exclusion region file by including it in the list of exclusion files contained in the setting for the parameter 'excregionfiles'.

- The SAO region files input into `xtdpixclip` must be made from Xtend images in DET coordinates. When using `ds9` to make the region files, select "image" or "physical" (not "detector") from the drop-down menu when saving a region file.
- Even though exclusion regions can be specified within *inclusion* files for input to `xtdpixclip`, the list of inclusion files and the list of exclusion files for input to `xtdpixclip` serve very different purposes. Each inclusion file is associated with a counts-per-pixel threshold value (specified as a list of thresholds by the input parameter 'thresholds'). Exclusion files are not associated with thresholds. The main purpose of exclusion regions in an *inclusion* file is to enable construction of custom, complex *inclusion* shapes.

3 Walk-through Examples

There are two sequences used in this document, 000145000 and 300014010. These sequences provide examples of a point and extended source respectively. Substitute the sequence number of your observation in the commands and file names in the following commands. The techniques for point sources and extended sources are essentially the same, except that extended sources will generally require a more complex treatment of regions. Also, sequence 000145000 has data in 1/8 window mode, whereas sequence 300014010 has data in full window mode. The latter has a larger detector area and, unlike 1/8 window mode, includes the calibration sources in the field.

The `xtdpixclip` tool will generally be run on the cleaned event file that is an output from the pipeline.

Note that due to the wobble of the satellite, images in sky coordinates show anomalous pixels much more prominently than images in DET coordinates. Because of this, the images are also provided using sky coordinates to better see the anomalous regions in the images. However, the true, physical anomalous pixels are in the DET coordinate images, because they are fixed to the detector.

3.1 ObsID 000145000 (3C 273)

3.1.1 Step 1: Select regions and generate histograms

An image from the input file (xa000145000xtd_p031100010_cl.evt) can be seen in Figure 1. While examining the image from this input file in ds9, inclusion, and optionally exclusion, regions can be created. These regions should be saved as ds9 format in physical coordinates. Any ds9 shape can be used, and multiple regions can be saved in a single file. If multiple regions are listed in a single file, the threshold that corresponds to that file will be applied to all of the regions it contains.

In this example three separate inclusion regions are selected, and these can be seen in Figure 1. The regions are listed in different files so that a histogram is generated for each one, and therefore a unique threshold can be selected for each one. One inclusion region is selected to cover the point source. This allows the tool to generate a histogram of the events that only take place inside of that region. Two other inclusion regions are also selected, one above and one below the source, because there are pixels in these regions with non-negligible event counts far from the source. The contents of the three region files can be seen in Figure 2.

Now that the regions files are prepared, the rest of the parameters for the histogram run can be selected. In order to run the tool in histogram mode we set `pmode=histo` (case insensitive). The parameter 'outroot' determines the names of any output files. For histogram mode the output files include histogram fits files with the name 'outroot'_xpc_hist.fits, and histogram plots with the names 'outroot'_xpc_regNN_hist.gif, where NN $= 01, 02, 03, ...$ up to the total number of inclusion regions files provided. The parameter 'exregionfiles' will be left at its default value 'NONE', because no exclusion regions are needed. Additionally, the parameter 'nhistbins' is set to its default (as of HEASoft 6.36) of -1. This sets the number of histogram bins to the maximum number of counts, which means the histograms will have one bin per count increment. The

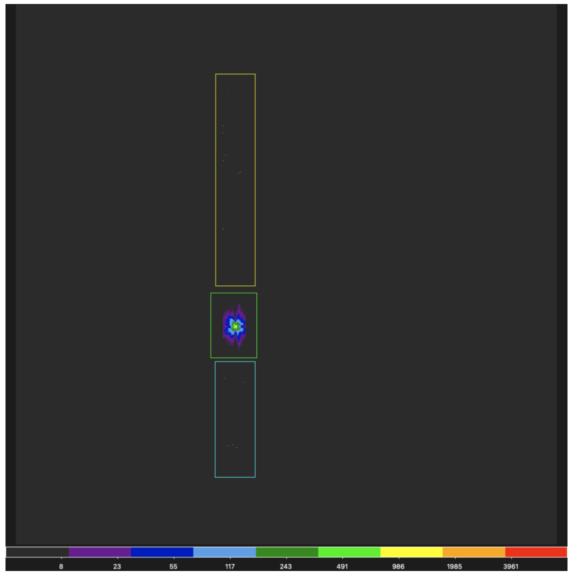

Figure 1: For ObsID 000145000, this is the image corresponding to the cleanded event file that is input into the xtdpixclip tool. There are three inclusion regions overlaid on the image. The yellow and cyan box regions cover non-source areas of the image with boundaries that extend almost to the edges of the chips. These regions may need a different threshold than the source region covered by the green box.

parameter ‘emax’ is not left at its default value of 12.0 keV, it is set to 24.575 keV instead. The parameter ‘emin’ on the other hand is left at its default of 0.0 keV. These values are chosen to cover the full energy range, as there are 4096 PI channels and 0.006 keV per channel. If we find that a threshold is not readily identifiable, we could try narrowing the energy range (for example to a soft energy band) to see if it makes a difference. Events with energy outside of the range emin to emax are not included in the histogram. The command used to do the histogram run can be seen below.

```
punlearn xtdpixclip

xtdpixclip evtinfile="xa000145000xtd_p031100010_cl.evt" outroot="000145000_run1" pmode=histo
nhistbins=-1 incregionfiles="1_top.reg,1_bottom.reg,1_src.reg" emax=24.575 chatter=2 clobber=yes
```

3.1.2 Step 2: Select the thresholds

Examining the histograms in Figure 3, some possible thresholds can be chosen. The histograms that correspond to the non-source regions can both have a large number of pixels with very low counts (<10), but also a number of pixels with high counts on the order of 1000. Both of these regions have very few pixels with counts between these two count ranges. Meanwhile, the histogram of the pixels in the source region shows an expected distribution of pixel counts between those same two count ranges. A reasonable choice of threshold for this source region might be 2000. This will preserve all of the pixels with fewer than 2000 counts in the source region and assume that they are not anomalous. That same threshold count could be applied to the non-source regions. The problem with using this threshold is that then there will be some pixels with counts between 10 and 1000 that might be anomalous but will be passed on to the cleaned event file. Instead, the pixels in these two regions can be clipped with a much lower threshold than can be used in the source region. For this example 10 will be used.

```
# Region file format:  DS9 version 4.1
global color=green dashlist=8 3 width=1 font="helvetica 10 normal roman" select=1
highlite=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
physical
box(735.766,1222.0195,130.392,708.101,0) # color=yellow
```

```
# Region file format:  DS9 version 4.1
global color=green dashlist=8 3 width=1 font="helvetica 10 normal roman" select=1
highlite=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
physical
box(729.4275,735.766,153.935,217.32,0) # color=green
```

```
# Region file format:  DS9 version 4.1
global color=green dashlist=8 3 width=1 font="helvetica 10 normal roman" select=1
highlite=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
physical
box(733.955,420.652,134.014,387.554,0) # color=cyan
```

Figure 2: Region files for ObsID 000145000. From top to bottom: 1_top.reg, 1_src.reg, 1_bottom.reg.

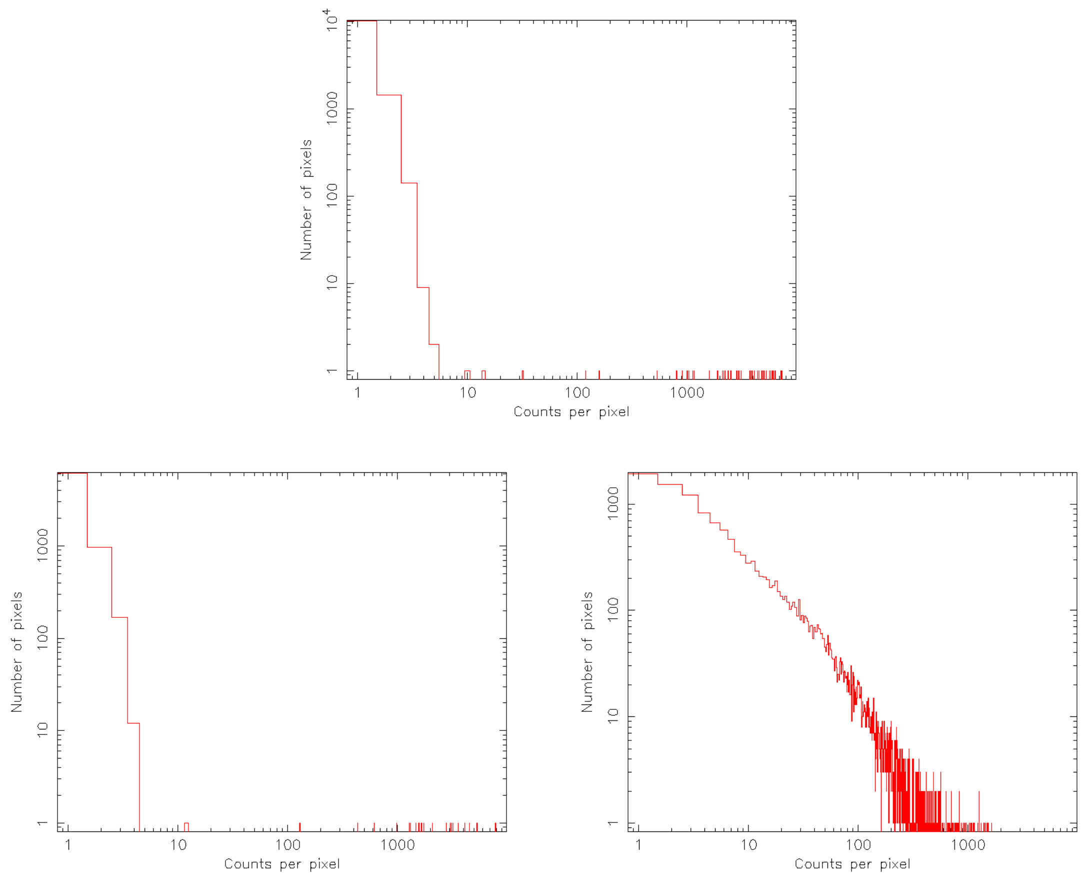

Figure 3: Histogram of number of pixels vs. counts per pixel for each inclusion region for the example with ObsID 000145000. **top)** Top yellow box, selected threshold 10 counts/pixel. **left)** Bottom cyan box, selected threshold 10 counts/pixel. **right)** Green source box, selected threshold 2000 counts/pixel.

3.1.3 Step 3: Apply the thresholds

The next step after selecting the thresholds is to apply them. Note that now the tool is being run again, but this time with `pmode=apply`. The same inclusion region files are being used as well. For this run the thresholds parameter is specified as the set of values determined in the previous step. It is important to list the threshold values in the same order as the corresponding inclusion region file, because those arguments are positional. The first threshold value, 10, goes with the first inclusion region file. The last threshold value, 2000, goes with the last inclusion region file that covers the source region.

```
punlearn xtdpixclip
```

```
xtdpixclip evtinfile="xa000145000xtd_p031100010_cl.evt" outroot="000145000_run1" pmode=apply
thresholds="10,10,2000" incregionfiles="1_top.reg,1_bottom.reg,1_src.reg" emax=24.575 chatter=2
clobber=yes
```

After the thresholds are applied, the tool outputs two files (assuming the mkclean parameter's value is yes, which is the default). The file ending in "_rmvpix.fpix" is the flagged pixel file. The first extension of this file contains the list of events from all of the pixels flagged as anomalous. The second extension contains the list of the pixels themselves, including their coordinates. The other output file is the cleaned events list ending in "_clnevt.fits". This file contains all of the events that were left after the anomalous pixels were removed.

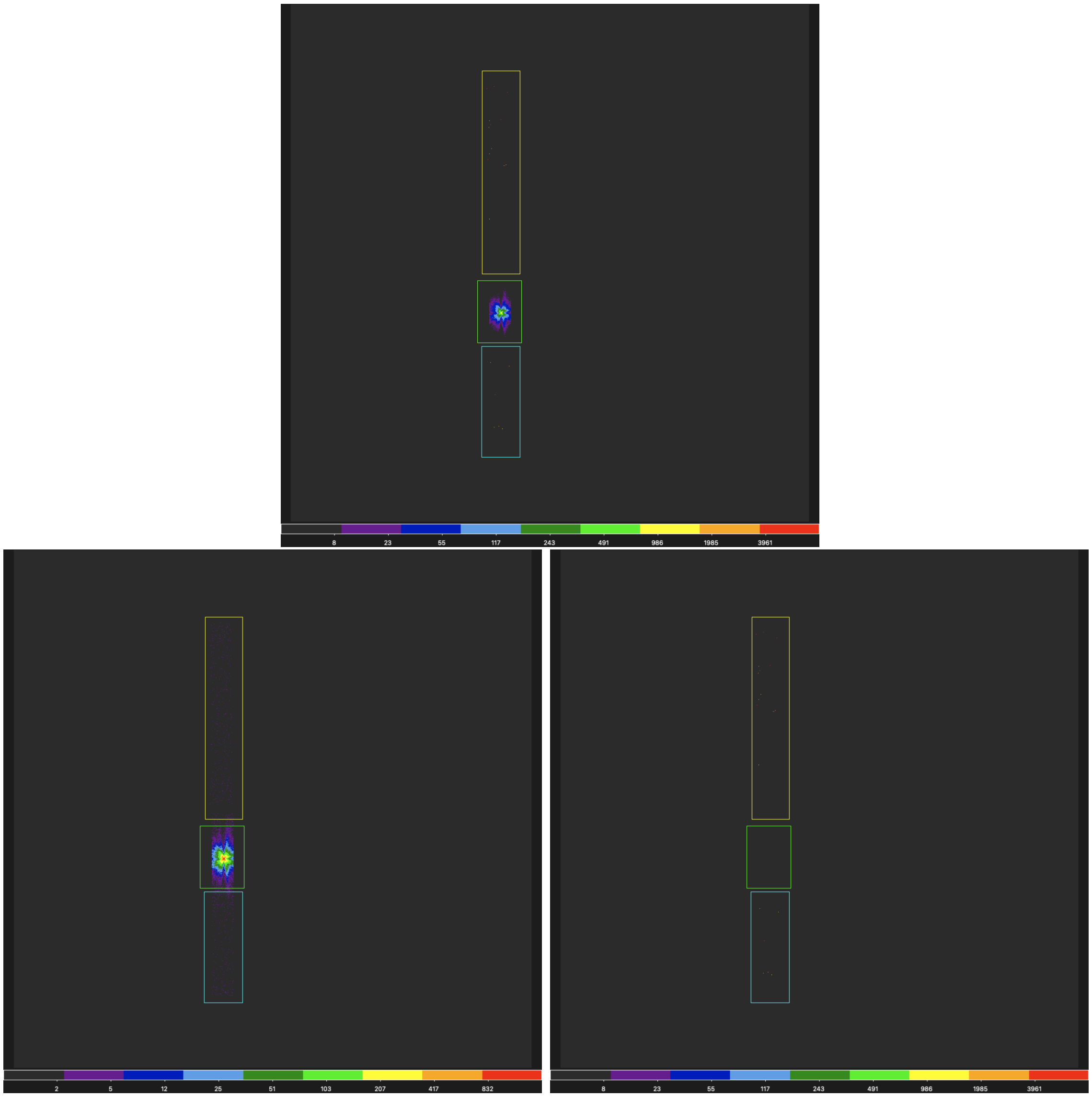

Figure 4: Images from input and output data in DET coordinates for ObsID 000145000. **top)** The input to xtdpixclip, note the events far away from the source. **left)** The output cleaned event file. This has the resulting events after the anomalous pixels were removed. **right)** The output flagged pixel file. This contains all of the events that take place inside of the anomalous pixels.

The output of the application step can be seen in Figure 4. The anomalous pixels in the image can be hard to make out, so the same images are shown in Figure 5, but using sky coordinates where the anomalous pixels are much easier to see. All of the pixels with counts over each region's threshold have been flagged and added to the flagged pixel file, while the rest have been passed on to the cleaned event file.

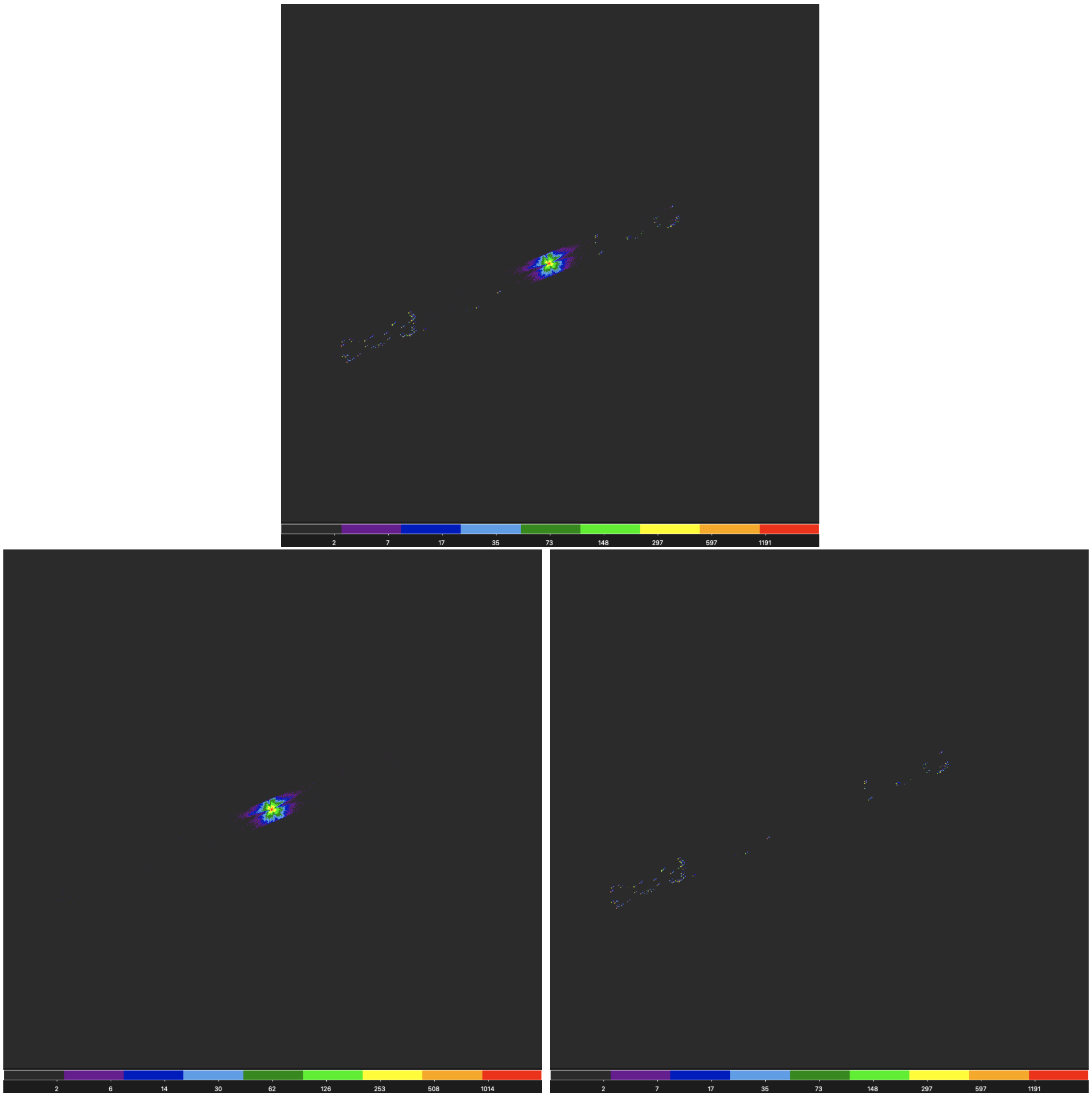

Figure 5: Images from input and output data in SKY coordinates for ObsID 000145000. **top)** The input to xtdpixclip, note the events far away from the source. **left)** The output cleaned event file. This has the resulting events after the anomalous pixels were removed. **right)** The output flagged pixel file. This contains all of the events that take place inside of the anomalous pixels.

3.2 ObsID 300014010 (M87)

This example will follow the same three-step process outlined in subsection 3.1: generate histograms, select thresholds, apply those thresholds.

3.2.1 Step 1: Select regions and generate histograms

This example uses a slightly more complicated set of region files than the previous example. There are four inclusion region files which means that there will be four histograms generated. These regions can be seen in Figure 8. An important detail to note is that the white box region in the middle is an inclusion region along with a circular exclusion region over the source, *in the same inclusion region file*. There is also a circular inclusion region over the source that is identical in size to the exclusion region that is already over the source.

This structure of regions allows for overlapping regions with different thresholds. In particular, it separates out a circle around a source for its own threshold, and a rectangular region around a source that doesn't include the source, for its own threshold. If there was no exclusion region over the source, then you would have to take care that the files and thresholds are in an order such that the proper threshold is applied. This is because the threshold of the first inclusion region a pixel is found to be inside of is what will be applied to the pixel. As such, if you have overlapping regions and want the lowest threshold to be applied to any pixel in the overlapping area you must input the files and thresholds in order from lowest threshold to highest.

This example also makes use of an exclusion region file for the calibration sources, refdata/calsrc_XTD_det.reg that was mentioned in section 1. The two elliptical regions colored red in Figure 8 cover the calibration source regions of the detector. All of the regions used are listed in Figure 6.

```
# Region file format:  DS9 version 4.1
global color=green dashlist=8 3 width=1 font="helvetica 10 normal roman" select=1
highlite=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
physical
box(935.88149,1321.6245,1410.769,512.51301,0) # color=yellow
```

```
# Region file format:  DS9 version 4.1
global color=green dashlist=8 3 width=1 font="helvetica 10 normal roman" select=1
highlite=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
physical
box(938.598,321.9525,1412.58,222.753,0) # color=cyan
```

```
# Region file format:  DS9 version 4.1
global color=green dashlist=8 3 width=1 font="helvetica 10 normal roman" select=1
highlite=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
physical
box(938.598,746.632,1412.58,604.874,0) # color=white
-circle(739.388,741.199,284.70109) # color=green
```

```
# Region file format:  DS9 version 4.1
global color=green dashlist=8 3 width=1 font="helvetica 10 normal roman" select=1
highlite=1 dash=0 fixed=0 edit=1 move=1 delete=1 include=1 source=1
physical
circle(739.388,741.199,284.70109) # color=green
```

```
physical
-ellipse(918, 264,87,95,0)
-ellipse(919,1534,87,95,0)
```

Figure 6: Region files for ObsID 300014010. From top to bottom: 2_top.reg, 2_bottom.reg, 2_src_box.reg, 2_src.reg, calsrc_XTD_det.reg.

After selecting the region and setting up the region files, the histograms can be generated.

```
punlearn xtdpixclip
```

```
xtdpixclip evtinfile="xa300014010xtd_p0300000a0_cl.evt" outroot="300014010_run2" pmode=histo
nhistbins=-1 incregionfiles=2_top.reg,2_bottom.reg,2_src_box.reg,2_src.reg
```

```
excregionfiles=$HEADAS/refdata/calsrc_XTD_det.reg emax=24.575 chatter=2 clobber=yes
```

3.2.2 Step 2: Select the thresholds

Using the histograms shown in Figure 7, reasonable thresholds can now be selected. The thresholds should exclude the few anomalous pixels with high counts per pixel, without removing the pixels with real event counts. The threshold values selected are listed in Figure 7.

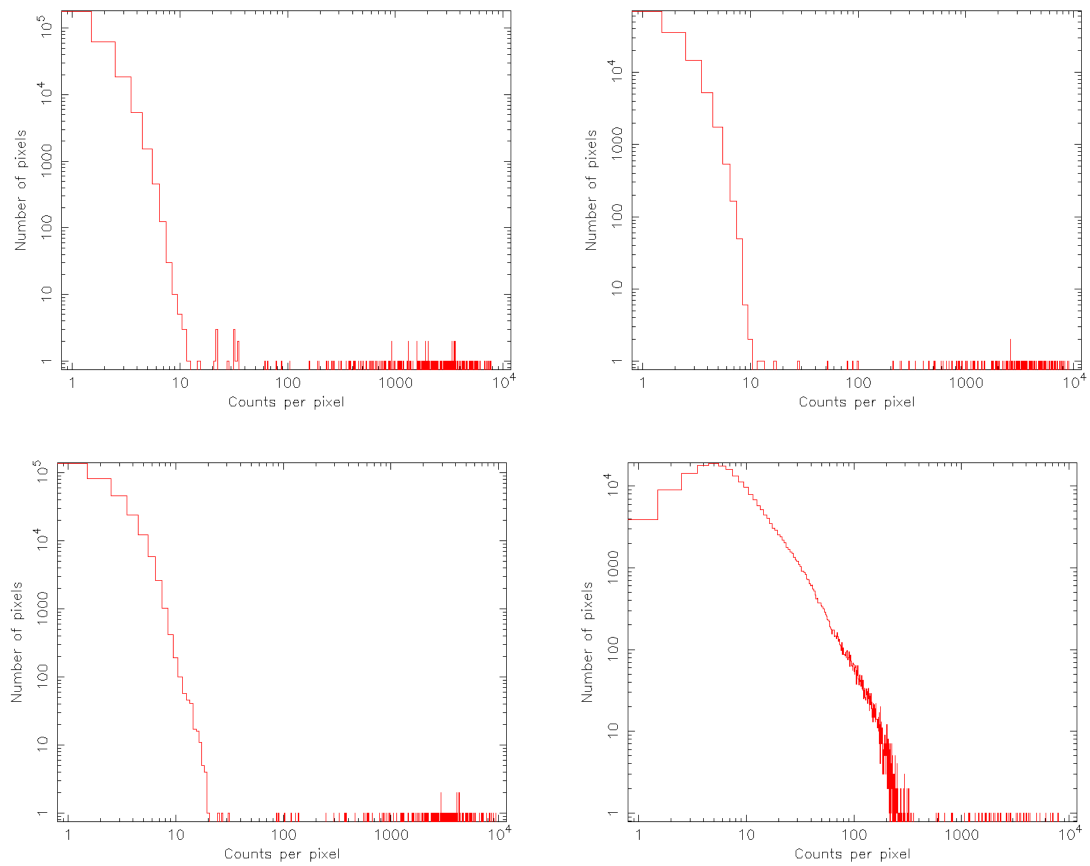

Figure 7: Histogram of number of pixels vs. counts/pixel for each of the four inclusion regions for ObsID 300014010. **top left)** Top yellow box, selected threshold 20 counts/pixel. **top right)** Middle white box excluding the circular region, selected threshold 10 counts/pixel. **bottom left)** Bottom cyan box, selected threshold 30 counts/pixel. **bottom right)** Green source circle, selected threshold 500 counts/pixel.

3.2.3 Step 3: Apply the thresholds

Now the thresholds can be applied and the results (Figure 8) examined. The anomalous pixels in the image can be hard to make out, so the same images are shown in Figure 9, but using sky coordinates where the anomalous pixels are much easier to see.

```
punlearn xtdpixclip
```

```
xtdpixclip evtinfile="xa300014010xtd_p00000a0_cl.evt" outroot="300014010_run2" pmode=apply
thresholds="20,10,30,500" incregionfiles="2_top.reg,2_bottom.reg,2_src_box.reg,2_src.reg"
excregionfiles=$HEADAS/refdata/calsrc_XTD_det.reg emax=24.575 chatter=2 clobber=yes
```

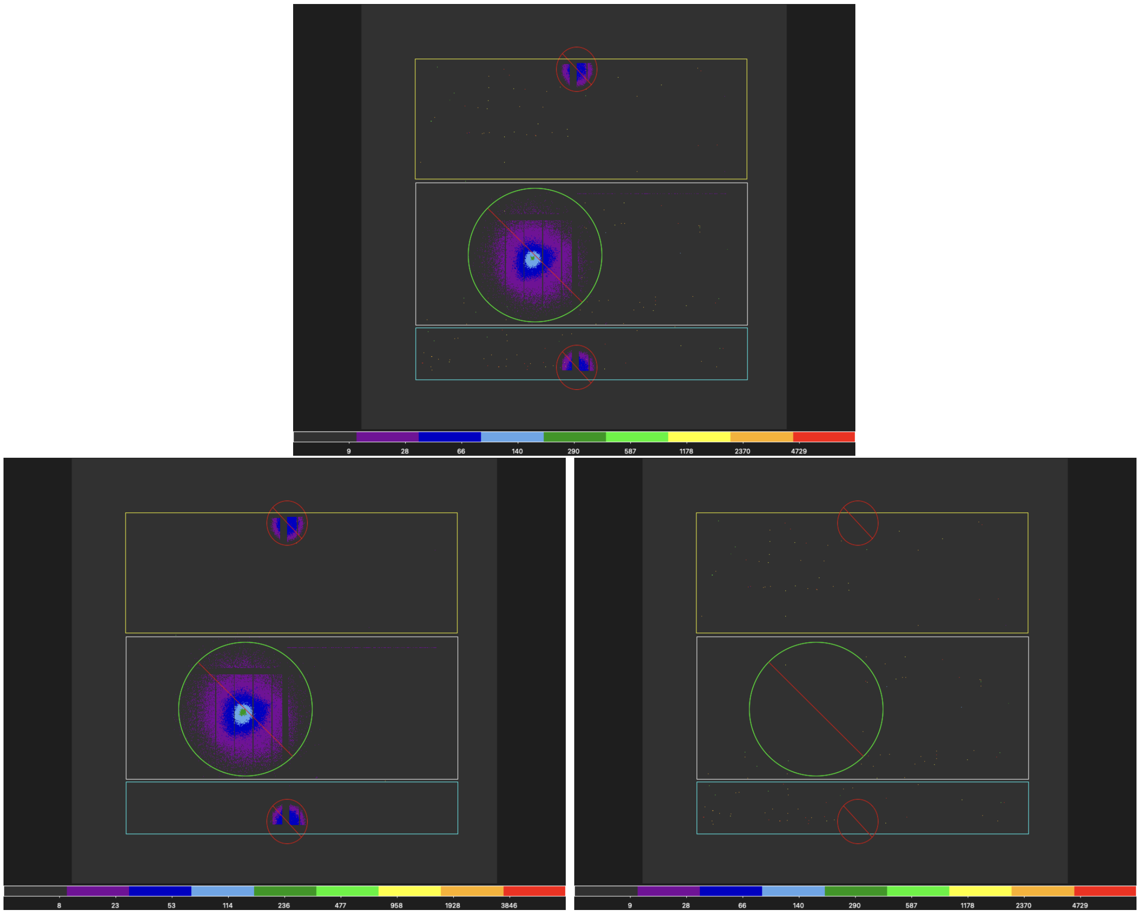

Figure 8: Images from input and output data in DET coordinates for ObsID 300014010. **top)** The input to xtdpixclip, note the events far away from the source. **left)** The output cleaned event file. This has the resulting events after the anomalous pixels were removed. **right)** The output flagged pixel file. This contains all of the events that take place inside of the anomalous pixels.

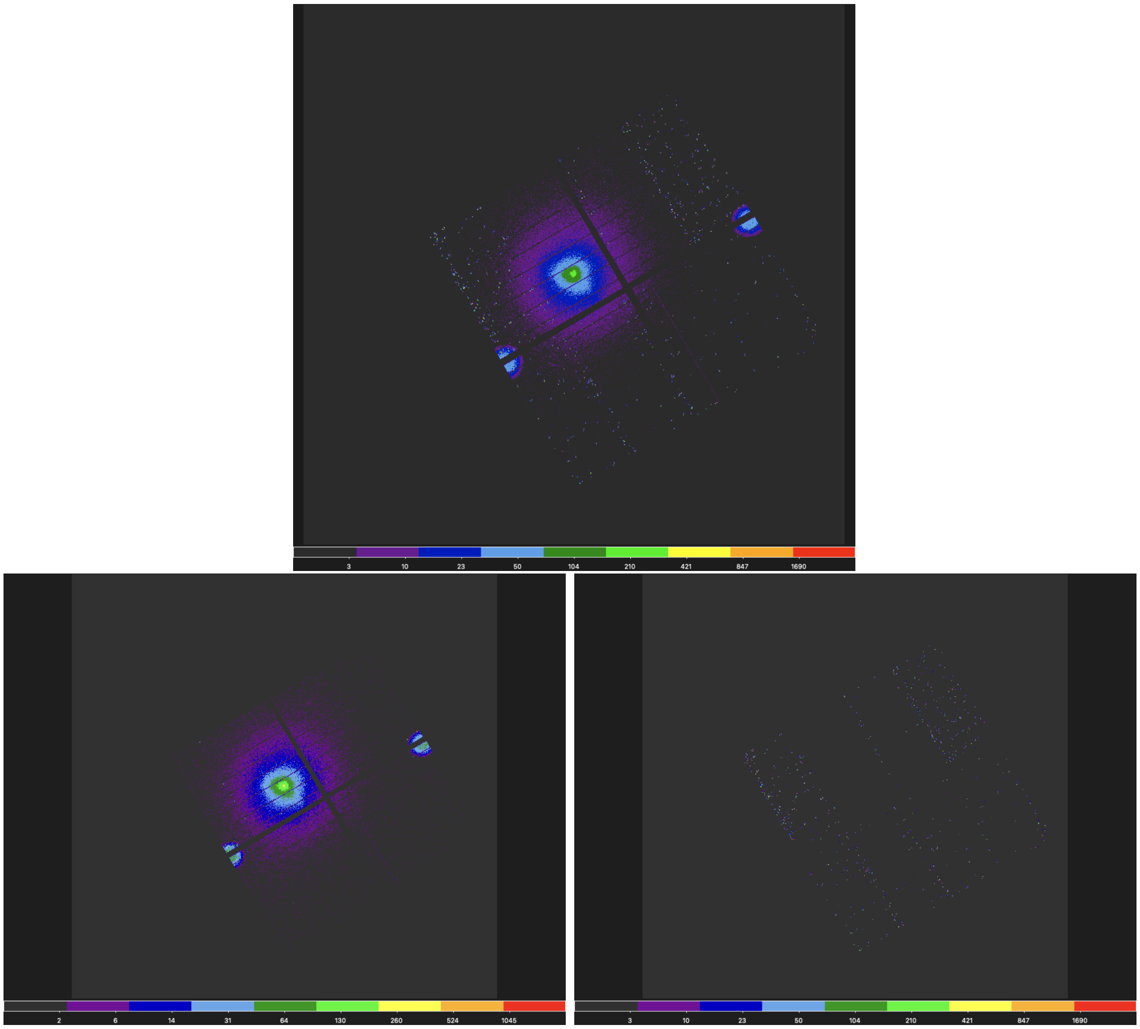

Figure 9: Images from input and output data in SKY coordinates for ObsID 300014010. **top)** The input to xtdpixclip, note all of the events outside of the source. **left)** The output cleaned event file. This has the resulting events after the anomalous pixels were removed. **right)** The output flagged pixel file. This contains all of the events that take place inside of the anomalous pixels.