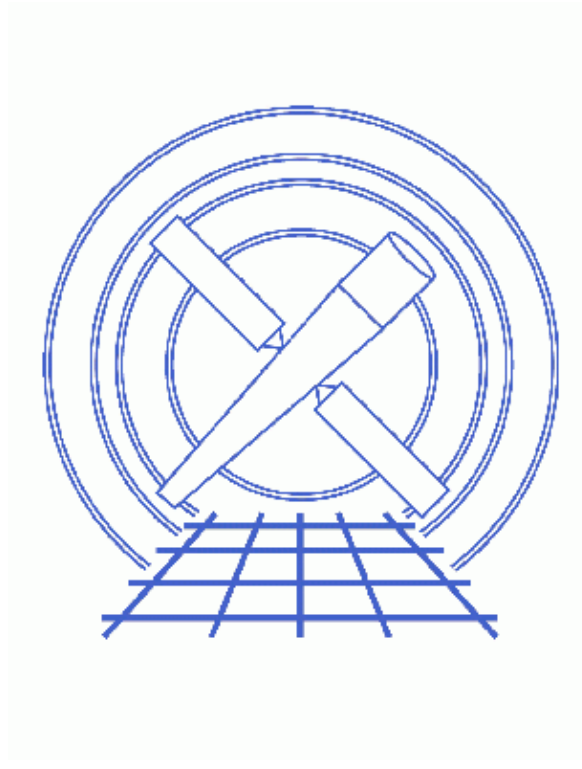


# Correcting Absolute Astrometry with `reproject_aspect`



## ***CIAO 3.4 Science Threads***

# Table of Contents

- *Get Started*
- *Creating Source Lists (wavdetect)*
- *Running reproject aspect: Which Method to Use*
- *Method 1: Create a New Aspect Solution*
  - ◆ Run reproject aspect
  - ◆ Compare the original and updated files
  - ◆ Apply the new aspect solution
- *Method 2: Update the Input File*
  - ◆ Run reproject aspect
  - ◆ Compare the original and updated files
  - ◆ Using the updated file in analysis
- *"ERROR: Cannot find at least 3 source matches"*
- *Parameter files:*
  - ◆ wavdetect
  - ◆ wavdetect
  - ◆ reproject aspect
  - ◆ reproject aspect
- *History*
- *Images*
  - ◆ Source detections from wavdetect

---

# Correcting Absolute Astrometry with `reproject_aspect`

*CIAO 3.4 Science Threads*

## Overview

**Last Update:** 1 Dec 2006 – updated for CIAO 3.4: CIAO version in error messages

### *Synopsis:*

`reproject_aspect` applies corrections for translation, scale, and rotation to the WCS of a file by comparing two sets of source lists from the same sky region. If three or more sources are found to be a close match in position, the tool calculates the transformation that relates the two files and updates the WCS mapping from SKY(X,Y) to (RA,Dec) either by modifying the aspect solution or by revising the WCS keywords, depending on how the parameters are set.

`reproject_aspect` is actually a script which runs two tools: `wcs_match` and `wcs_update`. These tools may be run individually for slightly more flexibility; see the help files for details.

### *Purpose:*

To eliminate the errors in the absolute astrometry between two files.

### *Read this thread if:*

you intend to merge ACIS or HRC imaging observations and then perform astrometry.

### *Related Links:*

- [Overview: Reprojecting Files](#)
- [Correcting Aspect Prior to Merging](#): a section of the [Merging Data from Multiple Imaging Observations thread](#) which shows how to correct the WCS by hand in the case where this thread cannot be used (i.e. "[Cannot find at least 3 source matches](#)").

**Proceed to the [HTML](#) or [hardcopy \(PDF: \[A4\]\(#\) / \[letter\]\(#\)\)](#) version of the thread.**

---

## Get Started

**Sample ObsIDs used:** 441 (ACIS-I, Chandra Deep Field South); 581 (ACIS-I, Chandra Deep Field South)

**File types needed:** evt2; asol1

In this thread, we assume that all relevant files are in the same working directory.

## Creating Source Lists (wavdetect)

The first step is to create a source list for each of the observations. Note that the source list from the archive (`src2.fits` file) *is not accurate enough* for use with this tool. Users should create a new source list by running one of the detect tools; wavdetect is recommended because of its ability to separate closely spaced point sources.

wavdetect operates on an image, so we begin by binning the event files. When selecting the image region and binning, keep in mind that you need good position determinations from a set of sources well distributed across the field. However, you also need good source position measurements to see the differences, which implies wavdetect run on full-resolution data. Due to the heavy computational load, it is recommended to run wavdetect images no larger than 1024x1024.

To accommodate all these requirements, we create an image of the full field of view, binned by a factor of 2. While not full resolution, this is a small enough binning to yield good source positions:


```
unix% dmcopu "acisf00441N002_evt2.fits[bin x=2425:6095:2,y=1620:5770:2]" \
  441_img.fits

unix% dmcopu "acisf00581N003_evt2.fits[bin x=2425:6095:2,y=1620:5770:2]" \
  581_img.fits
```

Next, run wavdetect on each image to create a source list.

```
unix% punlearn wavdetect
unix% pset wavdetect infile=441_img.fits
unix% pset wavdetect outfile=441_src.fits
unix% pset wavdetect scellfile=441_scell.fits
unix% pset wavdetect imagefile=441_imgfile.fits
unix% pset wavdetect defnbkgfile=441_nbgd.fits
unix% pset wavdetect regfile=441_src.reg
unix% wavdetect
Input file name (441_img.fits):
Output source list file name (441_src.fits):
Output source cell image file name (441_scell.fits):
Output reconstructed image file name (441_imgfile.fits):
Output normalized background file name (441_nbgd.fits):

unix% punlearn wavdetect
unix% pset wavdetect infile=581_img.fits
unix% pset wavdetect outfile=581_src.fits
unix% pset wavdetect scellfile=581_scell.fits
unix% pset wavdetect imagefile=581_imgfile.fits
unix% pset wavdetect defnbkgfile=581_nbgd.fits
unix% pset wavdetect regfile=581_src.reg
unix% wavdetect
Input file name (581_img.fits):
Output source list file name (581_src.fits):
Output source cell image file name (581_scell.fits):
Output reconstructed image file name (581_imgfile.fits):
Output normalized background file name (581_nbgd.fits):
```

Figure 1  shows the detections for each observation. The contents of the parameter file may be checked with `pllist wavdetect` for ObsID 441 and ObsID 581.

`reproject_aspect` requires that the input source lists have RA and Dec. columns, so the FITS format files – `441_src.fits` and `581_src.fits` – must be used.

---

## Running `reproject_aspect`: Which Method to Use

There are two output methods for this tool, depending on what type of file is provided in the `updfilename` parameter:

1. **Create a new aspect solution:** If `updfilename` is an *aspect solution file*, a new aspect solution file is created with changes applied to the RA, Dec, and roll columns.
2. **Update the input file:** If `updfilename` contains an *image or event file* with a WCS associated with it, the WCS elements within that file are updated. No new output file is created.

If you are planning on reprocessing the data, i.e. running `acis_process_events` or `hrc_process_events` to apply new calibration, then you should **run `reproject_aspect` on the aspect solution file(s)**. Then use the new aspect solution when reprocessing the data; see the "[Apply the new aspect solution](#)" section for further information. The modified aspect files should then be used for any further analysis that requires the files, such as running `asphist` when creating ARFs or exposure maps.

**Do not mix and match the original and reprojected files.** If you create new aspect solution files, you must reprocess the event file with them. You will get incorrect results if you create new aspect solution files and pair them with the archived event data.

In the case where you have created images already, it is easier to **update the WCS in those files**. Make sure to update all the images with which you are working, though. If you have a separate counts image, background image, and exposure image, apply the same correction to all three files.

---

## Method 1: Create a New Aspect Solution

### Run `reproject_aspect`

The source list for ObsID 441 is used as the reference file, and the WCS information is taken from the image of that observation. The aspect solution file for ObsID 581 is given in the `updfilename` parameter, which means that a new aspect solution file will be created.

If there is more than one aspect solution file for the observation, it is possible to reproject all of them with a single run of `reproject_aspect`. There must be a one-to-one match between the number of files listed in the `updfilename` and `outfile` parameters; see [ahelp\\_stack](#) for more information on using stack inputs in CIAO.

```
unix% punlearn reproject_aspect
unix% pset reproject_aspect infile=581_src.fits
unix% pset reproject_aspect refsrcfile=441_src.fits
unix% pset reproject_aspect updfilename=pcadf056322870N003_asol1.fits
unix% pset reproject_aspect outfile=056322870N003_aspect.fits
unix% pset reproject_aspect wcsfile=441_img.fits
```

Running the tool with `verbose=2` prints information about the transform to the screen:

```
unix% reproject_aspect verbose=2
Input file with duplicate srcs (581_src.fits):
Input file with reference srcs (441_src.fits):
Either input asol file, or file with WCS to be updated (pcadf056322870N003_asol1.fits):
Output asol file (056322870N003_aspect.fits):
```

## Reprojecting Aspect – CIAO 3.4

```
...
32 common sources found between:
441_src.fits
581_src.fits
After deleting poor matches, 30 sources remain
...
num_asolInFiles: 1      num_asolOutFiles: 1
Transform scale_factor: 0.999925
Transform rotation angle (deg): 0.011084
Transform x translation (pixels): 1.851266
Transform y translation (pixels): 1.254355
```

The tool goes through a few iterations of calculating the transform, then prints the final scale, rotation, and translation elements to the screen. The new aspect file is `056322870N003_aspect.fits`. The contents of the parameter file may be checked with [plist reproject\\_aspect](#).

## Compare the original and updated files

The `dmdiff` tool can be used to compare the new and old aspect solution files:

```
unix% dmdiff 056322870N003_aspect.fits pcadf056322870N003_asol1.fits keys=no | less
Infile 1: 056322870N003_aspect.fits
Infile 2: pcadf056322870N003_asol1.fits
```

```
-----
SUBSPACE VALUE DIFFERENCES
-----
```

Message:	Column:	Row	Value(s):	Diff:
----------	---------	-----	-----------	-------

```
TABLE NAME: aspsol
```

```
-----
TABLE VALUE DIFFERENCES
-----
```

Message:	Row:Cell:	Column:	Value(s):	Diff:
Values are not equal	1	ra	53.1192183104317 53.11950435554	+0.00028604
Values are not equal	1	dec	-27.8053811702734 -27.8055525825571	-0.000171
Values are not equal	1	roll	47.2929656214818 47.2818816761885	-0.011083
Values are not equal	2	ra	53.119338430698 53.1196244765415	+0.00028604
Values are not equal	2	dec	-27.805588529648 -27.8057599416841	-0.0001714
Values are not equal	2	roll	47.2905415893518 47.2794576440585	-0.011083
Values are not equal	3	ra	53.1193596871029 53.1196457329912	+0.00028604
Values are not equal	3	dec	-27.8055928038649 -27.8057642158571	-0.000171
Values are not equal	3	roll	47.290548665563 47.2794647202697	-0.011083
Values are not equal	4	ra	53.1193719832299 53.1196580291443	+0.00028604
Values are not equal	4	dec	-27.8055953252597 -27.8057667372266	-0.000171
Values are not equal	4	roll	47.2905515594926 47.2794676141993	-0.011083

## Apply the new aspect solution

The event file for ObsID 581 now needs to be reprocessed with the new aspect solution in order to apply the updated information. For instructions on how to do so, follow the [Create a New Level=2 Event File thread](#).

Here we compare two versions of the level=2 event file reprocessed with `acis_process_events`. `581_newaspect_evt2.fits` was reprocessed with the new aspect solution file created by

## Reprojecting Aspect – CIAO 3.4

reproject\_aspect, while the aspect solution from the Archive was applied to 581\_oldaspect\_evt2.fits.

```
unix% dmdiff 581_newaspect_evt2.fits 581_oldaspect_evt2.fits keys=no | less
Infile 1: 581_newaspect_evt2.fits
Infile 2: 581_oldaspect_evt2.fits
...
-----
TABLE VALUE DIFFERENCES
-----
Message:                               Row:Cell:   Column:      Value(s):
-----
Values are not equal                    1           x 3334.86254882812 3332.91088867188
Values are not equal                    1           y 4630.58349609375 4629.17919921875
Values are not equal                    2           x 2872.9736328125 2871.14331054688
Values are not equal                    2           y 4010.57666015625 4009.08178710938
Values are not equal                    3           x 3480.75708007812 3478.85571289062
Values are not equal                    3           y 4373.736328125 4372.36083984375
Values are not equal                    4           x 3488.25708007812 3486.392578125
Values are not equal                    4           y 4185.38232421875 4184.00830078125
```

Remember that the modified aspect files should be used for any further analysis that requires the files, such as running [asphist](#) when creating ARFs or exposure maps.

---

## Method 2: Update the Input File

### Run reproject\_aspect

Again, the source list for ObsID 441 is used as the reference file, and the WCS information is taken from the image of that observation. A copy of the Archive event file for ObsID 581 is given in the updfilename parameter, so the WCS information will be directly updated in that file. (If you choose this method, make sure the file has write permission so it can be updated by the tool.)

```
unix% cp acisf00581N003_evt2.fits 581_wcs_evt2.fits
unix% chmod +w 581_wcs_evt2.fits

unix% punlearn reproject_aspect
unix% pset reproject_aspect infile=581_src.fits
unix% pset reproject_aspect refsrcfile=441_src.fits
unix% pset reproject_aspect updfilename=581_wcs_evt2.fits
unix% pset reproject_aspect wcsfile=441_img.fits
unix% reproject_aspect
Input file with duplicate srcs (581_src.fits):
Input file with reference srcs (441_src.fits):
Either input asol file, or file with WCS to be updated (581_wcs_evt2.fits):
Output asol file ():
```

The contents of the parameter file may be checked with [plist reproject\\_aspect](#).

---

## Compare the original and updated files

The `cols` option of `dmlist` shows the WCS information for the event file:

```
unix% dmlist acisf00581N003_evt2.fits cols
...
-----
World Coord Transforms for Columns in Table Block EVENTS
```

## Reprojecting Aspect – CIAO 3.4

```
-----  
ColNo      Name  
...  
8:      EQPOS(RA ) = (+53.1226)[deg] +TAN[(-0.000136667)* (sky(x)-(4096.50))]  
        (DEC)      (-27.8061)          (+0.000136667) ( (y) (+4096.50))  
  
unix% dmlist 581_wcs_evt2.fits cols  
...  
-----  
World Coord Transforms for Columns in Table Block EVENTS  
-----  
ColNo      Name  
...  
8:      EQPOS(RA ) = (+53.1223)[deg] +TAN[(-0.000136656)* ROT(+0.0111 deg)* (sky(x)-(4096.50))]  
        (DEC)      (-27.8059)          (+0.000136656) ( (y) (+4096.50))
```

Some rounding is introduced when these values are displayed. For greater precision, look at the RA---TAN and DEC--TAN keywords in the raw header of the event files:

```
unix% dmlist acisf00581N003_evt2.fits header,raw  
...  
Key 534: C      *MTYPE7      = EQPOS          / DM Keyword: Descriptor name.  
Key 535: C      *MFORM7      = RA,DEC        / [deg]  
Key 536: C      *TCTYP11     = RA---TAN      /  
Key 537: R      *TCRVL11     = 53.12262703   /  
Key 538: R      *TCRPX11     = 4096.50000000 /  
Key 539: R      *TCDLT11     = -0.000136667  /  
Key 540: C      *TCUNI11     = deg           /  
Key 541: C      *TCTYP12     = DEC--TAN      /  
Key 542: R      *TCRVL12     = -27.80606296  /  
Key 543: R      *TCRPX12     = 4096.50000000 /  
Key 544: R      *TCDLT12     = 0.000136667   /  
Key 545: C      *TCUNI12     = deg           /  
...  
  
unix% dmlist 581_wcs_evt2.fits header,raw  
...  
Key 534: C      *MTYPE7      = EQPOS          / DM Keyword: Descriptor name.  
Key 535: C      *MFORM7      = RA,DEC        / [deg]  
Key 536: C      *TCTYP11     = RA---TAN      /  
Key 537: R      *TCRVL11     = 53.12234100   /  
Key 538: R      *TCRPX11     = 4096.50000000 /  
Key 539: R      *TCDLT11     = -0.000136656  /  
Key 540: C      *TCUNI11     = deg           /  
Key 541: C      *TCTYP12     = DEC--TAN      /  
Key 542: R      *TCRVL12     = -27.80589153  /  
Key 543: R      *TCRPX12     = 4096.50000000 /  
Key 544: R      *TCDLT12     = 0.000136656   /  
Key 545: C      *TCUNI12     = deg           /  
...
```

The keyword number will change depending on if you have ACIS or HRC data, and whether a grating was used.

---

## Using the updated file in analysis

No further processing related to the aspect correction is required. The updated event or image file may now be used in your data analysis.

---



**"ERROR: Cannot find at least 3 source matches"**

reproject\_aspect needs at least three sources with a close match in position to be present in the infile and refsrcfile. If the input files don't meet this requirement, the tool exits with errors from wcs\_match and wcs\_update:

```
# wcs_match (CIAO 3.4): ERROR: Cannot find at least 3 source matches between reference and dupli
# wcs_update (CIAO 3.4): ERROR: Cannot read transform data from file.
# wcs_update (CIAO 3.4): ERROR: Could not close wcsUpdBlock.
# wcs_update (CIAO 3.4): ERROR: Could not close wcsUpdDataset.
```

Users that encounter this problem should correct the WCS by hand as shown in the [Correcting Aspect Prior to Merging](#) section of the [Merging Data from Multiple Imaging Observations](#) thread.

---

Parameters for /home/username/cxcds\_param/wavdetect.par

```

    infile = 441_img.fits      Input file name
    outfile = 441_src.fits     Output source list file name
    scellfile = 441_scell.fits Output source cell image file name
    imagefile = 441_imgfile.fits Output reconstructed image file name
    defnbgfile = 441_nbgd.fits Output normalized background file name
    scales = 2.0 4.0          wavelet scales (pixels)
    (regfile = 441_src.reg)    ASCII regions output file
    (clobber = no)            Overwrite existing outputs?
    (kernel = default)        Output file format (fits|iraf|default)
    (ellsigma = 3.0)          Size of output source ellipses (in sigmas)
    (interdir = .)            Directory for intermediate outputs
    (bkginput = )             Input background file name
    (bkgerrinput = no)        Use bkginput[2] for background error
    (outputinfix = )          Output filename infix
    (sigthresh = 1e-06)       Threshold significance for output source pixel list
    (bkgthresh = 0.001)       Threshold significance when estimating bkgd only
    (exptime = 0)             Exposure time (if zero, estimate from map itself)
    (expfile = )              Exposure map file name (blank=none)
    (expthresh = 0.1)         Minimum relative exposure needed in pixel to analyze it
    (bkgtime = 0)             Exposure time for input background file
    (maxiter = 2)             Maximum number of source-cleansing iterations
    (iterstop = 0.0001)       Min frac of pix that must be cleansed to continue
    (xoffset = INDEF)         Offset of x axis from optical axis
    (yoffset = INDEF)         Offset of y axis from optical axis
    (eband = 1.4967)          Energy band
    (eenergy = 0.393)         Encircled energy of PSF
    (psftable = ${ASCDS_CALIB}/psfsize20010416.fits -< /soft/ciao/data/psfsize20010416.fits) Tab
    (log = no)                 Make a log file?
    (verbose = 0)             Log verbosity
    (mode = ql)
```

---

Parameters for /home/username/cxcds\_param/wavdetect.par

```

    infile = 581_img.fits      Input file name
    outfile = 581_src.fits     Output source list file name
    scellfile = 581_scell.fits Output source cell image file name
    imagefile = 581_imgfile.fits Output reconstructed image file name
    defnbgfile = 581_nbgd.fits Output normalized background file name
    scales = 2.0 4.0          wavelet scales (pixels)
    (regfile = 581_src.reg)    ASCII regions output file
```

"ERROR: Cannot find at least 3 source matches"

## Reprojecting Aspect – CIAO 3.4

```
(clobber = no)           Overwrite existing outputs?
(kernel = default)       Output file format (fits|iraf|default)
(ellsigma = 3.0)        Size of output source ellipses (in sigmas)
(interdir = .)          Directory for intermediate outputs
(bkginput = )           Input background file name
(bkgerrinput = no)      Use bkginput[2] for background error
(outputinfix = )        Output filename infix
(sigthresh = 1e-06)     Threshold significance for output source pixel list
(bkgsigthresh = 0.001) Threshold significance when estimating bkgd only
(exptime = 0)           Exposure time (if zero, estimate from map itself)
(expfile = )            Exposure map file name (blank=none)
(expthresh = 0.1)       Minimum relative exposure needed in pixel to analyze it
(bkgtime = 0)           Exposure time for input background file
(maxiter = 2)           Maximum number of source-cleansing iterations
(iterstop = 0.0001)     Min frac of pix that must be cleansed to continue
(xoffset = INDEF)       Offset of x axis from optical axis
(yoffset = INDEF)       Offset of y axis from optical axis
(eband = 1.4967)        Energy band
(eenergy = 0.393)       Encircled energy of PSF
(psftable = ${ASCDS_CALIB}/psfsize20010416.fits -< /soft/ciao/data/psfsize20010416.fits) Table of 1
(log = no)              Make a log file?
(verbose = 0)           Log verbosity
(mode = ql)
```

---

Parameters for /home/username/cxcds\_param/reproject\_aspect.par

```
infile = 581_src.fits    Input file with duplicate srcs
refsrcfile = 441_src.fits Input file with reference srcs
updfile = pcadf056322870N003_asol1.fits Either input asol file, or file with WCS to be updated
outfile = 056322870N003_aspect.fits Output asol file
(wcsfile = 441_img.fits) Input file with WCS used in transform
(radius = 12)           radius used to match sources (arcsec)
(residlim = 2)          src pairs with residuals > residlim are dropped (arcsec)
(residfac = 0)          src pairs with residuals > residfac * position error are dropped
(residtype = 0)         residfac applies to: (0) each residual, (1) avg residuals
(logfile = STDOUT)     debug log file ( STDOUT | stdout | <filename>)
(clobber = no)         Overwrite existing output dataset with same name?
(verbose = 0)          debug level (0-5)
(mode = ql)
```

---

Parameters for /home/username/cxcds\_param/reproject\_aspect.par

```
infile = 581_src.fits    Input file with duplicate srcs
refsrcfile = 441_src.fits Input file with reference srcs
updfile = 581_wcs_evt2.fits Either input asol file, or file with WCS to be updated
outfile =                Output asol file
(wcsfile = 441_img.fits) Input file with WCS used in transform
(radius = 12)           radius used to match sources (arcsec)
(residlim = 2)          src pairs with residuals > residlim are dropped (arcsec)
(residfac = 0)          src pairs with residuals > residfac * position error are dropped
(residtype = 0)         residfac applies to: (0) each residual, (1) avg residuals
(logfile = STDOUT)     debug log file ( STDOUT | stdout | <filename>)
(clobber = no)         Overwrite existing output dataset with same name?
(verbose = 0)          debug level (0-5)
(mode = ql)
```

---

## History

24 Apr 2006 new for CIAO 3.3: original version

01 Dec 2006 updated for CIAO 3.4: CIAO version in error messages

---

URL: [http://cxc.harvard.edu/ciao/threads/reproject\\_aspect/](http://cxc.harvard.edu/ciao/threads/reproject_aspect/)

Last modified: 1 Dec 2006

**Image 1: Source detections from wavdetect**

