Pointing Refinement

Version 1.1

Release Date: January 7, 2004

Issued by the Spitzer Science Center

California Institute of Technology

Mail Code 314-6

1200 E. California Blvd

Pasadena, California 91125 USA

http://ssc.spitzer.caltech.edu/

help@spitzer.caltech.edu

Issue Dates

Version 1.0 Release: August 4, 2003
Version 1.1 Release: January 7, 2004

Acknowledgements

This document was written by David Makovoz.
The author would like to acknowledge Luisa Rebull and Francine Marleau for editing this document.

Table of Contents

1. Overview

The software in this package performs pointing refinement for FITS images. This goal is achieved by minimizing the discrepancy between the positions of point sources matched in each pair of the images. Global minimization of the cumulative discrepancy with respect to small changes in the pointing of the input images is implemented using a generic linear sparse matrix solver. The software has two modes of operation: relative and absolute. In the relative refinement mode the images are made self-consistent. In the absolute mode a list of real point sources is used as one of input lists. This results in fixing the absolute pointing of the input images. The processing includes the following steps. Frst, point source extraction is performed by running apex_1frame.pl package. For each input image a list of point sources is created. The list is trimmed using SNR threshold. This is done by running select perl script. In the relative mode the point source lists then are fed into pointingrefine module, which produces an output table with the refined pointing and the uncertainties for each image. The refined pointing is also written in the header of each input image using new keywords. In the absolute refine mode a list of sources with absolute poining is needed. Module fiducial_image_frame computes an area of the sky that includes in some optimal fashion all of the input images. A query to the 2MASS catalog in IRSA database is then made and a list of 2MASS point sources within found that area is created. There is an option of running the package with simulated data. In this case in the absolute refine mode the list of sources with absolute pointing is replaced with the truth list of point sources used to simulate the images.

The package includes the executables (bin/), libraries (lib/), perl scripts (bin/), source code (source/), header files (source/), documentation (doc/), four sample namelist files (cdf/), a list of input images and Dmasks (data/), and a script to set the environment pointingrefine.csh.

Perl script pointing_refine.pl runs all the modules involved in pointing refinement which includes running script apex_1frame.pl to perform single frame point source extraction.

2. Data

Two sets of images have been included in the package. First set is 20 2mass images that have their pointing refined. The images are listed in file data/2mass_ImageList.txt and located in data/2mass. The purpose of including this set is to exercise the functionality of absolute pointing refinement using the IRSA database. The second set is 20 images simulated by David Shupe at SSC for IRAC chanel 1, where the pointing was deliberately changed. The images are listed in the file data/IRAC_refine_ImageList.txt and located in data/IRAC_refine. This set comes with the truth list to be used for absolute refinement. Also included are the file with the correct pointing for the 20 images data/IRAC_true_pointing.txt, and the truth list used to simulate the IRAC data data/IRAC_truthlist.txt.

3. Usage

Untar the package and set the environmental variables by sourcing pointinfrefine.csh.

The PERL_PATH will be set automatically. If that doesn't work, please, make sure that PERL_PATH is set to point to the location where your perl is installed. Now run the perl script from the command line. Here is the command line to run pointing refinement of the 2mass data (data/2mass):

```
pointing_refine.pl -n pointing_refine_2mass.nl -N apex_1frame_2mass.nl
```

When this works you will have the results written in output_2mass/ subdirectory.

By running:

```
pointing_refine.pl -n pointing_refine_IRAC.nl -N apex_1frame_IRAC.nl
```

pointing refinement is performed on the IRAC data (data/IRAC). The results are saved in the output_IRAC/ subdirectory.

The namelists used are copied to the output directory. By default the filenames of the namelists are pointing_refine.nl and apex_1frame.nl. If you use your own data, you would need to provide a PRF image. The details on how to produce those images see doc/PRF-How-To.txt.

4. Input

The input for pointing_refine.pl:

Input File	Default Name	Set in Namelist	Set on Commandline	Required
namelist for pointing_refine.pl	pointing_frame.nl	N/A	-n	y
namelist for apex_1frame.pl	apex_1frame.nl	N/A	-N	y
List of input images	image_stack.txt	IMAGE_STACK_FILE_NAME	-I	y
List of input uncertainty images	-	SIGMALIST_FILE_NAME	-S	n
permamently damaged pixels mask image	-	PMASK_FILE_NAME	-M	n
list of DCE status mask images (Dmask)	-	DCE_STATUS_MASK_LIST	-d	n

See the APEX User’s Guide for the description of apex_1frame.pl input. The fatal bit pattern for the D- and P- masks are set in the namelist for apex_1frame.pl. Do not set any input filenames in apex_1frame.nl, such as INPUT_IMAGE_FILE_NAME, SIGMA_FILE_NAME, PMASK_FILE_NAME, DCE_STATUS_MASK_FILE. These filenames will be passed down to apex_1frame.pl internally in pointing_refine.pl on the command line.

5. Namelist

The namelist file contains the namelists for all modules and some input parameters for the script itself. It is expected to be found in the directory specified by enironment variable SIRTF_CDF(see below).

Command line options take precedence over namelist options.

Here is an excerpt from a namelist file that pertains to script pointing_refine.pl

```
run_singleframePSE = 0
run_fiducial_image_frame = 1
run_2masscatalog = 1
run_select = 1
run_pointingrefine = 1

IMAGE_STACK_FILE_NAME = 2mass_ImageList.txt
SIGMALIST_FILE_NAME = output_IRAC/AllSigmaList.txt
OUTPUT_DIR = output_2mass
NICE = 1
select_conditions =
"SNR > 10 and deblend ! NO and deblend ! PO and deblend ! AO"
select_columns =
"srcid,detid,RA,delta_RA,Dec,delta_Dec,delta_RAD,flux,delta_flux"
ABSOLUTE_REFINE = 1
verbose = 1
delete_intermediate_files = 1
```

The script executes only the modules that are set to 1. In the example above apex_1frame.pl will NOT be called. It is expected to have been executed and produced the extraction lists for the input images listed in 2mass_ImageList.txt. The extraction lists are expected to be saved in output_2mass directory. Modules fiducial_image_frame, tmasspsc, select and pointingrefine will be executed in the order listed.

Switch ABSOLUTE_REFINE is set to perform the absolute pointing refinement, as opposed to the relative pointing refinement (see below pointingrefine module).

select_conditions and select_columns are used by select perl script (see below).

The script expects to find a list of uncertainty images named output_IRAC/AllSigmaList.txt. Whether or not it will be used depends on the setting in the apex_1frame.nl namelist file.

All the results will be written in OUTPUT_DIR, which can be specified as a relative path, as above, or as an absolute path, e.g. /ssc/pipe/davidm/band4.

Switch NICE =1 is used to run all the modules called by the script with "nice +19".

If delete_intermediate_files switch is set all the extraction lists produced by apex_1frame.pl as well as the selected lists produced by select will be deleted.

6. Environment Variables

The following environment variables are used in the script. SIRTF_BIN, SIRTF_CAL, SIRTF_ANC, SIRTF_CDF. SIRTF_CDF specifies the location of the namelist file, by default it's ./cdf. Also the log files for each module are written into that directory

SIRTF_CAL specifies the location of the PRF image used by apex_1frame.pl, by default it's ./cal.

SIRTF_BIN specifies the location of the executables, it has no useful for you default, you would have to set it to wherever you copy the executables to.

SIRTF_ANC specifies the location of the error header files included in the package. The default is not useful.

Also make sure that the place where you copy the libraries is included in your LD_LIBRARY_PATH.

Make sure PERL_PATH is set to point to the location where your perl is installed.

7. Modules

Here is a short description of the modules.

See the APEX User’s Guide for the description of the single frame point source extraction package apex_1frame.pl

Module fiducial_image_frame creates a fiducial image frame (FIF) that is used by all Post-BCD processes. It generates the WCS parameters for a minimally bounding region that encloses all the input images. If parameter CROTA2 is set in the namelist the orientation of the FIF will be defined by this value. Otherwise it will be computed by the module to optimize the overlap between the FIF and input images.

Module tmasspsc generates a list of 2mass point sources located within the FIF. It does so by querying IRSA database accessible online at http://irsa.ipac.caltech.edu/applications/Gator/. See README_tmasspsc for a more detailed description of the module and its usage.

Perl script select is run to weed out faint and mutilpe point sources. To see the tutorial run it without any arguments. It reads extracted point source lists produced by apex_1frame.pl and saves in separate files only the columns specified by select_columns. By default, the following columns are saved: "srcid,detid,RA,delta_RA,Dec,delta_Dec,delta_RAD,x,y,flux,delta_flux,chi2/dof,status,SNR", which is a subset of all the columns generated by the point source extractor. Only the point sources satisfying the conditions specified by select_conditions are saved. The only logical operation to combine different conditions is "and". Symbol "!" is equivalent to "!=" (not equal).

Module pointingrefine is the central part of the package. Input are the trimmed lists of point sources extracted from a set of overlapping images. The software uses this information to find a "global minimization" of all relative offsets amongst all overlapping images. This is a novel method utilizing a generic linear sparse matrix solver. The pointings and orientations of images can be refined in either a "relative" sense where pointings become fixed relativeto a single image in the set, or, in an "absolute" sense if absolute point source information is known. As an absolute point source list one can use either a subset of the 2MASS catalog (see tmasspsc module) for real data, or, for simulated data, the truth list of point sources used to simulate the images.

The variables in the namelist are defined as follows:

`Max_Search_Radius` = The maximum search radius for point source position matching between overlapping frames in units of arcseconds. If not specified, this will default to 5.0 arcsec. As a rule of thumb, this should be set to allow for the maximum expected pointing uncertainty (if known apriori) and the maximum uncertainty in the x,y centroids of point sources from extractions: =~ sqrt[sigma^2(centroid) + sigma^2(pointing)]

`Max_Flux_Diff` = Maximum allowable percentage flux difference tolerable between a point source match. Flux matching is only performed between extracted point sources (not with absolutes). The default value is 5.0 (%). In general, this is defined: 100*[|f1 - f2|/f1 + f2] where f1, f2 are the extracted flux of each source in a potential match pair.

`Flux_Threshold` = Flux threshold (in source extractor units) above which two sources must satisfy in order to be declared a match. The default value is 0.

`Max_Num_Sources_Per_Image` = The maximum number of point sources to use from each source extraction table (per image) if the number of extractions exceeds this number. The default is 100.

The module outputs table RefinedPointing.tbl that lists the refined pointing and uncertainties for each input image. There is also a QA file named QAlogfile.txt which lists the change in pointing for each file along with some other diagnostic information. The refined pointing and uncertainties are also written in the header of each file using the following keywords.

```
RARFND  =     82.1180602003437 / [deg] Refined RA
DECRFND =     35.8189421041455 / [deg] Refined DEC
PA_RFND =   0.0386636245264071 / [deg] Refined PA (= -CROTA2_refined)
ERARFND =   0.0057885798536793 / [deg] Error in refined RA
EDECRFND=  0.00550417637885658 / [deg] Error in refined DEC
EPA_RFND=     1.81683424260449 / [deg] Error in refined PA
NASTROM =                    2 / # Astrometric sources for absolute refinement
RARESID =    0.917008969418021 / [arcsec] Residual: Observed-Refined RA
DECRESID=     1.15092098836271 / [arcsec] Residual: Observed-Refined DEC
PA_RESID=     139.189048295066 / [arcsec] Residual: Observed-Refined PA
CD11RFND= -0.00432312097604533 / [deg/pix] Refined CD matrix element 1,1
CD12RFND= -2.89977161921492E-06 / [deg/pix] Refined CD matrix element 1,2
CD21RFND= -2.91727511156294E-06 / [deg/pix] Refined CD matrix element 2,1
CD22RFND=  0.00429718248480607 / [deg/pix] Refined CD matrix element 2,2
```

Please consult sds-s8.0_v4.5-pointingrefine.doc for further details.

8. Short Description of Intermediate and Final Products

1. Trimmed by select point source extraction tables. Naming convention: <input_image_name>_extract_select.tbl.

2. Fiducial image frame FIF.tbl.

3. irsa.tbl - the list of 2mass point source in the format provided by the IRSA server.

4. tmass_list.tbl - the list of 2mass point sources reformatted to be used as input to pointingrefine module

5. RefinedPointing.tbl is an output of pointingrefine. It is a list of the refined pointing for each input images.

6. QAlogfile.txt - some QA and diagnostic information produced by pointingrefine.