674-SO-43, Version 3.0,
SSC-PD-4045

SIRTF Science Center

Downlink Segment

Subsystem Design Specification

AOT Products Subsystem: DELTAFPOINT

5 November 2001

National Aeronautics and
Space Administration

California Institute of Technology
SIRTF Science Center

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

THIS PAGE IS INTENTIONALLY LEFT BLANK

SIRTF Science Center

Subsystem Design Specification

Prepared by:

Frank Masci

Concurred by:

Dave Shupe

Approved by:

Mehrdad Moshir

Concurred by:

Bill Green

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

1 Revision History

Version	Description	Date
1.0	Initial version	April 2, 2001
2.0	Performed error propagation and estimatation and included uncertainties in the output delta-files.	July 24, 2001
3.0	Included option to use specific (e.g. commanded) RA, DEC pointing keywords from FITS headers.	November 5, 2001

2 Table of Contents

1 REVISION HISTORY....IV
2 TABLE OF CONTENTS....V
3 LIST OF FIGURES....VII
4 LIST OF TABLES....VIII
1. INTRODUCTION....9
1.1. Purpose and Scope....9
1.2. Document Organization....9
1.3. Relationship to Other Documents....9
1.4. Change Procedure....10
2. OVERVIEW....10
2.1. DELTAFPOINT Requirements....10
2.2. Applicable Documents....11
2.3. Version History....11
2.3.1. Version 1.0....11
2.3.2. Version 2.0....11
2.3.3. Version 3.0....11
2.4. Liens. 12
3. INPUT....12
3.1. DELTAFPOINT Input....12
3.1.1. DELTAFPOINT NAMELIST Input....12
3.1.2. DELTAFPOINT Command-Line Input....15
3.1.3. DELTAFPOINT FITS Input....15
4. PROCESSING....16
4.1. DELTAFPOINT Processing....16

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

4.2 DELTAFPOINT Processing Phases 17
4.1.1. DELTAFPOINT Initialization 17
4.1.2. FITS Image Pointing Keywords Input 18
4.1.3. Defining the Average Reference Frame (Optional)....................19
4.1.4. Computing Relative Offsets 21
4.1.5. Rotational and Reference-Pixel Re-scaling 23
4.1.6. Error Propagation and Estimation 23
4.1.7. Delta-File (IPAC Table) Output 24
4.1.8. Termination....................25
5. ALGORITHM....................25
5.1. Algorithm Specifics....................25
5.2. Assumptions and Requirements....................25
6. OUTPUT 26
6.1. DELTAFPOINT Output....................26
6.1.1 DELTAFPOINT Log-File Output 27
6.1.2 Delta-File (IPAC-Table) Output 27
7. TESTING....................31
8. USAGE EXAMPLES 32
9. GLOSSARY 32

List of Figures

Figure 1. DELTAFPOINT data and processing flow 18

Figure 2. Schematic showing input and reference image planes 21

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

List of Tables

Table 1. Namelist File 12

Table 2. Command Line Options...15

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

1. Introduction

1.1. Purpose and Scope

The Subsystem Design Specification is a document that describes the basic requirements, assumptions, definitions, software-design details and necessary interfaces for each subsystem. The document will be used to trace the incremental development of each subsystem and also to allow trace-back of levied requirements; this document should have sufficient detail to allow future modification or maintenance of the software by developers other than the original developers. This document is an evolving document as changes may occur in the course of science instrument hardware design and maturity of operational procedures. This document is not intended to repeat sections or chapters from other Project documents; when appropriate, references to proper sections of primary reference documents will be made.

1.2. Document Organization

This document is organized along the major themes of Requirements; Assumptions; Operational Concept; Functional Descriptions; Functional Dependencies; Input; Output; Other S/S Interfaces; Algorithm Descriptions (when applicable); and Major Liens.

The material contained in this document represent the current understanding of the capabilities of the major SIRTF systems. Areas that require further analysis are noted by TBD (To Be Determined) or TBR (To Be Resolved). TBD indicates missing data that are not yet available. TBR indicates preliminary data that are not firmly established and are subject to change.

1.3. Relationship to Other Documents

The requirements on the operation of SIRTF flow down from the Science Requirements Document (674-SN-100) and the Facility Requirements Document (674-FE-100). The Science Operations System is governed by the SOS Requirements Document (674-SO-100). The current document is also cognizant of the requirements that appear in the Observatory Performance and Interface Control Document (674-SEIT-100) as well as the Flight Ground Interface Control Document (674-FE-101). This document is also affected by the FOS/SOS Interface Control Document (674-FE-102) that governs interfaces between the Flight Operations System and the Science Operations System. Related Software Interface Specifications (SIS) will be as indicated in Section 2.2 of this document.

1.4. Change Procedure

This document is a level 4 document according to the SIRTF Project Documentation Plan (674-FE-103). Changes to this document after approval require the approval of the SOS Change Board (TBD). The process for change control is described in the SOS Configuration Management Plan.

2. Overview

DELTAFPOINT reads celestial pointing keyword data from a set of standard FITS images and computes the translational and rotational offsets between frames in a Cartesian coordinate system relative to either: a reference frame defined by a single image, or, an "average reference" defined by the mean pointing of all frames in the input list. Additionally, the user can choose the rotational x, y center for the input frames and the pixel-scale in the reference frame. The user also has the option of using specific pointing keywords from image headers (e.g. predicted or commanded values). "DELTAFPOINT" attained its name to stand for "*Delta-File* from *Pointing*".

The software uses routines from the standard World Coordinate Library (WCS) (Doug Mink, 2001, SAO) to convert between celestial and Cartesian coordinates. All standard types of map-projections (as specified by relevant FITS keywords) are supported. The primary output from DELTAFPOINT is a table in IPAC format listing the input images and their relative offsets from a user-specified reference frame. Uncertainties in these relative offsets are also reported. This table is commonly referred to as a "Delta-File". DELTAFPOINT is written in standard C.

2.1. DELTAFPOINT Requirements

DELTAFPOINT is initiated by a startup script under the control of the pipeline executive and does its required functions for a given DCE image or pre-processed DCE image; this involves performing the following tasks.

A.) Retrieve the command line parameters passed by the start up script and use them to run the program.

B.) Read in as input a list of standard FITS images with relevant pointing header keywords, an optional single reference frame FITS image and various processing parameters.

C.) Produce as primary output an IPAC table.

D.) Provide exit codes to the pipeline executive and also provide logon and logoff messages identifying the version number and write any error messages to the standard output devices.

E.) Produce a processing summary either to standard output or a log file.

2.2. Applicable Documents

The following documents are relevant to the DELTAFPOINT program of the AOT PRODUCTS Subsystems.

A.) The SOS Requirements Document

B.) The SOS Downlink Requirements Document

C.) The SOS Downlink Software Development Guidelines

D.) The following Software Interface Specifications (SIS):

SOSDL-SIS-PD-3009 (Algorithm and IPAC table output)

2.3. Version History

2.3.1. Version 1.0

Initial version created on April 2, 2001.

2.3.2. Version 2.0

Error propagation is performed to compute uncertainties corresponding to the translational and rotational offsets for each input image. All uncertainties are reported in the output IPAC table.

2.3.3. Version 3.0

Instead of using standard pointing keywords defined under the standard WCS software, we now have the option to use different (RA and DEC) pointing keywords from image headers. An example where this may be used is using predicted (commanded) pointings instead of those returned by downlink telemetry to compute delta files.

2.4. Liens

No major liens have been identified.

3. Input

3.1. DELTAFPOINT Input

DELTAFPOINT takes all of its input from either the command line or namelist file, which is set up by the startup script that is controlled by the pipeline executive or standalone. If the namelist is not specified, then all required inputs are expected from the command line. If both namelist and command-line inputs are specified, then the command-line inputs override the namelist values. Prior to reading namelist and/or command-line parameters, default values for the relevant parameters are assigned.

3.1.1. DELTAFPOINT NAMELIST Input

DELTAFPOINT reads the NAMELIST file whose name is passed to it by start-up script. The name of the NAMELIST is DELTAFPOINTIN. The parameters that can be defined in the NAMELIST are listed in Table 1.

Namelist variable	Description	Dim.	Type	Units	Default
FITS_Image_List_Filename	Required filename containing list of FITS-images.	161	C	DN (redundant)	Null

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

FITS_Ref_Image_Filename	Optional FITS image from list representing a "single" image reference frame. Must specify if RefFrameFlag = 2 below.	161	C	DN (redundant)	Null
Data_Out_Filename	Required name of Delta-File.	161	C	-	Null
RefFrameFlag	1 = use average reference frame, 2 = use single reference frame specified above.	1	I*1	-	1
CenterFlag	1 = use CRPIX values as rotational center, 2 = use Xcenter, Ycenter below.	1	I*1	-	1
Xcenter	*x*-coord. of rotational center.	1	R*4	Input pixels	0
Ycenter	*y*-coord. of rotational center.	1	R*4	Input pixels	0
CdeltFlag	1 = use pixel scales from input reference image or 1st image in list. 2 = use values below.	1	I*1	-	1
Xpixscale	Pixel scale in *x* coordinate.	1	R*4	deg./pixel	0.001
Ypixscale	Pixel scale in *y* coordinate.	1	R*4	deg./pixel	0.001
PointingFlag	1 = Use CRVAL1, CRVAL2 header keywords for RA/DEC 2 = Use FITS keywords below	1	I*1	-	1
RAkeyword	R.A. FITS header keyword	8	C	-	Null
DECkeyword	DEC. FITS header keyword	8	C	-	Null
Log_Filename	Optional output log filename	161	C	-	stdout

Ancillary_File_Path	Pathname where supporting source files are installed.	161	C	-	./ (current directory)

Table 1. Namelist file

The following is an example of the contents of a "DELTAFPOINTIN" NAMELIST file that might be used, where the values specified are not necessarily realistic.

```
&DELTAFPOINTIN
 Comment = 'Generic namelist file for deltafpoint, default values.',
 Ancillary_File_Path = '../deltafpoint_v3',
 Comment = 'Following required if RefFrameFlag = 2 below:',
 FITS_Ref_Image_Filename = './testing/sws4322a00201.fits',
 FITS_Image_List_Filename = './testing/deltafpoint_images.list3',
 Data_Out_Filename = './testing/pointing.tbl',
 Comment = '1 = compute average, 2 = use ref_file above, Default = 1:',
 RefFrameFlag = 1,
 Comment = '1 = use CRPIX values, 2 = use values below, Default = 1:',
 CenterFlag = 1,
 Comment = 'Xcenter, Ycenter in units of pixels:',
 Xcenter = 0,
 Ycenter = 0,
 Comment = '1 = use keyword CDELT values from ref. frame, 2 = use values
               below, Default = 1:',
 CdeltFlag = 1,
 Comment = 'X-pixelscale, Y-pixelscale in units of [deg/pixel]:',
 Xpixscale = 0.001,
 Ypixscale = 0.001,
 Comment = '1 = use CRVAL RA,DEC pointings, 2 = use FITS keywords below,
               Default = 1:',
 PointingFlag = 2,
 RAkeyword = 'RAPRED',
 DECkeyword = 'DECPRED',
 Log_Filename = 'stdout',
&END
```

3.1.2. DELTAFPOINT Command-Line Input

Alternatively, all inputs can be specified via command line, in which case, a namelist file is not needed. Or, inputs can be provided with a hybrid of both namelist and command-line mechanisms, with the latter overriding the former. Table 2 lists the available command-line options associated with their namelist-variable counterparts, as well as other options for specifying the namelist-file name and making the standard output more verbose.

3.1.3. DELTAFPOINT FITS Input

DELTAFPOINT uses the FITSIO library routines to read in the FITS-formatted input data file. The routines used are: fits_open_file, fits_read_keys_lng, fits_read_keys_dbl, and fits_close_file.

Command-line option	Variable
-n	Namelist_Filename
-f	FITS_Image_List_Filename
-i	FITS_Ref_Image_Filename
-o	Data_Out_Filename
-g	RefFrameFlag
-cf	CenterFlag
-cx	Xcenter
-cy	Ycenter
-pf	CdeltFlag
-px	Xpixscale
-py	Ypixscale

-kf	PointingFlag
-kr	RAkeyword
-kd	DECkeyword
-l	Log_Filename
-a	Ancillary_File_Path
-v (verbose switch)	-
-vv (super-verbose switch)	-
-d (debug switch)	-

Table 2. Command-line options

4. Processing

4.1. DELTAFPOINT Processing

DELTAFPOINT begins processing by writing its name and version number to standard output (verbose mode only), and then it initializes relevant variables with defaults values, and checks that the required namelist parameters and/or command-line parameters were passed to it. If this condition is not true, then it writes a message stating which parameters are missing, recommends a look at this document, and terminates by issuing an appropriate exit code to the pipeline executive; otherwise it proceeds as follows.

If an error occurs during processing, then an error message is written to standard output, a termination-status code is written to the log file, and an exit code to the pipeline executive issued.

After processing, the program name and version number, namelist filename (if used), input, and output filenames, values of other input parameters, date and time, processing time, and a termination-status code are written a log file.

4.2 DELTAFPOINT Processing Phases

DELTAFPOINT operates in eight phases: initialization, FITS image pointing keywords input, computation of “average” reference frame if specified, computation of translational and rotational offsets, rotational and reference-pixel re-scaling, error-propagation and estimation, delta-file output table and termination. This processing level is depicted in Figure 1.

4.1.1. DELTAFPOINT Initialization

DELTAFPOINT initializes itself by performing the following tasks.

A.) A message is printed to STDOUT (verbose mode only), which includes the program name and version number.

B.) If specified on the command line, the NAMELIST file is opened and read. If any errors are encountered, a message is printed, and execution aborts.

C.) The remaining command-line inputs are read and checked for correct data range, consistency, etc. If any errors are encountered, a message is printed, and execution aborts.

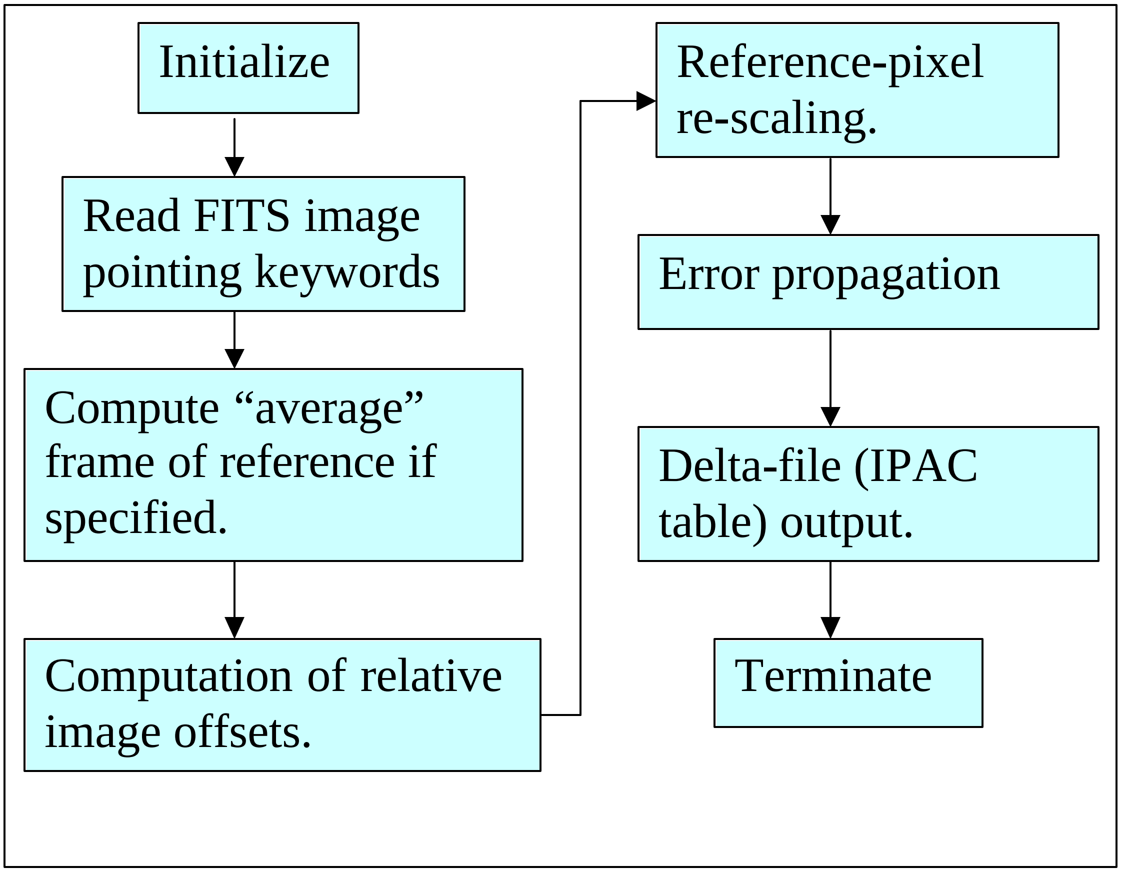

Figure 1. DELTAFPOINT data and processing flow

4.1.2. FITS Image Pointing Keywords Input

The pointing keywords in the list of input FITS images (namelist parameter: FITS_Image_List_Filename) are read using the *fitsrhead* routine from the WCS library. Each set of pointing keywords from each image header are stored in memory as structure variables using the WCS routine *wcsinit*. In general, all images must be composed of a single data plane. Below is a snippet from a standard FITS header containing the most commonly used WCS keyword conventions.

```
CTYPE1  = 'RA---TAN'           / RA Projection
CTYPE2  = 'DEC--TAN'           / DEC Projection
CRVAL1  =        323.306529753 / RA of CRPIX1
```

```
CRVAL2  =       8.31972997116 / DEC of CRPIX2
CRPIX1  =                64.5
CRPIX2  =                64.5
CDELT1  =    -0.0042971834632 / Sample pixel size (degrees)
CDELT2  =    0.00432296656398 / Line pixel size (degrees)
CROTA2  =       17.9706090448 / Twist angle wrt axis 2
```

Alternatively, one can use any pre-defined RA, DEC keywords from the FITS headers other than the WCS values CRVAL1 and CRVAL2. These can be specified by setting the namelist/command-line parameter PointingFlag = 2 followed by the RA and DEC keyword strings (parameters RAkeyword and DECkeyword).

4.1.3. Defining the Average Reference Frame (Optional)

If the namelist/command line parameter RefFrameFlag is set to "1", then a frame of reference defined by an average of all the input pointings is computed. All offsets are then computed relative to this frame (see Section 4.1.4). An average frame of reference is defined using the following steps:

(1). The position angles (header keyword CROTA2; measured East from North on the sky) from all input images are averaged. To avoid ambiguity in the average over the range 0 - 360° if CROTA2 > 180°, the following algorithm is used. A set of CROTA2 values: C_1 , C_2 , C_3 ¼C_N is first averaged in the normal way which we designate as C_{Av1}:

$$C_{Av1} = \frac{1}{N}\sum_i C_i$$

If any values C_i exceed 180°, they are replaced by $C_j = C_i - 360°$ and a new average C_{Av2} is computed:

$$C_{Av2} = \frac{1}{N}\sum_j C_j$$

Next, the "absolute differences" D_1 , D_2 of these averages from the initial C_i are computed:

$$D_1 = \sum_i \left|C_{Av1} - C_i\right| \quad \text{and} \quad D_2 = \sum_i \left|C_{Av2} - C_i\right|$$

The correct average value we will want to use is that which yields the smallest difference value. Let us define this average by <crota2>.

(2). We start by defining a fiducial (or arbitrary) frame with known RA, DEC (keywords CRVAL1, CRVAL2) and corresponding pixel coordinates (CRPIX1, CRPIX2) by choosing the first image in the input list. The CROTA2 and map projection (CTYPE keywords) of this "basis" frame are re-defined to be <crota2> and TAN. In other words, we will project onto a "TANGENT" (undistorted) reference plane. The *wcsdeltset* and *wcstype* routines are used to define this basis frame.

(3). All input images *i* are transformed to pixel coordinates in the basis reference frame defined in 2 above. We will define a pixel coordinate in the basis frame as (x'_i , y'_i). The *wcs2pix* routine is used:

$$RA_i, DEC_i \xrightarrow[<crota2>,\, TAN]{} (x'_i, y'_i) \qquad (1)$$

(4). All coordinates corresponding to all input images in the basis frame are averaged, yielding ($<x'_i>$, $<y'_i>$). These are then transformed to a "mean" sky pointing using a TANGENT projection. The *pix2wcs* routine is used:

$$< x'_i >, < y'_i > \xrightarrow[<crota2>,\, TAN]{} < RA >, < DEC >. \qquad (2)$$

The following set of keywords therefore fully define the ***average frame*** of reference:

CRVAL1 = $< RA >$

CRVAL1 = $< DEC >$

CRPIX1 = $CRPIX1_1$

CRPIX2 = $CRPIX2_1$

CROTA2 = $< crota2 >$

CDELT1 = $CDELT1_1$

CDELT2 = $CDELT2_1$

CTYPE1 = $RA{-}{-}{-}TAN$

CTYPE2 = $DEC{-}{-}TAN$

The quantities with a "1" subscript refer to values in the header of the first listed FITS image in the input list. The CDELT keywords which define the reference frame pixel scale can be modified via the namelist/command-line parameters: Xpixscale, Ypixscale (with CdeltFlag=2). These default to values given in the FITS header of the first listed image as shown above. The above keywords are copied to the header of the output delta-file if the averaging option is chosen (see Section 6.1.2).

4.1.4. Computing Relative Offsets

Our method for computing rotational and translational offsets of input images relative to either a single input image or an average frame (see Section 2.1.3) is as follows. Coordinates shown with a prime (e.g: x'_i , y'_i ; x''_i , y''_i etc..) refer to quantities transformed into the *reference* frame as shown in Figure 2. Un-primed coordinates are measured in the individual input image frames.

(1). All CRPIX1, CRPIX2 values of input images i are transformed to pixel coordinates in the reference frame using a TANGENT projection. The reference frame parameters can be either from the FITS header of a single image specified in the namelist/command-line, or an "average frame" described in Section 2.1.3. The *wcs2pix* routine is used:

$$RA_i, DEC_i \xrightarrow[\text{Reference Frame Parameters}]{} (x'_i, y'_i) \qquad (3)$$

(2). To compute the rotational offset with respect to the reference plane, we define a second point in each input image i with coordinates [x_{2i} , y_{2i}] = [0.5*NAXIS$_i$ + CRPIX1$_i$, CRPIX2$_i$] to transform into our reference frame. Our first point is simply [x_{1i} , y_{1i}] = [CRPIX1$_i$, CRPIX2$_i$]. This choice for the second point is arbitrary. It does not need to exist in the actual image, but together with first point, it must form a perpendicular bisector with the y-axis in the input image. Figure 2 shows the geometry involved. The primed axes refer to the reference frame.

In order to transform this second point from each image into the reference frame, we must first transform onto the sky:

$$x_{2i}, y_{2i} \xrightarrow[\text{Input image } i \text{ parameters with TAN projection}]{} RA_{2i},\ DEC_{2i}\ . \qquad (4)$$

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

To avoid uncertainties due to distortion in the telescope focal plane when projecting an input image onto the sky, a TANGENT projection is ensured. In other words, if the FITS header indicates a map projection other than "TAN" (e.g. "TNX") then the relevant keywords in the image header (CTYPE1, CTYPE2) are reset to "TAN". This is accomplished using the *wcstype* routine. The celestial coordinates of this second point are then transformed to the reference frame:

$$RA_{2i}, DEC_{2i} \xrightarrow[\text{Reference Frame Parameters}]{} (x''_i , y''_i) \quad (5)$$

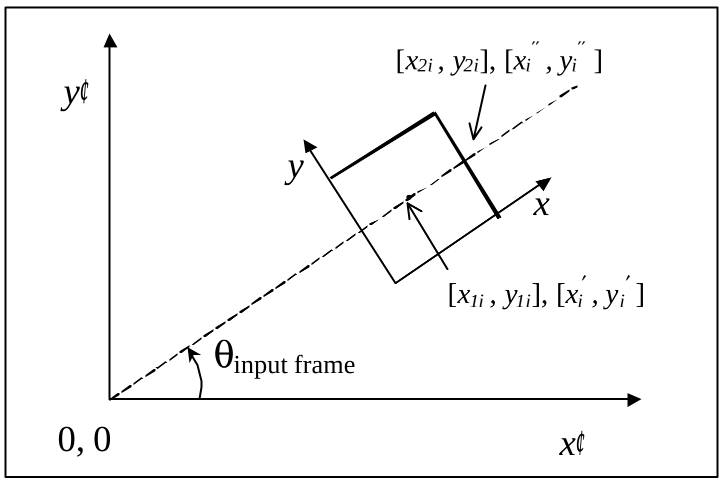

Figure 2. Schematic showing input and reference image planes

(3). Having transformed these two points into the reference plane for each image *i*: ([$x¢_i$, $y¢_i$] and [$x¢¢_i$, $y¢¢_i$]) the rotational offsets q_i can be computed:

$$q_i = -\arctan\left[\frac{y''_i - y'_i}{x''_i - x'_i}\right]. \quad (6)$$

(4). Once all rotational offsets are computed, the orthogonal translations ($X^i{}_T$, $Y^i{}_T$) between reference plane and input images are computed from the transformation equations:

$$X_T^i = x'_i - (x_{1i} - x_c)\cos J_i - (y_{1i} - y_c)\sin J_i - x_c \quad (7)$$

$$Y_T^i = y'_i + (x_{1i} - x_c)\sin J_i - (y_{1i} - y_c)\cos J_i - y_c \, , \quad (8)$$

where [x_{1i} , y_{1i}] ≡ [CRPIX1$_i$, CRPIX2$_i$] are input-image coordinates, [$x¢_i$, $y¢_i$] are corresponding coordinates in the reference plane (see step 1 in section 4.1.4) and [x_c , y_c] are input-image coordinates of the rotational center. q_i is defined by Eqn.6. In essence, X^i_T and Y^i_T are relative orthogonal translations between the reference frame and the co-rotated input-image, that is, image origin offsets after rotating the input-image so that it is unrotated with respect to the reference frame.

4.1.5. Rotational and Reference-Pixel Re-scaling

One can specify the pixel scale (in degrees/pixel) for the reference frame or units in which translational offsets are measured if desired. This is specified by setting the namelist/command-line parameter CdeltFlag = 2 and setting values for the parameters Xpixscale and Ypixscale. On the other hand if CdeltFlag = 1, then the pixel scale in the *single* input reference image is used. If an average reference frame is computed, the pixel scale in the first listed image is used.

4.1.6. Error Propagation and Estimation

Uncertainties in rotational and translational offsets for each input image are computed from propagating the pointing uncertainties associated with the CRVAL1 (RA) and CRVAL2 (DEC) FITS header keywords in equations (6), (7) and (8). The pointing uncertainties are specified by the header keywords CRDER1 and CRDER2 respectively. From Eqn.(6), the uncertainty in the relative rotational offset q is given by:

$$s(q) = s(x)/(1 + x^2) \, , \quad (9)$$

where $s(x) = x\left[\frac{2s^2_{py}}{(y''-y')^2} + \frac{2s^2_{px}}{(x''-x')^2}\right]^{1/2}$

and $x = \left|\frac{y''-y'}{x''-x'}\right|$,

THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

$$\sigma^2_{px} = \frac{1}{s_x^2}[\sigma^2(RA) + \sigma^2(DEC)].$$

$$\sigma^2_{py} = \frac{1}{s_y^2}[\sigma^2(RA) + \sigma^2(DEC)].$$

s_x and s_y are image scales in the x and y image plane respectively given in units of degrees/pixel and $\sigma(RA)$, $\sigma(DEC)$ are pointing uncertainties given by the header keywords CRDER1 and CRDER2 in degrees. The image scale parameters are also taken from the FITS headers (namely the CDELT1 and CDELT2 keywords).

Uncertainties in the translational shifts (X_T , Y_T) are given by the following

$$\sigma(X_T) = \sqrt{\sigma^2_{px} + [(x_1 - x_c)\sin\theta - (y_1 - y_c)\cos\theta]^2 \sigma^2(\theta)} \quad (10)$$

$$\sigma(Y_T) = \sqrt{\sigma^2_{py} + [(x_1 - x_c)\cos\theta + (y_1 - y_c)\sin\theta]^2 \sigma^2(\theta)} \quad (11)$$

As defined in equations (7) and (8), [x_1 , y_1] ≡ [CRPIX1, CRPIX2] and [x_c , y_c] are input-image coordinates of the rotational center. It is important to note that if one chooses values for [x_c , y_c] such that $x_c \neq x_1$ and $y_c \neq y_1$, the second term in equations (10) and (11) with the $\sigma(\theta)$ dependence will dominate leading to abnormally large errors. It is therefore suggested that if one wants to minimize uncertainties, that [x_c , y_c] = [CRPIX1, CRPIX2] be chosen. This is specified by setting the namelist parameter CenterFlag = 1.

4.1.7. Delta-File (IPAC Table) Output

All relevant keywords that define the reference frame, user-specified parameters, rotational and translational offsets, uncertainties and coordinates of the rotational center for all input images are output to what's called a delta-file in IPAC table format. A detailed description of the output is given in section 6.1.2.

4.1.8. Termination

Summary output is appended to the log file (the log file is created if previously non-existent), which includes diagnostic reports for the Q/A Subsystem and the appropriate exit code issued to be picked up by the pipeline executive. A detailed list of log file contents is given in Section 6.1.1.

5. Algorithm

5.1. Algorithm Specifics

The algorithm used by DELTAFPOINT has been adequately described in the previous section. As a detail, the WCS software library supports 26 different map projections with which to perform coordinate transformations. Pointing keywords need not conform to the standard FITS conventions (e.g. CRVAL, CRPIX, CROTA2 etc..), as long as they are recognised by the WCS library. In general, all celestial coordinates are measured in degrees with $0 \leq RA \leq 360°$, $-90° \leq DEC \leq 90°$ and $0 \leq CROTA2 \leq 360°$ (the position angle measured East from North).

5.2. Assumptions and Requirements

A. DELTAFPOINT assumes that each FITS image in the input list has a *single FITS header* that defines a unique pointing for a single plane contained therein, in other words, with the standard FITS keyword values: NAXIS = 2 <u>or</u> NAXIS3 = 1. If this is not true, the program will abort with a message sent to standard output.

B. Each image in the input list may have a different pixel scale, i.e. different values for the standard CDELT FITS keywords. Also, they may have different dimensions as specified by NAXIS1 and NAXIS2. They can also be non-square with NAXIS1 ≠ NAXIS2.

C. All specified pixel scales for the reference frame (namelist parameters: Xpixscale, Ypixscale) must be greater than zero.

D. FITS images in the input list (namelist parameter: FITS_Image_List_Filename) are listed one per line and do not have to be in any specific order.

E. The maximum number of images allowed in the input list is currently 3000. This is defined by the MAX_NUMBER_IMAGES parameter in the include file *deltafpoint.h*.

F. If a single reference image is specified in the namelist or command-line (FITS_Ref_Image_Filename), this image *need not* be listed in the input image list.

G. When offsets are computed relative to an average pointing of all frames (RefFrameFlag = 1), and if CdeltFlag = 1 is specified in the namelist or command-line (indicating that the pixel scale to use will be from the input FITS header) then the CDELT values from the first image in the input list are used as defaults.

H. DELTAFPOINT searches FITS headers for the keywords CRDER1 and CRDER2 which specify the uncertainties corresponding to CRVAL1 (RA) and CRVAL2 (DEC) respectively. If these are not found, they default to the value zero and a message is sent to standard output to indicate this.

I. If one chooses to use a set of RA, DEC FITS keywords other than the standard CRVAL1 and CRVAL2, they must set PointingFlag = 2. Following this, they must also set the keyword strings "RAkeyword" and "DECkeyword". If neither of these are specified, the program will abort with a message sent to standard output.

6. Output

6.1. DELTAFPOINT Output

DELTAFPOINT is capable of generating the following output:

A.) Standard-output processing and status messages.

B.) An IPAC table file (the "delta-file"). See Section 6.1.2 for details.

C.) A log file containing processing statistics, status messages and ancillary information.

All DELTAFPOINT disk output is written to the pathnames that are specified with the output filenames in the command-line or namelist inputs.

6.1.1 DELTAFPOINT Log-File Output

The information stored in the log file at the output of this program includes: program name and version number, values of all namelist and/or command-line inputs, a message indicating the type of calculation performed, status code, processing time, date and time, and a message indicating program termination.

6.1.2 Delta-File (IPAC-Table) Output

The primary output from DELTAFPOINT is a table in IPAC format which consists of two parts: first, the table header and second, the relative offsets for each image. The table header lists: the program name and version number, time and date the table was generated, input image list filename, filename of single reference image if specified, otherwise a message indicating that an average reference of all frames was computed, pointing information of the single reference image if specified, otherwise these are replaced by values that define the average computed reference frame.

The table portion lists in column order left to right: image index label, image filename (including path), offset in *x*- and *y*-directions in units of reference frame pixels with scales (in degrees/pixel) indicated in the table header (i.e. by CDELT1 and CDELT2), rotational offset in degrees, uncertainties in these three quantities, and last, *x,y* coordinates of the rotational center for each individual input image from the input list.

Below are two examples of delta-file outputs corresponding to two different frames of reference for the relative offsets where the numerical values are not necessarily realistic.

Offsets relative to single input image:

```
\character Delta_File_Program = Output from deltafpoint, version  2.00
\character Creation_Date_Time = Tue Jul 24 18:44:53 2001
\character Input_Image_List = ./testing/deltafpoint_images.list3
\character Reference_Image = ./testing/sws4322a00205.fits
\integer Number_of_Frames = 9
\real RefFrCRVAL1 = 82.218150
\real RefFrCRVAL2 = 35.836323
\real RefFrCRPIX1 =   64.5
\real RefFrCRPIX2 =   64.5
\real RefFrCROTA2 = -0.092466
\real RefFrCDELT1 = -0.004323
\real RefFrCDELT2 = 0.004297
|Index   |Filename           |Xshift     |Yshift     |Rotation   |err_Xshift |err_Yshift |err_Rot
|Xcenter |Ycenter |
|i       |c                  |r          |r          |r          |r          |r          |r          |r
|r       |
        1 sws4322a00201.fits      -2.488      -5.856     0.00030      0.136      0.137     0.17375
64.5     64.5
        2 sws4322a00202.fits      -5.731     -11.004     0.00325      0.136      0.137     0.17375
64.5     64.5
        3 sws4322a00203.fits      -2.399      -1.482     0.01365      0.136      0.137     0.17375
64.5     64.5
        4 sws4322a00204.fits      -2.559      16.615     0.00924      0.136      0.137     0.17375
64.5     64.5
        5 sws4322a00205.fits       0.000       0.000     0.00000      0.000      0.000     0.00000
64.5     64.5
        6 sws4322a00206.fits     -24.193       4.841     0.01471      0.136      0.137     0.17375
64.5     64.5
        7 sws4322a00207.fits       3.275      -5.591     0.00829      0.136      0.137     0.17375
64.5     64.5
        8 sws4322a00208.fits      12.851      -6.618     0.01152      0.136      0.137     0.17375
64.5     64.5
        9 sws4322a00209.fits      18.627      -4.224     0.01690      0.136      0.137     0.17375
64.5     64.5
```


THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

Offsets relative to an average of all input images:

```
\character Delta_File_Program = Output from deltafpoint, version  2.00
\character Creation_Date_Time = Tue Jul 24 18:44:53 2001
\character Input_Image_List = ./testing/deltafpoint_images.list3
\character Reference_Image = Average of all input images
\integer Number_of_Frames = 9
\real RefFrCRVAL1 = 82.218150
\real RefFrCRVAL2 = 35.836323
\real RefFrCRPIX1 =   64.5
\real RefFrCRPIX2 =   64.5
\real RefFrCROTA2 = -0.092466
\real RefFrCDELT1 = -0.004323
\real RefFrCDELT2 = 0.004297
|Index   |Filename          |Xshift     |Yshift     |Rotation   |err_Xshift |err_Yshift |err_Rot
|Xcenter |Ycenter |
|i       |c                 |r          |r          |r          |r          |r          |r          |r
|r       |
        1 sws4322a00201.fits      -2.488      -5.856     0.00030       0.136       0.137     0.17375
64.5     64.5
        2 sws4322a00202.fits      -5.731     -11.004     0.00325       0.136       0.137     0.17375
64.5     64.5
        3 sws4322a00203.fits      -2.399      -1.482     0.01365       0.136       0.137     0.17375
64.5     64.5
        4 sws4322a00204.fits      -2.559      16.615     0.00924       0.136       0.137     0.17375
64.5     64.5
        5 sws4322a00205.fits       3.211       8.093     0.03781       0.163       0.137     0.17375
64.5     64.5
        6 sws4322a00206.fits     -24.193       4.841     0.01471       0.136       0.137     0.17375
64.5     64.5
        7 sws4322a00207.fits       3.275      -5.591     0.00829       0.136       0.137     0.17375
64.5     64.5
        8 sws4322a00208.fits      12.851      -6.618     0.01152       0.136       0.137     0.17375
64.5     64.5
        9 sws4322a00209.fits      18.627      -4.224     0.01690       0.136       0.137     0.17375
64.5     64.5
```


THIS IS A PRELIMINARY DOCUMENT, the module described here may or may not be utilized in the final pipelines as described.

7. Testing

DELTAFPOINT has been successfully unit-tested as a stand-alone program for a variety of different input cases. The tests were designed to check DELTAFPOINT robustness and capability of generating corrected results.

Here is a summary of the unit tests that were conducted:

1. Tested DELTAFPOINT on a list of WIRE images and compared the delta-file output with image offsets computed by an independent program. Image offsets were also computed using routines in the IRAF package.
2. Tested DELTAFPOINT on lists of simulated FITS images by updating all relevant pointing keywords to test different combinations of relative rotations, translations and map projections.
3. Executed DELTAFPOINT with inputs read from and output written to directories different from where the program was run. Both namelist and command-line input mechanisms were exercised.
4. Executed DELTAFPOINT for all combinations of input parameters, in order to test that they function properly.
5. Executed DELTAFPOINT on non-square images.
6. Executed DELTAFPOINT on a list of images differing in pixel scale.
7. Tested DELTAFPOINT using non-standard WCS keywords for RA and DEC specified in the FITS headers (namelist option PointingFlag).

8. Usage Examples

Using a namelist file with verbose (-v) output re-directed to a file "out.log":

```
DELTAFPOINT –n deltafpoint.nl -v | & tee out.log
```

Without using a namelist file (with offsets relative to a **single input frame**):

```
DELTAFPOINT –f image_list.txt –i ref_image.fits –o
delta_file.tbl –g 2 –cf 2 –cx 0 –cy 0 –pf 2 –px 0.005 –py
0.005 –kf 1 –v
```

Without using a namelist file (with offsets relative to the **average pointing of all frames**):

```
DELTAFPOINT –f image_list.txt –o delta_file.tbl –cf 2 –cx 0 –
cy 0 –pf 2 –px 0.005 –py 0.005 –kf 1 –v
```

9. Glossary

DCE Data Collection Event

DN Data Number

IOC	In-Orbit Checkout
IRAF	Image Reduction and Analysis Facility.
SDS	Subsystem Design Specification
SIS	Software Interface Specification
TBD	To Be Determined
TBR	To Be Resolved
WCS	World Coordinate System
WIRE	Wide-Field Infrared Explorer