### TIRS Alignment Calibration Algorithm

#### Background/Introduction

The TIRS alignment calibration algorithm combines the functions of the OLI sensor alignment and focal plane alignment calibration algorithms. Using an OLI SWIR band image as a reference it compares an SCA-separated precision and terrain corrected (L1T) TIRS 10.8 micrometer band image (see note #1) with the OLI SWIR reference image. Each SCA in the TIRS L1T image is compared to the SCA-combined OLI reference to measure both systematic full-scene TIRS-to-OLI misregistration and SCA-specific deviations from the scene-average registration. The measured deviations are used to estimate corrections to the TIRS-to-OLI alignment matrix and to the 10.8 micrometer band Legendre polynomial coefficients that model the nominal lines-of-sight for each SCA.

The algorithm is implemented in two steps: 1) a mensuration/setup step in which the separated-SCA L1T image is correlated with the OLI reference image to measure the within-SCA deviations, and; 2) a calibration update computation step in which the measured deviations are used to compute TIRS-to-OLI alignment corrections and TIRS line-of-sight model correction Legendre coefficients that adjust the original LOS model to minimize the residual image deviations. The calibration update step includes applying an outlier filter to the image measurements. Separating the algorithm into two distinct steps makes it possible to run the calibration update step multiple times, using different outlier filter thresholds, for example, without having to perform the time consuming image mensuration/correlation setup procedure more than once.

Results from individual calibration scenes are stored in the geometric trending database so that results from multiple scenes can be analyzed together when deciding whether and how to adjust the operational ACS-to-TIRS alignment and TIRS focal plane calibrations. If a 10.8 micrometer band focal plane calibration update is generated, the TIRS 12.0 micrometer spectral band would subsequently be re-registered to the 10.8 micrometer band using the TIRS band alignment calibration procedure.

The TIRS alignment calibration procedure is derived from the OLI focal plane calibration algorithm. The implementation should be very similar for the setup step, which measures the SCA-specific deviations relative to the reference image. The legendre step, which calculates the Legendre polynomial coefficient updates, will be enhanced to include the computation of the full-scene TIRS-to-OLI alignment update. Since the Legendre coefficients and alignment angles are not completely independent, some additional constraints are required to make the parameters separable. The default approach is to constrain the Legendre coefficients so that they cannot model roll, pitch, or yaw effects (more about this below). An alternate option is to constrain the alignment angles. This option makes the calibration solution mimic the heritage focal plane calibration procedure.

#### Dependencies

The TIRS alignment calibration algorithm assumes that the L1T process flow has created a substantially cloud-free SCA-separated (nadir-viewing) path-oriented L1T 10.8 micrometer band image, over a band registration calibration site, which has been registered to an OLI reference image either by using systematic LOS models for both the OLI and TIRS images, or by transferring the OLI-derived precision correction model to the TIRS LOS model. This would be accomplished by copying the OLI precision correction parameters from the OLI LOS model to the TIRS LOS model. The SCA-separated TIRS L1T image will be framed to exactly match the SCA-combined OLI reference image, by using the TRANSFER\_FRAME framing option during TIRS LOS projection grid generation. This algorithm also assumes that the CPF, TIRS LOS model, TIRS grid file, and DEM used to produce the TIRS L1T image, are available.

#### Inputs

The TIRS alignment calibration algorithm uses the inputs listed in the following table. Note that some of these “inputs” are implementation conveniences (e.g., using an ODL parameter file to convey the values of and pointers to the input data). The second column shows which algorithm step (image mensuration or correction model computation) uses the input.

|  |  |
| --- | --- |
| **Algorithm Inputs** | **Processing Step** |
| ODL File (implementation) | Both |
| CPF Name | Both |
| TIRS L1T Image File Name (see note #2) | Step 1 |
| TIRS LOS Model File Name | Step 1 |
| TIRS LOS Projection Grid File Name | Step 1 |
| DEM File Name | Step 1 |
| OLI Reference Image File Name | Step 1 |
| Correlation Data File Name | Both |
| Report File Name | Step 2 |
| Processing Parameters | Both |
| Number of Tie Points per Cell | Step 1 |
| Outlier tolerance | Step 2 |
| Constraint Type:  0 (default) = constrain Legendre coefficients (solve all parameters),  1 = constrain angles (solve Legendre only) | Step 2 |
| Work order ID (for trending) | Step 2 |
| WRS Path (for trending) | Step 2 |
| WRS Row (for trending) | Step 2 |
| Calibration effective dates for updated parameters | Step 2 |
| Trending flag | Step 2 |
| CPF | Both |
| Calibration effective dates | Step 2 |
| ACS-to-OLI alignment matrix (3x3 orientation matrix) | Step 2 |
| ACS-to-TIRS alignment matrix (3x3 orientation matrix) | Step 2 |
| Algorithm Parameters (formerly system table parameters) |  |
| Size of Correlation Window | Step 1 |
| Peak Fit Method | Step 1 |
| Min Correlation Strength | Step 1 |
| Max Correlation Displacement | Step 1 |
| Fill Threshold Fraction (max percent of window containing fill value) | Step 1 |
| Tie point weight (in units of 1/microradians2) | Step 2 |
| Alignment constraint weight (in units of 1/microradians2) (new) | Step 2 |
| Fit order | Step 2 |
| Post-fit RMSE Thresholds (trending metrics) (in units of microradians) | Step 2 |
| TIRS Grid File (see LOS Projection ADD for details) | Step 1 |
| Number of SCAs | Step 1 |
| For each SCA: | Step 1 |
| Grid cell size in lines/samples | Step 1 |
| Number of lines/samples in grid | Step 1 |
| Number of z-planes, zero z-plane index, z-plane spacing | Step 1 |
| Array of grid input line/sample locations | Step 1 |
| Array of output line/sample locations (per z-plane) | Step 1 |
| Array of forward mapping coefficients | Step 1 |
| Array of inverse mapping coefficients | Step 1 |
| Rough mapping polynomial coefficients | Step 1 |
| TIRS LOS Model File | Step 1 |
| TIRS Along-Track IFOV (in radians) | Step 1 |
| Number of SCAs | Step 1 |
| Number of Bands | Step 1 |
| Number of Detectors per SCA per Band | Step 1 |
| Focal Plane Model Parameters (Legendre Coefficients) (in radians) | Step 2 |
| ACS-to-TIRS Alignment Matrix (3x3 orientation matrix) (see note #5) | Step 2 |
| TIRS L1T Image (separated SCA) | Step 1 |
| Image corner coordinates | Step 1 |
| Pixel size (in meters) | Step 1 |
| Image size | Step 1 |
| Search image pixel data (10.8 micrometer band) | Step 1 |
| DEM | Step 1 |
| DEM corner coordinates | Step 1 |
| Pixel size (in meters) | Step 1 |
| DEM size | Step 1 |
| Elevation data | Step 1 |
| OLI Reference Image | Step 1 |
| Image corner coordinates | Step 1 |
| Pixel size (in meters) | Step 1 |
| Image size | Step 1 |
| Reference image pixel data (SWIR1 or SWIR2 band) | Step 1 |
| Correlation Data File (output of Step 1) | Step 2 |
| Correlation results in TIRS input space pixels | Step 2 |
| LOS errors in radians | Step 2 |
| Correlation results in output space pixels | Step 2 |

#### Outputs

|  |
| --- |
| Step 1: TIRS Alignment Setup |
| Correlation Data File (temporary output passed to Update step) |
| Correlation results in output space pixels |
| Correlation results in TIRS input space pixels |
| LOS errors in radians |
| Step 2: TIRS Alignment Update (see Table 6‑35 below) |
| Report File (see Table 6‑35 below for details) |
| Standard report header |
| Acquisition date |
| Ref (OLI)/Search (TIRS) image names |
| Constraint Type (Legendre or Angle) |
| Original TIRS-to-OLI roll-pitch-yaw |
| Estimated TIRS-to-OLI roll-pitch-yaw correction |
| Updated TIRS-to-OLI roll-pitch-yaw |
| Original TIRS-to-OLI alignment matrix (3x3) |
| Updated TIRS-to-OLI alignment matrix (3x3) |
| Number of SCAs |
| For each SCA: |
| SCA Number |
| Old Along- and Across-track Legendre coefficients (NSCAx2x4) |
| Along- and Across-track Legendre error (fit) coefficients (NSCAx2x4) |
| New Along- and Across-track Legendre coefficients (NSCAx2x4) |
| Pre-fit along- and across-track offset statistics (mean, stddev, RMSE) |
| Post-fit along- and across-track residual statistics (mean, stddev, RMSE) |
| Confidence level used for outlier rejection |
| Legendre polynomial fit order (see note #3) |
| Number of tie points used for current SCA |
| CPF LOS\_LEGENDRE and ATTITUDE\_PARAMETERS Groups  (separate ODL-format files) (see note #6) |
| Effective Dates (embedded in output file names) |
| New ACS-to-TIRS alignment matrix (3x3) |
| New Legendre polynomial coefficients (NSCAx2x4) |
| Measure Tie Point Data |
| For each point: |
| SCA Number |
| Grid Cell Column Number |
| Nominal Output Space Line |
| Nominal Output Space Sample |
| Measured LOS Error Delta Line (in pixels) |
| Measured LOS Error Delta Sample (in pixels) |
| Measured LOS Error Along-Track Delta Angle (in microradians) |
| Measured LOS Error Across-Track Delta Angle (in microradians) |
| Tie Point State (outlier) Flag |
| Along-Track Fit Residual (in microradians) |
| Across-Track Fit Residual (in microradians) |
| TIRS Alignment Trending Database (see Table 6‑35 below for details) |
| Geometric Characterization ID |
| Work Order ID |
| WRS path/row |
| Acquisition date |
| Ref (OLI) image name |
| Constraint Type (Legendre or Angle) |
| Original TIRS-to-OLI roll-pitch-yaw |
| Estimated TIRS-to-OLI roll-pitch-yaw correction |
| Updated TIRS-to-OLI roll-pitch-yaw |
| Number of SCAs |
| For each SCA: |
| SCA Number |
| Old Along- and Across-track Legendre coefficients (NSCAx2x4) |
| Along- and Across-track Legendre error (fit) coefficients (NSCAx2x4) |
| New Along- and Across-track Legendre coefficients (NSCAx2x4) |
| Pre-fit along- and across-track offset statistics (mean, stddev, RMSE) |
| Post-fit along- and across-track residual statistics (mean, stddev, RMSE) |
| Confidence level used for outlier rejection |
| Number of tie points used for current SCA |

#### Options

TIRS Alignment Calibration Trending On/Off Switch

#### Prototype Code

Input to both executables is an ODL file containing all parameters needed by both programs; outputs are a binary tie point mensuration file (used internally only), an ASCII report file, two ASCII ODL-formatted CPF fragments, and trending data written to the stdout and captured in an ASCII log file.

The prototype code was compiled with the following options when creating the test data files:

-g -Wall –O2 -march=nocona -m32 –mfpmath=sse –msse2

The code units of the prototype implementation are briefly described here. Additional details are provided below for units that perform core algorithm processing logic.

TIRS\_Align\_Setup.c - This routine is the main driver for the setup portion of the TIRS alignment calibration.

get\_tirs\_align\_parms.c - This function gets the input parameters from the ODL parameter file. It is used by both the setup and legendre executables.

check\_images\_match.c - This function checks to make sure the TIRS search, OLI reference, and DEM images all match. The corners of all the images should match to within half a pixel, and the reference and search images should be the same resolution (pixel size) and the same size. This function is an initial check to make sure that all the images are consistent before correlation is attempted. This function assumes the OLI reference image and the DEM are one-band images.

set\_up\_grid.c - This function reads the grid file into the grid data structure. The whole grid structure is returned so the caller can free all memory allocated when the grid was read using the grid deallocation call.

select\_corr\_pts.c - This function selects nominal correlation points evenly distributed about the center point of each grid cell (output space).

calc\_input\_space\_errors.c - This function calculates the errors in input space pixels. This is done by first correlating in output space and converting the correlated locations to input space.

perform\_correlation.c - This function performs the normalized gray-scale correlation at each point. It does this by invoking the correlation library routines described in the GCP Correlation Algorithm Description Document (e.g., math\_submit\_chip\_to\_corr).

map\_coords\_to\_input\_space.c - This function uses the inverse mapping coefficients in the grid to calculate the input space line/sample for each output space line/sample.

calc\_los\_errors.c - This function uses the tie point reference and search input space locations to calculate the angular line-of-sight errors.

output\_correlation\_info.c - This function writes the tie point correlation results to a file.

Generate\_Legendre\_Polynomials.c - This routine is the main driver for the Legendre polynomial and TIRS alignment angle estimation portion of TIRS alignment calibration.

read\_correlation\_info.c - This function reads the correlation information from the file generated by the output\_correlation\_info routine above.

filter\_outliers.c - This function separates the correlation data into groups for each SCA for the X (sample) and Y (line) directions. It then finds the standard deviation for the points in each group. Outlier rejection is then performed on the points based on the tolerance selected by the user and the Student's T distribution. This procedure is described in the OLI Geometric Accuracy Assessment Algorithm Description Document.

calculate\_point\_weights.c - This function calculates the weight associated with each correlation point for doing the Legendre polynomial and TIRS alignment angle fit. Currently, this routine assigns the weight passed in to each point, effectively assigning each point an equal weight. Originally, it was thought that the correlation strength would factor into the weight, but that was determined to not be needed. This routine was left in to allow point-specific weight factors to be added at a later date.

fit\_polynomials.c - This function performs the weighted least-squares fit of the correlation data points (using the angular error) to find the TIRS alignment angle and Legendre error polynomials. Note that unlike the heritage OLI focal plane calibration procedure, TIRS alignment calibration performs a simultaneous solution for all parameters. This is necessary because the TIRS alignment angles are correlated with Legendre polynomial terms, linking the solutions for the three TIRS SCAs.

calculate\_post\_fit\_residuals.c - This function calculates the residual statistics after the TIRS alignment angles and Legendre polynomial coefficients have been fit. This unit includes the calc\_legendre\_poly() function that calculates the Legendre polynomial for the input normalized detector value.

calculate\_alignment\_matrix.c – Computes the 3-by-3 alignment matrix corresponding to a set of 3 roll-pitch-yaw alignment angles.

create\_tirs\_alignment\_report.c - This function generates a file reporting the results of the global TIRS alignment angle and SCA-specific Legendre polynomial fit calculations. The report file contents are shown in Table1 below. This unit also includes the write\_coeffs() function that writes an entire set of coefficients to the indicated output file.

trending\_dummy.c - This function is a placeholder for the logic that will write the results of the TIRS alignment angle and SCA Legendre polynomial fit calculations to the geometric characterization database. In the prototype implementation the actual database output is replaced by dummy ASCII output to stdout.

update\_alignment\_parameters\_cpf.c – Applies the computed TIRS-to-OLI alignment corrections to the current TIRS-to-OLI alignment matrix calculated from the CPF Attitude\_To\_OLI\_Matrix and Attitude\_To\_TIRS\_Matrix parameters, and then uses the updated TIRS-to-OLI alignment to compute an updated Attitude\_To\_TIRS\_Matrix calibration parameter set.

write\_alignment\_matrix\_ODL.c - This function writes the updated ATTITUDE\_PARAMETERS parameter group of the CPF, in the ODL format used by the CPF, to an output file. This group contains the updated Attitude\_To\_TIRS\_Matrix alignment parameters.

write\_SCA\_parameters\_cpf.c - This function writes the updated LOS\_LEGENDRE parameter group of the CPF, in the ODL format used by the CPF, to an output file.

#### Procedure

The TIRS Alignment Calibration Algorithm is used for on-orbit calibration of the TIRS-to-OLI instrument alignment as well as for the alignment of the lines-of-sight of the TIRS SCAs relative to each other. This calibration is necessary to meet the TIRS-to-OLI band registration, and the TIRS image registration, geodetic accuracy, and geometric accuracy requirements.

*Procedure Overview*

The TIRS alignment algorithm adjusts the overall TIRS field of view and each TIRS SCA to an OLI reference image. By simultaneously aligning the TIRS SCAs to a common reference, any measured inter-SCA misalignment is removed. Each TIRS SCA is correlated against a reference image created from an OLI SWIR band, acquired at approximately the same time. A new set of 3rd order Legendre LOS coefficients, representing updates or corrections to the original polynomials, are generated by fitting a set of coefficients to the measured LOS deviations. A set of roll-pitch-yaw angular adjustments are also computed to remove any alignment biases between the TIRS and OLI instruments.

Substantially cloud free scenes should be used for TIRS alignment calibration. The imagery should have ground control applied and terrain displacements removed, i.e., the imagery should be a terrain corrected (L1T) data set. Both the OLI SWIR and TIRS images should be path oriented and resampled to 30m output pixel size.

Stage 1: Setup – Correlate L1T Image with OLI Reference

An array of test points is generated for each TIRS SCA based upon the number of points per grid cell specified in the input parameters. The TIRS LOS projection grid is used to generate the test point array by spacing the test points at regular intervals in TIRS input space, and then computing the corresponding output space coordinates for each. Constructing the test point array in the TIRS input space ensures that the test points fall within the active area of each TIRS SCA.

Image windows extracted from the L1T image at the test point locations are correlated with corresponding windows extracted from the reference OLI SWIR image, using normalized gray scale correlation. This procedure is the same as that described in the OLI Focal Plane Calibration Algorithm Description Document. Since the expected offsets are small, the TIRS L1T and OLI SWIR image windows are the same size. The correlation procedure yields measured deviations (or correlation failure flag) in the line and sample directions, estimated to subpixel accuracy. These measured LOS deviations are in units of output space pixels.

The deviations measured in output space are converted to differences in LOS along- and across-track angles by mapping the reference point location from output space to TIRS input space and then mapping the search point location from output space to TIRS input space. The mappings are performed using the TIRS LOS projection grid that was used to resample the L1T image, and include the test point elevation interpolated from the input DEM. This three-dimensional output space to input space mapping (3d\_ols2ils) is described in the TIRS Image Resampling Algorithm Description Document. This logic is identical to that used for OLI focal plane calibration. Once a TIRS input space location is found for both points, the LOS vectors are calculated for each input sample location using the TIRS LOS model. This is described in the Find LOS section of the TIRS LOS Projection Algorithm Description Document.

The angular LOS offsets in TIRS input space are then:

 (1-1)

 (1-2)

where:

rx,ry,rz = reference x,y,z vector components of LOS

sx,sy,sz = search x,y,z vector components of LOS

input reference line = input line location for reference point

input search line = input line location for search point

Stage 2: Update – Compute TIRS Alignment Calibration Update

A constrained least squares solution is used to generate the fit between the angular offsets and the corrections to the TIRS alignment angles and per-SCA Legendre polynomial coefficients. Constraints are necessary to separate the alignment angle estimates from the Legendre polynomial coefficients since the angular effects could be largely absorbed by the Legendre polynomials. The constraints are implemented as supplemental observations that enforce relationships between solution parameters.

There are two options for constraining the parameters. The first, default, option is to constrain the Legendre coefficients so that they do not attempt to model the rotation effects. The second option is to constrain the angular corrections to be zero. This effectively reduces the TIRS alignment calibration solution to be a simple focal plane calibration solution, similar to the OLI heritage. Both options use three additional constraint observations (described in more detail below) with an associated 3-by-3 constraint weight matrix. The weight matrix is a diagonal matrix with the diagonal terms containing a common weight, read from the CPF, for all 3 constraints.

Unlike the OLI focal plane calibration procedure, which calibrates one SCA at a time, the TIRS alignment solution is simultaneous so that both the SCA-specific Legendre coefficients and the global alignment angles can be estimated. This also requires that both the along- and across-track Legendre coefficients be solved for at the same time. There are thus 27 unknowns to be solved for: 3 alignment angles + 3 SCAs \* ( 4 along-track Legendre coefficients + 4 across-track Legendre coefficients).

There are Nk tie point observations in SCA k (k=1,2,3), so the total number of observations is N = N1 + N2 + N3. The observation matrix is an Nx2 matrix containing the measured X and Y offsets in TIRS input space angular units. Note that, unlike the OLI focal plane calibration, there is a single observation matrix containing both the along-track (X) offsets and the across-track (Y) offsets.

 (2-1)

The design matrix is an 2Nx27 matrix with each row of the matrix containing the alignment angle partial derivatives and the Legendre polynomial terms associated with the reference sample location of the corresponding tie point measurement. The calculation of these angle partial derivatives and Legendre polynomial terms, as functions of the input sample location, is described below.

 (2-2)

Where A0 and A1 are submatrices defined as:



And where:

*x’j* = the x coordinate of tie point j, evaluated using the current estimate of the Legendre polynomials.

*y’j* = the y coordinate of tie point j, evaluated using the current estimate of the Legendre polynomials.

*l*i,j = ith Legendre polynomial term associated with the tie point location of the jth measurement.

The Legendre polynomial terms (contained in the 2x8 A1,j submatrix) will be in the set of 8 columns associated with the SCA containing tie point j. Thus, columns 4-11 are associated with SCA 1, columns 12-19 are associated with SCA 2, and columns 20-27 are associated with SCA 3. In equation (2-2) tie points 1 through N1 fall in SCA 1, points N1+1 through N1+N2 fall in SCA 2, and points N1+N2+1 through N (=N1+N2+N3) fall in SCA 3. Note that the partial derivative of the X offset with respect to the yaw correction is the tie point Y coordinate and the partial derivative of the Y offset with respect to the yaw correction is the –X tie point coordinate. Thus, the y’j coordinate appears in the first (X observation) row of the A0 submatrix and the –x’j coordinate appears in the second (Y observation) row of the A0 submatrix.

The tie point observations are weighted by a diagonal weight matrix. The weight matrix [**Wt**] is a 2Nx2N diagonal matrix where the diagonal elements are the tie point weight value read from the CPF.

 (2-3)

where *wt* = tie point weight

The weight matrix is included to make it possible to differentially weight the measured deviations based on correlation strength, but this is not implemented in the baseline algorithm. Instead, a common weight, read from the CPF, is used for all points. This weight value provides a relative weighting of the tie point observations relative to the constraints.

The form of the constraint observations depends upon the constraint option selected. For the default case, the Legendre coefficient adjustments are constrained to prevent them from modeling angular alignment effects. This is done by considering the effects of angular alignment changes at the center of each SCA. The x and y coordinates at the center of SCA k are as follows:

*x0’*k = ax0k – ax2k / 2 (2-4)

*y0’*k = ay0k – ay2k / 2

where: ax0k and ax2k are the first and third x Legendre coefficients for SCA k, and

ay0k and ay2k are the first and third y Legendre coefficients for SCA k.

Therefore, the Legendre corrections at the center of each SCA are as follows:

*x0’*k = ax0k – ax2k / 2 (2-5)

*y0’*k = ay0k – ay2k / 2

where: ax0k and ax2k are the first and third x Legendre corrections for SCA k,

ay0k and ay2k are the first and third y Legendre corrections for SCA k.

To control the modeling of alignment effects in the Legendre polynomials we constrain the Legendre adjustments to the locations of the SCA center points such that the mean center point x (pitch) and y (roll) locations do not change and such that the x (along-track) coordinates of the center points of the two outboard SCAs (1 and 3) do not change in opposite directions (yaw):

ay01 + ay02 + ay03 – (ay21 + ay22 + ay23)/2 = 0 (roll)

ax01 + ax02 + ax03 – (ax21 + ax22 + ax23)/2 = 0 (pitch) (2-6)

ax01 – ax21/2 – ax03 + ax23 / 2 = 0 (yaw)

The corresponding constraint matrix is:

 (2-7)

The second constraint option fixes the alignment angle adjustments at zero, allowing all adjustments to be modeled through the Legendre coefficients. The corresponding constraint matrix is:

 (2-8)

The constraint weight matrix [Wc] is a 3-by-3 diagonal matrix containing the constraint weight read from the CPF:

 (2-9)

where *wc* = constraint weight

Note that the observation matrix is zero for the constraints no matter which constraint option is selected.

The solution for the updates to the TIRS alignment angles and to the Legendre LOS coefficients can be found from:

 (2-10)

The matrix [] is a 27x1 vector containing the corrections to be applied to the alignment angles and to the Legendre coefficients. These corrections are added to the original alignment angles and Legendre LOS coefficients to compute the updated TIRS alignment parameters.

   (2-11)

The 10.8 micrometer band is used for TIRS alignment. TIRS band calibration uses the 10.8 micrometer band as the reference for the 12.0 micrometer band. A TIRS band alignment calibration should be performed following an update to the TIRS alignment calibration to avoid degrading the band-to-band registration.

Figure 6‑77 shows the architecture for the setup portion of the TIRS Alignment Calibration algorithm.

Retrieve Parameters

Check Images

Match

Read LOS Model

Create Tie Points

ODL

Reference

Image

Correlation

Results

Check Fill

Correlate Chips

L1T

Image

DEM

Read Projection Grid

Read CPF

Process Tie Points

Map Coordinates to

Input Space

Calculate LOS

Errors

Write Correlation

Information

Extract Windows

Precision

Projection

Grid

CPF

Precision

LOS

Model

DEM

Figure 6‑77. TIRS Alignment Calibration Setup Algorithm Architecture

Figure 6‑78 shows the architecture of the alignment solution portion of the TIRS Alignment Calibration algorithm.

Retrieve Parameters

Read LOS Model

Read Correlation

Results

ODL

Correlation

Results

Calculate Along-Track

Legendre Polynomials

Calculate Post-Fit

Residuals

Filter Outliers

Calculate Post-

Correction Statistics

Write Alignment

Report

Precision

LOS

Model

TIRS

Alignment

Report

Read CPF

CPF

Process Tie Point

Observations

Compute Corrections

Write CPF

DB

Calculate Across-Track

Legendre Polynomials

Write DB Output

Process Constraints

Figure 6‑78. TIRS Alignment Calibration Update Algorithm Architecture

**TIRS Alignment Cal Setup Sub-Algorithm (TIRS\_Align\_Setup)**

This routine is the main driver for the setup portion of the TIRS alignment calibration. The setup portion consists of correlating points between the search TIRS image and the reference OLI SWIR image, and converting the correlation offsets into line-of-sight deviations that can be used to correct each SCA's detector array, modeled by a cubic Legendre polynomial. The results of this program are used in the second portion of TIRS alignment calibration, the generation of new TIRS-to-OLI alignment angles and TIRS SCA Legendre polynomials. This program creates a temporary output file that is read by the second portion of TIRS alignment calibration.

**Get TIRS Alignment Parameters Sub-Algorithm (get\_tirs\_align\_parms)**

This function gets the input parameters from the input parameter files.

**Check Images Match Sub-Algorithm (check\_images\_match)**

This function checks to make sure the L1T TIRS search, OLI L1T SWIR reference, and DEM images all match. The corners of all the images should match to within half a pixel, and the reference and L1T search image should be the same resolution (pixel size) and the same size. This function is an initial check to make sure that all the images are consistent before correlation is attempted. This function assumes the OLI SWIR reference image and the DEM are one-band images.

**Set Up Grid Sub-Algorithm (set\_up\_grid)**

This function reads the TIRS grid file into the grid data structure. The whole grid structure is returned so the caller can free all memory allocated when the grid was read using the grid deallocation call.

**Select Correlation Points Sub-Algorithm (select\_corr\_pts)**

This function selects nominal correlation points evenly distributed about the center point of each grid cell (output space). To ensure evenly distributed tie point locations during correlation, locations are defined to lie at the center of each resampling grid cell, or sub-cell. There will be pts\_per\_cell equally-spaced points per grid cell. For example, if there are 4 points per cell, they will be placed as shown in Figure 6‑79.

Figure 6‑79. Correlation Point Placement in Grid Cell

The cell is divided into a 2-by-2 grid of 4 sub-cells, and each sub-cell is divided in half to place the point in the middle, yielding points at (0.25,0.25), (0.75,0.25), (0.25,0.75), and (0.75,0.75).

The calculation of the output space line/sample coordinates of the tie points is done as follows:

a) Compute number of rows and columns of tie points in each cell.

ncol = (int)ceiling( sqrt(pts\_per\_cell) )

nrow = (int)ceiling( (double)pts\_per\_cell/(double)ncol )

This creates an array of tie points containing at least pts\_per\_cell points.

b) For each tie point, i = 1 to ncol and j = 1 to nrow:

b1) Compute the grid cell fractional location (cfrac,rfrac).



b2) Compute the output space line, olij, using bilinear interpolation on the output line numbers at the grid cell corners, where lUL, lUR, lLL, and lLR are the output space line coordinates at the grid cell upper-left, upper-right, lower-left, and lower-right corners, respectively:

olij = lUL \* (1-cfrac)\*(1-rfrac)

+ lUR \* cfrac \* (1-rfrac)

+ lLL \* (1-cfrac) \* rfrac

+ lLR \* cfrac \* rfrac

b3) Compute the output space sample, osij, using bilinear interpolation on the output sample numbers at the grid cell corners, where sUL, sUR, sLL, and sLR are the output space sample coordinates at the grid cell upper-left, upper-right, lower-left, and lower-right corners, respectively:

osij = sUL \* (1-cfrac)\*(1-rfrac)

+ sUR \* cfrac \* (1-rfrac)

+ sLL \* (1-cfrac) \* rfrac

+ sLR \* cfrac \* rfrac

Note that the bilinear weights are the same for the line and sample computations and only need be computed once.

The heritage version of this sub-algorithm locates the points by computing the intersection of the cell diagonals and then calculating offsets from that point. It is more straightforward to simply use bilinear interpolation, as described above, to calculate the tie point output space coordinates, so this unit will be reworked from the heritage implementation.

**Calculate Input Space Errors Sub-Algorithm (calc\_input\_space\_errors)**

This function calculates the errors in TIRS input space pixels. This is done by first correlating in output space and converting the correlated locations to input space.

**Perform Correlation Sub-Algorithm (perform\_correlation)**

This function performs the normalized gray-scale correlation at each point. It does this by invoking the correlation library routines described in the GCP Correlation Algorithm Description Document.

**Map Coordinates to Input Space Sub-Algorithm (map\_coords\_to\_input\_space)**

This function uses the inverse mapping coefficients in the grid to calculate the TIRS input space line/sample for each output space line/sample. It does this for both the reference (OLI SWIR) image line/sample location and the search (TIRS L1T) image line/sample location, mapping both to TIRS input space.

a) For each SCA

a1) For each tie point

a1.1) Interpolate a height from the DEM at the location corresponding to the tie point reference image line/sample coordinates (ALIAS xxx\_get\_elevation).

a1.2) Map the reference output line/sample location to its corresponding input line/sample location using axx\_3d\_ols2ils routine

a1.3) Interpolate a height from the DEM at the location corresponding to the tie point search image line/sample coordinates (ALIAS xxx\_get\_elevation).

a1.4) Map the search output line/sample location to its corresponding input line/sample location using the axx\_3d\_ols2ils

**Calculate LOS Errors Sub-Algorithm (calc\_los\_errors)**

This function uses the tie-point reference and searches TIRS input space locations to calculate the angular line-of-sight errors.

b) For each SCA

b1) For each tie point

b1.1) Calculate reference line of sight vector for sample location using the nominal detector type and the precision LOS model (ALIAS axx\_findlos).

b1.2) Calculate the search LOS vector for sample location using the nominal detector type and the precision LOS model.

b1.3) Calculate the deviations in terms of the difference in the LOS along and across-track angles:

along-track LOS error = ref los.x/los.z – srch los.x/los.z

+ input line error \* along-track IFOV

across-track LOS error = ref los.y/los.z – srch los.y/los.z

**Output Correlation Information Sub-Algorithm (output\_correlation\_info)**

This function writes the tie point correlation results to a file. The correlation points are dumped to a binary file, so the second phase of TIRS alignment calibration (TIRS alignment update) can read them directly back in. First, a long integer is written to indicate the number of records, and then all the records are written. Each record contains the following fields:

|  |  |  |
| --- | --- | --- |
| **Type** | **Field** | **Description** |
| int | sca\_number | SCA number (0-relative) |
| int | grid\_column | grid column number (0-relative) |
| int | grid\_row | grid row number (0-relative) |
| double | nom\_os\_pt.line | nominal output space point line |
| double | nom\_os\_pt.samp | nominal output space point sample |
| double | ref\_os\_pt.line | reference output space line |
| double | ref\_os\_pt.samp | reference output space sample |
| double | srch\_os\_pt.line | search output space line |
| double | srch\_os\_pt.samp | search output space sample |
| double | ref\_is\_pt.line | reference TIRS input space line |
| double | ref\_is\_pt.samp | reference TIRS input space sample |
| double | srch\_is\_pt.line | search TIRS input space line |
| double | srch\_is\_pt.samp | search TIRS input space sample |
| double | los\_err.line | angular along-track LOS error |
| double | los\_err.samp | angular across-track LOS error |
| double | los\_err\_pix.line | line LOS error in pixels |
| double | los\_err\_pix.samp | sample LOS error in pixels |
| double | correlation\_accuracy | correlation accuracy |
| ActiveFlag | active\_flag | correlation success flag |
| double | pt\_weight | point weight for use in fit |
| double | fit\_residual.line | line residual from fit of pts |
| double | fit\_residual.samp | sample residual from fit of pts |

This is not a human-readable (ASCII) file, because it is only used to transport information from the first phase of calibration to the second. If the file already exists, it will be overwritten.

**TIRS Alignment Update Sub-Algorithm (Generate\_Legendre\_Polynomials)**

This routine is the main driver for the least squares solution and alignment update portion of TIRS alignment calibration. The TIRS alignment update portion reads the results of the TIRS alignment setup, filters the outliers, fits the data to a set of angular alignment corrections and Legendre polynomial corrections, updates the TIRS-to-OLI alignment and TIRS SCA models, and generates output reports. This process is outlined below.

1. Initialize the normal equation matrix [N] (27x27) and constant vector [L] (27x1) to zero.
2. Process each tie point (j)

b.1) Build design matrix [Aj].

Calculate normalized detector for reference sample location. Note that in this context the “detector” number is the input sample number within the SCA containing the tie point (SCA #k).



where:

detector = reference sample location (0 ... Ndet-1)

number of detectors = number of detectors in current SCA

Calculate two rows of design matrix associated with current tie point:

*l*0,j = 1

*l*1,j = normalized detector

*l*2,j = (3\*(normalized detector)2– 1) / 2

*l*3,j = normalized detector \*(5 \*(normalized detector)2– 3) / 2

where j = tie point number





where:



 if k = i

 otherwise

b.2) Build weight matrix [Wj] using the (fixed) input weight value.



b.3) Build the observation matrix [Bj] from the measured x and y offsets in terms of angular differences.



b.4) Add this observation to the normal equations matrix [N] and to the constant vector [L].

[N] = [N] + [Aj]T [Wj] [Aj]

[L] = [L] + [Aj]T [Wj] [Bj]

1. Add the constraints.

c.1) Form the constraint design matrix [C] based upon the user-selected constraint option, using equation (2-7) or equation (2-8) above.

c.2) Form the constraint weight matrix [Wc] per equation (2-9) above.

c.3) Add the constraint contribution to the normal equations matrix [N]. Note that there is no constraint contribution to the constant vector [L].

[N] = [N] + [C]T [Wc] [C]

1. Solve for alignment angle and Legendre coefficient corrections using weighted least-squares routine (see the Fit Parameters sub-algorithm below).
2. Calculate pre-fit statistics from the original measured deviations.

e.1) Calculate statistics for along- and across-track offsets used in b.3.

Compute mean, standard deviation and RMSE for the offsets, grouping by SCA. Values are calculated for along- and across-track directions independently. For each SCA k=1,2,3:

For along-track offsets in SCA k:

e.1.1) Calculate x mean

e.1.2) Calculate x standard deviation

e.1.3) Calculate x RMSE

For across-track offsets in SCA k:

e.1.4) Calculate y mean

e.1.5) Calculate y standard deviation

e.1.6) Calculate y RMSE

1. Calculate post fit residual statistics for correction coefficients

Post-fit residuals are calculated by updating the original measured offsets/deviations used in step b.3 above using the alignment angle and Legendre polynomial corrections. The differences between the original measurements and the offsets modeled by the correction parameters are the residuals. The post-fit statistics are calculated on these residuals.

f.1) For each tie point

f.1.1) Calculate normalized detector for reference sample location (as shown in b.1) and construct the design matrix as shown in b.1.

f.1.2) Calculate the modeled LOS angle corrections by multiplying the design matrix by the [] solution vector.

f.1.3) Find the residuals as the difference between the original angular offsets and the LOS angle corrections from f.1.2.

f.2) Calculate statistics for the along- and across-track residuals calculated in f.1.

Compute mean, standard deviation and RMSE for residuals, grouping by SCA. Values are calculated for along- and across-track directions independently. For each SCA k=1,2,3:

For along-track residuals:

f.2.1) Calculate x mean

f.2.2) Calculate x standard deviation

f.2.3) Calculate x RMSE

For across-track residuals:

f.2.4) Calculate y mean

f.2.5) Calculate y standard deviation

f.2.6) Calculate y RMSE

1. For each SCA, add the correction coefficients to original Legendre LOS coefficients (see note #4):

new along legendrei,sca = update along legendrei,sca + old along legendrei,sca

new across legendrei,sca = update across legendrei,sca + old across legendrei,sca

where:

i = 0,1,2,3 Legendre polynomial number

sca = SCA number

1. Use the computed roll, pitch, and yaw alignment corrections to update the TIRS-to-OLI rotation matrix.

h.1) Compute the delta rotation matrix [M] from the r, p, and y corrections.



h.2) Combine the delta rotation matrix [M] with the original rotation matrix [TIRS2OLI], derived from the ACS-to-OLI and ACS-to-TIRS matrices in the CPF, to form the updated rotation matrix [TIRS2OLI]’.

[TIRS2OLI] = [ACS2OLI] [ACS2TIRS]T

[TIRS2OLI]’ = [TIRS2OLI] [M]

h.3) Compute original and updated TIRS-to-OLI alignment angles.

Roll = atan( -[TIRS2OLI]3,2 / [TIRS2OLI]3,3 )

Pitch = asin( [TIRS2OLI]3,1 )

Yaw = atan( -[TIRS2OLI]2,1 / [TIRS2OLI]1,1 )

Roll’ = atan( -[TIRS2OLI]’3,2 / [TIRS2OLI]’3,3 )

Pitch’ = asin( [TIRS2OLI]’3,1 )

Yaw’ = atan( -[TIRS2OLI]’2,1 / [TIRS2OLI]’1,1 )

h.4) Compute the updated ACS-to-TIRS rotation matrix.

Compute the updated OLI-to-TIRS rotation matrix as the transpose of the updated TIRS-to-OLI matrix:

[OLI2TIRS]’ = [TIRS2OLI]’T

Compute the updated ACS-to-TIRS rotation matrix using the ACS-to-OLI matrix from the CPF and the updated OLI-to-TIRS rotation matrix:

[ACS2TIRS]’ = [OLI2TIRS]’ [ACS2OLI]

**Read Correlation Information Sub-Algorithm (read\_correlation\_info)**

This function reads the correlation information from the file generated by the Output Correlation Information sub-algorithm above.

**Filter Outliers Sub-Algorithm (filter\_outliers)**

This function separates the focal plane correlation data into groups for each SCA for the X (sample) and Y (line) directions. It then finds the standard deviation for the points in each group. Outlier rejection is then performed on the points based on the tolerance selected by the user and the Student's T distribution. This procedure is described in the Geometric Accuracy Assessment Algorithm Description Document.

**Calculate Point Weights Sub-Algorithm (calculate\_point\_weights)**

This function calculates the weight associated with each correlation point for doing the Legendre polynomial fit.

Currently, this routine assigns the weight passed in to each point, effectively assigning each point an equal weight. Originally, it was thought that the correlation strength would factor into the weight, but that was determined to not be needed. This routine was left in to allow point-specific weight factors to be added at a later date.

**Fit Parameters Sub-Algorithm (fit\_polynomials)**

This function performs the weighted least-squares fit of the correlation data points (using the angular error) to find the alignment angle corrections and the Legendre error polynomials. The least squares correction parameter vector [ is given by solving:

[N] [ = [L]

Where:

[N] is the [27 x 27] normal equations matrix

[ is the [27 x 1] unknown vector containing the alignment angle and Legendre coefficient corrections we are looking for

[L] is the [27 x 1] constant vector

Solving the above equation for [ yields: [] = ([N])-1 \* [L]

**Calculate Post-fit Residuals Sub-Algorithm (calculate\_post\_fit\_residuals)**

This function calculates the residual statistics, as described in step f) above, after the alignment corrections and Legendre polynomial coefficient corrections have been calculated.

**Calculate Legendre Polynomial Sub-Algorithm (calc\_legendre\_poly)**

This function calculates the Legendre polynomial for the input normalized detector value, x:

along *=* coeff\_along0 + coeff\_along1 x+ coeff\_along2(3\*x2 – 1)/2 + coeff\_along3 x\*(5\* x2 – 3)/2

across *=* coeff\_across0 + coeff\_across1 x+ coeff\_across2 (3\*x2 – 1)/2 + coeff\_across3 x\*(5\* x2 – 3)/2

**Create TIRS Alignment Report Sub-Algorithm (create\_tirs\_alignment\_report)**

This function generates a file reporting the results of the TIRS-to-OLI alignment angle and TIRS SCA Legendre polynomial fit calculations. The report file contents are shown in Table 6‑35 below.

**Write Coefficients Sub-Algorithm (write\_coeffs)**

This function writes an entire set of coefficients to the indicated output file.

**Write TIRS Alignment Calibration Results to Characterization Database (trend\_to\_database)**

This function writes the results of the TIRS-to-OLI alignment angle and TIRS SCA Legendre polynomial fit calculations to the geometric characterization database. The output is only written to the database if the post-fit along- and across-track RMSE statistics are all below the threshold values specified in the CPF (the trending metrics). The characterization database output is listed in Table 6‑35 below.

**Write SCA Parameters CPF Sub-Algorithm (write\_SCA\_parameters\_cpf)**

This function writes the updated Legendre coefficients to a new LOS\_LEGENDRE CPF parameter group, in the ODL format used by the CPF, to a separate ASCII output file. All the CPF LOS\_LEGENDRE parameter values except for the new Legendre coefficients are extracted from the original CPF or the LOS Model structure. Current plans call for actual calibration updates to be based on multiple scene results extracted from the characterization database, so this capability is primarily a convenience for testing purposes.

**Write TIRS-to-OLI Alignment Parameters CPF Sub-Algorithm (update\_alignment\_parameters\_cpf)**

This function writes the ATTITUDE\_PARAMETERS parameter group of the CPF, in the ODL format used by the CPF, to a separate ASCII output file. The updated ACS-to-TIRS rotation matrix computed in h.4) above is written to the parameter group along with the current values for all other parameters in that group. Current plans call for actual calibration updates to be based on multiple scene results extracted from the characterization database, so this capability is primarily a convenience for testing purposes.

**Algorithm Output Details**

The contents of the output TIRS alignment calibration report file and the corresponding geometric characterization database outputs are summarized in Table 6‑35 below. All fields are written to the output report file but only those with "Yes" in the "Database Output" column are written to the characterization database. Note that the first eleven fields listed constitute the standard report header.

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Database Output** |
| 1. Date and time | 1. Date (day of week, month, day of month, year) and time of file creation. | 1. Yes |
| 1. Spacecraft and instrument source | 1. Landsat 8/9 and TIRS | 1. Yes |
| 1. Processing Center | 1. EROS Data Center SVT | 1. Yes |
| 1. Work order ID | 1. Work order ID associated with processing (blank if not applicable) | 1. Yes |
| 1. WRS path | 1. WRS path number | 1. Yes |
| 1. WRS row | 1. WRS row number | 1. Yes |
| 1. Software version | 1. Software version used to create report | 1. Yes |
| 1. Off-nadir angle | 1. Scene off-nadir roll angle (in degrees) (only nadir-viewing scenes are used for TIRS alignment) | 1. Yes |
| 1. Acquisition type | 1. Earth, Lunar, or Stellar (only Earth-viewing scenes are used for TIRS alignment calibration) | 1. Yes |
| 1. Geo Char ID | 1. Geometric Characterization ID | 1. Yes |
| 1. L1T image file | 1. Name of TIRS L1T used to measure tie points | 1. No |
| Acquisition date | Date of L1T image acquisition (new) | Yes |
| Reference image file | Name of reference (OLI) image used to measure tie points | Yes |
| Original TIRS-to-OLI angles | Original TIRS-to-OLI roll-pitch-yaw alignment angles in radians (new) | Yes |
| TIRS-to-OLI correction angles | Estimated roll-pitch-yaw corrections to the TIRS-to-OLI alignment knowledge in radians (new) | Yes |
| Update TIRS-to-OLI angles | Updated TIRS-to-OLI roll-pitch-yaw alignment angles in radians (new) | Yes |
| Original TIRS alignment matrix | Original 3x3 TIRS-to-OLI alignment matrix (new) | No |
| Updated TIRS alignment matrix | Updated 3x3 TIRS-to-OLI alignment matrix (new) | No |
| Confidence Level | Confidence level used for outlier rejection | Yes |
| Fit Order | Order of Legendre fit | Yes |
| Number of SCAs | Number of SCAs calibrated (3) | Yes |
| For each SCA: |  |  |
| SCA Number | Number of the current SCA (1-3) | Yes |
| Original AT Legendre coeffs | Original along-track Legendre coefficients: a0, a1, a2, a3 | Yes |
| Original XT Legendre coeffs | Original across-track Legendre coefficients: b0, b1, b2, b3 | Yes |
| Error AT Legendre coeffs. | The computed updates to the along-track Legendre coefficients: c0, c1, c2, c3 | Yes |
| Error XT Legendre coeffs. | The computed updates to the across-track Legendre coefficients: d0, d1, d2, d3 | Yes |
| New AT Legendre coeffs | New along-track Legendre coefficients: a'0, a'1, a'2, a’3 | Yes |
| New XT Legendre coeffs | New across-track Legendre coefficients: b'0, b'1, b'2, b’3 | Yes |
| Pre-fit AT statistics | Pre-fit along-track offset mean, standard deviation, and RMSE statistics | Yes |
| Pre-fit XT statistics | Pre-fit across-track offset mean, standard deviation, and RMSE statistics | Yes |
| Post-fit AT residual statistics | Post-fit along-track residual mean, standard deviation, and RMSE statistics | Yes |
| Post-fit XT residual statistics | Post-fit across-track residual mean, standard deviation, and RMSE statistics | Yes |
| Number of Points | Number of tie points used for current SCA | Yes |
| CPF Group: | Written to a separate output ODL file |  |
| Effective Date Begin | Beginning effective date of CPF group (from the original CPF): YYYY-MM-DD | No |
| Effective Data End | Ending effective date of CPF group (from the original CPF): YYYY-MM-DD | No |
| ACS-to-TIRS rotation matrix | Updated 3x3 attitude control system-to-TIRS rotation matrix | No |
| Number of SCAs | Number of SCAs (3): Num\_SCA = 3 | No |
| For each SCA: |  |  |
| New Legendre polynomial coefficients | Four (one per band/row) arrays of four along-track Legendre coefficients followed by four arrays of four across-track Legendre coefficients. | No |
| Tie Point Data: | For each tie point: |  |
| SCA Number | SCA where the tie point was measured | No |
| Grid Cell Column Number | Column number of the grid cell containing the tie point | No |
| Nominal Output Space Line | Predicted tie point output space line location | No |
| Nominal Output Space Sample | Predicted tie point output space sample location | No |
| LOS Line Error | Measured LOS error delta line (in pixels) | No |
| LOS Sample Error | Measured LOS error delta sample (in pixels) | No |
| LOS AT Error | Measured LOS error along-track delta angle  (in microradians) | No |
| LOS XT Error | Measured LOS error across-track delta angle  (in microradians) | No |
| State Flag | Tie point state (outlier) flag | No |
| AT Fit Residual | Along-track fit residual (in microradians) | No |
| XT Fit Residual | Across-track fit residual (in microradians) | No |

Table 6‑35. TIRS Alignment Calibration Output Details

**Accessing the TIRS Alignment Results in the Characterization Database**

Though not part of the formal TIRS alignment calibration algorithm, some comments regarding the anticipated methods of accessing and analyzing the individual scene TIRS alignment calibration results stored in the characterization database may assist with the design of the characterization database.

The database output from the TIRS alignment calibration algorithm will be accessed by a data extraction tool that queries the characterization database to retrieve TIRS alignment calibration results from multiple scenes. The only processing required on the returned results is to compute the average "new" TIRS-to-OLI alignment angles and the average "new" Legendre coefficients for each SCA across all returned scenes. The returned scene results and computed mean alignment angles and mean Legendre coefficient values will be output in a report containing a comma-delimited table of the retrieved trending results as well as the summary averages.

The geometric results would typically be queried by acquisition date and/or WRS path/row. The most common query would be based on acquisition date range, for example, selecting all of the results for a given calendar quarter:

Acquisition\_Date is between 01APR2012 and 30JUN2012

The average alignment angles would be calculated from the updated alignment angles for the individual scenes returned, as:



for angle j = roll, pitch, yaw.

The average coefficients would be calculated from the "new" Legendre coefficients for the individual scenes returned, as:



for coefficient j (j=0,1,2,3) for each SCA (SCA=1,2,3).

The query results would be formatted in a set of comma-delimited records (for ease of ingest into Microsoft Excel), one record per scene. Each record would contain all of the "header" fields written to the characterization database (items with "Yes" in the rightmost column of Table 6‑35 above) but only the "new" alignment angles and the "new" Legendre coefficients for each SCA. The other fields would be retrieved using general purpose database access tools, if and when desired. A header row containing the field names should precede the database records.

Following the scene records the average alignment angles and Legendre coefficients should be written out in the same CPF/ODL syntax used in the report file. This output uses the same structure shown in the final row in Table 6‑35 above, but contains the average, rather than a single scene's, angles and Legendre coefficients.

#### Maturity

Most of the OLI focal plane alignment calibration logic was reusable, but the TIRS version was adapted to include the TIRS-to-OLI alignment computation. The focal plane alignment calibration logic were also adapted to the TIRS sensor parameters:

1. There are 3 separate SCAs to calibrate (vs. 14 OLI SCAs).
2. Analysis of the TIRS optical model indicated that the heritage Legendre polynomial order of 2 is not sufficiently accurate for the TIRS due to the significantly longer SCAs. Prototype tests indicate that a Legendre polynomial order of 3 is sufficient. This was a relatively minor change to the algorithm.
3. Since the alignment angle and Legendre coefficient updates are being computed together, the solution must be simultaneous. This is a departure from the heritage method in which each SCA was calibrated separately.

#### Notes

Some additional background assumptions and notes include the following:

1. The 10.8 micrometer thermal band will be used to provide the geometric reference for the TIRS instrument. If subsequent band correlation studies show that the 12.0 micrometer band provides superior correlation performance relative to the OLI SWIR bands, it could be used as the reference instead and this decision will be revisited. Such a change would affect the TIRS band alignment calibration algorithm as well.
2. The input TIRS and OLI L1T images are treated as separate inputs in the baseline TIRS algorithm. These could ultimately be contained in the same output image, but since the TIRS image is SCA-separated and the OLI image is SCA-combined it may be best to keep them as two distinct input images even if merged OLI and TIRS images can be produced.
3. The TIRS focal plane model uses third order Legendre coefficients to model the line-of-sight directions for each SCA, as noted in maturity note #2 above.
4. The baseline assumption is that Legendre coefficient sets will be stored in the CPF for both active detector rows for each band (10.8 and 12.0 micrometer) on each SCA, but that only the primary detector set will be calibrated. Only the coefficients for the primary row for the 10.8 micrometer band will be updated by this calibration procedure. This assumption may be modified if it is decided that the TIRS CPF will contain only Legendre coefficients for the primary detector rows, in which case there will only be two sets of Legendre coefficients in the CPF, or if the Legendre coefficients for the redundant rows are to be maintained by the calibration procedures, in which case the computed updates will be added to the Legendre coefficients from both rows for the 10.8 micrometer band.
5. The ATTITUDE\_PARAMETERS CPF parameter group contains the ACS-to-TIRS and ACS-to-OLI alignment matrices. These two matrices are related by the TIRS-to-OLI alignment matrix, which is maintained by the TIRS alignment calibration algorithm, as follows: [ACS2OLI] = [TIRS2OLI] [ACS2TIRS]. To avoid the redundancy inherent in retaining all three of these matrices in the CPF, the [TIRS2OLI] matrix is constructed, when needed, from the ACS-to-sensor matrices as:

[TIRS2OLI] = [ACS2OLI] [ACS2TIRS]-1

This [TIRS2OLI] matrix is updated by the TIRS alignment calibration procedure and the result is then used to update the [ACS2TIRS] matrix in the CPF as:

[ACS2TIRS] = [TIRS2OLI]-1 [ACS2OLI].

1. The LOS\_LEGENDRE CPF group contains the TIRS focal plane model in the form of the along-track and across-track Legendre coefficients updated by this algorithm.