Department of the Interior
U.S. Geological Survey

LANDSAT 7 (L7)
IMAGE ASSESSMENT SYSTEM (IAS)
GEOMETRIC ALGORITHM THEORETICAL BASIS
DOCUMENT (ATBD)

Version 1.0

December 2006

USGS
science for a changing world

LANDSAT 7 (L7)
IMAGE ASSESSMENT SYSTEM (IAS)
GEOMETRIC ALGORITHM THEORETICAL BASIS DOCUMENT
(ATBD)

December 2006

Prepared By: Reviewed By:

_____   _____
J. Storey   Date   D. Strande   Date
Chief Systems Engineer   IAS Systems Engineer
SAIC   SAIC

Reviewed By: Reviewed By:

_____   _____
R. Hayes   Date   A. Meyerink   Date
Landsat CalVal Task Lead   Landsat A&P Task Lead
SAIC   USGS

Reviewed By: Approved By:

_____   _____
S. Labahn   Date   J. Lacasse   Date
Landsat Ground Segment Manager   Landsat Mission Managmement Officer
SAIC   USGS

EROS
Sioux Falls, South Dakota

LS-IAS-01
                                                Version 1.0

Contents

List of Figures

List of Tables

Section 1   Introduction

1.1   Purpose

This document describes the geometric algorithms used by the Landsat 7 Image Assessment System (IAS). These algorithms are implemented as part of the IAS Level 1 processing, geometric characterization, and geometric calibration software components.

The overall purpose of the IAS geometric algorithms is to use Earth ellipsoid and terrain surface information, in conjunction with spacecraft ephemeris and attitude data, and knowledge of the Enhanced Thematic Mapper Plus (ETM+) instrument and Landsat 7 satellite geometry to relate locations in ETM+ image space (band, scan, detector, sample) to geodetic object space (latitude, longitude, and height). These algorithms are used to create accurate Level 1 output products, characterizing the ETM+ absolute and relative geometric accuracy, and to derive improved estimates of geometric calibration parameters such as the sensor-to-spacecraft alignment.

This document presents background material that describes the relevant coordinate systems, time systems, ETM+ sensor geometry, and Landsat 7 spacecraft geometry, as well the IAS processing algorithms.

Level 1 processing algorithms include:

- Payload Correction Data (PCD) processing
- Mirror Scan Correction Data (MSCD) processing
- ETM+/Landsat 7 sensor/platform geometric model creation
- Sensor line-of-sight generation and projection
- Output space/input space correction grid generation
- Image resampling
- Geometric model precision correction using ground control
- Terrain correction

These algorithms are discussed in detail in 3.1.5.1.3.

The geometric calibration algorithms, discussed in 3.1.5.4, include:

- ETM+ sensor alignment calibration
- Focal plane calibration (focal plane band-to-band alignment)
- Scan mirror calibration

The geometric characterization algorithms, discussed in 3.1.5.5, include:

- Band-to-band registration
- Image-to-image registration
- Geodetic accuracy assessment (absolute external accuracy)
- Geometric accuracy assessment (relative internal accuracy)

LS-IAS-01
Version 1.0

Section 2   Overview and Background Information

The Landsat 7 Enhanced Thematic Mapper Plus (ETM+) continues the Landsat program of multispectral space-borne Earth remote sensing satellites that began with the launch of Landsat 1 in 1972. Landsat 7 provides data consistent with the Landsat historical record in the time period between the decommissioning of the still-functioning Landsat 5 and the introduction of successor instruments based on newer technology in the next century.

The basic sensor technology used in the ETM+ is similar to the Thematic Mapper (TM) instruments flown in Landsats 4 and 5 and the Enhanced Thematic Mapper (ETM) built for Landsat 6, which suffered a launch failure. The geometric processing, characterization, and calibration algorithms described in this document take into account the new 15-meter panchromatic band (also present in Landsat 6) and the higher resolution thermal band in adapting the processing techniques applied to Landsat 4/5 TM data to Landsat 7 ETM+ data.

The inclusion of an Image Assessment System (IAS) as an integral part of the Landsat 7 ground system illustrates the more systematic approach to instrument calibration, in-orbit monitoring, and characterization necessitated by the more stringent calibration requirements of the Landsat 7 ETM+ (as compared to earlier Landsat missions). This is especially true of the radiometric calibration requirements, which include using partial- and full-aperture solar calibrations to monitor the stability and performance of ETM+ detectors and on-board calibration lamps. The IAS also provides a platform for systematic geometric performance monitoring, characterization, and calibration as described in the remainder of this document.

2.1   Experimental Objective

The objective of the Landsat 7 ETM+ mission is to provide high-resolution (15-meter panchromatic, 30-meter multispectral, 60-meter thermal) imagery of Earth's land areas from near polar, sun-synchronous orbit. These data extend the continuous data record collected by Landsats 1–5, provide higher spatial resolution in the new panchromatic band, provide greater calibration accuracy to support new and improved analysis applications, and provide a high-resolution reference for the Earth Observing System (EOS)-AM1 Moderate Resolution Imaging Spectroradiometer (MODIS), Multiangle Imaging Spectroradiometer (MISR), and Advanced Spaceborne Thermal Emission and Reflection (ASTER) instruments.

The geometric algorithms described in this document are used by the Landsat 7 IAS and by the Level 1 Product Generation System (LPGS) to ensure that the complex ETM+ internal geometry is sufficiently modeled and characterized to permit generation of Level 1 products that meet the Landsat 7 system geometric accuracy specifications.

2.2 Historical Perspective

The Landsat 7 ETM+ mission continues the evolutionary improvement of the Landsat family of satellites, which began with Landsats 1, 2, and 3 carrying the Multi-Spectral Scanner (MSS), and builds on the Landsat 4/5 TM heritage. The ETM+ adds a 15-meter Ground Sample Distance (GSD) panchromatic band, similar to that used in the ETM instrument built for Landsat 6, improves the thermal band spatial resolution from 120 meters (TM/ETM) to 60 meters (ETM+), and improves the radiometric calibration accuracy from 10 percent (TM/ETM) to 5 percent (ETM+). The Landsat 7 system also takes a more systematic approach to monitoring and measuring system geometric performance in flight than was applied to Landsats 4 and 5, which were turned over to commercial operation in October 1985.

2.3 Instrument Characteristics

The fundamental ETM+ instrument geometric characteristics include:

- Nadir viewing +/–7.5 degree instrument field of view (FOV)
- Bi-directional, cross-track scanning mirror
- Along-track scan line corrector mirrors
- Eight bands, three resolutions, two focal planes
- Even/odd detectors staggered on focal planes
- Pre-launch, ground-calibrated, non-linear mirror profile
- Scan angle monitor that measures actual mid-scan and scan end times (as deviations from nominal)
- High frequency (2–125 Hertz [Hz]) angular displacement sensors that measure instrument jitter

2.4 Ancillary Input Data

The Landsat 7 ETM+ geometric characterization, calibration, and correction algorithms are applied to the wideband data (image plus supporting Payload Correction Data [PCD] and Mirror Scan Correction Data [MSCD]) contained in Level 0R (raw) or 1R (radiometrically corrected) products. Some of these algorithms also require additional ancillary input data sets, including:

- Precise ephemeris from the Flight Dynamics Facility—used to minimize ephemeris error when performing sensor-to-spacecraft alignment calibration
- Ground control/reference images for geometric test sites—used in precision correction, geodetic accuracy assessment, and geometric calibration algorithms
- Digital elevation data for geometric test sites—used in terrain correction and geometric calibration
- Pre-launch ground calibration results, including band/detector placement and timing, scan mirror profiles, and attitude sensor data transfer functions (gyro and

Attitude Displacement Sensors [ADS]), to be used in the generation of the initial Calibration Parameter File
- Earth parameters, including static Earth model parameters (e.g., ellipsoid axes, gravity constants) and dynamic Earth model parameters (e.g., polar wander offsets, Universal Time Code (UTC)-Corrected (UT1) time corrections)—used in systematic model creation and incorporated into the Calibration Parameter File

Section 3   Algorithm Descriptions

This section presents the underlying theory and mathematical development of the Image Assessment System (IAS) geometric algorithms.

3.1   Theoretical Description

The supporting theoretical concepts and mathematics of the IAS geometric algorithms are presented in the following subsections. Section 3.1.1 presents a review of the Landsat 7 Enhanced Thematic Mapper Plus (ETM+) viewing geometry to put the subsequent discussion in context. Sections 3.1.2 and 3.1.3 address the coordinate systems used by the algorithms and the relationships between them, citing references where appropriate. Section 3.1.4 briefly defines and discusses the various time systems used by the IAS algorithms. Section 3.1.5 presents the mathematical development of, and solution procedures for, the Level 1 processing, geometric calibration, and geometric characterization algorithms. Section 3.1.6 discusses estimates of uncertainty (error analysis) associated with each of the algorithms.

3.1.1 Landsat 7 ETM+ Viewing Geometry Overview

The ETM+ instrument detectors are aligned in parallel rows on two separate focal planes: the primary focal plane, containing bands 1–4 and 8 (panchromatic), and the cold focal plane, containing bands 5, 6, and 7. The primary focal plane is illuminated by the ETM+ scanning mirror, primary mirror, secondary mirror, and scan line corrector mirror. In addition to these optical elements, the cold focal plane optical train includes the relay folding mirror and the spherical relay mirror. This is depicted in Figure 3-1. The ETM+ scan mirror provides a nearly linear cross-track scan motion that covers a 185-kilometer (km) wide swath on the ground. The scan line corrector compensates for the forward motion of the spacecraft and allows the scan mirror to produce usable data in both scan directions.

Within each focal plane, the rows of detectors from each band are separated in the along-scan (cross-track) direction. The odd and even detectors from each band are also separated slightly. The detector layout geometry is shown in Figure 3-2. Samples from the ETM+ detectors are packaged into minor frames as they are collected (in time order) for downlink in the wideband data stream.

Within each minor frame, the output from the odd detectors from bands 1–5 and 7 and all band 8 detectors are held at the beginning of the minor frame. The even detectors from bands 1–5 and 7 and all band 8 detectors are held at the midpoint of the minor frame. The even and odd detectors from band 6 are held at the beginning of alternate minor frames.

The bands are nominally aligned during Level 0R ground processing by delaying the even and odd detector samples from each band to account for the along-scan motion needed to view the same target point. These delays are implemented as two sets of fixed integer offsets for the even and odd detectors of each band—one set for forward scans and one

set for reverse scans. Two sets of offset delays are necessary since the band/detector sampling sequence with respect to a ground target is inverted for reverse scans. Level 0R ground processing also reverses the sample time sequence for reverse scans to provide nominal spatial alignment between forward and reverse scans.
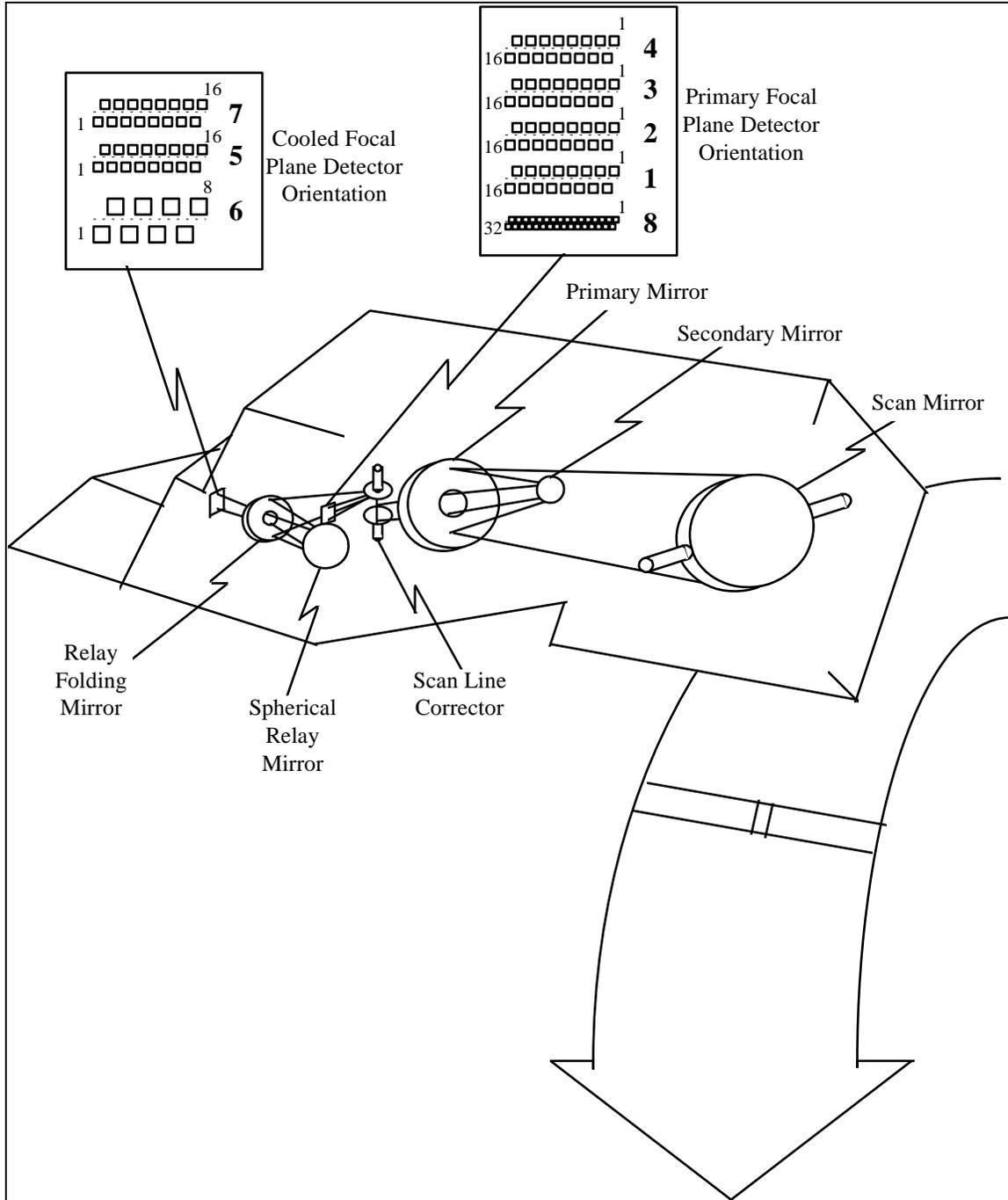
Figure 3-1. ETM+ Geometry Overview

The instrument's 15-degree field of view is swept over the ETM+ focal planes by the scan mirror. Each scan cycle, consisting of a forward and reverse scan, is 142.245 milliseconds. The scan mirror period for a single forward or reverse scan is nominally 71.462 milliseconds. Of this time, approximately 60.743 milliseconds is devoted to the Earth viewing portion of the scan, with 10.719 milliseconds devoted to the collection of calibration data and mirror turnaround. Within each scan, detector samples (minor frames) are collected every 9.611 microseconds (with two samples per minor frame for the 15-meter resolution panchromatic band). More detailed information on the ETM+ instrument's construction, operation, and pre-flight testing is provided in Reference 5.

Primary Focal Plane						Cold Focal Plane		
Band 8	Band 1	Band 2	Band 3	Band 4	Optical Axis	Band 7	Band 5	Band 6

25 IFOV   25 IFOV   25 IFOV   25 IFOV   10.4   45 IFOV   26 IFOV   34.75 IFOV
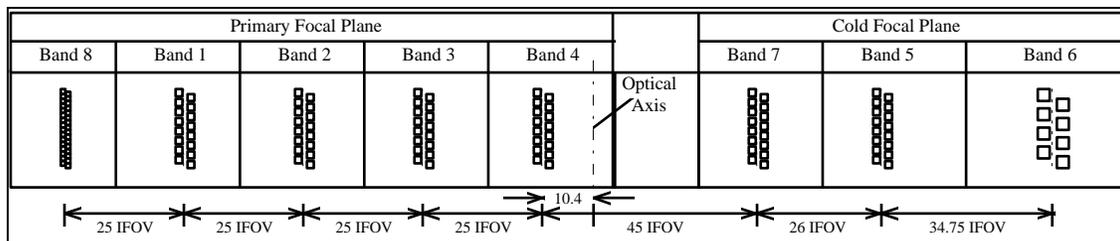
Figure 3-2. ETM+ Detector Array Ground Projection

The 15-degree instrument field of view sweeps out a ground swath approximately 185 kilometers (km) wide. This swath is sampled 6,320 (nominally) times by the ETM+ 30-meter resolution bands. Since 16 detectors from each 30-meter band are sampled in each data frame, the nominal scan width is 480 meters. The scan line corrector compensates for spacecraft motion during the scan to prevent excessive (more than 1 pixel) overlap or underlap at the scan edges. Despite this, the varying effects of Earth rotation and spacecraft altitude (both of which are functions of position in orbit) will lead to variations in the scan-to-scan gap observed.

3.1.2   Coordinate Systems

The coordinate systems used by Landsat 7 are described in detail in Reference 2. There are ten coordinate systems of interest for the Landsat 7 IAS geometric algorithms. These coordinate systems are referred to frequently in the remainder of this document and are briefly defined here to provide context for the subsequent discussion. They are presented in the order in which they would be used to transform a detector and sample time into a ground position.

1. ETM+ Array Reference (Focal Plane) Coordinate System

   The focal plane coordinate system is used to define the band and detector offsets from the instrument optical axis, which are ultimately used to generate the image space viewing vectors for individual detector samples. It is defined so that the Z-axis is along the optical axis and is positive toward the scan mirror. The origin is where the optical axis intersects the focal planes (see Figure 3-3). The X-axis is parallel to the scan mirror's axis of rotation and is in the along-track direction after a reflection off the scan mirror, with the positive direction toward detector 1. The Y-axis is in the focal plane's along-scan direction with the positive direction toward

band 8. This definition deviates from the description in the Landsat 7 Program Coordinates System Standard Document in that the origin is placed at the focal plane/optical axis intersection rather than at an arbitrary reference detector location.
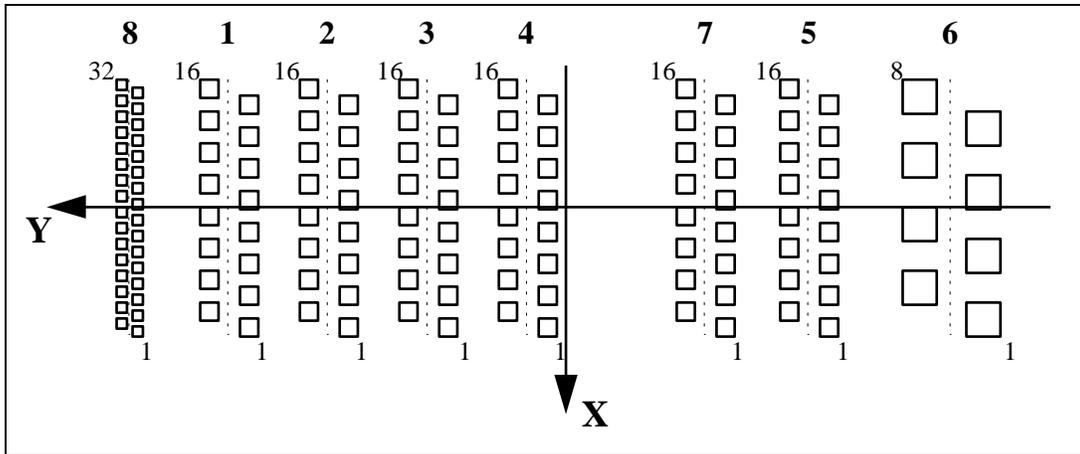
Figure 3-3. ETM+ Focal Plane Coordinate System

2. ETM+ Sensor Coordinate System

The scan mirror and scan line corrector models are defined in the sensor coordinate system. This is the coordinate system in which an image space vector, representing the line of sight from the center of a detector, and scan mirror and scan line corrector angles are converted to an object space-viewing vector.

The Z-axis corresponds to the Z-axis of the focal plane coordinate system (optical axis) after reflection off the scan mirror and is positive outward (toward the Earth). The X-axis is parallel to the scan mirror axis of rotation and is positive in the along-track direction. The Y-axis completes a right-handed coordinate system.

Scan mirror angles are rotations about the sensor X axis and are measured in the sensor Y-Z plane with the Z axis corresponding to a zero scan angle and positive angles toward the Y axis (west in descending mode). Scan line corrector angles are rotations about the sensor Y-axis and are measured in the sensor X-Z plane with the Z-axis corresponding to a zero scan line corrector angle and positive angles toward the X-axis (the direction of flight).

The sensor coordinate system is depicted in Figure 3-4. In this coordinate system, the scan line corrector performs a negative rotation (from a positive angle to a negative angle) on every scan, while the scan mirror performs a positive-to-negative rotation for forward scans and a negative-to-positive rotation for reverse scans.
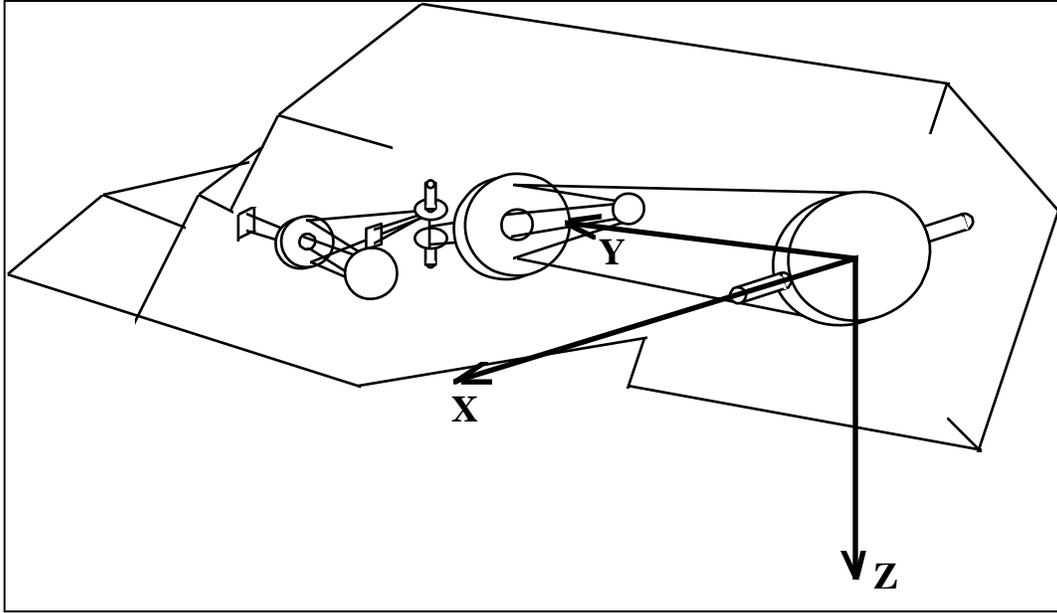
Figure 3-4. ETM+ Sensor Coordinate System

3. Navigation Reference Coordinate System

The navigation reference frame is the body-fixed coordinate system used for spacecraft attitude determination and control. The coordinate axes are defined by the spacecraft attitude control system (ACS), which attempts to keep the navigation reference frame aligned with the orbital coordinate system so that the ETM+ optical axis is always pointing toward the center of the Earth. It is the orientation of this coordinate system relative to the inertial coordinate system that is captured in spacecraft attitude data.

4. IMU Coordinate System

The spacecraft orientation data provided by the gyros in the inertial measurement unit are referenced to the Inertial Measurement Unit (IMU) coordinate system. This coordinate system is nominally aligned with the navigation reference coordinate system. The actual alignment of the IMU with respect to the navigation reference is measured pre-flight as part of the ACS calibration. This alignment transformation is used by the IAS to convert the gyro data contained in the Landsat 7 Payload Correction Data (PCD) to the navigation reference coordinate system for blending with the ACS quaternions and the Angular Displacement Assembly (ADA) jitter measurements.

5. ADA Coordinate System

    The high-frequency angular displacements of the sensor line of sight (jitter)
    measured by the ADA are referenced to the ADA coordinate system. This
    reference frame is at a nominal 20-degree angle with respect to the sensor
    coordinate system, corresponding to a 20-degree rotation about the ADA X-axis,
    which is nominally coincident with the sensor X-axis.

    The ADA output recorded in the Landsat 7 PCD is converted from the ADA
    coordinate system to the sensor coordinate system (using the ADA alignment
    matrix), and then to the navigation reference coordinate system (using the sensor
    alignment matrix), so that the high-frequency jitter data can be combined with the
    low-frequency gyro data to determine the actual ETM+ sensor line-of-sight pointing
    as a function of time.

6. Orbital Coordinate System

    The orbital coordinate system is centered on the satellite, and its orientation is
    based on the spacecraft position in inertial space (see Figure 3-5). The origin is
    the spacecraft's center of mass, with the Z-axis pointing from the spacecraft's
    center of mass to the Earth's center of mass. The Y-axis is the normalized cross
    product of the Z-axis and the instantaneous (inertial) velocity vector and
    corresponds to the negative of the instantaneous angular momentum vector
    direction. The X-axis is the cross product of the Y and Z-axes.
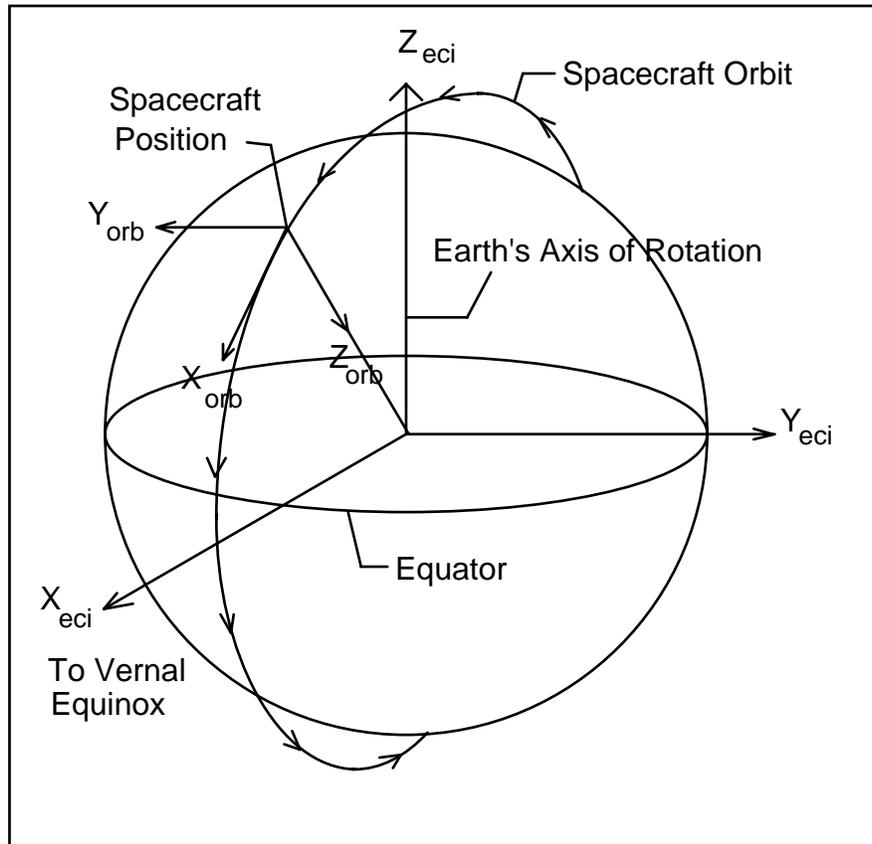
Figure 3-5. Orbital Coordinate System

7. ECI Coordinate System

The Earth-Centered Inertial (ECI) coordinate system is space-fixed with its origin at the Earth's center of mass (see Figure 3-6). The Z-axis corresponds to the mean north celestial pole of epoch J2000.0. The X-axis is based on the mean vernal equinox of epoch J2000.0. The Y-axis is the cross product of the Z and X axes. This coordinate system is described in detail in Reference 2 and Reference 7. Data in the ECI coordinate system are present in the Landsat 7 ETM+ Level 0R product in the form of ephemeris and attitude data contained in the spacecraft PCD.

Although the IAS uses J2000, there is some ambiguity in the existing Landsat 7 system documents that discuss the inertial coordinate system. Reference 2 states that the ephemeris and attitude data provided to and computed by the spacecraft on-board computer are referenced to the ECI True of Date (ECITOD) coordinate system (which is based on the celestial pole and vernal equinox of date rather than at epoch J2000.0). This appears to contradict Reference 1, which states that the ephemeris and attitude data contained in the Landsat 7 PCD (i.e., from the spacecraft on-board computer) are referenced to J2000.0. The relationship between these two inertial coordinate systems consists of the slow variation in

orientation due to nutation and precession, which is described in Reference 2 and Reference 7.
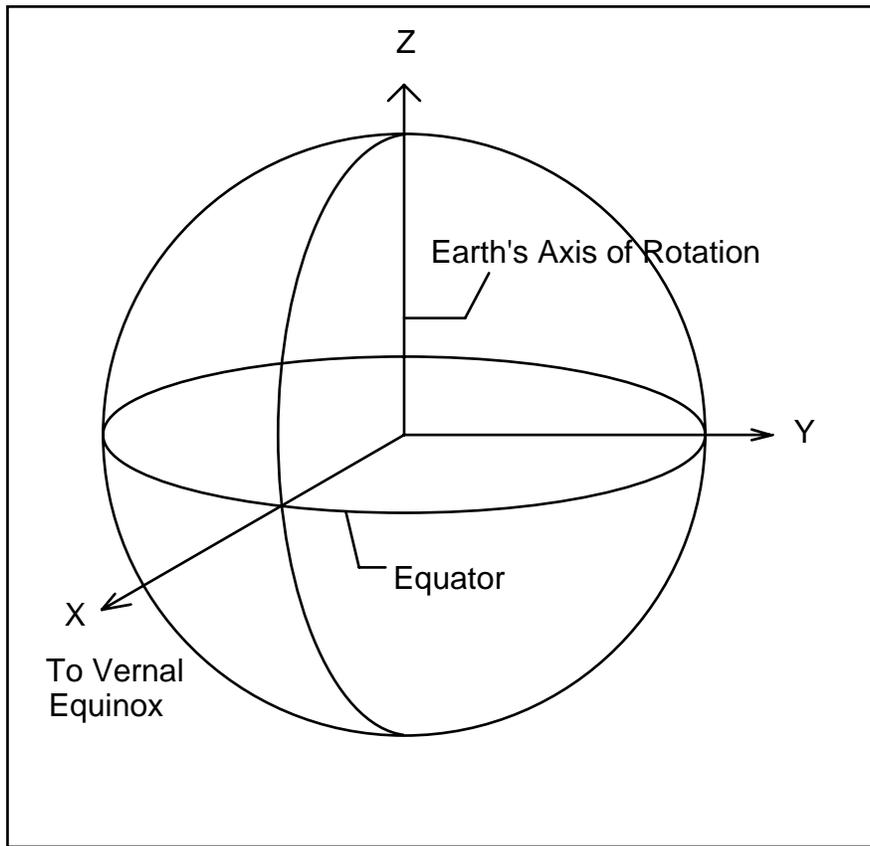
Figure 3-6. Earth-Centered Inertial (ECI) Coordinate System

8. ECR Coordinate System

The Earth-Centered Rotating (ECR) coordinate system is Earth-fixed with its origin at the Earth's center of mass (see Figure 3-7). It corresponds to the Conventional Terrestrial System defined by the Bureau International de l'Heure (BIH), which is the same as the U.S. Department of Defense World Geodetic System 1984 (WGS84) geocentric reference system. This coordinate system is described in Reference 7.
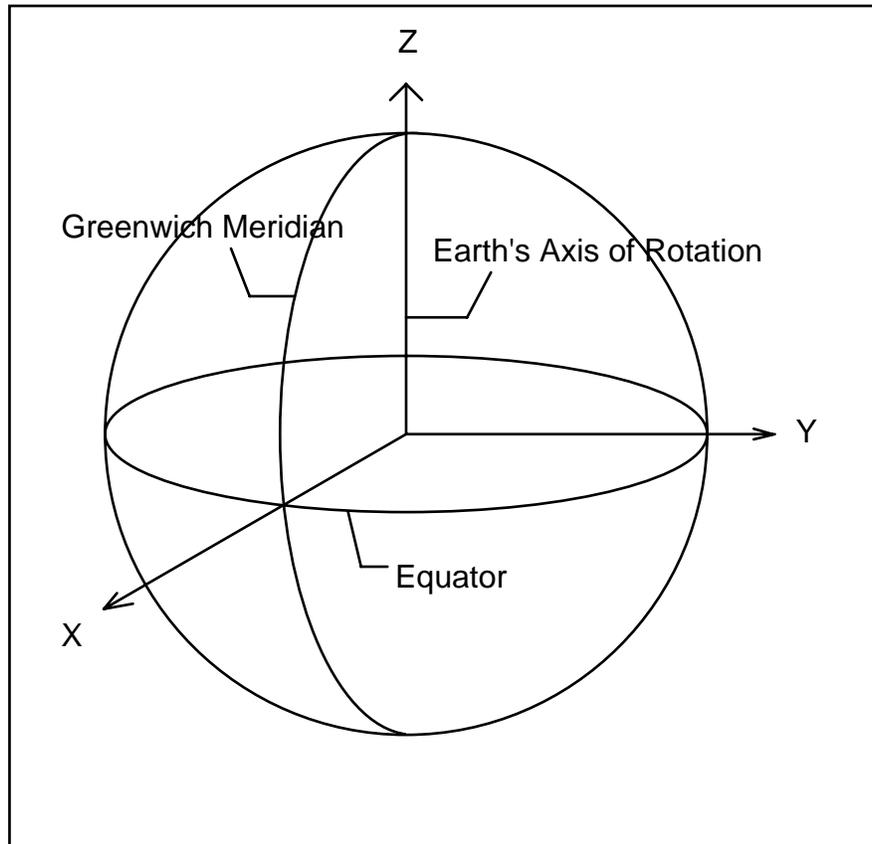
Figure 3-7. Earth-Centered Rotating (ECR) Coordinate System

9. Geodetic Coordinate System

The geodetic coordinate system is based on the WGS84 reference frame with coordinates expressed in latitude, longitude, and height above the reference Earth ellipsoid (see Figure 3-8). No ellipsoid is required by definition of the ECR coordinate system, but the geodetic coordinate system depends on the selection of an Earth ellipsoid. Latitude and longitude are defined as the angle between the ellipsoid normal and its projection onto the equator and the angle between the local meridian and the Greenwich meridian, respectively. The scene center and scene corner coordinates in the Level 0R product metadata are expressed in the geodetic coordinate system.
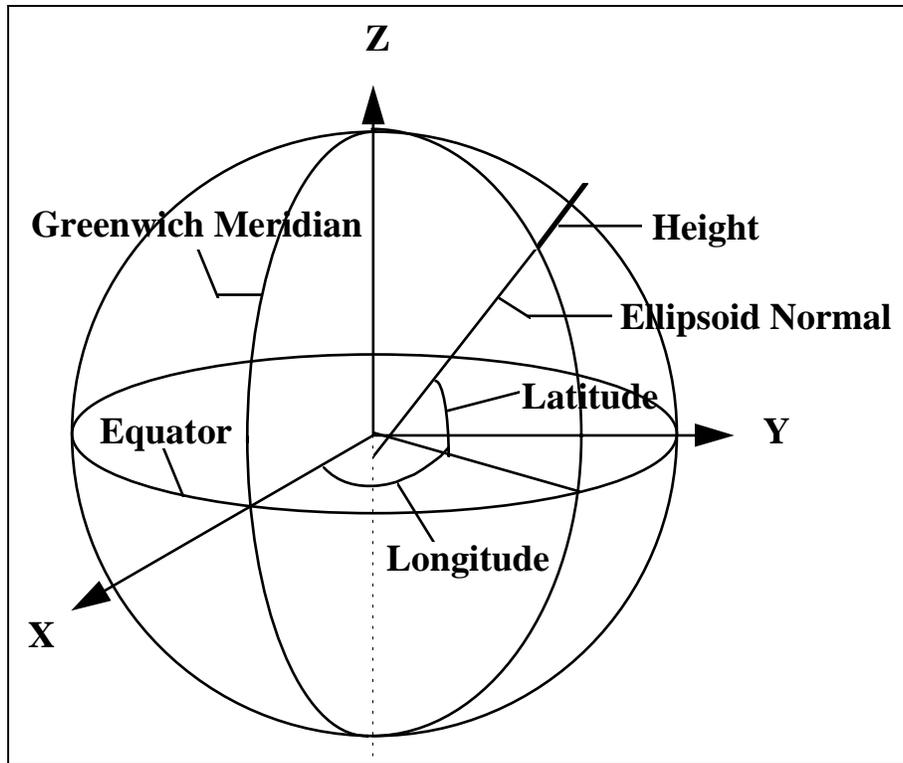
Figure 3-8. Geodetic Coordinate System

10. Map Projection Coordinate System

Level 1 products are generated with respect to a map projection coordinate
system, such as the Universal Transverse Mercator, which provides mapping from
latitude and longitude to a plane coordinate system that is an approximation of a
Cartesian coordinate system for a portion of the Earth's surface. It is used for
convenience as a method of providing digital image data in an Earth-referenced
grid that is compatible with other ground-referenced data sets. Although the map
projection coordinate system is only an approximation of a true local Cartesian
coordinate system at the Earth's surface, the mathematical relationship between
the map projection and geodetic coordinate systems is defined precisely and
unambiguously; the map projections the IAS uses are described in Reference 8.

3.1.3 Coordinate Transformations

There are nine transformations between the ten coordinate systems used by the IAS
geometric algorithms. These transformations are referred to frequently in the remainder
of this document and are defined here. They are presented in the logical order in which a
detector and sample number would be transformed into a ground position.

LS-IAS-01
                                                                                                              Version 1.0

1. Focal Plane-to-Sensor Transformation

    The relationship between the focal plane and sensor coordinate systems is a rapidly varying function of time, incorporating the scan mirror and scan line corrector models. The focal plane location of each detector from each band can be specified by a set of field angles in the focal plane coordinate system, as described in 3.1.5.1.1. They are transformed to sensor coordinates based on the sampling time (sample/minor frame number from scan start), which is used to compute mirror and scan line corrector angles based on the models described in 3.1.5.1.2 and 3.1.5.1.3.

2. Sensor-to-Navigation Reference Transformation

    The relationship between the sensor and navigation reference coordinate systems is described by the ETM+ instrument alignment matrix. The transformation from sensor coordinates to navigation reference coordinates is described in 4.21 of Reference 2. It includes a three-dimensional rotation, implemented as a matrix multiplication, and an offset to account for the distance between the ACS reference and the instrument scan mirror. The transformation matrix is initially defined as fixed (non-time varying), with improved estimates provided post-launch. Subsequent analysis may detect repeatable variations with time, which can be effectively modeled, making this a (slowly) time-varying transformation. The nominal rotation matrix is the identity matrix.

3. IMU-to-Navigation Reference Transformation

    The IMU coordinate system is related to the navigation reference coordinate system by the IMU alignment matrix, which captures the orientation of the IMU axes with respect to the navigation base. This transformation is applied to gyro data prior to their integration with the ADA data and the ACS quaternions. The IMU alignment is measured pre-flight and is nominally the identity matrix. This transformation is defined in 4.22 of Reference 2.

4. ADA -to-Sensor Transformation

    The angular displacement assembly (ADA) is nominally aligned with the sensor coordinate system with a 20-degree rotation about the X-axis. The actual alignment is measured pre-launch and is used to rotate the ADA jitter observations into the sensor coordinate system, from which they can be further rotated into the navigation reference system using the sensor-to-navigation reference alignment matrix.

5. Navigation Reference-to-Orbital Transformation

The relationship between the navigation reference and orbital coordinate systems is defined by the spacecraft attitude. This transformation is a three-dimensional rotation matrix with the components of the rotation matrix being functions of the spacecraft roll, pitch, and yaw attitude angles. The nature of the functions of roll, pitch, and yaw depends on the exact definition of these angles (i.e., how they are generated by the attitude control system). In the initial model, it is assumed that the rotations are performed in the order roll-pitch-yaw as shown in 4.17 of Reference 2. Since the spacecraft attitude is constantly changing, this transformation is time varying. The nominal rotation matrix is the identity matrix, since the ACS strives to maintain the ETM+ instrument pointing toward the Earth's center.

6. Orbital-to-ECI Transformation

The relationship between the orbital and ECI coordinate systems is based on the spacecraft's instantaneous ECI position and velocity vectors. The rotation matrix to convert from orbital to ECI can be constructed by forming the orbital coordinate system axes in ECI coordinates:

$\underline{P}$ = spacecraft position vector in ECI
$\underline{V}$ = spacecraft velocity vector in ECI
$T_{eci/orb}$ = rotation matrix from orbital to ECI

$\underline{b}_3 = -\underline{p} / |\underline{p}|$ (nadir vector direction)
$\underline{b}_2 = (\underline{b}_3 \times \underline{v}) / |\underline{b}_3 \times \underline{v}|$ (negative of angular momentum vector direction)
$\underline{b}_1 = \underline{b}_2 \times \underline{b}_3$ (circular velocity vector direction)
$T_{eci/orb} = [\ \underline{b}_1 \quad \underline{b}_2 \quad \underline{b}_3\ ]$

7. ECI-to-ECR Transformation

The transformation from ECI-to-ECR coordinates is a time-varying rotation due primarily to the Earth's rotation, but it also contains more slowly varying terms for precession, astronomic nutation, and polar wander. The ECI-to-ECR rotation matrix can be expressed as a composite of these transformations:

$T_{ecr/eci} = A\ B\ C\ D$

A = polar motion
B = sidereal time
C = astronomic nutation
D = precession

Each of these transformation terms is defined in Reference 2 and is described in detail in Reference 7.

8. ECR-to-Geodetic Transformation

The relationship between ECR and geodetic coordinates can be expressed simply in its direct form:

$$e^2 = 1 - b^2 / a^2$$
$$N = a / (1 - e^2 \sin^2(lat))^{1/2}$$
$$X = (N + h) \cos(lat) \cos(lon)$$
$$Y = (N + h) \cos(lat) \sin(lon)$$
$$Z = (N (1 - e^2) + h) \sin(lat)$$

where:

X, Y, Z = ECR coordinates
lat, *lon*, *h* = geodetic coordinates
N = ellipsoid radius of curvature in the prime vertical
e^2 = ellipsoid eccentricity squared
a, b = ellipsoid semi-major and semi-minor axes

The closed-form solution for the general inverse problem (the problem of interest here) involves the solution of a quadratic equation and is not typically used in practice. Instead, an iterative solution is used for latitude and height for points that do not lie on the ellipsoid surface. A procedure for performing this iterative calculation is described in 6.4.3.3 of Reference 4.

9. Geodetic-to-Map Projection Transformation

The transformation from geodetic coordinates to the output map projection depends on the type of projection selected. The mathematics for the forward and inverse transformations for the Universal Transverse Mercator (UTM), Lambert Conformal Conic, Transverse Mercator, Oblique Mercator, Polyconic, Polar Stereo Graphic, and the Space Oblique Mercator (SOM) are given in Reference 8. Further details of the SOM mathematical development are presented in Reference 17.

3.1.4 Time Systems

Four time systems are of primary interest for the IAS geometric algorithms: International Atomic Time (Temps Atomique International [TAI]), Universal Time—Coordinated (UTC), Universal Time—Corrected for polar motion (UT1), and Spacecraft Time (the readout of the spacecraft clock, nominally coincident with UTC). Spacecraft Time is the time system applied to the spacecraft time codes found in the Level 0R PCD and MSCD. UTC, which the spacecraft clock aspires to, is the standard reference for civil timekeeping. UTC is

adjusted periodically by whole leap seconds to keep it within 0.9 seconds of UT1. UT1 is based on the actual rotation of the Earth and is needed to provide the transformation from stellar-referenced inertial coordinates (ECI) to terrestrial-referenced Earth-fixed coordinates (ECR). TAI provides a uniform, continuous time stream that is not interrupted by leap seconds or other periodic adjustments. It provides a consistent reference for resolving ambiguities arising from the insertion of leap seconds into UTC, which can lead to consecutive seconds with the same UTC time. These and a variety of other time systems, and their relationships, are described in Reference 4. The significance of each of these four time systems with respect to the IAS geometric algorithms is described below.

1. Spacecraft Time

The Landsat 7 Mission Operations Center attempts to maintain the on-board spacecraft clock time to within 145 milliseconds of UTC (see Reference 5, 3.7.1.3.16) by making clock updates nominally once per day during periods of ETM+ inactivity. Additionally, the spacecraft PCD includes a quadratic clock correction model, which can be used to correct the spacecraft clock readout to within 15 milliseconds of UTC, according to Reference 1. The clock correction algorithm is:

$$dt = t_{s/c} - t_{update}$$
$$t_{UTC} = t_{s/c} + C_0 + C_1\, dt + 0.5\, C_2\, dt^2$$

where:

dt = spacecraft clock time since last update
t_{UTC} = corrected spacecraft time (UTC +/− 15 milliseconds)
$t_{s/c}$ = spacecraft clock time
t_{update} = spacecraft clock time of last ground-commanded clock update
C_0 = clock correction bias term
C_1 = clock drift rate
C_2 = clock drift acceleration

The Landsat 7 onboard computer uses corrected spacecraft clock data to interpolate the ephemeris data (which are inserted into the PCD) from the uplinked ephemeris (which is referenced to UTC). The residual clock error of 15 milliseconds (3 sigma) leads to an additional ephemeris position uncertainty of approximately 114 meters (3 sigma), which is included in the overall geodetic accuracy error budget. Since the time code readouts in both the PCD and MSCD are uncompensated, the clock correction algorithm must be applied on the ground to get the Level 0R time reference to within 15 milliseconds of UTC.

If the on-board computer were not using the clock drift correction model for ephemeris interpolation, the ground-processing problem would be more complicated. Since the spacecraft time is used as a UTC index to interpolate the

ephemeris points, the times associated with these ephemeris points are true UTC—they simply do not correspond to the actual (corrected) UTC spacecraft time. In this case, the PCD time code errors of up to 145 milliseconds would cause a temporal misregistration of the interpolated ephemeris and the associated orbital orientation reference coordinate system, relative to the spacecraft clock. In making the spacecraft clock corrections, the times associated with the ephemeris points would not be changed in recognition of the fact that they are interpolation indices against a UTC time reference.

2. UTC

As mentioned above, UTC is maintained within 0.9 seconds of UT1 by the occasional insertion of leap seconds. It is assumed that the insertion of leap seconds into the spacecraft clock will be performed on the appropriate dates as part of the daily clock update and that this clock update will be coordinated with the ephemeris updates provided by the Flight Dynamics Facility (FDF), so that the spacecraft clock UTC epoch is the same as the FDF provided ephemeris UTC epoch. Even though UTC provides a uniform time reference for any given scene or sub-interval, it is beneficial for the IAS to have access to a table of leap seconds relating UTC to TAI to support time-related problem tracking and resolution. This information can be obtained from the National Earth Orientation Service (NEOS) Web site at http://maia.usno.navy.mil.

3. UT1

UT1 represents time with respect to the actual rotation of the Earth and is used by the IAS algorithms, which transform inertial ECI coordinates or lines of sight to Earth-fixed ECR coordinates. Failure to account for the difference between UT1 and UTC in these algorithms can lead to ground position errors as large as 400 meters at the equator (assuming the maximum 0.9-second UT1-UTC difference). The UT1-UTC correction typically varies at the rate of approximately 2 milliseconds per day, corresponding to an Earth rotation error of about 1 meter. Thus, UT1-UTC corrections should be interpolated or predicted to the actual image acquisition time to avoid introducing errors of this magnitude. This information can be obtained from the National Earth Orientation Service (NEOS) Web site at http://maia.usno.navy.mil.

4. TAI

Although none of the IAS algorithms are designed to use TAI in normal operations, it is included here for completeness. As mentioned above, there is the possibility that system anomalies will be introduced as a result of timing errors, particularly in the vicinity of UTC leap seconds. The ability to use TAI as a standard reference that can be related to UTC and spacecraft time using a leap second file will assist IAS operations staff in anomaly resolution. For this reason, even though there are no IAS software requirements for a leap second reference, it is strongly recommended that access be provided to the ECS SDP Toolkit leap second file.

3.1.5 Mathematical Description of Algorithms

3.1.5.1   ETM+ Instrument Model

3.1.5.1.1   Detector/Focal Plane Geometry

The location of each detector for each band is described in the ETM+ array reference coordinate system by using a pair of angles, one in the along-scan, or coordinate Y, direction and one in the across-scan, or coordinate X, direction. In implementing the IAS Level 1 processing algorithm, each of these angles is separated into two parts: the ideal band/detector locations used to model the relationship between the sensor image space and ground object space and the sub-pixel offsets unique to each detector, which are applied during the image resampling process. This approach makes it possible to construct a single analytical model for each band from which nominal detector projections can be interpolated and then refined during resampling, rather than performing the image-to-ground projection computation for each detector. The capability to rigorously project each detector, including all sub-pixel detector placement and timing offsets, has been developed to support geometric characterization and calibration.

Along-scan angles are modeled as the sum of the band odd detector angle from the optical axis, common to all detectors for one band, and the sub-pixel offset and time delay, unique to each detector within the band. The IAS algorithms assume that the angular displacement between even and odd detectors in each band is exactly compensated for by the time delay used to align nominally the even and odd detectors. Band center location angles, relative to the sensor optical axis, are stored in the Calibration Parameter File. The odd detector offset from band center (1.25 IFOVs for the 30-meter bands) is subtracted from the band center angle to yield the nominal detector locations. The along-scan angle is computed as:

along_angle = bandoff_along[BAND] – odd_det_off[BAND]

where:

    along_angle = nominal along-scan location of detector
    bandoff_along = along-scan band center offset (indexed by band)
    BAND = band number (1.8)
    odd_det_off = odd detector offset from band center (indexed by band)

The individual detector offsets from their ideal locations are stored in the Calibration Parameter File in the form of detector delays, which incorporate both sample timing offsets and detector along-scan positioning errors. These sub-pixel corrections are applied during the image resampling process.

Similarly, the across-scan angles are separated into the nominal detector location component, used for line-of-sight construction and projection, and the sub-pixel detector placement corrections unique to each detector, applied during resampling. The nominal detector location is computed as:

    cross_angle = bandoff_cross[BAND] + ((ndets[BAND]+1)/2 − N)*IFOV[BAND]

where:

    cross_angle = nominal across-scan location of detector N
    bandoff_cross = across-scan band center offset (indexed by band)
    BAND = band number (1.8)
    Ndets = number of detectors in band (indexed by band)
    N = detector number (1.ndets)
    IFOV = detector angular field of view (indexed by band)

Using these nominal detector locations in the line-of-sight projection computations makes it possible to project rigorously the first and last detectors in each band and to interpolate the image space-to-ground space transformation for the interior detectors. This reduces the computational load in Level 1 processing, where the unique sub-pixel detector placement corrections are applied by the resampling algorithm. To project an individual detector-to-ground space precisely, the sub-pixel along-scan and across-scan corrections are added to the nominal detector location.

3.1.5.1.2   Scan Mirror Model

The ETM+ scan mirror rotates about the sensor X-axis, providing nearly linear motion in the along-scan direction in both the forward and reverse motions. The scan mirror's departure from linear motion is characterized pre-launch using a set of fifth-order polynomials, which model the repeatable acceleration and deceleration deviations from linearity for forward and reverse scans. This fifth-order polynomial is then adjusted for high-frequency roll jitter components using ADS data from the PCD. Fifth-order mirror polynomials are also used to model the across-scan mirror deviation for forward and reverse scans. Thus, a total of four fifth-order polynomials characterize the scan mirror's repeatable deviation from nominal linear scanning: (1) along-scan deviation for forward

scans, (2) across-scan deviation for forward scans, (3) along-scan deviation for reverse scans, and (4) across-scan deviation for reverse scans. This set of four polynomials constitutes the nonlinear portion of the scan mirror profile.

The ETM+ instrument contains two sets of Scan Mirror Electronics (SME) used to control the mirror motion. Either SME can be used to drive the mirror in either operational mode: Scan Angle Monitor (SAM) mode or Bumper mode. For each mode, each set of SMEs exhibits a characteristic set of mirror profile polynomials, as well as scan start and scan stop mirror angles. This leads to a total of four sets of mirror profile polynomials: (1) SME1 SAM mode, (2) SME2 SAM mode, (3) SME1 Bumper mode, and (4) SME2 Bumper mode. The appropriate mirror profile coefficients and scan start/stop angles must be selected for a particular data set based on the SME and mode settings in the PCD.

Within each scan, the scan mirror assembly measures the deviation from the nominal time of scan from scan start to mid-scan (first half scan error) and from mid-scan to scan stop (second half scan error) and records this information in the scan line data minor frames for the following scan. These scan time errors are measured as departures from nominal, in counts, with each count being 0.18845 microseconds.

The first-half and second-half scan errors are used to correct the nominal scan profile by adjusting the fifth-order, along-scan polynomial to compensate for the actual mid-scan and end-of-scan times. This correction is applied by rescaling the polynomial coefficients to the actual scan time and by computing correction coefficients, which are added to the linear and quadratic terms in the fifth-order polynomial. The calculation procedure is given in 3.1.5.3.3.

3.1.5.1.3   Scan Line Corrector Mirror Model

The Scan Line Corrector (SLC) mirror rotates the ETM+ line of sight about the sensor Y-axis to compensate for the spacecraft's along-track motion during the scanning interval. Although the IAS algorithms support fifth-order correction polynomials for the SLC profile, in practice, no significant SLC non-linearity has been observed for Landsats 4, 5, and 6. Refer to 3.1.5.3.4 for more information on the Scan Line Corrector mirror model.

3.1.5.2   Landsat 7 Platform Model

3.1.5.2.1   Sensor Pointing (Attitude and Jitter) Model

Sensor line-of-sight vectors constructed using the ETM+ instrument model must be related to object (ground) space by determining the orientation and position of the sensor coordinate system with respect to object space (ECI coordinates) as a function of time. The sensor orientation is determined using the sensor pointing model, which includes the effects of sensor-to-spacecraft (navigation reference base) alignment, spacecraft attitude as determined by the on-board computer using the inertial measurement unit gyros and star sightings, and the high-frequency attitude variations (jitter) measured by the angular displacement assembly. The inputs to this model are the attitude quaternions, gyro drift

estimates, raw gyro outputs, and ADS outputs contained in the Payload Correction Data, as well as the sensor alignment matrix measured pre-launch and updated in flight by the Image Assessment System using the sensor alignment calibration algorithm described in 3.1.5.4.1.

The Landsat 7 attitude control system generates estimates of the spacecraft attitude every PCD major frame (4.096 seconds). These estimates are provided in the form of attitude quaternions, which describe the orientation of the spacecraft navigation reference system with respect to the ECI (J2000) coordinate system. These estimates are derived from the IMU gyro data combined with periodic star transit observations from the Celestial Sensor Assembly. As a byproduct of the integration of the gyro and star tracker observations, the ACS also computes estimates of gyro drift rates. A gyro drift rate estimate is also included in each PCD major frame, but changes only after a star sighting.

The raw outputs from each of the three gyro axes are recorded in the PCD every 64 milliseconds. In Landsats 4, 5, and 6, there was a 28-millisecond delay between the gyro sampling time and the PCD major frame start time (i.e., gyro sampling was indexed from PCD start time 28 milliseconds). From Reference 1, it appears that this offset does not apply to Landsat 7. The 15.625-Hertz (Hz) gyro sampling rate is not sufficient to capture the high-frequency spacecraft angular dynamic displacement (jitter). This sampling frequency captures attitude variations up to the Nyquist frequency of 7.8125 Hz, but aliases the higher frequencies to the extent that the gyros respond to these higher frequencies. This response is expressed in the IMU transfer functions. For Landsats 4 and 5, the transfer functions for each IMU axis were taken to be the same.

The high-frequency jitter information is measured by the angular displacement sensors (ADS), which are sampled at 2-millisecond intervals and are sensitive to frequency components between approximately 2 and 125 Hz. Like the gyros, the ADS samples are offset from the PCD major frame start time. Unlike the gyros, the ADS axes are sampled at different times. The ADS samples begin at the PCD major frame time plus 375 microseconds for the X axis, plus 875 microseconds for the Y axis, and plus 1,375 microseconds for the Z axis. The 500-Hz ADS sampling frequency could capture jitter frequencies up to 250 Hz, but the output of the ADS is band-limited to the maximum expected frequency of 125 Hz by a pre-sampling filter. Historically, the three ADS axes have had different transfer functions. For Landsats 4 and 5, the ADS transfer functions were not separated from the pre-sampling filter, but were provided as a net response. For Landsat 6, the pre-sampling filter transfer function was provided separately.

The gyro and ADS transfer functions are used to blend the gyro/attitude and ADS/jitter data in the frequency region where their pass bands overlap from 2 to 7 Hz. The IAS uses the classical approach defined by Sehn and Miller (

Reference 10) to construct the crossover filters used to combine the gyro and ADS data. This is described in more detail in 3.1.5.3.3.

3.1.5.2.2   Platform Position and Velocity

The spacecraft state vectors contained in the Payload Correction Data (PCD) stream provide position and velocity information in the Earth-Centered Inertial of epoch J2000 coordinate system every PCD major frame (unlike earlier Landsats, which only provided new state vectors every other major frame). These state vectors are interpolated on-board using the uplinked predicted ephemeris and may contain significant errors. Gross blunders (i.e., transmission errors) are detected and removed during PCD pre-processing. The systematic errors that accumulate in the predicted ephemeris are estimated and removed as part of the precision correction algorithm described in 3.1.5.3.8.

3.1.5.3   Level 1 Processing Algorithms

The diagrams that follow describe the high-level processing flows for the IAS Level 1 processing algorithms. Figure 3-9 describes the process of initializing and creating the Landsat 7 geometry model. Figure 3-10 shows the process of creating a Geometric Correction Grid and the application of that grid in the resampling process. Figure 3-11 describes the process of refining the Landsat 7 geometry model with ground control, resulting in a precision geometry model. Figure 3-12 again describes the creation of a Geometric Correction Grid (this time precision) and resampling with terrain correction.

Detailed algorithms for each of the main process boxes in these diagrams are given in the sections that follow.
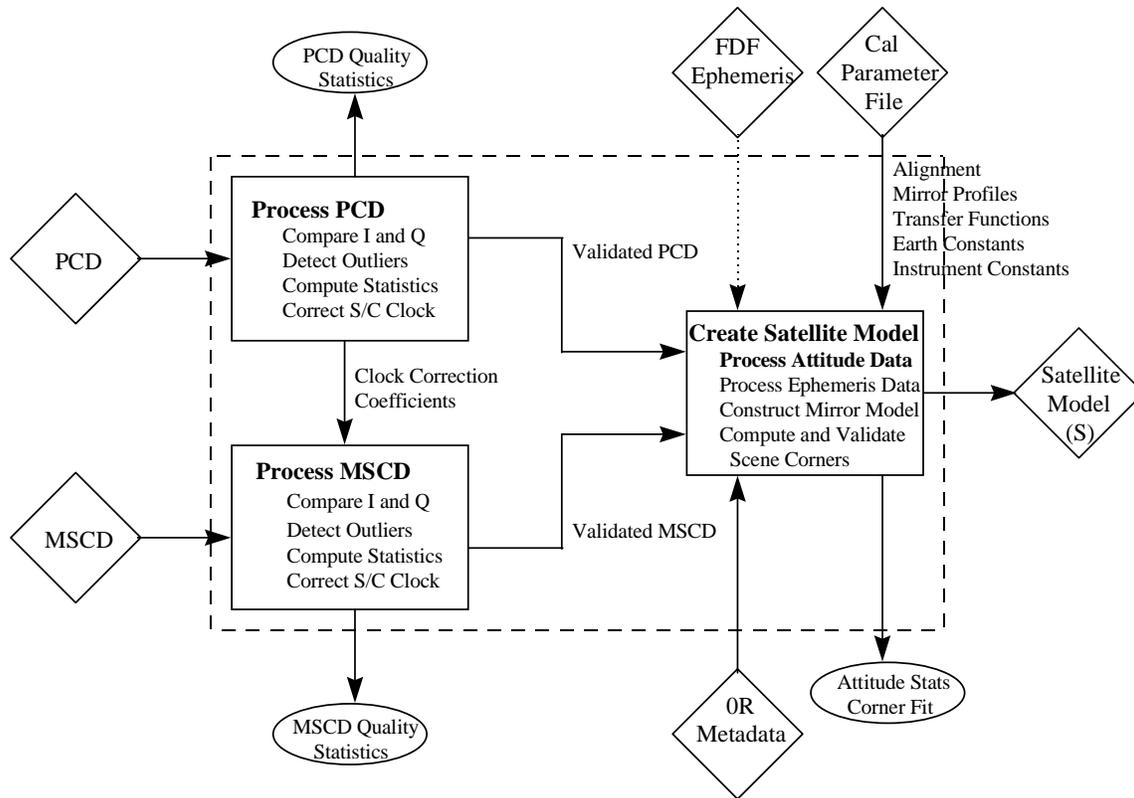
Figure 3-9. Model Initialization
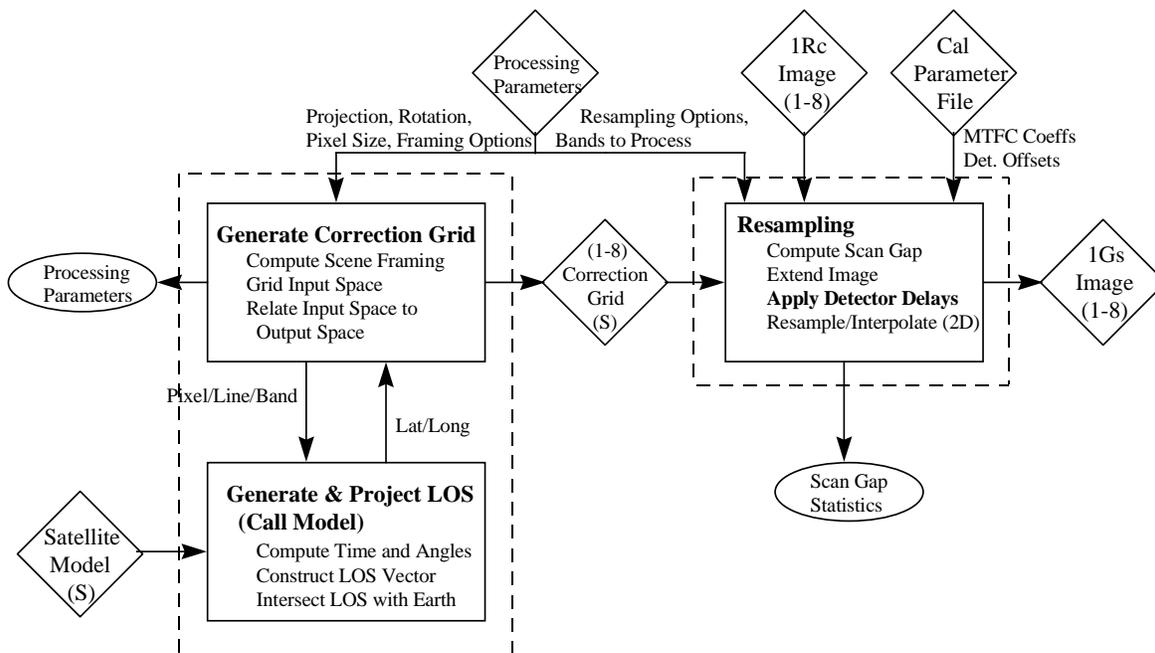
Figure 3-10. Rectification and Resampling
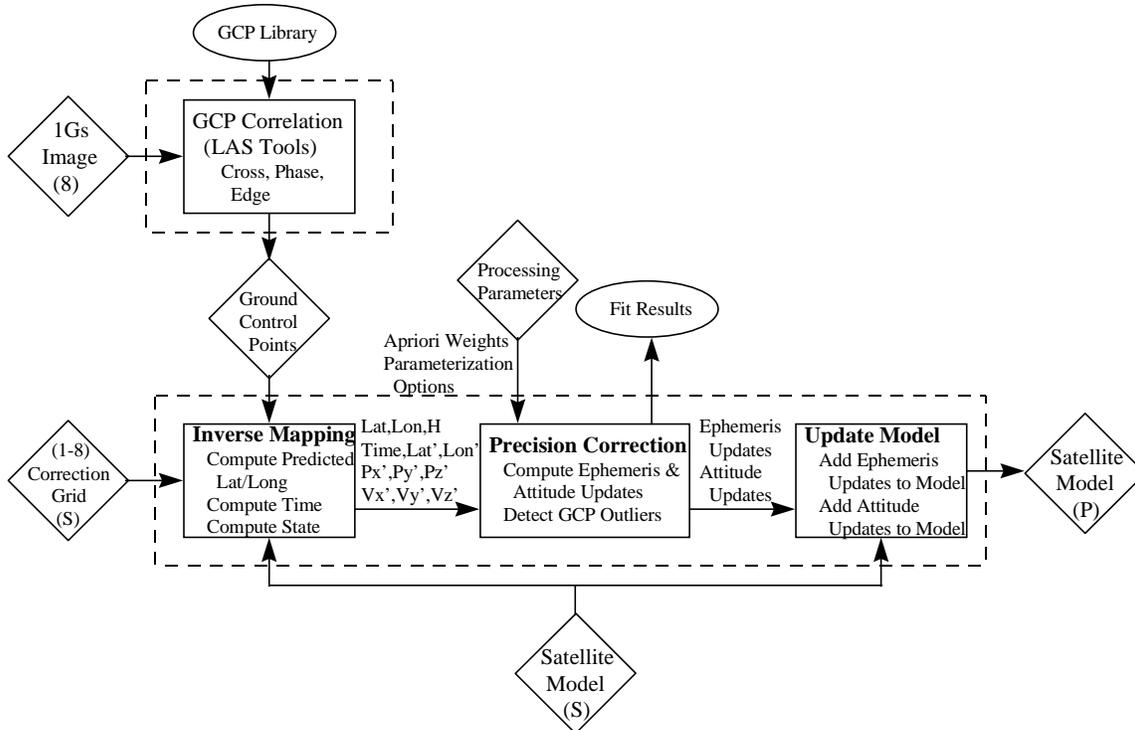
Figure 3-11. Precision Correction Solution
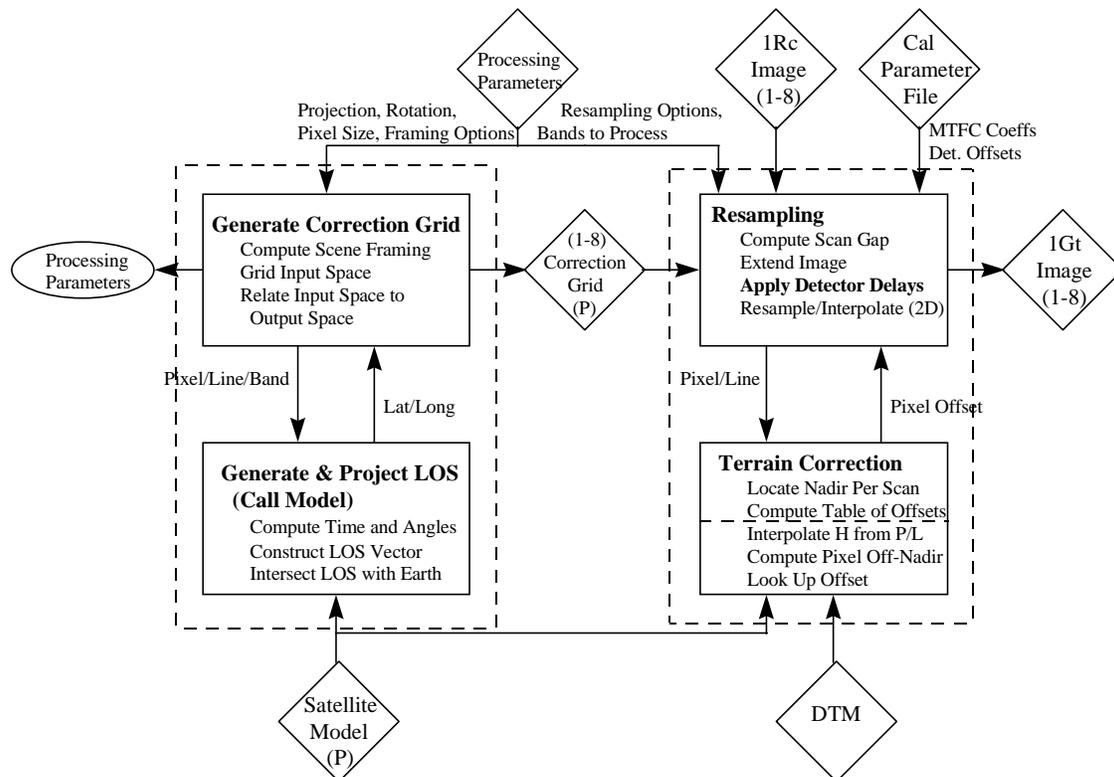
Figure 3-12. Precision/Terrain Correction

LS-IAS-01
Version 1.0

3.1.5.3.1   PCD Processing

The Payload Correction Data (PCD) contain information on the state of the satellite and sensor. The PCD information is transmitted in both the In-Phase (I) and Quadrature (Q) channels of the wide-band data. The two PCD data streams are converted to engineering units and stored in Hierarchical Data Format (HDF) format for the Level 0R product. The I channel is used as the default PCD for processing. The PCD is recorded for the entire sub-interval or full interval. The PCD information required for geometric processing includes the ephemeris, the spacecraft attitude, the gyro data, the gyro drift rate data, the ADS data, the SME flag, the gyro unit flag, and the spacecraft clock corrections. The ephemeris and spacecraft attitude are updated every major frame; the gyro data are updated 64 times per axis in a major frame; and the ADS is updated 2,048 times per axis in a major frame. A PCD cycle includes four PCD major frames. Housekeeping values are also included in the PCD.

3.1.5.3.1.1   Validate Ephemeris Data

The PCD is transmitted in both the I and Q channels of telemetry data. Under normal conditions, the I and Q channels are identical. Therefore, a comparison between the ephemeris values contained in the I and Q channels is used as a validation test. Any differences are flagged and reported in the processing history file. The I channel is assumed to be correct, unless the other validation checks determine it to be in error.

The semi-major axis, inclination, and angular momentum of the Landsat 7 orbit, calculated from the ephemeris data, should not deviate substantially from their nominal values. A second validation test compares these values to their nominal values, and any large deviations are flagged. If a large deviation is detected, the Q channel's values are checked. If the Q channel values are correct, they are used instead of the I channel values. Any ephemeris points that do not pass the I and Q channel tests are not used in the interpolation routine. The average of and the standard deviation from the nominal value for the semi-major axis, inclination, and angular momentum for the scene to be processed are saved for trending. The equations to calculate the satellite's semi-major axis, inclination, and angular momentum are as follows:

$$\text{Angular momentum} = |\ R \times V\ |$$

where R is the satellite's position vector, and V is the satellite's velocity vector.

$$\text{Inclination} = \text{acos}(|H \text{ dot } k| / |H|)$$

where H = R x V, and k is the unit vector in the z-direction.

$$\text{Semi-major axis} = -\mu / (2.0 * E)$$

where $E = |V|^2 / 2.0 - \mu / |R|$ and $\mu = G * M$ (G is the Earth's gravity constant and M is the mass of the Earth).

3.1.5.3.1.2   Validate Spacecraft Attitude Data

The on-board computer (OBC) calculates a flight segment attitude estimate every 512 milliseconds. The OBC generates one of eight sets of spacecraft attitude data in the telemetry every 4.096 seconds. The attitude information is generated in the form of Euler parameters that specify the vehicle attitude relative to the Earth-centered inertial frame (J2000). The spacecraft's attitude is calculated using star crossings, gyro data, and ADS data. Care must be taken when using the spacecraft's attitude data, because star crossing updates may cause discontinuities. The IAS converts the Euler parameters to roll, pitch, and yaw, which are then used in the model.

The I and Q channel values are compared, and any differences are flagged. The spacecraft's attitude data should not deviate greatly from the calculated value that was found using a linear interpolation of the values around the value to be checked. The linear interpolated value is used to validate the spacecraft's attitude. Any large deviations are flagged. If a large deviation is detected, the Q channel is validated. If the Q channel value passes the validation test, it is used instead of the I channel's value. If the attitude data in both the I and Q channels have large deviations, the interpolated value is used. The Euler parameters should satisfy the following equation:

$$EPA_1{}^2 + EPA_2{}^2 + EPA_3{}^2 + EPA_4{}^2 = 1 +/- \varepsilon$$

where EPA_1, EPA_2, EPA_3, and EPA_4 are the Euler parameters that have been converted by the Landsat 7 Processing System (LPS). The ε is some very small error term due to system noise.

The I channel values are validated using the above equation, and any deviation is flagged. If a deviation occurs, the Q channel values are checked. If both channels fail the validation test, the interpolated value for each of the Euler parameters is used.

The averages and standard deviations of the Euler parameters and the deviation from the equation above are stored for trending.

3.1.5.3.1.3   Validate Gyro Data

The I and Q channel values are compared, and any differences are flagged. The gyro data should not deviate greatly from the calculated value. This calculated value is found using a forward and backward differencing of the values around the value in question.

The steps include:

1. Checking the current gyro value against a tolerance
2. Doing a forward prediction for each point using the two previous points and comparing this predicted value against the observed value
3. Doing a backward prediction for each point using the next two points and comparing this predicted value against the observed value

Any point that passes (1) and either (2) or (3) is a good point. If a deviation is detected in the I channel, the Q channel is checked. If the Q channel passes the validation test, it is used in the processing. If both the I and Q channel values fail, the value in question is flagged as an outlier. After all values are checked, the flagged values are replaced. Linear interpolation is used to replace all outliers. The first valid points before and after the outlier are found. These two values are then used to interpolate all outliers that lie in between. Care must be taken to account for counter resets in this check. The register resets after ($2^{23} - 1$) positive and -2^{23} negative.

3.1.5.3.1.4   Validate Gyro Drift Data

The I and Q channel values are compared, and any deviations are flagged. The values for each major frame are compared. Any changes in the I channel are compared to the values in the Q channel to determine if changes have occurred in the Q channel. If both channels display the same change, and the change is determined to be within an acceptable level, then it is assumed that a star sighting has occurred. The PCD major frame that displays the change is flagged, and the magnitude of the change is saved for trending.

3.1.5.3.1.5   Validate ADS Data

The I and Q channel values are compared, and any differences are flagged. The ADS data should not deviate greatly from the calculated value that was found using a linear interpolation of the values around the value in question. Any large deviations from the calculated value are flagged. If a large deviation is detected, the Q channel value is checked using the same interpolation method. If the Q channel's value passes the validation test, the Q channel's value replaces the I channel's value. If both channels fail the test, the value in question is corrected using the interpolated value.

The linear interpolation test has been verified using an empirical test of Landsat 5 data. The results showed that the difference between linear predicted and actual ADS data points are typically within 10 counts. For a test scene over Iowa, after five actual outliers that had deviations of several thousand counts were removed, the maximum deviations in roll, pitch, and yaw were 12, 11, and 11 counts, respectively.

3.1.5.3.1.6   Validate Spacecraft Time Correction

The I and Q channel values are compared, and any differences are flagged. The number of occurrences are saved for trending. The Mission Operations center attempts to

maintain the on-board spacecraft clock to 145 milliseconds. A validation test calculates the correction of the clock, and this value should not exceed the 145-millisecond value. If the absolute value of the clock correction is much greater than 145 milliseconds, the Q channel values are checked. If the Q channel clock correction values are less than 145 milliseconds, the Q channel's values are used. If both the I and Q channel values are greater than 145 milliseconds, then the clock coefficients are set to zero.

The clock correction is:

$$dt = \text{spacecraft time} - \text{clock update time}$$
$$\text{ClockCorrection}_I = \text{TimeCoef1}_I + \text{TimeCoef2}_I * dt + 0.5 * \text{TimeCoef3}_I * dt^2$$

3.1.5.3.1.7   Validate and Save Instrument On-Time

The I and Q channel values for the instrument on-time are compared. Any differences are flagged, and the values are saved. The difference between the instrument on-time and the start of the first major frame is saved for trending and analysis.

3.1.5.3.1.8   Correct PCD Spacecraft Time

The clock correction is added to the major frame times.

3.1.5.3.1.9   Other Geometric PCD Parameter Validations

Other PCD geometric processing parameters, such as the SME mode and gyro unit flags, are validated by comparing the I and Q channel values. Any differences are flagged and reported. The I channel is used as the default value.

3.1.5.3.2   MSCD Processing

For Landsat 7, the Mirror Scan Correction Data (MSCD) are contained in the HDF formatted Level 0R product. The counted line length, scan direction, First Half Scan Error (FHSERR), and Second Half Scan Error (SHSERR) are associated with the previous scan. The scan start time, however, is for the current scan. Due to the association of the counted line length, scan direction, FHSERR, and SHSERR with the previous scan, the number of MSCD records must be one more than the number of scans. The MSCD for the output product is subsetted to match the imagery when the L0R product is generated.

The values from the MSCD required for ETM+ processing are:

- FHSERR
- SHSERR
- Scan Direction
- Scan Start Time
- Counted line length

3.1.5.3.2.1   Validate Scan Direction

The I and Q channels are checked for consistency, and any differences are flagged and reported in the processing history file. The IAS assumes that the I channel has the correct value.

The flag for scan direction is 1 or 0. The scan direction is for the previous scan. The validation test checks the first scan direction flag and the direction flags thereafter. Any errors in the direction flags are corrected using the first valid direction flag as a reference. The errors are flagged and reported in the processing history file and are saved for trending analysis.

3.1.5.3.2.2   Validate FHSERR and SHSERR

The MSCD is transmitted in both the I and Q channels of the telemetry data. Under normal conditions, the I and Q channels are identical. Therefore, a comparison of the FHSERR and SHSERR values contained in the I and Q channels is used as one of the validation tests. Any differences detected are flagged and reported in the processing history file. The I channel is assumed to be correct, unless the other validation check determines it to be in error.

The FHSERR and SHSERR values should not deviate greatly from their nominal values. After the nominal values for the FHSERR and SHSERR have been characterized, the FHSERR and SHSERR are checked for deviations from their nominal values. Large deviations are flagged, and the average difference and its standard deviation are saved for trending. If a large deviation is detected, the Q channel is validated. If the Q channel passes the validation check, the Q channel value is used in processing.

3.1.5.3.2.3   Validate Scan Start Time

The I and Q channels are checked for consistency, and any differences are flagged and reported in the processing history file. The IAS assumes that the I channel has the correct value.

The difference in time between the start of scan times should not vary greatly from the nominal value. The start of scan time is validated by comparing the difference in time between scans. Any large variation from nominal is flagged, and the Q channel is checked for its difference between start times. If the Q channel values pass the validation

check, it will be used during processing. Otherwise, the scan start time is corrected using the nominal value. The average difference and the standard deviation between the scans times for the scene being processed are saved for trending.

3.1.5.3.2.4   Validate Counted Line Length

The I and Q channel values are compared, and any differences are flagged and reported in the processing history file. The IAS assumes that the I channel has the correct value.

The counted line length is the number of minor frames sampled from the start-of-scan to the end-of-scan mark. The counted line length should be the same as the truncated calculated line length, which is found using the FHSERR, SHSERR, and DWELLTIME (as well as the conversion of counts to seconds). The counted line length should not deviate greatly from its nominal value. The validation algorithm for the counted line length checks the deviation of the counted line length from its nominal value and its deviation from the truncated calculated line length. Large deviations from the nominal counted line length and any deviation from the calculated line length are flagged. If a large deviation is detected, the Q channel counted line length is compared to the calculated line length, and if it passes, the Q channel is used in the processing. The average counted line length and standard deviation of the counted line length are saved for trending.

The first-half scan time, the second-half scan time, and the total scan time are calculated using the following equations.

For each forward scan:

> first_half_time = T_{fhf} – FHSERR * T_{unit}
> second_half_time = T_{shf} – SHSERR * T_{unit}
> total_scan_time = first_half_time + second_half_time

For each reverse scan:

> first_half_time = T_{fhr} – FHSERR * T_{unit}
> second_half_time = T_{shr} – SHSERR * T_{unit}
> total_scan_time = first_half_time + second_half_time

The line length is calculated using:

> calculated_line_length = total_scan_time / DWELLTIME

where:

> T_{fhf} = the forward nominal first half scan time
> T_{shf} = the forward nominal second half scan time
> T_{fhr} = the reverse nominal first half scan time
> T_{shr} = the reverse nominal second half scan time
> T_{unit} = the conversion factor from counts to time

3.1.5.3.2.5   Correct Spacecraft Time in the MSCD

The correction value for the spacecraft time is found in the process PCD module. This value must be added to the MSCD start-of-scan times.

3.1.5.3.3   Platform/Sensor Model Creation

The platform/sensor model creation algorithm reads the telemetry data from the PCD and the MSCD. These data are converted to a form that is usable by the geometric model. The gyro and ADS data are combined and converted to the common Navigational Reference Base system to create a time-ordered table of attitude information. The ephemeris data are converted to the Earth-Centered, Earth-Fixed (ECEF) coordinate system and fit with a polynomial that is a function of time. The mirror scan fifth-order polynomial is corrected for each scan's variation in velocity.

3.1.5.3.3.1   Calculate Satellite State Vector

For each valid ephemeris data point, the following coordinate transformation is used to convert the satellite position and velocity from the inertial system (J2000.0 system for Landsat 7; true-of-date system for Landsats 4 and 5) to the Earth Fixed system:

- Transform the J2000.0 system to the mean-of-date system through the precession matrix (Landsat 7 only).
- Transform the mean-of-date system to the true-of-date system through the nutation matrix (Landsat 7 only).
- Transform the true-of-date system to the Earth Fixed system through the Earth rotation and polar-motion matrix. The Earth Orientation Parameters (UT1-UTC, xp, yp) are passed in from the Calibration Parameter File.

Two sets of polynomial coefficients are generated, one for the orbit position and velocity in the Earth Fixed system and the other for the orbit position and velocity in the inertial system (J2000.0 for Landsat 7; true-of-date system for Landsat 5). The methods for generating the coefficients for the two sets are the same: solving the Vandermonde equation system using an algorithm given in Reference 6. The order of the polynomial is automatically chosen as the number of valid ephemeris data points in the scene; i.e., an even-determined fit with no redundancy.

The polynomial in the Earth Fixed system is used to interpolate the orbit position and velocity at each (grid point) sensor time for the purpose of constructing a look vector and intersecting the Earth model in the Earth Fixed system. The polynomial in the inertial system is used to interpolate the orbit position and velocity at each gyro data point time for the purpose of correcting the portion of gyro observation caused by satellite motion.

3.1.5.3.3.2   Process Gyro Data

The Inertial Measurement Unit (IMU), which houses the gyroscopic units, is sampled every 64 milliseconds for each of the three axes. The sampled values are referred to as

the gyro data. This results in 64 samples per axis per PCD major frame or 256 samples per axis per PCD cycle. The gyro data register reset if a positive count occurs when the register's value is a positive $2^{23} - 1$ or if a negative count occurs when the register's value is a negative 2^{23}. The time of the gyro data samples is referenced from the start of a PCD cycle by the following equation:

Gyro_data_sample_time = 64N milliseconds

where:

N = 0 ... 255

The OBC calculates the gyro drift rates for each of the axes, using the results of star sightings. The results of the OBC gyro drift calculations are in the attitude control system (ACS) system. The gyro drift parameters are updated asynchronously at a rate of up to once per minute. When a gyro drift value changes, the value is updated at the PCD time code for the cycle minus 8.192 seconds.

3.1.5.3.3   Processing Summary

High-level flow of gyro processing is as follows:

1. The register resets are accounted for.
2. For each axis, the first gyro value is subtracted from the rest.
3. The gyro data are rotated to the ACS system.
4. The orbital drift components are removed.
5. The gyro, gyro drift, and quaternion data are combined.
6. The data from step 5 are rotated to the ADS system.
7. The data from step 6 are filtered using the G-filter.
8. The data from step 7 are synchronized with the ADS data.
9. The ADS data are synchronized.
10. The data from step 8 are combined with the ADS data.
11. The combined data are filtered using the F-filter.
12. The combined data are rotated to the ACS system.

For more detail on step 5, see 3.1.5.3.3.11, Smoothing Low Frequency Information.

For more detail on steps 6, 8, 9, 10, 11, and 12, see 3.1.5.3.3.12, Combining Low- and High-Frequency Attitude Information.

3.1.5.3.3.4   Processing Gyro Data to a Common (ADS) Reference System

Processing of the gyro data to a common ADS reference system is as follows:

1. Since the LPS converts the gyro data counts to arc-seconds, the register reset correction must be accomplished using arc-seconds as the units. Thus, a positive reset occurs at 511705.027 arc-seconds, and a negative reset occurs at –

511705.088 arc-seconds. After a reset is detected, the values following the reset are offset by the appropriate reset value.

2. Due to the gyro data measuring changes in attitude, the first value for each of the axes, which corresponds with the spacecraft's attitude values, is subtracted from the rest of the values.

3. Each axis of the gyro data is rotated to the ACS reference system using a pre-launch-derived transformation matrix. This is accomplished using the following:

> Define A as the rotation matrix due to roll. Define B as the rotation matrix due to pitch, and define C as the rotation matrix due to a yaw. Then the Perturbation matrix P is defined as:
>
> P = CBA
>
> Use the Transformation matrix from the gyro axes to the ACS [IMU2ACS] and its inverse to find the P matrix in the ACS reference system or P':
>
> $$P' = [IMU2ACS] \; P \; [IMU2ACS]^{-1} \;\; = \begin{bmatrix} a\,b\,c \\ d\,e\,f \\ g\,h\,i \end{bmatrix}$$
>
> Then the transformed roll (r') is:
>
> $$r' = -\tan^{-1}(\,h\,/\,i)$$
>
> where h and i are elements of P'
>
> Then the transformed pitch (p') is:
>
> $$p' = \sin^{-1}(g)$$
>
> where g is an element of P'
>
> and the transformed yaw (y') is:
>
> $$y' = -\tan^{-1}(\,d\,/\,a)$$
>
> where d and a are elements of P'

4. The drift of the gyro data due to the orbital motion is removed using the polynomial for the ephemeris data in the inertial system derived in the Calculate Satellite State Vector subroutine.

5. The gyro, gyro drift, and quaternion data are combined into one data stream (details of step 5 can be found in 3.1.5.3.3.11, Smoothing Low-frequency Attitude Information).

6. Each axis of the gyro data is transformed to the ADS reference system. This is accomplished by combining the ACS-to-ETM+ transformation matrix and the ETM+-to-ADS transformation matrix. The ACS-to-ETM+ transformation matrix is the inverse of the ETM+-to-ACS matrix. The ETM+-to-ACS matrix is first determined pre-launch and is updated after launch. Its values are contained in the Calibration Parameter File. The ETM+-to-ADS matrix is determined pre-launch, and its members are contained in the Calibration Parameter File.

The transformation from the ACS system to the ADS is accomplished using the following:

Define A as the rotation matrix due to roll. Define B as the rotation matrix due to pitch, and define C as the rotation matrix due to a yaw. Then the Perturbation matrix P is defined as:

P = CBA

Combine the Transformation matrix from the ACS to the ETM+ T and the transformation matrix from the ETM+ to the ADS system R. Calculate the combined matrix S inverse. Then use S and its inverse to find the P matrix in the ADS reference system or P':

S = RT

$$P' = SPS^{-1} = \begin{bmatrix} a\ b\ c \\ d\ e\ f \\ g\ h\ i \end{bmatrix}$$

Then the transformed roll (r') is:

$$r' = -\tan^{-1}(h / i)$$

where h and i are elements of P'

Then the transformed pitch (p') is:

$$p' = \sin^{-1}(g)$$
where g is an element of P'

Then the transformed yaw (y') is:

$$y' = -\tan^{-1}(d / a)$$

where d and a are elements of P'

7. Apply the G-filter (refer to 3.1.5.3.3.11). Steps 7, 8, 9, 10, and 11 are shown in more detail in 3.1.5.3.3.12, Attitude Processing.

8. Calculate the inverse of step 7:

$$P'' = S^{-1}P'S$$

3.1.5.3.3.5   Spacecraft Attitude Processing

The OBC calculates a flight segment attitude estimate every 512 milliseconds. The OBC outputs one of the eight sets of data in the telemetry every 4.096 seconds or once a PCD major frame. The attitude is output in the form of Euler parameters that specify the vehicle's attitude relative to the Earth-Centered Inertial (ECI) reference frame. The Euler parameters (EPA_1, EPA_2, EPA_3, and EPA_4) are components of the reference quaternion, as propagated from the gyro data, which defines the spacecraft's attitude. Components 1–3 define the Eigen axis of the rotation in ECI coordinates, and the fourth component defines the rotation about the Eigen axis.

Each PCD cycle includes four attitude estimates. The time associated with the attitude data contained within the PCD can be derived from the time code contained in words 96–102 of the first PCD major frame in the cycle. The derivation for Landsat 7 (see Reference 1, pages 3–29) is as follows:

PCD Major Frame	Time Computation
0	PCD time code $-$ 8.192 seconds
1	PCD time code $-$ 4.096 seconds
2	PCD time code
3	PCD time code $+$ 4.096 seconds

3.1.5.3.3.6   Processing

The PCD attitude quaternions define the attitude of the spacecraft at a reference time. This attitude measurement reflects the gyro information up to that point. The IAS determines the spacecraft low-frequency attitude by processing the gyro, gyro drift, and quaternions into one data stream by using a Kalman filter/smoother. This integrates attitude measurements from all time into a single best estimate of the attitude over an entire interval. The average associated with this data stream serves as the scene reference attitude state. The average is removed from the data stream. This data stream is then combined with the ADS information to get the change in state of the attitude from this reference state. To accomplish this, the Euler parameters need to be converted to ACS roll, pitch, and yaw values.

The direction cosines (transformation) matrix from the ACS reference axis to the ECI reference system (ACS2ECI) is defined as:

ACS2ECI =

$$\begin{bmatrix} EPA_1{}^2 - EPA_2{}^2 - EPA_3{}^2 + EPA_4{}^2 & 2(EPA_1 EPA_2 - EPA_3 EPA_4) & 2(EPA_1 EPA_3 + EPA_2 EPA_4) \\ 2(EPA_1 EPA_2 + EPA_3 EPA_4) & -EPA_1{}^2 + EPA_2{}^2 - EPA_3{}^2 + EPA_4{}^2 & 2(EPA_2 EPA_3 - EPA_1 EPA_4) \\ 2(EPA_1 EPA_3 - EPA_2 EPA_4) & 2(EPA_2 EPA_3 + EPA_1 EPA_4) & -EPA_1{}^2 - EPA_2{}^2 + EPA_3{}^2 + EPA_4{}^2 \end{bmatrix}$$

The ACS2ECI transformation matrix can also be defined as the product of the inverse of the spacecraft's attitude perturbation matrix P and the transformation matrix from the orbital reference system to the ECI reference system (ORB2ECI).

$$ACS2ECI = [ORB2ECI][P^{-1}]$$

The orbital reference system to ECI matrix must be defined at the same time as the spacecraft's attitude matrix. The ORB2ECI matrix is constructed of the negative of the spacecraft's position vector divided by its magnitude (z-direction vector), the cross of the z-direction vector with the spacecraft's velocity vector divided by its magnitude (y-direction vector), and the cross of the y-direction vector with the z-direction vector divided by its magnitude (x-direction vector). The resulting ORB2ECI matrix is:

$$ORB2ECI = \begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix}$$

where:

Z_x is the x component of the z-direction vector
Z_y is the y component of the z-direction vector
Z_z is the z component of the z-direction vector
X_x is the x component of the x-direction vector
X_y is the y component of the x-direction vector
X_z is the z component of the x-direction vector
Y_x is the x component of the y-direction vector
Y_y is the y component of the y-direction vector
Y_z is the z component of the y-direction vector

The roll, pitch, and yaw values are contained in the P^{-1} matrix, thus:

$$P^{-1} = [ORB2ECI]^{-1}[ACS2ECI] = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

Then the spacecraft roll (r) is:

$r = \tan^{-1}(h / i)$

where h and i are elements of P^{-1}

Then the spacecraft pitch (p) is:

$p = -\sin^{-1}(g)$

where g is an element of P^{-1}

Then the spacecraft yaw (y) is:

$y = \tan^{-1}(d / a)$

where d and a are elements of P^{-1}

3.1.5.3.3.7   Converting ADS to Common Reference System

The Attitude Displacement Sensor Assembly (ADSA) consists of three nominally orthogonal ADS sensors. The ADSA is mounted on the ETM+ telescope. A digital

sample count of 0 is the maximum positive angular displacement, and a digital count of 4,095 is the maximum negative angular displacement. The Least Significant Bit (LSB) of each count is $250 / 2^{11}$ microradians. The nominal zero angular displacement output of the ADSA is 2,048 counts. Each axis of the ADSA is sampled every 2 milliseconds, resulting 8,192 samples of each axis in a PCD cycle. The axes are sampled sequentially, as given below:

ADSA AXIS   Sampling Times

Roll   PCD Time Code + (2N + 3/8) milliseconds
Pitch   PCD Time Code + (2N + 7/8) milliseconds
Yaw   PCD Time Code + (2N + 11/8) milliseconds

where:

N = 0,1,2,...,8191 for the given axis

Each axis of the ADSA has a nominal 2.0- to 125.0-Hz bandwidth. The transfer function-to-rotational motion is measured pre-launch and is supplied in the Calibration Parameter File. The nominal relative alignment matrix between the ADS and the ETM+ Sensor is determined pre-launch and is also supplied in the Calibration Parameter File.

3.1.5.3.3.8   Processing

The first three steps are summarized here. Refer to 3.1.5.3.3.12, Combining Low- and High-Frequency Attitude Information, for more information. Each of the samples for each axis is transformed to rotational motion using the transfer function.

1. Synchronize the ADS data.

2. Combine the ADS data with the smoothed, low-frequency attitude data.

3. Rotate data to the ACS system. Due to the ADSA being mounted on the ETM+ telescope, the transformation to the ACS reference system requires the ADSA measurements to be transformed to the ETM+ reference system and then transformed through the ETM+ sensor to the ACS alignment matrix. Since the ETM+-to-ACS reference system matrix is determined after launch, this transformation must be a 2-step process.

   Define the transformation matrix from the ADSA-to-ETM+ sensor as ADS2ETM and the transformation matrix from the ETM+-to-ACS as ETM2ACS. Then the transformation matrix between the ADSA and the ACS (ADS2ACS) is:

   ADS2ACS = [ETM2ACS][ADS2ETM]

To transform the axis of the ADSA, the perturbation matrix P must first be calculated. To accomplish this, define matrix A as the rotation matrix due to roll, matrix B as the rotation matrix due to pitch, and a matrix C as the rotation matrix due to yaw. Thus, the perturbation matrix is:

P = CBA

Then using the transformation matrix ADS2ACS and its inverse, the perturbation matrix in the ACS reference P' is found:

$$P' = [ADS2ACS]P[ADS2ACS]^{-1} = \begin{bmatrix} a\ b\ c \\ d\ e\ f \\ g\ h\ i \end{bmatrix}$$

Then the transformed roll (r') is:

$$r' = -\tan^{-1}(h / i)$$

where h and i are elements of P'

Then the transformed pitch (p') is:

$$p' = \sin^{-1}(g)$$

where g is an element of P'

Then the transformed yaw (y') is:

$$y' = -\tan^{-1}(d / a)$$

where d and a are elements of P'

3.1.5.3.3.9   Calculate Scan Mirror Correction

The scan mirror (as introduced in 3.1.5.1.2) is modeled using a linear motion with a fifth-order polynomial correction. A fifth-order polynomial is determined during the ground testing of the instrument; this along with an adjustment due to high-frequency roll jitter components combine to give an overall fifth-order polynomial correction. Along with the high-frequency jitter component, in-orbit perturbations within the satellite cause the scan mirror velocity to change from scan to scan.

The scan mirror has sensors attached to it that measure the mirror position at the start of scan, mid-scan, and end of scan. The data from these sensors are output in the form of counts, which are the deviation from nominal scan mirror position in time. For example, the scan mirror crossed the mid-scan detector so many counts ahead (or behind) of the nominal time. The unit of the counts is 16/84.9037e-6 seconds or 0.18849 microseconds.

In the IAS model of the scan mirror motion, the mirror starts at a positive angle (A_{sm}) and becomes more negative as a function of time, until it reaches A_{me}, where A_{sm} is the angle from the start of scan to the mid-scan position, and A_{me} is the angle of the mirror from mid-scan to end of scan. This sign convention is used to place the scan mirror motion in the spacecraft coordinate system. This convention is different from the convention used in References 1 and 2.

Remaining in the true time domain for the forward and reverse scan direction, a positive result from the polynomial corresponds to the mirror being ahead of its linear location, and a negative result corresponds to the mirror being behind its linear location. However, whether the results are added to or subtracted from the linear portion depends on the slope of the linear motion. For example, using the IAS sign conventions, adding a positive polynomial value to the forward scan direction causes the scan mirror to be behind its linear location. Thus, for the forward scan direction, a positive polynomial value must be subtracted from the linear location or the sign of each of the coefficients must be changed. The latter was chosen. For a reverse scan, a positive polynomial value added to the linear location corresponds to the mirror being ahead of it linear location. Thus, adding the polynomial value is correct. The MSCD values of scan direction, counted line length, first-half scan error, and second-half scan error are for the previous scan.

Define:

$b_{j = 0, 5}$ The forward scan polynomial coefficients for the SME mode
$d_{j = 0,5}$ The reverse scan polynomial coefficients for the SME mode
A_{smf} The forward start to middle angle for the SME mode
A_{mef} The forward middle to end angle for the SME mode
A_{smr} The reverse start to middle angle for the SME mode
A_{mer} The reverse middle to end angle for the SME mode
T_{fhf} The forward nominal first-half scan time
T_{shf} The forward nominal second-half scan time
T_{fhr} The reverse nominal first-half scan time
T_{shr} The reverse nominal second-half scan time
T_n The nominal scan time
T_{unit} The conversion factor from counts to time
$mprofile_{i,j}$ The corrected mirror polynomial for each scan

Compute the actual scan time—find true first-half time (T_{fh}), true second-half time (T_{sh}), and total scan time (t_s):

For a forward scan:

$T_{fh} = T_{fhf} - FHSERR_i * T_{unit}$ (The first-half scan time)
$T_{sh} = T_{shf} - SHSERR_i * T_{unit}$ (The second-half scan time)
$t_s = T_{fh} + T_{sh}$ (The total scan time)

For a reverse scan:

$$T_{fh} = T_{fhr} - FHSERR_i * T_{unit} \quad \text{(The first-half scan time)}$$
$$T_{sh} = T_{shr} - SHSERR_i * T_{unit} \quad \text{(The second-half scan time)}$$
$$t_s = T_{fh} + T_{sh} \quad \text{(The total scan time)}$$

Calculate the correction to the linear model needed to make the mid-scan angle equal to the observed value (zero) at time T_{fh} (see Figure 3-13).

For a forward scan: $A_f = -(A_{mef}T_{fhf} + A_{smf}T_{shf}) / t_s$

For a reverse scan: $A_r = -(A_{mer}T_{fhr} + A_{smr}T_{shr}) / t_s$

Figure 3-13. Scan Mirror Profile Deviations

Rescale the fifth-order polynomial coefficients to account for the difference between the actual and nominal scan time:

For a forward scan: $b_j = b_j * (T_n / t_s)^{\,j}$

where: $j = 0$ to 5

For a reverse scan: $d_j = d_j * (T_n / t_s)^j$

where: j = 0 to 5

Calculate the amplitude component of the correction quadratic (B_f) due to the polynomial location of the mirror to the linear location. This is done by evaluating the rescaled fifth-order polynomial at time T_{fh} ($f(b, T_{fh})$ in Figure 3-13).

For a forward scan:

$$B_f = b_0 + (b_1 * T_{fhf}) + (b_2 * T_{fhf}^2) + (b_3 * T_{fhf}^3) + (b_4 * T_{fhf}^4) + (b_5 * T_{fhf}^5)$$

For a reverse scan (a [–] sign is applied to the d coefficients):

$$B_r = d_0 + (d_1 * T_{fhr}) + (d_2 * T_{fhr}^2) + (d_3 * T_{fhr}^3) + (d_4 * T_{fhr}^4) + (d_5 * T_{fhr}^5)$$

The net correction to the mirror profile at the mid-scan time (T_{fh}) is (D_f and D_r):

For a forward scan: $D_f = A_f - B_f$

For a reverse scan: $D_r = A_r - B_r$

Using the constraints $a(0) = a(t_s) = 0.0$ and $a(t_{sh}) = D$, the coefficients of the quadratic are:

For a forward scan: $a_2 = -D_f / (T_{fhf} * T_{shf})$

For a reverse scan: $a_2 = -D_r / (T_{shr} * T_{fhr})$

For both scan directions: $a_1 = -t_s * a_2$
$a_0 = 0.0$

3.1.5.3.3.10   Integration of Roll Jitter into the Scan Mirror Model

The high-frequency roll attitude components (jitter) must be considered together with the scan mirror profile, since the scan mirror is not rigidly attached to the instrument structure in the along-scan direction, but instead is driven by a flex pivot. This acts as a mechanical buffer that does not transmit the high-frequency along-scan jitter directly to the mirror. Therefore, the roll jitter terms have two effects: the ETM+ instrument rotates relative to the orbital coordinate system, and the scan mirror rotates relative to the ETM+ instrument (in the opposite direction). Effectively, the scan mirror does not "see" the high-frequency roll jitter. The roll rotation of the ETM+ instrument (i.e., the focal planes) independent of the scan mirror causes the effective line of sight to be deflected in the opposite direction from that caused by a rigid rotation of the entire focal plane/scan mirror assembly (see Reference 35). This is depicted in Figure 3-14.

The scan mirror angle relative to the rest of the ETM+, and to the orbital coordinate system, is known at the three points where it is measured by the scan angle monitor (SAM): scan start, mid-scan, and scan end. The line of sight at these three times is based on the known mirror angles at beginning, middle, and end of scan, modified by the high-frequency roll at the corresponding times, so the combined mirror/roll jitter model must match these known angles at these three times. This is accomplished by modifying the scan mirror profile for each scan to include the scan start, mid-scan, and scan end high-frequency roll effects. This is implemented as a roll jitter correction to the scan mirror polynomial, $\hat{\theta}(t)$. The differential jitter relative to this profile is then $\theta_t - 2\hat{\theta}(t)$, where θ_t is the measured roll jitter and $\hat{\theta}(t)$ is doubled to account for the fact that a scan mirror rotation of $\hat{\theta}(t)$ leads to a line-of-sight deflection of $2\hat{\theta}(t)$.

As described above and depicted in
Figure 3-14, this differential jitter leads to a deflection of the line of sight in the opposite direction, or $-[\theta_t - 2\hat{\theta}(t)]$. So, the net deflection of the line of sight relative to the orbital coordinate system is $-\theta_t + 2\hat{\theta}(t)$. This deflection is implemented by inverting the sign on the roll jitter component (θ_t) and including the scan mirror profile adjustment $(\hat{\theta}(t))$ in the mirror model. All that remains is to determine what the scan mirror profile adjustment should be.

As noted above, the net line-of-sight deflection must account for the roll angles at scan start (θ_s), mid-scan (θ_m), and scan end (θ_e). If the mirror profile adjustment is constructed so that it is equal to the measured roll jitter at scan start, mid-scan, and scan end (i.e., it modifies the mirror profile to include roll jitter effects at those three points), then the net line-of-sight deflection at T=0, T=t_{fh}, and T=T_s is:

$$\theta_{net} = -\theta_t + 2\theta_t = \theta_t$$

where t = s, m, or e for scan start, mid-scan, and scan end.

This is the desired result, so the mirror correction term, $\hat{\theta}(t)$, must be constructed so that:

$$\hat{\theta}(0) = \theta_s$$
$$\hat{\theta}(t_{fh}) = \theta_m$$
$$\hat{\theta}(T_s) = \theta_e$$

a) Nominal Focal Plane/Scan Mirror Geometry

b) Rigid Rotation of Focal Plane and Scan Mirror

c) Rotation of Focal Plane Independent of Scan Mirror
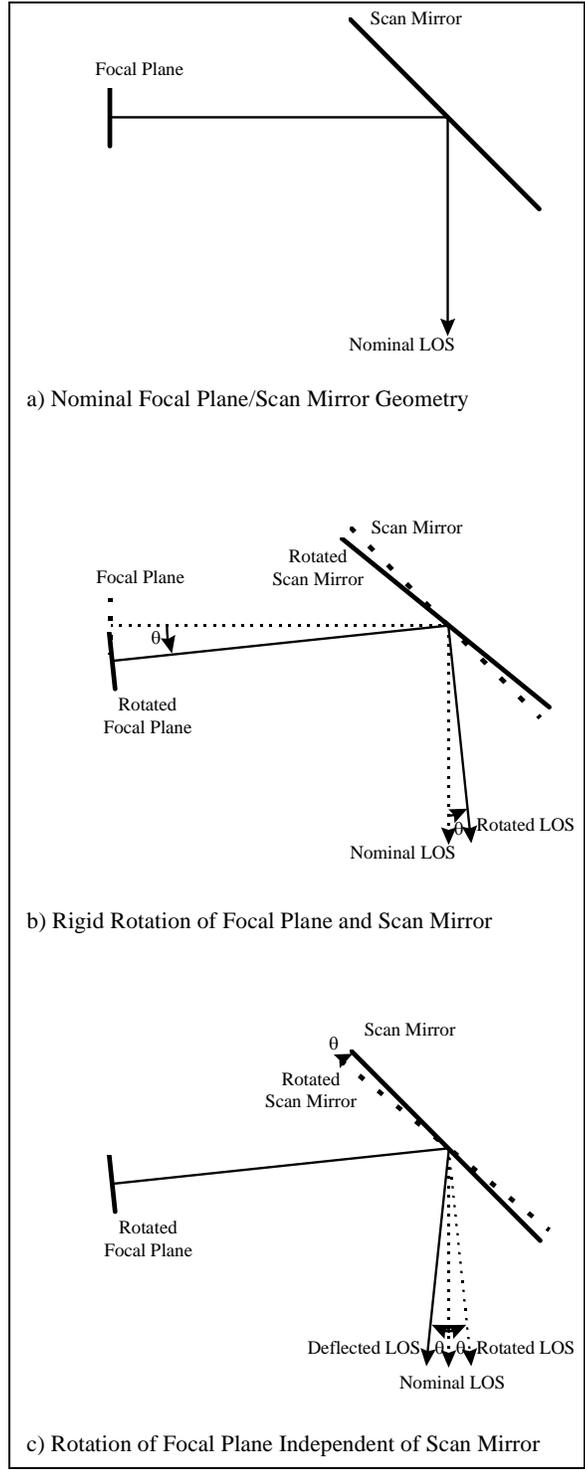
Figure 3-14. Effect of Roll Jitter on Line-of-Sight Direction

This can be accomplished using a correction quadratic similar to that used to perform the mid-scan correction. The coefficients of this polynomial are shown to be:

$$c_0 = \theta_s$$

$$c_1 = \frac{\theta_e - \theta_s}{T_s} - \frac{(\theta_e - \theta_m)t_{fh} + (\theta_s - \theta_m)t_{sh}}{t_{fh}t_{sh}}$$

$$c_2 = \frac{(\theta_e - \theta_m)t_{fh} + (\theta_s - \theta_m)t_{sh}}{T_s t_{fh} t_{sh}}$$

This makes the net mirror profile:

$$b'_0 = b_0 + a_0 + c_0$$

$$b'_1 = b_1 + a_1 + c_1$$

$$b'_2 = b_2 + a_2 + c_2$$

$$b'_3 = b_3$$

$$b'_4 = b_4$$

$$b'_5 = b_5$$

where the b_i are the nominal mirror profile coefficients normalized for actual scan time; the a_i are the mid-scan correction quadratic coefficients computed from the first-half and second-half scan errors; and the c_i are the roll jitter correction quadratic coefficients defined above. Inverting the sign on the measured roll jitter data and using this modified mirror profile creates the desired net line-of-sight deflection.

3.1.5.3.3.11   Smoothing Low-Frequency Attitude Information

In the IAS release 1 model, the quaternion and gyro drift observations from the PCD major frame immediately preceding the scene were used as a reference point. All subsequent attitude data for the scene were referenced from this point. Therefore, if a stellar observation occurred later in the scene, this change in the reference frame was not be reflected in the model. This change can be thought of as an update in the reference frame of the system. The update could produce a discontinuity in the attitude data stream. To incorporate all of the stellar observation information in the PCD quaternion samples, a Kalman filter is used as a post-processing filter of the attitude data stream.

The Kalman filter is implemented in a forward-predictor, backward-smoother configuration. Under this scenario, the filtering serves as a forward-recursive sweep, while the smoothing serves as a backward sweep. The backward-smoother processing propagates the stellar observations, represented by the quaternion data, backward in time to provide a best estimate of the spacecraft attitude given data from all times in the sub-interval. The smoothed data stream is a new attitude data stream representing the combined gyro, quaternion, and drift data. This smoothed data set represents the low-

frequency attitude information of the satellite. The data is then combined with the ADS information to produce the overall attitude of the sensor/satellite.

The rest of this section contains three parts. The first is a brief description of the filtering and smoothing equations. The reader is encouraged to pursue the references given later in the text for further details on the subject. The second portion involves calculating the equations that are specific to this application. The final portion explains the details of running the process with actual data.

The discrete Kalman filter assumes a random process modeled as follows:

$$[X]_{K+1} = [\phi]_K [X]_K + [q]_K$$

where:

$[X]_K$ = (n x 1) is the state vector at time t_K
$[\phi]_K$ = (n x n) is the state transition matrix relating X_K to X_{K+1}
$[q]_K$ = (n x 1) is the process noise, white, uncorrelated, with covariance Q

Measurements of the process are modeled as:

$$[Z]_K = [H]_K [X]_K + [v]_K$$

where:

$[Z]_K$ = (m x 1) is the measurement vector at time t_k
$[H]_K$ = (m x n) is the observation matrix relating the state vector at t_k to the measurement vector
$[v]_K$ = (m x 1) is the measurement error, white, uncorrelated, with covariance R

The first step involves using the Kalman filter as a forward prediction process. The Kalman filtering equations (all equations refer to the discrete domain) are as follows:

(Superscript *P* refers to prediction.)

Given an initial state vector and covariance matrix $[X]^P_0$ and $[P]^P_0$:

for k=1, …, N (number of data points)

Filter data:

$$[K]_K = [P]_K^p [H]_K^T \left([H]_K [P]_K^p [H]_K^T + [R]_K \right)^{-1}$$

$$[X]_K = [X]_K^p + [K]_K ([Z]_K - [H]_K [X]_K^p)$$

$$[P]_K = ([I] - [K]_K [H]_K)[P]_K^p$$

Predict:

$$[X]_{K+1}^p = [\phi]_K [X]_K$$

$$[P]_{K+1}^p = [\phi]_K [P]_K [\phi]_K^T + [Q]_K$$

where:

[I] is the identity matrix
[K] is the Kalman Gain matrix
[P] is the state error covariance matrix
[Q] = E[qq]
[R] = E[vv]

The second step involves using the filtered and predicted data to produce a smoothed data set. Let:

$[X]^P_K$ = estimate of [X] given measurements through t_{K-1}

$[P]^P_K$ = error covariance associated with estimate $[X]_K$

$[X]_K$ = filtered estimate through t_K

$[P]_K$ = state error covariance associated with filtered estimate at t_K

Using the definitions above, the new notation is:

$[X]_{K|K-1} = [X]^P_K$

$[P]_{K|K-1} = [P]^P_K$

$[X]_{K|K} = [X]_K$

$[P]_{K|K} = [P]_K$

Smooth data for K=N–1, N–2, .., 0:

$$[X]_{K|N} = [X]_{K|K} + [A]_K ([X]_{K+1|N} - [X]_{K+1|K})$$

$$[A]_K = [P]_{K|K} [\phi]^T_{K+1,K} [P]^{-1}_{K+1|K}$$

$$[P]_{K|N} = [P]_{K|K} + [A]_K ([P]_{K+1|N} - [P]_{K+1|K})[A]^T_K$$

For more information on the Kalman filter and smoothing data, see References 23 and 34.

The rest of this document focuses on the how the matrix equations specific to the attitude processing were obtained and some of the processing flow issues.

The system for the continuous time case is first described, then the corresponding discrete time case is solved for. The equations for the continuous case are:

$$\dot{X} = FX + w$$

$$Y = CX$$

where:

       X = the state vector
       F = the state matrix
       Y = the output response vector
       C = the matrix relating state to measurement
       w = the vector of white noise processes
       F, G, C may be time-varying matrices

The state matrix for the attitude processing contains three states: attitude, gyro rate, and gyro drift. The state transition matrix is:

$$[\dot{X}] = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} 0 \\ w_2 \\ w_3 \end{bmatrix}$$

where:

       X_1 = attitude
       X_2 = gyro rate
       X_3 = gyro drift
       w = process noise (white)

LS-IAS-01
Version 1.0

The above model states that the gyro rate is the time derivative of the attitude, and the time derivatives of the gyro rate and drift are independent white noise processes.

The output response vector has the following form:

$$[Y] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

where r = measurement noise (white)

The above observation matrix states that there are three kinds of measurements involved in the process:

1. Quaternion observations, which measure the attitude
2. Gyro/gyro rate observations, which measure the attitude rate biased by the gyro drift
3. Gyro drift observations, which measure the bias in the gyro measurements

The gyro measurements have no absolute origin; therefore, they only have meaning relative to each other. By differencing the gyro samples and dividing by the sampling rate, they can be converted to attitude rate measurements. The quaternions are then used as the reference. After the gyro samples are converted into rates, the drift can be directly applied, since it is measured in units of radians per unit of time. The measurements then used by the process, as measured by the satellite, are quaternions, gyro rate, and gyro drift.

The matrix [F] does not contain any time-varying elements and is small enough so that the direct method can be used to determine the discrete transition matrix.   The equation takes the following form:

$$[\phi]_k = \left[L^{-1} \left\{ [[I]s - [F]]^{-1} \right\} \right]_{t=dt}$$

where:

L^{-1} = inverse Laplace transform
[I] = identity matrix
$[]^{-1}$ = matrix inversion
dt = sampling interval

This results in the following:

$$L^{-1}\left\{\left[[I]s-[F]\right]^{-1}\right\} = L^{-1}\left\{\begin{bmatrix} s & -1 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix}^{-1}\right\} = L^{-1}\left\{\begin{bmatrix} \dfrac{1}{s} & \dfrac{1}{s^2} & 0 \\ 0 & \dfrac{1}{s} & 0 \\ 0 & 0 & \dfrac{1}{s} \end{bmatrix}\right\} = \begin{bmatrix} 1 & t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Substituting dt for t in the above matrix results in the discrete state transition matrix.

The discrete process noise matrix can also be solved directly. The discrete process noise is found from the following equation:

$$[Q]k = \int_0^{dt}\int_0^{dt} [\phi(dt,v)][G(u)]E[w(u)w^T(v)][G(v)]^T[\phi(dt,v)]^T \, du\,dv$$

Note that w_2 and w_3 are uncorrelated and that σ is the variance of the associated white noise process:

$$G(u)E[w(u)w^T(v)]G(v) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & u & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} uv\sigma_2^2 & u\sigma_2^2 & 0 \\ v\sigma_2^2 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{bmatrix}$$

Solving the double integral for the above matrix results in the discrete process noise matrix:

$$\int_0^{dt}\int_0^{dt}\begin{bmatrix} uv\sigma_2^2 & u\sigma_2^2 & 0 \\ v\sigma_2^2 & \sigma_2^2 & 0 \\ 0 & 0 & \sigma_3^2 \end{bmatrix} = \begin{bmatrix} \dfrac{dt^4\sigma}{4} & \dfrac{dt^3\sigma_2^2}{2} & 0 \\ \dfrac{dt^3\sigma_2^2}{2} & dt^2\sigma_2^2 & 0 \\ 0 & 0 & dt^2\sigma_3^2 \end{bmatrix}$$

LS-IAS-01
Version 1.0

The observation matrix relating the state vector to the output/measurement vector is the same for the continuous and discrete cases:

$$[H] = [C]$$

Under this system model, each axis can be processed independently. The attitude of the satellite is contained in the first state of the state vector. The gyro and quaternion data for Landsat 7 are not sampled at the same instance in time or at the same rate. This needs to be accounted for prior to and during processing. The data also have to be in a common coordinate system. The drift is measured in the attitude control system (ACS). The quaternions are given as Euler parameters and can be converted to the ACS (ATBD, 3.1.5.3.3, Satellite Attitude Processing). The gyro data are measured in the inertial measurement unit (IMU) axis and can be converted to the ACS (ATBD, 3.1.5.3.3, Processing Gyro Data). The system therefore works in the ACS. The output is also in the ACS. The time offset is accounted for by looking at the sampling times associated with each data source.

Gyro samples:

PCD time code + 0.064N seconds N = 0, 1, 2, …, 255

Attitude:

PCD
Major Frame
0   PCD time code – 8.192 seconds
1   PCD time code – 4.096 seconds
2   PCD time code + 0.0 seconds
3   PCD time code + 4.096 seconds

Drift:

PCD
Major Frame
0   PCD time code – 8.192 seconds

At the first PCD cycle, PCD major frame 0, the gyro data correspond to the quaternion value of the first PCD cycle, PCD major frame 2. This is the initial/start condition for the filtering. The gyro drift value is zero until the data stream reaches a point with valid drift data. The initial covariance error matrix, $[P]$, is set equal to the process noise matrix. This value may need to be changed after some test runs with different data sets. Due to the fact that the gyro, attitude, and drift data have different sampling rates, there are three different observation matrices: one when only the gyro is present, a second when both the gyro and quaternions are present, and a third when all three measurements are available.

When only the gyro is available, the observation matrix is:

$$[Y] = \begin{bmatrix} 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + [r]$$

When both the gyro and quaternions are available, the observation matrix is:

$$\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} + \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}$$

When the gyro, quaternion, and gyro drift measurements are available, the observation matrix is:

$$\begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} + \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

Although it states that "gyro measurements are available," it is actually the gyro rate with which the system is working at this point.

The state transition matrix is the same for all cases.

The final step is to determine the values associated with the measurement noise matrices and the standard deviations associated with the process noise matrix. The measurement error for the gyro corresponds to the increment associated with one gyro count, 0.061 arc-seconds. The measurement error for the gyro rate is found by dividing 0.061 arc-seconds by the sampling rate of the gyro, 0.064 seconds. The measurement error used for the quaternion corresponds to the error associated with the Celestial Sensor Assembly and is 4.2 arc-seconds (see Reference 35). The measurement error for the drift and the standard deviations for the process noise were determined by trial and error. The criteria for choosing values for trial, however, fell under the following premises:

- The drift values were considered much less reliable than either the gyro or the quaternion values.
- The measurement error for the quaternion was larger than the gyro (rate) measurement error, which allowed the quaternions to act as a very low frequency reference and the gyro (rate) to represent the higher frequency reference. This maintained the scan-to-scan consistency needed for processing. The attitude processing should not affect the scan-to-scan accuracy to any measurable degree.

- The measurements for the gyro (rate) were assumed to carry more weight than the model, while the opposite was assumed for the drift. The values were chosen such that more consideration was given to the model for the drift, while more consideration was given to the measurements for the gyro (rate).

Through trial and error, this led to a value of 50.0 arc-seconds for the drift measurement, a standard deviation of 0.1 for the process noise associated with the gyro rate, and a standard deviation for 0.00001 for the process noise associated with the drift. These values were determined from data associated with the Landsat 4 satellite; therefore, the values may need to change slightly to reflect Landsat 7. These values were used to process 100 consecutive major frames of Landsat 4 attitude data. Processing the quaternion, gyro, and drift produced a change of less than 1 meter in the difference between adjacent gyro samples. This provides proof that the effect on between-scan alignment is negligible.

After all the data are processed, filtered, and smoothed, the new attitude data can be used by the G and F filters, along with the ADS information, to get the overall satellite attitude. To be consistent with the previous way of processing, however, the output from the filtering and smoothing should have the first value kept as a frame of reference and subtracted from all other values.

3.1.5.3.3.12   Combining Low- and High-Frequency Attitude Information

Much of the premise for the design of the attitude data processing algorithm comes from Landsat Thematic Mapper Attitude Data Processing by G. J. Sehn and S. F. Miller (Reference 10). References 38, 39, and 40 are also good sources on the subject.

The following symbols are used throughout this section:

FFT	Fast Fourier Transform with a complex component of -1 on the exponent associated with the forward transform (1 for the reverse)
Hg	Frequency response of the gyro sensor
P	Frequency response of the ADS pre-filter
Ha	Frequency response of the ADS sensor
PHa	Frequency response of the ADS sensor combined with pre-filter
G(f)	Frequency response of the G filter
F(f)	Frequency response of the F filter
|function|	Magnitude response of function
j	The imaginary part of a complex number

The gyro has the following transfer function for Landsats 4 and 5:

$$Hg = ak * (1.0 + K * s) / (1.0 + tau * s) * [(s / wn)^2 + (2.0 * zeta * s) / wn + 1.0]$$

where ak, K, tau, wn, and zeta are measured constants and

$$s = 2j * PI * freq.$$

The ADS and pre-filter combined have the following transfer function for Landsats 4 and 5:

$$Hads = (A5 * s^5 + A4 * s^4 + A3 * s^3) /$$
$$(s^6 + B5 * s^5 + B4 * s^4 + B3 * s^3 + B2 * s^2 + B1*s + B0)$$

where A5, A4, A3, B5, B4, B3, B2, B1, and B0 are measured constants and

$$s = 2j * PI * freq.$$

The Landsat 4 and 5 satellites have two sensors to measure satellite attitude perturbations. One of the sensors measures low-end frequencies, while the other measures high-end frequencies. They are called the gyro and attitude displacement sensor (ADS), respectively. The gyro is sampled every 64 milliseconds, and the ADS is sampled every 2 milliseconds. The samples taken from these two systems cannot simply be added together to get the overall satellite attitude perturbations due to a discontinuity between the transition regions of the responses of the two systems and the sensor phase shift properties.

A plot of the frequency responses of the two systems is shown in Figure 3-15.
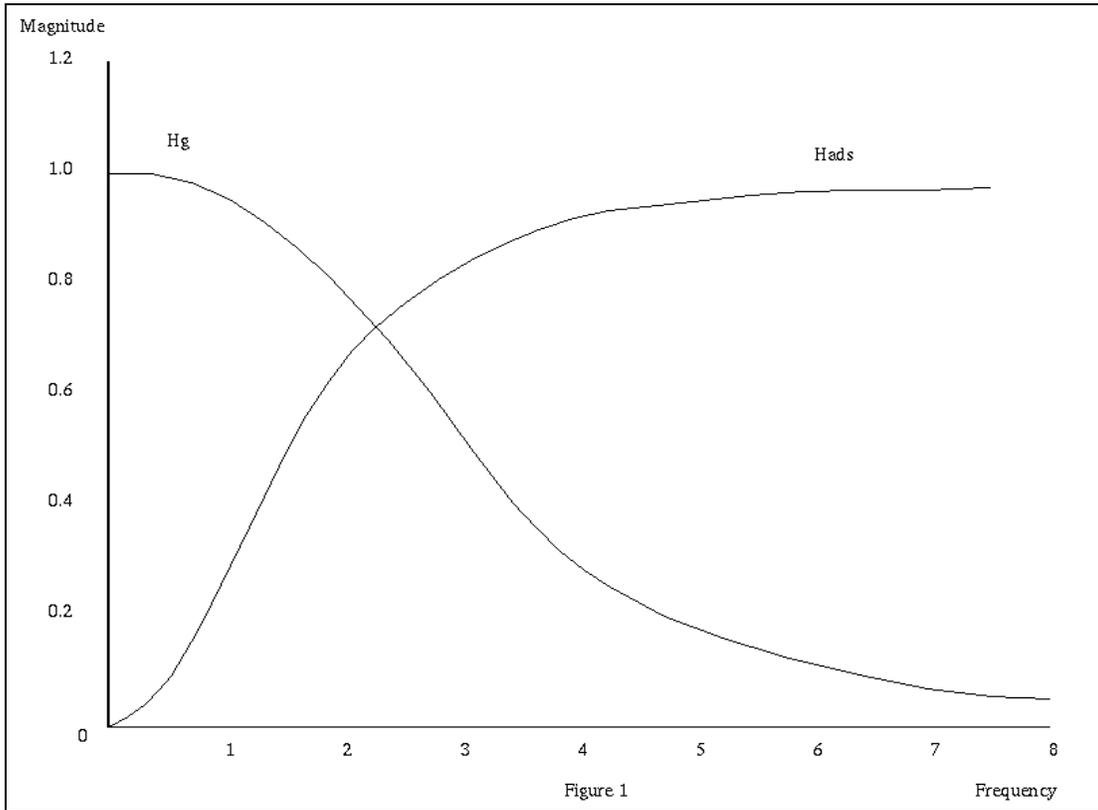
Figure 3-15. Magnitude Response of Gyro and ADS

The magnitude of the sum |Hg + PHa| is shown in Figure 3-16.
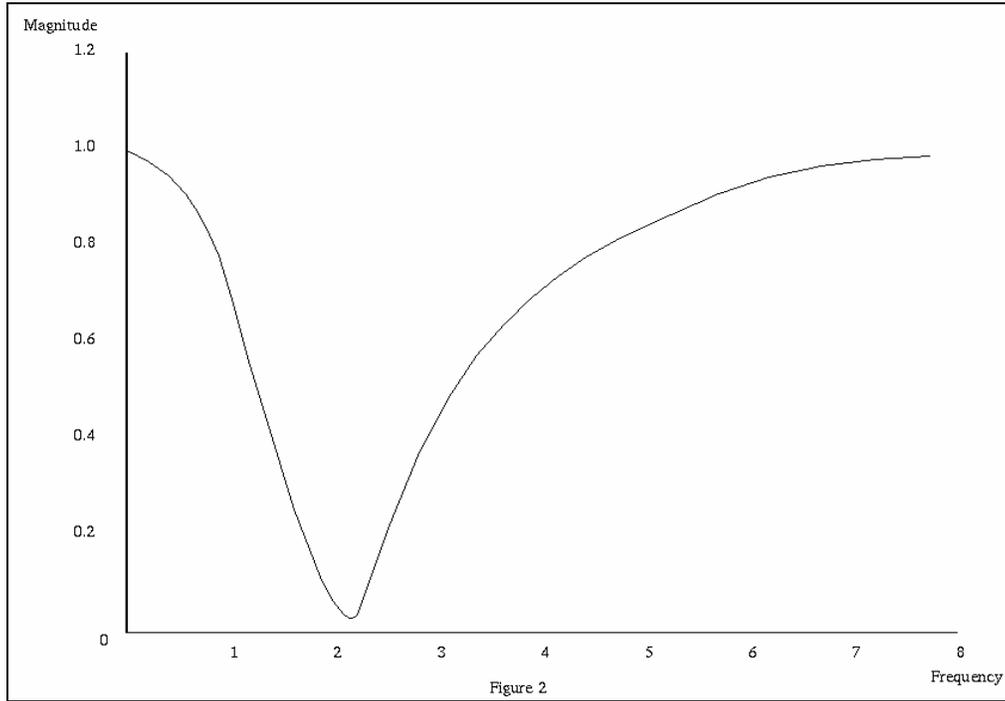
Figure 3-16. Magnitude Response of Gyro Plus ADS

The magnitude of the sum |Hg| + |PHa| is shown in Figure 3-17.
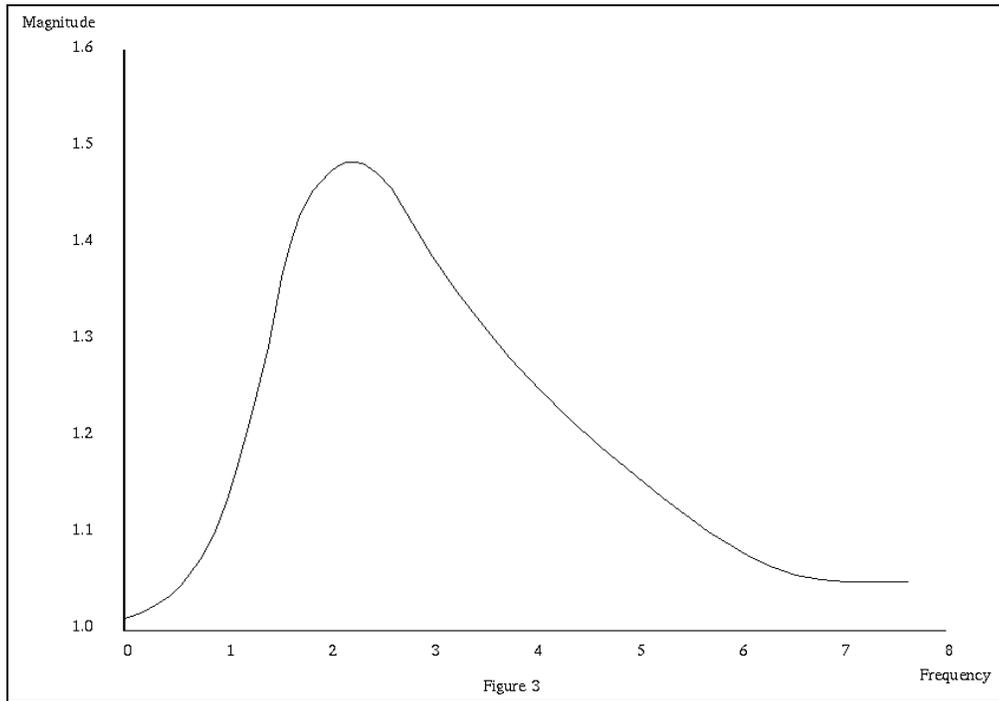
Figure 3-17. Magnitude Response of Gyro + Magnitude Response of ADS

These figures help to explain the discontinuity between the system responses.

To combine these two signals, along with the quaternions and gyro drift values, a network is used that unites the traditional Sehn and Miller approach with the Kalman filtering that was explained in 3.1.5.3.3.11, Smoothing Low-Frequency Attitude Information.
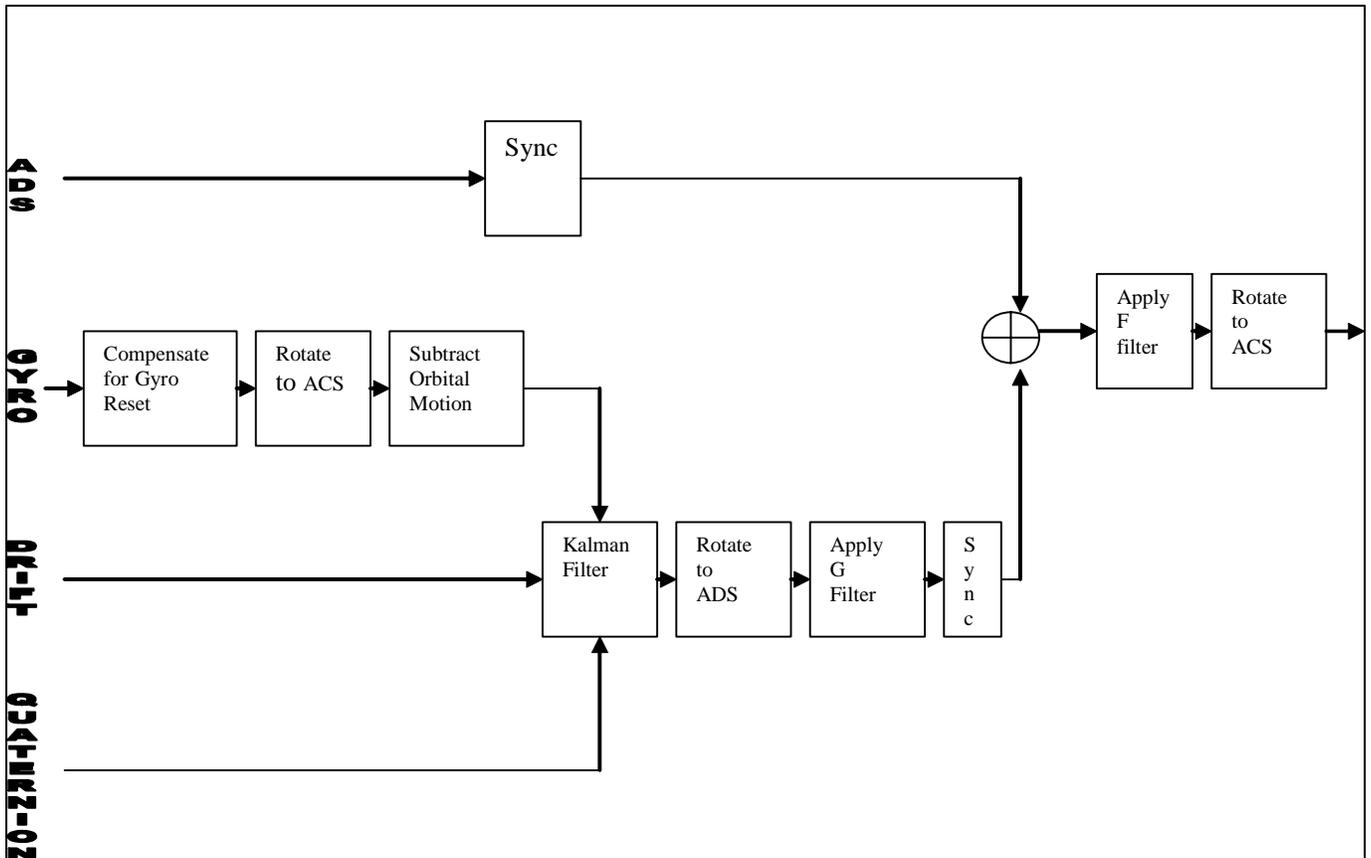
Figure 3-18. Attitude Processing Network

This algorithm and the steps needed to implement a workable system are listed below. A diagram of the algorithm steps is shown in Figure 3-18.

1. Account for the register resets.

2. For each axis, subtract the first gyro value from the rest.

3. Rotate the gyro data to the ACS.

4. Remove the orbital drift component. (Details of steps 1–4 can be found in 3.1.5.3.3, Process Gyro Data.)

5. Smooth the gyro, gyro drift, and quaternion. (Details of step 5 can be found in 3.1.5.3.3.11, Smoothing Low-Frequency Attitude Information.)

6. Rotate the data from step 5 to the ADS system. (Details of step 6 can be found in 3.1.5.3.3, Process Gyro Data.)

7. Filter the data from step 6. This is done with a filter traditionally called the pre-summing G filter, or simply the G filter. This filter helps to eliminate any frequencies above a predetermined cutoff, while also providing an adequate transition for combining the gyro and ADS. The gyro data are not pre-filtered as the ADS data are. The G filter serves this purpose. The requirements for the G filter are to provide unity gain at low frequencies and to allow the ADS system to provide gain at high frequencies.

The G filter takes on the following values in the frequency domain:

Define: F1 = 2.0
          F2 = 7.8125
          freq = current frequency

F1 is the chosen frequency where Hads reaches −1.0 dB, while F2 is the chosen Nyquist frequency associated with sampling of the gyro data.

If freq is less than F1, then:

Vk = 1

If freq is greater than F1, but less than F2, then:

a = (F2 – freq) * (F2 – freq);
b = (freq – F1) * (freq – F1);
Vk = (a * |Hpdf| + b * |PHa|) / (a + b)

(This is just an interpolation over the cross-over region of the two systems.)

If freq is greater than F2, then:

Vk = PHa
G(freq) = 0.0

Solve for the G filter by looking at the response of the attitude processing system before the F filter is applied (see Reference 10).

|GHg + PHa| = Vk

Write the equation as:

(GHg + PHa) * conjugate{(GHg + PHa)} = Vk * Vk

Multiplying this out and noting that G is real so that G = conjugate{G}, the equation becomes:

$$G^2|Hg|^2 + 2GRe\{ Hg \text{ conjugate}\{PHa\} \} + |PHa|^2 - Vk^2 = 0$$

Use the quadratic equation to solve for G. Choose a solution for the minimum magnitude. This solution produces the following equations for the G filter:

Hc = conjugate{PHa}
Gpa = Hg * Hc

a = |Hg| * |Hg|
b = 2.0 * Gpa.re
c = |PHa| * |PHa| – Vk * Vk

Re{G} = (–b – (b * b – 4.0 * a * c)$^{1/2}$) / (2.0 * a)
Im{G} = 0.0

Due to the fact that the gyro data are not pre-filtered, aliasing may be associated with the measured signal. Apply the G filter in the spatial domain to try to remove some of the aliasing at frequencies above 2 Hz and to provide a smooth transition between the gyro and ADS transition region. Use the above definition of the G filter in the frequency domain to achieve a spatial domain filter:

Define: Filter size = N
          Padded filter size = M

This assumes that values for the filter are stored in an array. If speaking in the frequency domain, then the buffer location 0 in the array is the DC term, and location N/2 is the Nyquist frequency.

   a. Determine an appropriate size for the G filter. Sehn and Miller state that a G filter of length 31 guarantees less than 1% filter approximation error. Since the G filter is determined in the frequency domain, it makes sense to choose a value as close to a power of 2 as possible, since this is what the FFT routine needs. If a routine is used that does not need a power of 2, then this should not be an issue. The IAS uses a value of 63.

   b. Find the nearest power of 2 larger than the associated filter size N. Zero pad the data to this length (this makes the data size M).

   c. Step through the number of chosen values (filter length padded to a power of 2), calculating the frequency response of the filter; the equations are provided above. This is calculated at frequency increments that are dependent on the gyro sampling rate and the number of samples from step 2. In practice, only one half plus one of the filter size needs to be calculated due to step 7d.

d. Ensure that the G filter is real in the spatial domain. The Fourier Transform must be the symmetrical complex conjugate with respect to the origin or, in this case, the Nyquist frequency. Reference 6 provides an explanation of the Fast Fourier Transform (FFT) and some of its properties, which further explains this process.

e. Calculate the inverse FFT of the results from step 7d.

f. Adjust the data so that they are of the original desired filter length N. They should be symmetrical about the center of the filter.

g. Apply the filter coefficients calculated in step 7f to the gyro data. When the filter coefficients are applied to the ends of the gyro data, the data are mirrored about their end points.

8. Synchronize (up sample) the data from step 7, so that they agree with the 2-millisecond sampling rate of the ADS data. The prototype uses a series of filters that allows better control over the response of the overall interpolation (final sampling rate). A further understanding and explanation of this process can be found in Reference 33. The filtering coefficients used in the IAS are as follows:

Number	Scale	h0	h1	h2	h3	h4
Filter 1	1.0	1.0	0.0	0.0	0.0	0.0
Filter 2	2.0	1.0	0.0	0.0	0.0	0.0
Filter 3	16.0	9.0	−1.0	0.0	0.0	0.0
Filter 4	32.0	19.0	−3.0	0.0	0.0	0.0
Filter 5	256.0	150.0	−25.0	3.0	0.0	0.0
Filter 6	346.0	208.0	−44.0	9.0	0.0	0.0
Filter 7	512.0	302.0	−53.0	7.0	0.0	0.0
Filter 8	802.0	490.0	−116.0	33.0	−6.0	0.0
Filter 9	8,192.0	5,042.0	−1,277.0	429.0	−116.0	18.0

These numbers are reflected around the point of interest (the point to be interpolated). The filter coefficients are multiplied by the corresponding pixel they are associated with, and the sum is then divided by the scale factor. This gives an interpolated point. For example, if interpolating a point between n and n+1 with filter 4, pix(n) defines gray value at point n.

sum =   h1 * pix(n−1) +
         h0 * pix(n) +
         h0 * pix(n+1) +
         h1 * pix(n+2)

new pix = sum / scale[4]

The sampling start time, and the time between samples for the smoothed low frequency data, is:

x, y, x axis   PCD + 64N N = 0, …, 255

while the ADS is sampled at increments of:

x axis   PCD + 2N + 3/8 msec
y axis   PCD + 2N + 7/8 msec
z axis   PCD + 2N + 11/8 msec

where N = 0, …, 8,191

To get the smoothed low frequency data to have a time difference of 0.002 seconds between samples, the data are interpolated five times with the coefficients listed above. To account for the offsets in the ADS data, interpolation with the above coefficients is performed four times. This allows a time difference between samples to be chosen such that every sixteenth sample falls on an even increment of 0.002N seconds from a PCD cycle. Since there are no valid data at N=0, the first value for each axis of the time-synchronized ADS is zero. The offset is accounted for by stepping ahead by a set number of data points.   The first valid synchronized ADS occurs at 0.002 seconds.

Interpolating four times gives the ADS a time difference of 0.000125 seconds.

x-axis (0.002 – 0.000375) / 0.000125 = 13 samples
y-axis (0.002 – 0.000875) / 0.000125 = 9 samples
x-axis (0.002 – 0.001375) / 0.000125 = 5 samples

Every sixteenth sample is then chosen after taking into account the offset. This returns the time difference between samples to 0.002 seconds.

9. Sum the synchronized data from step 8.

10. Apply the F filter to the data obtained in step 9. Since the ADS data are pre-filtered, and due to the fact that the gyro data have been G filtered, the F filter can be applied in the frequency domain (aliased frequencies have been suppressed). Both Sehn and Miller and the Earth Observation Satellite Company (EOSAT) determine a spatial domain filter for the F filter in much the same way that the G filter was determined. This approach applies the F filter in the spatial domain and eliminates the need for the data in step 9 to be a power of 2.

The F filter takes on the following values in the frequency domain:

F(freq) = 1.0 / [PHa + G(freq)Hg]

Use the following steps to apply the F filter.

a. Calculate the FFT of the data from step 9. If the FFT is used, then the number of data elements, when summed, must be a power of 2.

b. Step through the data from step 10a, calculating the response of the F filter given the definition above. It is calculated at frequency increments that are dependent on the ADS sampling rate and the number of samples from step 10a.

c. Reflect the complex conjugate of the data from step 10b around the Nyquist data point (this is analogous to what was done for the G filter). This assures that the inverse FFT of this data set is real.

d. Multiply the data from steps 10a and 10c.

e. Calculate the inverse Fourier Transform of the data.

f. These data are the roll, pitch, and yaw due to attitude perturbations as measured from the gyro, gyro drift, quaternions, and ADS of the satellite.

11. The combined data are rotated to the ACS.

(Details of steps 6 and 11 can be found in 3.1.5.3.3.7, Convert ADS to a Common Reference System.)

3.1.5.3.3.13   Attitude Processing Sign Conventions

The PCD quaternions are used to construct an orientation matrix that rotates the ACS axes to ECI, as described on page 39 of the ATBD. This is premultiplied by a matrix that rotates ECI-to-orbital coordinates, constructed from the ephemeris data, to create an ACS-to-orbital rotation matrix (body-to-orbit). This matrix is mapped to the roll, pitch, and yaw perturbation matrix:

$$
\begin{bmatrix}
\cos(p)\cos(y) & \sin(r)\sin(p)\cos(y)+\cos(r)\sin(y) & \sin(r)\sin(y)-\cos(r)\sin(p)\cos(y) \\
-\cos(p)\sin(y) & \cos(r)\cos(y)-\sin(r)\sin(p)\sin(y) & \cos(r)\sin(p)\sin(y)+\sin(r)\cos(y) \\
\sin(p) & -\sin(r)\cos(p) & \cos(r)\cos(p)
\end{bmatrix}
$$

From the last row and first column of this matrix:

$$\text{roll} = \tan^{-1}(-m_{32}/m_{33}) = -\tan^{-1}(m_{32}/m_{33})$$

$$\text{pitch} = \sin^{-1}(m_{31})$$

$$\text{yaw} = \tan^{-1}(-m_{21}/m_{11}) = -\tan^{-1}(m_{21}/m_{11})$$

These are the body-to-orbit rotation angles.

Orbital motion is removed from the gyro data as follows. The ephemeris data are used to construct the ECI-to-orbital rotation matrix at the first PCD major frame time (ECI2ORB$_0$). At each subsequent gyro point, the ephemeris is interpolated and used to construct the instantaneous ECI-to-orbital rotation matrix, which is inverted to form the instantaneous orbital-to-ECI rotation matrix (ORB$_t$2ECI). These arrays are multiplied (ECI2ORB$_0$ ORB$_t$2ECI) to form the rotation from the instantaneous orbital coordinate system to the orbital coordinate system at the beginning of the PCD interval (ORB$_t$2ORB$_0$).

The same logic used above for the quaternions is used to extract roll, pitch, and yaw values from this composite matrix, yielding body-to-orbit rotation angles where the "body" coordinate system is taken to be the ideal ACS, which is perfectly aligned with the instantaneous orbital coordinate system relative to the reference orbital coordinate system at PCD start. These "ideal" angles are subtracted from the gyro samples so that the gyro data reflect deviations from the ideal orbital coordinate system or body-to-orbit rotation angles. These gyro body-to-orbit angles are combined with the body-to-orbit angles derived from the quaternions in the Kalman filtering procedure.

After the quaternion and gyro data are combined, the first filtered attitude sample value is subtracted from all of the samples and saved as a zero-frequency (DC) reference. This DC reference is sign inverted to form an orbit-to-body rotation reference. The residual gyro samples are combined with the ADS data and, in the course of this filtering process, are sign inverted so that the blended high-frequency attitude data represent orbit-to-body perturbations. The orbit-to-body DC reference and the orbit-to-body high-frequency perturbations are added to form the net orbit-to-body rotation, which is used in the satellite model.

3.1.5.3.4   Line-of-Sight Generation and Projection

The algorithms that follow are used by the geometric correction grid algorithm. The inputs to this algorithm are line, sample, and band. This algorithm also uses the information that was compiled and converted into a usable form in the create model algorithm. The result of this algorithm is the geodetic latitude and longitude for a point on the Earth, or a height (h) above the surface, which corresponds to the entered line, sample, and band of the input image.

3.1.5.3.4.1   Calculate Sample Time

The scan line number is found by subtracting 1 from the line number, dividing by the number of lines per scan for the given band, and then truncating.

The detector number is found by taking the decimal value remaining from the scan line number, before truncating, and multiplying by the number of lines per scan.

N = number of lines per scan

$$S = remainder \left[\frac{LineNumber - 1}{N} \right]$$

detector = N – floor($S*N$)

The time into the current scan is found by using the following equations:

For forward scans:

time_in_scan = (sample – 1.0 + bandoffset) * DWELLTIME

For reverse scans

time_in_scan = (scan[*scan_line].counted_line_length – sample + bandoffset) *
                    DWELLTIME

For Landsat 7, the bands are aligned using left-justified padding, so the padding must be removed from the calculation as a function of band. The L0R scan line offset (SLO) data are used to accomplish this.

The time from start of image is found using the MSCD start of scan value for the current scan minus the MSCD start of scan value for the first scan plus the time in scan.

3.1.5.3.4.2   Calculate Viewing Angles

Find the across-scan angle of a detector due to its location within the band (this example is for the multi-spectral bands):

detector_angle = (LINES_SCAN / 2.0 – detector + 0.5) * IFOVMS

Find the along-scan fifth-order polynomial correction. For forward scans, use the time-in-scan value to determine the correction; for reverse scans, use the total scan time minus the time-in-scan value to determine the correction value. Do the same for the across-scan fifth-order correction.

The polynomial coefficients and scan start and stop angles are provided as measures of mirror rotation. Since the instrument optical axis is reflected off the scan mirror, the effect on the line of sight in object space is twice the mirror angle. The scan mirror model and the focal plane geometry must be combined to calculate object space along-scan angles in the sensor coordinate system. The IAS sign convention is defined so that positive scan angles are in the direction of the sensor Y axis (as described in 3.1.2). The along-scan angle for band "BAND" is calculated as a function of time "TIME" as follows:

For forward scans:

$s_{2m} = A_{fsm}$ = start to mid-scan angle
$s_{2e} = A_{fsm} + A_{fme}$ = start to end angle
along_angle = 2.0 * ($s_{2m} - s_{2e}$ * TIME/T_s + f(b, TIME))
                        + bandoff_along[BAND] – odd_det_off[BAND]

where f(b, TIME) is the corrected fifth-order polynomial model.

For reverse scans:

$s_{2m} = A_{rsm}$
$s_{2e} = A_{rsm} + A_{rme}$
along_angle = 2.0 * ($-s_{2m} + s_{2e}$ * TIME/T_s + r(b, T_s – TIME))
                        + bandoff_along[BAND] – odd_det_off[BAND]

where r(b, TIME) is the corrected fifth-order polynomial model.

The across-scan angle is found using a linear interpolation of the SLC mirror's position, which is a function of time in scan. Also accounted for is the scan mirror's across-scan fifth-order correction, the detector angle, and the across-scan focal-plane offset.

Thus, the model for the SLC angle as a function of time in scan is:

slc_angle = slc_start[SCANDIR] – slc_max[SCANDIR] * TIME/Ts

where:

slc_angle = across-scan pointing angle (positive in the direction of flight)
slc_start = SLC angle at the beginning of scan (indexed by scan direction)
SCANDIR = index for scan direction (forward or reverse)
slc_max = total SLC angle (indexed by scan direction)
TIME = time in scan
Ts = total scan time

The same equation applies for forward and reverse scans, with the appropriate selection of the SLC start and SLC total angles. The sign convention and direction of SLC motion

is the same in either case: positive angles are in the direction of flight (toward the sensor X axis) with the SLC rotating from a positive to a negative angle in each scan.

The SLC angle must be combined with the across-scan detector angle and the scan mirror across-scan correction polynomial to compute the total across-scan look angle for detector N in band "BAND":

For forward scans:

$$cross\_angle = slc\_start[FWD] - slc\_max[FWD]*TIME/Ts$$
$$+ 2.0 * f(b, TIME) + bandoff\_cross[BAND]$$
$$+ ((ndets[BAND]+1)/2 - N)*IFOV[BAND]$$

where f(b, TIME) is the scan mirror across-scan correction polynomial evaluated at the time of interest and all other components are as previously defined.

For reverse scans:

$$cross\_angle = slc\_start[REV] - slc\_max[REV]*TIME/Ts$$
$$+ 2.0 * r(b, TIME) + bandoff\_cross[BAND]$$
$$+ ((ndets[BAND]+1)/2 - N)*IFOV[BAND]$$

where r(b, TIME) is the scan mirror across-scan correction polynomial evaluated at the time of interest. The only difference between forward and reverse scans is the selection of the appropriate SLC parameters, as mentioned above, and the application of the scan mirror across-scan correction polynomial.

3.1.5.3.4.3   Calculate LOS Vector

This algorithm uses the time in which a pixel was imaged to find the attitude, position, and velocity of the spacecraft. The along- and across-scan angles are used to construct an ideal line-of-sight (LOS) vector in sensor space. This LOS vector is transformed to the spacecraft navigation reference base and is then transformed using the attitude information to the Orbital Coordinate System (OCS). The OCS LOS vector is then transformed to the Earth-centered coordinate system.

1. To find the attitude, this algorithm uses the time from first ADS value to the time of pixel imaging to look up the ADS and gyro combined attitude in the combined attitude table. This table was constructed in the Process Attitude module. Added to the roll, pitch, and yaw values from the table are the spacecraft's roll, pitch, and yaw values (the DC reference values).

2. The move satellite algorithm uses the time from the reference ephemeris point to the time of pixel imaging to interpolate a new, Earth-Centered, Earth-Fixed (ECEF) satellite position and velocity. The polynomials for this interpolation are defined in the Create Model module in the calculate satellite state vector algorithm.

3. The find LOS algorithm uses the along- and across-scan angles to construct a LOS vector in the sensor's coordinate system.

$los_x = \sin(across\_angle) * \cos(along\_angle)$

$los_y = \sin(along\_angle)$

$los_z = \cos(across\_angle) * \cos(along\_angle)$

4. The attitude algorithm transforms the LOS vector to the spacecraft's ACS coordinate system. This vector is then transformed using the spacecraft's attitude to the OCS, and then the vector is transformed to the ECEF coordinate system.

   a. Transform the LOS from the sensor coordinate system to the ACS coordinate system:

$$\begin{bmatrix} ACS \\ LOS \end{bmatrix} = \begin{bmatrix} 3x3 \\ Transformation \\ Matrix \end{bmatrix} \begin{bmatrix} Sensor \\ LOS \end{bmatrix}$$

   The 3-x-3 transformation matrix from the sensor to the ACS is determined pre-launch. As part of the IAS calibration activities, this matrix is periodically checked and, if required, updated. The members of this matrix are contained in the Calibration Parameter File.

   b. Transform the ACS LOS to the OCS using the attitude information:

$$\begin{bmatrix} Orbital \\ LOS \end{bmatrix} = [YAW]^{-1}[PITCH]^{-1}[ROLL]^{-1}\begin{bmatrix} ACS \\ LOS \end{bmatrix}$$

   c. Transform the Orbital LOS to the ECEF coordinate system:

$$\begin{bmatrix} ECEF \\ LOS \end{bmatrix} = \begin{bmatrix} 3x3 \\ Transformation \\ Matrix \end{bmatrix} \begin{bmatrix} Orbital \\ LOS \end{bmatrix}$$

   The 3-x-3 OCS-to-ECEF transformation matrix is found using the unit vectors of the axes for the OCS determined in the ECEF coordinate system. The Z axis of the OCS is defined as the negative of the spacecraft's position vector. The Y axis of the OCS is the negative of the spacecraft's angular momentum vector or (V x R). The X axis is the cross product of the Y axis with the Z axis (Y x Z).

3.1.5.3.4.4   Intersect LOS Vector with Earth

This algorithm finds the geocentric location on the ellipsoid of the intersection of the LOS vector to the ellipsoid or at a constant elevation for the entire scene above the ellipsoid.

The satellite's position vector, the LOS vector, and an elevation are the inputs to this algorithm and the geocentric location of the intersection is the output in both the Cartesian and spherical coordinates.

This algorithm uses an iterative method to locate the intersection. The satellite's geocentric latitude is used as the initial guess in the process. This latitude is used to find the geocentric radius of the ellipsoid plus the elevation at this latitude. Using the angle between the satellite's position vector and the LOS vector, the radius of the Earth, and the plane trigonometry's law of cosines, the magnitude of the LOS vector is found. The intersection is the addition of the satellite's position vector and the scaled LOS vector. However, this location is based on a slightly incorrect latitude. Thus, the calculated intersection's geocentric latitude is used as the second guess. This process is continued until the LOS magnitude difference between iterations is small (< 0.01 meter).

Another approach is to use the reduced latitude in the calculation. This allows for a direct solution for a point on the surface of the ellipsoid. The solution is as follows:

> Define R_s as the position vector of the satellite, LOS as the line-of-sight vector, and R_t as the position vector of the target in question. The position vector of the target using the reduced latitude psi, longitude Lt, and the semi-major *a* and semi-minor *b* axes is:

> $$R_t = X_t i + Y_t j + Z_t k$$

> $X_t = a * \cos (psi) * \cos(Lt)$
> $Y_t = a * \cos (psi) * \sin(Lt)$
> $Z_t = b * \sin (psi)$

> The solution requires the following equation to be true:

> $$R_t = R_s + m * LOS$$

> where m is the magnitude of the final LOS vector. The current LOS vector has a magnitude of 1.

As separate components:

$$X_t = a * \cos(psi) * \cos(Lt) = X_s + m * X_{los}$$
$$Y_t = a * \cos(psi) * \sin(Lt) = Y_s + m * Y_{los}$$
$$Z_t = b * \sin(psi) = Z_s + m * Z_{los}$$

A vector R_t' with a magnitude of 1 can be constructed by dividing the X_t and Y_t components by *a* and dividing the Z_t component by *b*:

$$R_t' = X_t'i + Y_t'j + Z_t'k$$

$$X_t' = X_t / a = \cos(psi) * \cos(Lt) = X_s / a + m * X_{los} / a$$
$$Y_t' = Y_t / a = \cos(psi) * \sin(Lt) = Y_s / a + m * Y_{los} / a$$
$$Z_t' = Z_t / b = \sin(psi) = Z_s / b + m * Z_{los} / b$$

or:

$$R_t' = R_s' + m * LOS'$$

Using the law of cosines for a plane:

$$|R_t'|^2 = m^2 * |LOS'|^2 + |R_s'|^2 - 2 * m * |LOS'| * |R_s'| * \cos(w) = 1$$

where w is the angle between the satellite's position vector and the line-of-sight vector. Since R_t' is defined to have a magnitude of 1, the only unknown left in the equation is *m*.

Re-arranging the equation to use the quadratic formula to solve for *m* results in:

$$0 = m^2 * |LOS'|^2 - 2 * m * |LOS'| * |R_s'| + |R_s'|^2 - 1$$

and thus, the solution for R_t is:

$$R_t = R_s + m * LOS$$

The geocentric location is then converted to a geodetic location. The theory behind this routine is found in Reference 20, page 398.

3.1.5.3.4.5   Speed of Light Correction

Due to the non-infinite velocity of light, the velocity of a point on the surface of the Earth and the velocity of the satellite may cause pixel location errors if not accounted for. The speed of light correction due to the Earth's rotation is sub-meter and is currently not accounted for. The speed of light correction due to the satellite's velocity is almost a constant for Landsat 7. The magnitude is about 15 meters in the along-track direction and is applied in the model.

3.1.5.3.5 Correction Grid Generation

The ETM+ Geometric Correction Grid defines the piece-wise (gridded) geometric transformation of imagery in satellite perspective to corrected imagery in a user-specified output projection of user-specified geographic extent. The grid is used by the geometric transformation function (the resampler) to rectify the image data to the corrected image output space.

This algorithm assumes that the Landsat satellite model has been initialized using the input image's associated PCD and MSCD files. It also assumes that the output projection system is defined and read from a projection definition file (see Reference 19). This algorithm also requires the frame (geographic extent) of the resulting output space image and the dimension of each pixel in the output coordinate space. The PCD and MSCD files are used by the Landsat satellite model to define the input image's (0R or 1R) geometric space.

Refer to the LAS Geometric Manipulation Overview Document (Reference 18) for more information on specific projection parameters for a given projection and for information on the Projection Transformation Package, a coordinate transformation subroutine library based on the U.S. Geological Survey's General Cartographic Transformation Package (GCTP).

3.1.5.3.5.1 Defining the Frame

The first step in the gridding process is to determine the geographic extent of the output image to be generated by the resampler. The geographic extent of the output image space is referred to as the output space "frame" and is specified in output image projection coordinates. Five different methods are used to determine the output frame.

1. Method 1

   The user defines the upper-left and lower-right corner coordinates of the area of interest in geographic (latitude/longitude) coordinates. These coordinates are then projected to the output projection coordinate system using the Projection Transformation Package. This usually results in a non-rectangular area, so a minimum bounding rectangle is found (in terms of minimum and maximum X and Y projection coordinates) in the resulting output space. This minimum bounding rectangle defines the output space frame. The output image pixel size is then applied to the projection space to determine the number of lines and samples in the output space.

2. Method 2

   The user defines a rectangular area in the output space by specifying upper-left and lower-right output space projection coordinates. The output image pixel size is then applied to the projection space to determine the number of lines and samples in the output space.

3. Method 3

   The user defines an upper-left output space projection coordinate, the output image pixel size, and the number of lines and samples in the output image. The coordinate pair must be the minimum X and maximum Y projection coordinates. The maximum X and minimum Y coordinates (the other end of the bounding rectangle) are then calculated.

4. Method 4

   The user allows the framing software to project the four corners of the input L0R data set to the Earth ellipsoid. This produces the minimum bounding rectangle that contains all of the input image data.

5. Method 5

   The user specifies a path-oriented Landsat product in either the Space Oblique Mercator (SOM) or Universal Transverse Mercator (UTM) projection. In this case, the framing coordinates are not user-specified. The frame is a preset number of lines and samples based on the Landsat Worldwide Reference System (WRS) scene size and the maximum rotation needed to create a path-oriented product. For a pixel size of 30 meters, a rectified image in the SOM or UTM projection is approximately 6,440 lines by 6,850 samples in size.

For diagrams and more information on the framing process, refer to the LAS Geometric Manipulation Package Overview Document (Reference 18).

3.1.5.3.5.2   Path-Oriented Framing and Rotation

A path-oriented projection is basically a rotation from a typical "map-north" up projection that better represents the near polar orbit of the Landsat satellite (nominal inclination angle of 98.2 degrees). The path-oriented projection also does a better job of maintaining the scan line to the output space line.

The first step in generating a path-oriented projection is to calculate the center location in the output space about which the frame is to be rotated. This is done by computing the nominal WRS scene center for the current path and row. The frame is centered in the output projection space at this projection coordinate and is rotated about this point.

The procedure for computing the WRS scene center and the associated rotation angle is described below

Given:

Parameter	Value	Source
Earth Parameters		
Semi-major Axis	6378137.0 m	CPF: Earth_Constants:Semi_Major_Axis
Semi-minor Axis	6356752.314 m	CPF: Earth_Constants:Semi_Minor_Axis
L7 Orbit Parameters		
Inclination	98.2 degrees	CPF: Orbit_Parameters:Inclination_Angle
L7 WRS Parameters		
Number of Paths	233	CPF: Orbit_Parameters:WRS_Cycle_Orbits
Number of Rows	248	CPF: Orbit_Parameters:Scenes_Per_Orbit
WRS Repeat Cycle	16 days	CPF: Orbit_Parameters:WRS_Cycle_Days
Desc Node Row	60	CPF: Orbit_Parameters:Descending_Node_Row
Long of 001/060	−64.6 degrees	CPF: Orbit_Parameters:Long_Path1_Row60
Input Parameters		
Path	Integer 1−233	User
Row	Float 0 < R < 249	User

Find:

WRS Scene Center Latitude
WRS Scene Center Longitude
Scene Heading

Method:

Convert input angles to radians:

Inclination_Angle = Pi / 180 * Inclination_Angle
Long_Path1_Row60 = Pi / 180 * Long_Path1_Row60

Compute the Earth's angular rotation rate:

earth_spin_rate = 2 * Pi / (24 * 3600)

The solar rotation rate is used rather than the sidereal rate, as called for in the General Electric Landsat D Program Information Release (PIR), which describes the Worldwide Reference System, to account for the orbital precession that is designed to make the orbit sun synchronous. Thus, the apparent Earth angular velocity is the inertial (sidereal) angular velocity plus the precession rate, which, by design, is equal to the solar angular rate.

Compute the spacecraft's angular rotation rate:

SC_Ang_Rate = 2 * Pi * WRS_Cycle_Orbits/(WRS_Cycle_Days * 24 * 3600)

Compute the central travel angle from the descending node:

Central_Angle = (Row – Descending_Node_Row)/Scenes_Per_Orbit * 2 * Pi

Compute the WRS geocentric latitude:

WRS_GCLat = asin(–sin(Central_Angle) * sin(Inclination_Angle))

Compute the longitude of Row 60 for this Path:

Long_Origin = Long_Path1_Row60 – (Path – 1) * 2 * Pi/WRS_Cycle_Orbits

Compute the WRS longitude:

Delta_Long = atan2(tan(WRS_GCLat)/tan(Inclination_Angle),
        cos(Central_Angle)/cos(WRS_GCLat))

WRS_Long = Long_Origin – Delta_Long – Central_Angle *
        Earth_Spin_Rate / SC_Ang_Rate

Make sure the longitude is in the range +/– Pi:

where (WRS_Long > Pi): WRS_Long = WRS_Long – 2*Pi

where (WRS_Long < –Pi): WRS_Long = WRS_Long + 2*Pi

Compute the scene heading:

Heading_Angle = atan2(cos(Inclination_Angle)/cos(WRS_GCLat),
        –cos(Delta_Long)*sin(Inclination_Angle))

Convert the WRS geocentric latitude to geodetic latitude:

WRS_Lat = atan(tan(WRS_GCLat) * (Semi_Major_Axis/Semi_Minor_Axis)
        * (Semi_Major_Axis/Semi_Minor_Axis))

Convert angles to degrees:

WRS_Lat = WRS_Lat * 180 / Pi
WRS_Long = WRS_Long * 180 / Pi
Heading_Angle = Heading_Angle * 180 / Pi

Round off WRS lat/long to the nearest whole arc-minute:

WRS_Lat = round(WRS_Lat*60) / 60
WRS_Long = round(WRS_Long*60) / 60

Return the answers:

WRS_Lat
WRS_Long
Heading_Angle

The WRS heading angle, from geodetic north, needs to be converted from the WRS scene center to a frame orientation angle in map coordinates. This can be done by first converting the WRS scene center latitude/longitude to map projection X1, Y1 and moving it to a point slightly north of the WRS scene center. This is done by adding 1 microradian (0.2 seconds) to the WRS scene center latitude and projecting this point to X2, Y2. Next, the azimuth of this line is computed in grid space as the arctangent of the X difference (X2–X1) over the Y difference (Y2–Y1). This is the grid azimuth of geodetic north at the WRS scene center. This angle is added to the geodetic heading to produce the grid heading. A standard framed scene puts the satellite direction of flight at the bottom of the scene, so the scene orientation angle is the grid heading +/– 180 degrees. If the grid heading is less than zero, add 180 degrees.   If the grid heading is greater than zero, subtract 180 degrees. This is the scene orientation angle to be used with the WRS scene center and the fixed standard scene size to be used in calculating the frame corners.

The preceding approach does not take into account the skewing of the image data. Skewing of the image is due to the Earth's' rotation relative to the spacecraft's absolute velocity.

A delta X and delta Y are calculated from the center of the projection space to the four corners of the output frame area using the output pixel size and the number of lines and samples in the output space.

For example, calculation of the upper-right delta X and delta Y:

delta_x = (number_samples – center_sample) * pixel_size
delta_y = (center_line – 1.0) * pixel_size

The deltas and the scene orientation angle are then used to calculate the rotated projection coordinate of each corner:

ur_proj_x = delta_x * cos(angle) + delta_y * sin(angle) + center_proj_x
ur_proj_y = –delta_x * sin(angle) + delta_y * cos(angle) + center_proj_y

A least squares routine is used to generate a first-order polynomial, which maps output image (line, sample) coordinates to rotated output projection coordinates.

The output pixel size and projection coordinates are saved in the mapping grid file.

3.1.5.3.5.3 Gridding the Input Space

For the process of rectifying the ETM+ satellite imagery to a given projection, a regular grid is defined over the input space, one set per scan. This separate, forward mapping is necessary to correct properly for gaps between scans and because an inverse satellite model cannot be defined. The first and last lines from each scan are represented in the gridding of the input. The difference between the last line of a scan (line 16 for TM multi-spectral bands, line 8 for the thermal band, and line 32 for the PAN band) and the first line of the next scan is used to determine the scan gap. These scan gaps are corrected in the ETM+ geometric resampling process. Refer to the Landsat 7 Wide Band Data Format Book (Reference 1) for more information about the Landsat 7 instrument and scan gaps. The grid spacing in the along-scan (sample) direction is 96 pixels—a factor that divides into the total number of input image samples and that coincides with the ADS sampling rate. (The typical value used for Landsats 4 and 5 was 128 pixels.) Figure 3-19 illustrates the gridding of an input image:
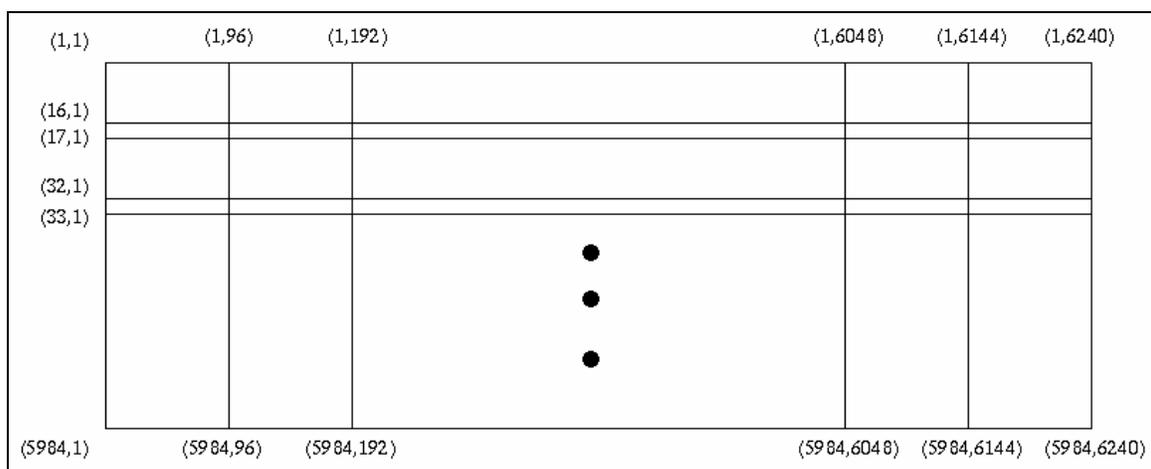
Figure 3-19. Input Image Gridding Pattern Relating Input to Output Space

The Landsat 7 mapping grids are defined in image (line, sample) coordinates that map the input (Level 0) line, sample coordinates to the output projected space line, and sample coordinates. At each grid intersection (input line and sample), the initialized Landsat satellite model is called to calculate the geometric latitude and longitude at that point. Using the projection definition information as input to the Projection Transformation Package, the output projection coordinate is then calculated for each latitude and longitude point. From the output projection coordinates, an output line and sample location, in the defined output frame, are calculated for each point. For a path-oriented product, the polynomial is evaluated at each output projection coordinate. This mapping takes care of both the rotation of the projection space and the scaling from projection coordinates to image (line, sample) coordinates. The polynomial (called once for x coordinates and once for y coordinates) is of the form:

$$x' = a0 + a1x + a2y$$

where:

x' is the output line or sample
x and y are output space projection coordinates

For the other output projections (when there is no rotation between the image coordinate system and the projection coordinate system), the output line and samples are calculated directly using the following equations:

out_line = (ul_proj_y – proj_y) / pixel_size + 1
out_samp = (proj_x – ul_proj_x) / pixel_size + 1

The input line and sample, output line and sample, and projection information are written to a grid file. This geometric mapping grid file defines the transformation between the satellite image and the specified output projection. The grid file is used by the resampler to rectify geometrically the input image to the user-specified output image frame, projection, and pixel size.

The above process is used to create a grid for each band that is to be processed. The net effect is that there is a grid for each band in the grid file. This is necessary because of timing offset differences between bands.

3.1.5.3.6 Resampling

This section describes the IAS image resampling algorithm.

3.1.5.3.6.1 Calculate Scan Gap

This algorithm calculates the scan gap and the sample shift between each grid cell of two consecutive scans. This information is used to create a synthesized or extended image that fills any gaps or "missing data" between two scans. A "scan gap" refers to a line offset between consecutive scans that is larger than the nominal scan line spacing, while a "misalignment" refers to a sample offset between consecutive scans in the image. A least squares routine is used in this algorithm to calculate mapping coefficients. The algorithm follows:

1. Loop through each grid cell, calculating the scan gap and misalignment. This is done using the mapping equations for each grid cell:

    Forward mapping equations:

    $$So = a0 + a1 * Si + a2 * Li + a3 * Si * Li$$
    $$Lo = b0 + b1 * Si + b2 * Li + b3 * Si * Li$$

    where :

    Si is the input sample
    So is the output sample
    Li is the input line
    Lo is the output line
    a0, a1, a2, a3, b0, b1, b2, b3 are the bilinear mapping coefficients

    Inverse mapping equations:

    $$Si = c0 + c1 * So + c2 * Lo + c3 * So * Lo$$
    $$Li = d0 + d1 * So + d2 * Lo + d3 * So * Lo$$

    where:

    Si is the input sample
    So is the output sample
    Li is the input line
    Lo is the output line
    c0, c1, c2, c3, d0, d1, d2, d3 are the bilinear mapping coefficients

    The four corners of each grid cell are used to solve for the mapping coefficients.

    By using the forward and inverse mapping equations for each grid cell of a scan, along with the mapping equations for the grid cell of the scan directly below, the maximum gap is obtained. (If scan N cell M is being processed, then it is compared with scan N+1 cell M). This is shown in Figure 3-20.

    Also during this time, the mapping coefficients are calculated (forward and inverse) for each grid cell, and the inverse coefficients are stored in the grid structure.

    These values are obtained from the following steps (scan N cell M is used to designate one grid cell):

a. Loop through each grid cell of each scan, except for the last scan, since this scan cannot have a gap. Repeat steps 2–8 for the number of grid cells.

b. Create a fictitious "extended" pixel 17E, located directly (one line length) below scan N cell M.

c. Find the output-to-input (inverse coefficients) and input-to-output (forward coefficients) mapping coefficients for scan N cell M.

d. Store the inverse mapping coefficients in the grid structure.

e. Map pixel 17E to the output space using the forward mapping equations for scan N cell M.

f. Find the output-to-input mapping coefficients for scan N+1 cell M.

g. Map the output pixel created in step 5 back to the input space using the inverse mapping equations for scan N+1 cell M (call this point xg, yg).

h. Note that the difference between (xg, yg) and sample 1/line 1 of scan N+1 cell M represents the misalignment and the scan gap for scan N cell M. See Figure 3-20.

i. Find inverse mapping coefficients for grid cells of last scan and store these coefficients in the grid structure.

2. The maximum scan gap in the image is found. The maximum scan gap and the resampling kernel size determine the size of the intermediate extended image. These two parameters determine the number of gap, or extended, lines needed (gap_lines).

   A gap of greater than 1/32 of a pixel is considered large enough to add one extra line to each scan. This number represents the pixel increments of the resampling kernel.

The image must also be extended according to the resampling kernel size. A value of kernel size minus one makes it possible to need only one scan of data at a time during resampling. If this were not done, then at the bottom of each scan, the resampling kernel would step into the scan directly below. Due to scan gaps and misalignment, these data would need to be resampled to "fit" evenly spaced pixels, the current scan being processed. To avoid a redundancy of this step, it is only done once up front during extended image generation.

See Reference 26 and Reference 27 for more information about the characteristics of a scan gap. Refer to Reference 28, Reference 29, and Reference 18 for more information about geometric corrections and transformations.

3.1.5.3.6.2   Creating the Extended Image

References 26 and 27 describe scan gap correction. These documents refer to Landsats 4 and 5. These documents also follow the traditional three-pass approach for extending the gap and creating an output image.

Each scan is extended by both the maximum number of lines missing due to the scan gap and due to the resampling kernel size.

The algorithm is as follows:

1. Create weight tables. A set of cubic convolution weights is set up with 1/32 pixel increments. This table is used to resample the data in the sample direction. The weights are set up for a shift of 0.0–1.0 pixels at 1/32 increments.   A set of spline weights is set up with 1/32 pixel increments.   This table is used to resample in the line direction. The weights are set up for a shift of 0.0 to the maximum gap possible from the system. A value of 3.0 covers the maximum gap possible in the Landsat 5 satellite. A value of 6.0 is needed for the Landsat 7 satellite to cover the PAN band.

2. Loop through the image, creating extra image data after each grid cell of every scan, except for last scan, which does not need to be extended (for the processing of scan N cell M). Repeat steps 3–6 for the appropriate number of extended lines needed.

3. Use the grid to calculate misalignment and scan gap for the current grid cell.

4. Determine data needed.

    a. If the gap is zero, then the lines are extracted only from the scan below and resampled to adjust for scan misalignment and detector delays.

    b. If the scan gap is greater than zero, then the first extended line is created from the last two lines from scan N and first two lines from scan N+1. These lines are used as long as an extended line falls in between their line locations. When the extended line starts to fall into scan N+1, going beyond line 1 of scan N+1, then the lines used must be incremented appropriately. This can be explained by looking at Figure 3-20. If x1=1.0, x2=2.0, x3=4.5, and x4=5.5 correspond to the line locations of the last two lines in scan N and the first two lines in scan N+1, respectively, then extended lines are created at 3.0 and 4.0 using these four lines. In this case, both extended lines fall between these four lines. However, when an extended line is created at 5.0, a new set of lines needs to be chosen, such that x1=3.0, x2=4.0, x3=5.5, and x4=6.5. At this point, the first two lines are the previously created extended lines, while the other two lines are from scan N+1 and correspond to lines 2 and 3. This pattern then follows with two extended lines being used along with two lines from scan N+1 for all other lines created.

    c. If the scan gap is less than zero, then the first extended line is created with two lines from scan N and scan N+1. The lines taken correspond to the last two from scan N and two from scan N+1, incremented by the appropriate offset. If the scan gap is 0.5, then lines 15 and 16 from scan N are used, along with lines 2 and 3 from scan N+1. After the first extended line is created, it is used for further processing and only one new line from scan N+1 is needed. For the example just given, the second iteration would

include line 16 from scan N, the extended line created in the previous iteration, along with lines 3 and 4 from scan N+1.

5. The lines taken from scan N are resampled to adjust for detector delays.   The data are resampled using cubic convolution for a constant shift across the whole line, where the shift is equal to the detector delay of that line.

6. The lines taken from scan N+1 are resampled to adjust for detector delay and misalignment. The data are resampled using cubic convolution so that they are scaled and shifted to account for geometric misalignment and scaling differences between the scan N cell M and scan N+1 cell M. The resampling also takes into account detector delays. The transformation is treated as a linear relationship, mapping the lines from scan N+1 so that the samples line up in the sample direction.

 This is done through the following relationship:

        S1N1 = seg * ns + 1
        S2N1 = (seg+1)*ns + 1

        S1N2 = S1N1 – misalignment[seg];
        S2N2 = S2N1 – misalignment[seg+1];

        m = (S2N2 – S1N2) / (S2N1 – S1N1);
        b = S1N2 – S1N1 * a1 + delay;

        new_sample = m * old_sample + b;

  where:

        S1N1 = first sample from scan N
        S2N1 = second sample from scan N
        S1N2 = first sample from scan N + 1
        S2N2 = second sample from scan N + 1
        m and b = mapping coefficients
        out_sample = newly mapped sample
        in_sample = sample from scan N+1
        delay = detector delay for line from scan N+1
        seg = current scan cell
        misalignment = scan misalignment between two cells
        ns = length of a cell

7. A new line is produced using the four lines created from steps 4 and 5. This is done using a cubic spline, which takes into account the unequal spacing between pixels due to the scan gap. The scan gap is treated as a linear function over the scan cell.

This is done through the following relationship:

$$m = (gap\_at\_end - gap\_at\_start) / ns$$
$$b = gap\_at\_start$$
$$new\_gap = m * sample + b$$

where:

gap_at_start = size of scan gap at the start of grid cell
gap_at_end = size of scan gap at the end of grid cell
m and b = mapping coefficients
ns = length of a cell
sample = sample position of current resampling step (These values are from 0 to SAMP_RATE − 1.)
new_gap = gap associated with sample

See Figure 3-22 and Figure 3-23.

3.1.5.3.6.3   Notes on the Cubic Spline

The cubic spline weights were developed by Fischel (Reference 30) and take on the form of (see Figure 3-21):

$$w1 = (U2/(3*h1*h2*h2))*(2*F-G)$$
$$w2 = (x3-x)/h2- (F/(3*h1*h2*h2*h2))*(2*U2*(h1+h2)+h1*V2)$$
$$\quad + (G/(3*h1*h2*h2*h2))*(U2*(h1+h2)+2.0*V2*h1)$$
$$w3 = ((x-x2)/h2)+(F/(3*h2*h2*h2*h3))*(2*U2*h3+V2*(h2+h3))$$
$$\quad - (G/(3*h2*h2*h2*h3))*(U2*h3+2*V2*(h2+h3))$$
$$w4 = (V2/(3.0*h2*h2*h3))*(2.0*G-F)$$

where:

$$h1 = x2 - x1$$
$$h2 = x3 - x2$$
$$h3 = x4 - x3$$
$$F = (x3-x)*((x3-x)*(x3-x) -h2*h2)$$
$$G = (x-x2)*((x-x2)*(x-x2) -h2*h2)$$
$$U2 = h2/(h1+h2)$$
$$V2 = h2/(h2+h3)$$

Figure 3-24 shows the spline weights calculated for different size gaps. At larger gaps, the weights show greater variation. When the PAN band is used for Landsat 7, this could cause some anomalies in cases of large, varying grey values between pixels that are located in an area with a large scan gap.

3.1.5.3.6.4   The Pseudo-Inverse Model

Due to the fact that there is no inverse model, a method is needed to find which output pixel corresponds to which input pixel. This is done using the inverse mapping coefficients for each grid cell. To do this, it must be determined which grid cell goes with which output pixel. A "rough" polynomial is set up that gives an approximate input pixel for a given output pixel. An approximate grid cell location can then be determined from this pixel. After this approximation is performed, the correct grid cell can be found using a trial and error process with the inverse mapping coefficients for the grid cells.

3.1.5.3.6.5   Calculating the "Rough" Polynomial

A first-order polynomial is used for the "rough" mapping polynomial. The data used to produce this polynomial are taken from the grid input and output pixel locations. The first-order polynomial is used primarily for speed and computational reasons. The algorithm is as follows:

1. Read input and output lines/samples (not all are needed).
2. Use a least squares routine to calculate a first-order polynomial. This creates a "rough" polynomial:

$$Si = a0 + a1 * So + a2 * Lo + a3 * So * Lo$$
$$Li = b0 + b1 * So + b2 * Lo + b3 * So * Lo$$

where:

Si is the input sample
So is the output sample
Li is the input line
Lo is the output line
a0, a1, a2, a3 are the polynomial mapping coefficients
b0, b1, b2, b3 are the polynomial mapping coefficients

3.1.5.3.6.6   Finding the Corresponding Grid Cell

The algorithm to determine the corresponding input grid cell for a given output grid cell is as follows (If the inverse mapping routine is caught or jumps back and forth between two scans, then it is assumed that the output pixel falls in the gap between the two scans. In this case, the scan above the gap is passed back as the appropriate grid cell.):

1. Map the output pixel to the current_input_space_point using the "rough" polynomial.
2. Set the current_grid_cell from the current_input_space_point.
3. Loop:

a. Calculate the new_input_space_point using the inverse polynomial for the current_grid_cell.
b. Calculate the new_grid_cell from the new_input_space_point.

- If the current_grid_cell and the new_grid_cell are the same, the correct grid cell has been found; exit loop. Otherwise, copy the new_grid_cell to the current_grid_cell. Save the current_grid_cell to the history list.
- If the current_grid_cell is the same as the grid cell from the previous iteration, then:
  - If the matching grid cell is from two iterations ago, and the intervening grid cell is from the same row, use the last grid cell as the correct grid cell. Otherwise, set a flag to indicate the output pixel maps to the gap; use the grid cell in the cycle from the upper-most row as the correct cell.

4. Exit loop.

See Figure 3-25, Figure 3-26, and Figure 3-27.

3.1.5.3.6.7   The Resampling Process

The output image is created by using the extended image and the image grid. Currently, a two-dimensional cubic convolution kernel is used for resampling.

This processing flow may need to be revised if a new weight table is needed due to detector delays or to the Modulation Transfer Function (MTF). This process does not contain any precision or terrain processing concerns.

The algorithm is as follows:

1. Create resampling table weights (see 3.1.5.3.6.12).
2. Step through the output space (for each pixel). Repeat steps 3–8 for all output pixels.
3. Use the "rough" polynomial to map an output pixel to input space.
4. Use the find corresponding grid cell algorithm to determine which input image grid cell the output pixel falls in, along with the input pixel location for the corresponding output pixel.
5. Determine and obtain necessary data in the extended image.
6. Determine correct the resampling weights.

- If cubic convolution or MTF compensation is being used, find the fractional portion of the input line/sample to look up the correct set of resampling weights.
- If nearest neighbor resampling is being used, then find the nearest input pixel.

7. Perform resampling. Multiply each input pixel by the corresponding weight and sum the results.
8. Store the output pixel.

See References 12, 27, and 28 for more information about the resampling process.
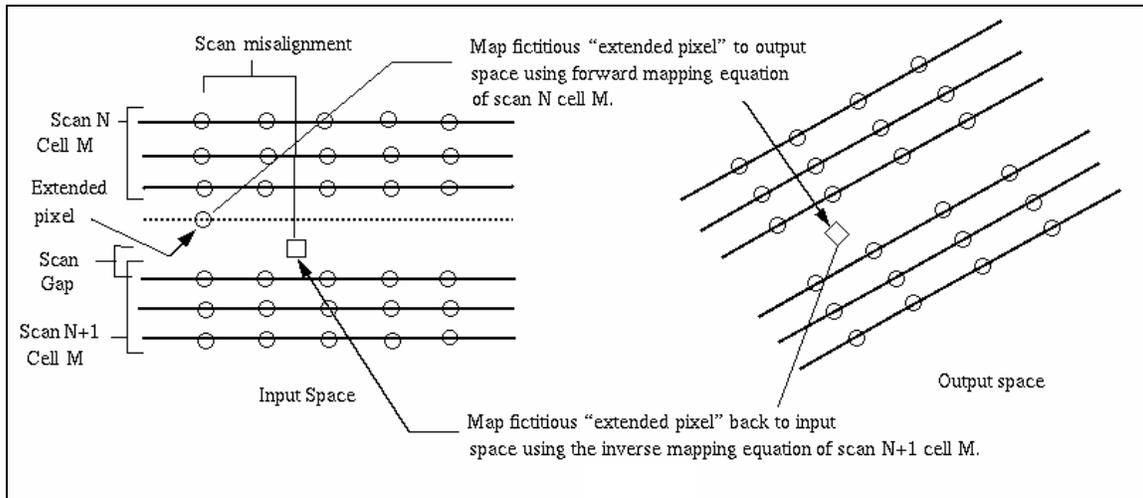
Figure 3-20. Extended Pixels and Scan Alignment
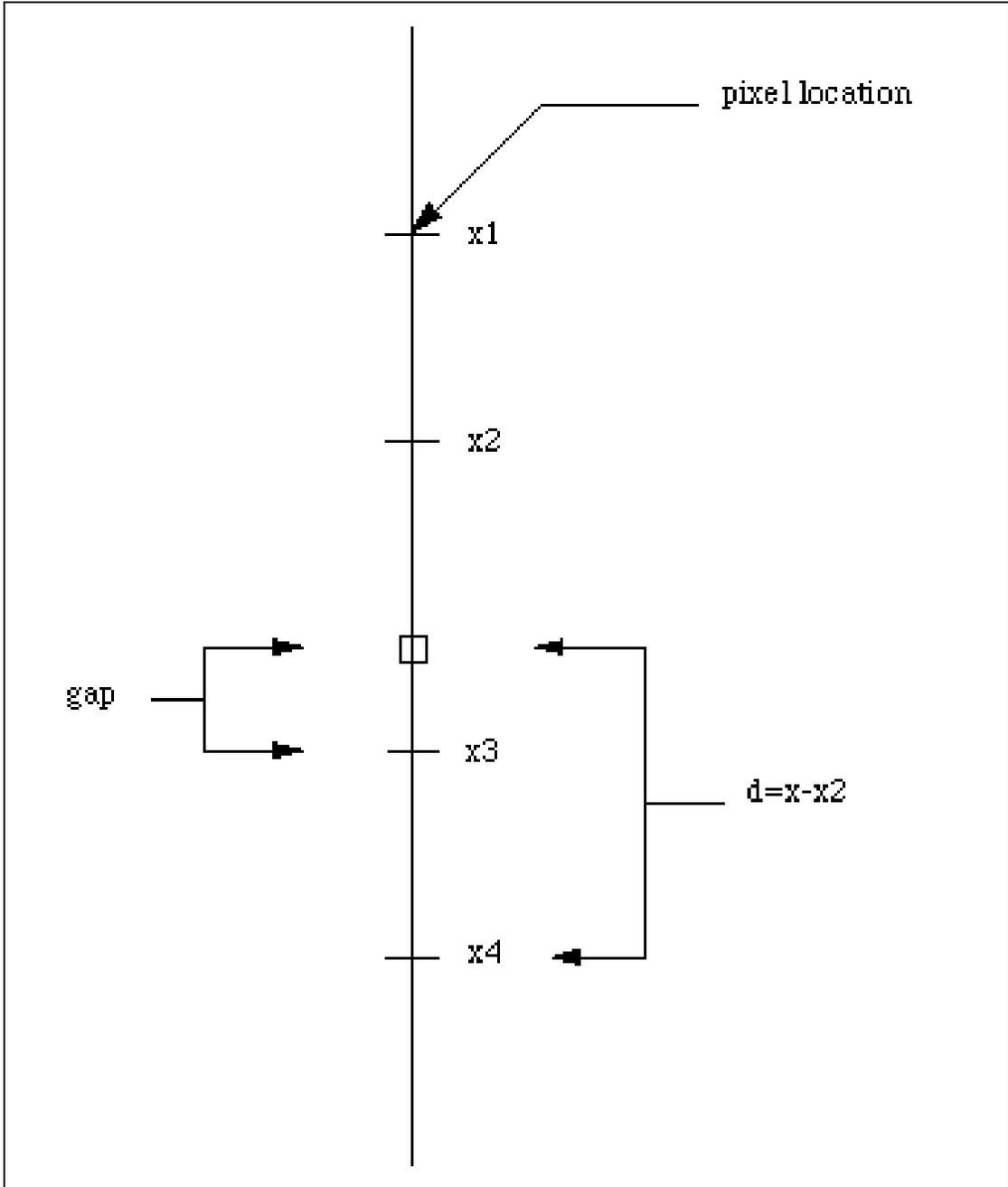
Figure 3-21. Calculation of Scan Gap

The diagram above represents the last two lines of scan N and the first two lines of scan N+1. The diagram shows detector delays within a scan along with the scan misalignment and gap between the scans. These four lines will be used to create an extended line. The first step will be to resample the data in the sample direction. This will align the pixels in the sample direction. It should be noted that the pixels are equally spaced in the sample direction so cubic convolution can be used. Lines 15 and 16 from scan N need to be resampled to remove detector delays. Lines 1 and 2 from scan N+1 need to resampled to remove detector delays and to remove the scan misalignment between it and scan N.

Figure 3-22. Scan Misalignment and Gap

The diagram above shows the data from scan N and N+1 resampled so that the pixels align in the sample direction. It also shows the extended line that will be created by resampling the data in the line direction. It should be noted that the pixels are not evenly spaced in the line direction due to the gap, therefore a cubic spline is used for resampling.
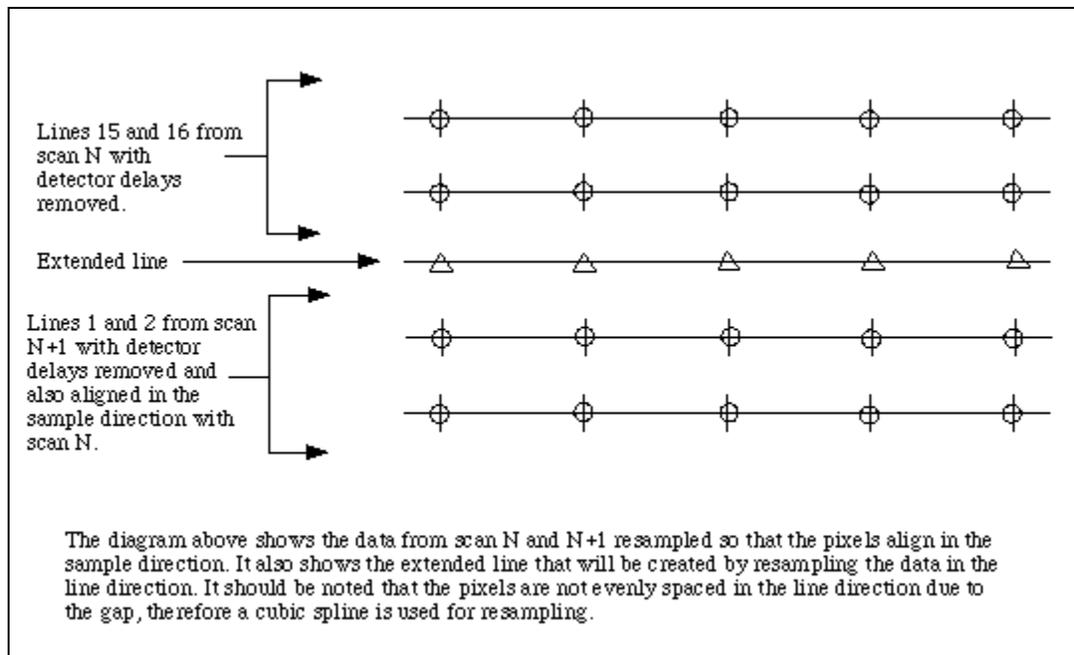
Figure 3-23. Extended Scan Lines

LS-IAS-01
Version 1.0

Gap size	w1	w2	w3	w3
0.000001	0.000000	0.000001	1.000000	0.0000000
0.3	-0.030100	0.206843	0.923591	-0.100334
0.6	-0.086538	0.425481	0.805288	-0.144231
1.0	-0.166667	0.666667	0.666667	-0.166667
1.3	-0.222661	0.810219	0.583720	-0.171278
1.6	-0.273504	0.928337	0.516108	-0.170940
2.0	-0.333333	1.055556	0.444444	-0.166667
2.3	-0.372798	1.133619	0.401264	-0.162086
2.6	-0.408213	1.200215	0.365003	-0.157005
3.0	-0.450000	1.275000	0.325000	-0.150000
3.3	-0.477841	1.322734	0.299907	-0.144800
3.6	-0.503106	1.364704	0.278153	-0.139752
4.0	-0.533333	1.413333	0.253333	-0.133333
4.3	-0.553759	1.445264	0.237276	-0.128781
4.6	-0.572511	1.473949	0.223021	-0.124459
5.0	-0.595238	1.507937	0.206349	-0.119048
5.3	-0.610785	1.530712	0.195315	-0.115242
5.6	-0.625199	1.551496	0.185346	-0.111643

Figure 3-24. Cubic Spline Weights
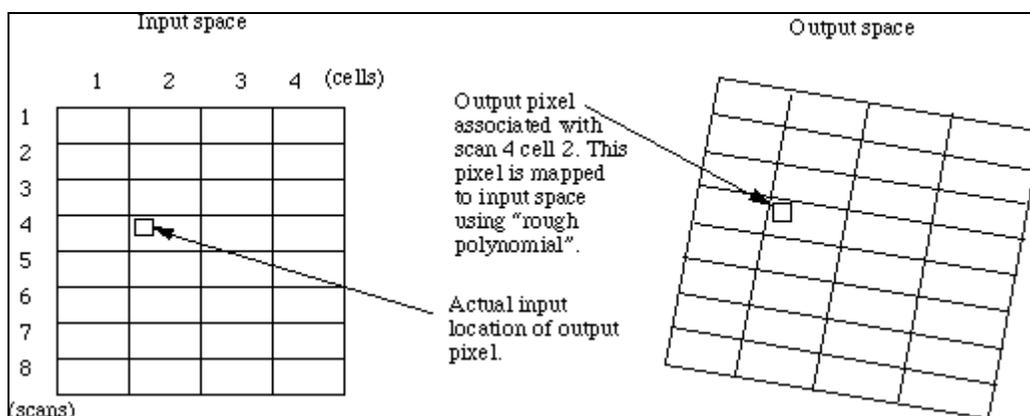
Figure 3-25. Inverse Mapping with "Rough" Polynomial

LS-IAS-01
Version 1.0

"rough polynomial" or best guess as to where output pixel falls in input space. The scan and cell associated with this input pixel is then calculated (scan 2 cell 3). The bilinear polynomial representing the inverse mapping for this scan and cell is now used to map the output pixel to the input space.

Note: In input space the scan number is found by dividing the input line by the number of lines per scan while the cell number is found by dividing the sample number by the number of samples per cell.

Figure 3-26. "Rough" Polynomial – First Iteration

When output pixel is mapped back to input space using the bilinear polynomial for scan 2 cell 3 a new input pixel is found. The scan and cell for this new input pixel is found (scan 2 cell 2). If this is the same scan and cell used to map the output pixel (scan 2 cell 3) then the correct scan and cell has been found. If the scan and cell are not the same then a new scan and cell is chosen and the process is repeated.

Figure 3-27. Results Mapped Back to Input Space

Figure that shows the nearest neighbor resampling with labels: Pixels in a scan, Scan N, line 14, line 15, line 16, Scan gap 1 line, Scan N+1, line 1, line 2, line 3, Output pixel mapped to input space using equations for scan N., Output pixel mapped to input space using equations for scan N+1, To the left is an example of how the same output pixel can be mapped to two different input locations when the mappings are associated with two different grid cells.

Figure 3-28. Nearest Neighbor Resampling

3.1.5.3.6.8 Nearest Neighbor Resampling

To perform nearest neighbor resampling, changes to the above algorithm must be made. To get the true nearest neighbor, the scan gap is not extended. This presents two scenarios: one case where the pixel falls within a grid cell, and the resampling follows the traditional nearest neighbor routine; and a second case where the pixel falls in the scan gap area. In the second scenario, extra work must be performed to determine which pixel is really the true nearest neighbor. The confusion in determining the true nearest neighbor lies in the fact that each grid cell defines its own coordinate system. When a pixel falls within the gap, the transformations between both grid cells must be used to determine which pixel is truly closest for nearest neighbor resampling. This is demonstrated in Figure 3-28.

3.1.5.3.6.9 Linear Interpolation Resampling

Due to the effects of detector delays, the linear interpolation employed is slightly different than the classical bilinear approach. Detector delays are shifts in the sample direction with respect to the raw imagery. When these offsets are taken into account during linear interpolation, the sample actually represents the centroid of the polygon, which is made up of the four samples that surround the output sample location. These values are solved directly. There is not a table of values representing a set of offsets for linear interpolation resampling as there is for cubic convolution resampling.

3.1.5.3.6.10 Notes on the Correction Grid

Input lines in the grid must contain the top and bottom lines of a scan (for scan N, lines 1 and 16 must be contained in the input lines of the grid). This is made necessary by the way the scan gap is calculated. This algorithm expects the sample spacing between grid points to divide evenly the total number of samples in the image. The pixel locations given in the grid refer to the pixel center location. See 3.1.5.3.5, Correction Grid Generation, for more information on the gridding process.

3.1.5.3.6.11   Notes on the Least Squares Routine

This algorithm uses the QR decomposition functions, which are based on information in References 6 and 31.

3.1.5.3.6.12   Detector Delays

A detector delay represents the virtual distance of each detector from its respective band center. It contains the net effect of detector misplacement on the focal plane and the effective sampling delays due to electronics and scan angle monitor delays. The manner in which they are defined is illustrated in Figure 3-29.

Since the delays are different for each detector of each band, they cannot be centered in the sparse Geometric Correction Grid. Therefore, the detector delay information has been incorporated into the weight tables for resampling. The way in which they have been applied assumes that the nominal position of the detectors has been used in determining the along-scan angle in the forward satellite model (see Figure 3-29). It is the difference between the real and ideal detector positions that is applied to the input sample when determining the resampling weight table. Following this method, the difference is added to the input samples of forward scans and subtracted from reverse scans. To simplify the application of detector delays in calculating the resampling weights, the minus sign has been included in the definition of the delay correction term for the reverse scan so that it can be added regardless of scan direction. Therefore:

new sample = old sample + delay correction

where:

$$
\text{delay correction} = \begin{cases} (\text{delay} - \text{nominal}): \text{Forward} \\ -\ (\text{delay} - \text{nominal}): \text{Reverse} \end{cases}
$$
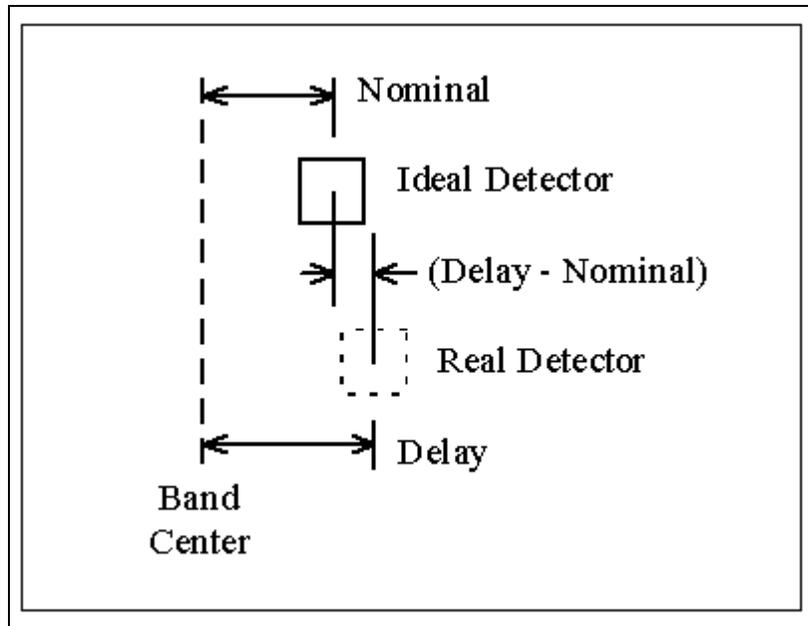
Figure 3-29. Detector Delay Definition

Detector delays are defined as displacements relative to band center.

Prior to resampling, a set of weights are determined that incorporate the non-ideal locations of the detectors. In Figure 3-30, the ideal line and sample locations are shown as dotted lines. When an output (line, sample) pair is mapped to input space, it generally falls on a non-integer (line, sample). Typically, the surrounding integer pixels are used to generate weights for resampling (i.e., cubic convolution, MTF, bilinear, etc.). To incorporate detector delays, it is necessary to determine the weights based on the distance from the non-integer (line, sample) in input space to the actual detector location (see Figure 3-30). Weights for the synthetic "detectors" in the extended lines are calculated under the assumption that they act as ideally placed detectors.

The standard cubic convolution uses a four-segment, piece-wise third-order polynomial over the range [–2, +2] (see Reference 12). It is defined so that beyond [–2, +2] the weights are zero. From Figure 3-30, note that if the detectors were ideally located at S-1... S+2, then it would be necessary to calculate weights at these four samples only (for each of the four lines L-1... L+2). However, since the detectors can be displaced from their nominal positions in either the positive or negative direction, it is necessary to calculate weights over the range S-2... S+3. For every line, two of these weights are zero, but which two weights are zero depends on the magnitude and sign of the displacement and the location of the non-integer (line, sample).
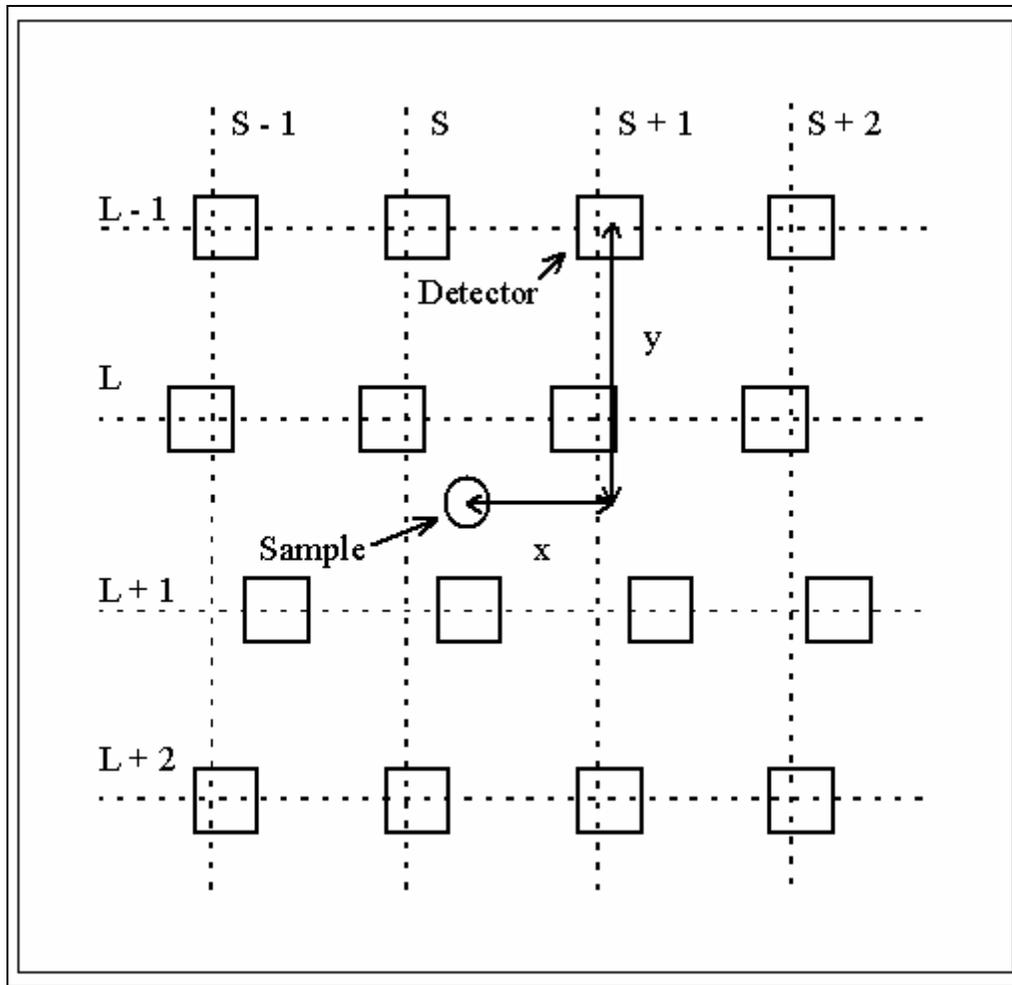
Figure 3-30. Resampling Weight Determination

Resampling weights are determined based on the x and y distances between the non-integer (line, sample) and the real detector locations

3.1.5.3.7 Precision Correction

3.1.5.3.7.1 Defining the Precision Correction Problem

The geometric measurement in the ETM sensor system can be regarded as the look vector, l_{sc}, in the spacecraft body-fixed system. This vector is transformed into the
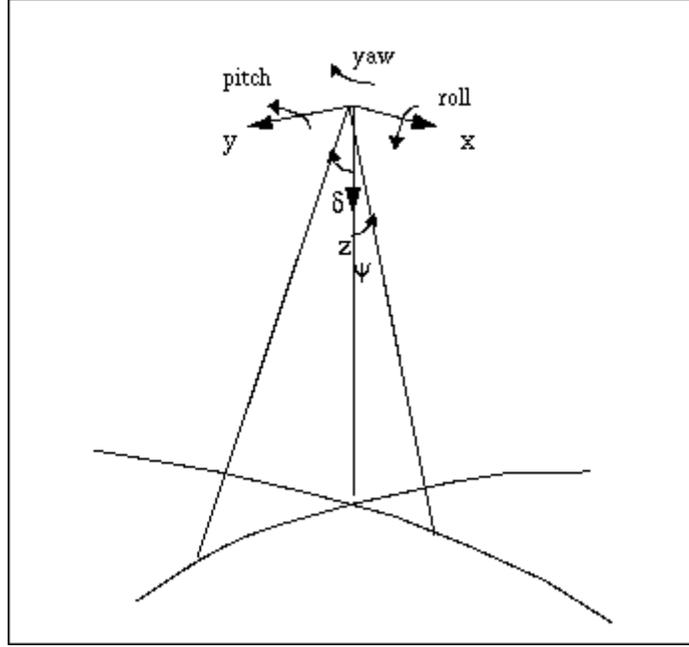
Figure 3-31. Definition of Orbit Reference System

Orbit Reference Frame (OB) system (see Reference 2 and Figure 3-31), through the spacecraft attitude parameters:

$$I_{ob} = T^T(\xi_r, \xi_p, \xi_y)\, I_{sc} \tag{1.1}$$

where ξ_r, ξ_p, and ξ_y are roll, pitch, and yaw angles, and T is the transformation matrix, which can be expressed as:

$$T(\xi_r, \xi_p, \xi_y) = R_3(\xi_y)R_2(\xi_p)R_1(\xi_r)$$

$$= \begin{bmatrix} \cos\xi_p\cos\xi_y & \cos\xi_r\sin\xi_y + \sin\xi_r\sin\xi_p\cos\xi_y & \sin\xi_r\sin\xi_y - \cos\xi_r\sin\xi_p\cos\xi_y \\ -\cos\xi_p\sin\xi_y & \cos\xi_r\cos\xi_y - \sin\xi_r\sin\xi_p\sin\xi_y & \sin\xi_r\cos\xi_y + \cos\xi_r\sin\xi_p\cos\xi_y \\ \sin\xi_p & -\sin\xi_r\cos\xi_p & \cos\xi_r\cos\xi_p \end{bmatrix}$$

where the approximation for small angles ξ_r, ξ_p, ξ_y

$$= \begin{bmatrix} 1 & \xi_y & -\xi_p \\ -\xi_y & 1 & \xi_r \\ \xi_p & -\xi_r & 1 \end{bmatrix} \tag{1.2}$$

where R_1, R_2, and R_3 are the coordinate system transformation matrix for rotation around x, y, and z axes, respectively.

The vector l_{ob} is further transformed into the Earth-fixed (EF) system:

$$l_{ef} = T_f(r_{ef}, v_{ef})\, l_{ob} \tag{1.3}$$

where T_f is the forward transformation for vectors from the OB system to the EF system, as a function of the satellite position r_{ef} and velocity v_{ef} vectors in the EF system. Note that v_{ef} should be the "inertial" velocity expressed in the EF system, as described by Reference 13. This vector, l_{ef}, together with the satellite position vector, r_{ef}, is then used to intersect the ellipsoid Earth surface to define a point position, R_{ef}, as the target point on the Earth. This is the common forward image pixel geolocation calculation.

Mathematically, R_{ef} is a function of l_{sc}, ξ_r, ξ_p, ξ_y, r_{ef}, and v_{ef}:

$$R_{ef} \;=\; F(l_{sc},\, \xi_r,\, \xi_p,\, \xi_y,\, r_{ef},\, v_{ef}) \tag{1.4}$$

Because of errors in the satellite orbit ephemeris and attitude data, this calculated R_{ef} is different from the true location of the image pixel. If we know the true location of a landmark pixel, R_{cp}, from other sources (e.g., base map, survey, etc.), this point can be taken as a Ground Control Point (GCP) and used to check the accuracy of the computed image pixel location. The precision correction process uses the GCP coordinates to estimate the correction to the satellite ephemeris and attitude data, so that with the corrected parameters in equation 1.4, the calculated image pixel location, R_{ef}, is close to its true location, R_{cp}, (depending on the GCP positional accuracy).

To calculate the precision correction, the difference between R_{ef} and R_{cp} is taken as the observable, and the observation equation becomes:

$$dR \;=\; R_{cp} - F(l_{sc},\, \xi_r,\, \xi_p,\, \xi_y,\, r_{ef},\, v_{ef}) \tag{1.5}$$

according to equation 1.4. However, the actual calculation of R_{ef} is usually not an explicit function of the orbit and attitude parameters, especially for the intersecting procedure. Therefore, it is inconvenient to linearize equation 1.5 with standard estimation techniques. Instead, the calculation of look vector l_{cp}, corresponding to R_{cp}, in the OB system is much more explicit:

$$\mathbf{l}_{cp} = \mathbf{T}_i(\mathbf{r}_{ef}, \mathbf{v}_{ef}) \frac{\left(\mathbf{R}_{cp} - \mathbf{r}_{ef}\right)}{\left|\mathbf{R}_{cp} - \mathbf{r}_{ef}\right|} \tag{1.6}$$

where $(R_{cp} - r_{ef})$ is the line-of-sight vector in the EF system corresponding to R_{cp}, and $T_i(r_{ef}, v_{ef})$ is the inverse transformation for the look vector from the EF system to the OB system. If all the satellite attitude and ephemeris parameters are accurate, the l_{cp} from equation 1.6 and the l_{ob} from equation 1.1 should be equal. Since the measurement l_{sc} is

accurate compared to the attitude and ephemeris information, any systematic difference between l_{cp} and l_{ob} can be attributed to the attitude and orbit errors. Thus, we can use the difference between l_{cp} and l_{ob} as the observable:

$$dl \;=\; l_{cp} - l_{ob} \;=\; T_i(r_{ef}, v_{ef}) \frac{(\mathbf{R}_{cp} - \mathbf{r}_{ef})}{\left| \mathbf{R}_{cp} - \mathbf{r}_{ef} \right|} \;-\; T(\xi_r, \xi_p, \xi_y)\, l_{sc} \qquad (1.7)$$

The task of precision modeling is then to calculate the correction to those satellite ephemeris and attitude parameters (i.e., r_{ef}, v_{ef} and ξs), so that the residuals of dl after correction are minimized for all selected GCPs. The orbit correction is modeled as a linear function of time for each component in the OB system. Referred to as the short arc method, this purely geometric method shifts and rotates the short arc of orbit defined by the original ephemeris points to fit the GCP measurements.

3.1.5.3.7.2   Linearizing the Observation Equations

In equation 1.7, the calculation of l_{ob} can also be carried out through:

$$l_{ob} = T_i(r_{ef}, v_{ef})(R_{ef} - r_{ef}) \,/\, |R_{ef} - r_{ef}| \qquad (2.1)$$

if R_{ef} is more conveniently accessible. Since equation 2.1 is simply the inverse of equation 1.4 and equation 1.3, the l_{ob} calculated from equation 2.1 is the same as the one in equation 1.1, except for the possible inclusion of numerical errors. However, the true relationship between l_{ob} and the parameters is always equation 1.1. Equation 2.1 should not be confused with this, because R_{ef} in equation 2.1 is not an independent variable but a function of equation 1.4. Therefore, in observation 1.7, information about the attitude parameters is contained in l_{ob}, and information about orbit parameters comes from l_{cp} .

Since the measurement of l_{sc} is two-dimensional in nature, only two-dimensional information is contained in equation 1.7, though three components are involved. If a look vector (either l_{cp} or l_{ob}) has the three components in the OB system:

$$l \;=\; \{xl,\; yl,\; zl\} \qquad (2.2)$$

the real information in these three components can be summarized in two variables, like the original look angle measurements.

The following two variables are used:

$$\delta = \text{atan}(yl/zl) \tag{2.3}$$

$$\psi = \text{atan}(xl/zl) \tag{2.4}$$

so that the three components of equation 1.7 can be reduced to these two equations:

$$\alpha = \delta_{cp} - \delta_{ob} \tag{2.5}$$

$$\beta = \psi_{cp} - \psi_{ob} \tag{2.6}$$

In equations 2.3 and 2.4, the components of *xl, yl, zl* can be those of the line-of-sight vector instead of the unit look vector, so that δ_{cp} and ψ_{cp} are explicit functions of orbit position. In that case, *zl* is approximately the height of the satellite.

If we define:

true value = approximate value + correction

and differentiate equations 2.3 and 2.4 with respect to the orbit position (for δ_{cp} and ψ_{cp}) and equation 1.1 with respect to the satellite attitude (for δ_{ob} and ψ_{ob}) at their corresponding approximate values, then equations 2.5 and 2.6 can be linearized as the function of correction parameters:

$$\alpha \cong (\cos^2\delta_{cp}/h)\, dy - (\cos\delta_{cp}\sin\delta_{cp}/h)\, dz + d\xi_r \tag{2.7}$$

$$\beta \cong (1.0/h)\, dx - d\xi_p + \tan\delta_{cp}\, d\xi_y \tag{2.8}$$

where dx, dy, and dz are the corrections to satellite position vector r_{ob} in the OB system, and $d\xi$s are the corrections to the satellite attitude angle ξs. Other quantities are functions evaluated at the approximate values of r_{ef}, v_{ef}, and ξs.

The above linearization is done by directly differentiating equations 2.3 and 2.4, with transformation T_i regarded as unaffected by the error in r_{ef} and v_{ef}. This, however, ignores the curvature of the satellite orbit and the Earth, resulting in about 10% error in the coefficients of dx, dy, and dz. A more accurate way to evaluate these coefficients is to examine the sensitivity terms $d\psi_{cp}/dx$, $d\delta_{cp}/dy$, and $d\delta_{cp}/dz$ through the geometry of the look vector (see Figure 3-32).
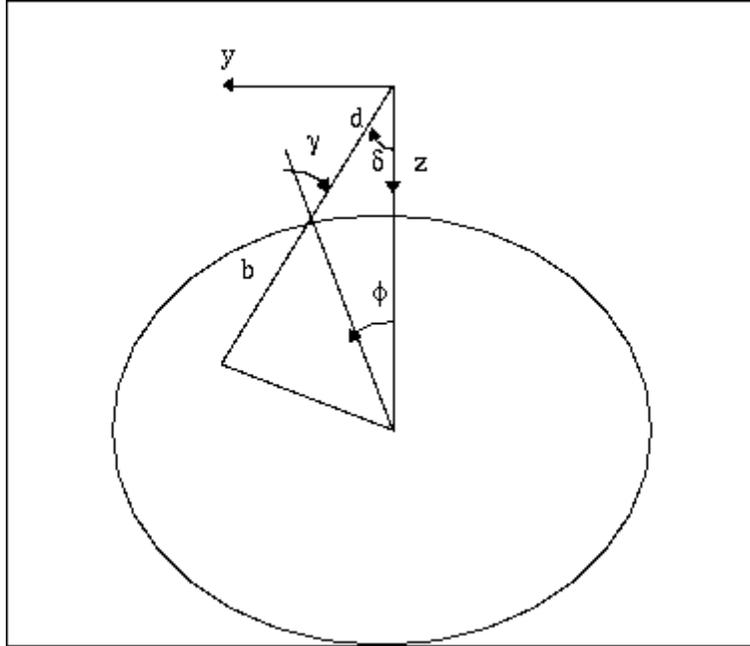
Figure 3-32. Look Vector Geometry

Denote:

R = the radius of the Earth
r = the radius of the satellite position
h = the altitude of the satellite
d = the magnitude of the look vector (from satellite to target)
δ = the along-scan angle of the look vector
ϕ = the Earth-centered angle between the satellite and the target
γ = the zenith angle of the look vector at the target
x, y, z = the coordinates of the satellite position in the OB system

then we have:

$$R \sin(\delta + \phi) = r \sin \delta \qquad (2.9)$$

Differentiating the equation (holding R and r constant) yields:

$$R \cos(\delta + \phi)(d\delta + d\phi) = r \cos\delta \, d\delta \qquad (2.10)$$

Note that $\delta + \phi = \gamma$, and $d\phi = -dy / r$. Then we have:

$$\alpha = d\delta = (-b / (r\,d)) \, dy \qquad (2.11)$$

Similarly, for the cross-scan direction, we have:

$$\beta = d\psi = (-(r - d\cos\delta) / (r\,d\,\cos\delta)) \, dx \qquad (2.12)$$

LS-IAS-01
Version 1.0

For the effect of altitude error, we can differentiate equation 2.9 with respect to δ and r (holding ϕ constant and noting dr = –dz), and then we have:

$$\alpha = d\delta = (\sin \delta / d) \, dz \tag{2.13}$$

The dx, dy, and dz in equations 2.11, 2.12, and 2.13 are error terms, which are opposite in sign from the correction terms. With this in mind, we can replace the correction terms in equations 2.7 and 2.8 and rewrite the linearized observation equation as:

$$\alpha = (b / (r \, d)) \, dy - (\sin \delta / d) \, dz + d\xi_r \tag{2.14}$$

$$\beta = ((r - d \cos\delta) / (rd \cos\delta)) \, dx - d\xi_p + \tan \delta \, d\xi_y \tag{2.15}$$

where:

$$b = R \cos\gamma = \text{sqrt}(R^2 - (r^2 \sin^2 \delta)) \tag{2.16}$$

$$d = r \cos \delta - b \tag{2.17}$$

3.1.5.3.7.3   Weighted Least Squares Solution of the Parameters

The correction parameters in equations 2.14 and 2.15 can be expanded to include the correction to the change rates of the satellite attitude and position by defining:

$$dx = dx_0 + dx_{dot}dt \qquad \text{and} \quad d\xi_r = d\xi_{r0} + d\xi_{rdot}dt \tag{3.1}$$

If the major random error source in computing α and β is the error in the coordinates of the Ground Control Point, then the covariance matrix for the observation equations 2.14 and 2.15 should be the covariance matrix of R_{cp} in equation 1.6, mapped through equations 1.6, 2.3, and 2.4.

In the observation equations 2.14 and 2.15, α is related only to parameters dy, dz, and $d\xi_r$, and β is related only to dx, $d\xi_p$ and $d\xi_y$. The parameters are uncoupled in the two observations. In the simplified case, where observational errors of α and β are uncorrelated, the observation equations can be separated into two independent equations and solved individually.

Define the parameter vectors as:

$$X_1' = \{d\xi_{r0}, dy_0, dz_0, d\xi_{rdot}, dy_{dot}, dz_{dot}\} \tag{3.2}$$

$$X_2' = \{dx_0, d\xi_{p0}, d\xi_{y0}, dx_{dot}, d\xi_{pdot}, d\xi_{ydot}\} \tag{3.3}$$

where ' represents the transpose of a vector or matrix. Then the two observation equations can be written as:

$$\alpha = h_1\,X_1 + \varepsilon_a \qquad\qquad (3.4)$$

$$\beta = h_2\,X_2 + \varepsilon_b \qquad\qquad (3.5)$$

where:

$$h_1 = \{\,1.0, b/(d\ r),\ -\sin\delta/d,\ dt,\ b\ dt/(d\ r),\ -\sin\delta\ dt/d\,\} \qquad\qquad (3.6)$$

$$h_2 = \{(r - d\cos\delta)/(r\ d\cos\delta),\ -1.0,\ \tan\delta,\ (r{-}d\cos\delta)\ dt/(r\ d\cos\delta),$$
$$-dt,\ \tan\delta\ dt\} \qquad\qquad (3.7)$$

ε_a and ε_b are the random errors of α and β, respectively. With all GCPs included, the observation equation can be written as:

$$A = H_1 X_1 + \varepsilon_A \qquad\qquad (3.8)$$

$$B = H_2 X_2 + \varepsilon_B \qquad\qquad (3.9)$$

and the parameters can be solved by Weighted Least Squares (WLS) estimation as:

$$X_1 = (H_1{}'WaH_1)^{-1}\,(H_1{}'WaA) \qquad\qquad (3.10)$$

$$X_2 = (H_2{}'WbH_2)^{-1}\,(H_2{}'WbB) \qquad\qquad (3.11)$$

where A and B are the observation vectors, composed of α and β for all GCPs, respectively; H_1 and H_2 are corresponding coefficient matrices with h_1 and h_2 as rows corresponding to each α and β; Wa and Wb are the diagonal weight matrices for A and B, respectively, composed of the inverse of the variance of each individual ε_a and ε_b.

One problem with this solution is the nearly linear correlation between parameter dx and $d\xi_p$ in observation equation 3.7. The along-track orbit error and the pitch angle error have a very similar effect on β, and the two parameters cannot be separated in the solution without additional information. Including both parameters in the observation equations results in a near-singular normal equation and therefore an unstable solution of the parameters. Similarly, high correlation exists between the cross-track position and the roll attitude errors in equation 3.6, and an ill-conditioned normal equation results.

For the purpose of correcting the image, we do not have to distinguish between orbit position correction and attitude correction parameters. Letting either the orbit or attitude correction parameters absorb the existing errors corrects the image in a similar manner. Therefore, we can choose to estimate either dx and dy or $d\xi_p$ and $d\xi_r$. This can be done by setting those coefficients in h_1 and h_2 that correspond to the unwanted parameters to

zero. In practice, the allocation of correction offsets to ephemeris and attitude terms is controlled by the selection of a priori weights for the correction parameters.

3.1.5.3.7.4   Separating Orbit Position and Attitude Errors

One of challenging tasks in the Landsat 7 Image Assessment System (IAS) is to distinguish satellite attitude error from orbit positional error. The purpose of precision correction estimation is not only to correct the image, but also to extract information about the sensor alignment, which is reflected in the attitude correction parameters. To separate the ephemeris error from the attitude error as much as possible, we should first use the most precise ephemeris data available and correct systematic errors with available models. Second, we should use available a priori information in addition to the observation to cure the ill-condition of the normal equation in statistical estimation.

Let the observation equation be:

$$Y = HX + \varepsilon$$
$$\mathrm{E}[\varepsilon] = 0, \quad \mathrm{Cov}[\varepsilon] = s^2 C \qquad\qquad (4.1)$$

and the a priori information of the parameters be:

$$X = X + \varepsilon_x$$
$$\mathrm{E}[\varepsilon_x] = 0, \quad \mathrm{Cov}[\varepsilon_x] = q^2 C_x \qquad\qquad (4.2)$$

then the normal equation for the Best Linear Unbiased Estimate (BLUE) of X (see Reference 14), $X^\wedge$, is:

$$((1/s^2)H'WH + (1/q^2)W_x)X^\wedge = (1/s^2)H'WY + (1/q^2)W_xX \qquad\qquad (4.3)$$

where W and W_x are weight matrices:

$$W = C^{-1};$$
$$W_x = C_x^{-1} \qquad\qquad (4.4)$$

The covariance matrix of $X^\wedge$ is:

$$\mathrm{Cov}[X^\wedge] = ((1/s^2)H'WH + (1/q^2)W_x)^{-1} \qquad\qquad (4.5)$$

Usually, however, Cov[ε] and Cov[ε_x] cannot be exactly known. In the case of ground control points, for example, the position error involves many factors like base map error and human marking error, etc. If unknown scale factors s^2 and q^2 exist, we can still obtain the WLS estimate from the normal equation:

$$(H'WH + W_x)X^\wedge = H'WY + W_xX \qquad\qquad (4.6)$$

In such a case, the inverse of the normal matrix cannot be taken directly as the Cov[X^]. Factors s^2 and q^2 should be estimated with appropriate variance component estimation from the residual of the solution of Equation 4.6. The weighted residual square summation can be calculated as:

$$V'WV = Y'WY - 2 X^{\wedge}{}'M + X^{\wedge}{}' NX^{\wedge} \tag{4.7}$$

$$V_x' W_x V_x = X'W_x X - 2 X^{\wedge}{}'W_x X + X^{\wedge}{}'W_x X^{\wedge} \tag{4.8}$$

where:

$$V = Y - HX^{\wedge} \tag{4.9}$$

$$V_x = X - X^{\wedge} \tag{4.10}$$

$$N = H'WH \tag{4.11}$$

$$M = H'WY \tag{4.12}$$

3.1.5.3.7.5   Estimating Weight Factor for Observation and A Priori Information

When the factors s^2 and q^2 are appropriately estimated, the weight matrices W and W_x should be correspondingly corrected by factors $1/s^2$ and $1/q^2$, respectively, and Equation 4.6 should be resolved with the new weight matrices. In the new solution, information from the observation and the a priori information are appropriately combined and the $(H' WH + W_x)^{-1}$ is the Cov[X^].

One of the estimates of s^2 and q^2 is the Helmert type estimate. For the problem here, the equation for the estimate can be derived following Helmert's variance component analysis (see Reference 15):

$$E s^2 + D q^2 = V'WV \tag{5.1}$$

$$D s^2 + G q^2 = V_x'W_x V_x \tag{5.2}$$

where:

$$E = n - 2\, tr\{Q N\} + tr\{Q N Q N\} \tag{5.3}$$

$$G = m - 2\, tr\{Q W_x\} + tr\{Q W_x Q W_x\} \tag{5.4}$$

$$D = tr\{Q N Q Wx\} \tag{5.5}$$

$$Q = (H' W H + W_x) \tag{5.6}$$

n = number of observations
m = number of parameters

Equations 5.1 and 5.2 do not guarantee positive solutions of s^2 and q^2. In some cases, especially for a small s^2 and q^2, noise can drive the solution negative. Another type of estimate, the iterative Maximum Likelihood Estimate (MLH), guarantees positive solution (see Reference 16), though the estimates s^2 and q^2 may not be statistically unbiased. The MLH solution is obtained by iteratively solving Equation 4.6 and:

$$s^2 = V'WV / n \tag{5.7}$$

$$q^2 = V_x'W_xV_x / m \tag{5.8}$$

$$W = W / s^2 \tag{5.9}$$

$$W_x = W_x / q^2 \tag{5.10}$$

until s^2 and q^2 converge.

The weight factor estimates procedure is very sensitive to the presence of blunders in the input data and therefore should not be used until the outlier rejection procedure (see 3.1.5.3.7.7) has been performed.

3.1.5.3.7.6   Propagating the Parameter Errors into Look Vector Error

The above solution provides an estimate of the corrections to the ephemeris and attitude data, as well as to their covariance matrix. The covariance information can be used as a measure of precision for assessing the alignment errors of the sensor system. It can also be propagated to any pixel in the scene to evaluate the pixel location error after the precision correction.

Given the sample time and along-scan look angle of a pixel, the coefficients h1 and h2 can be calculated for α and β according to equations 3.6 and 3.7. The variance of α and β are then calculated as:

$$\sigma_\alpha^2 = h_1 \, Cov[X^\wedge]h_1' \tag{6.1}$$

$$\sigma_\beta^2 = h_2 \, Cov[X^\wedge]h_2' \tag{6.2}$$

respectively. These are the variance of the pixel location in sample and line directions due to the uncertainty of the estimated precision correction parameters. They are in angles, but easily can be converted into Instrument Field of View (IFOV), according to the sensor system specifications.

3.1.5.3.7.7   Precision Correction Outlier Detection Algorithm Background

Outlier detection for the IAS precision correction solutions seeks to identify GCPs that are likely to be in error due to miscorrelation. This is done by analyzing the GCP residuals,

taking into account the relative importance of the GCP as reflected in the precision solution normal equation matrix (see References 36 and 37).

Definitions:

A = matrix of coefficients (partial derivatives) relating parameters to observations
$\underline{\theta}$ = parameter vector
$\underline{X}$ = observation vector
$\underline{V}$ = residual vector
C = observation covariance matrix
n = the number of observations
p = the number of parameters
A is n x p, $\underline{\theta}$ is p x 1, $\underline{X}$ and $\underline{V}$ are n x 1, and C is n x n

Observation equation:

$$A\underline{\theta} = \underline{X} - \underline{V}$$

$$\underline{X} = \underline{X}_{true} + \underline{E}$$

where E = error vector $\sim G(\underline{0}, C)$

$$A\underline{\theta}_{true} = \underline{X}_{true}$$

where $\underline{\theta}_{true}$ is the "true" parameter vector

$$A\underline{\theta} = \underline{X}_{true} + \underline{E} - \underline{V}$$

therefore:

$$\underline{V} = \underline{E} \text{ if } \underline{\theta} = \underline{\theta}_{true}$$

Minimum variance parameter estimate:

$$\underline{\theta}' = [A^T C^{-1} A]^{-1} A^T C^{-1} \underline{X}$$

Estimated residual error:

$$\underline{V}' = \underline{X} - A[A^T C^{-1} A]^{-1} A^T C^{-1} \underline{X}$$

Define projection matrix P:

$$P = A[A^T C^{-1} A]^{-1} A^T C^{-1}$$

This matrix projects the observation vector into the parameter subspace (the column space of A).

This projection is only orthogonal if C has the special structure described below.

Substituting:

$$\underline{V}' = \underline{X} - P\underline{X} = [I - P]\underline{X}$$

$[I - P]$ projects $\underline{X}$ into the parameter null space.

Looking at the error estimate $\underline{V}'$:

$$\underline{V}' = [I - P]\underline{X} = [I - P][\underline{X}_{true} + \underline{E}] = [I - P]\underline{X}_{true} + [I - P]\underline{E}$$

but $[I - P]\underline{X}_{true} = \underline{0}$, since $\underline{X}_{true}$ lies entirely within the parameter subspace.

Therefore:

$$\underline{V}' = [I - P]\underline{E} = \underline{E} - P\underline{E}$$

Some comments about $\underline{V}'$ and $\underline{E}$:

1. For a given precision solution, the elements of $\underline{E}$ are not random variables; they are realizations of random variables.
2. $\underline{V}'$ is an estimate of the actual (realized) error $\underline{E}$, which includes an estimation error equal to $P\underline{E}$.
3. We cannot exactly recover $\underline{E}$ from $[I - P]^{-1}\underline{V}'$ because $[I - P]$ is singular (it is an n x n matrix of rank n–p).
4. We can attempt to predict how accurate our estimate ($\underline{V}'$) of $\underline{E}$ is likely to be by looking at the estimation error $\underline{R} = P\underline{E}$.
5. Since we want the predicted accuracy to apply in general, we treat $\underline{R}$ as a random vector that is a function of another random vector $\underline{E}$.

Expected value: $E[\underline{R}] = E[P \underline{E}] = P E[\underline{E}] = P \underline{0} = \underline{0}$

Variance: $E[\underline{R} \underline{R}^T] = E[P \underline{E} \underline{E}^T P^T] = P E[\underline{E} \underline{E}^T] P^T = P C P^T$

Special structure of observation covariance matrix for IAS precision correction:

$$C = \sigma^2 I$$

since the observation errors are realizations of independent and identically distributed zero mean Gaussian random variables with variance σ^2.

Substituting into the equation for P yields:

$$P = A[(1/\sigma^2)A^T I A]^{-1} A^T I (1/\sigma^2) = A\sigma^2 [A^T A]^{-1} A^T (1/\sigma^2) = A[A^T A]^{-1} A^T$$

and the equation for the variance of $\underline{R}$ is:

$$E[\underline{R}\,\underline{R}^T] = \sigma^2\, P\, I\, P^T = \sigma^2\, P$$

noting that $P^T = P$ and $P\,P = P$

therefore:

$$\underline{R} \sim G(\underline{0},\, \sigma^2\, P)$$

For a particular component of $\underline{R}$ r_i:

$$E[r_i] = 0$$
$$E[r_i^2] = \sigma^2\, p_{ii}$$

where p_{ii} is the ith diagonal component of P.

Looking at the equation for P, we see that:

$$p_{ii} = \underline{A_i}^T [A^T A]^{-1} \underline{A_i}$$

where $\underline{A_i}^T$ is the ith row of A.

Considering a particular component of the residual error vector $\underline{V}$':

$$v_i = e_i - r_i$$

where e_i is the corresponding component of the observation error vector, so v_i is an unbiased estimate of e_i with variance $\sigma^2\, p_{ii}$.

3.1.5.3.7.8 Outlier Detection

If we know what e_i is, then we can test it against a probability threshold derived from its standard deviation, σ, to determine if it is likely to be an outlier. Instead of e_i, however, we have v_i, which includes the additional error term r_i. Including the additional estimation error in the threshold computation leads to:

$$\sigma_v{}^2 = \sigma^2 + \sigma^2\,p_{ii}$$

where σ^2 is the term due to the actual error variance, and $\sigma^2\,p_{ii}$ is the term due to the estimation error variance.

This may seem like cheating, since e_i and r_i are not independent for a given realization. In fact:

$$E[\,v_i{}^2\,] = E[\,(e_i - r_i)^2\,] = E[\,e_i{}^2 - 2e_i r_i + r_i{}^2\,]\ \text{and}\ r_i = \Sigma_j\,p_{ij}\,e_j$$
$$E[\,v_i{}^2\,] = \sigma^2\,(1 - p_{ii})$$

It is tempting to use $v_i\,/\,(1 - p_{ii})^{1/2}$ for e_i in the outlier test (or, equivalently, to test v_i against a threshold based on $\sigma^2\,[1 - p_{ii}]$), but this becomes dangerous as p_{ii} approaches 1. The factor p_{ii} can be interpreted as a measure of the uniqueness of or as the information content of the ith observation. As p_{ii} approaches 1, the ith observation lies almost entirely within the parameter subspace, which implies that it is providing information to the solution that the other observations are not. Such "influential" observations can be identified from the structure of the coefficient matrix A without reference to the observation residuals. Attempting to use $1/(1 - p_{ii})^{1/2}$ to rescale the residual v_i to better approximate e_i in a sense punishes this observation for being important. Instead, we view p_{ii} as a measure of how poor an estimate of the actual error e_i, the residual v_i, is and ignore the fact that v_i tends to underestimate e_i. We therefore use $\sigma_v{}^2 = \sigma^2\,(1 + p_{ii})$, as shown above, to construct the outlier detection threshold.

One remaining problem is that we do not know exactly what σ^2 is and must estimate it from the observation residuals. This is done by scaling the a priori observation variance using the variance of unit weight computed in the precision solution. The fact that we are using an estimated variance to establish our outlier detection threshold modifies the algorithm in two ways: (1) we compensate for the fact that removing a point as an outlier alters the computation of the variance of unit weight by removing one residual and reducing the number of degrees of freedom, and (2) we base the detection threshold computation on Student's t-distribution rather than the Gaussian distribution.

The variance of unit weight is computed as:

$$\text{var}_0 = \underline{V}^T C^{-1} \underline{V}/(n - p) = \underline{V}^T\underline{V}/\sigma_0{}^2(n - p) = \Sigma v_j{}^2/\sigma_0{}^2(n - p)$$

where:

n = number of observations
p = number of parameters
$\sigma_0{}^2$ = the a priori variance

The estimated variance is:

$$\text{var} = \text{var}_0\, \sigma_0{}^2 = \Sigma v_j{}^2/(n - p)$$

Removing the kth observation makes this:

$$\text{var}_k = (\Sigma v_j{}^2 - v_k{}^2)/(n - 1 - p) = (n - p)/(n - p - 1) * (\Sigma v_j{}^2 - v_k{}^2)/(n - p)$$
$$\text{var}_k = (n - p)/(n - p - 1) * \text{var} - v_k{}^2/(n - 1 - p)$$

To normalize the kth residual, we divide it by the estimated standard deviation $\sigma' = (\text{var})^{1/2}$:

$$w_k = v_k / \sigma'$$

We can rescale this normalized residual to reflect the removal of this observation from the variance estimate without actually having to compute a new variance:

$$w_k' = v_k / \sigma_k' = w_k\, \sigma'/\sigma_k' = w_k\, (\text{var}/\text{var}_k)^{1/2}$$
$$\text{var}/\text{var}_k = 1 / [(n - p)/(n - p - 1) - v_k{}^2/\text{var}\, (n - p - 1)] = (n - p - 1)/(n - p - v_k{}^2/\text{var})$$
$$\text{var}/\text{var}_k = (n - p - 1)/(n - p - w_k{}^2)$$

noting that $v_k{}^2/\text{var} = w_k{}^2$

$$w_k' = w_k\, [(n - p - 1)/(n - p - w_k{}^2)]^{1/2}$$

Finally, we include the $(1 + p_{kk})$ factor discussed above, and our normalized and reweighted residual becomes:

$$w_k' = w_k\, [(n - p - 1)/(1 + p_{kk})(n - p - w_k{}^2)]^{1/2}$$

where:

$$w_k = v_k / \sigma'$$

This normalized and reweighted residual is compared against a probability threshold computed using Student's t-distribution with $(n - p)$ degrees of freedom.

3.1.5.3.7.9   Inverse Mapping

For the precision correction process, the information needed from the systematic ground control point includes the satellite position and velocity vectors in Earth-centered inertial coordinates, along with the latitude and longitude associated with the point. From the reference ground control point, the precision process needs the latitude and longitude, along with the height of the control point above the WGS84 ellipsoid. Following are the steps involved in the process:

1. Use the Geometric Correction Grid to find an input line and sample for an associated output line and sample of the systematic Ground Control Point (GCP) (see the resampling algorithm description).
2. Determine the scan time associated with the systematic GCP.   Since the model truncates to determine the scan number from the input line, the input line has a value of 0.5 added to it so that it can be rounded to the nearest integer line.
3. Use an interpolating polynomial for satellite position and velocity to get the position and velocity vectors for the time associated with the input line and sample of systematic GCP.
4. Call the Projection Transformation Package to convert the projection coordinate of the systematic image GCP to latitude and longitude.
5. Call the Projection Transformation Package to convert the projection coordinate of the base image GCP to latitude and longitude.

3.1.5.3.7.10   Precision Ephemeris Update

The purpose of this algorithm is to update the systematic orbit model with the information from the precision correction solution parameters, so that a precision image can be generated.

Given:

t_{ref} = reference time for the precision correction
x_{ob} = orbit position correction in the Orbit System x (along-track) direction
y_{ob} = orbit position correction in the Orbit System y (cross-track) direction
z_{ob} = orbit position correction in the Orbit System z (altitude) direction
vx_{ob} = orbit velocity correction in the Orbit System x (along-track) direction
vy_{ob} = orbit velocity correction in the Orbit System y (cross-track) direction
vz_{ob} = orbit velocity correction in the Orbit System z (altitude) direction

For each ephemeris data point, the algorithm takes the following ephemeris information:

t_{eph} = time for the ephemeris data point
x = orbit position in the Earth Fixed system x direction
y = orbit position in the Earth Fixed system y direction
z = orbit position in the Earth Fixed system z direction
vx = orbit velocity in the Earth Fixed system x direction
vy = orbit velocity in the Earth Fixed system y direction
vz = orbit velocity in the Earth Fixed system z direction

and performs the following steps:

1. Calculate the orbit position correction in the Orbit system at the time:

    $dx_{ob} = x_{ob} + vx_{ob} * (t_{eph} - t_{ref})$
    $dy_{ob} = y_{ob} + vy_{ob} * (t_{eph} - t_{ref})$
    $dz_{ob} = z_{ob} + vz_{ob} * (t_{eph} - t_{ref})$

2. Calculate the transformation matrix from the Orbit system to the Earth Fixed system at the time.

    a. Compose the transformation matrix, T_{eo}, from the Earth Fixed system to the Orbit system:

        $\underline{W}_z$, unit vector in negative radial direction, pointing to the Earth's center ($\underline{X}$ = {x, y, z} and $\underline{V}$ = {v_x, v_y, v_z}):

        $\underline{W}_z = - \underline{X} / |\underline{X}|$

        $\underline{W}_y$, unit vector in cross-track direction (− angular momentum):

        $\underline{W}_y = \underline{W}_z$ cross $\underline{V} / |\underline{W}_z$ cross $\underline{V}|$

        $\underline{W}_x$, unit vector in along-track direction:

        $\underline{W}_x = \underline{W}_y$ cross $\underline{W}_z$

        $$T_{eo} = \begin{bmatrix} \underline{W}_x(1) & \underline{W}_x(2) & \underline{W}_x(3) \\ \underline{W}_y(1) & \underline{W}_y(2) & \underline{W}_y(3) \\ \underline{W}_z(1) & \underline{W}_z(2) & \underline{W}_z(3) \end{bmatrix}$$

b. Transpose the T_{eo} matrix to get the transformation from the Orbit System to the Earth Fixed system:

$$T_{oe} = \begin{bmatrix} \underline{W}_x(1) & \underline{W}_y(1) & \underline{W}_z(1) \\ \underline{W}_x(2) & \underline{W}_y(2) & \underline{W}_z(2) \\ \underline{W}_x(3) & \underline{W}_y(3) & \underline{W}_z(3) \end{bmatrix}$$

$$\begin{bmatrix} dx \\ dy \\ dz \end{bmatrix} = \begin{bmatrix} \underline{W}_x(1) & \underline{W}_y(1) & \underline{W}_z(1) \\ \underline{W}_x(2) & \underline{W}_y(2) & \underline{W}_z(2) \\ \underline{W}_x(3) & \underline{W}_y(3) & \underline{W}_z(3) \end{bmatrix} * \begin{bmatrix} dx_{ob} \\ dy_{ob} \\ dz_{ob} \end{bmatrix}$$

3. Calculate the orbit position correction in the Earth Fixed system:

4. Calculate the orbit velocity correction in the Earth Fixed system:

$$\begin{bmatrix} dvx \\ dvy \\ dvz \end{bmatrix} = \begin{bmatrix} \underline{W}_x(1) & \underline{W}_y(1) & \underline{W}_z(1) \\ \underline{W}_x(2) & \underline{W}_y(2) & \underline{W}_z(2) \\ \underline{W}_x(3) & \underline{W}_y(3) & \underline{W}_z(3) \end{bmatrix} * \begin{bmatrix} vx_{ob} \\ vy_{ob} \\ vz_{ob} \end{bmatrix}$$

5. Update the orbit position and velocity in the Earth Fixed system:

x = x + *dx*
y = y + *dy*
z = z + *dz*
vx = vx + *dvx*
vy = vy + *dvy*
vz = vz + *dvz*

This algorithm computes the orbit correction for each ephemeris data point in the Earth Fixed system and adds them up to yield the new ephemeris data point information.

3.1.5.3.8   Terrain Correction

The terrain correction algorithm uses elevation data from a digital elevation model (DEM) to generate offsets, which correct for the effect of terrain parallax. These corrections are required only in the along-scan (sample) direction.

To avoid having to call the Projection Transformation Package for every point in the output space, the DEM is processed to the same pixel size, projection, and frame as the precision image, and its elevations are relative to the WGS84 ellipsoid rather than mean sea level or some other vertical datum. It is also assumed that the precision corrections have been applied to the forward model. The process is as follows:

1. Read the DEM and determine minimum and maximum elevations present in the scene.

2. Determine the location of nadir in terms of the raw space input sample.   The secant method of root finding is used to determine the sample location at which the y-component of the line-of-sight (LOS) vector in orbital coordinates goes to zero for each scan. It stores them in an array. This segment also stores the latitude and magnitude of the satellite position vector (in meters) for the center of the scene.
3. Using the minimum and maximum elevations found in step 1, determine the sample displacement due to terrain for every (off-nadir-sample, elevation) pair. Off-nadir-samples should range from 0 to 3,600 (or some number that guarantees coverage of all possible input samples). Elevation should be in meters and range from minimum to maximum in the scene. The algorithm steps elevation in increments of 1 meter.
4. For every line, sample pair in the output image space, the terrain correction algorithm needs the corresponding line, sample in the input space (see 3.1.5.3.6).
5. Given the current output line and sample, the elevation is read from the DEM buffer from step 1.

    Given the corresponding input sample, the distance from nadir in pixels is determined from the calculation of nadir pixels in step 2.

$$off\_nadir\_pixel = in\_sample - nadir\_sample$$

    From the ($|off\_nadir\_pixel|$, elevation) pair determined in steps 6 and 7, the value of the terrain offset is looked up from the table generated by step 3.

    The next step in terrain correction (applied in the resampler) is to adjust the input sample locations by the terrain offset and resample.

The algorithm assumes that a DEM is available in the same projection, datum, and framing as the output space, as defined by the precision corrected grid and that it has been resampled to the same pixel size. It also assumes that for every line, sample in output space, the corresponding line, sample can be found in input space. The offsets due to terrain are applied in the resampler.

This algorithm was investigated using the 1°x 1° (3 arc-second) Digital Terrain Elevation Data Level 1 (DTED-1), available via anonymous file transfer protocol (FTP) from the National Center for Earth Resources Observation and Science (EROS). Slightly better accuracy could be achieved using higher spatial resolution 7.5 x 7.5 arc-minute (30-meter) USGS DEM data; vertical accuracy is better than 15-meter Root Mean Square Error (RMSE).

3.1.5.3.8.1   Terrain Correction Algorithm

The terrain correction algorithm generates a table of terrain-induced sample offsets, corresponding to (off-nadir-sample, elevation) pairs, as generated in step 3 above. The method used for determining these terrain corrections comes from Reference 22. Following is a summary of that document. A schematic of the geometry is shown in Figure 3-33.

Given distance in pixels of a sample in input space from the nadir pixel location, the ground range, S, is given by:

S = pixel_size * off_nadir_pixel

where pixel_size is the nominal pixel size in meters for the given band.

This is associated with an Earth-centered angle, s:

s = S/Re

where Re is the radius of the Earth reference ellipsoid at the scene center latitude.

The magnitude and direction of the line-of-sight vector relative to nadir can be found as:

LOS = sqrt(Re^2 + (Re + Alt) ^{2} – 2*Re(Re + Alt)*cos(s))
d = asin(Re*sin(s)/LOS)

where Alt is the altitude of the satellite above the ellipsoid.

Looking at Figure 3-33, elevation h at location S causes the information at the top of the hypothetical mountain to be mapped to the location S + dS on the ellipsoid used by the forward sensor model. What the terrain correction does is determine the offset dS, so that when the time comes to resample, we can grab the information projected on the ellipsoid at location S + dS and move it back to the mountaintop at location S. This is done by first determining the angular difference between the LOS to a spot on the ellipsoid below the terrain feature and the LOS' to the projected location of that feature on the ellipsoid.

dd = atan{(Re+Alt) * sin(d) * (1–(Re+h)/Re) / [(Re+h) * sqrt(1–(Re+Alt)^2 *
        sin^2(d)/Re^2) –(Re+Alt)*cos(d)]}
z" = asin[(Re+Alt) * sin(d+dd)/Re]
ds = z" – s – (d+dd)
dS = Re * ds
terrain_offset = dS / pixel_size
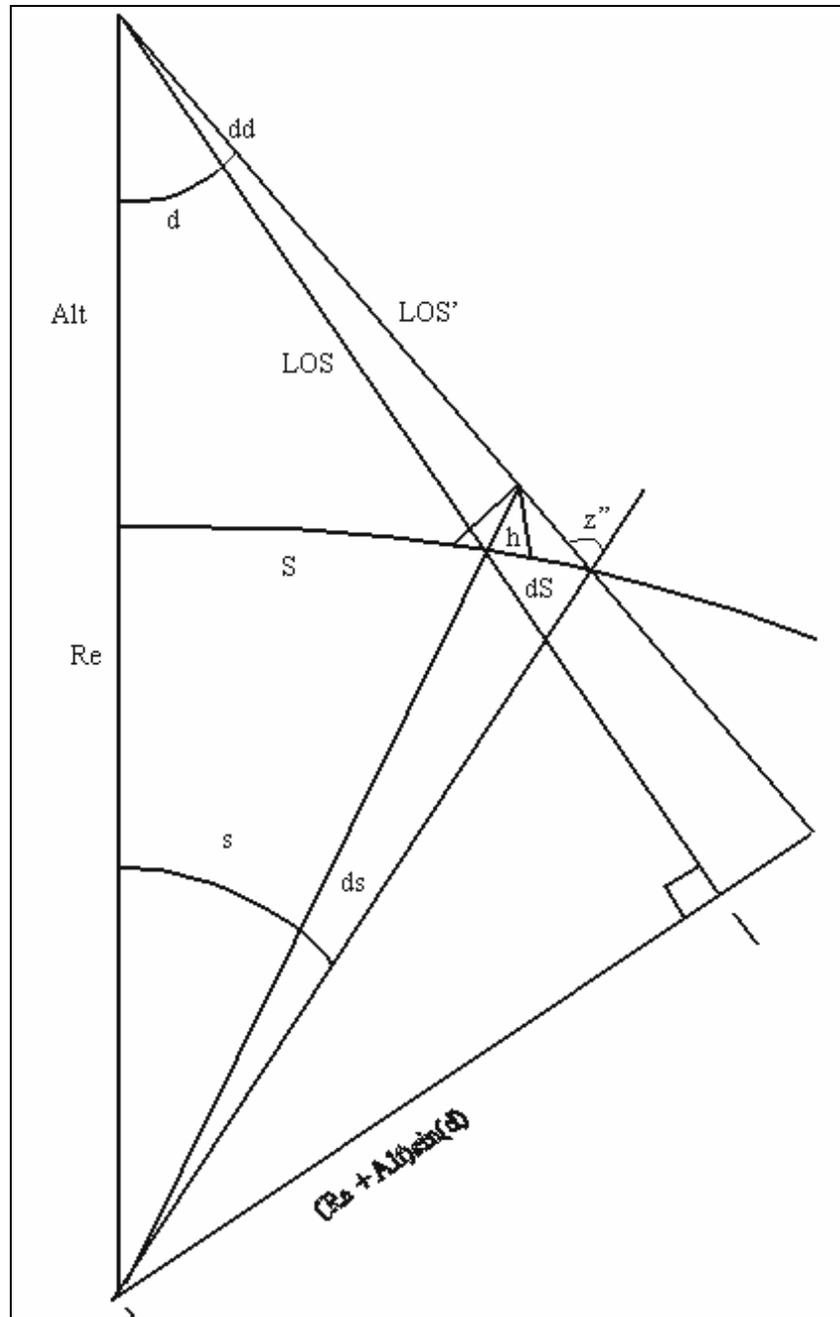
Figure 3-33. Terrain Correction Geometry

Then for resampling, rather than getting data from input(sample), the data for the location in output space is taken from input (sample + terrain_offset * (sign of off_nadir_pixel)).

The assumption of a constant satellite altitude above the geoid for one Landsat scene was tested with an absolute worst-case scenario. A terrain feature with an elevation of 8,850 meters (the height of Mt. Everest) was placed near the equator at the end of a scan line. The orbit was modeled using the highest eccentricity allowed within specifications and in the vicinity of the greatest rate of change of orbit radius. In this case, the

maximum error introduced was only ~0.03 panchromatic pixels. This is approximately the same size as the resampling increment, 0.03125 pixels. In a test case using Landsat 5 data from an actual test site (path 28, row 30), the error from the altitude assumption for an elevation of 500 meters at the end of scan was ~0.001 pixels. For these reasons, the assumption of a constant satellite altitude over a scene is considered valid.

3.1.5.4   Geometric Calibration Algorithms

3.1.5.4.1   Sensor Alignment Calibration

The purpose of this algorithm is to use a sequence of precision correction solutions, including both correction parameter estimates and estimated covariance information, to estimate the underlying ETM+ instrument to Landsat 7 navigation reference base alignment. A Kalman filter implementation is used to isolate the systematic alignment trend from the attitude and ephemeris knowledge error, which varies from scene to scene.

This algorithm is one step of the Kalman filter (see Reference 23) for a linear dynamic model (constant rate model). It takes one precision correction solution file (generated from one scene) and updates the estimate of the alignment state parameters. It appends the newest state parameter estimate to the orbit parameter sequence file and attitude parameter sequence file, and also appends an innovation sequence file. This algorithm does not make a decision on whether the alignment is anomalous or whether it should be corrected. This algorithm provides the information (the sequence files) to an analyst who makes that decision.

This algorithm is applied each time a calibration scene is processed through the precision correction software to update the alignment estimate. Operationally, the precision correction processing includes the use of definitive ephemeris data, obtained from the Goddard Space Flight Center (GSFC) Flight Dynamics Facility (FDF), to help separate ephemeris position errors from angular alignment and attitude errors. Error propagation analyses and simulations run with the algorithm prototype software suggest that it requires four or five precision-corrected calibration scenes to drive the alignment knowledge from its pre-launch accuracy of approximately 240 arc-seconds per axis (1 sigma) to the IAS calibrated specification value of 24 arc-seconds per axis.

3.1.5.4.1.1 Algorithm Description

Defining the filter:

The state vector of this filter is a 12 x 1 vector:

	|roll|	roll attitude angle, in microradians (urad)
	|pitch|	pitch attitude angle, in urad
	|yaw|	yaw attitude angle, in urad
	|x|	along-track orbit position error, in meters
	|y|	cross-track orbit position error, in meters
X =	|z|	radial orbit position error, in meters
	|r_dot|	roll angle rate, in urad/day
	|p_dot|	pitch angle rate, in urad/day
	|yaw_dot|	yaw angle rate, in urad/day
	|vx|	velocity in x, in meters/day
	|vy|	velocity in y, in meters/day
	|vz|	velocity in z, in meters/day

See the precision correction algorithm description for the definition of above parameters, but note the units for the rate and velocity terms are different here. In fact, the rate and velocity state vector components represent long-term linear trends in the bias data, rather than the short-term rate of change errors estimated by the precision correction process. The covariance matrix for X is a 12 x 12 matrix P.

The observation vector for this filter is a 6 x 1 vector:

	|roll|	roll attitude angle, in urad
	|pitch|	pitch attitude angle, in urad
Y =	|yaw|	yaw attitude angle, in urad
	|x|	along-track orbit position error, in meters
	|y|	cross-track orbit position error, in meters
	|z|	radial orbit position error, in meters

The covariance matrix for Y is a 6 x 6 matrix R. The observation is from the precision correction solution, but with the velocity and attitude rate parameters discarded. The velocity and rate estimated from precision correction are short-term variations in a scene; what is to be monitored in the alignment calibration is the long-term drift.

The precision correction estimates for the attitude bias terms (roll, pitch, and yaw) are actually estimates of the combination of long-term alignment error (what we are looking for) and short-term alignment and attitude errors. These short-term errors appear to be constant, or nearly constant, biases over time intervals of minutes to hours, but decorrelate over the time interval between calibration scenes (days). These short-term variations thus appear to be random noise superimposed on the alignment bias observations taken from each calibration scene. This manifests itself in the calibration

scene precision correction results as sequences of attitude bias estimates whose dispersion is greater than would be expected based on the a posteriori covariance estimates computed in the precision correction process. Any systematic linear trend present in the attitude error is indistinguishable from alignment and is incorporated into the alignment bias estimated by the Kalman filter.

The random attitude and dynamic alignment errors are, in effect, an additional measurement error vector, added to the estimation error from the precision correction solution, in the Kalman filter formulation. The observation covariance matrix R is therefore the sum of the a posteriori covariance matrix from the precision correction solution and the expected covariance of the "random" attitude and dynamic alignment errors. According to the Landsat 7 System Specification (Reference 5), the expected variance of the additional random attitude and dynamic alignment errors is less than or equal to the sum of the squares of 45 arc-seconds of attitude knowledge error (see Reference 5, 3.7.1.3.9), 8 arc-seconds of dynamic ETM+ to Navigation Base Reference alignment error (see Reference 5, 3.7.1.3.10), and 15 arc-seconds of line of sight to ETM+ reference dynamic alignment error (see Reference 5, 3.7.8.1.16.7). This yields a net random component of 48 arc-seconds. Since these errors are assumed to be uncorrelated with each other, they are added to the diagonal elements of the R matrix, corresponding to the alignment biases.

Although we do not expect to detect any systematic offset in the position bias terms (x, y, and z), they are included because of their high correlation with the attitude biases. This is reflected in the observation covariance matrix, where significant off-diagonal terms exist for x-pitch and y-roll. Any particular precision correction solution resolves the correlation between the parameters by allocating the along-track and across-track errors between the ephemeris and attitude parameters based on their a priori weights. Thus, some of the systematic alignment bias could end up allocated to the ephemeris correction terms. Over multiple precision correction solutions, the net ephemeris bias should be very close to zero, so we can establish small a priori state variances for the ephemeris bias terms and allow the Kalman filter to reallocate systematic trends in the ephemeris corrections back to the alignment parameters. In practice, since accurate post-pass ephemeris is available for the calibration test scenes, most of the correction is allocated to the attitude terms in the precision correction solutions anyway.

The dynamic model used for this filter is the constant rate (velocity) model:

$$\mathbf{X}_k = \mathbf{F}(t_k, t_{k-1})\mathbf{X}_{k-1}$$

where the state transition matrix is:

$$\mathbf{F}(t_k, t_{k-1}) = \begin{bmatrix} \mathbf{I} & dt\,\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$$

I is a 6 x 6 identity matrix, and dt = t_k − t_{k-1} is the time difference between epoch t_{k-1} and t_k.

The observation equation is:

Y = H*X + v;

where H = [I 0], and v is the net measurement error with covariance R.

Process noise can be added with constant covariance matrix Q. Q is diagonal with 12 components. The process noise is initially assumed to be zero. After several precision correction solution sets are available for analysis, the innovation sequence and a posteriori state covariance can be assessed to decide whether process noise should be added to better model the dynamic behavior observed in the data.

3.1.5.4.1.1.1   Procedure

For each run, this algorithm performs the steps described in the following section.

1. Setup option

    Read in the input option file to define the input and output file names, options, and process noise variance.

2. Initialization

    Read in the previous epoch time t_p, state vector X_p, and covariance matrix P_p. For a regular run, they are read from the output state file of the previous run. For the first run, they are read from an initial observation file, which contains the a priori knowledge of the state vector (alignment terms are zero +/− 240 arc-seconds, ephemeris terms are zero +/− 1 meter).

3. Read and preprocess an observation as follows:

    a. Extract the observation data from the precision correction solution

        Read in the current epoch time t_c, precision correction observation Z, and covariance matrix R. They are read from a solution file output by the precision correction algorithm. The precision correction solution file contains both bias and rate correction terms, whereas the Kalman filter observation uses only the bias components. The terms corresponding to roll, pitch, yaw, x, y, and z bias are extracted from the precision correction solution vector

and covariance matrix to assemble Y and R, respectively. The additional observation noise due to scene-to-scene attitude and dynamic alignment error is accounted for by adding the expected error variance (48 arc-seconds)^2 to the diagonal elements of R corresponding to the roll, pitch, and yaw bias correction parameters.

b. Convert the attitude observation to an absolute pointing error observation.

The precision correction solution contains updates to the alignment knowledge from the Calibration Parameter File (CPF) used to process the test scene. We want the Kalman filter to model the absolute alignment angles so that observations processed with different versions of the CPF can be combined. To make this possible, we must construct the composite CPF alignment/precision correction angles for input to the Kalman filter.

The precision correction roll (r), pitch (p), and yaw (y) update values can be composed into an alignment correction rotation matrix as follows:

$$\Delta A = \begin{bmatrix} \cos(p)\cos(y) & \sin(r)\sin(p)\cos(y) + \cos(r)\sin(y) & -\cos(r)\sin(p)\cos(y) + \sin(r)\sin(y) \\ -\cos(p)\sin(y) & -\sin(r)\sin(p)\sin(y) + \cos(r)\cos(y) & \cos(r)\sin(p)\sin(y) + \sin(r)\cos(y) \\ \sin(p) & -\sin(r)\cos(p) & \cos(r)\cos(p) \end{bmatrix}$$

The Calibration Parameter File (CPF) alignment matrix is defined as the rotation from the Navigation Base Reference (NBR) coordinate system to the ETM+ sensor coordinate system. To convert a look vector constructed in ETM+ sensor coordinates (X_{ETM}) to the orbital coordinate system (X_{orb}), where the precision correction solution is performed, the following rotations must be performed:

$$X_{orb} = T^T_{att} A^T X_{ETM}$$

where:

T_{att} is the attitude rotation matrix, and A is the alignment matrix.

The transpose of A is used, assuming that A is an orthogonal rotation matrix so that its transpose is the same as its inverse.

The precision correction solution computes updates to the composite $T_{att} A^T$ rotation to best fit the observed ground control. This update takes the following form:

$$X_{orb} = \Delta A \, T^T_{att} A^T X_{ETM}$$

where:

ΔA is the alignment correction matrix defined above

For small rotations, the ΔA and T_{att} commute to a good approximation. Using the maximum off-nadir attitude pointing of 1,080 arc-seconds (900 arc-seconds maximum off-nadir commanded pointing per the Landsat 7 System Specification, 3.7.1.3.8.2, plus 180 arc-seconds of pointing error per 3.7.1.3.8) and the maximum expected alignment error of 720 arc-seconds (ibid., 3.7.1.3.11), the error in commuting these rotations is less than 0.01 arc-seconds. This allows the above equation to be rewritten as:

$$X_{orb} = T^T_{att} \, \Delta A \, A^T \, X_{ETM}$$

The composite rotation $A' = \Delta A \, A^T$ captures the net measured alignment offset. We extract roll, pitch, and yaw angles from this composite matrix using the following equations, where a'_{ij} indicates the element from the ith row and jth column of A':

$$roll' = \tan^{-1}(-a'_{32}/a'_{33})$$
$$pitch' = \sin^{-1}(a'_{31})$$
$$yaw' = \tan^{-1}(-a'_{21}/a'_{11})$$

These composite angles are inserted into the observation vector Y, replacing the precision correction output values.

4. Propagate X_p and P_p to the current epoch time t_c.

Calculate:

$$dt = t_c - t_p$$
$$X_c = F(t_c,t_p) * X_p$$
$$P_c = F(t_c,t_p) * P_p * F^T(t_c,t_p)$$

Because of the special structure of F, the computation can be simplified if we divide the X vector into two blocks:

$$X = \begin{vmatrix} X_1 \\ X_2 \end{vmatrix}$$

and P into four sub-matrices:

$$P = \begin{vmatrix} P_{11} & P_{12} \\ ---- + ---- \\ P_{21} & P_{22} \end{vmatrix}$$

Then:

$X_{c1} = X_{p1} + dt^*X_{p2}$
$X_{c2} = X_{p2}$
$P_{c11} = P_{p11} + dt^*P_{p12} + dt^*P_{p21} + dt^*dt^*P_{p22}$
$P_{c12} = P_{p12} + dt^*P_{p22}$
$P_{c21} = P_{p21} + dt^*P_{p22}$
$P_{c22} = P_{p22}$

And add the process noise if that option is selected:

$P_c = P_c + Q$

5. Calculate the Kalman gain matrix K:

$K = P_c^*H^T{}^*[H^*P_c^*H^T + R]^{-1}$

Because of the structure of H:

$H^*P_c^*H^T = P_{c11}$

and:

$$P_c^*H^T = \begin{vmatrix} P_{c11} \\ \\ P_{c21} \end{vmatrix}$$

which gives the Kalman gain matrix the following structure:

$$K = \begin{vmatrix} K_1 \\ \\ K_2 \end{vmatrix} = \begin{vmatrix} P_{c11} \\ \\ P_{c21} \end{vmatrix} * [P_{c11} + R]^{-1}$$

6. Update the current state with the filtered innovation to create a new estimate and update the new filtered state covariance:

$X_n = X_c + K^*(Y - H^*X_c) = X_c + K^*(Y - X_{c1})$
$P_n = P_c - K^*H^*P_c$

which reduces to:

$$P_n = \quad P_c \; - \; \begin{vmatrix} K_1 {*} P_{c11} & K_1 {*} P_{c12} \\ & \\ K_2 {*} P_{c11} & K_2 {*} P_{c12} \end{vmatrix}$$

7. Append the orbit elements in X_n to the orbit sequence file.

8. Append the attitude elements in X_n to the attitude sequence file.

9. Append the innovation $Y - H{*}X_c$ to the innovation sequence file.

10. Write out X_n and P_n to the new state file.

11. Update the Calibration Parameter File alignment matrix.

Periodically, the analyst may decide that the alignment knowledge has improved sufficiently to justify modifying the Calibration Parameter File. This typically occurs after the first five or six calibration scenes have been processed post-launch and then at least once every quarter after the initial on-orbit calibration. The update is performed using the most current roll, pitch, and yaw correction estimates in the attitude sequence file.

The current filter estimates of roll (r), pitch (p), and yaw (y) are composed into a net alignment correction rotation matrix (as was done for the differential corrections above) as follows:

$$A_{net} \; = \; \begin{bmatrix} \cos(p)\cos(y) & \sin(r)\sin(p)\cos(y) \,+\, \cos(r)\sin(y) & -\cos(r)\sin(p)\cos(y) \,+\, \sin(r)\sin(y) \\ -\cos(p)\sin(y) & -\sin(r)\sin(p)\sin(y) \,+\, \cos(r)\cos(y) & \cos(r)\sin(p)\sin(y) \,+\, \sin(r)\cos(y) \\ \sin(p) & -\sin(r)\cos(p) & \cos(r)\cos(p) \end{bmatrix}$$

The net alignment rotation A_{net} is the transpose of the new alignment matrix, so the updated alignment matrix for the CPF is computed as follows:

$$A_{new} = A_{net}{}^{T}$$

Once this update is incorporated into the CPF, subsequent calibration scenes are processed with the improved alignment knowledge. Since this alignment knowledge is combined with the precision correction observation, as described in 3.1.5.3.7.7, for input into the Kalman filter, the filter need not be reset after CPF updates. The state covariance also does not need to be reset, since it accurately reflects the confidence of the current alignment knowledge.

After a sufficient number of precision correction results have been collected, an analyst may come up with some idea about the process noise by looking at the innovation sequence and the output filter covariance. The process noise sigma can be fed back to the filter through the input option file. Without the additional process noise, the filter may begin to diverge after a certain time (the filtered state covariance begins to grow). Tuning

the process noise is a difficult part of Kalman filtering practice, and some experimentation is needed to optimize the algorithm.

3.1.5.4.2   Focal Plane Calibration

3.1.5.4.2.1   Focal Plane—Band Placement Calibration Algorithm Description

The objective of this algorithm is to estimate updates to the focal plane locations of the eight ETM+ band centers to improve band-to-band registration as specified in Reference 3.

Input to this algorithm includes the band-to-band comparison point correlation results from the band-to-band characterization algorithm, with outliers detected and removed as described in 3.1.5.5.1. They are used in conjunction with the resampling grid file (used to produce the Level 1Gs test image) as input to the band-to-band calibration process.

3.1.5.4.2.2   Algorithm

The band location parameter vector is of the form:

$$\mathbf{X} = \begin{bmatrix} \Delta\mathbf{b}_{1x} \\ \Delta\mathbf{b}_{1y} \\ \dots \\ \Delta\mathbf{b}_{8x} \\ \Delta\mathbf{b}_{8y} \end{bmatrix}$$

where:

Δb_{ix} = band offset in the focal plane x (along-scan) direction for band i, in microradians

Δb_{iy} = band offset in the focal plane y (across-scan) direction for band i, in microradians

The band-to-band characterization test point correlation output data are assembled, point by point, to construct a band-to-band calibration measurement vector:

$$
Y = \begin{bmatrix} m_{21P1} \\ m_{21L1} \\ \ldots \\ m_{21PN} \\ m_{21LN} \\ m_{31P1} \\ \ldots \\ m_{81LN} \\ m_{32P1} \\ \ldots \\ m_{87LN} \end{bmatrix}
$$

where:

m_{jiPk} = observed offset for point k, in output space pixels, between band j and band i (reference band) = $pixel_{jk} - pixel_{ik}$

m_{jiLk} = observed offset for point k, in output space lines, between band j and band i (reference band) = $line_{jk} - line_{ik}$

The parameter vector and measurement vector are related by the following observation equation:

$$
A_{jik} X = \begin{bmatrix} 0 & \ldots & -\dfrac{dp}{dx_{ik}} & -\dfrac{dp}{dy_{ik}} & \ldots & 0 & \ldots & \dfrac{dp}{dx_{jk}} & \dfrac{dp}{dy_{jk}} & \ldots & 0 \\ 0 & \ldots & -\dfrac{dl}{dx_{ik}} & -\dfrac{dl}{dy_{ik}} & \ldots & 0 & \ldots & \dfrac{dl}{dx_{jk}} & \dfrac{dl}{dy_{jk}} & \ldots & 0 \end{bmatrix} X
$$

$$
= \begin{bmatrix} m_{jiPk} \\ m_{jiLk} \end{bmatrix} + V_{jik} = Y_{jik} + V_{jik}
$$

Where V_{jik} is the 2 x 1 vector of measurement residuals for point k, assumed to be zero mean Gaussian errors with covariance matrix C_{jik}. Typically, the C_{jik} would be assumed to be the same for all k, so $C_{jik} = C_{ji}$.

$$
C_{ji} = \begin{bmatrix} \sigma_p^2 & 0 \\ 0 & \sigma_l^2 \end{bmatrix} \quad \sigma_p \approx \sigma_l \approx 0.25
$$

The partial derivatives in the observation coefficient matrix A_{jik} are computed numerically using the resampling grids for bands j and i, from the input grid file, as follows:

For band i partials:

If band i is resolution adjusted for correlation with a lower resolution band, adjust the measured pixel, line to match the grid:

If $I = 8$ (i.e., this band is panchromatic):

$$p_{ik} = 2*p_{ik} - 1$$
$$l_{ik} = 2*l_{ik} - 1$$

If $j = 6$ (i.e., the other band is thermal):

$$p_{ik} = 2*p_{ik} - 1$$
$$l_{ik} = 2*l_{ik} - 1$$

Use $GRID_i$ to inverse map the test point pixel, line coordinates p_{ik}, l_{ik} to input image space pixel, line coordinates s_{ik}, d_{ik}.

Add a small epsilon value, e_s, to s_{ik}.

Use $GRID_i$ to map $(s_{ik} + e_s, d_{ik})$ to output space (p_{ix}, l_{ix}).

Add a small epsilon value, e_d, to d_{ik}.

Use $GRID_i$ to map $(s_{ik}, d_{ik} + e_d)$ to output space (p_{iy}, l_{iy}).

Compute the partials numerically, dividing by the band IFOV in microradians to rescale the partial derivatives from output pixels per input pixel to output pixels per microradian:

$$dp/dx_{ik} = (p_{ix} - p_i)/e_s/IFOV_i$$
$$dl/dx_{ik} = (l_{ix} - l_i)/e_s/IFOV_i$$
$$dp/dy_{ik} = (p_{iy} - p_i)/e_d/IFOV_i$$
$$dl/dy_{ik} = (l_{iy} - l_i)/e_d/IFOV_i$$

If band i has been resolution adjusted for correlation with a lower resolution band, adjust the partial derivative to match the actual measurements:

If $I = 8$ (i.e., this band is panchromatic):

$$dp/dx_{ik} = (dp/dx_{ik})/2$$
$$dl/dx_{ik} = (dl/dx_{ik})/2$$
$$dp/dy_{ik} = (dp/dy_{ik})/2$$
$$dl/dy_{ik} = (dl/dy_{ik})/2$$

If j = 6 (i.e., the other band is thermal):

$$dp/dx_{ik} = (dp/dx_{ik})/2$$
$$dl/dx_{ik} = (dl/dx_{ik})/2$$
$$dp/dy_{ik} = (dp/dy_{ik})/2$$
$$dl/dy_{ik} = (dl/dy_{ik})/2$$

The same procedure is used with the grid file for band j, GRID $_j$, to yield the band j partials: dp/dx_{jk}, dl/dx_{jk}, dp/dy_{jk}, and dl/dy_{jk}. In this case, i and j are interchanged in the above procedure.

The normal equations can be constructed directly from a sequence of observation equations:

$$N = \Sigma\ A_{jik}{}^T\ C_{ji}{}^{-1}\ A_{jik} \qquad \text{summed over all k and all i, j combinations}$$
$$L = \Sigma\ A_{jik}{}^T\ C_{ji}{}^{-1}\ Y_{jik} \qquad \text{summed over all k and all i, j combinations}$$

With only the band difference observations, this system of equations is indeterminate, since the solution is insensitive to the addition of constant offsets to the Δb terms. To stabilize the solution, we add another observation requiring the band offset for band 8 (panchromatic) to be zero in each direction. Band 8 is constrained because it is used for scan mirror calibration and sensor alignment calibration, making it the geometric calibration reference band. This constraint is treated as another observation of the form:

$$\mathbf{A}_{00} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{Y}_{00} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \mathbf{C}_{00} = \begin{bmatrix} \sigma_0^2 & 0 \\ 0 & \sigma_0^2 \end{bmatrix} \qquad \sigma_0 \approx 0.01\,\text{urad}$$

After adding this observation to the normal equations, the minimum variance estimate for X is:

$$X_{est} = N^{-1}\ L$$

The measurement residuals for each input point can be calculated as:

$$V_{jik} = A_{jik}\ X_{est} - Y_{jik}$$

These residuals can be assessed by band pair to determine whether additional uncompensated biases exist in particular band combination measurements and to evaluate the point matching accuracy of particular band combinations.

The resulting parameter vector X_{est} contains the estimates of the eight band offset pairs (x and y directions) in microradians. These values can be used to update the band offsets

(expressed in microradians) in the Focal Plane Parameters parameter group in the Calibration Parameter File, as defined in Reference 24.

3.1.5.4.3   Scan Mirror Calibration

The purpose of the Scan Mirror Calibration segment of the Landsat 7 Image Assessment System is to check and, if necessary, correct the along-scan and across-scan mirror profiles.

As a preliminary approach to scan mirror calibration, a precision terrain-corrected Landsat panchromatic band is matched to a reference image with correlation. Currently, the best candidate for the reference is USGS digital orthophoto quadrangle (DOQ) imagery. Using the precision mapping grid, correlation windows can be selected that lie entirely within one scan, thus allowing forward and reverse mirror scans to be separated.

For each successful correlation between a point in the reference image and a point in the Landsat scene, the respective image coordinates (lines, samples) in output space are mapped back through the inverse grid into input space (3.1.5.3.6). After being mapped to input space, the effects of terrain are removed from the two samples to locate the (line, sample) input space coordinate associated with the measured output space (line, sample) (3.1.5.3.8). The remaining difference between the samples/lines is then taken to be the error in the scan mirror profile. Using the forward model, the time in scan of each sample is calculated from the input sample (3.1.5.3.4).

This correlation and inverse mapping procedure is used to generate a number of mirror error data points for several forward and reverse scans, each with an associated time in scan. Outlier rejection is performed by grouping the data points by scan direction and scan angle. Points that deviate from the mean are removed.

After the data have been filtered for outliers, the data from forward and reverse are fit with fifth-order Legendre polynomials, as developed below. If the error in the mirror profile is beyond some predetermined value, the existing mirror coefficients can be updated in the Calibration Parameter File. The error polynomial coefficients are saved for trending.

3.1.5.4.3.1   Algorithm Development

The pre-launch mirror profiles are characterized as deviating from linear by a fifth-order polynomial:

$$b(t) = b_0 + b_1 t + b_2 t^2 + b_3 t^3 + b_4 t^4 + b_5 t^5 \qquad 0 < t < T_{nom} \qquad (1)$$

This can also be written as an orthogonal fifth-order Legendre polynomial:

$$p(x) = a_0 + a_1x + a_2(3/2x^2 - 1/2) + a_3(5/2x^3 - 3/2x) + a_4(35/8x^4 - 30/8x^2 + 3/8)$$
$$+ a_5(63/8x^5 - 70/8x^3 + 15/8x) \qquad -1 < x < 1 \qquad (2)$$

where $x = 2t/T_{nom} - 1$

To convert back and forth between the mirror coefficients b_i and the Legendre coefficients a_i, we equate the two above polynomials and gather terms of the same order. This leaves us with the transformations:

$$b_0 = a_0 - a_1 + a_2 - a_3 + a_4 - a_5$$
$$b_1 = (2a_1 - 6a_2 + 12a_3 - 20a_4 + 30a_5)/T_{nom}$$
$$b_2 = (6a_2 - 30a_3 + 90a_4 - 210a_5)/T_{nom}^2 \qquad (3)$$
$$b_3 = (20a_3 - 140a_4 + 560a_5)/T_{nom}^3$$
$$b_4 = (70a_4 - 630a_5)/T_{nom}^4$$
$$b_5 = (252a_5)/T_{nom}^5$$

$$a_5 = (b_5T_{nom}^5)/252$$
$$a_4 = (b_4T_{nom}^4)/70 + (b_5T_{nom}^5)/28$$
$$a_3 = (b_3T_{nom}^3)/20 + (b_4T_{nom}^4)/10 + (b_5T_{nom}^5)5/36 \qquad (4)$$
$$a_2 = (b_2T_{nom}^2)/6 + (b_3T_{nom}^3)/4 + (b_4T_{nom}^4)2/7 + (b_5T_{nom}^5)25/84$$
$$a_1 = (b_1T_{nom})/2 + (b_2T_{nom}^2)/2 + (b_3T_{nom}^3)9/20 + (b_4T_{nom}^4)2/5 + (b_5T_{nom}^5)5/14$$
$$a_0 = b_0 + (b_1T_{nom})/2 + (b_2T_{nom}^2)/3 + (b_3T_{nom}^3)/4 + (b_4T_{nom}^4)/5 + (b_5T_{nom}^5)/6$$

As shown in the section on scan mirror correction 3.1.5.3.3, the coefficients of the mirror scan profile are scaled by the total time of each scan, which is calculated from the first-half and second-half scan errors:

$$b_i' = b_i(T_{nom}/T_s)^i \qquad (5)$$
$$b_i'(T_s)^i = b_i(T_{nom})^i \qquad (6)$$

From the transformations in Equation 4, the relationship in Equation 6 leads directly to:

$$a_i' = a_i \qquad (7)$$

where a_i' are the coefficients of a polynomial in the form of Equation 2 ,but where the auxiliary variable x has been replaced by:

$$x' = 2t/T_s - 1 \qquad (8)$$

Therefore, using the actual scan time T_s instead of the nominal scan time T_{nom} to compute the auxiliary variable x takes care of the scan time scaling without changing the Legendre polynomial coefficients a_i .

In addition to the fifth-order deviation from a linear profile, as discussed in 3.1.5.3.3 (scan mirror correction), an adjustment to the quadratic terms is used to fit the mid-scan error.

The linear portion of the mirror model is:

$$L(t) = (a_{me}t + a_{sm}(T_s - t))/T_s \qquad (9)$$
$$l(x) = (a_{me} + a_{sm})/2 + x(a_{me}-a_{sm})/2 \qquad (10)$$

where a_{sm} and a_{me} are the mirror angles at start and end of scan, respectively. The predicted mirror angle at the time of first-half scan t_{fh} is:

$$L(t_{fh}) = (a_{me}t_{fh} + a_{sm}t_{sh})/T_s = l(\varepsilon) \qquad (11)$$

where:

$$t_{sh} = T_s - t_{fh}$$
$$\varepsilon = 2t_{fh}/T_s - 1 = (t_{fh}-t_{sh})/T_s \qquad (12)$$

Thus, as shown in the section on scan mirror correction, the mid-scan correction due to deviation from linear (the difference between what the linear model predicts and what is observed; i.e., zero) is:

$$A_f = -(a_{me}t_{fh} + a_{sm}t_{sh})/T_s = -l(\varepsilon) \qquad (13)$$

Therefore, the total mid-scan correction, including the mirror profile, is:

$$D_f = A_f - p(\varepsilon) = -l(\varepsilon)- p(\varepsilon) \qquad (14)$$

Again referring to the section on scan mirror correction, the coefficients for the quadratic adjustment to the fifth-order polynomial are:

$$\Delta b_0' = 0$$
$$\Delta b_1' = (D_fT_s)/(t_{fh}t_{sh}) \qquad (15)$$
$$\Delta b_2' = -D_f/(t_{fh}t_{sh})$$

Using the transformations in Equation 4, these corrections can be written in terms of Legendre coefficients as:

$$\Delta a_0 = \Delta b_0' + (\Delta b_1'T_s)/2 + (\Delta b_2'T_s^2)/3 = (D_fT_s^2)/(6t_{fh}t_{sh})$$
$$\Delta a_1 = (\Delta b_1'T_s)/2 + (\Delta b_2'T_s^2)/2 = 0 \qquad (16)$$
$$\Delta a_2 = (Db_2'T_s^2)/6 = -(D_fT_s^2)/(6t_{fh}t_{sh})$$

They can be rewritten as:

$$\Delta a_0 = (A_f - p(\varepsilon))W_s = - (I(\varepsilon) + p(\varepsilon))W_s$$
$$\Delta a_1 = 0 \tag{17}$$
$$\Delta a_2 = -(A_f - p(\varepsilon))W_s = (I(\varepsilon) + p(\varepsilon))W_s$$

where:

$$W_s = T_s^2/(6t_{fh}t_{sh})$$

Adding these to the coefficients in Equation 2, we get:

$$p'(x') = a_0 + a_1 x' + a_2(3/2x'^2 - 1/2) + a_3(5/2x'^3 - 3/2x') \tag{18}$$
$$+ a_4(35/8x'^4 - 30/8x'^2 + 3/8) + a_5(63/8x'^5 - 70/8x'^3 + 15/8x')$$
$$+ p(\varepsilon)W_s(3/2x'^2 - 3/2) - A_f W_s(3/2x'^2 - 3/2)$$

since $t_{fh} \approx t_{sh} \approx T_s/2$ and $\varepsilon \approx 0$, looking at three special cases gives us:

$$p'(\varepsilon) = p(\varepsilon) + 2/3\, p(\varepsilon)(3/2\varepsilon^2 - 3/2) + 2/3\, I(\varepsilon)(3/2\, \varepsilon^2 - 3/2) \approx -I(\varepsilon)$$
$$p'(-1) = p(-1) \tag{19}$$
$$p'(1) = p(1)$$

From this we see that the quadratic correction does not change the scan mirror profile at the beginning and end of scan, where it is already constrained to be zero, and that any mirror profile deviation at mid-scan is suppressed by the correction quadratic. Therefore, any update to the mirror profile should obey the beginning and end of scan constraints:

$$\Delta p(-1) = 0 \tag{20}$$
$$\Delta p(1) = 0$$

Also, any update to the mirror profile at mid-scan is suppressed by the quadratic correction, which implies that any errors in the mirror profile at mid-scan can be neither measured nor corrected.

To account for these three special cases in our mirror correction model, we apply three constraints to the correction profile $\Delta p(x')$ and treat them as additional observations:

$$\Delta p(1) = a_0 + a_1 + a_2 + a_3 + a_4 + a_5 = 0$$
$$\Delta p(-1) = a_0 - a_1 + a_2 - a_3 + a_4 - a_5 = 0 \tag{21}$$
$$\Delta p(\varepsilon) = a_0 + a_1\varepsilon + a_2(3/2\varepsilon^2 - 1/2) + a_3(5/2\varepsilon^3 - 3/2\varepsilon)$$
$$+ a_4(35/8\varepsilon^4 - 30/8\varepsilon^2 + 3/8) + a_5(63/8\varepsilon^5 - 70/8\varepsilon^3 + 15/8\varepsilon) = 0$$

where the a_is are now the corrections to the nominal profile. These constraints have the practical effect of removing the low-order terms a_0, a_1, and a_2 from the correction model. The first two constraints can be reformulated and simplified:

$$a_0 + a_2 + a_4 = 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (22)$$
$$a_1 + a_3 + a_5 = 0$$

Since the last constraint applies at ε, which is very close to zero, it can be approximated by:

$$a_0 - 1/2a_2 + 3/8a_4 = 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad (23)$$

These equations can be rearranged to determine constrained values for the low-order coefficients a'_0, a'_1 and a'_2 from the other three coefficients:

$$a'_0 = -7/12a_4$$
$$a'_1 = -a_3 - a_5 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (24)$$
$$a'_2 = -5/12a_4$$

A fifth-order polynomial that obeys these constraints can be used to model the higher order mirror non-linearity, which cannot be removed by the mid-scan correction quadratic. The residual quadratic difference between the estimated mirror profile and the constrained mirror profile can be computed as:

$$c_0 = a_0 - a'_0 = a_0 + 7/12a_4$$
$$c_1 = a_1 - a'_1 = a_1 + a_3 + a_5$$
$$c_2 = a_2 - a'_2 = a_2 + 5/12a_4 \qquad\qquad\qquad\qquad\qquad\qquad (25)$$

This residual quadratic contains information about apparent mirror angle deviations at the three mirror control points: scan start, mid-scan, and scan end. It can be used to estimate corrections to the mirror scan angle monitor (SAM) angles as follows:

$$p(x) = c_0 + c_1 \, x + c_2 \, (3/2x^2 - 1/2)$$

$$p(-1) = \Delta_{-1} = c_0 - c_1 + c_2 \qquad\qquad\qquad \text{deviation at scan start}$$
$$p(0) = \Delta_0 = c_0 - 1/2c_2 \qquad\qquad\qquad\quad\ \text{deviation at mid-scan}$$
$$p(1) = \Delta_1 = c_0 + c_1 + c_2 \qquad\qquad\qquad\ \text{deviation at scan end}$$

$$\text{Delta SAM Angle 1} = \Delta_0 - \Delta_{-1} = c_1 - 3/2c_2$$
$$\text{Delta SAM Angle 2} = \Delta_1 - \Delta_0 = c_1 + 3/2c_2 \qquad\qquad\qquad\qquad (26)$$

The measured deviations, converted to mirror angle deviations as a function of time, provide observations of the $\Delta p(x')$ mirror correction function. A weighted least squares technique is used to fit a Legendre polynomial, like Equation 2, to these mirror angle deviations. The user selects the weight for the observations based on the expected reference image and correlation accuracy. The constraints are then applied to the coefficients computed by the weighted least squares procedure, and the residual quadratic deviation is computed. The calculated SAM angle corrections for the forward and reverse scan directions are averaged to compute a single correction for each half

scan. This enforces the additional constraint of common SAM angles for both scan directions. In computing the averages, the signs are reversed for the reverse scan estimates due to the opposite sign convention used for reverse scan angles:

Delta FWD Angle 1 = Delta REV Angle 2 =
$$[(FWD\ c_1 - 3/2\ FWD\ c_2) - (REV\ c_1 + 3/2\ REV\ c_2)] / 2$$

Delta FWD Angle 2 = Delta REV Angle 1 =
$$[(FWD\ c_1 + 3/2\ FWD\ c_2) - (REV\ c_1 - 3/2\ REV\ c_2)] / 2 \quad (27)$$

The existing mirror polynomial coefficients are converted to Legendre space according to the transformations in Equation 4. If the higher order terms of the determined error polynomial are deemed by the operator to be significant and reproducible, then they are added to the existing coefficients and transformed back into mirror space according to Equation. 3. Similarly, if the operator considers the computed corrections to the SAM angles to be significant and reliable, they are added to the existing mirror angles.

3.1.5.4.3.2   Inputs

As input, the mirror calibration algorithm requires a precision corrected grid and a terrain-corrected image. It also requires the same digital elevation model that was used in the terrain correction algorithm and the terrain correction table it generates. To calculate the time in scan for each sample, it also needs to have the forward sensor model. Finally, it requires a reference image in the same framing as the terrain-corrected image. The current plan is to use DOQ imagery as a reference and the PCD flags that indicate the scan mirror electronics (SME) number and scanning mode (Scan Angle Monitor or Bumper) to know which mirror profile is applicable.

3.1.5.4.3.3   Outputs

The scan mirror calibration algorithm generates a set of fifth-order Legendre polynomial coefficients for correction to the along- and across-scan mirror profiles for both forward and reverse scans. If the deviation from the existing profile is deemed significant, then the existing higher order mirror coefficients are transformed into Legendre space. The deviations are added to them, and then they are converted back into mirror coefficients. The lower order terms can be kept for characterization, although (as shown above) they cannot be uniquely determined and therefore should not be used for calibration. In addition, all of the data generated by the correlation routines is stored.

3.1.5.4.3.4   Correlation Testing

One of the key components of the mirror calibration procedure is the selection of an accurate reference image source. Digital orthophoto quad data are 1-meter resolution, panchromatic aerial imagery, which have been terrain corrected and rectified to a standard Universal Transverse Mercator (UTM) projection with an accuracy specification of 6 meters, 1 σ. They are available in 7.5 arc-minute quadrangles or 3.75 arc-minute

quarter-quads. To use DOQs as a reference, they need to be downsampled to the same resolution as the ETM+ imagery and mosaicked to cover several scans. In the case of Landsat 7, this means downsampling to the level of the panchromatic band.

Ideally, we would like to mosaic the DOQ tiles at the highest resolution possible to blend more effectively the radiometric differences that may be present. Practically speaking, however, the full-resolution DOQs are too large to mosaic more than a few at a time. For example, in one of the potential test sites (Landsat 4/5 path 28, row 30), the quarter-quad images are ~7,700 lines by ~6,900 samples, and it takes ~200 of them to completely cover the first 20 scans of the Landsat scene (see Figure 3-34). As a compromise, the DOQs may be mosaicked and reduced in resolution in stages.

For testing purposes, 16 quarter-quads from four full quadrangles were low-pass filtered and downsampled to 8-meter pixels. They were then mosaicked together and reprojected to the Space Oblique Mercator (SOM) projection in a Landsat Path-orientation at a factor of the final desired pixel size (7.5 meters). The mosaicked image was then low-pass filtered again and downsampled twice more to the final 30-meter pixel size of the test image.
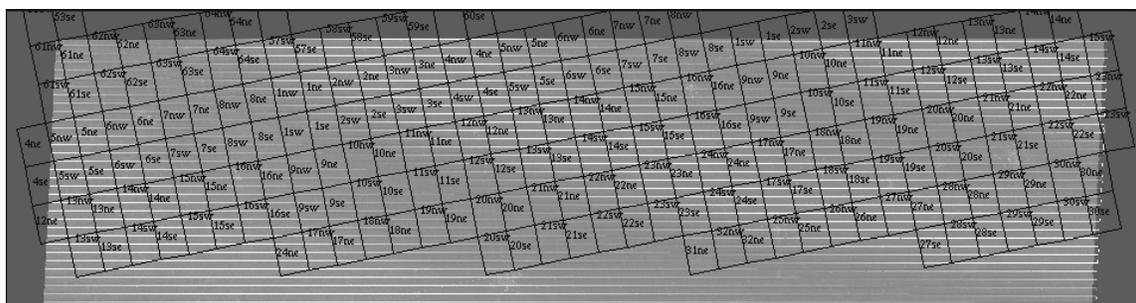
Figure 3-34. DOQ Quarter-quad Images Covering About 20 TM Scans

Since a precision terrain-corrected image was not available, a section of a Landsat 5 systematic image was correlated with the assembled DOQ imagery using hand-picked points. The TM image used for the correlation was a synthetic panchromatic image (at 30-meter pixel size), generated by a linear combination of bands 1, 2, 3, and 4. Using the correlation offsets, a polynomial transformation was generated to remap the TM sub-scene into the frame and projection of the DOQ. Another set of features was hand-picked, and the two images were correlated. Without any sort of outlier rejection, the standard deviation of the correlation was ~0.25 pixels. Using a regular grid of correlation points, the standard deviation was ~0.35 pixels, again without any outlier rejection. Error analysis indicates that a correlation accuracy of 0.25 pixels should be sufficient. From these less-than-ideal tests, it appears as though this goal is achievable, particularly if the real panchromatic band of the ETM+ is to be used.

A consideration to keep in mind is that the DOQ reference images are generally taken in mid-spring. Therefore, it would be best to choose test sites where seasonal variability has the least impact.

3.1.5.4.3.5   Polynomial Fitting

To test the ability of the algorithm to determine corrections to the scan mirror coefficients, two systematic images were generated, one with the fifth-order mirror correction polynomial applied and one without. This allowed us to back out the original higher order coefficients. The two images were correlated using the first 96 scans. Each scan was broken down into 256 overlapping windows, measuring eight lines by 128 samples in output space. The inverse grid was used to make sure the correlation windows stayed within the scans in input space.

The correlated (line, sample) pairs in the image and reference output space were run through the inverse grid to find their corresponding (line, sample) in input space. In practice, it is at this point that the terrain adjustment would be added to the input samples. Since neither image had any terrain correction applied, it was not necessary for this test. Knowing the input (line, sample) of the image and having initialized the forward model, we can get a time-in scan corresponding to each correlation point. We can also get the actual scan time T_s for each scan, which allows us to determine the auxiliary variable x', as shown in Equation 8. The difference between samples/lines in the two TM images (in practice, a terrain-corrected ETM+ image and a reference) in input space is assumed to be an error that may be correctable by adjusting the coefficients of the scan mirror profile.

It is this difference as a function of the auxiliary variable x' that we attempt to fit with a Legendre polynomial. In practice, we would apply the constraints in Equation 19, though this was not done for this test.
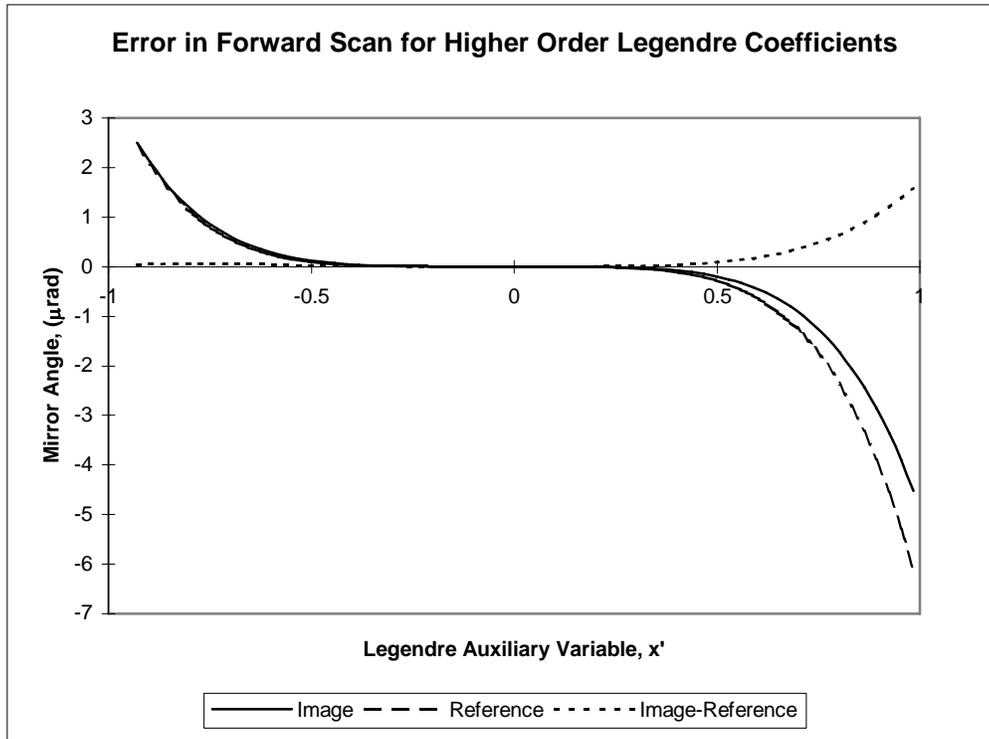
Figure 3-35. Legendre Differences
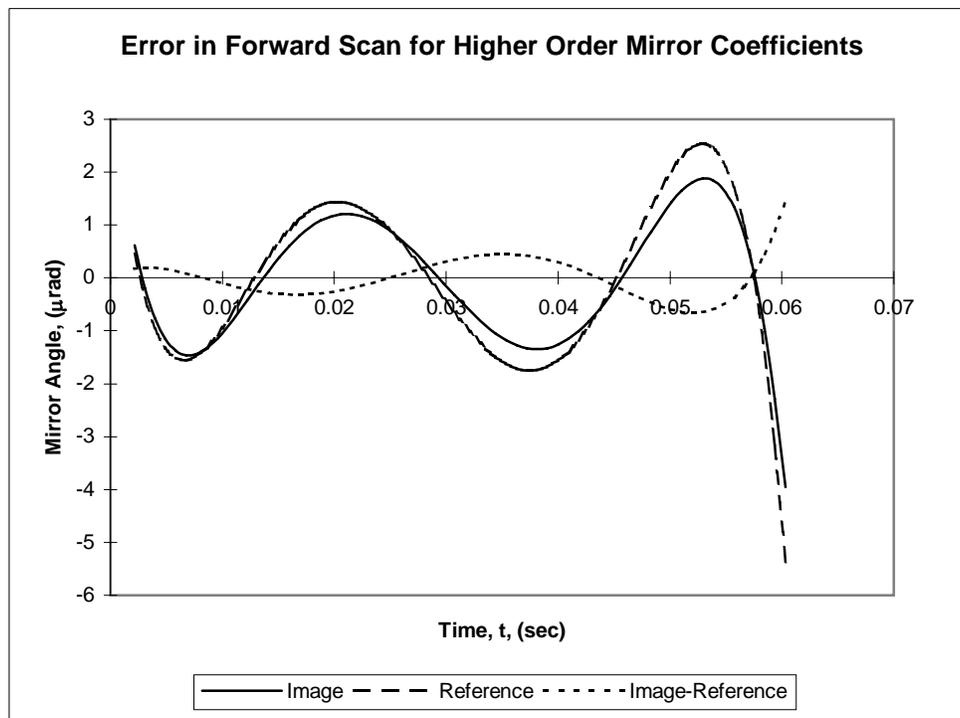
Figure 3-36. Difference between Actual Mirror Profiles

Figure 3-35 and Figure 3-36 show errors in the higher order terms of the correction polynomial in Legendre terms and mirror terms, respectively. In this test case, we expect higher order terms of the polynomial we find to be the same as the actual along-scan mirror coefficients used in the base image. As shown in the figures, the algorithm was able to back out the polynomial to better than 1 μrad over most of the scan and better than 2 μrad in all locations. The mirror profile repeatability specification for the Landsat 7 ETM+ is 1.75 μrad 1σ, so the measurements using 42.5 μrad IFOV TM data are of the same order as the random mirror variations.

3.1.5.5   Geometric Characterization Algorithms

3.1.5.5.1   Band-to-Band Registration

The Band-to-Band Registration Assessment segment of the Landsat 7 Image Assessment System (IAS) is used to check the registration quality between bands of the same image.

3.1.5.5.1.1   Matching Resolutions

Due to the difference in pixel size between bands, the higher resolution data needs to be reduced so that it matches that of the lower resolution. The band-to-band characterization is then performed at the lower resolution. The higher resolution imagery is reduced using the Gaussian Pyramid approach (see Reference 25). This approach works by low-pass filtering the image data. The low-pass filtering is accomplished by convolving the image with an appropriate kernel or set of weights. Applying this kernel produces an image at a lower resolution than that of the original. The lower resolution image can then also be reduced in size. The process takes on the following form:

$$f_r(j,k) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m,n) f(2j+m, 2k+n)$$

j = 1, …, M/2
k = 1, …, N/2

where f is the original image data, f_r is the reduced sub-sampled image, and w(m, n) are the low-pass filter coefficients. In the following case, the imagery is reduced in size by one half in both the line and sample direction. The low-pass filter coefficients can be represented as a set of separable one-dimensional weights and are determined from the following representation (see Reference 25):

w(2) = 1/4 − *a*/2
w(1) = 1/4
w(0) = *a*
w(−1) = 1/4
w(−2) = 1/4 − *a*/2

where *a* typically takes on a value between 0.3 and 0.6. More information on this technique can be found in Reference 25.

In the case of Landsat 7 data, this process is used to reduce the 15-meter PAN band to the 30-meter multispectral bands and to the 60-meter thermal band. It is also used to reduce the 30-meter multispectral band to the 60-meter thermal band. In the case of Landsat 4 and 5 test data, the 30-meter multispectral band is reduced to the 120-meter thermal band.

3.1.5.5.1.2   Correlation Routine

Normalized cross-correlation is used to produce a discrete correlation surface for an array of points between two bands. This correlation routine helps to remove any gray-scale differences between two images. This helps to alleviate some of the problems with the different radiometric responses between the bands. The correlation routine uses the following formula when correlating between two image windowed subsets f and g, each with length and width of M and N, respectively:

$$R(x, y) = \frac{\sum\limits_{j=-N/2}^{N/2} \sum\limits_{i=-M/2}^{M/2} \left[\left(f(j,i) - \bar{f} \right) \left(g(x+j, y+i) - \bar{g} \right) \right]}{\left[\left[\sum\limits_{j=-N/2}^{N/2} \sum\limits_{i=-M/2}^{M/2} \left[\left(f(j,i) - \bar{f} \right)^2 \right] \right] \left[\sum\limits_{j=-N/2}^{N/2} \sum\limits_{i=-M/2}^{M/2} \left[\left(g(x+j, y+i) - \bar{g} \right)^2 \right] \right] \right]^{\frac{1}{2}}}$$

where:

$$\bar{f} = \frac{1}{(M+1)(N+1)} \sum\limits_{j=-N/2}^{N/2} \sum\limits_{i=-M/2}^{M/2} f(j,i)$$

$$\bar{g} = \frac{1}{(M+1)(N+1)} \sum\limits_{j=-N/2}^{N/2} \sum\limits_{i=-M/2}^{M/2} g(x+j, y+i)$$

After the correlation surface is generated, the peak of that surface must be determined. This peak represents the geometric shift between two windowed data sets and is calculated to a subpixel location.

3.1.5.5.1.3   Peak Finding Method

A sub-pixel location for the peak is found by using a polynomial fit method. This method works by fitting a second-order polynomial around the peak of the discrete correlation surface. The polynomial represents the surface associated with a 3 x 3 area around the peak. The discrete peak of the correlation surface is found first, then this peak and its corresponding eight neighbors are used for fitting the polynomial. A least squares fit can be used to calculate the polynomial coefficients after the 3 x 3 area and corresponding

data points are found. From this polynomial, the location of the peak can be solved analytically to a sub-pixel location.

The polynomial equation has the following form:

$$P(x,y) = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2$$

The coefficients of P(x, y) can be found by solving the matrix equation:

$$[Y] = [X][a]$$

where:

> [Y] is a 9 x 1 matrix corresponding to the eight correlation points found around, and including, the peak

> [X] is a 9 x 6 matrix with each row consisting of:

> > $1 \qquad x_i \qquad y_i \qquad x_i\,y_i \qquad x^2_i \qquad y^2_i$

> where x_i and y_i are the x and y locations of the corresponding peak i in [Y]

> [a] is the polynomial coefficient

The solution of [a] is then:

$$[a] = ([X]^T[X])^{-1}[X]^T[Y]$$

where T refers to the transpose of a matrix, and −1 refers to the inverse of a matrix.

After the coefficients are solved for, the sub-pixel peak can be found. The maximum correlation peak can be found by taking the partial derivative of P(x, y) with respect to x and y:

$$\frac{\partial}{\partial x}P(x,y) = a_1 + a_3y + 2a_4x$$
$$\frac{\partial}{\partial y}P(x,y) = a_2 + a_3x + 2a_5y$$

By setting the partial derivatives to zero and solving for x and y, the sub-pixel peak location can be determined. This leads to the following solution for the x and y offsets:

$$x\_offset = \frac{2a_1a_5 - a_2a_3}{a_3^2 - 4a_4a_5}$$

$$y\_\text{offset} = \frac{2a_2a_4 - a_1a_3}{a_3^2 - 4a_4a_5}$$

3.1.5.5.1.4   Outlier Detection

After computing a number of sub-pixel shifts between two bands, the data are checked for outliers. These outliers are then removed from the data set.

Two steps are involved in detecting outliers. The first step involves looking at the measured offsets and comparing them with a threshold established by an error budget associated with the Landsat 7 satellite and the correlator. If any measurements are greater than this threshold, they are removed as outliers. If a large percentage of the measurements are thrown out at this point, then the satellite is either behaving unexpectedly according to specifications or none of the imagery corresponding to where the measurement was taken is of good quality, a possibility for this would be a large amount of cloud cover. The second step involves comparing the measured value with the bias estimate of all the measurements left after applying the first threshold.

The first threshold is found from the following:

  μrad = microradians

The known errors associated with the satellite and correlator are:

  Scan mirror repeatability = 3.5 μrad
  Field angle = 3.6 μrad
  Jitter = 1.45 μrad (spacecraft)
  Jitter = 0.65 μrad (ETM+)
  Processing error = 2.84 μrad

This leads to a total error for each band of:

$$\sqrt{(3.5)^2 + (3.6)^2 + (1.6)^2 + (2.84)^2} = 6.0\,\mu\text{rad}$$

 The total error associated with two bands is then:

$$\sqrt{(6.0)^2 + (6.0)^2} = 8.5\,\mu rad$$

Converting the microradian to pixels results in approximately 0.2 pixels. (The correlator error budget is 0.25 pixel.)

The total error budget is then:

$$\sqrt{(0.2)^2 + (0.25)^2} = 0.32\ \text{pixels}$$

The maximum band alignment error is 0.3 pixel.

The total measurement error is then:

$$\sqrt{(0.3)^2 + (0.32)^2} = 0.44 \, \text{pixels}$$

The threshold is then chosen as three times the total measurement error budget:

3 * 0.44 = 1.32 pixels

This value, 1.32 pixels, represents the first threshold that is applied to the data. All measurements above this threshold are considered outliers and are removed.

The second threshold is found by calculating the average or bias and the standard deviation for the data set left after applying the first threshold outlier detection. The second threshold is any point that is three standard deviations above the measured bias or average. The data set left after applying the second threshold outlier detection represents the valid measurements taken.

3.1.5.5.1.5   Statistical Measure

After the outliers have been detected and removed, the resulting data are used to calculate a bias or average band offset between two bands. The standard deviation of the data set is also calculated:

$$\text{avg} = \frac{1}{N} \sum_{i=0}^{N} m_i$$

$$(\text{std})^2 = \frac{1}{N-1} \sum_{i=0}^{N} (m_i - \text{avg})^2$$

where:

> avg = average
> std = standard deviation
> m_i = measured offset
> N = number of valid measurements

3.1.5.5.1.6   Test Simulation

A test simulation was generated to determine the accuracy of the correlation and peak finding routines by using an image of path 28, row 30 and resampling it with various geometric offsets. A 16 x 16 truncated sinc (sin[x]/[x]) resampling function was used to produce geometric shifts. Correlation windows of 512 x 512 were chosen. Sixteen windows were chosen at random for measurement. The various images of geometric offsets were then correlated with the original. This provided an indication of the accuracy of the correlator itself. The results are listed below:

Offset	Average Offset	Standard Deviation
0.0	0.0	0.0
0.1	0.089817	0.003823
0.2	0.152810	0.001479
0.3	0.236563	0.002199
0.4	0.344935	0.002791
0.5	0.485531	0.006224
0.6	0.632771	0.008459
0.7	0.745469	0.008163
0.8	0.831895	0.006718
0.9	0.903604	0.004832

This indicates that the correlator works to within one-tenth of a pixel. This represents an ideal situation in that the image is being correlated to itself. Through work with different band combinations, an accuracy of 0.25 pixels can be expected across bands from the correlator. A more thorough explanation of errors associated with correlation can be found in Reference 21. The image-to-image characterization algorithm also performed a similar test simulation; this can also be used as a reference on the accuracy of the correlator.

3.1.5.5.1.7   Measured Landsat 4 and 5 Data

Correlation may be done in such a manner that all bands are referenced either directly or indirectly to each other.   The test site that is chosen to perform band-to-band registration should have features that are present throughout all bands.

It was found that desert scenes with little vegetation presented the best scenario. A scene corresponding to path 41, row 36 was chosen as a test site. Correlation windows were chosen at uniform increments throughout the image. When correlating the multispectral bands, an offset of 1,000 lines from the top and bottom and 1,000 samples

from the right and left sides of the image was chosen. Windows were then chosen at 500 lines and 500 sample increments throughout the image. The chosen window size was 256 x 256. When correlating two bands, one of which was the thermal band, offsets of 250 lines, 250 samples, increments of 125 lines, 125 samples, and window sizes of 128 x 128 were chosen. The following values were then calculated.

3.1.5.5.1.8 X Offsets

Offsets in terms of multispectral pixels:

Reference	Search	Average	Standard Deviation
1	2	0.003244	0.010120
1	3	0.001516	0.021785
1	4	−0.025383	0.113919
1	5	−0.261743	0.061539
1	7	−0.252938	0.053082
2	3	−0.000870	0.012164
2	4	−0.014538	0.134436
2	5	−0.266884	0.055054
2	7	−0.249701	0.047483
3	4	−0.035892	0.079654
3	5	−0.267662	0.050735
3	7	−0.250002	0.048614
4	5	−0.227007	0.083736
4	7	−0.194449	0.145899
5	7	0.012561	0.020213

Offsets in terms of thermal pixels:

Reference	Search	Average	Standard Deviation
1	6	−0.786044	0.177695
2	6	−0.788873	0.164740
3	6	−0.808790	0.159983
4	6	−0.758903	0.194022
5	6	−0.757974	0.108728
6	7	0.754215	0.165351

3.1.5.5.1.9   Y Offsets

Offsets in terms of multispectral pixels:

Reference	Search	Average	Standard Deviation
1	2	0.003527	0.009219
1	3	0.014177	0.019110
1	4	0.024184	0.060681
1	5	0.366080	0.062386
1	7	0.382187	0.051700
2	3	0.010344	0.011238
2	4	0.014033	0.075062
2	5	0.362300	0.047765
2	7	0.374961	0.044400
3	4	0.006703	0.067899
3	5	0.350089	0.034363
3	7	0.365271	0.036863
4	5	0.319364	0.073798
4	7	0.328139	0.097649
5	7	0.007921	0.014011

Offsets in terms of thermal pixels:

Reference	Search	Average	Standard Deviation
1	6	0.737981	0.177695
2	6	0.726864	0.197395
3	6	0.718318	0.169007
4	6	0.747355	0.191474
5	6	0.633007	0.095764
6	7	−0.611179	0.153688

The measured between-band alignment does not meet the specifications for the Landsat 7 system. These data are used by the focal plane band-to-band calibration algorithm to relate the parameters back to the focal plane to adjust the focal plane alignment, if necessary.

3.1.5.5.2   Image-to-Image Registration

The Image-to-Image Assessment segment of the Landsat 7 Image Assessment System (IAS) is used to check the registration quality between two images. The two images are assumed to be from the same band and same (path, row), taken at different times, processed to the same level—either precision- or terrain-corrected products.

3.1.5.5.2.1   Accuracy Requirements

According to the Landsat 7 IAS Requirements Document, "The IAS shall provide the capability to perform image-to-image registration to an accuracy of 0.4 multispectral sensor GSD, 0.9p, in the along-track and cross-track directions, provided all inputs are within specification." The understanding is that this requirement refers to two temporally spaced images processed to the same level using the same ground control. Furthermore, since the accuracy requirements for systematically processed images are on the order of ~250 meters (400 meters, 0.9p), this requirement is assumed applicable only to precision- and/or terrain-corrected images.

As for requirements placed on the image-to-image correlation technique, "The IAS shall be capable of digitally correlating common features in the…same bands of separate images to an accuracy of 0.1 pixel, 0.9p." This is equivalent to requiring that the correlation technique be capable of backing out a known offset to within 0.1 pixel, 0.9p (~0.06 pixels at one sigma).

3.1.5.5.2.2   Procedure

The IAS Image-to-Image Assessment segment provides two options for selecting image-to-image points. Correlation points can be generated either by creating a regular grid of hundreds of points or by manually selecting 50–100 well distributed "human-recognizable" ground control features. Given that images need to be processed only to the full-precision and terrain-corrected levels for a small number of test sites, the latter is the more likely operational implementation. Using the grid of points, however, may be useful in geometric analysis.

The normalized cross-correlation technique is defined as:

$$R(x, y) = \frac{\displaystyle\sum_{j=-N/2}^{N/2} \sum_{i=-M/2}^{M/2} \left[\left(f(j,i) - \bar{f} \right) \left(g(x+j, y+i) - \bar{g} \right) \right]}{\left[\left[\displaystyle\sum_{j=-N/2}^{N/2} \sum_{i=-M/2}^{M/2} \left[(f(j,i) - \bar{f})^2 \right] \right] \left[\displaystyle\sum_{j=-N/2}^{N/2} \sum_{i=-M/2}^{M/2} \left[(g(x+j, y+i) - \bar{g})^2 \right] \right] \right]^{\frac{1}{2}}}$$

where:

$$\bar{f} = \frac{1}{(M+1)(N+1)} \sum_{j=-N/2}^{N/2} \sum_{i=-M/2}^{M/2} f(j,i)$$

$$\bar{g} = \frac{1}{(M+1)(N+1)} \sum_{j=-N/2}^{N/2} \sum_{i=-M/2}^{M/2} g(x+j, y+i)$$

The resulting correlation surface is searched for a peak. To determine the sub-pixel location of the true correlation maximum, a surface is fit to the peak and its neighbors. The sub-pixel location of the peak value for the best-fit function is determined analytically and stored. The correlation functions included options for fitting parabolic and Gaussian surfaces using a 5 x 5 neighborhood of pixels around the peak. The correlation function was modified to use a 3 x 3 neighborhood as testing showed these results to be more accurate and reliable. This is discussed in greater detail below.

The verification routine uses the points and offsets determined by the correlation function to calculate statistics. For each correlation point, it reports the line, sample, and total errors in the search image relative to the reference. It also calculates the mean, standard deviation, median, range, and RMSE for those errors and determines the number (and percentage) of points not meeting user-specified accuracy requirements. All of these values are stored for trending.

To assist geometric assessment, the statistics compiled by verification routine can be plotted. Useful plots include, but are not limited to, surface plots of line or sample error versus line and sample, vector plots showing line and sample errors distributed on the image plane, or color-coded residuals overlaid on an image.

Some outlier detection and rejection is performed within the correlation function. The routine flags and rejects correlation points if they are too near the edge of the data window, similar in magnitude to subsidiary peaks, or below a user-specified strength threshold. A flag also rejects points if the diagonal distance between the reference and search sub-images is greater than a user-specified value.

Whether further outlier rejection will be used on the final residuals is yet to be determined, since initially, analysis is primarily concerned with determining whether outliers indicate a problem in the data or a problem in the system. An exception flag should be raised if the line and sample error mean and standard deviations are in excess of the requirement values specified above.

3.1.5.5.2.3   Inputs

Image-to-image assessment requires a reference image that has been processed to the precision- or terrain-corrected level and a search image processed to the same level with the same ground control (and the same DEM in the case of terrain-corrected images). It also requires a set of pre-selected image features or a grid whose locations are refined by the correlation function.

3.1.5.5.2.4   Outputs

The correlation function generates a file containing the points and offsets that is used by the verify routine to compile statistics. The verify routine generates a report file containing information about all points used for correlation, including line, sample, and total errors. It also lists overall statistics, including mean, standard deviation, median, and RMSE for

line, sample, and total errors, and the number of points not meeting user-specified error thresholds. The information in the report is used for trending .

3.1.5.5.2.5   Algorithm Testing

To test the accuracy and precision of the correlation routine, a base image using band 1 of a Landsat 5 test scene (path 28, row 30) of northwestern Iowa was generated. Artificial shifts in the sample direction were then incorporated into the image in 0.1-pixel increments using cubic convolution resampling accurate to 1/32 of a pixel. Image-to-image correlation was then performed with artificial offsets ranging from 0.0 to 1.0 pixels, using a set of 51 manually picked points. The resulting plots of the mean and standard deviation values are shown in Figure 3-37 and Figure 3-38.

At an offset of 0.5 pixels, the correlation routine using a 5 x 5 neighborhood about the correlation peak became unreliable, giving a bimodal error distribution surrounding the true offset. A possible explanation for this peculiar behavior is that the objects used in the correlation are typically linear features, approximately 1 pixel wide (e.g., roads), surrounded by relatively uniform backgrounds (e.g., fields). In these cases, we would not expect a parabola to be a good fit beyond 1 pixel to either side of the peak.
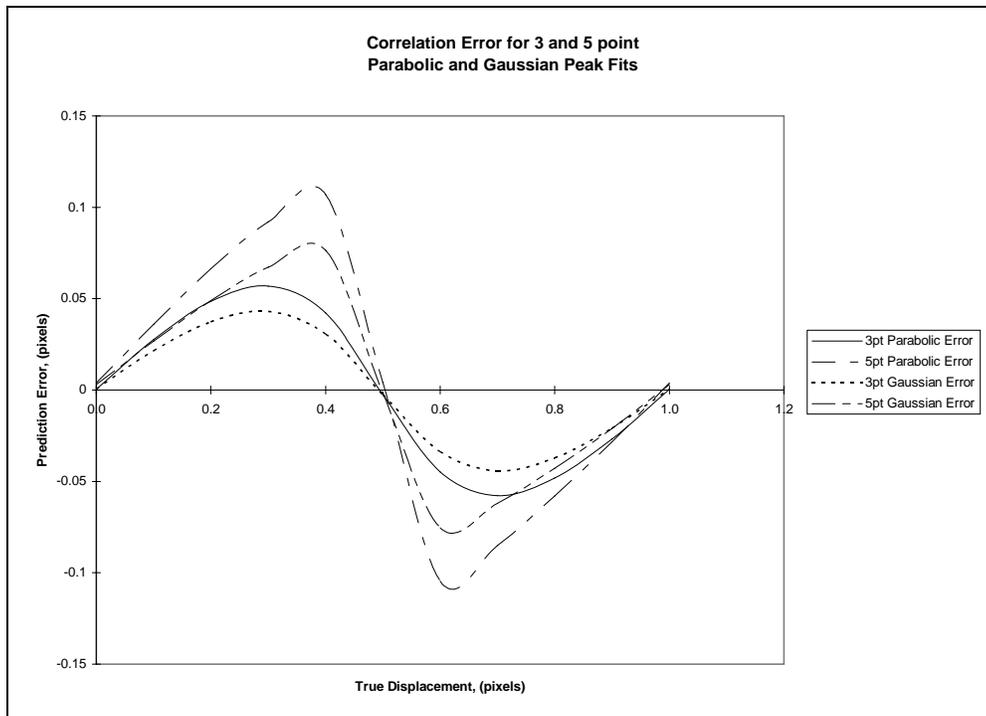
Figure 3-37. Correlation Errors
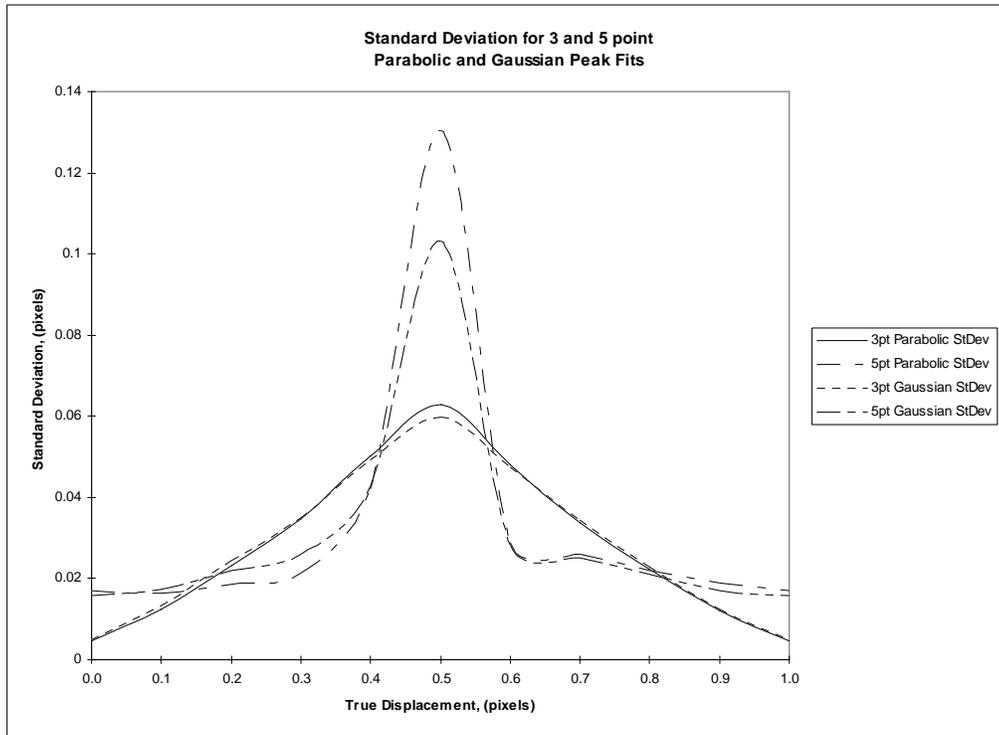
Figure 3-38. Correlation Standard Deviations

As an example, consider the case of a bright, 1-pixel-wide road cutting through a uniformly dark field. If this road were displaced to the right by 0.5 pixels, the resultant image would have the road represented by 2 pixels of medium brightness surrounded again by the dark field. This situation is shown schematically in Figure 3-39 as a one-dimensional cross-section. Figure 3-37 shows the cross-correlation (non-normalized) of these two data sets and the resulting parabolas, which occur when trying to fit the data with five points using both the left and right maxima as the center point. The resulting peaks of the parabolas in Figure 3-37 occur at 7.27 and 7.73, rather than at 7.5, where the true peak should occur. So while the average value is correct, the standard deviation for this case is unacceptable, as are shown in Figure 3-37 and Figure 3-38.

By inspecting Figure 3-39, it is evident that fitting a parabola using three points, rather than five points, results in a peak that occurs at 7.5, regardless of which maxima is taken as the center point. Figure 3-40 shows the three-point method to be more accurate than the five-point method. For these reasons, the correlation routine was modified to use a 3 x 3 neighborhood around the peak, rather than a 5 x 5 neighborhood.

By looking at the shape of the cross-correlation data in Figure 3-40, one would expect that a Gaussian curve might be a better fit than a parabola, thus giving reduced errors. The results of doing so are also shown in Figure 3-37, and indeed, they lead to ~20% reduction in error at the peak locations. The sinusoidal shape of the error curve shown in Figure 3-37 is consistent with the theoretical errors determined by Dvornychenko (see Reference 21).
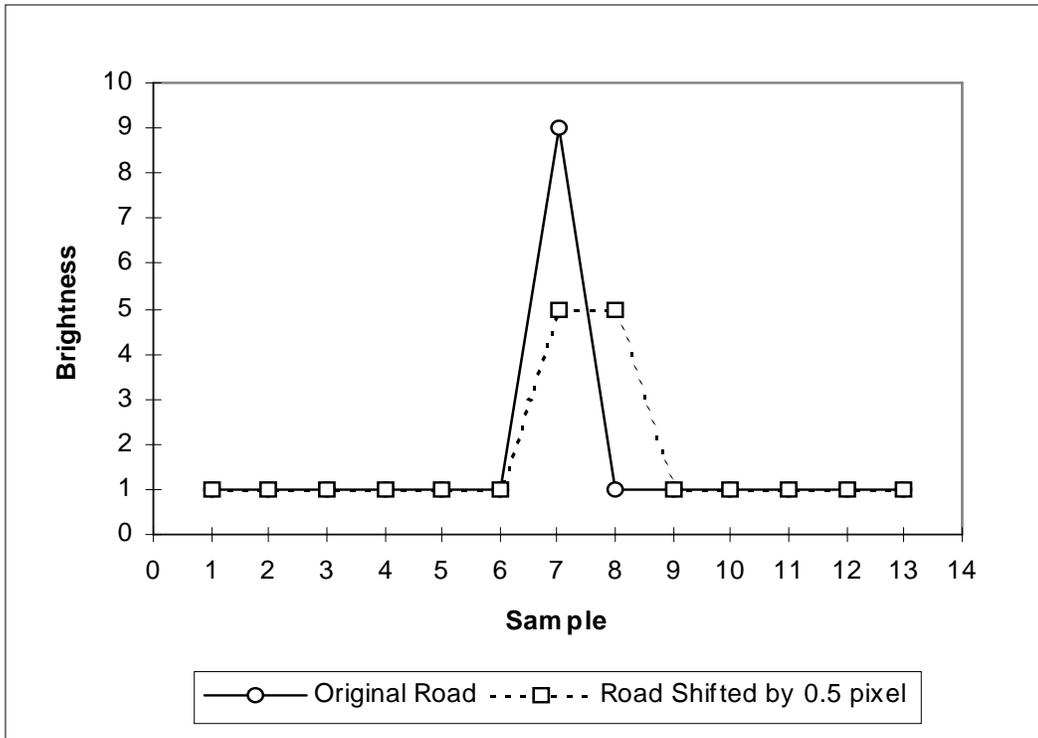
Figure 3-39. Road Cross-Section Example
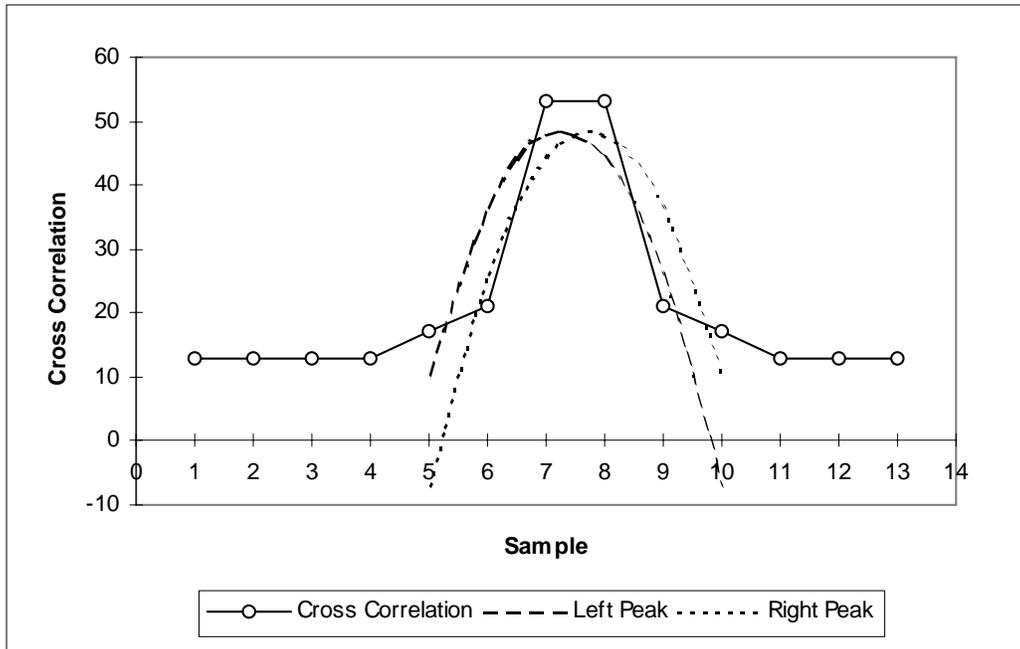
Figure 3-40. Correlation Errors

3.1.5.5.2.6   Further Investigations

While it appears as though the accuracy of the modified correlation routine is sufficient to meet the requirements, it must be noted that these tests were run under ideal conditions; i.e., the same image data (although shifted and aliased by using a cubic convolution resampler accurate to 1/32 pixel). Under real circumstances, where there may be significant changes in the scenes from image to image, the standard deviations can be expected to increase.

Normalized cross-correlation is designed to take some image differences into account and has been used successfully in projects involving Landsat MSS and TM imagery. However, it has not been used to measure the registration accuracy under such stringent requirements as in the IAS.

Further investigations may include widening the reference chips to greater than 64 x 64. This should help by having more data available and thus smoothing out the correlation surface. Also, other techniques such as edge or phase correlation may be useful when correlating images with differing spectral data (e.g., data from different seasons).

3.1.5.5.3   Geodetic Accuracy

This section describes the algorithm, input data, output data, and proposed usage of the output data applied in the Geodetic Characterization Module for the Landsat 7 Image Assessment System.

3.1.5.5.3.1   The Algorithm

In the system data flow chart, this algorithm immediately follows the precision correction solution module. Thus, this description should be read in context to the description of precision correction solution module. The Ground Control Point (GCP) data quality requirement applies to the input data of the precision correction solution module and is based on the assumption that GCP marking and correlation are done to 0.1 pixel accuracy.

3.1.5.5.3.2   Purpose

The purpose of this algorithm is to verify the absolute accuracy of the systematic and precision products by analyzing the GCP measurement residuals before and after the precision correction solution.

3.1.5.5.3.3   Process

This algorithm applies standard statistical error analysis on the residual file of the precision correction solution. The residual file contains residuals for GCPs before and after precision correction. The algorithm takes one input file (the residual file), and generates two output files (the record file and the outlier data file). The following steps are performed:

2. Calculate and write to the record file the following three values:

- the total number of GCPs in the residual
- the mean latitude of all GCPs
- the mean longitude of all GCPs

3. Calculate and write to the record file the following five values for the pre-fit residuals:

- Mean value of the cross-track residuals
- Mean value of the along-track residuals
- RMSE about the mean value for cross-track residuals
- RMSE about the mean value for along-track residuals
- Correlation coefficient between along- and cross-track residuals

4. Calculate and write to the record file the following five values for the post-fit residuals:

   - Mean value of the cross-track residuals
   - Mean value of the along-track residuals
   - Standard deviation for cross-track residuals
   - Standard deviation for along-track residuals
   - Correlation coefficient between along- and cross-track residuals

5. Compare the post-fit residuals with their standard deviations. Mark those GCPs with > 2 sigma residual as outliers.

6. Write to the outlier data file the information for each outlier GCP.

3.1.5.5.3.4   The Input Data

This module requires only one input data file, which is the output residual file from the Precision Correction Module. This file contains the residual information from iteration 0, which is the pre-fit residual, through the final iteration, which is the post-fit residual.

3.1.5.5.3.5   GCP Data Accuracy Requirement

The following properties of the GCP measurements are required to produce a meaningful precision correction solution and geodetic characterization result:

- The GCP coordinate should be in a system consistent with that of satellite ephemeris. WGS84 is preferred; NAD83 in North America is acceptable.
- The accuracy of the GCP coordinate should be on the level of 3 meters in horizontal and 20 meters in elevation to guarantee 0.1 pixel accuracy.
- A minimum number of 15 valid GCPs is required for reliable characterization (6/number of precision correction parameters + 9/reasonable number of degrees of freedom). To allow a certain portion of outlier editing, 25 to 50 input GCPs are suggested.

3.1.5.5.3.6   Use of the Output Data

The outlier information file is used to remove GCPs from both the GCP data file and the EPH data file for a precision correction solution (see the algorithm description for the precision correction solution module). A new round of precision correction solution and geodetic characterization is performed after the editing.

The record file is used as a measure of the absolute accuracy of the systematic product (pre-fit residual statistics) and precision product (post-fit residual statistics).

Trending analysis can be done on the parameters provided in the record file.

Error characterization for checkpoint data (GCPs not used for the precision correction solution) is done in a different round after the systematic model is updated with the precision correction parameters.

3.1.5.5.4 Geometric Accuracy

For the purposes of determining geometric accuracy, three tests are to be performed:

1. A visual inspection of the output image—The image is inspected visually for any geometric distortions. These distortions could include scan misalignment, artifacts produced by the resampling process such as ringing, or detector (line-to-line) discontinuity.
2. A plot of the residuals of the ground control points and their corresponding locations in the terrain-corrected image—This can be either a vector plot indicating the line/sample offsets or an image with the residuals color coded according to their magnitude superimposed on a paper product of the image.
3. A polynomial fit involving the ground control points and their corresponding locations in the terrain-corrected output product.

3.1.5.5.4.1 Polynomial Fit

A polynomial fit is used to estimate the amount of scaling in x and y, rotation, offset in x and y, and non-orthogonality angle associated with a terrain-corrected image. The polynomial takes the form of:

$$x_o = a + b*x_i + c*y_i$$
$$y_o = d + e*x_i + f*y_i$$

where:

x_o = output projection x coordinate (measured)
y_o = output projection y coordinate (measured)
x_i = input projection x coordinate (truth)
y_i = input projection y coordinate (truth)
a, b, c, d, e, f = transformation coefficients

The polynomial coefficients are determined from a least squares fit of the x and y coordinates of features located in the terrain-corrected product and their corresponding x and y coordinates for the same feature in the ground control source. The transformations that "correct" x_o, y_o to x_i, y_i are given by:

$$x_i = S_x (x - x_o) \cos(\theta) - S_y (y - y_o) \sin(\theta - \alpha)$$

The error or distortions can then be determined from the polynomial transformation coefficients by the following:

$$y_i = S_x(x - x_o)\sin(\theta) + S_y(y - y_o)\cos(\theta - \alpha)$$

$$S_x = \frac{\sqrt{e^2 + f^2}}{bf - ce}$$

$$S_y = \frac{\sqrt{b^2 + c^2}}{bf - ce}$$

$$theta = \tan^{-1}(\frac{-e}{f})$$

$$alpha = \tan^{-1}(\frac{-be - cf}{bf - ce})$$

S_x = scale error in x direction
S_y = scale error in y direction
alpha = non-orthogonality angle
theta = rotation angle
a = offset in x direction
d = offset in y direction
a, b, c, d, e, f = transformation coefficients

The scale error can be converted to pixels by the following:

(S_x–1) * (image pixel size in x)
(S_y–1) * (image pixel size in y)

Theta can be converted to pixels by the following:

theta * (size of image diagonal)
size of image diagonal = SQRT((image pixel size $_x$)^2 + (image pixel size $_y$) ^{2})

Alpha can be converted to pixels by the following:

alpha * (maximum pixel size)
maximum pixel size = MAX(image pixel size in x, image pixel size in y)

All of these values should be very close to zero, except S_x and S_y, which should be close to 1. If any of them are not zero, or if S_x and S_y are not close to 1, then they represent the amount of geometric distortion not corrected for by the system.

The residuals left after applying the polynomial coefficients to the terrain-corrected x and y coordinates represent higher order distortions. These values should also be very small.

This algorithm requires a terrain-corrected image produced from the IAS system, along with at least 50 well distributed ground control points. The PAN band provides the highest precision when calculating errors; however any band can be used in the process. All points, from both the image and the control source, need to be in the same projection with the same datum and pixel size. The control points and the points taken from the IAS terrain-corrected product should be to sub-pixel accuracy. The absolute accuracy of the points picked depends on the method chosen for point matching.

3.1.6 Variance or Uncertainty Estimates

The following sections contain error budget analyses for the key Landsat 7 ETM+ geometric accuracy requirements. These analyses were used to predict the Level 1Gs geodetic accuracy performance and to develop algorithm and support data accuracy requirements for the geometric calibration and characterization activities.

3.1.6.1   Level 1 Processing Accuracy

The purpose of the Level 1 Processing Accuracy requirement is to produce systematically corrected (1Gs) products that are accurate to within 250 meters per coordinate (one sigma) at nadir. This is stated in the IAS Element Specification as requirement 3.2.3.4.

An error budget for 1Gs geodetic accuracy performance can be built up using the expected accuracy of the data components that are part of the Landsat 7 ETM+ geometric model. These components include spacecraft ephemeris and attitude knowledge, spacecraft clock errors, knowledge and stability of the alignment of the ETM+ instrument to the Landsat 7 spacecraft, and knowledge and stability of the internal geometry of the ETM+ instrument. Accuracy bounds for most of these data elements are specified in the Landsat 7 System Specification. Allocations for ground processing and test point mensuration are included in the total geodetic accuracy error budget.

Errors in each of the geometric model data components were assumed to be independent zero mean random variables in the geodetic accuracy analysis. The relevant accuracy values were extracted from the Landsat 7 System Specification, converted to 1 sigma values, if necessary, and mapped to position errors in meters on the ground.   For along-track and across-track angular error values, this mapping was done by multiplying the angular error (in radians) by the nominal Landsat 7 above-ground height of 705 kilometers. The spacecraft clock error was converted to ground meters by multiplying the time error by a conservative estimate of the nominal Landsat 7 orbital velocity of 7.5 kilometers per second.

Two cases are considered in the geodetic accuracy error budget: (1) the expected accuracy at launch based on the pre-launch knowledge of the alignment between the Landsat 7 navigation base reference coordinate system and the ETM+ optical axis and (2) the expected accuracy in routine operations after the IAS has calibrated the instrument to spacecraft alignment. The 250-meter accuracy specification applies to both the along-track and across-track directions, but the presence of the spacecraft clock error makes the along-track direction the critical case. The at-launch, along-track accuracy

analysis is presented in Table 3-1. The columns in the table contain the following information:

- Column 1 identifies the data element error source.
- Column 2 shows the error contribution mapped to ground meters at nadir (1 sigma).
- Column 3 shows the error contribution in the units in which it is specified (1 sigma).
- Column 4 identifies the specification units.
- Column 5 shows the original source of the specified value since some are Root-Sum-Square (RSS) combinations of lower level components
- Column 6 shows where the error source was allocated at the system level (e.g., to the spacecraft, instrument, or ground system).

Source	Meters 1 σ	Spec 1 σ	Units	Comments	Segment
Vehicle Attitude	154	45.0	arc-sec	45.0 arc-sec (1 σ) (R, P, Y)	Satellite
Scan Mirror Repeatability	3	0.72	arc-sec	1.75 microradians * 2 (1 σ)	ETM+
Field Angle	3	0.78	arc-sec	RSS of:	
				0.18 arc-sec (1 σ) calibration	IAS
				0.76 arc-sec (1 σ) stability (0.2 pan)	ETM+
Alignment Knowledge (R, P, Y) - stability	58	17.0	arc-sec	RSS of:	
				8.0 arc-sec (1 σ) ETM+ to NBR	Satellite
				15.0 arc-sec (1 σ) LOS to ETM+	ETM+
Alignment Uncertainty (R, P, Y) - calibration	846	247.4	arc-sec	RSS of:	
				720.0 arc-sec (3 σ) pre-launch	Satellite
				60.0 arc-sec (1 σ) pre-launch	ETM+
Ephemeris (I, C, R)	133	133.33	meter	400 m (3 σ) from sum of:	
				375 m (3 σ) predict	FDF
				20 m (3 σ) interpolation	Satellite
Jitter (R, P, Y)	2	0.44	arc-sec	RSS of:	
				0.30 arc-sec (1 σ) low-frequency	Satellite
				0.30 arc-sec (1 σ) high-frequency	Satellite
				0.134 arc-sec (1 σ) high-frequency	ETM+
Timing (along-track)	38	5	msec	15 msec (3 σ) random error	Satellite
Mensuration	15	15	meter	0.5 pixels (1 σ)	User
Processing Error	2	2	meter	Model budget	IAS
RSS Estimate	**873**				
RSS Margin	**-**				
Specification	**250**				

Table 3-1. At-Launch 1Gs Geodetic Error Budget

As Table 3-1 demonstrates, the 250-meter geodetic accuracy requirement is not expected to be met based on the at-launch knowledge of the ETM+ instrument to Landsat 7 spacecraft alignment.

The Landsat 7 System Specification requires the IAS to estimate the on-orbit alignment to an accuracy of 24 arc-seconds (as compared to the pre-launch accuracy of about 248 arc-seconds). This improvement in alignment knowledge accuracy allows the geodetic accuracy budget to meet the 250-meter specification with a significant margin. Table 3-2 updates the along-track accuracy analysis with the post-calibration alignment knowledge accuracy. The table column definitions are the same as for Table 3-1.

Source	Meters 1 σ	Spec 1 σ	Units	Comments	Segment
Vehicle Attitude	154	45.0	arc-sec	45.0 arc-sec (1 σ) (R, P, Y)	Satellite
Scan Mirror Repeatability	3	0.72	arc-sec	1.75 microradians * 2 (1 σ)	ETM+
Field Angle	3	0.78	arc-sec	RSS of:	
				0.18 arc-sec (1 σ) calibration	IAS
				0.76 arc-sec (1 σ) stability (0.2 pan)	ETM+
Alignment Knowledge (R, P, Y) - stability	58	17.0	arc-sec	RSS of:	
				8.0 arc-sec (1 σ) ETM+ to NBR	Satellite
				15.0 arc-sec (1 σ) LOS to ETM+	ETM+
Alignment Uncertainty (R, P, Y) - calibration	82	24.0	arc-sec	24.0 arc-sec (1 σ) (R, P, Y)	IAS
Ephemeris (I, C, R)	133	133.33	meter	400 m (3 σ) from sum of:	
				375 m (3 σ) predict	FDF
				20 m (3 σ) interpolation	Satellite
Jitter (R, P, Y)	2	0.44	arc-sec	RSS of:	
				0.30 arc-sec (1 σ) low-frequency	Satellite
				0.30 arc-sec (1 σ) high-frequency	Satellite
				0.134 arc-sec (1 σ) high-frequency	ETM+
Timing (along-track)	38	5	msec	15 msec (3 σ) random error	Satellite
Mensuration	15	15	meter	0.5 pixels (1 σ)	User
Processing Error	2	2	meter	Model budget	IAS
RSS Estimate	**231**				
RSS Margin	**97**				
Specification	**250**				

Table 3-2. 1Gs Geodetic Error Budget after Alignment Calibration

3.1.6.2   Geometric Calibration Accuracy

The development of the three IAS geometric calibration algorithms included estimation error analyses to ensure that the relevant geometric parameters could be estimated with sufficient accuracy to justify updating the pre-launch knowledge in post-launch releases of the Calibration Parameter File. The results of these analyses are presented in the following sections.

3.1.6.2.1   Sensor Alignment Calibration Accuracy

As noted above, the IAS is required to calibrate the ETM+ instrument to Landsat 7 spacecraft alignment knowledge to an accuracy of 24 arc-seconds to achieve a 1Gs product geodetic accuracy of 250 meters. This is stated in the IAS Element Specification as requirement 3.2.3.12. The IAS sensor alignment algorithm uses the results of a series of precision correction solutions to extract the systematic alignment bias from the random scene-to-scene attitude error, as described above. The goal of the sensor alignment error analysis is to determine the number of calibration scenes required to achieve the 24 arc-second alignment knowledge accuracy as a function of the accuracy of the supporting definitive ephemeris and ground control data.

The precision correction solution model is used to perform an error propagation analysis to predict the accuracy of the pointing error (alignment plus attitude) estimates produced by a single precision correction solution. This analysis is based on the assumption that all inputs are accurate to their pre-launch specification values and that 14 well distributed ground control points are successfully measured with an accuracy of 15 meters (ground position error plus image mensuration error) for each point.

The input accuracy assumptions and the resulting pointing error accuracy estimates are summarized in Table 3-3.

Input	1 σ Error	Rationale
Along-Track Position	63 meters	RSS of 50-meter estimate of FDF definitive ephemeris accuracy and 38-meter spacecraft clock error
Across-Track Position	50 meters	Estimate of FDF definitive ephemeris accuracy
Radial Position	50 meters	Estimate of FDF definitive ephemeris accuracy
Roll	252 arc-sec	RSS of 247 arc-sec pre-launch alignment knowledge error, 17 arc-sec alignment stability error, and 45 arc-sec attitude knowledge error
Pitch	252 arc-sec	RSS of 247 arc-sec pre-launch alignment knowledge error, 17 arc-sec alignment stability error, and 45 arc-sec attitude knowledge error
Yaw	252 arc-sec	RSS of 247 arc-sec pre-launch alignment knowledge error, 17 arc-sec alignment stability error, and 45 arc-sec attitude knowledge error
Ground Control	15 meters	RSS of 12-meter position error (1:50,000-scale map source) and 7.5-meter (0.5 panchromatic pixel) image mensuration error
Roll Estimate	**13.1 arc-sec**	A posteriori error estimate
Pitch Estimate	**16.6 arc-sec**	A posteriori error estimate
Yaw Estimate	**16.7 arc-sec**	A posteriori error estimate

Table 3-3. Pointing Error Estimate Accuracy Analysis—Inputs and Results

The roll estimate accuracy is somewhat better than the pitch and yaw estimates, primarily because the across-track roll must be separated from the across-track position error, while both the pitch and yaw must be separated from the along-track position error, which is less accurate due to the spacecraft clock error.

The roll, pitch, and yaw pointing error estimates generated by the precision correction process represent the combined effect of static alignment error, attitude error, and dynamic alignment error, which can be expressed as:

pointing error observation = attitude + static alignment + dynamic alignment

It is the static alignment error that we seek to calibrate. Rearranging the previous equation yields:

static alignment = pointing error observation – attitude – dynamic alignment

The attitude and dynamic alignment errors are assumed to be zero mean, so the best estimate of the static alignment error is the observed pointing error estimated by the precision correction solution. The variance of the static alignment estimate is the sum of the variance of the three terms on the right side:

$$\sigma^2_{alignment} = \sigma^2_{observation} + (45)^2 + (17)^2$$

where:

> 45 arc-seconds = expected attitude standard deviation
> 17 arc-seconds = expected dynamic alignment standard deviation

Table 3-4 summarizes the resulting standard deviations for the three alignment error components.

Component	Standard Deviation (Single Estimate)
Roll	49.9 arc-seconds
Pitch	50.9 arc-seconds
Yaw	50.9 arc-seconds

Table 3-4. Alignment Estimate Accuracy

As multiple test scene results become available, the observed pointing errors are combined using a Kalman filter, as described above. To a first approximation, the effect on the resulting alignment estimate is to improve the accuracy as the square root of the number of test scenes processed. The resulting improvement in alignment knowledge accuracy as a function of the number of calibration scenes processed is shown in Figure 3-41.

The figure suggests that the 24 arc-second alignment accuracy requirement can be met after five calibration scenes have been processed. The accuracy estimates based on this relatively simple model of the alignment estimation process have been confirmed by processing simulated data (with known alignment biases introduced) through the precision correction and sensor alignment prototype software and analyzing the results. These tests have shown that the alignment knowledge can be expected to reach the 24 arc-second accuracy threshold after five or six calibration test scenes have been processed.

3.1.6.2.2   Scan Mirror Calibration Accuracy

Although there is no formally specified accuracy requirement for the scan mirror calibration process, an accuracy threshold of 1.75 microradians (in mirror space) has been selected as the goal for mirror profile estimation accuracy. This value corresponds to the scan mirror repeatability specification called out in the Landsat 7 System Specification in 3.7.8.1.17.4. Determining the scan mirror profile to this accuracy keeps the systematic mirror profile knowledge within the random variation of the scan mirror motion.

Error propagation analysis (based on a set of relatively conservative assumptions regarding the accuracy of the input data) was used to determine the number of ETM+ scans and the number of test points per scan that must be correlated with a DOQ reference image to achieve a 1.75-microradian estimation accuracy. The input data and image-to-image correlation accuracy assumptions are presented in Table 3-5.
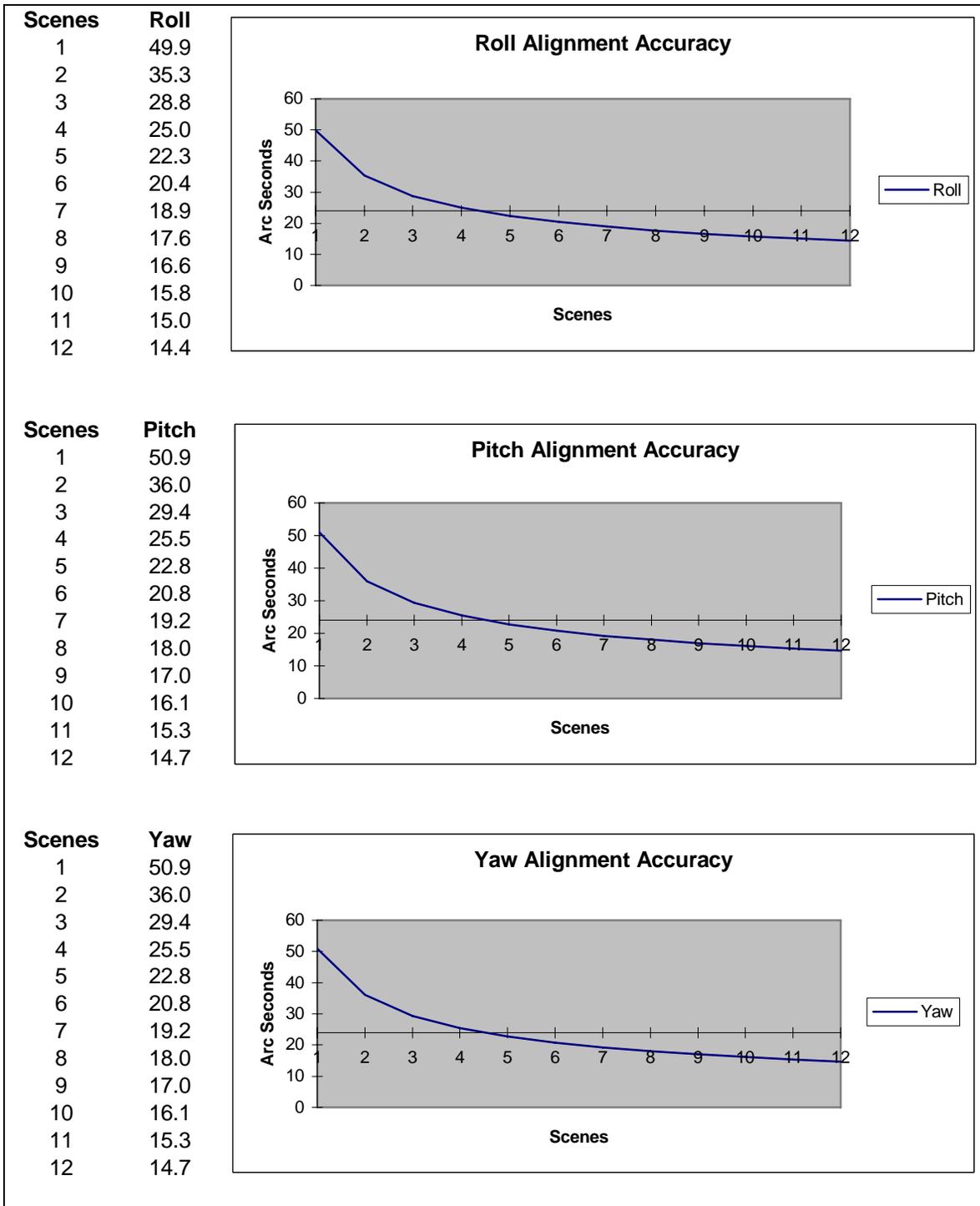
Source	1 σ error	Rationale
DOQ Reference Image	8.5 urad	Corresponds to 6-meter DOQ accuracy spec.
ETM+ 1Gt Image	21.25 urad	Corresponds to 1 panchromatic pixel
Correlation Error	10.625 urad	Corresponds to 0.5 panchromatic pixel
Net Observation Error	25.23 urad	RSS of DOQ, 1Gt, and correlation errors
Obs Error - Mirror Space	12.62 urad	One half the object space value above
A priori Coefficient Error	42.5 urad	Corresponds to 1 reflective band pixel
Number of Points per Scan	60	Corresponds to 1 per geometric grid cell

Table 3-5. Scan Mirror Calibration Accuracy Analysis Input Assumptions

The error propagation analysis assumed a set of test points that were well distributed as a function of scan angle to form the normal equations that were used to estimate the mirror profile Legendre polynomial coefficients. The normal equations were then inverted to form the predicted a posteriori error covariance matrix from which variance estimates for the Legendre coefficients were extracted. These coefficient variances were then summed to compute the net mirror profile uncertainty. The number of test points used to construct the normal equation matrix was then varied until the net mirror profile uncertainty was less than 1.75 microradians.

Two separate cases were investigated:

1. Estimation accuracy for an unconstrained fifth-order polynomial with six unknown coefficients
2. Estimation accuracy for a fifth-order polynomial constrained to meet the beginning of scan, end of scan, and mid-scan constraints imposed by the ETM+ mid-scan correction process, as described in the scan mirror calibration algorithm section

Scenes	Roll
1	49.9
2	35.3
3	28.8
4	25.0
5	22.3
6	20.4
7	18.9
8	17.6
9	16.6
10	15.8
11	15.0
12	14.4

Roll Alignment Accuracy

Scenes	Pitch
1	50.9
2	36.0
3	29.4
4	25.5
5	22.8
6	20.8
7	19.2
8	18.0
9	17.0
10	16.1
11	15.3
12	14.7

Pitch Alignment Accuracy

Scenes	Yaw
1	50.9
2	36.0
3	29.4
4	25.5
5	22.8
6	20.8
7	19.2
8	18.0
9	17.0
10	16.1
11	15.3
12	14.7

Yaw Alignment Accuracy

Figure 3-41. Alignment Estimation Accuracy vs. Number of Scenes

In the unconstrained case, it is necessary to process 40 scans (forward or reverse) to achieve the 1.75-microradian accuracy target. This implies that the DOQ reference images must be a full scan wide (cross-track) and at least 80 scans long (along-track). Allowing for some loss of data due to cloud cover and poor correlation targets leads to a

requirement for 90–100 scans, or about one-quarter of a Landsat Worldwide Reference System (WRS) scene. Applying the scan constraints simplifies the estimation problem by reducing the number of degrees of freedom in the solution and, consequently, reduces the data requirements to 15 forward or reverse scans. This can be satisfied by a DOQ reference image that is a full scan wide and 30 scans long, or about one-eighth of a WRS scene.

Based on this analysis, we conclude that a single scan mirror calibration test site covering at least one-quarter of a WRS scene would be sufficient to provide usable results. The goal is to provide at least two test sites, each covering a full WRS scene, if possible. As noted above, the input data accuracy assumptions are conservative, so using a full scene reference image, a more accurate 1Gt product, and a better image-to-image correlation, the performance of the algorithm with real Landsat 7 data is expected to exceed the 1.75-microradian design threshold.

3.1.6.2.3 Band Placement Calibration Accuracy

The IAS is required to determine band center field angles to an accuracy of 0.18 arc-seconds (1 σ per coordinate) per IAS Element Specification requirement 3.2.3.10. The band-to-band registration assessment algorithm uses test points that appear in all eight ETM+ bands and performs cross-band image correlation to measure any residual sub-pixel band-to-band offsets. These measurements are then used in the band placement calibration algorithm to estimate band center offsets in the ETM+ focal planes, relative to band 8 (panchromatic), which is treated as the reference band. These algorithms are described above.

In the course of developing the band-to-band registration assessment algorithm, it was determined that, given a suitable test scene, it is possible to perform cross-band image correlation to an accuracy of approximately 0.1 pixel, using pre-selected test points, and to an accuracy of approximately 0.25 pixel, using randomly generated test points. When correlating bands of differing resolution (i.e., bands 6 or 8 to any of the others), the measurements are performed at the coarser band resolution. Thus, the 0.1 or 0.25 pixel measurement accuracy corresponds to a larger angular uncertainty for band offset measurements for band pairs that include band 6. This makes band 6 the critical case that drives the number of band-to-band test points required to meet the 0.18 arc-second accuracy specification.

An error propagation analysis was used to determine the number of test points required to achieve a band center location accuracy of 0.18 arc-seconds as a function of the expected accuracy of the test point measurements. This analysis assumed that all test points were measured in all band pairs and that the mapping between image pixel measurements and focal plane offsets could be approximately modeled as a constant scale factor. The availability of measurements from seven band pair combinations makes the band center location solution overdetermined. Therefore, a least squares procedure was set up to form a normal equation matrix, which could be used to estimate band center offsets from test point measurements. These normal equations were then inverted to compute the predicted a posteriori covariance matrix for the band center estimates.

The square roots of the diagonal terms of this matrix were taken to be estimates of the band center location standard deviations. The number and accuracy of the input test point measurements were then varied to examine the sensitivity of the band center estimate standard deviations to the input data volume and accuracy.

As expected, the predicted standard deviation for the band 6 location was highest. Figure 3-42 shows a plot of the number of test points needed to achieve a predicted standard deviation of 0.18 arc-seconds versus the accuracy of the input test point measurements.

As the figure shows, the accuracy requirement can be met with 20–25 high-quality, hand-picked test points or with 125–150 automatically generated test points. The accuracy of hand-picked points should be more reliable, since the analyst selecting the test points can ensure their quality and consistency.

Based on the relatively small number of test points needed (if they are sufficiently accurate), we decided to use approximately 25 analyst-selected test points to perform the band-to-band registration assessment and subsequent band placement calibration. The error propagation analysis predicts a band center location accuracy of 0.09 arc-seconds for bands 1–5 and 7, and 0.15 arc-seconds for band 6. Band 8 is the reference and is assumed to be known exactly. This meets the requirement for all bands, if the test point measurements are accurate to 0.12 pixels, and for all but the thermal band, if the measurements are accurate to 0.19 pixels.

Test Point Accuracy (pixels)	Test Points Needed
0.50	407
0.45	332
0.40	263
0.35	202
0.30	149
0.25	103
0.20	66
0.15	37
0.10	17

Band Placement Calibration
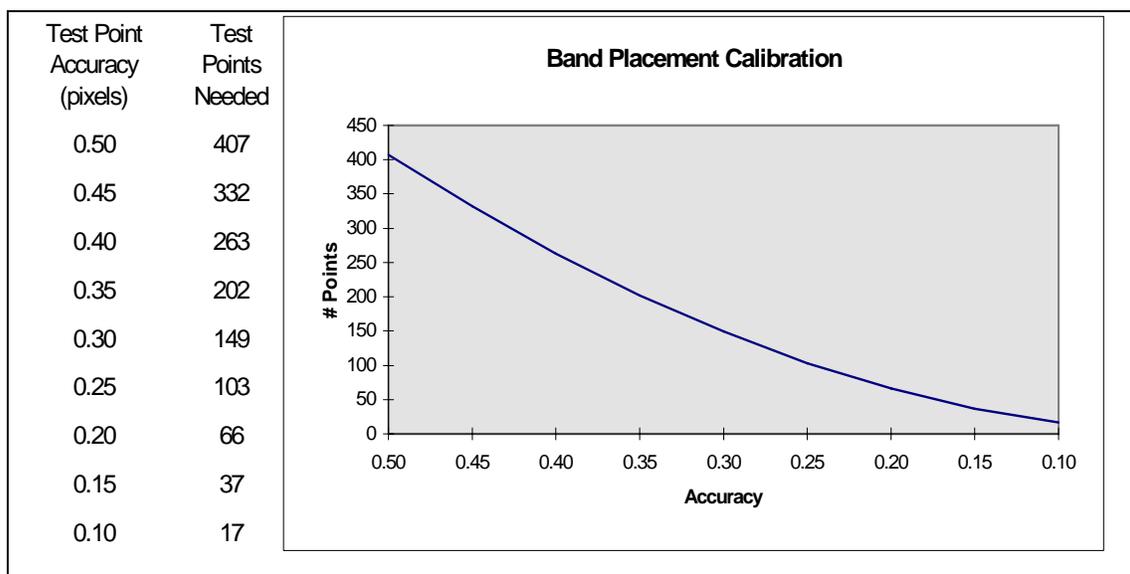
Figure 3-42. Band-to-Band Test Point Number vs. Accuracy

3.1.6.3   Geometric Characterization Accuracy

Only two of the four geometric characterization algorithms implemented in the Landsat 7 IAS have quantifiable accuracy requirements. The geometric accuracy assessment is

primarily a qualitative evaluation of the internal distortions present in ETM+ images. Geodetic accuracy assessment provides a verification of the Level 1Gs product geodetic accuracy requirement analyzed above. Test point mensuration error was included in the error budget for Level 1 processing geodetic accuracy to account for the contribution of the assessment process. The band-to-band and image-to-image registration accuracy assessments have specific performance requirements (called out in the IAS Element Specification), which are analyzed in the following sections.

3.1.6.3.1   Band-to-Band Registration Accuracy

The IAS is required to perform band-to-band registration to an accuracy of 0.28 pixels (at the ground sample distance of the coarser of the two bands) per coordinate at the 90% confidence level. This requirement is stated in the IAS Element Specification as requirement 3.2.3.6. The error budget for band-to-band registration includes the high-frequency jitter displacements that can occur in the time interval over which the eight ETM+ bands sample the same ground location. Since the maximum time between band samples is approximately 2 milliseconds, which corresponds to the jitter sensor sampling period, the band-to-band jitter error is essentially the error in the difference between two jitter measurements. Other components of the band-to-band registration error include band field angle errors, processing model errors, and test point mensuration errors.

Table 3-6 shows the error components combined into an error budget for the band-to-band registration assessment process. The table columns contain the following information:

- Column 1 identifies the data element error source.
- Column 2 contains the total effect of the error source in units of 30-meter pixels at the 90% confidence level (this is equal to the square root of two times the error for one band for the errors which are independent for all bands).
- Column 3 contains the contribution of the error source to a single band in units of 30-meter pixels at the 90% confidence interval.
- Column 4 contains the original specified or estimated error source one sigma accuracy.
- Column 5 contains the units of the original error specification.
- Column 6 provides a brief text explanation of the error estimate source.
- Column 7 shows where the error source is allocated.

The field angle stability and test point mensuration contributions are relative errors which are only counted once. High frequency jitter, field angle knowledge, and processing errors apply independently to all bands and are therefore counted twice in the band-to-band error budget.

Based on the analysis shown in Table 3-6 the 0.28 pixel (90%) requirement can be met but there is little error budget margin. Manually selected test points must be used to ensure the highest correlation accuracy since relaxing the correlation accuracy to 0.11 pixels exceeds the registration budget.

Source	Both Bands	30-m pixels (90%)	Spec 1 σ	Units	Comments	Segment
High-frequency Jitter	0.079	0.056	0.3	arc-sec	0.30 arc-sec (1 σ)	Satellite
ETM+ Jitter (high-frequency)	0.035	0.025	0.134	arc-sec	0.134 arc-sec (1 σ) high-frequency	ETM+
Field Angle Knowledge	0.048	0.034	0.18	arc-sec	0.18 arc-sec (1 σ) calibration	IAS
Field Angle Stability	0.143	0.143	0.76	arc-sec	0.76 arc-sec (1 σ) stability (0.2 pan)	ETM+
Processing Error	0.140	0.099	1.8	meter	Model budget	IAS
Mensuration	0.164	0.164	0.1	pixel	Correlation accuracy	IAS
RSS Estimate	**0.277**	**(90%)**				
RSS Margin	**0.041**	**(90%)**				
Specification	**0.280**	**(90%)**				

Table 3-6. Band-to-Band Registration Error Budget

3.1.6.3.2 Image-To-Image Registration Accuracy

The IAS has a requirement to perform image-to-image registration to an accuracy of 0.4 pixels (at the multi-spectral band pixel ground sample distance of 30 meters) per coordinate at the 90% confidence level. This requirement is stated in the IAS Element Specification as requirement 3.2.3.8. The error budget for image-to-image registration includes the high frequency components of the overall Landsat 7 ETM+ geometric error budget which model the internal geometric variability unique to each image, the expected accuracy of the precision correction process which uses image-to-image tie points to perform the image registration, and the expected mensuration accuracy for the registration assessment test points.

The high frequency internal errors are a subset of the geodetic error budget components presented in Table 3-7 and include the jitter, scan mirror variability, and modeling/processing errors. The image-to-image registration correction model accuracy is required to be 3.6 meters (1 σ) according to the IAS Element Specification requirement 3.2.3.9. The same estimation error covariance propagation model used to analyze the sensor alignment estimation accuracy was used to verify this requirement. In this case, the expected accuracy of the PCD ephemeris and attitude data was combined with 20

ground control points assumed to be accurate to 10 meters (corresponding to control from a 1:24,000-scale map and 0.5 pan pixel identification/mensuration error) to yield predicted post-fit ephemeris and attitude accuracy estimates. These estimates were then propagated to ground coordinate accuracy for a range of scan angles and then root-sum-squared to establish an accuracy estimate for the precision correction in ground space. This procedure resulted in an accuracy estimate of 3.5 meters (1 σ) for the precision correction process. The specification value of 3.6 meters was used for the image-to-image error budget.

The final component of the error budget is the image-to-image test point correlation accuracy. Testing with Landsat 5 data has shown that correlation accuracy of 0.1 pixel is achievable for selected targets. Pre-selected test points will be used to evaluate the image-to-image registration accuracy.

Table 3-7 shows these components combined into an error budget for the image-to-image registration assessment process. The table columns contain the following information:

- Column 1 identifies the data element error source.
- Column 2 contains the total effect of the error source in units of 30-meter pixels at the 90% confidence level (this is equal to the square root of two times the error for one image for those errors which that contained in both the reference and object images).
- Column 3 contains the contribution of the error source to a single image in units of 30-meter pixels at the 90% confidence interval.
- Column 4 contains the original specified or estimated error source one sigma accuracy.
- Column 5 contains the units of the original error specification.
- Column 6 provides a brief text explanation of the error estimate source.
- Column 7 shows where the error source is allocated.

The high-frequency errors are independent for the two images and therefore contribute twice, whereas the precision correction and mensuration errors apply to the process of relating the two images to each other and therefore are counted only once.

Source	Both Images	30-m pixels (90%)	Spec 1 σ	Units	Comments	Segment
High-frequency Jitter	0.079	0.056	0.3	arc-sec	0.30 arc-sec (1 σ)	Satellite
Low-frequency Jitter	0.079	0.056	0.3	arc-sec	0.30 arc-sec (1 σ)	Satellite
ETM+ Jitter (high-frequency)	0.035	0.025	0.134	arc-sec	0.134 arc-sec (1 σ) high freq	ETM+
Scan Mirror Repeatability	0.191	0.135	0.72	arc-sec	1.75 microradians * 2 (1 σ)	ETM+
Processing Error	0.140	0.099	1.8	meter	Model budget	IAS
Correction Model Error	0.198	0.198	3.6	meter	Estimation error	IAS
Mensuration	0.164	0.164	0.1	pixel	Correlation accuracy	IAS
RSS Estimate	**0.318**	**(90%)**				
RSS Margin	**0.243**	**(90%)**				
Specification	**0.400**	**(90%)**				

Table 3-7. Image-to-Image Registration Error Budget

Based on the analysis shown in Table 3-7, the 0.4-pixel (90%) requirement can be met, but there is little error budget margin. Pre-selected test points must be used to ensure the highest correlation accuracy, since relaxing the correlation accuracy to 0.14 pixels exceeds the registration budget. Similarly, it is necessary to use the precision correction process to fit the object image directly to the reference image, rather than fitting both to a separate control source, leading to independent correction model errors, which, when combined, would exceed the 3.6-meter image-to-image correction model budget.

Section 4   Constraints, Limitations, Assumptions

4.1   Atmospheric Refraction Correction

The atmosphere causes a ray of light to bend toward the satellite as it propagates through the atmosphere. The amount of bending depends on the angle of propagation and the density of the atmosphere at a given instant. For Landsat 7, the magnitude of this effect is less than 1 meter at end of scan. Currently, this effect is not accounted for.

4.2   Resampling

Resampling Weight Tables (i.e., for cubic convolution, Modulation Transfer Function [MTF], or other resampling kernels) are calculated at 1/32 pixel increments. These weights assume evenly spaced pixels and are used in the sample direction.

Spline weights (used for resampling over the scan-gap where the pixels are not necessarily evenly spaced) are also calculated at 1/32 pixel increments. These weights are applied in the line direction in the area between two scans. The weights are set up for a shift of 0.0 to the maximum gap possible from the system. A value of 3.0 covers the maximum gap possible in the Landsat 5 satellite. A value of 6.0 is needed for the Landsat 7 satellite to cover the panchromatic band. When the panchromatic band is used for Landsat 7, it could cause some anomalies in cases of large varying grey values between pixels that are located in an area with a large scan gap.

4.3   Speed of Light Correction

Due to the non-infinite velocity of light, the velocity of a point on the surface of the Earth and the velocity of the satellite cause pixel location errors if not accounted for. The speed of light correction due to the Earth's rotation is sub-meter and is currently not accounted for.

References

Reference 1
Martin Marietta Astro Space. 23007702-IVH. Landsat 7 System Data Format Control Book (DFCB). Volume IV. Revision H. February 1998.

Reference 2
Martin Marietta Astro Space. PS23007610B. Landsat 7 System Program Coordinates System Standard. Revision B. December 1994.

Reference 3
NASA/GSFC. 430-15-01-001-1. Landsat 7 System Image Assessment System (IAS) Element Specification. Revision 1. February 1998.

Reference 4
EOSDIS Core System (ECS) Project. 445-TP-002-002. Theoretical Basis of the Science Data Processing Toolkit Geolocation Package for the ECS Project. May 1995.

Reference 5
Goddard Space Flight Center. 430-L-0002-H. Landsat 7 System Specification. Revision H. June 1996.

Reference 6
Press, W.H., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. Numerical Recipes in C: The Art of Scientific Computing. 2nd edition. Cambridge: Cambridge University Press, 1992.

Reference 7
Defense Mapping Agency (DMA) WGS84 Development Committee. DMA TR 8350.2-A. DMA Technical Report, Supplement to Department of Defense World Geodetic System 1984 Technical Report. December 1987.

Reference 8
Snyder, J.P. Map Projections—A Working Manual. U.S. Geological Survey Professional Paper 1395. Washington: U.S. Government Printing Office, 1987.

Reference 9
Hughes Santa Barbara Research Center. Landsat 7 Space Segment Technical Description Document—Enhanced Thematic Mapper Plus. May 1994. Updated by NASA Goddard Space Flight Center. September 1995.

Reference 10

Sehn, G.J., and S.F. Miller. "LANDSAT Thematic Mapper Attitude Data Processing."
Presented at the AIAA/AAS Astrodynamics Conference. Seattle, WA. August 1984.

Reference 11

Landsat to Ground Station Interface Description. Revision 9. Greenbelt, MD: Goddard
Space Flight Center, 1986.

Reference 12

Park, S.K., and R.A. Schowengerdt. "Image Reconstruction by Parametric Cubic
Convolution." Computer Vision, Graphics and Image Processing 23.3 (1983): 258-
272.

Reference 13

Rosborough, G.W., D.G. Baldwin, and W.J. Emery. "Precise AVHRR Image
Navigation." IEEE Transaction on Geoscience and Remote Sensing 32.3 (1994):
644-657.

Reference 14

Rao, C.R. Linear Statistical Inference and Its Applications. New York: John Wiley and
Sons, Inc., 1973.

Reference 15

Sahin, M., P.A. Cross, and P.C. Sellers. "Variance Component Estimation Applied to
Satellite Laser Ranging." Bulletin Geodesique 66 (1992): 284-295.

Reference 16

Yuan, Dah-ning. "Optimum Determination of the Earth's Gravitational Field from
Satellite Tracking and Surface Measurements." Dissertation. The University of Texas
at Austin. 1991.

Reference 17

Snyder, J.P. Space Oblique Mercator Projection Mathematical Development. U.S.
Geological Survey Bulletin 1518. Washington: U.S. Government Printing Office,
1981.

Reference 18

USGS/EROS. LAS 6.0 Geometric Manipulation Package Overview Document.

Reference 19

USGS/EROS. LAS 6.0 Programmer's Document.

Reference 20

Escobal, P.R. Methods of Orbit Determination. 2nd edition. New York: John Wiley and
Sons, Inc., 1976.

Reference 21

Dvornychenko, V.N. "Bounds on (Deterministic) Correlation Functions with Application to Registration." IEEE Transactions on Pattern Analysis and Machine Intelligence 5.2 (1983): 206-213.

Reference 22

Steinwand, D., and C. Wivell. Landsat Thematic Mapper Terrain Corrections in LAS. HSTX Inter-Office Memo OAB8-21. USGS/EROS. August 1993.

Reference 23

Brown, R.G. Introduction to Random Signal Analysis and Kalman Filtering. New York: John Wiley and Sons, Inc., 1983.

Reference 24

NASA/GSFC. 430-15-01-002-0. Landsat 7 System Calibration Parameter File Definition. Revision 1. January 1998.

Reference 25

Burt P., and E. Adelson. "The Laplacian Pyramid as a Compact Image Code." IEEE Transactions on Communications Com-31.4 (1983): 532-540.

Reference 26

Prakash, A., and E.P. Beyer. Landsat D Thematic Mapper Image Resampling for Scan Geometry Correction.

Reference 27

Beyer, E.P. An Overview of the Thematic Mapper Geometric Correction System.

Reference 28

Wolberg, G. Digital Image Warping. Los Alamitos, CA: IEEE Computer Science Press, 1990.

Reference 29

Gonzalez, R.C., and P. Wintz. Digital Image Processing. 2nd edition. Reading, MA: Addison-Wesley Publishing Company, 1987.

Reference 30

Fischel, D. "Validation of the Thematic Mapper Radiometric and Geometric Correction Algorithms." IEEE Transactions on Geoscience and Remote Sensing GE-22.3 (1984): 892-901.

Reference 31
Golub, G.H., and C.F. Van Loan. Matrix Computations. 2nd edition. Baltimore: John Hopkins University Press, 1989.

Reference 32
UGSS/EROS. Landsat 7 ETM+ Geometric Calibration Plan. December 1996.

Reference 33
Goodman, D.J., and M.J. Carey. "Nine Digital Filters for decimation and Interpolation." IEEE Transactions on Acoustics, Speech, and Signal Processing 25.2 (1977): 121-126.

Reference 34
Haykin, Simon. Adaptive Filtering Theory. Englewood Cliffs, NJ: Prentice Hall, 1991.

Reference 35
Martin Marietta Astro Space. PS23007911A. Landsat 7 Celestial Sensor Assembly, Including Star Transit Processing Electronics and Celestial Sensor Sun Shield. September 1994.

Reference 36
Cook, R.D. "Detection of Influential Observations in Linear Regression." Technometrics 19.1 (1977): 15-18.

Reference 37
Goddard Space Flight Center. IAS Radiometry Algorithm 6.6a. "Outliers in a Regression." 1996.

Reference 38
G/S-ICD-2040. EOSAT Ground Segment Interface Control Document. Volume 2D. Revision D. November 1996.

Reference 39
Gore, Robert. "Landsat 4/5 Attitude Jitter Processing Analysis." Marietta, GA: Lockheed/Martin. Interoffice memo.

Reference 40
Gore, Robert. "Landsat 7 System ETM+ Process Algorithms (Attitude Jitter Processing)." Contract No. NAS5-32633.