SPOT
IMAGE

SPOT SATELLITE GEOMETRY HANDBOOK

S-NT-73-12-SI

Edition 1 – Revision 0

2002-01-15

5 rue des Satellites

31030 Toulouse Cedex 4,France

http://www.spotimage.fr

This document was prepared by Serge RIAZANOFF (GAEL Consultant).

The document was subjected to the technical approval of a working revision group established by experts of Spot Image, CNES and IGN.

Contents

TABLE OF FIGURES

1 INTRODUCTION

1.1 Purpose of this document

This document describes the geometry by which Earth observation images of SPOT missions are acquired. The proposed viewing model allows to produce enhanced products at various processing levels. Equations are given to generate standard level 1A, 1B and 2A images. The provided model allows computing and controlling orthorectified images to produce space maps with high accuracy location on the Earth.

This handbook deals with the whole SPOT mission series (SPOT123, 4 and 5) summarizing the differences between these missions in term of payload, auxiliary data and image geometry.

Being first intended to software editors, the handbook pays a particular attention to precisely locate the auxiliary data involved within the equations to be implemented. This location is given in reference to the CEOS format (CAP) and to the new DIMAP format of SPOT products.

1.2 Document Overview

- Section 1 provides all the conventions (reference documents, abbreviation, definitions and symbols) used in the document;
- Section 2 summarizes the history and differences of SPOT missions ;
- Section 3 presents the main features (processing levels and data format) of SPOT products ;
- Section 4 defines the viewing geometry model that will be used in the next sections ;
- Section 5 provides an algorithm enabling to produce level 2A images ;
- Section 6 defines the two methods that have been used to produce level 1B images ;
- Appendix A gives the formula and algorithm required in sections 4 and 5 to convert the coordinates between the cartesian geocentric and the geodetic systems ;
- Appendix B defines where the symbols used within the formula of this document can be retrieved within the CEOS and DIMAP formats.

1.3 Reference documents

This section lists some of the documents which have been used to edit this handbook and which are considered as being references within their respective domains. All these documents are not necessarily supposed to be public.

R-1	S-NT-73-11-SI	*SPOT Image level 1B geometric modelization* Edition 1, Revision 1 – March 13th, 2000 SPOT IMAGE
R-2	S-IF-O/E-10-SI	*SPOT to Direct Receiving Station Interface Document* Edition 2, Revision 0 – December 1st, 1997 SPOT IMAGE
R-3	S4-ST-73-01-SI	*The SPOT Scene Standard Digital Product Format* Edition 1, Revision 2 – November 17th, 1997 SPOT IMAGE
R-4	DIMAP RC1	*DIMAP 1.0 Release Candidate 1* September 2001 SPOT IMAGE http://www.spotimage.fr/accueil/proser/geninfo/format/dimap/welcome.htm

R-5	TR8350.2	*World Geodetic System 1984 – Its Definition and relationships with Local Geodetic Systems – Third edition, Amendment 1* January 3rd, 2000 NIMA ftp://164.214.2.65/pub/gig/tr8350.2/wgs84fin.pdf
R-6	NT/G N°8	*Formulaire pour transformations de coordonnées tridimensionnelles cartésiennes ou géographiques entre 2 systèmes géodésiques* January, 1995 C. Boucher, Institut Géographique National
R-7	REC-xml-20001006	*Extensible Markup Language (XML) 1.0 (Second Edition)* W3C Recommendation, Version 1.0 – October 6, 2000 World Wide Web Consortium http://www.w3.org/TR/2000/REC-xml-20001006
R-8	REC-xpath-19991116	*XML Path Language (XPath), Version 1.0* November 16th, 1999 W3C Recommendation http://www.w3.org/TR/xpath
R-9	SI/AT/85.0113 v.A.A rev.3	*The SPOT Standard CCT Format* November 24th, 1987 SPOT IMAGE

1.4 Abbreviations and Acronyms

This section sets the definition of all abbreviations and acronyms used within this document. A special attention has been paid to inherit abbreviations, acronyms and their definitions from international standards as ISO, ANSI or ECSS.

ANSI	American National Standards Institute
AOCS	Attitude and Orbit Control System
CAP	Centre d'Archivage et de Pré-traitement
CCD	Charged Coupled Device
CNES	Centre National d'Études Spatiales
DEM	Digital Elevation Model
DORIS	Doppler Orbitography and Radiopositioning Integrated by Satellite
DIODE	Détermination Immédiate d'Orbite par DORIS Embarqué
ECSS	European Cooperation for Space Standardization
EGM96	Earth Gravitational Model 1996
GCP	Ground Control Points
GPS	Global Positioning System
HRG	Haute Résolution Géométrique (High Geometric Resolution)
HRS	Haute Résolution Stéréoscopique (High Stereoscopic Resolution)
ISO	International Organization for Standardization
ITRF	International Terrestrial Reference Frame
MCV	Miroir de Changement de Visée (Viewing Change Mirror)
NIMA	National Imagery and Mapping Agency
ROLT	Repère Orbital Local Tourné (*Orbital Coordinate System)*
Rp	Repère à piloter (*Navigation Reference Coordinate System)*
SLR	Satellite Laser Ranging
SPOT	Satellite Pour l'Observation de la Terre
SPOT123	Satellite SPOT1 or SPOT2 or SPOT3
SPOT4	Satellite SPOT4 only
SPOT5	Satellite SPOT5 only
SSM	Strip Selection Mirror
SWIR	Short wave InfraRed (MIR: Moyen InfraRouge)
ULS	Unité de Localisation Stellaire (*Star tracking Unit*)
UTC	Universal Time Coordinated
WGS84	World Geodetic System 1984
W3C	World Wide Web Consortium
XML	eXtended Markup Language

1.5 Definitions

This section controls the definition of all common terms used within this document. A special attention has been paid to inherit definitions from international standards as ISO, ANSI or ECSS.

GCP	Ground Control Points used to register the processed scene on a geodetic system. For SPOT scenes, GCPs may be used to refine values of the location model, such as the initial attitude values. These points are taken from external cartographic data such as digital maps, GPS measurements, geodesic points, other geocoded scenes…
WGS84	World Geodetic System established in 1984 and refined during the successive years has set up ellipsoid, datum and geoid to serve as reference. This model realizes the best compromise from accurate global positioning data (GPS) and altimetry data measured by satellites (GEOSAT, ERS-1 and TOPEX/POSEIDON).

1.6 table of symbols

This section defines all the symbols (mathematical or specific to SPOT) involved within at least one of the equations of this document. Location within the SPOT product of value(s) matching each one of these symbols is to be found in Appendix B.

aec	Average encoder count
(a_i)	Five (5) coefficients of polynomial A
$a_{oor}(t_i)$	Attitude value is "OUT_OF-RANGE" indicator
$(a_p)_1$	Rotation angle around the pitch axis at the beginning of the scene
$a_p(t)$	Interpolated rotation angle around the pitch axis at time t
$a_p(t_i)$	Rotation angle around the pitch axis (attitude sample i)
$a_p'(t_i)$	Average rotation speed around the pitch axis (attitude sample i)
$(a_r)_1$	Rotation angle around the roll axis at the beginning of the scene
$a_r(t)$	Interpolated rotation angle around the roll axis at time t
$a_r(t_i)$	Rotation angle around the roll axis (attitude sample i)
$a_r'(t_i)$	Average rotation speed around the roll axis (attitude sample i)
$(a_y)_1$	Rotation angle around the yaw axis at the beginning of the scene
$a_y(t)$	Interpolated rotation angle around the yaw axis at time t
$a_y(t_i)$	Rotation angle around the yaw axis (attitude sample i)
$a_y'(t_i)$	Average rotation speed around the yaw axis (attitude sample i)
(b_i)	Six (6) coefficients of polynomial B
c_0	On-board clock matching the date t_0
$c(l)$	On- board clock value of the image line l
(i_i)	Six (6) coefficients of polynomial I
(j_i)	Four (4) coefficients of polynomial J
l	Image line number (1 being the first line of the scene)
l_0	First line of the raw scene in the raw segment

$\mathbf{L_0}$	First line of the level 1B scene in the level 1B segment
$\mathbf{l_1}$	Line of absolute attitude $[(a_p)_1, (a_r)_1, (a_y)_1]$ at the beginning of the scene
$\mathbf{l_c}$	Line containing the scene center
$\mathbf{l_i}$	Line of average rotation speed $[a_p'(t_i), a_r'(t_i), a_y'(t_i)]$
$\mathbf{(l_i)}$	Six (6) coefficients of polynomial L
$\mathbf{l_m}$	Mean value of lines l in 1A segment
$\mathbf{L_m}$	Mean value of lines L in 1B segment
(l,p)	Coordinates of the pixel in 1A image
(L,P)	Coordinates of the pixel in 1B image
lsp	Line sampling period
$\mathbf{M_{MCV}}$	Look direction change matrix (*Matrice de changement de visée*)
$\mathbf{(O_1,X_1,Y_1,Z_1)}$	Navigation Reference Coordinate System (*Repère à piloter*)
$\mathbf{(O_1,X_1',Y_1',Z_1')}$	Inverted Navigation Reference Coordinate System (*Repère à piloter inverse*)
$\mathbf{(O_1,X_2,Y_2,Z_2)}$	Orbital Coordinate System (*Repère orbital local*)
$\mathbf{(O_T,X_T,Y_T,Z_T)}$	Terrestrial Coordinate System (*Repère terrestre*)
$\mathbf{(p_i)}$	Four (4) coefficients of polynomial P
$\mathbf{P(t_i)}$	Satellite position coordinates (ephemeris sample i)
p	Number of the pixel along the line (1 being the leftmost first pixel)
$\mathbf{p_0}$	First pixel of the raw scene in the raw segment (p_0=1)
$\mathbf{P_0}$	First pixel of the last line of the level 1B scene in the level 1B segment
$\mathbf{q_m}$	Mean value of columns p in 1A segment
$\mathbf{Q_m}$	Mean value of columns P in 1B segment
$\mathbf{t_0}$	UTC reference date
$\mathbf{t_c}$	Scene center date
$\mathbf{t_i}$	Universal times corresponding to the positions P(ti) and velocities $V(t_i)$
$\mathbf{u_1}$	Look direction in Navigation Reference Coordinate System
$\mathbf{u_2}$	Look direction in Orbital Coordinate System
$\mathbf{u_3}$	Look direction in Terrestrial Coordinate System
$\mathbf{V(t_i)}$	Satellite velocity coordinates
$\mathbf{\Delta l}$	Interval half-width for lines l in 1A segment
$\mathbf{\Delta L}$	Interval half-width for lines L in 1B segment
$\mathbf{\Delta p}$	Interval half-width for columns p in 1A segment
$\mathbf{\Delta P}$	Interval half-width for columns P in 1B segment
φ	Latitude in geodetic reference system
λ	Longitude in geodetic reference system
ω	On-board clock period
ψ_X	Along-track angle of the look direction in Navigation Reference Coord. System

ψ_Y	Across- track angle of the look direction in Navigation Reference Coord. System
$(\psi_X)_1$	Along-track look angle for the first pixel
$(\psi_X)_N$	Along-track look angle for the last pixel (N=3000 or N=6000)
$(\psi_X)_p$	Along-track look angle for the pixel number p (p=1..N)
$(\psi_Y)_1$	Across-track look angle for the first pixel
$(\psi_Y)_N$	Across-track look angle for the last pixel (N=3000 or N=6000)
$(\psi_Y)_p$	Across-track look angle for the pixel number p (p=1..N)

2 SPOT MISSIONS

2.1 Historical background

Since 1986, SPOT satellites have been acquiring images of the Earth. Except SPOT3 that stopped acquisition on November 1996, SPOT1, 2 and 4 form a constellation which will be reinforced with the launching of SPOT5 foreseen in the early 2002.

As illustrated in fig. 3, the first three SPOT missions belong to the first generation satellite and will be called SPOT123 in this handbook. The fourth satellite (SPOT4) represents the second generation including improvements in term of payload and positioning capabilities.

The next mission (SPOT5) belongs to the latest generation of SPOT missions with significant improvements in terms of on-board instruments and autonomous system of positioning and attitude control that will enable a high absolute location accuracy.

fig. 1 Artist view of SPOT4 - © 1998 - CNES

fig. 2 Artist view of SPOT5 - © 2002 - CNES

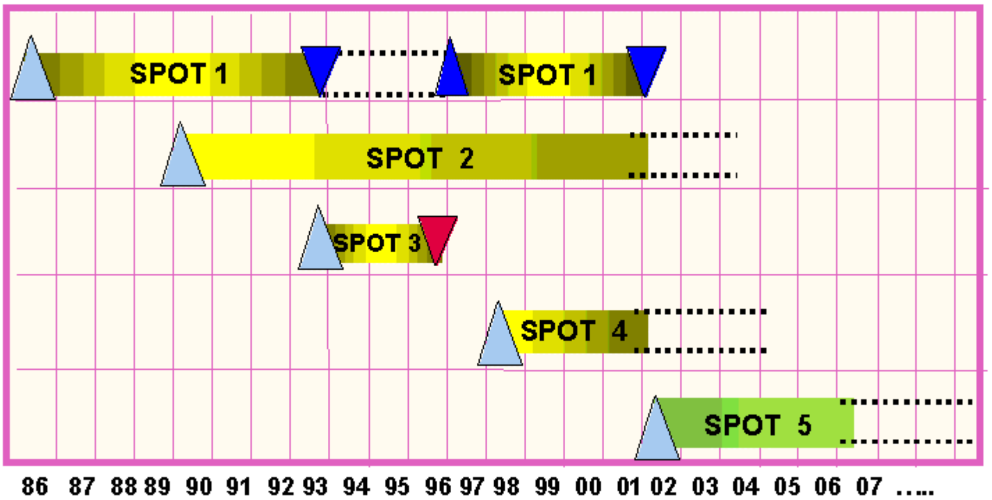


fig. 3 Chronology of SPOT missions

2.2 Orbit

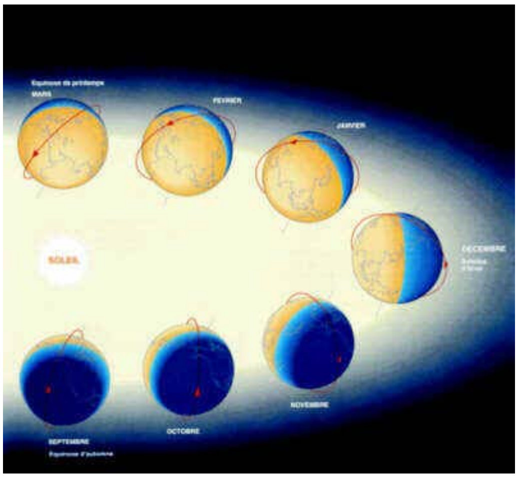
This section defines the main characteristics of the SPOT orbit, the way SPOT satellite constellation has been arranged to ensure the best Earth coverage, the instruments used to get satellite position in space (DORIS) and the instruments controlling the attitude.

For additional information, visit the CNES site http://spot4.cnes.fr

SPOT orbit characteristics

SPOT satellites are designed to acquire images of Earth in such a way as the images taken on different dates can be compared between each other. This can only be achieved if each SPOT satellite is exactly on the same orbit:

- The orbit is **phased**, which means that the satellite passes repeatedly over a ground point after a whole number of days. The SPOT satellite cycle ends 26 days to complete 369 orbital revolutions. The orbital period is 101.5 minutes. The ground track is accurately repeated every 26 days (cycle) and the satellite follows adjacent ground tracks every five-day sub-cycle.
- The orbit is **sun-synchronous**, i.e. the angle between the orbital plane and the Earth-Sun direction is constant. For SPOT satellites, angle is 22.5°, which means that the local time of the descending node is 10:30 (nominally between 10:15 and 10:30, see **Error! Reference source not found.**). Therefore, at a given latitude, the imagery is acquired with constant illumination (see fig. 4).
- The orbit is near-**polar**. This characteristic is a consequence of the previous two properties. The orbit inclination (tilt angle) with respect to the equatorial plane is about 98.8°. This characteristic enables a full coverage of the Earth (given the imaging instrument's oblique-viewing capability).
- The orbit is near-**circular**, with a perigee close to the Earth North Pole. This means that a constant altitude may be maintained over a given point on the ground. SPOT's altitude over a point located at 45° North is about 830 km.



The angle between the orbital plane and the Sun - Earth direction remains constant: this is why the orbit is known as sun-synchronous.

The orbital plane intersects the equatorial plane at two points along a straight line known as the line of nodes. A node is the point at which the satellite crosses the equatorial plane: on its journey from north to south it is the descending node; for SPOT's orbit, the descending node takes place during the "day" (in the sunlit part of the orbit) on its journey from south to north it is the ascending node; for SPOT's orbit, the ascending node takes place during the "night".

fig. 4 Sun-synchronous orbit - © 2000 - CNES

SPOT constellation

SPOT5 is part of a system currently operating 3 satellites in the SPOT family: SPOT1, SPOT2 and SPOT4. The first two satellites SPOT1 and SPOT2 no longer have storage capacity. With SPOT4, these satellites are currently in orbits that have the same characteristics as those described in table 1 here below. However, they are in different orbital phases with respect to each other along the orbit.

Type	Sun-synchronous
Altitude	832 km
Inclination	98.7 °

Period	101.4 minutes
Cycle	26 days
Local time	10 : 30

Table 1 SPOT orbit nominal characteristics

The SPOT5 orbital position phase has been analysed in terms of the missions to be carried out by the SPOT constellation and the upgrading required for the current infrastructure. Once the SPOT5 is put into orbit, there is a total of three operational satellites: SPOT2, SPOT4 and SPOT5. SPOT1 will be decommissioned and set in storage-only mode right before launching SPOT5. The latter will be put into orbit with a precise phasing in relation to SPOT2 and SPOT4. SPOT2 and SPOT4 positions will not undergo any modifications: in terms of phasing, SPOT4 is ahead by about a quarter of an orbit in relation to SPOT2 (97° orbital dephasing). SPOT5 will be placed at 97° ahead of SPOT4 (cf. figure 3).

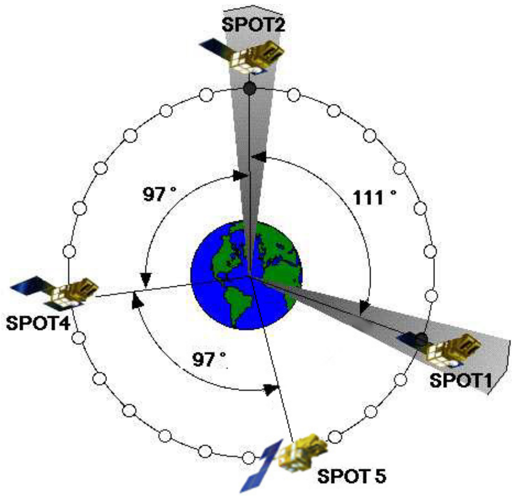


fig. 5 Orbital positions of the SPOT satellites

Orbit control - DORIS

DORIS (Doppler Orbitography and Radiopositioning Integrated by Satellite) is a radio-electrical system for high accuracy orbit determination and station positioning. It has been designed and developed by the "Centre National d'Études Spatiales" (CNES, the French space agency), the "Groupe de Recherches de Géodésie Spatiale" (space geodesy research group) and the "Institut Géographique National" (IGN, the French national survey agency).

The DORIS system is composed of the following elements:

- A world-wide permanent network of transmitting stations, called "orbitography network",
- Receivers on-board several satellites (currently SPOT2, SPOT4 and Topex-Poseidon, in the future Jason, Envisat and SPOT5),
- So-called "ground location stations", whose position is unknown a priori,
- A control center that performs system monitoring, instrument programming, and data processing / archiving.

Operation

The on-board receiver measures the Doppler shift on the signal emitted by the ground stations, at both frequencies (400 MHz and 2 GHz). The double frequency measurement is necessary to reduce errors due to ionospheric propagation delays. The collected data are stored in the instrument's memory and downloaded to the ground each time the satellite can see a receiving station: SPOT image facilities (Toulouse, France) and Kiruna (Sweden) for SPOT2 and 4. Once the forces acting on the satellite have been modeled, these measurements are processed to determine its precise trajectory. These orbit computation results can then be used to determine the exact position of stations to be located.

Two stations of the permanent network, called "master stations", have a specific function: they are used to synchronize the system with the International Atomic Time, and to upload programming data to the on-board receivers.

This is called an "uplink" system, since the signal is emitted by the ground beacons and received on-board the satellites, as opposed to the GPS for which the emitters are on the satellites.

fig. 6 DORIS antenna - © 1998 - CNES

Orbit determination

DORIS can determine very accurately the orbit of the satellites. The DORIS instrument on-board SPOT4 has an added function for the real-time determination of the host satellite's orbit, called DIODE (Détermination Immédiate d'Orbite par DORIS Embarqué). This function allows in particular to locate precisely the SPOT4 satellite.

For SPOT5, DORIS-DIODE system has been coupled with an orbit propagator software called TRIODE able to compute position and velocity. These ephemeris are dated every 30 seconds in UTC referential and ITRF terrestrial referential and transmitted to the ground along with the image telemetry.

In degraded case (failure of DORIS TRIODE systems), ephemeris are computed using the MADRAS orbit propagator that would take in input data uploaded from the TC&C command stations. MADRAS generates ephemeris data every 30 seconds.

Precision of the restituted orbits using DORIS or DIODE are indicated in the table here below giving the location errors along the three axis of the Orbital Coordinate System. These values match the RMS of samples acquired during validation campaigns performed on SPOT 3 and SPOT 4 satellites.

	ΔX perpendicular to the orbital plane	ΔY tangential to the orbit	ΔX radial location
MADRAS	31 m	209 m	48 m
DORIS	0.71 m	0.67 m	0.36 m

Attitude control

Attitude control (angular orientation) is needed so that the optical system covers the programmed ground area at all times. However, the satellite tends to change its orientation due to torque produced by the environment (drag of the residual atmosphere on the solar array, solar radiation pressure, etc.) or by itself (due to movement of mechanical parts such as on-board recorders or solar panel rotation). Therefore, the angular orientation has to be actively controlled. Another reason is the need to prevent "blurring" of the scenes acquired.

Local Orbital Reference System

The attitude is continuously controlled by a programmed control loop (AOCS): sensors measure the satellite's attitude, the onboard computer then processes these measurements and generates commands which are carried out by the actuator, to ensure correct pointing.

SPOT satellites are "three-axis stabilized", which means that its orientation is fully controlled relative to three axes. One of these directions corresponds to the line between the satellite and the center of the Earth, also called the "geocentric direction" (or "position vector") another is perpendicular to this geocentric axis, in the plane formed by the geocentric direction and the satellite velocity vector; the third is perpendicular to the first two.

These three axes form the local orbital reference system (ROLT).

The local orbital reference system (see fig. 7) is defined at each point of the orbit by three unitary vectors. These vectors are derived from the satellite position and velocity vectors:

- Vector $\vec{L}$ (from *Lacet* in French) is collinear with position vector $\vec{P}$ (on the axis between the Earth's center and the satellite). It defines the yaw axis.
- Vector $\vec{T}$ (from *Tangage* in French) is perpendicular to the orbital plane (vector $\vec{L}$, velocity vector $\vec{V}$). It defines the pitch axis.
- Vector $\vec{R}$ (from *Roulis* in French) completes the set of orthogonal axes. It lies in the plane defined by vectors $\vec{L}$ and $\vec{V}$ and defines the roll axis. It does not coincide exactly with the velocity vector due to the eccentricity of the orbit.

This (P,T,R,L) reference system is called (O,X_2,Y_2,Z_2) in the rest of the document.

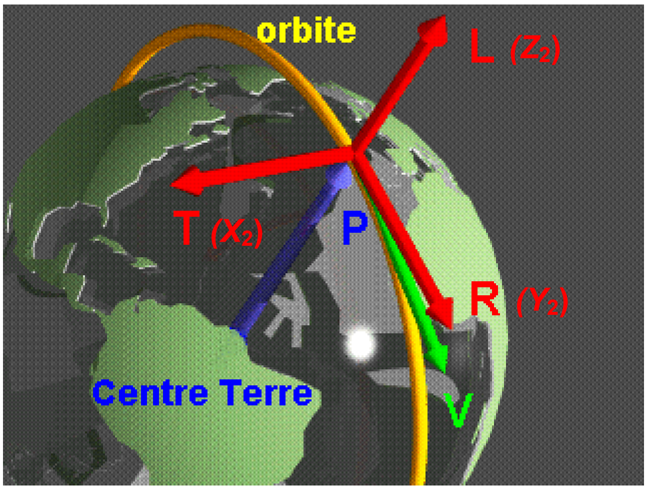


fig. 7 Local orbital reference system (ROLT)

Navigation Reference Coordinate System

Axes X_1, Y_1, Z_1 represent an orthogonal reference frame related to the satellite (satellite axes). Nominal attitude pointing consists of the best possible alignment of this set of axes with the local orbital reference system (while ensuring stability and limiting angular rates about this position).

With perfect geocentric pointing, this gives:

$\overrightarrow{X_1} = \vec{T}$
$\overrightarrow{Y_1} = \vec{R}$
$\overrightarrow{Z_1} = \vec{L}$

To maintain such equalities, the following equipment is used for attitude control:

sensors

- an inertial platform consisting of four rate gyros ; these are two-axis rate gyros and two of them are enough to provide angular rate measurements along the three axes of the satellite. The two others are thus used as backups. Gyros measure angular velocities along each one of the three axis,

On-board SPOT4:

- two digital Earth sensors (STD), one nominal and one redundant, are used to measure angular displacement about the pitch and roll axes,
- two digital Sun sensors (SSD), one nominal and one redundant, are used to measure angular displacement around the yaw axis (once per orbit).

On-board SPOT5:

- a star tracking unit computes absolute angles along the three attitude axis identifying constellations on the celestial vault. These measurements combined with the AOCS values provide high accuracy attitude measures to the ground (ULS).

Actuators

- Three magnetic-bearing reaction wheels (RRPM) are used to apply torque to the satellite and thus to rotate it about one of the X, Y or Z axes.
- Two magnetic torquers (MAC), which, through interaction with the Earth's magnetic field, create torques which are used to control the speed of the reaction wheels.

- Two types of thrusters (burning hydrazine) each producing a force of 3.5 or 15 Newtons; their orientation relative to the satellite's center of gravity inducing rotation about one of the axes, X, Y or Z.

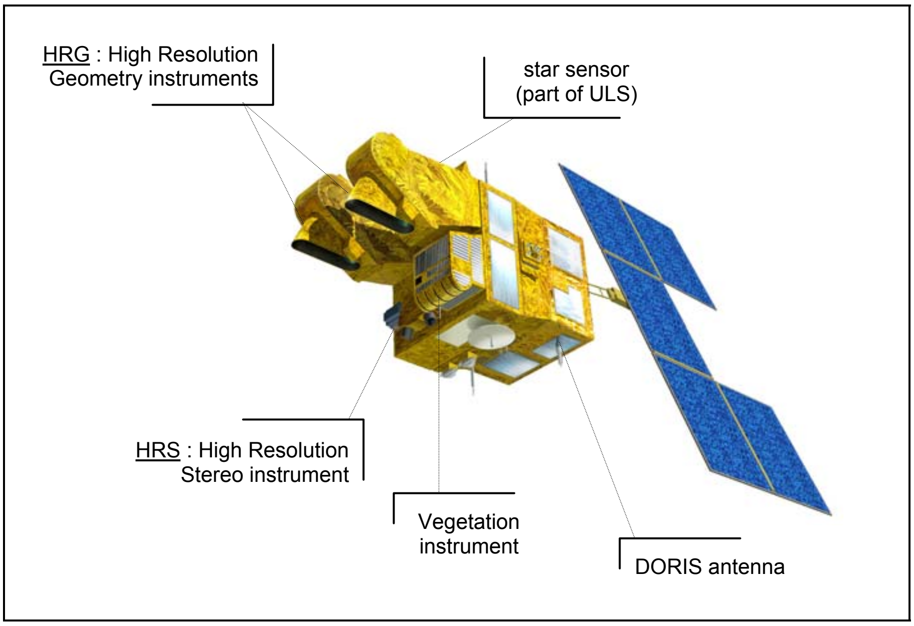


fig. 8 SPOT5 satellite

Yaw steering mode

For SPOT5 only, the platform is controlled in yaw steering mode in order to inbalance the line drift effect due to the Earth rotation (see fig. 9). Yaw angle is continuously changed depending on the latitude of the satellite. This operation allows to suppress the "black pixels" introduced when processing level 1B scenes (see fig. 10).

Yaw angle is given by the following formulae:

$$\beta = arctg\left[\frac{V_e \times \sin(i) \times \cos(\alpha_v)}{V_{sat} - V_e \times \cos(i)}\right]$$

Where

β	is the yaw angle to be maintained,
V_e	is the terrestrial velocity on equator,
V_{sat}	is the mean velocity of the satellite projected on ground,
i	is the orbital inclination ($i = 98.72°$),
α_v	$\alpha_v = \omega + v$, is the geocentric angle from ascending node (considering a spherical orbit, this angle may be condidered as a geocentric "latitude" of the satellite),
ω	is the argument of perigee, and
v	is the true anomaly.

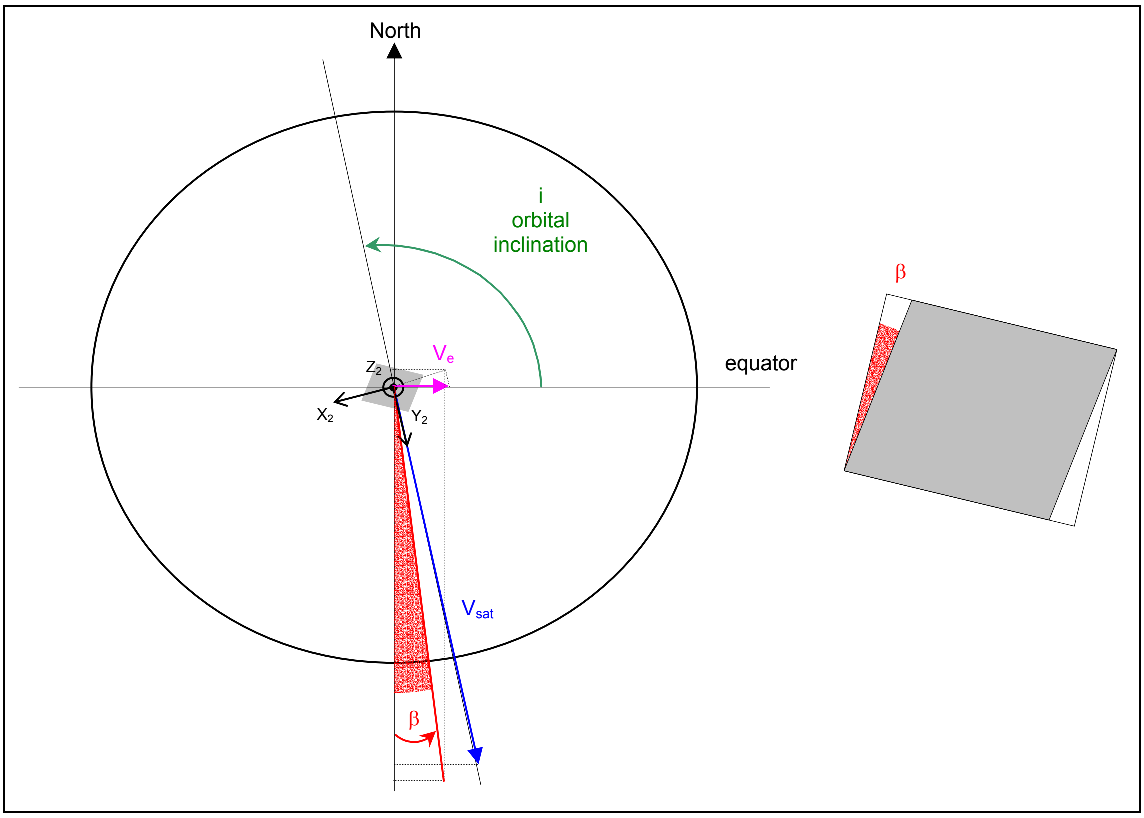


fig. 9 On-ground SPOT scene pattern without yaw steering

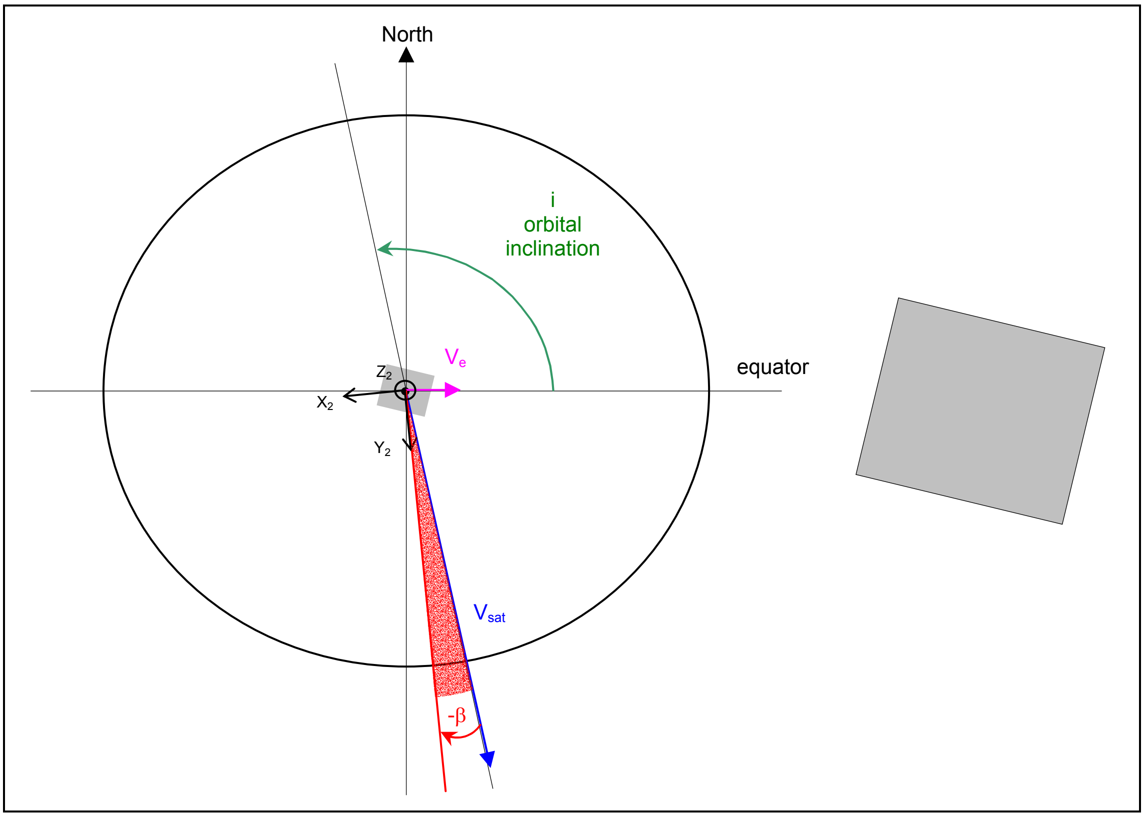


fig. 10   On-ground SPOT scene pattern with yaw steering

2.3 Payload

Instruments

The following table of this section records the main characteristics of the instruments on-board the SPOT missions and which are covered by this handbook.

satellite	*instrument*	*band name*	*wavelength range*	*sampling distance(1)*	*CCD per line*
SPOT123	HRV1 or HRV2	XS1	0.50-0.59 µm	20 m	3000
		XS2	0.61-0.68 µm	20 m	3000
		XS3	0.78-0.89 µm	20 m	3000
		PAN	0.50-0.73 µm	10 m	6000
SPOT4	HRVIR1 or HRVIR2	XS1	0.50-0.59 µm	20 m	3000
		XS2	0.61-0.68 µm	20 m	3000
		XS3	0.78-0.89 µm	20 m	3000
		SWIR	1.58-1.75 µm	20 m	3000
		M	0.61-0.68 µm	10 m	6000
SPOT5	HRG1 or HRG2	XS1	0.495-0.605 µm	10 m	6000
		XS2	0.617-0.687 µm	10 m	6000
		XS3	0.780-0.893 µm	10 m	6000
		SWIR	1.545-1.750 µm	20 m	3000
		HMA	0.475-0.710 µm	5 m	12000
		HMB	0.475-0.710 µm	5 m	12000
	HRS	HRS1 (*fore view*)	0.49-0.69 µm	10 m x 5 m	12000
		HRS2 (*aft view*)	0.49-0.69 µm	10 m x 5 m	12000

(1) Ground sampling distance at vertical viewing.

Payload modes of operations

Data acquired by the instruments are multiplexed to be downloaded to the receiving stations. Because of limitations in the downlink rate, all the data acquired on-board cannot be sent to the receiving stations. Table here below shows the data segments in output of each instrument and the possible combinations.

<table>
<tr><td>satellite
download rate
On-board memory</td><td>data segments</td><td colspan="13">combinations of data segments</td></tr>
<tr><td rowspan="3">SPOT123
2 x 25 Mbits/s
2 tape recorder</td><td rowspan="3">HRV1-XS
HRV1-PAN
HRV2-XS
HRV2-PAN</td><td colspan="13">Two (2) among the 4 data segments</td></tr>
<tr><td>HRV1</td><td>XS</td><td>P</td><td>XS</td><td>P</td><td>P+ XS</td><td></td><td colspan="6" rowspan="2"></td></tr>
<tr><td>HRV2</td><td>XS</td><td>P</td><td>P</td><td>XS</td><td></td><td>P+ XS</td></tr>
<tr><td rowspan="3">SPOT4
2 x 25 Mbits/s
2 tape recorder +
9 GB solid state memory</td><td rowspan="3">HRVIR1-XS(1)
HRVIR1-XS
HRVIR1-M
HRVIR2-XS(1)
HRVIR2-XS
HRVIR2-M</td><td colspan="13">Two (2) among the 6 data segments</td></tr>
<tr><td>HRVIR1</td><td>XS</td><td>P</td><td>XS</td><td>P</td><td>P+ XS</td><td></td><td>XS</td><td>M</td><td>XS</td><td>M</td><td>M+ XS</td><td></td></tr>
<tr><td>HRVIR2</td><td>XS</td><td>P</td><td>P</td><td>XS</td><td></td><td>P+ XS</td><td>XS</td><td>M</td><td>M</td><td>XS</td><td></td><td>M+ XS</td></tr>
</table>

SPOT5 2 x 50 Mbits/s 110 GB solid state memory	HRG1 – XS HRG1 – HMA HRG1 – HMB HRG2 – XS HRG2 – HMA HRG2 – HMB HRS - ST	Five (5) among the 7 data segments, 2 being directly transmitted and 3 having been recorded.

(1) Because of the limitations for some of the ground stations to receive the whole SPOT4 data, it has been introduced a degraded mode not downloading the SWIR channel, and thus providing full compatibility with SPOT123 modes of operation.

Channel superimposition – SPOT123

For SPOT123 satellites, the panchromatic scene and the multispectral scene acquired at the same time and by the same instrument are not strictly superimposable due to the different locations of the CCDs in the focal plane of the instrument(R-2). In particular the different look directions (ψ_X,ψ_Y) may lead to parallax effect in regions with much changing elevations.

At the opposite, all the bands (XS1, XS2 and XS3) of the multispectral scene are perfectly superimposable.

The image line acquired at a given instant by any one of the HRV instrument is approximately (see fig. 11):
- +7.5 km in front of the subsatellite point in panchromatic mode, and
- -7.5 km behind the subsatellite point in multispectral mode.

Consequently, with the SSM in vertical viewing (center position 48),

- the look directions passing through the center of each detector in bands XS1, XS2, XS3 intersect a series of equally spaced points lying on a plane perpendicular to Z_1. The look direction matching the central detector has an angle specified to be –0.529° with Z_1. The detector look directions for bands XS1, XS2 and XS3 are coincident;
- the look directions passing through the center of PAN detector intersects a series of equally spaced points lying on a plane perpendicular to Z_1. The look direction matching the middle of the two central detectors has an angle specified to be +0.529° with Z_1.

In the next figures of this section, viewing planes include the look directions of all the detectors of a particular band and are shown in the Navigation Reference Coordinate System defined in section 4.1. Inclination of the viewing plane with regard to the Z axis matches the look direction of the median detector (ψ_X).

Therefore, due to differences of instrument making between SPOT satellites, one must read the (ψ_X,ψ_Y) viewing angles in order to perform an accurate physical model. Precise values shall be read inside the products.

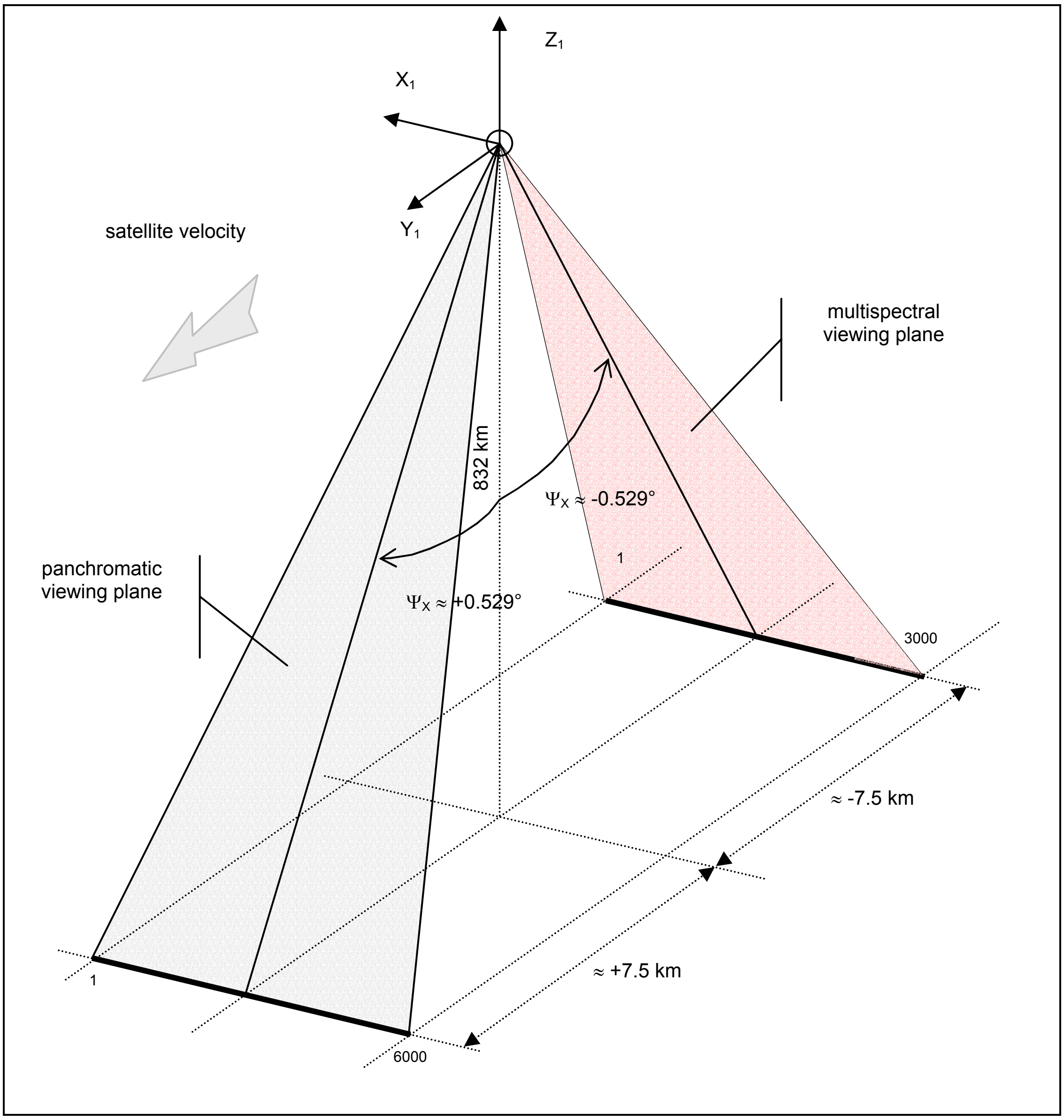


fig. 11 SPOT123 – PAN and XS look directions

Channel superimposition – SPOT4

For SPOT4, the monospectral (M) / multispectral (XS) superimposition defect observed for SPOT123 (see previous section) has been settled. Indeed, the monospectral image lines are in output of the B2 detector that is also used to acquire the XS2 image band. This feature allows producing “Merge products” including multispectral XS bands with a 10 metres resolution.

From a physical point of view, detectors arrays matching XS1, XS2 and XS3 bands respectively are assemblies of four 1500 CCD arrays each one. In multispectral mode, output of two adjacent CCDs are averaged to produce 3000 radiometric values per line.

Technology for the SWIR detector is more complex. The SWIR array is an assembly of 10 bricks, each one of 300 detectors. Within each brick, odd and even detectors are located along two parallel lines (see fig. 12). Look directions for odd SWIR detector are coincident with those for bands XS1, XS2 and XS3. While look directions for even SWIR detectors make an angle with Z_1 axis of typically $+4.8\ 10^{-5}$ radians.

This difference in look directions is corrected in products distributed to users, interpolating successive acquisition lines.

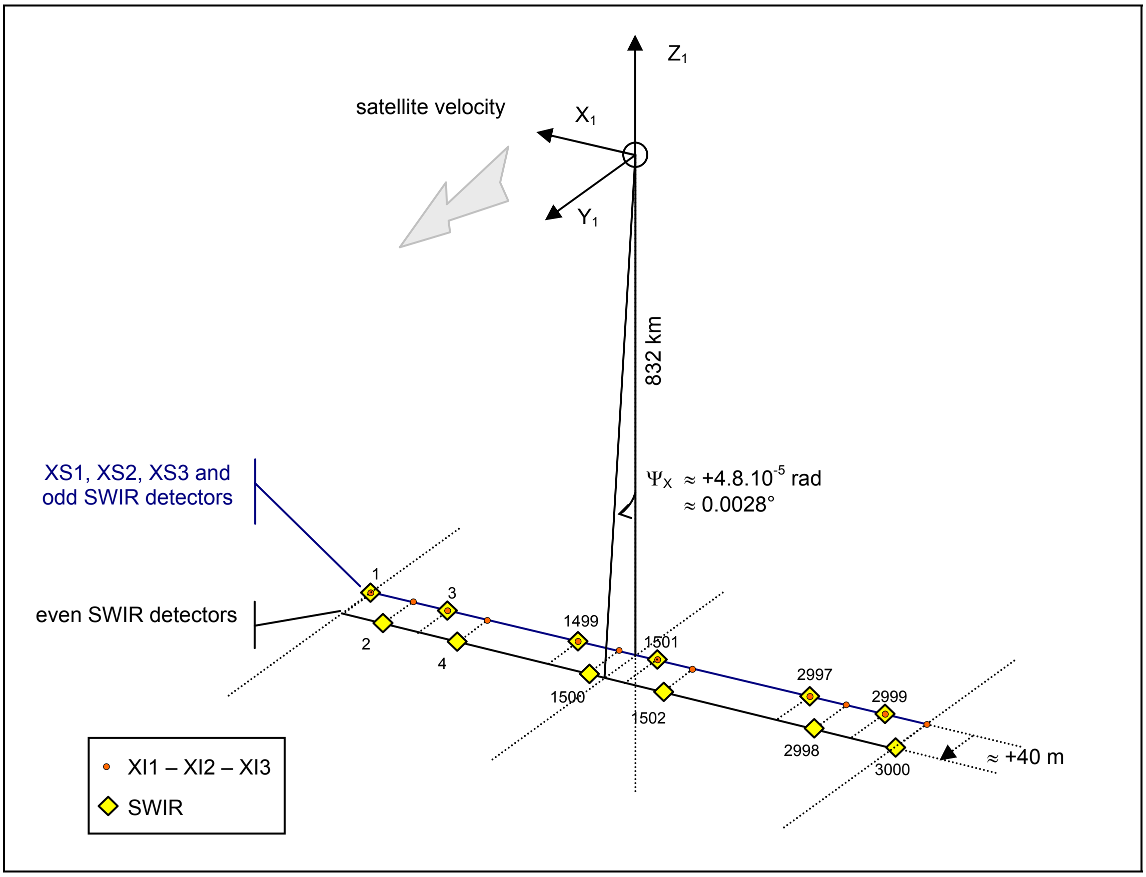


fig. 12 SPOT4 – SWIR and XS1, XS2, XS3 pixel look directions

SPOT5 –HRG instrument overview

On-board SPOT5 (see fig. 13), the two 5-metres resolution monospectral bands (HMA and HMB) acquire ground reflectance values located foreward, while the multispectral bands (XS1, XS2, XS3 and SWIR) match ground reflectance values located backward (rear view).

With the Strip Selection Mirror (SSM) in vertical viewing (center position 48),

- the look direction matching the central detector of bands XS1, XS2, XS3 and SWIR has an angle specified to be $\underline{-9.242.10^{-3}\ \text{rad}}$ (≈-0.529528°) with Z_1. The detector look directions for bands XS1, XS2, XS3 and SWIR are coincident(1);
- the look direction matching the central detector of HMA has an angle specified to be $\underline{+9.242.10^{-3}\ \text{rad}}$ (≈+0.529528°) with Z_1;
- the look direction matching the central detector of HMB has an angle specified to be $\underline{+9.221.10^{-3}\ \text{rad}}$ (≈+0.528324°) with Z_1.

(1) *<u>Informative note</u>: SWIR line projected on ground has one pixel less than the other XS1, XS2 and XS3 bands. Pre-processing algorithms from level 1A products overcome this defect.*

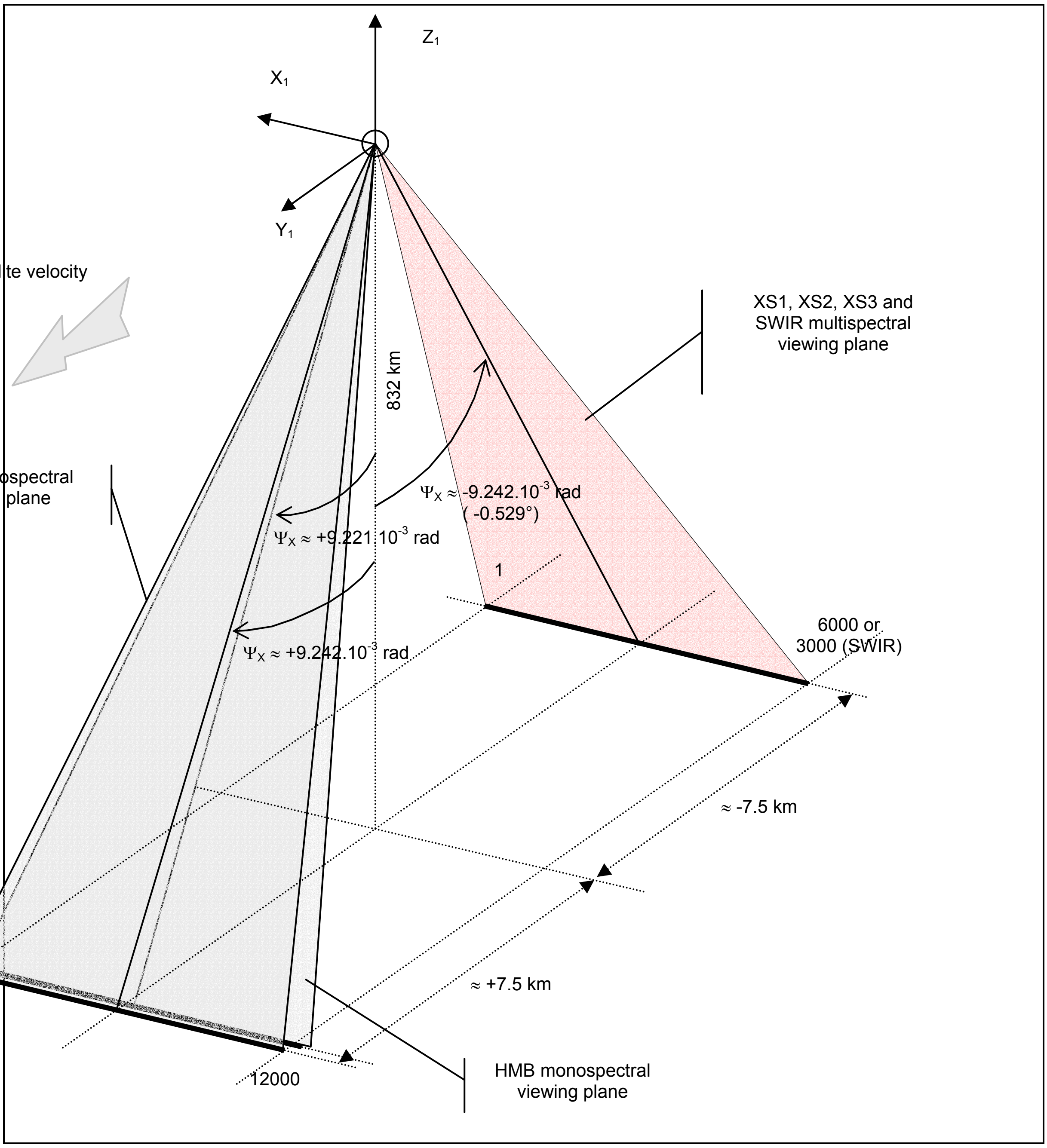


fig. 13 SPOT5 – HRG instrument

<u>SPOT5 – HRG instrument – HMA / HMB superimposition</u>

On-ground pixels of HMA and HMB scenes are interleaved to allow 2.5 metres interpolation of “SuperMode” image.

Along and across track shifts are consequence of the difference between the typical look direction angles of the two detector arrays.

Along track

$$\left.\begin{array}{l}(\Psi_X)_{HMA} = 9.242 \times 10^{-3}\, rad \\ (\Psi_X)_{HMB} = 9.221 \times 10^{-3}\, rad\end{array}\right\} \Rightarrow \Delta\Psi_X \approx 2.1 \times 10^{-5}\, rad$$

Assuming an altitude of 832 km, such angular difference leads to a spatial shift of approximately 17.5 metres, i.e. 3.5 pixels of 5-metres resolution.

Across track

$$(\Psi_Y)_{HMB} \approx (\Psi_Y)_{HMA} + 3 \times 10^{-6}\, rad$$

Assuming the same altitude, such angular difference leads to a spatial shift of approximately 2.5 metres, i.e. 0.5 pixels.

fig. 14 shows the relative location on Earth of HMA and HMB pixels. In nominal conditions, pixel (l,p) of HMB scene (where l and p are respectively the line and column numbers) matches its homologous pixel (l+3,p) in HMA scene. HMB pixel is shifted +0.5 pixel along the lines and columns.

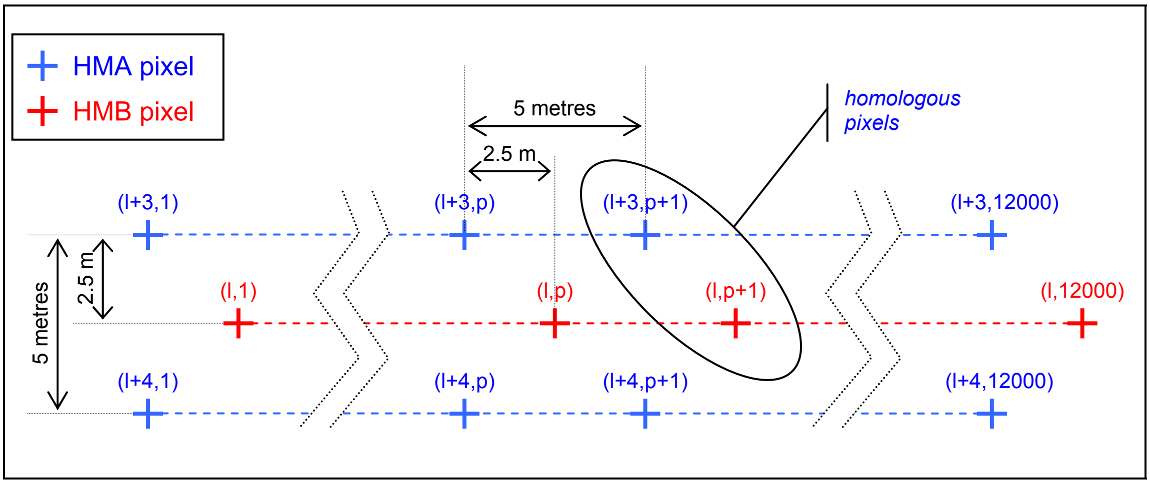


fig. 14 SPOT5 – Interleaving HMA and HMB pixels

SPOT5 – Supermode

HMA / HMB interleavings allows to compute a high resolution image called « Supermode ». Each line of this level 1A interpolated and restored image has 24000 pixels with a ground sampling distance of 2.5 metres.

As for the other SPOT5 products in DIMAP format, look directions (φ_X,φ_Y) are given for each one of the pixels which have been computed as they would have been acquired by a « pseudo CCD array » of 24000 detectors.

SPOT5 –HRS instrument

Unlike the HRG instrument, HRS telescopes do not include a mirror mechanism. HRS scenes (see fig. 15) are acquired under the satellite track with a swath 120 km width (12000 detectors x 10 metres resolution). Look directions of telescopes are +20° (*fore view*) and –20° (aft view).

Such look directions lead to incidence angles of 22.748° (see equation 32 in APPENDIX B -) and therefore to an efficient stereo B/H ratio of 0.84 (≈ 2 x tan(22.748°)).

Foreward and backward acquisitions cannot be performed at the same time. As a consequence, the maximum stereo segment that can be acquired is a little bit more than 600 km (≈ 832 km altitude x 2 x tan(20°)).

Foreward and backward images are obtained on the same panchromatic spectral band as for HRG. The size of the pixels on ground are 10m x 10m. However, HRS instrument has been designed for a ground sampling distance of 5 metres along the track. In a direction close to the epipolar planes, this along-track over-sampling allows higher altimetric accuracy of the DEM to be obtained (absolute planimetric resolution from 10 to 15 metres).

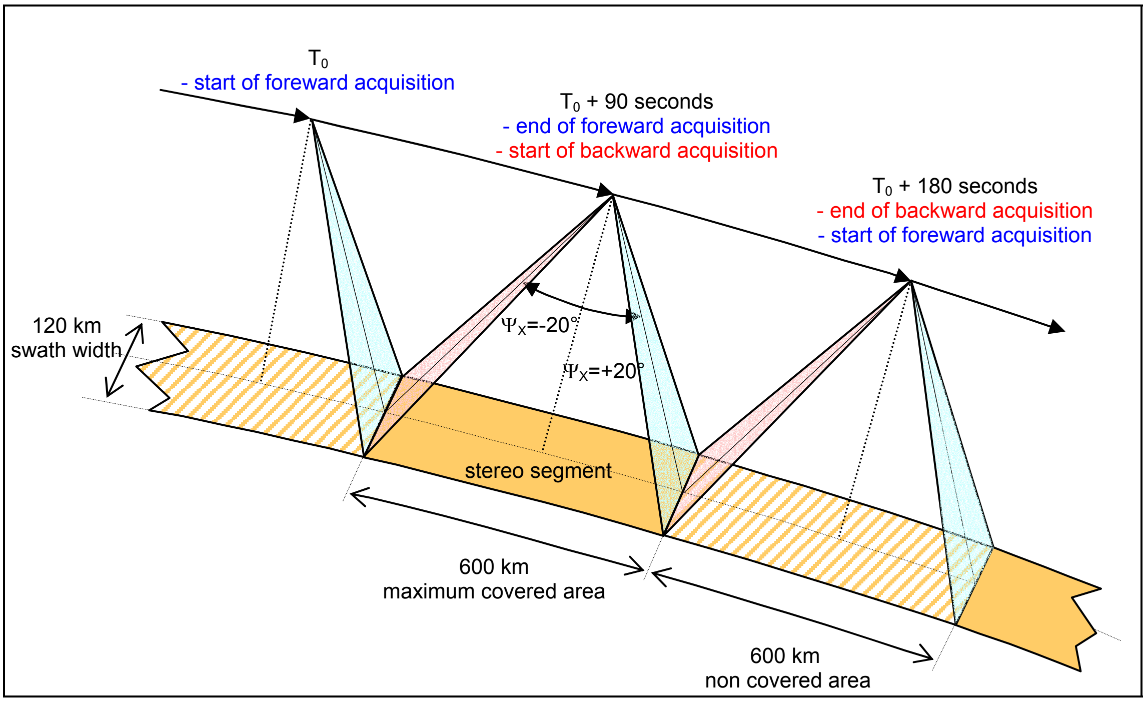


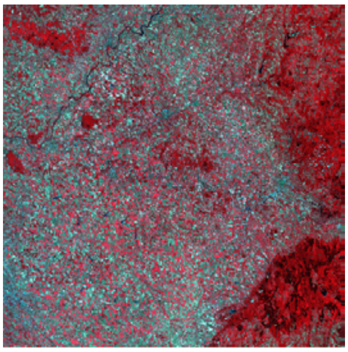
fig. 15 SPOT5 – HRS instrument

3 SPOT PRODUCTS

3.1 Processing levels

This section summarises the categories of processing levels applied when generating SPOT products. Some equivalencies are given for the other missions.

Level 1A – Raw product

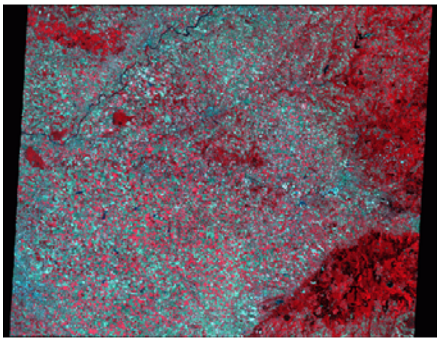


For SPOT 1A scenes, only a radiometric processing has been applied to compensate the differences the differences of sensitivities between the various elements of a CCD array.

Geometry of the product is kept unchanged leading to an image where width matches exactly the number of detectors (SPOT123-4: 3000 for XS, 6000 for PAN or M, SPOT5: 6000 for XS or 12000 for HM).

For SPOT4 and SPOT4 level 1A scenes, SWIR band is registered on the XS bands.

Level 1B – System corrected product



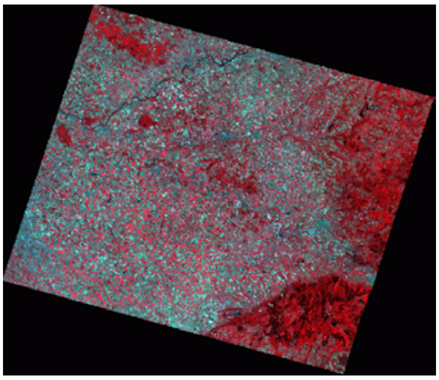
Scope of level 1B processing is to remove the internal geometric distortion of the image. Angles observed on Earth will have the same values within the image while the distortion of lengths measured in whatever direction will be minimised.

In particular, level 1B processing compensates the geometric distortions caused by:

1. Earth rotation,
2. Earth curvature, and
3. Panoramic effect.

Algorithms used to generate level 1B products are presented in section 6.
Earth rotation correction is visible on SPOT123 and SPOT4 images (black pixels at the start and at the end of image lines). SPOT5 images will not show such black pixels because of the yaw steering mode altitude control of the platform (see section 2.2).
Level 1B products are generated only using mono-dimensional resampling and are not geocoded.

Level 2A – Projected product without ground reference points



A level 2 product is a cartographic product, i.e. a product which geometry matches a standard projection and a standard geodetic system (ellipsoid / datum). Projected products are computed using a viewing geometry model presented in section 4.

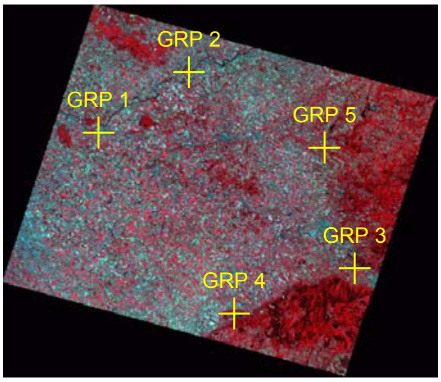
For SPOT123 and SPOT4, level 2A products are generated using only the ancillary data of the acquired image and do not use any ground control point.

Level 2A products are generated only using bidimensional resampling and are geocoded.

A coarse digital elevation model can be used to compensate for major variations of altitude.

For SPOT5, GLOBE DEM is used for orthorectification.

Level 2B – Projected product with ground reference points

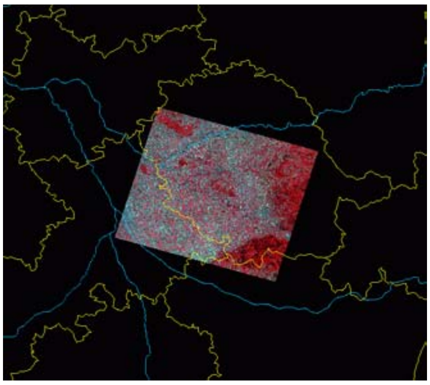


Level 2B products are generated using the ancillary data of the acquired image together with ground reference points taken from maps, GPS data, geodetic points, orthorectified images…

Ground reference points are used to estimate one or more unknown variables of the viewing geometry model.

For SPOT123 and SPOT4 the unknown variables are first the initial attitude values.

Level 3 – Orthorectified products



Orthorectified products (also called "level 3" products) are generated using the ancillary data of the acquired image, ground reference points and a Digital Elevation Model (DEM) to correct effects of the relief (parallax due to altitude variations for non vertical look direction).

Because of the ability of SPOT instruments to make vary the viewing angle, orthorectification processing is a major issue for SPOT products.

Orthorectified products are superimposable on cartographic layers and are used to produce "space maps".

3.2 Data format

This section summarizes the three formats that have been used to disseminate SPOT products.

SISA format

From 1986 to 1995, the first SPOT123 products were delivered using a CEOS format customized for SPOT products, also called “SISA format”. This format is today definitively superseded by the “CAP format” and is out of the scope of this handbook.

During the transition period, this format has also been called “Old format” to distinguish from the CAP format, called “New format”.

CAP format

“CAP format” has been released in September 1995 to accomodate with the SPOT4 satellite to be launched in 1998. The new instruments on-board this satellite and new processing techniques have obliged to rearrange the “SISA format” to take into accound the 4^{th} band (SWIR).

A second edition issued in November 1997 (see R-3) remains the reference format specification for SPOT123 and SPOT4 products.

DIMAP format

For SPOT5, SPOT IMAGE has decided to use a new emerging standard to encode auxiliary data: XML (R-4 and R-7). This flexible language allows to break down data in semantic blocks and can be accessed by many parsers or navigators off-the-shelf.

An other advantage of DIMAP is that images are delivered and directly accessible through standard formats (e.g. GeoTIFF).

Products made from SPOT123 or SPOT4 data are also proposed with DIMAP format.

4 VIEWING GEOMETRY MODEL – THE DIRECT MODEL

This section provides algorithms allowing to compute orthorectified products from level 1A products.

When only a level 1B product is available, it is recommended to transform back this product in level 1A (see section 5.4) before applying the algorithms.

4.1 Principle

A "*viewing geometry model*" consists in establishing a relation between any pixel (l,p) of the level 1A image and the relative point (λ,φ) on a terrestrial reference system. In this relation, the altitude h of the point on the ground is supposed to be known.

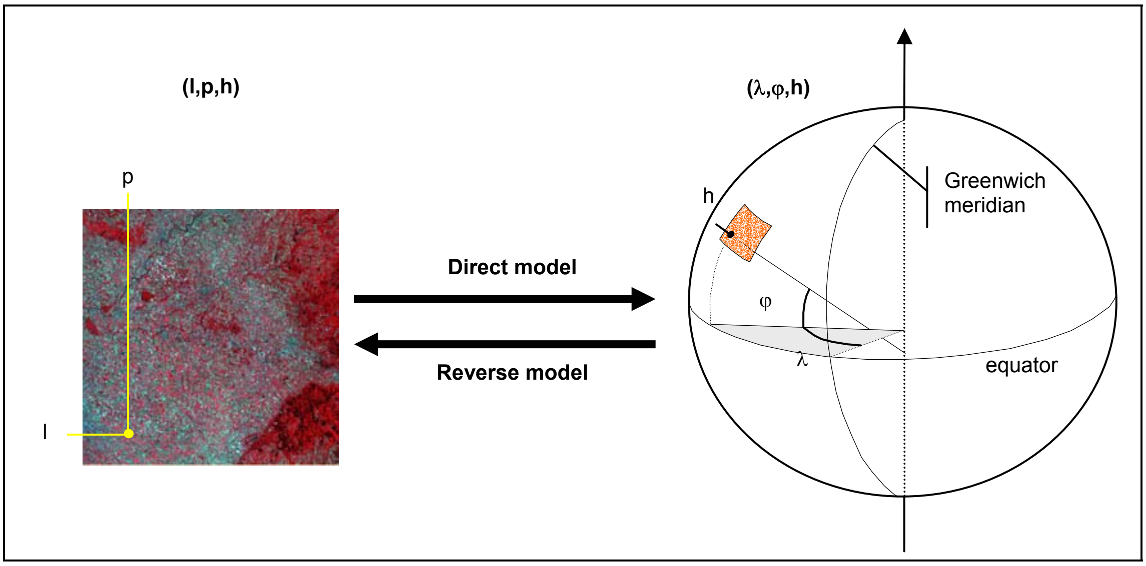


fig. 16 Viewing geometry model

The goal of the *direct model* is to compute the intersection between the look direction of any pixel (l,p) with an Earth model. This Earth model could be estimated by using a Digital Elevation Model (DEM) above an ellipsoid (see section "Using a DEM – Altitude pre-processing") or just considering a constant elevation (e.g. h=0) above an ellipsoid.

The direct model is computed performing a series of elementary transforms described within the next sections:

1. Line dating: set a relation between any pixel (l,p) of the image and the date t of its acquisition.
2. Ephemeris interpolation: determinates the position P(t) and velocity V(t) of the satellite at the date t.
3. Look direction in Navigation Reference Coordinate System: determinates the look direction within a reference system firmly attached to the satellite.
4. Look direction in Orbital Coordinate System: express the look direction with a reference system integrating the navigation constraints, and in particular the attitude variations.
5. Look direction in Terrestrial Coordinate System: determinates the look direction attached to the ITRF Earth reference system.
6. Location on Earth model: computes the intersection of the line of sight with the Earth model (ellipsoid + DEM).

4.2 Line dating

All the SPOT satellites have an on-board oscillator able to associate any acquired line with a sequential integer count. In particular, SPOT4 and SPOT5 use the ultra stable oscillator of DORIS able to provide an on-board time with datation accuracy better than 100 µs.

Date t of any line l is given with reference to the scene center date.

$$t = t_c + lsp \times (l - l_c) \qquad \text{(Eq. 1)}$$

Where

t_c is the scene center date,

l_c is the line containing the scene center,

lsp is the line sampling period.

4.3 Ephemeris interpolation

SPOT products include ephemeris auxiliary data giving the position and velocity of the satellite within an interval of time including the acquisition. Ephemeris are interpolated from this samples using the Lagrangian formula.

For SPOT123 and SPOT4, ephemeris data are given every minute. For a standard 60 km scene, eight (8) ephemeris values are provided. In order to compute position and velocity of any line within the scene, the Lagrangian interpolation requires 4 ephemeris points before this line and 4 points after this line. In the case where one ephemeris point fall into the scene, 9 ephemeris points are provided: 4 before the scene and 4 after the scene (see fig. 17).

For SPOT5, ephemeris data are given every 30 seconds for the whole data strip including magines at the start and at the end of the data strip.

Let t being the time of the line l for which position P(t) and velocity V(t) shall be computed, select the four ephemeris samples at time t_1, t_2, t_3 and t_4 before t and the four samples at time t_5, t_6, t_7, t_8 after t; all the time t_i being out of the acquisition range.

Position and velocity are given by the following formulae:

$$\vec{P}(t) = \sum_{j=1}^{8} \frac{\vec{P}(t_j) \times \prod_{\substack{i=1 \\ i \neq j}}^{8} (t - t_i)}{\prod_{\substack{i=1 \\ i \neq j}}^{8} (t_j - t_i)} \qquad \text{(Eq. 2a)}$$

$$\vec{V}(t) = \sum_{j=1}^{8} \frac{\vec{V}(t_j) \times \prod_{\substack{i=1 \\ i \neq j}}^{8} (t - t_i)}{\prod_{\substack{i=1 \\ i \neq j}}^{8} (t_j - t_i)} \qquad \text{(Eq. 2b)}$$

Where

$P(t_i)$ are the satellite position coordinates,

$V(t_i)$ are the satellite velocity coordinates,

t_i are the universal times corresponding to the positions and velocities.

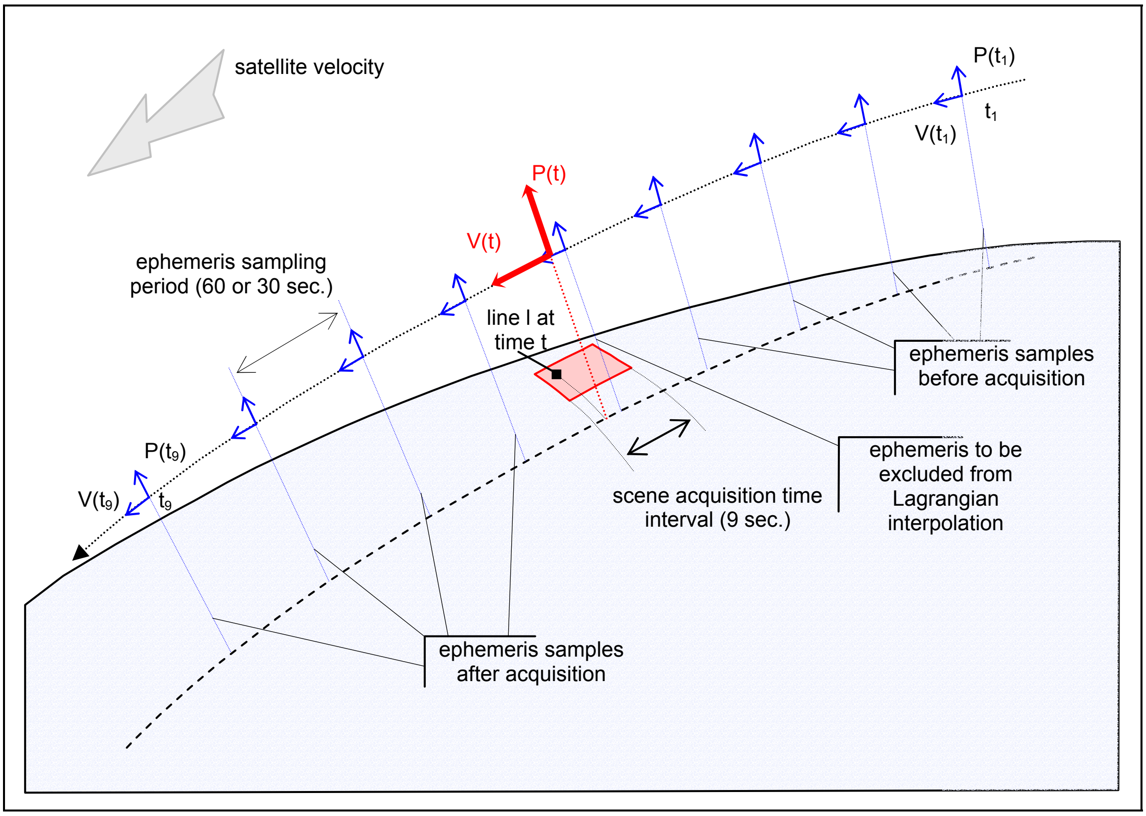


fig. 17 Interpolating position and velocity from ephemeris data – Example for one full scene

4.4 Look direction in Navigation Reference Coordinate System

The Navigation Reference Coordinate System

The (O_1,X_1,Y_1,Z_1) Navigation Reference Coordinate System, also called "*Repère à piloter*" in French, is the body-fixed system used for spacecraft attitude determination and control. The coordinates axes are defined by the spacecraft attitude control system (ACS) which attempts to keep the navigation reference frame aligned with the Orbital Coordinate System (see section 4.5) so that the optical axis of instrument without mirror deviation is always pointing towards the center of the Earth.

As illustrated in fig. 18, Y_1 axis is not necessarily strictly aligned with the satellite velocity vector. This misalignment may be due to small drift of yaw and pitch values in nominal case or and systematic yaw steering mode control for SPOT5 acquisition.

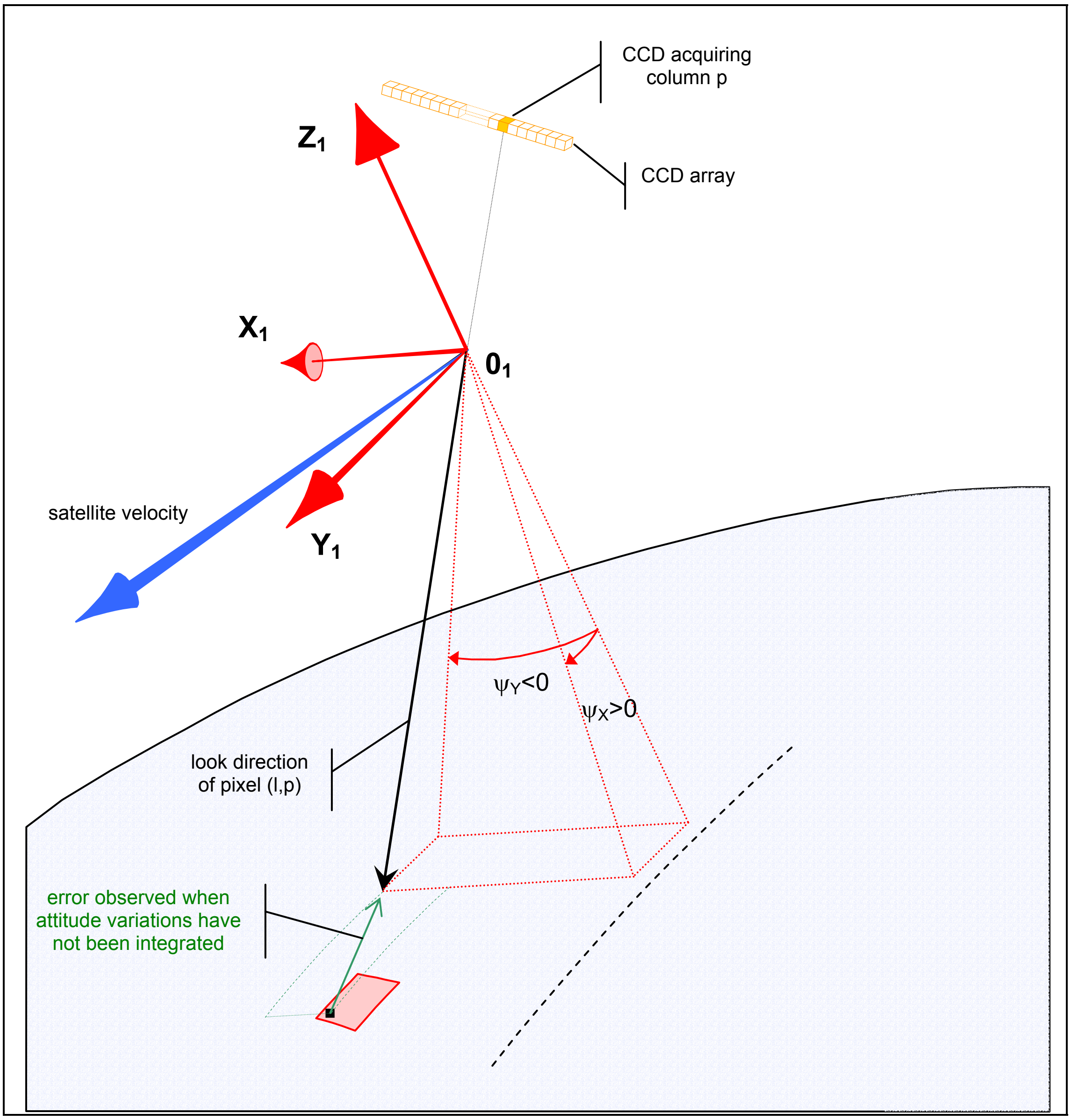


fig. 18 Navigation Reference Coordinate System

Look direction of pixel p

For any line l of a level 1A scene, pixel of a particular column q has been acquired by a unique CCD. The look direction matching this CCD is defined by the two angles ψ_X and ψ_Y expressed within the (O_1,X_1,Y_1,Z_1) Navigation Reference Coordinate System. These two angles are computed from look direction angles $(\psi_X)_i$ and $(\psi_Y)_i$ matching two or more CCDs and are provided in auxiliary data.

Case of SPOT123 and SPOT4

For SPOT123 and SPOT4, the CCD look angles $[(\psi_X)_i,(\psi_Y)_i]$ are only given for the first and last CCD of the detector arrays:

- $[(\psi_X)_1,(\psi_Y)_1]$ and $[(\psi_X)_{3000},(\psi_Y)_{3000}]$ for XS1, XS2, XS3 and SWIR bands, or
- $[(\psi_X)_1,(\psi_Y)_1]$ and $[(\psi_X)_{6000},(\psi_Y)_{6000}]$ for PAN or M bands.

The look direction for any pixel p is computed by linear interpolation from the look directions u_{p1} and u_{pN} of the first and last pixels respectively :

$$\vec{u}'_{p1} = \begin{pmatrix} -tg[(\psi_Y)_1] \\ +tg[(\psi_X)_1] \\ 1 \end{pmatrix} \qquad \text{(Eq. 3a)}$$

$$\vec{u}_{p1} = \frac{\vec{u}'_{p1}}{\left\| \vec{u}'_{p1} \right\|} \qquad \text{(Eq. 3b)}$$

$$\vec{u}'_{pN} = \begin{pmatrix} -tg[(\psi_Y)_N] \\ +tg[(\psi_X)_N] \\ 1 \end{pmatrix} \qquad \text{(Eq. 3c)}$$

$$\vec{u}_{pN} = \frac{\vec{u}'_{pN}}{\left\| \vec{u}'_{pN} \right\|} \qquad \text{(Eq. 3d)}$$

$$\vec{u}_1 = \vec{u}_{p1} + \frac{p-1}{N-1} \times \left[\vec{u}_{pN} - \vec{u}_{p1} \right] \qquad \text{(Eq. 3e)}$$

Where

N	is the number of CCDs for this band (N=3000 or N=6000)
p	is the number of the column (p=1..N)
$(\psi_X)_1$	is the along-track look angle for the first pixel,
$(\psi_Y)_1$	is the across-track look angle for the first pixel,
$(\psi_X)_N$	is the along-track look angle for the last pixel,
$(\psi_Y)_N$	is the across-track look angle for the last pixel.

Case of SPOT5

For SPOT5, look angles $[(\psi_X)_i,(\psi_Y)_i]$ are given for every detectors (i=1..N, where N=3000, 6000 or 12000) and all the bands. The look direction for pixel p is therefore immediately given by the value posted in auxiliary data:

$$\psi_X = (\psi_X)_p \qquad \text{(Eq. 4a)}$$

$$\psi_Y = (\psi_Y)_p \qquad \text{(Eq. 4b)}$$

Where

q	is the number of the column (p=1..N)

$(\psi_X)_p$ is the along-track look angle for the CCD number p (p=1..N),

$(\psi_Y)_p$ is the across-track look angle for the CCD number p (p=1..N),

The look direction for any pixel p is given by the following formula:

$$\vec{u}_1^{'} = \begin{pmatrix} -tg(\psi_Y) \\ +tg(\psi_X) \\ 1 \end{pmatrix} \qquad \text{(Eq. 5a)}$$

$$\vec{u}_1 = \frac{\vec{u}_1^{'}}{\left\|\vec{u}_1^{'}\right\|} \qquad \text{(Eq. 5b)}$$

<u>Look direction and mirror step value (informative)</u>

The look directions of CCDs $[(\psi_X)_i,(\psi_Y)_i]$ includes the physical arrangement of arrays within the telescope and the angle of the mirror.

<u>Case of SPOT123 and SPOT4</u>

Mirror is rotated step by step with incremental values of 0.6°. Mirror position is measured by an integer value in the range [3,93] leading to an angular interval [-27°,+27] (see fig. 19).

Viewing angle for the acquired scene is given by:

$$u_0 = (pms - 48) \times 0.6° \qquad \text{(Eq. 6)}$$

Where

pms is the pointing mirror step (pms=3..93).

This viewing angle is approximately equal to the average of the Y component of the (ψ_Y) angles of the look directions of the extreme detectors on the array.

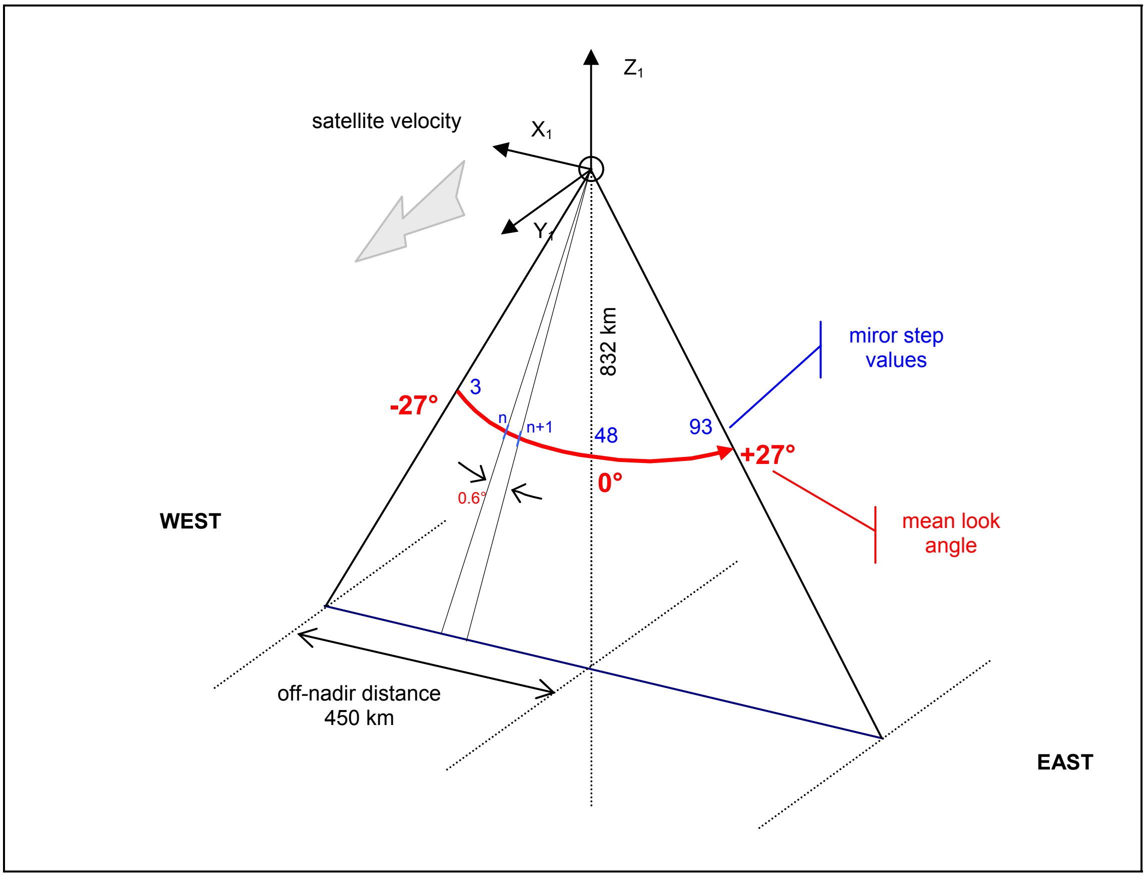


fig. 19 Viewing angle

Case of SPOT5

For SPOT5, the mirror position is monitored by a high accuracy angular encoder. Values of this encoder is in range [-8000,+8000] and precisely indexes the look angle direction.

Step of the optical encoder is 6 micro-rad (approximately one pixel HM) with a precision of 30 micro-rad. Encoder value is transmitted together with the image telemetry at a frequency of 1 Hz; i.e. every 7 kilometres. An average encoder count (*aec*) is computed along the whole segment to provide an unique M_{MCV} matrix used to compute the look directions.

Values of *aec* and M_{MCV} matrix are provided in DIMAP format (see APPENDIX C -).

4.5 Look direction in Orbital Coordinate System

The Orbital Coordinate System

The Orbital Coordinate System (O_2,X_2,Y_2,Z_2) is centered on the satellite, and its orientation is based on the spacecraft position in space. The origin is the spacecraft center of mass O_2, with the Z_2 axis pointing from the Earth center of mass to the spacecraft center of mass. The X_2 axis is the normalized cross product of the instantaneous velocity vector with Z_2 axis. Y_2 is the third unitary vector of the system (see fig. 20).

$$\vec{Z}_2 = \frac{\vec{P}(t)}{\left\|\vec{P}(t)\right\|} \qquad \text{(Eq. 7a)}$$

$$\vec{X}_2 = \frac{\vec{V}(t)\Lambda\vec{Z}_2}{\left\|\vec{V}(t)\Lambda\vec{Z}_2\right\|} \quad \text{(Eq. 7b)}$$

$$\vec{Y}_2 = \vec{Z}_2\Lambda\vec{X}_2 \quad \text{(Eq. 7c)}$$

Where

P(t) is the interpolated position of the satellite computed in eq.2a, and

V(t) is the interpolated velocity of the satellite computed in eq.2b.

Note: Because the trajectory is not perfectly circular, directions of Y_2 and V(t) are close but the vectors are not perfectly collinear.

On SPOT, no distinction is made between the barycentre of instruments O_1 of the Navigation Reference Coordinate System and the spacecraft center of mass O_2 of the Orbital Coordinate System ($O_1=O_2$).

Terrestrial velocities

For SPOT5, position and velocity ephemeris are given in the International Terrestrial Reference Frame (ITRF), being a Earth Centered Rotating (ECR) Coordinate System close to the WGS84 system. In particular, the velocities given in auxiliary data are sum of inertial velocities and instantaneous Earth rotation.

Inertial velocities

For SPOT123 and SPOT4, position and ephemeris are given with reference to GRS80 geodetic system that is also very close to the WGS84.

$$\vec{V}_I = \vec{V}_T - \vec{\Omega} \wedge \vec{P}$$

Where

V_I is the inertial velocity,

V_T is the terrestrial velocity,

Ω is the instantaneous rotation vector of the Earth, and

P is the position of the satellite.

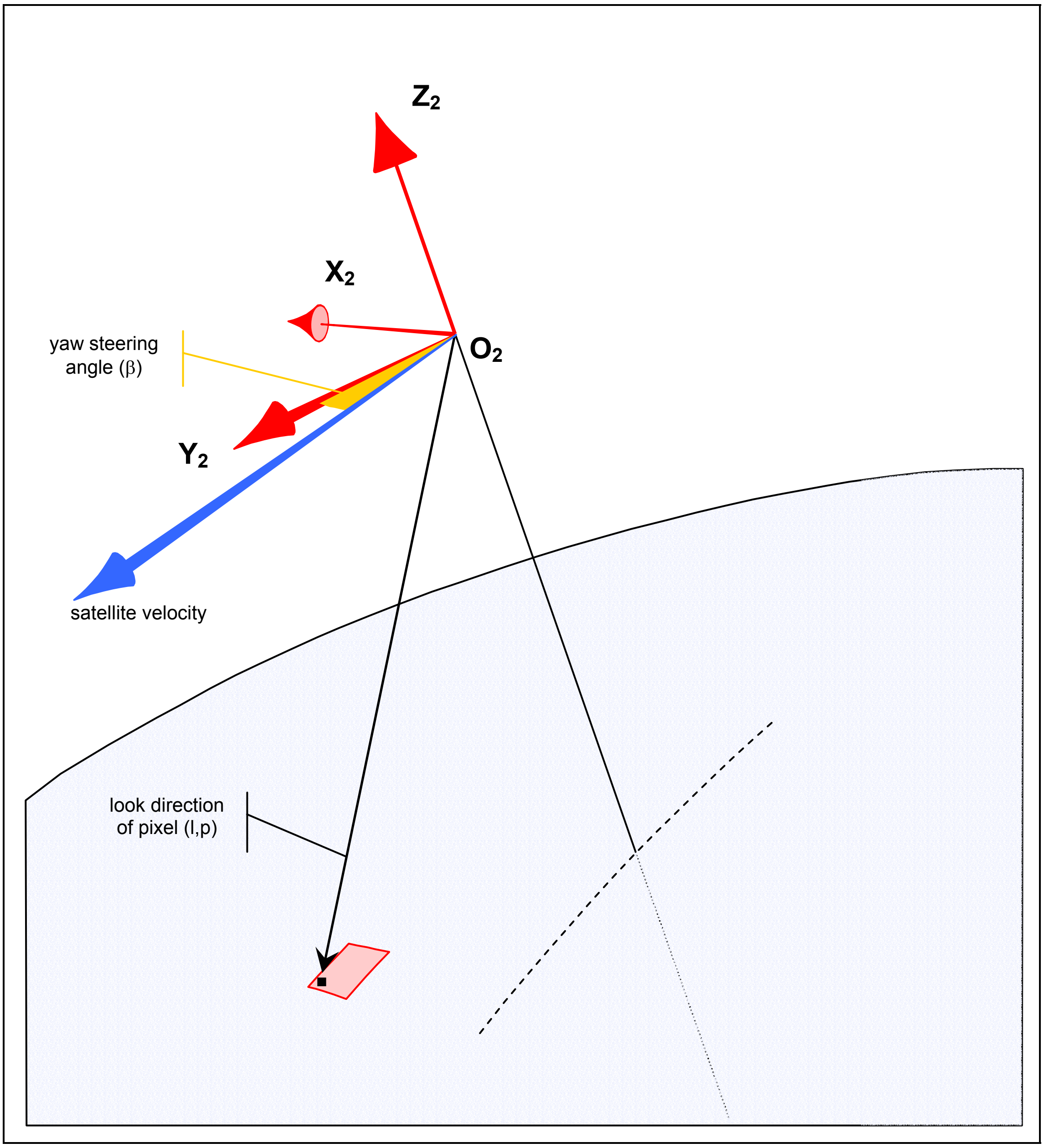


fig. 20 Orbital Coordinate System

SPOT123 and SPOT4 - Integration of attitude variations

All the SPOT satellites have sensors measuring the attitude variations (accelerations): -pitch (rotation speed a_p around X_2 axis), -roll (rotation speed a_r around Y_2 axis), and –yaw (rotation speed a_y around Z_2 axis). These accelerations are turned into angular velocities by the on board computer.

Values of attitude variations are given for a particular line at a frequency of 8 Hz (8 measurements per second). For example, a SPOT4 XS scene of 60 km acquisition includes 72 to 73 attitude samples giving each one the line they are attached to (approximately each 42 lines). Section 4.2 describes the relation between the line number p and the acquisition time.

An example of attitude variation profiles is given in section D.3 of APPENDIX D -.

For anyone of the time t_i (i=1..72) corresponding to the attitude measurements, the absolute attitude is computed performing the integration of the attitude variations (series of elementary rotation velocities).

$$a_p(t_i) = (a_p)_1 + \sum_{j=2}^{i} \left[a'_p(t_j) \times (t_j - t_{j-1}) \right] \quad \text{(Eq. 8a)}$$

$$a_r(t_i) = (a_r)_1 + \sum_{j=2}^{i} \left[a'_r(t_j) \times (t_j - t_{j-1}) \right] \quad \text{(Eq. 8b)}$$

$$a_y(t_i) = (a_y)_1 + \sum_{j=2}^{i} \left[a'_y(t_j) \times (t_j - t_{j-1}) \right] \quad \text{(Eq. 8c)}$$

Where

l_1	is the line of absolute attitude [$(a_p)_1$, $(a_r)_1$, $(a_y)_1$] at the beginning of the scene,
t_1	is the date matching line l_1,
$(a_p)_1$	is the rotation angle around the pitch axis at the beginning of the scene,
$(a_r)_1$	is the rotation angle around the roll axis at the beginning of the scene,
$(a_y)_1$	is the rotation angle around the yaw axis at the beginning of the scene,
i	is the index of the attitude measurement (i=1..72 or 73)
l_i	is the line of average rotation speed [$a_p'(t_i)$, $a_r'(t_i)$, $a_y'(t_i)$],
t_i	is the date matching line l_i
$a_p'(t_i)$	is the average rotation speed around the pitch axis,
$a_r'(t_i)$	is the average rotation speed around the roll axis,
$a_y'(t_i)$	is the average rotation speed around the yaw axis,
$a_p(t_i)$	is the rotation angle around the pitch axis at time t_i
$a_r(t_i)$	is the rotation angle around the roll axis at time t_i
$a_y(t_i)$	is the rotation angle around the yaw axis at time t_i

Initial attitude values

Initial attitude values $(a_p)_1$, $(a_r)_1$, and $(a_y)_1$ are periodically uploaded to the satellite or set to 0 at the beginning of a segment acquisition. These values are measured along the whole segment and guaranty the geometrical continuity between two adjacent scenes of the segment, even when these two scenes have been processed separately.

The uncertainty of the values of these initial attitudes is the major cause of location errors. The best way to enhance this location is to approximate (for example using mean square analysis) these initial values from external data such as ground reference points.

SPOT5 – Absolute attitudes

On-board SPOT5, sensor data are sampled at 1 Hz and data from the gyroscopes at 8 Hz. When mixing these data, ULS quaternions are produced at 8 Hz.

Within SPOT5 products in DIMAP format (section "Dimap_Document/Data_Strip/Satellite_Attitudes"), are provided:

✓ "Raw" attitude values (section "Raw_Attitudes") including:

- attitude values restituted from the inertial central and the terrestrial sensors given as angular and velocity profiles (section "Aocs_Attitude"), and

- attitude values restituted from the star tracker and the inertial central given as quaternions (section "Star_Tracker_Attitude").

✓ "Corrected" attitude values (section "Corrected_Attitudes") which contents depends on the value of "STAR_TRACKER_USED" field:

- angular profiles of AOCS attitudes when " STAR_TRACKER_USED" has value "N", or
- angular profiles of ULS attitudes when " STAR_TRACKER_USED" has value "Y".

The star tracking unit (ULS) on-board SPOT5 allows measuring absolute attitude values. Even if also rotation speeds are present in auxiliary data, it is not necessary to integrate these values.

Precision of ULS system will allow an absolute location better than 50 metres on-ground.

On the contrary, in the case ULS would not behave correctly, absolute attitudes present in auxiliary are computed from the rotations speeds produced by the gyros in the way described in the previous section.

For SPOT5 only, attitude are flagged with a "OUT_OF_RANGE" indicator $a_{oor}(t_i)$ (value "Y" or "N") defining whether attitude value $[a_p(t_i), a_r(t_i), a_y(t_i)]$ is invalid or not.

Sign of attitude data

For historical reasons, attitude values (rotation speed or absolute angle) are not expressed within the Navigation Reference Coordinate System $(0_1,X_1,Y_1,Z_1)$ (see section 4.4) but within its inverted system $(0_1,X_1',Y_1',Z_1')$.

$$\vec{X}_1' = -\vec{X}_1 \\ \vec{Y}_1' = -\vec{Y}_1 \\ \vec{Z}_1' = \vec{Z}_1 \qquad \text{(Eq. 9)}$$

Sign of roll and pitch values (rotation speed or absolute angle) found in auxiliary data will therefore be multiplied by (-1) except the yaw values which will be left unchanged.

This change of sign will appear within the formula computing the look direction here after.

Attitude interpolation

Attitude variations being small and because of the presence of many samples within the scene (see Appendix D.3), only a linear interpolation is performed to get the attitude values $[a_p(t),a_r(t),a_y(t)]$ at the look time t matching the line l of the image.

$$a_p(t) = a_p(t_i) + \left(a_p(t_{i+1}) - a_p(t_i)\right) \times \frac{t - t_i}{t_{i+1} - t_i} \qquad \text{(Eq. 10a)}$$

$$a_r(t) = a_r(t_i) + \left(a_r(t_{i+1}) - a_r(t_i)\right) \times \frac{t - t_i}{t_{i+1} - t_i} \qquad \text{(Eq. 10b)}$$

$$a_y(t) = a_y(t_i) + \left(a_y(t_{i+1}) - a_y(t_i)\right) \times \frac{t - t_i}{t_{i+1} - t_i} \qquad \text{(Eq. 10c)}$$

Where

i	is the index of valid attitude measurement whose time is just before t ($t_i \le t \le t_{i+1}$)
$a_p(t)$	is the rotation angle around the pitch axis at time t
$a_r(t)$	is the rotation angle around the roll axis at time t
$a_y(t)$	is the rotation angle around the yaw axis at time t

$a_p(t_i)$ is the rotation angle around the pitch axis at time t_i computed by using equation 8a or found in auxiliary data

$a_r(t_i)$ is the rotation angle around the roll axis at time t_i computed by using equation 8b or found in auxiliary data

$a_y(t_i)$ is the rotation angle around the yaw axis at time t_i computed by using equation 8c or found in auxiliary data

Computing look direction

The three rotations associated to attitude variations are applied to the look direction u_1 computed within the Navigation Reference Coordinate System.

$$\vec{u}_2 = \frac{\vec{u}_2^{'}}{\left\|\vec{u}_2^{'}\right\|} \qquad \text{(Eq. 11a)}$$

$$\vec{u}_2^{'} = M_p \bullet M_r \bullet M_y \bullet \vec{u}_1 \qquad \text{(Eq. 11b)}$$

$$M_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(a_p(t)) & \sin(a_p(t)) \\ 0 & -\sin(a_p(t)) & \cos(a_p(t)) \end{bmatrix} \qquad \text{(Eq. 11c)}$$

$$M_r = \begin{bmatrix} \cos(a_r(t)) & 0 & -\sin(a_r(t)) \\ 0 & 1 & 0 \\ \sin(a_r(t)) & 0 & \cos(a_r(t)) \end{bmatrix} \qquad \text{(Eq. 11d)}$$

$$M_y = \begin{bmatrix} \cos(a_y(t)) & -\sin(a_y(t)) & 0 \\ \sin(a_y(t)) & \cos(a_y(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{(Eq. 11e)}$$

Where

u_1 is the look direction in the Navigation Reference Coordinate System

$a_p(t)$ is the rotation angle around the pitch axis at time t

$a_r(t)$ is the rotation angle around the roll axis at time t

$a_y(t)$ is the rotation angle around the yaw axis at time t

4.6 Look direction in Terrestrial Coordinate System

International Terrestrial Reference Frame (ITRF)

The International Earth Rotation Service (IERS) has been established since 1988 jointly by the International Astronomical Union (IAU)and the International Union of Geodesy and Geophysics (IUGG). The IERS mission is to provide to the worldwide scientific and technical community reference values for Earth orientation parameters and reference realizations of internationally accepted celestial and terrestrial reference systems.

The IERS is in charge to realize, use and promote the International Terrestrial Reference System (ITRS) as defined by the IUGG resolution No 2 adopted in Vienna,1991.

In the geodetic terminology, a reference frame is a set of points with their coordinates (in the broad sense) which realize an ideal reference system. The frames produced by IERS as realizations of ITRS are named **International Terrestrial**

Reference Frames (ITRF). Such frames are all (or a part of) the tracking stations and the related monuments which constitute the IERS Network, together with coordinates and their time variations.

The measurements of the Earth's rotation are under the form of time series of the so-called Earth Orientation Parameters (EOP). Universal time (UT1), polar motion and the celestial motion of the pole (precession/nutation) are determined by VLBI. The satellite-geodesy techniques, GPS, SLR and DORIS, determine polar motion and the rapid variations of universal time.

The satellite-geodesy programs used in the IERS give access to the time variations of the Earth's gravity field, reflecting the evolution of the Earth's shape, as well as the redistribution of masses in the planet. They have also detected changes in the location of the center of mass of the Earth relative to the crust. This makes it possible to investigate global phenomena such as mass redistributions in the atmosphere, oceans and solid Earth.

Since SPOT2, the DORIS instrument on-board the SPOT satellites contributes to these Earth measurements.

On contrary to most navigation systems, DORIS is based on an uplink device: the receiver is on-board the satellite while the transmitters are on the ground. This creates a naturally centralized system, in which the complete set of observations is downloaded from the satellite to the ground center, from where they are distributed after unified editing.

The current system started operation in 1990. Its permanently tracked network includes 50 beacons evenly distributed on the earth, including stations on all major tectonic plates.

The geocentric positioning results obtained in 1998 have a precision of 2 cm, and daily polar motion determinations have a precision of about 1-2 milliarcseconds.

In general the ITRS (and its realizations ITRFyy) are close to WGS84 at one metre. Table here below shows for example the parameters from ITRF90 to WGS84.

For additional information, visit the Web site http://lareg.ensg.ign.fr/ITRF/index.html.

T1 (m)	T2 (m)	T3 (m)	D (ppm)	R1 (arcsec.)	R2 (arcsec.)	R3 (arcsec.)
0.060	-0.517	-0.223	-0.011	0.0183	-0.0003	0.0070

From Orbital Coordinate System to Terrestrial Coordinate System

Because position P(t) and velocity V(t) are already expressed in ITRF, transformation from Orbital Coordinate System to Terrestrial Coordinate System is reduced to a simple base change.

$$\vec{u}_3 = \begin{bmatrix} (X_2)_X & (Y_2)_X & (Z_2)_X \\ (X_2)_Y & (Y_2)_Y & (Z_2)_Y \\ (X_2)_Z & (Y_2)_Z & (Z_2)_Z \end{bmatrix} \bullet \vec{u}_2 \qquad \text{(Eq. 12)}$$

Where

u_2 is the look direction in Orbital Coordinate System

u_3 is the look direction in Terrestrial Coordinate System

X_2 is the pitch axis computed in eq. 7a

Y_2 is the roll axis computed in eq. 7b

Z_2 is the yaw axis computed in eq. 7c

4.7 Location on Earth model

Intersection of look direction with the Earth ellipsoid

Getting the look direction u_3 from the satellite at the position P(t), we now compute the intersection with the ellipsoid located at an altitude h above the standard ITRF ellipsoid (see fig. 21).

We consider here that the altitude h of the intersection point M is known. This assumption will allow computing the intersection with a topographic surface (see next section). In the case no DEM would be available, altitude h shall be set to 0.

As said in the previous section, the ITRF ellipsoid varies from the WGS84 in a magnitude less than one metre. A reasonable approximation may be performed using the standard WGS84 values:

$$a = 6\,378\,137.0 \text{ m} \qquad \text{(Eq. 13a)}$$

$$b = 6\,356\,752.3 \text{ m} \qquad \text{(Eq. 13b)}$$

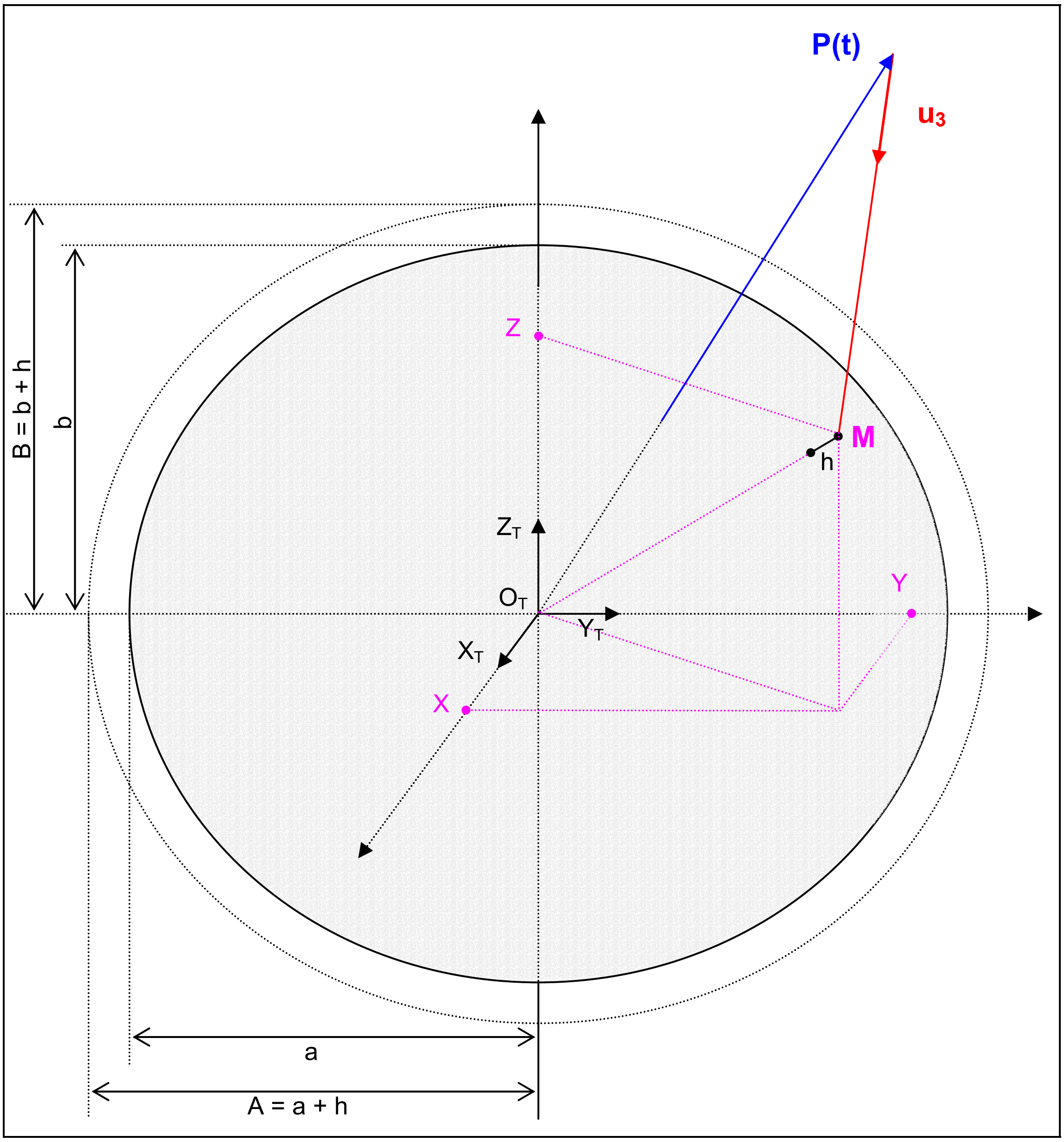


fig. 21 Intersection of look direction with the Earth ellipsoid

Let M=(X,Y,Z) the geocentric coordinates to be found, point M is involved within the two following equations:

$$\overrightarrow{O_3M} = \vec{P}(t) + \mu \times \vec{u}_3 \Rightarrow \begin{cases} X = X_P + \mu \times (u_3)_X \\ Y = Y_P + \mu \times (u_3)_Y \\ Z = Z_P + \mu \times (u_3)_Z \end{cases} \qquad \text{(Eq. 14)}$$

and

$$\frac{X^2 + Y^2}{A^2} + \frac{Z^2}{B^2} = 1 \qquad \text{with } \begin{cases} A = a + h \\ B = b + h \end{cases} \qquad \text{(Eq. 15)}$$

Leading to solve the 2nd degree equation:

$$\left[\frac{(u_3)_X^2 + (u_3)_Y^2}{A^2} + \frac{(u_3)_Z^2}{B^2}\right] \times \mu^2 + 2 \times \left[\frac{X_P(u_3)_X + Y_P(u_3)_Y}{A^2} + \frac{Z_P(u_3)_Z}{B^2}\right] \times \mu + \left[\frac{X_P^2 + Y_P^2}{A^2} + \frac{Z_P^2}{B^2}\right] = 1$$

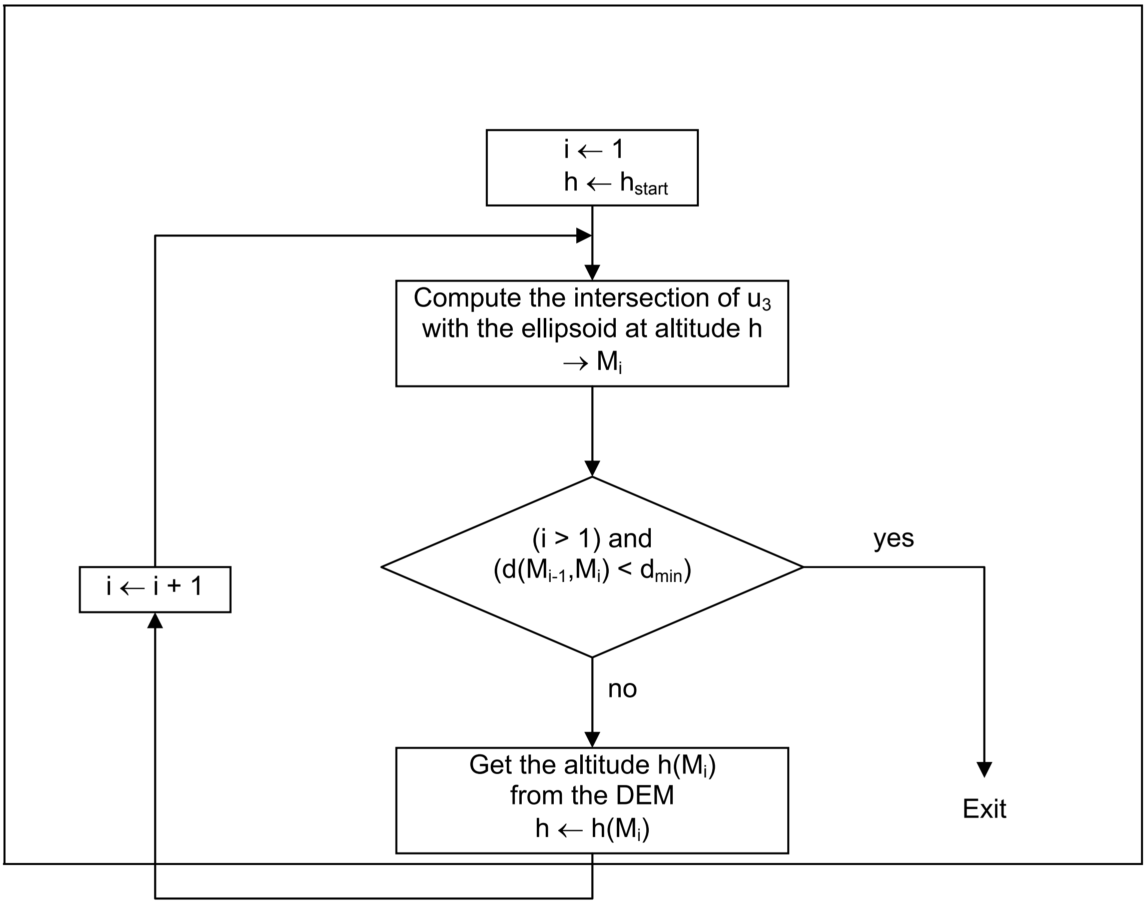
(Eq. 16)

This equation has necessarily two distinct solutions (μ_1, μ_2). The smallest one (μ_{min}) shall be kept. Re-introducing this value within equation 14 gives the geocentric coordinates (X,Y,Z) of point M.

The geodetic coordinates (λ,φ,h) may be computed using the procedure given in Appendix A.1).

Using a DEM - Algorithm

When a DEM is available, the intersection with the topographic surface may be computed using the iterative algorithm detailed here below and illustrated by fig. 22.



Algorithm stops when the geodetic distance between the computed intersection M_i and the one computed at the previous step becomes below a fixed threshold (d_{min}).

Speeding-up convergence

First altitude h_{start} shall be chosen as close as possible the final one. When pixels are processed sequentially, along the line, it is recommended to get the altitude of the previous pixel; or better, to estimate the local slope, setting the altitude in the continuity of this slope.

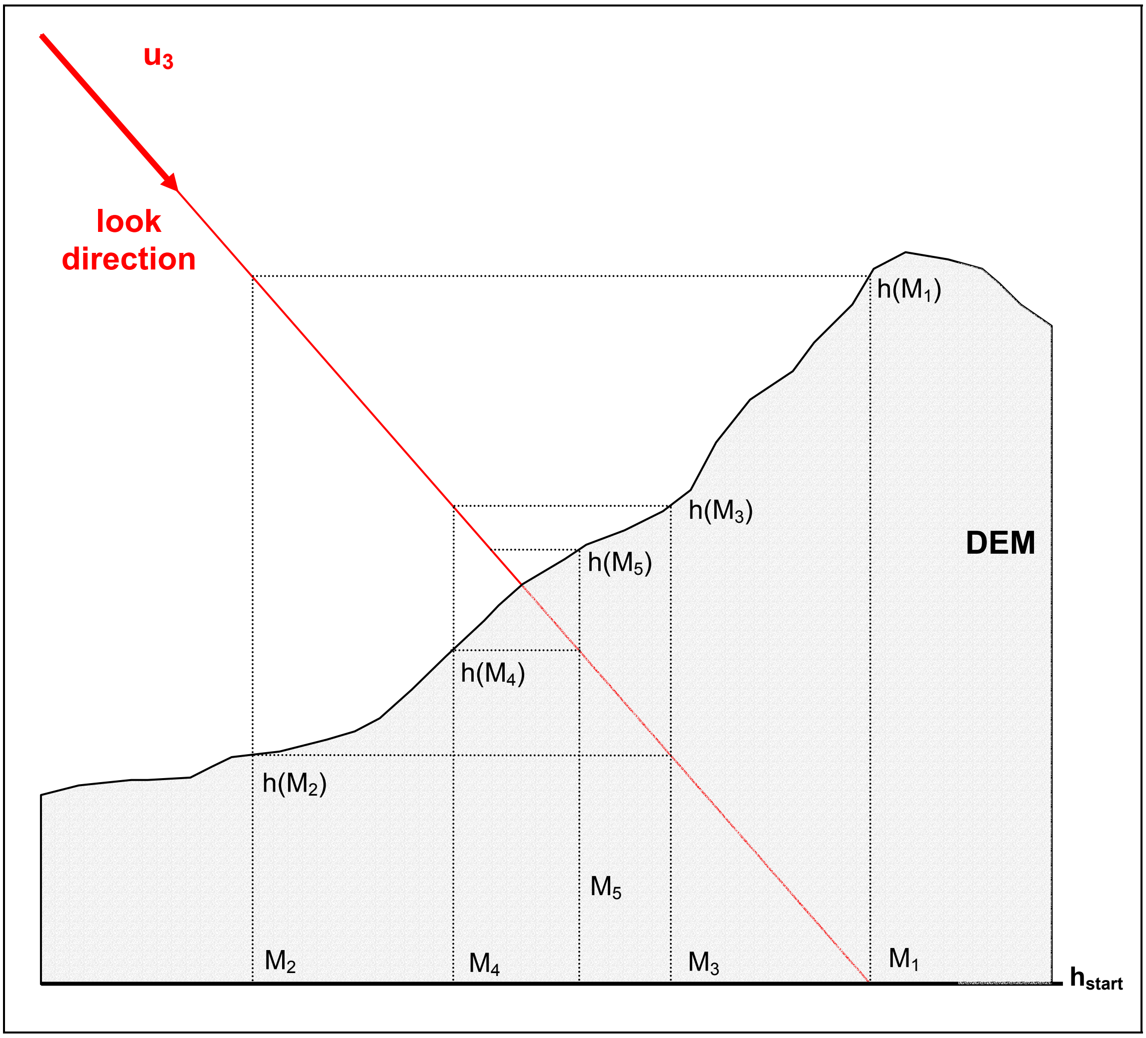


fig. 22 Intersection of look direction with a topographic surface

Algorithm convergence

When the DEM has a high resolution and when the look direction is very inclined, algorithm may diverge; in particular, within areas showing slopes with different orientations.

Various strategies may be adopted:
1. stop when $d(M_{i-1},M_i)$ is not strictly decreasing,
2. process a DEM pyramid at various resolution ("coarse to fine" algorithm),
3. use DEM represented by facets (TIN),
4. …

Using a DEM – Altitude pre-processing

In many cases, DEMs have altitude relative to custom geoid or ellipsoid.

To be efficiently used in the algorithm presented here above, the DEM shall be projected in ITRF and its altitudes shall be in reference to ITRF ellipsoid (see fig. 23).

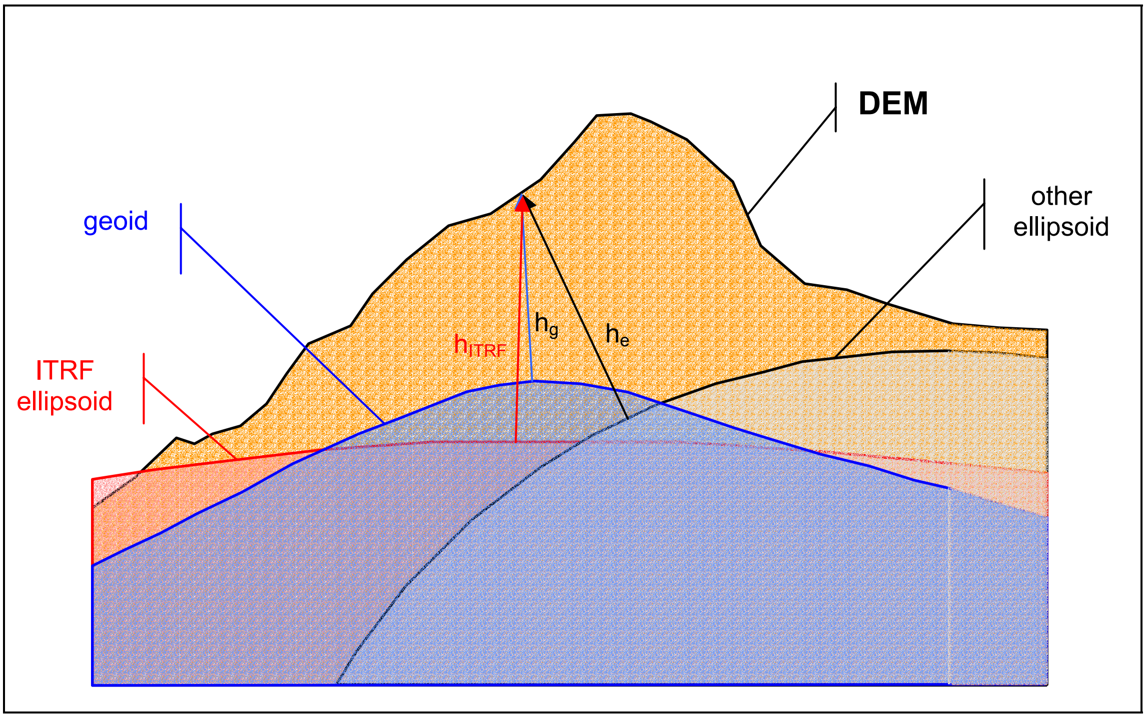


fig. 23 Altitude referential of a DEM

5 GENERATION OF LEVEL 2 PRODUCTS – THE INVERSE MODEL

5.1 Principle

Previous section (see 4) defines a direct location model:

$$(\lambda, \varphi, h) = f(l, p, h) \Leftrightarrow \begin{cases} \lambda = f_\lambda(l, p, h) \\ \varphi = f_\varphi(l, p, h) \end{cases} \qquad \text{(Eq. 17)}$$

Where

l	is the number of the line in SPOT 1A image (first line being 1)
p	is the number of the pixel of the line in SPOT 1A image (first pixel being 1)
h	is the altitude of point above ITRF ellipsoid
λ	is the longitude of matching point in Earth Reference System
φ	is the latitude of matching point in Earth Reference System

A rough reverse location model f_h^{-1} for a given altitude h is computed from a regular grid of points $(p_i,q_i,\lambda_i,\varphi_i)$ where (λ_i,φ_i) are results of the direct location model applied to points (p_i,q_i,h).

For any geodetic coordinates (λ,φ), the matching point (l,p) in level 1A image is retrieved iteratively combining a polynomial prediction of this reverse location model with a refinement using the direct location model.

When Ground Control Points are available, the geometrical viewing model may be registeres minimizing parameters of this model. Document **Error! Reference source not found.**, and in particular its Annexe 2, describes the algorithms to achieve such registration.

5.2 The geodetic to image predictor

Sampled data (p_i,q_i)

We suggest to use a polynomial interpolating the (l_i,p_i) points sampled with a period of 5 pixels x 5 lines (one line each 5 lines along the scene or the segment, and one pixel each pixel along the line).

Computing the polynomial predictor

Two polynomials f_{hmin}^{-1} and f_{hmax}^{-1} are computed according to two altitudes hmin and hmax; where coefficients $(a_{k,l},b_{k,l})$ of each one of the two polynomials are computed to minimize the mean quadratic error among the sample series $(l_i,p_i,\lambda_i,\varphi_i,h_{min})$ and $(l_i,p_i,\lambda_i,\varphi_i,h_{max})$ respectively.

$$(l, p) = f_h^{-1}(\lambda, \varphi) \Leftrightarrow \begin{cases} \left(f_h^{-1}\right)_p(\lambda, \varphi) = l = \sum_{k=0}^{K}\sum_{l=0}^{L} a_{k,l} \times \lambda^k \times \varphi^l \\ \left(f_h^{-1}\right)_q(\lambda, \varphi) = p = \sum_{k=0}^{K}\sum_{l=0}^{L} b_{k,l} \times \lambda^k \times \varphi^l \end{cases} \qquad \text{(Eq. 18)}$$

Where

hmin	is the assumed minimum altitude within the whole scene or segment
hmax	is the assumed maximum altitude within the whole scene or segment
K, L	are the degree of the polynomials (we suggest to use K=L=3)

Use of the predictor

Given an altitude h at a point (λ,φ), the location (l,p) within the level 1A image is computed interpolating linearly the locations predicted by the reverse location models for altitudes hmin and hmax (see fig. 24).

$$f^{-1}(\lambda,\varphi,h)=(l,p)\Leftrightarrow\begin{cases} l=k\times\left(f_{h\max}^{-1}\right)_p(\lambda,\varphi)+(1-k)\times\left(f_{h\min}^{-1}\right)_p(\lambda,\varphi) \\ p=k\times\left(f_{h\max}^{-1}\right)_q(\lambda,\varphi)+(1-k)\times\left(f_{h\min}^{-1}\right)_q(\lambda,\varphi)\end{cases}\qquad\text{(Eq. 19)}$$

$$k=\frac{h-h_{\min}}{h_{\max}-h_{\min}}$$

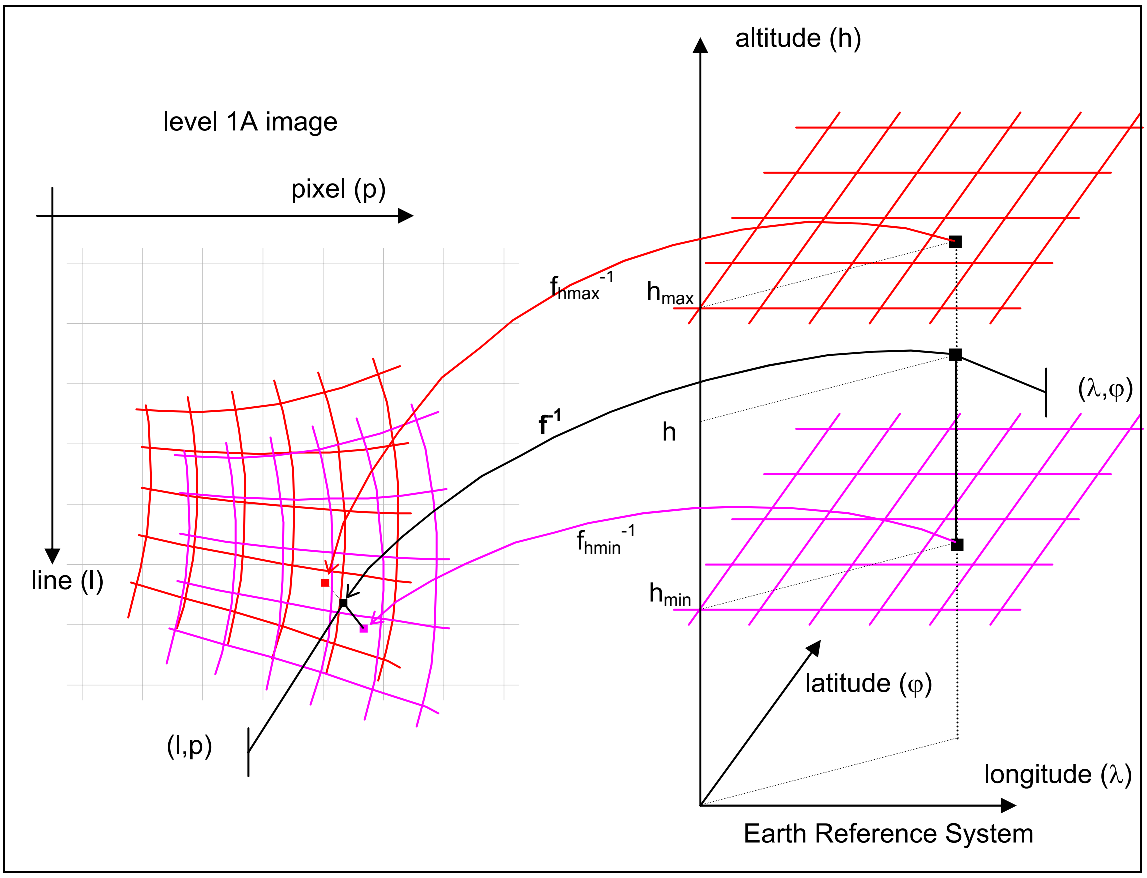


fig. 24 Reverse location predictor

5.3 Algorithm

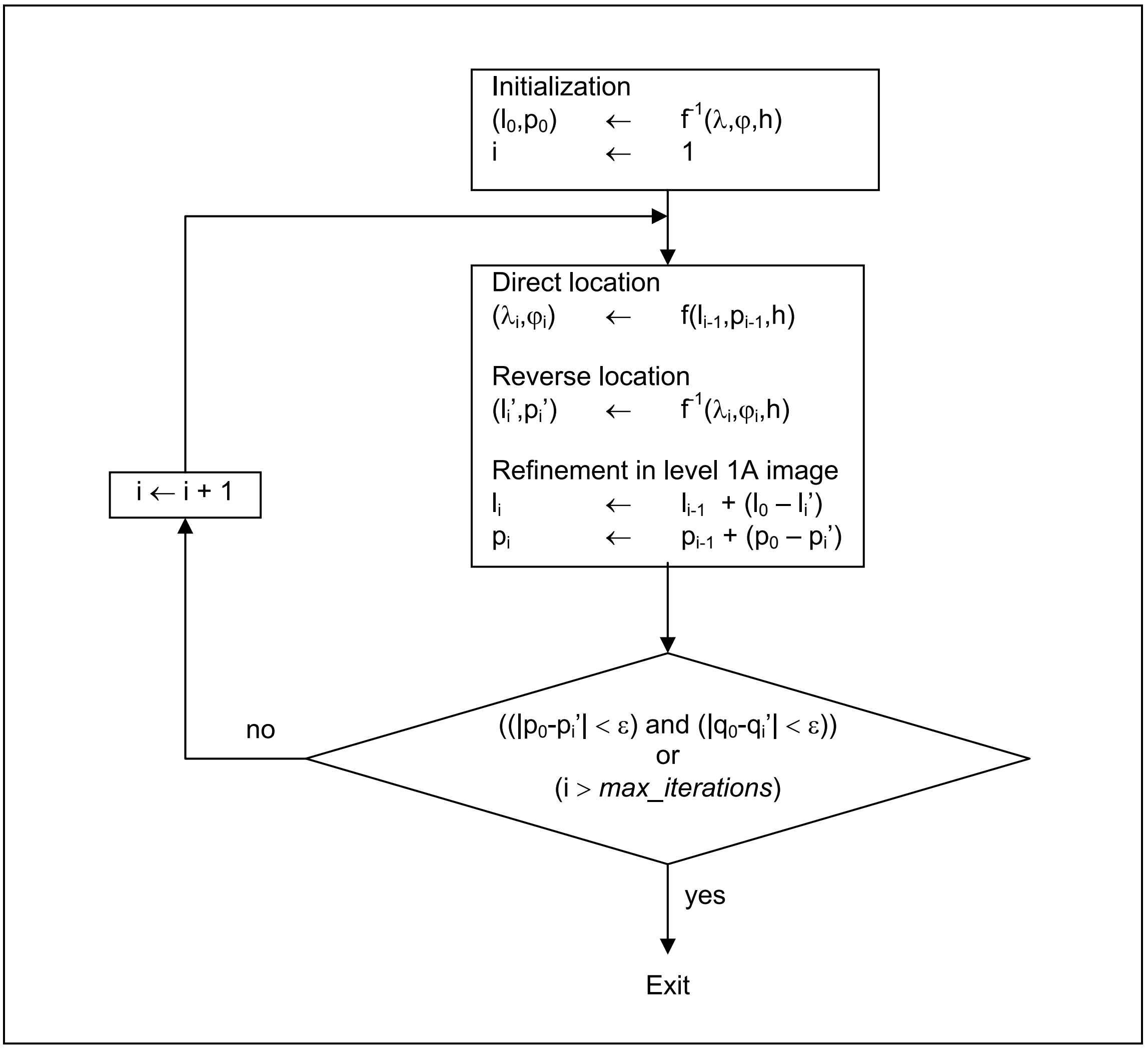
The purpose of the algorithm is to determine the point (p_n,q_n) in level 1A image that has an image as close as possible of (λ,φ) point in Earth reference System by the direct location model f and considering an altitude h (this altitude is given by a DEM).

$$f(l_n,p_n,h)\approx(\lambda,\varphi,h)\qquad\text{(Eq. 20)}$$

Let (l_0,p_0) be the point obtained applying the reverse location predictor f^{-1} to (λ,φ), transforming this point by the direct location model f and turning back by the reverse predictor allows to compute a first refinement of the location of (l_0,p_0) leading to point (l_1,p_1).

This iterative processing is repeated leading to a series of location refinements smaller and smaller.

The figure here below (see fig. 25) shows the first two iterations of this refinement loop.



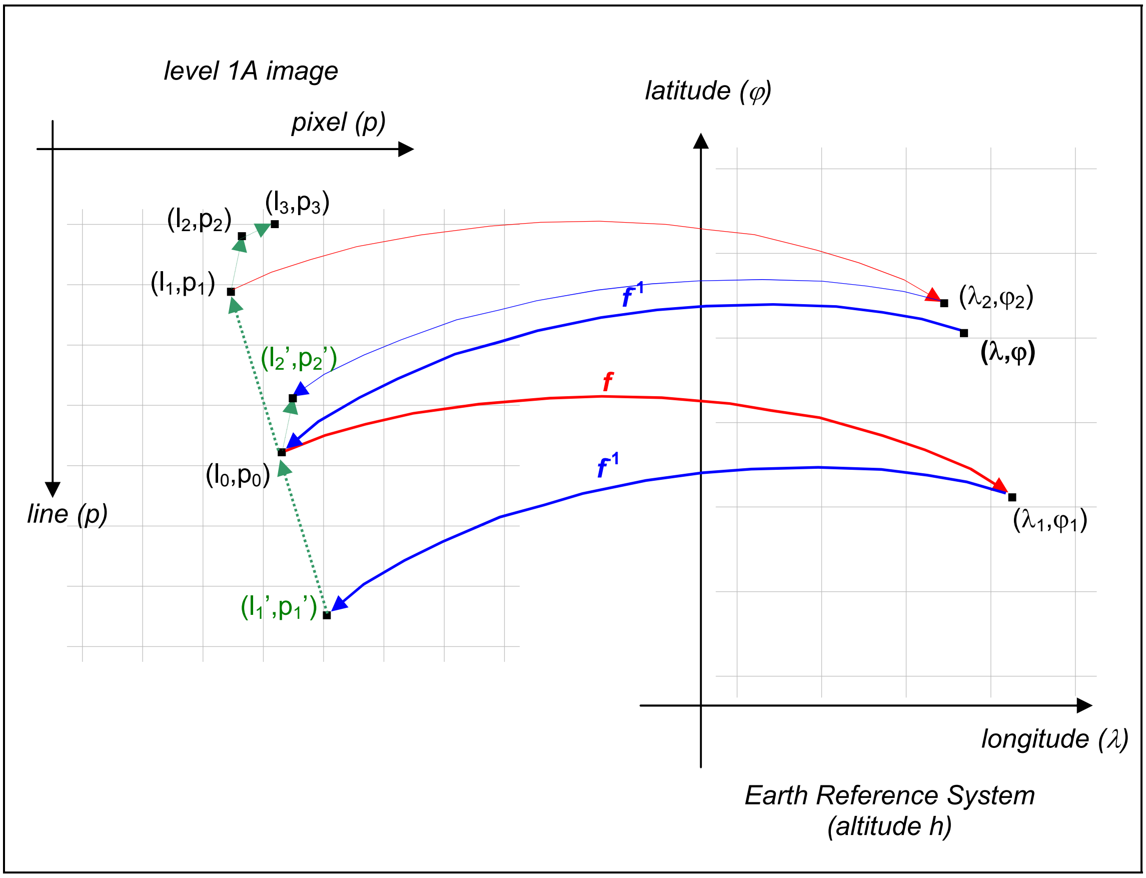


fig. 25 Reverse location iterative refinements

5.4 Using level 1B products in input

Section 5 is basically aimed to process level 1A scenes in input. Nevertheless, cases may occur when only level 1B scenes are available in input.

Note that this situation is not optimal because level 1B images have already been interpolated.

Rationale

As mentioned in section 3.1, level 1B products have been processed to minimize the internal distortion of terrestrial landscapes.

In particular, within flat areas where the altitude of points of the scene is close the ellipsoid used to process this level 1B product, a post-processing to make match the SPOT image with a cartographic referential may be achieved with few ground control points (GCP).

At the opposite, the production of an orthorectified product matching a cartographic referential within mountainous areas (or more generally within areas showing important altitude variations) requires to “undo” the level 1B processing in order to return to a level 1A product.

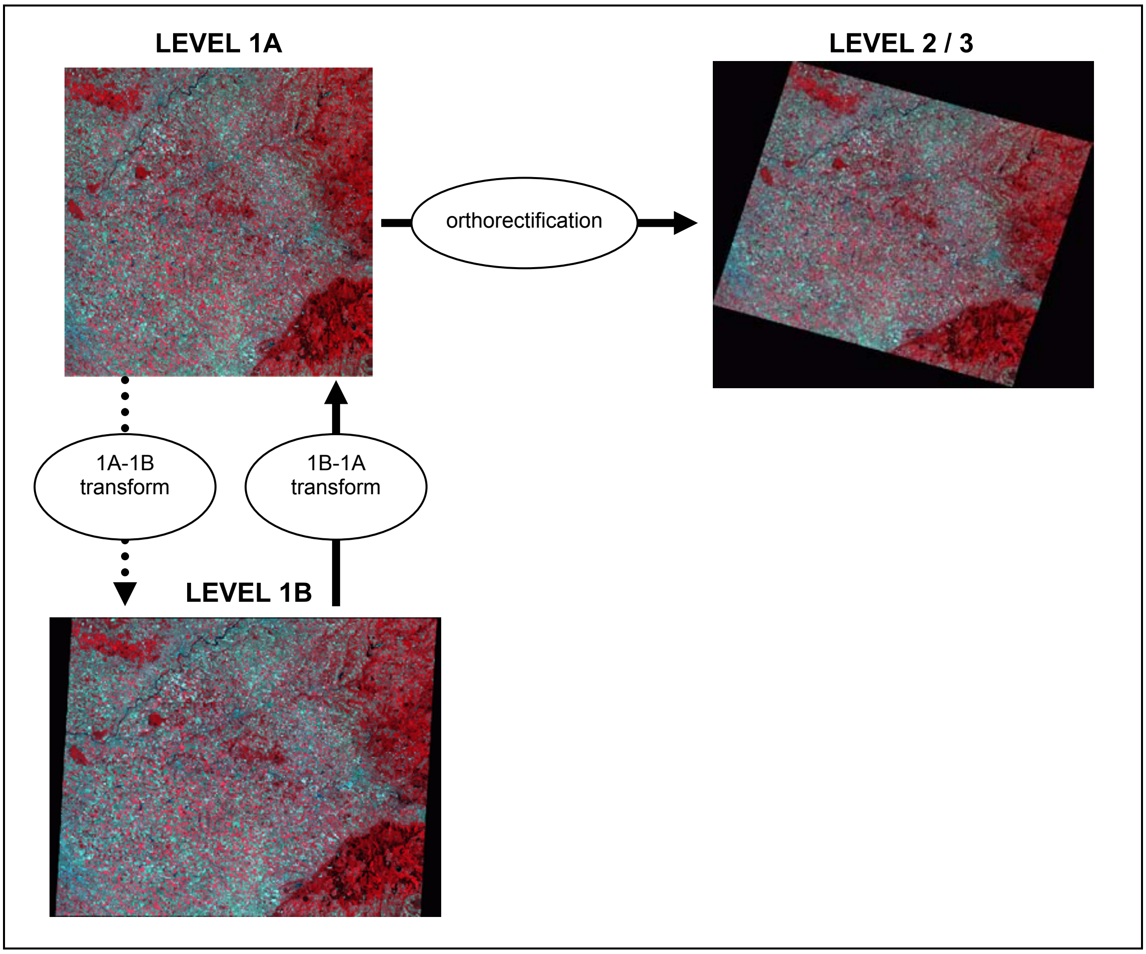


fig. 26 From level 1B to level 2/3

Retrieving the pixel location in level 1B image

To produce a level 2 or 3 orthorectified product from a level 1B scene, it is not necessary to first produce a 1A scene from the level 1B scene. Such strategy would consume time and would lead to a degradation of the data interpolating once again data.

The algorithm described in section 5.3 allows to retrieve the coordinates (l,p) in level 1A scene corresponding to the geodetic point (λ,φ,h) on Earth.

Equations 21 (case of scene-based transform) or 25 (case of segment-based transform) of the "1A to 1B direct model" allows to retrieve the corresponding point (L,P) in level 1B scene.

Choice between the two formulae depends on the date level 1B product has been generated (see section 6.1).

6 PROCESSING OF 1B PRODUCTS

Scope of this section is to explain how SPOT level 1B scenes are produced and to provide the formulae allowing to establish the correspondence between level 1A and level 1B coordinates system.

6.1 Historical background

Level 1B scenes have been produced in two different ways.

SISA model

From the delivery of first SPOT products in March 1986 to February 1995, level 1B scenes were produced using a simple polynomial model. This model is based on the transform of the central line and column of the scene and leads to misregistration between successive scenes along the path (see fig. 27).

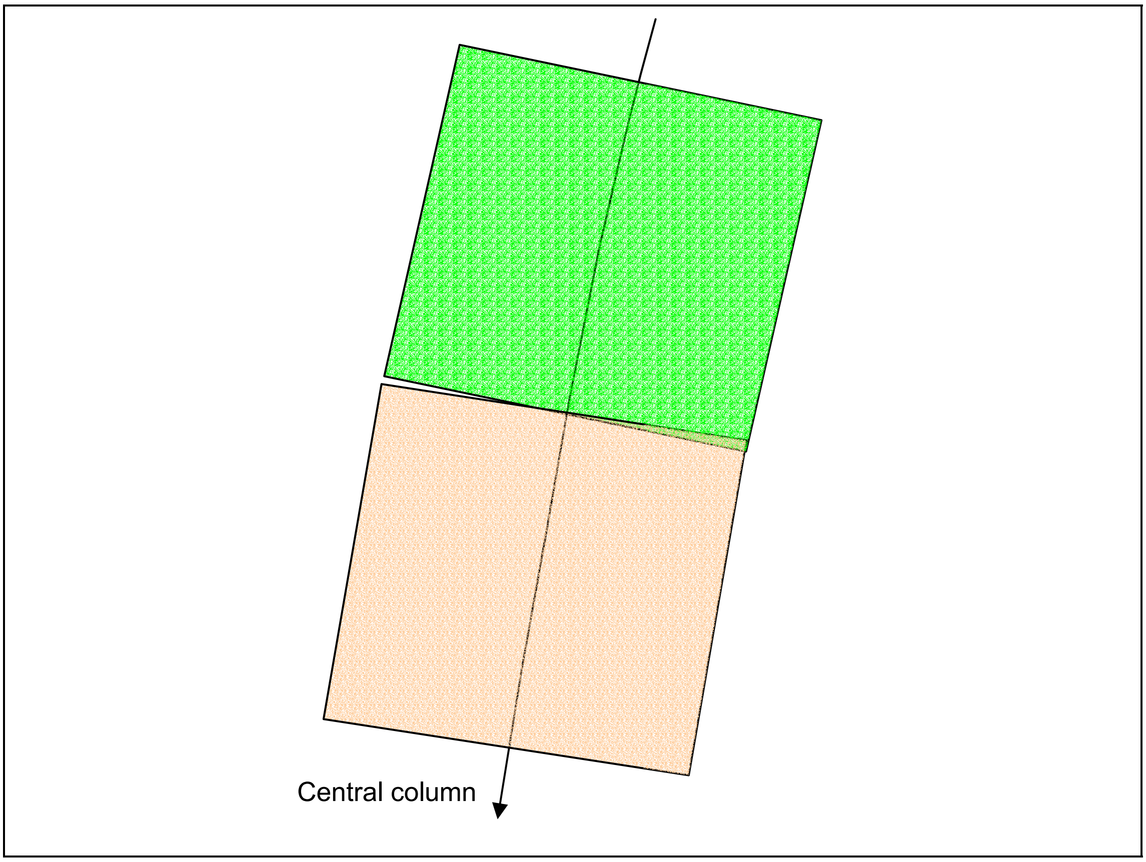


fig. 27 Misregistration defect in SISA model

CAP model

Since March 1995, this defect has been overcome setting a model on the whole segment.

6.2 Generation of 1B products

Geometry of level 1B products is close to a cartographic projection, generally a Cartesian system (for example UTM), combined with a rotation.

The transformation from (or to) coordinates (l,p) in the level 1A raw scene to (or from) the coordinates (L,P) is estimated minimizing the mean square errors from a set of samples and using the direct model described in section 4. This viewing geometry model is applied to a standard ellipsoid and assuming altitudes of intersections with this ellipsoid equal to 0.

In the scene-based transform, sampled are regularly spaced within the scene. In segment-based transform, samples are regularly dispatched along the whole segment. Document R-2 recommends to use a grid compound of 4 columns by 10 equally spaced rows along the segment (see fig. 28).

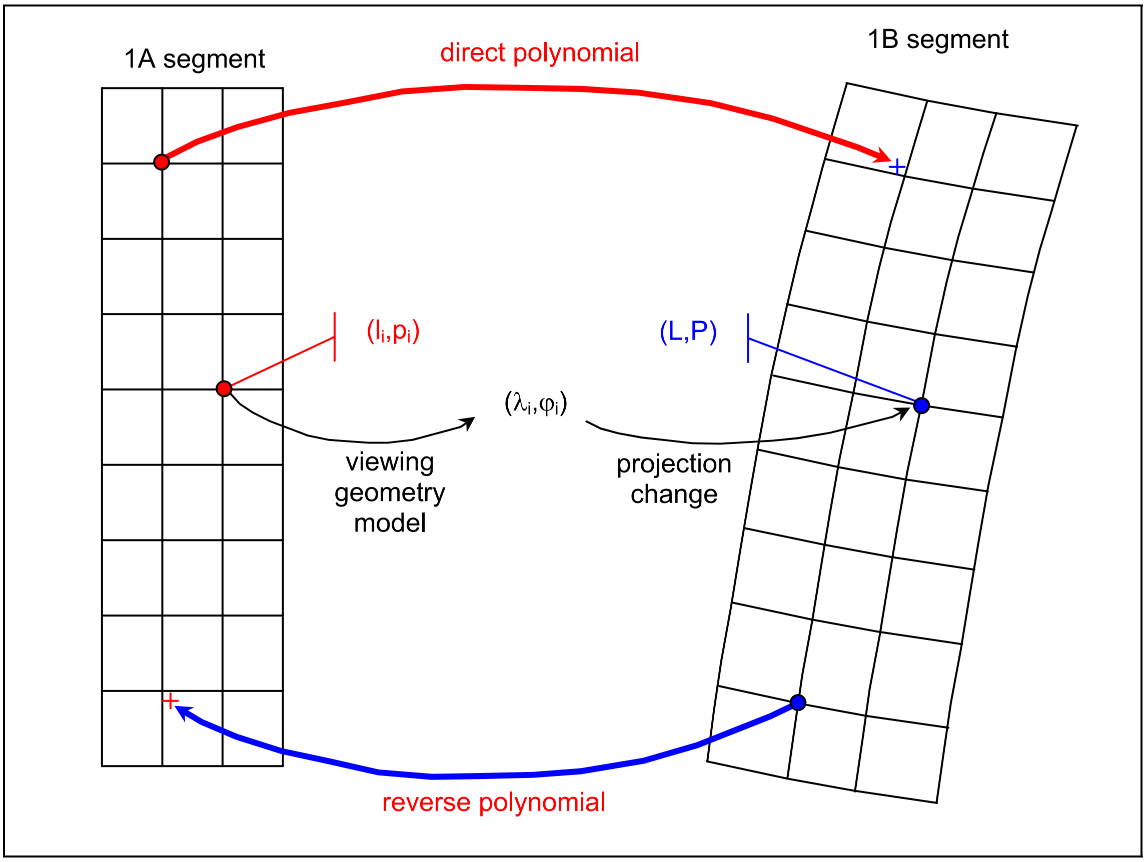


fig. 28 Samples distribution in the segment-based transform.

6.3 Method 1 – Scene-based transform (SISA)

Five polynomials A, B, C, D and E of degree 4 are used to compute the direct transform [(l,p) → (L,P)], the reverse transform [(L,P) → (l,p)] and the line translations (skew).

The four coefficients of each polynomial have to be found within the Ephemeris record of the LEADER FILE (see SISA format document R-9). The field number and byte location are specified within the equations here below.

Direct transform

Transform from 1A coordinates to 1B coordinates is given by the following formulae:

$$L = a_0 + a_1 \times l + a_2 \times l^2 + a_3 \times l^3 \qquad \text{(Eq. 21a)}$$

$$P' = b_0 + b_1 \times p + b_2 \times p^2 + b_3 \times p^3 \qquad \text{(Eq. 21b)}$$

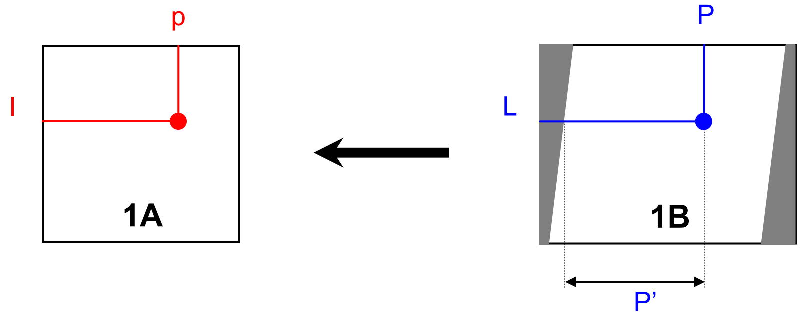
$$\Delta P = e_0 + e_1 \times L + e_2 \times L^2 + e_3 \times L^3 \qquad \text{(Eq. 21c)}$$

$$P = \Delta P + P' \qquad \text{(Eq. 21d)}$$

Where

- (l,p) are the coordinates of the pixel in 1A image (first pixel being (1,1)),
- (L,P) are the coordinates of the pixel in 1B image (first pixel being (1,1)),
- P' is the coordinate along the image line (without left background),
- ΔP is the line translation (number of left background pixels),
- (a_i) are the four coefficients of polynomial A (resampling of the raw central column) – field 20 #3001-3032,
- (b_i) are the four coefficients of polynomial B (resampling of the raw central line without line shift) – field 21 #3033-3064,
- (e_i) are the four coefficients of polynomial E (line translation model) – field 24 #3129-3160.

Reverse transform



Transform from 1B coordinates to 1A coordinates is given by the following formulae:

$$l = c_0 + c_1 \times L + c_2 \times L^2 + c_3 \times L^3 \qquad \text{(Eq. 22a)}$$

$$p = d_0 + d_1 \times P' + d_2 \times P'^2 + d_3 \times P'^3 \qquad \text{(Eq. 22b)}$$

Where

(l,p) are the coordinates of the pixel in 1A image (first pixel being (1,1)),

(L,P) are the coordinates of the pixel in 1B image (first pixel being (1,1)),

P' is the coordinate along the image line (without left background) given by equations 21c and 21d,

(c_i) are the four coefficients of polynomial C (resampling model along columns) – field 22 #3065-3096,

(d_i) are the four coefficients of polynomial D (resampling model along lines) – field 23 #3097-3128.

6.4 Method 2 – Segment-based transform (CAP)

Transformations between level 1A and level 1B coordinate systems are performed passing through segment-based reference systems. The polynomial transform is computed between these two segment-based reference systems.

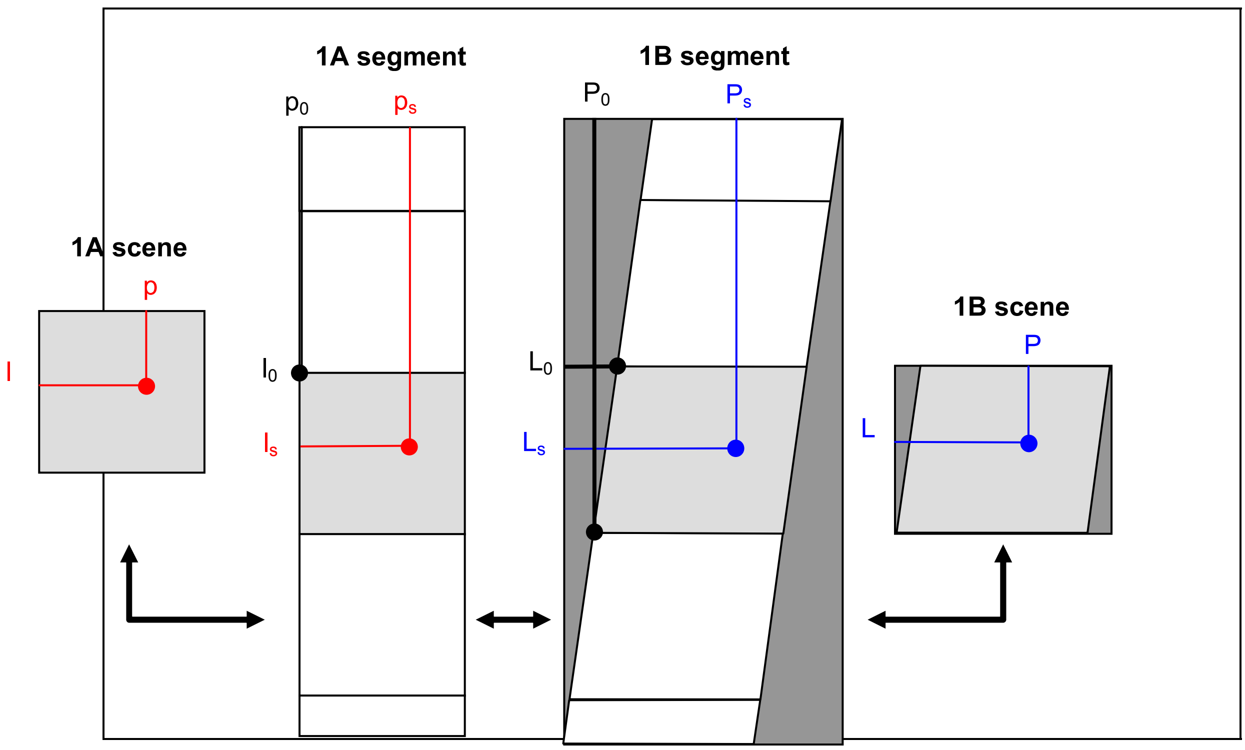


fig. 29 Level 1A to/from 1B reference systems.

To avoid accuracy lost in computation, the mean square analysis (see section 6.2), is performed from centered and normalized values; i.e. all the values are set in range [0,1]. As a consequence, the polynomial transforms (direct or reverse) shall involve centered and normalized values (see figure fig. 30).

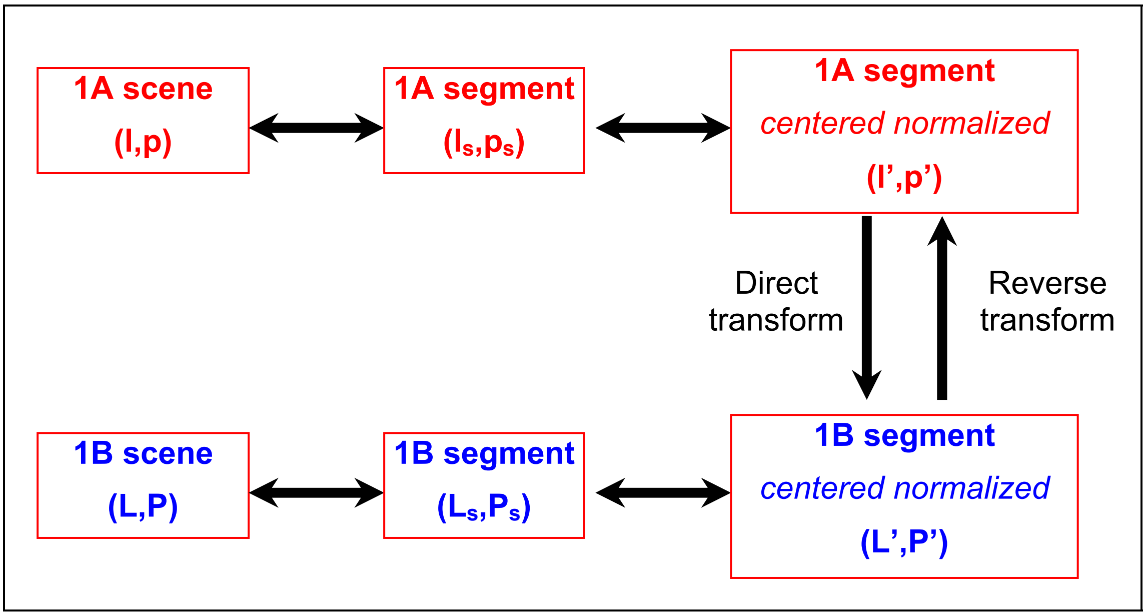


fig. 30 Level 1A to/from 1B processing steps.

1A scene to / from 1A segment

This transform is a simple translation from the values found in ancillary data.

$$\begin{cases} l_s = l_0 + l \\ p_s = p_0 + p \end{cases} \qquad \text{(Eq. 23)}$$

Where

(l,p)	are the coordinates of the pixel in 1A scene (first pixel being (1,1)),
(l_s,p_s)	are the coordinates of the pixel in 1A segment (first pixel being (1,1)),
l_0	is the first line of the raw scene in the raw segment,
p_0	is the first pixel of the raw scene in the raw segment (q_0=1).

1A segment centered and normalized coordinates

Centering and normalization of (l_s,p_s) coordinates are processed from the mean values l_m or p_m and the half-widths $\Delta l=(l_{max}-l_{min})/2$ or $\Delta p=(p_{max}-p_{min})/2$ of the value range.

$$\begin{cases} l' = \dfrac{l_s - l_m}{\Delta l} \\ p' = \dfrac{p_s - p_m}{\Delta p} \end{cases} \qquad \text{(Eq. 24)}$$

Where

(l_s,p_s)	are the coordinates of the pixel in 1A segment (first pixel being (1,1)),
(l',p')	are the centered and normalized coordinates of the pixel in 1A segment,
l_m	is the mean value of lines l in 1A segment,

Δl is the interval half-width for lines l in 1A segment,

p_m is the mean value of columns p in 1A segment,

Δp is the interval half-width for columns p in 1A segment.

Direct transform (1A → 1B)

Along lines and along columns models are used on centered and normalized coordinates. These involve four polynomials I, J, A and B respectively of degree 5, 3, 4 and 5.

$$\begin{cases} L' = I(l') \\ P' = \dfrac{J(p') + B(L')}{A(L')} \end{cases} \quad \text{(Eq. 25a)}$$

$$\begin{aligned} I(l') &= i_0 + i_1 \times l' + i_2 \times l'^2 + i_3 \times l'^3 + i_4 \times l'^4 + i_4 \times l'^5 \\ J(p') &= j_0 + j_1 \times p' + j_2 \times p'^2 + j_3 \times p'^3 \\ A(L') &= a_0 + a_1 \times L' + a_2 \times L'^2 + a_3 \times L'^3 + a_4 \times L'^4 \\ B(L') &= b_0 + b_1 \times L' + b_2 \times L'^2 + b_3 \times L'^3 + b_4 \times L'^4 + b_5 \times L'^5 \end{aligned} \quad \text{(Eq. 25b)}$$

Where

(l',p') are the centered and normalized coordinates of the pixel in 1A segment,

(L',P') are the centered and normalized coordinates of the pixel in 1B segment,

(i_i) are the 6 coefficients of polynomial I,

(j_i) are the 4 coefficients of polynomial J,

(a_i) are the 5 coefficients of polynomial A,

(b_i) are the 6 coefficients of polynomial B.

Reverse transform (1B → 1A)

Along lines and along columns models are used on centered and normalized coordinates. These involve four polynomials L, P, A and B respectively of degree 5, 3, 4 and 5.

$$\begin{cases} l' = L(L') \\ p' = P\left[A(L') \times P' - B(L')\right] \end{cases} \quad \text{(Eq. 26a)}$$

$$\begin{aligned} L(L') &= l_0 + l_1 \times L' + l_2 \times L'^2 + l_3 \times L'^3 + l_4 \times L'^4 + l_4 \times L'^5 \\ P(X) &= p_0 + p_1 \times X + p_2 \times X^2 + p_3 \times X^3 \\ A(L') &= a_0 + a_1 \times L' + a_2 \times L'^2 + a_3 \times L'^3 + a_4 \times L'^4 \\ B(L') &= b_0 + b_1 \times L' + b_2 \times L'^2 + b_3 \times L'^3 + b_4 \times L'^4 + b_5 \times L'^5 \end{aligned} \quad \text{(Eq. 26b)}$$

Where

(l',p') are the centered and normalized coordinates of the pixel in 1A segment,

(L',P') are the centered and normalized coordinates of the pixel in 1B segment,

(l_i) are the 6 coefficients of polynomial L,

(p_i) are the 4 coefficients of polynomial P,

(a_i) are the 5 coefficients of polynomial A,

(b_i) are the 6 coefficients of polynomial B.

1B segment centered and normalized coordinates

Centering and normalization of (L_s,P_s) coordinates are processed from the mean values L_m or P_m and the half-widths $\Delta L=(L_{max}-L_{min})/2$ or $\Delta P=(P_{max}-P_{min})/2$ of the value range.

$$\begin{cases} L' = \dfrac{L_s - L_m}{\Delta L} \\ P' = \dfrac{P_s - P_m}{\Delta P} \end{cases} \qquad \text{(Eq. 27)}$$

Where

(L_s,P_s) are the coordinates of the pixel in 1B segment (first pixel being (1,1)),

(L',P') are the centered and normalized coordinates of the pixel in 1B segment,

L_m is the mean value of lines L in 1B segment,

ΔL is the interval half-width for lines L in 1B segment,

P_m is the mean value of columns P in 1B segment,

ΔP is the interval half-width for columns P in 1B segment.

1B scene to / from 1B segment

This transform is a simple translation from the values found in ancillary data.

$$\begin{cases} L_s = L_0 + L \\ P_s = P_0 + P \end{cases} \qquad \text{(Eq. 28)}$$

Where

(L,P) are the coordinates of the pixel in 1B scene (first pixel being (1,1)),

(L_s,P_s) are the coordinates of the pixel in 1B segment (first pixel being (1,1)),

L_0 is the first line of the level 1B scene in the level 1B segment,

P_0 is the first pixel of the last line of the level 1B scene in the level 1B segment.

APPENDIX A - CARTESIAN GEOCENTRIC AND GEODETIC REFERENCE SYSTEMS

General equations

Coordinate transforms of this section are given in reference R-6: The following equations record some of the general properties relative to the ellipsoid (see fig. 31).

$$e = \frac{\sqrt{a^2 - b^2}}{a} \qquad \text{(Eq. 29a)}$$

$$N = \frac{a}{\sqrt{1 - e^2 \times \sin^2(\varphi)}} \qquad \text{(Eq. 29b)}$$

Where

a	is the semi-major axis
b	is the semi-minor axis
e	is the ellipsoid eccentricity
λ, φ	are the geodetic latitude and longitude respectively
N	is the ellipsoid normal (*grande normale* in French)

Programs – Rights to use

For each one of the two transforms, an example program written in C language is provided. These programs are free software; GAEL Consultant grants the rights to redistribute it and/or modify it under the terms as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

These programs are distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

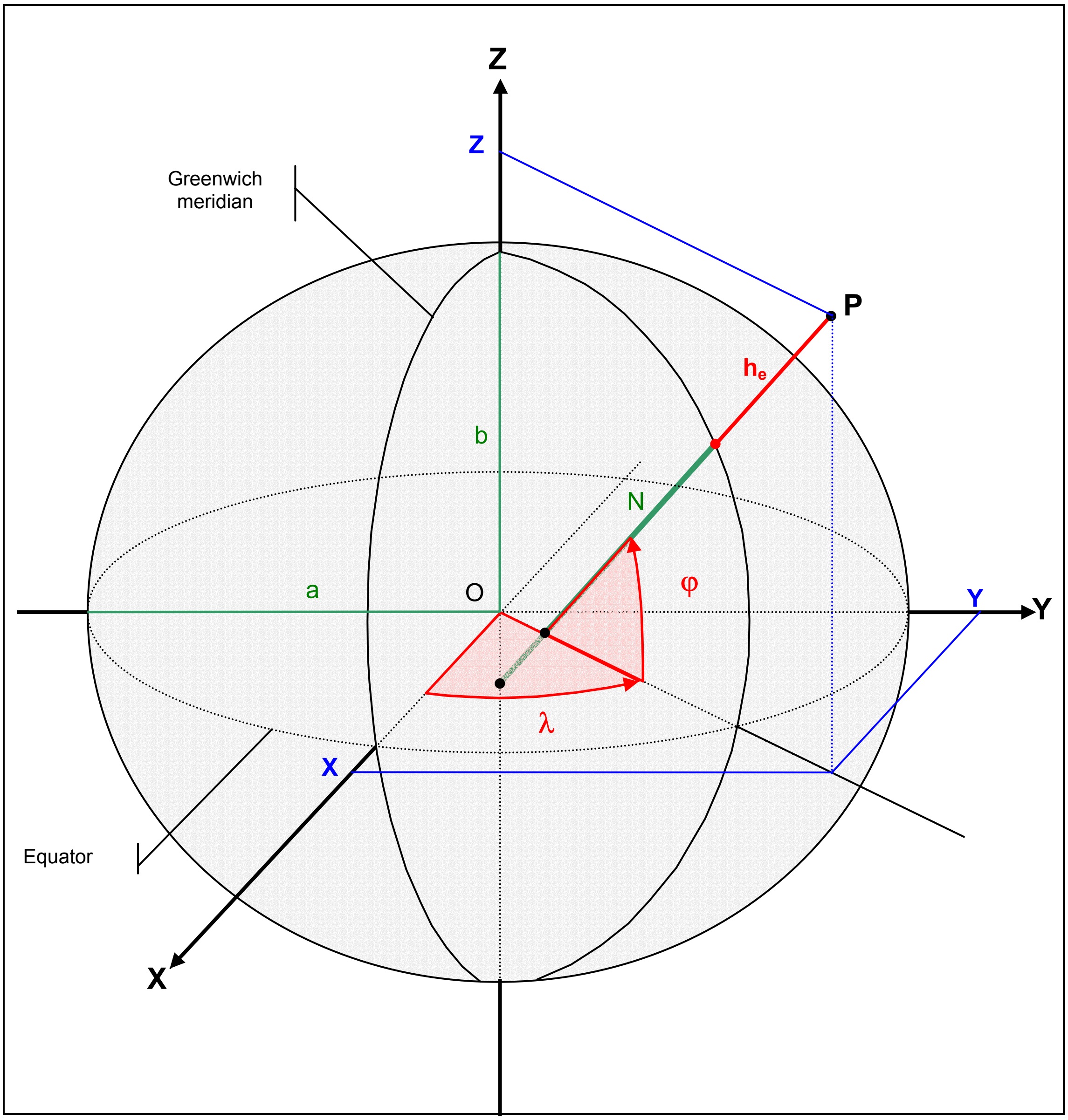


fig. 31 Cartesian geocentric and geodetic coordinates

A.1 From Cartesian Geocentric to Geodetic Reference System

Algorithm

Latitude computation is performed iteratively in the way depicted here below.

Initialization

$$\lambda \leftarrow \tan^{-1}\left(\frac{Y}{X}\right) \qquad \text{Eq. 30a}$$

$$\varphi_0 \leftarrow \tan^{-1}\left(\frac{Z}{\sqrt{X^2+Y^2}}\right) \qquad \text{Eq. 30b}$$

$$i \leftarrow 0$$

Compute the ellipsoid normal, the altitude and latitude

$$N_i \leftarrow \frac{a}{\sqrt{1-e^2\times\sin^2(\varphi_i)}} \qquad \text{Eq. 30c}$$

$$h_{e_i} \leftarrow \frac{\sqrt{X^2+Y^2}}{\cos(\varphi_i)} - N_i \qquad \text{Eq. 30d}$$

$$\varphi_{i+1} \leftarrow \tan^{-1}\left[\frac{Z}{\sqrt{X^2+Y^2}} \times \frac{1}{1-e^2\times\dfrac{N_i}{N_i+h_{e_i}}}\right] \qquad \text{Eq. 30e}$$

$((\,|\,\varphi_{i+1} - \varphi_i\,| < \varepsilon)$
or
$(i > \mathit{max\_iterations})$

no: $i \leftarrow i + 1$

yes

Exit

Program example

```
/******************************************************************************/
/* NAME                                                                       */
/* TimPrjGeocentricToGeo - computes geographic coordinates from XYZ geocentric*/
/* coordinates.                                                               */
/******************************************************************************/
/* SYNOPSIS                                                                   */
/* int TimPrjGeocentricToGeo(                                                 */
/*    TimRDouble          x,                                                  */
/*    TimRDouble          y,                                                  */
/*    TimRDouble          z,                                                  */
/*    TimPrjEllipsoidPtr  ellipsoid_p,                                        */
/*    TimRDouble          *latitude_p,                                        */
/*    TimRDouble          *longitude_p,                                       */
/*    TimRDouble          *altitude_p)                                        */
/******************************************************************************/
/* DESCRIPTION                                                                */
/* TimPrjGeocentricToGeo computes geodetic coordinates (latitude,             */
/* longitude, altitude) from XYZ geocentric coordinates for a given ellipsoid.*/
/*                                                                            */
/* <x>, <y>, <z> parameters are cartesian coordinates in geocentric system    */
/* expressed in metres.                                                       */
/*                                                                            */
/* <ellipsoid_p> is a pointer to an initialized TimPrjEllipsoid structure.    */
/* This can be done using TimPrjAllocEllipsoid function. This pointer should  */
/* not be NULL.                                                               */
/*                                                                            */
/* Output geodetic coordinates are pointed by <latitude_p>, <longitude_p>     */
/* and <altitude_p> parameters. <latitude_p> and <longitude_p> are expressed  */
/* in RADIAN and <altitude_p> in METRE. These pointers should not be NULL.    */
/*                                                                            */
/* Altitude is the height above the specified ellipsoid.                      */
/*                                                                            */
/* TimPrjGeocentricToGeo returns TIM_OK on success, and TIM_ERROR otherwise.  */
/******************************************************************************/
/* REFERENCE                                                                  */
/* [1] CHANGEMENT DE SYSTEME GEODESIQUE - ALGORITHMES                          */
/*     Note technique NT/G 80                                                 */
/*     IGN, janvier 1995                                                      */
/******************************************************************************/
/* SEE_ALSO                                                                   */
/* TimPrj manual                                                              */
/* TimPrj(5)                                                                  */
/* TimPrjGeoToGeocentric(3)                                                   */
/******************************************************************************/
/* COPYRIGHTS                                                                 */
/* See section "Programs - Rights to use"                                     */
/******************************************************************************/
/* ADMINISTRATION                                                             */
/* Stephane MBAYE   | 05.05.95 | v01.01 | Creation of the SW component        */
/* David LE CORFEC  | 05.03.98 | v02.01 | Added ANSI C support                */
/******************************************************************************/

/******************************************************************************/
/* ANSI C inclusion files                                                     */
/******************************************************************************/
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#ifndef M_PI
#include <values.h>
#endif

/******************************************************************************/
/* TELIMAGO inclusion files                                                   */
/******************************************************************************/
#define _TimNeedData
#define _TimNeedPrj
#include <Tim.h>
```

```
/***************************************************************************/
/* Constant definitions                                                    */
/***************************************************************************/
#define TIM_LATITUDE_COMPUTING_PRECISION 1.e-11 /* computing tolerance */
#define TIM_MAX_ITERATION_NUMBER         100    /* max iteration number */

/***************************************************************************/
/* TimGeoToUtm CORE                                                        */
/***************************************************************************/
int TimPrjGeocentricToGeo
#ifdef TIM_ANSI_PROTOTYPE
(
TimRDouble          x,                  /* X geocentric coordinate            */
TimRDouble          y,                  /* Y geocentric coordinate            */
TimRDouble          z,                  /* Z geocentric coordinate            */
TimPrjEllipsoidPtr  ellipsoid_p,        /* Ellipsoid structure pointer        */
TimRDouble          *latitude_p,        /* pointer to geographic latitude     */
TimRDouble          *longitude_p,       /* pointer to geographic longitude    */
TimRDouble          *altitude_p         /* altitude above ellipsoid           */
)
#else
(x,y,z,ellipsoid_p,latitude_p,longitude_p,altitude_p)
TimRDouble          x;
TimRDouble          y;
TimRDouble          z;
TimPrjEllipsoidPtr  ellipsoid_p;
TimRDouble          *latitude_p;
TimRDouble          *longitude_p;
TimRDouble          *altitude_p;
#endif
{
/***************************************************************************/
/* local variables                                                         */
/***************************************************************************/
   TimRDouble       XY_norme;           /* norme of geocentric vector on XY   */
   TimRDouble       XYZ_norme;          /* norme of geocentric vector         */
   TimRDouble       XY_radius;          /* projection of radius on XY plane   */
   TimRDouble       latitude;           /* current latitude in iterative comp.*/
   TimRDouble       new_latitude;       /* new latitude in iterative computing*/
   int              done;               /* "computing iteration is done" flag */
   int              iteration_count;    /* current iteration count            */

/***************************************************************************/
/* Check parameters                                                        */
/***************************************************************************/
   if ((latitude_p  == NULL)                                               ||
       (longitude_p == NULL)                                               ||
       (altitude_p  == NULL)                                               ||
       (ellipsoid_p == NULL))
   {
      fprintf (TimStderr,"TimPrjGeocentricToGeo : Wrong parametres\n");
      return (TIM_ERROR);
   }
/***************************************************************************/
/* Compute longitude                                                       */
/***************************************************************************/
   if (x == 0)
      *longitude_p = TIM_PI / 2;
   else
      *longitude_p = atan2 (y,x);
/***************************************************************************/
/* Compute latitude                                                        */
/***************************************************************************/
   XY_norme  = sqrt(x*x + y*y);
   XYZ_norme = sqrt(x*x + y*y + z*z);
   XY_radius = XY_norme * (1.0 - (ellipsoid_p->a *
                               ellipsoid_p->e2) /
                               XYZ_norme);
   if (TimDEqual(XY_radius,(TimRDouble)0.0,
       (TimRDouble)TIM_LATITUDE_COMPUTING_PRECISION))
      latitude = TIM_PI /2;
   else
```

```
      latitude = atan2 (z,XY_radius);

   iteration_count = 0;
   done            = TIM_FALSE;

   while ((!done) && (iteration_count < TIM_MAX_ITERATION_NUMBER))
   {
      new_latitude = atan ((z / XY_norme) * (1.0 /
                      (1.0 - (ellipsoid_p->a *
                              ellipsoid_p->e2 *
                              cos(latitude)) /
                      (XY_norme *
                       sqrt(1.0 - ellipsoid_p->e2 *
                            pow(sin(latitude),2.0))))));
      if (TimDEqual(new_latitude,latitude,
          (TimRDouble)TIM_LATITUDE_COMPUTING_PRECISION))
         done = TIM_TRUE;
      else
         latitude = new_latitude;

      iteration_count += 1;
   }
/******************************************************************************/
/* Report latitude value                                                      */
/******************************************************************************/
   *latitude_p = latitude;
/******************************************************************************/
/* Compute altitude                                                           */
/******************************************************************************/
   *altitude_p = sqrt(x*x + y*y) / cos(latitude)                    -
                 ellipsoid_p->a                /
                 sqrt(1.0 - ellipsoid_p->e2 *
                            pow(sin(latitude),2.0));
/******************************************************************************/
/* Return OK status                                                           */
/******************************************************************************/
   return (TIM_OK);
} /* TimPrjGeocentricToGeo */
```

A.2 From Geodetic to Cartesian Geocentric Reference System

Algorithm

Transformation from geodetic to cartesian reference systems is simply given by the following formula.

$$\begin{cases} X = (N + h_e) \times \cos(\varphi) \times \cos(\lambda) \\ Y = (N + h_e) \times \cos(\varphi) \times \sin(\lambda) \\ Z = \left[N \times (1 - e^2) + h_e\right] \times \sin(\varphi) \end{cases} \qquad \text{(Eq. 31)}$$

Where

X, Y, Z   are the cartesian coordinates

λ, φ   are the geodetic latitude and longitude respectively

h_e is the altitude above ellipsoid

e   is the ellipsoid eccentricity (see Eq. XX)

N   is the ellipsoid normal (*grande normale* in French) computed by Eq. 29b.

Program example

```
/*****************************************************************************/
/* NAME                                                                      */
/* TimPrjGeoToGeocentric - computes XYZ geocentric coordinates from geographic*/
/* coordinates.                                                              */
/*****************************************************************************/
/* SYNOPSIS                                                                  */
/* int TimPrjGeoToGeocentric(                                                */
/*    TimRDouble          latitude,                                          */
/*    TimRDouble          longitude,                                         */
/*    TimRDouble          altitude,                                          */
/*    TimPrjEllipsoidPtr  ellipsoid_p,                                       */
/*    TimRDouble          *x_p,                                              */
/*    TimRDouble          *y_p,                                              */
/*    TimRDouble          *z_p)                                              */
/*****************************************************************************/
/* DESCRIPTION                                                               */
/* TimPrjGeoToGeocentric computes XYZ cartesian coordinates in geocentric    */
/* system from geodetic (latitude, longitude) coordinates and for a given    */
/* altitude and ellipsoid.                                                   */
/*                                                                           */
/* Geodetic coordinates (<latitude> and <longitude>) in input are            */
/* expressed in radians and must be in range [-PI/2,+PI/2] and [-PI,+PI]     */
/*                                                                           */
/* Altitude (<altitude>) is the height of point above the ellipsoid.         */
/* Altitude is expressed in metre.                                           */
/*                                                                           */
/* <ellipsoid_p> is a pointer to an initialized TimPrjEllipsoid structure.   */
/* This action can be done using TimPrjAllocEllipsoid function. This pointer */
/* should not be NULL.                                                       */
/*                                                                           */
/* Cartesian coordinates to be computed are pointed by parameters <x_p>,     */
/* <y_p>, <z_p>. The output values are expressed in metres. These pointers   */
/* should not be NULL.                                                       */
/*                                                                           */
/* TimPrjGeoToGeocentric returns TIM_OK on success, and TIM_ERROR otherwise. */
/*****************************************************************************/
/* REFERENCE                                                                 */
/* [1] CHANGEMENT DE SYSTEME GEODESIQUE - ALGORITHMES                        */
/*     Note technique NT/G 80                                                */
/*     IGN, janvier 1995                                                     */
/*****************************************************************************/
/* SEE_ALSO                                                                  */
/* TimPrj manual                                                             */
/* TimPrj(5)                                                                 */
/* TimPrjGeocentricToGeo(3)                                                  */
/*****************************************************************************/
/* COPYRIGHTS                                                                */
/* See section "Programs - Rights to use"                                    */
/*****************************************************************************/
/* ADMINISTRATION                                                            */
/* Serge RIAZANOFF  | 20.05.95 | v01.01 | Creation of the SW component       */
/* Stephane MBAYE   | 05.03.98 | v01.02 | Allows large geographic coordinates */
/* David LE CORFEC  | 13.03.98 | v02.01 | Added ANSI C support               */
/*****************************************************************************/

/*****************************************************************************/
/* ANSI C inclusion files                                                    */
/*****************************************************************************/
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#ifndef M_PI
#include <values.h>
#endif

/*****************************************************************************/
/* TELIMAGO inclusion files                                                  */
/*****************************************************************************/
#define _TimNeedData
#define _TimNeedPrj
#include <Tim.h>
```

```
/******************************************************************************/
/* Constant definitions                                                       */
/******************************************************************************/

/******************************************************************************/
/* TimGeoToUtm CORE                                                           */
/******************************************************************************/
int TimPrjGeoToGeocentric
#ifdef TIM_ANSI_PROTOTYPE
(
TimRDouble          latitude,           /* geographic coordinate : latitude   */
TimRDouble          longitude,          /* geographic coordinate : longitude  */
TimRDouble          altitude,           /* geographic coordinate : altitude   */
TimPrjEllipsoidPtr  ellipsoid_p,        /* pointer to Ellipsoid structure     */
TimRDouble          *x_p,               /* X geocentric coordinate to compute */
TimRDouble          *y_p,               /* Y geocentric coordinate to compute */
TimRDouble          *z_p                /* Z geocentric coordinate to compute */
)
#else
(latitude,longitude,altitude,ellipsoid_p,x_p,y_p,z_p)
TimRDouble          latitude;
TimRDouble          longitude;
TimRDouble          altitude;
TimPrjEllipsoidPtr  ellipsoid_p;
TimRDouble          *x_p;
TimRDouble          *y_p;
TimRDouble          *z_p;
#endif
{
/******************************************************************************/
/* local variables                                                            */
/******************************************************************************/
   TimRDouble       normal;            /* large normal to ellipsoid           */

/******************************************************************************/
/* Check parameters                                                           */
/******************************************************************************/
   if ((ellipsoid_p == NULL)                                                  ||
       (x_p         == NULL)                                                  ||
       (y_p         == NULL)                                                  ||
       (z_p         == NULL))
   {
      fprintf (TimStderr,"TimPrjGeoToGeocentric : Wrong parametres\n");
      return (TIM_ERROR);
   }
/******************************************************************************/
/* Compute large normal to ellipsoid                                          */
/******************************************************************************/
   normal = ellipsoid_p->a                          /
      sqrt(1.0 - ellipsoid_p->e2 * pow(sin(latitude),2.0));
/******************************************************************************/
/* Compute cartesian values                                                   */
/******************************************************************************/
   *x_p = (normal + altitude) * cos(latitude) * cos(longitude);
   *y_p = (normal + altitude) * cos(latitude) * sin(longitude);
   *z_p = (normal * (1.0 - ellipsoid_p->e2) + altitude) *
           sin(latitude);
/******************************************************************************/
/* Return OK status                                                           */
/******************************************************************************/
   return (TIM_OK);
} /* TimPrjGeoToGeocentric */
```

APPENDIX B - COMPUTATION OF INCIDENCE ANGLE

Considering the spherical representation of the Earth (see fig. 32), incidence angle on any point of the Earth is given by the following formula.

$$\beta = -\sin^{-1}\left[\frac{R_T + h}{R_T} \times \sin(\alpha)\right] \qquad \text{(Eq. 32)}$$

Where

β is the incidence angle

α is the look direction from the satellite

R_T is the mean value of WGS84 ellipsoid semi-axes (R_T = 6 371 008.7714 m)

h is the altitude of satellite above ellipsoid (nominal value: h = 832 000 m)

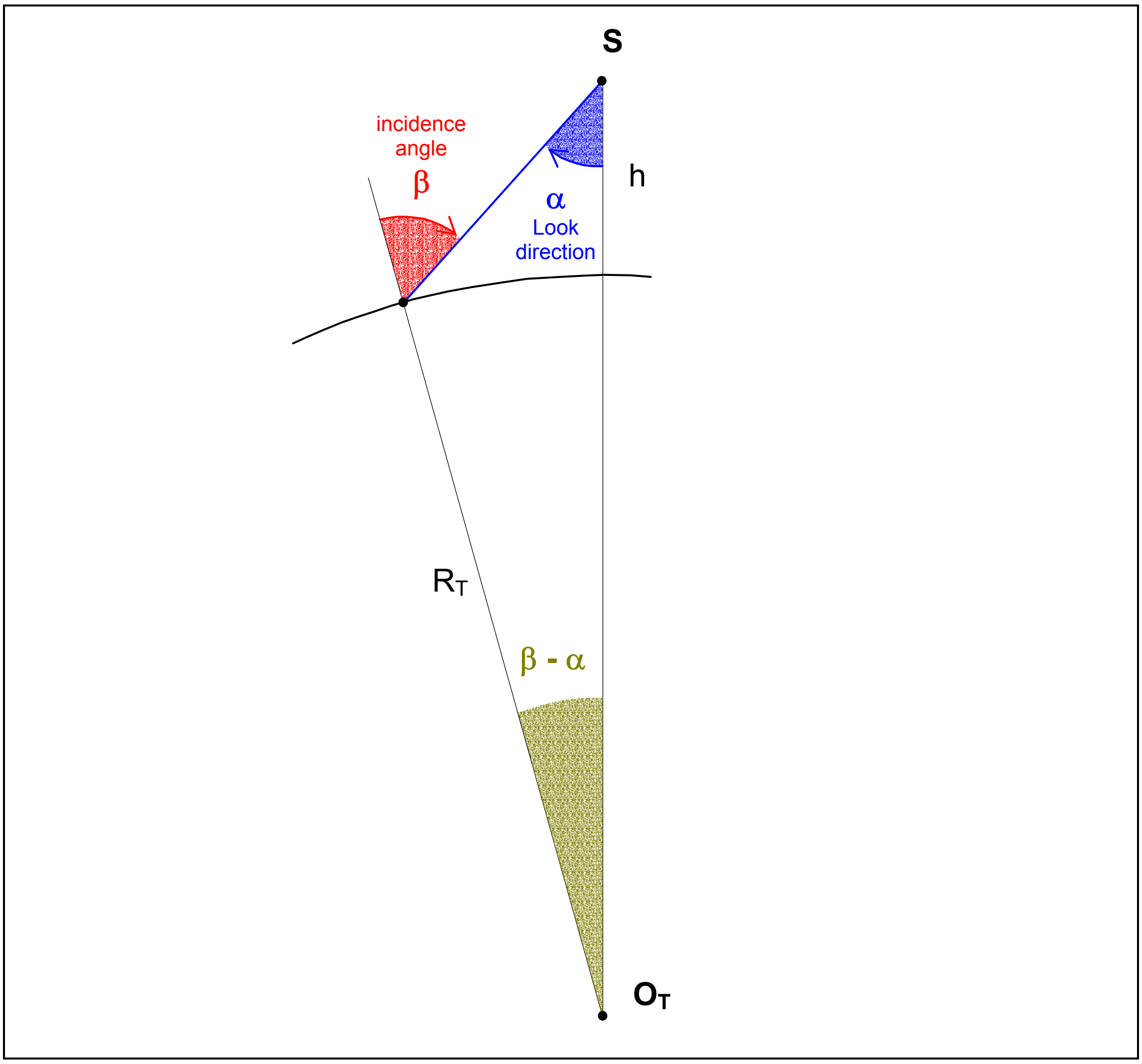


fig. 32 Look direction and incidence angle

APPENDIX C - LOCATION OF VALUES IN SPOT PRODUCTS

C.1 Introduction

An historical background and a short description of the SPOT data format are provided in section 3.2.

This Appendix gives the location of all the ancillary data involved within the formulae with reference to the two more recent formats: DIMAP and CAP.

Location in CAP format

Like any other CEOS format, any field in CAP format is completely located providing the:

1. name of FILE,
2. name of Record, and
3. byte range *#start-stop* within the record (first byte being 1).

Location in DIMAP format

Location of the field in DIMAP format is provided giving the full XML path (see XPath in R-7).

C.2 Location of values

Symbol	Description		
	FILE (CAP format)	Record	*#start-stop*
	DIMAP location of the field (XPath syntax with hyperlink to,the dictionary page)		
aec	*Average encoder count*		
	Not present		
	Dimap_Document/Data_Strip/Sensor_Configuration/Mirror_Position/AVERAGE_ENCODER_COUNT		
(a_i)	*Five (5) coefficients of polynomial A*		
	LEADER	MODELIZATION	#497-512..#561-576
	Dimap_Document/Data_Strip/Models/OneB_Model/A/c		
$a_{oor}(t_i)$	*Attitude value is "OUT_OF-RANGE" indicator*		
	LEADER	EPHEMERIS/ATTITUDE	#959-960
	Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles/OUT_OF_RANGE		
$(a_p)_1$	*Rotation angle around the pitch axis at the beginning of the scene*		
	LEADER	EPHEMERIS/ATTITUDE	#3025-3032
	Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles[1]/PITCH		

$a_p'(t_i)$	*Average rotation speed around the pitch axis (attitude sample i)* LEADER EPHEMERIS/ATTITUDE #1015-1019,…, #2455-2459 Dimap_Document/Data_Strip/Satellite_Attitudes/Raw_Attitudes/Aocs_Attitude/Angular_Speeds_List/Angular_Speeds/PITCH
$a_p(t_i)$	*Rotation angle around the pitch axis (attitude sample i)* *Not present* Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles/PITCH
$(a_r)_1$	*Rotation angle around the roll axis at the beginning of the scene* LEADER EPHEMERIS/ATTITUDE #3017-3024 Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles[1]/ROLL
$a_r(t_i)$	*Rotation angle around the roll axis (attitude sample i)* *Not present* Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles/ROLL
$a_r'(t_i)$	*Average rotation speed around the roll axis (attitude sample i)* LEADER EPHEMERIS/ATTITUDE #1010-1014,…, #2450-2454 Dimap_Document/Data_Strip/Satellite_Attitudes/Raw_Attitudes/Aocs_Attitude/Angular_Speeds_List/Angular_Speeds/ROLL
$(a_y)_1$	*Rotation angle around the yaw axis at the beginning of the scene* LEADER EPHEMERIS/ATTITUDE #3009-3016 Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles[1]/YAW
$a_y(t_i)$	*Rotation angle around the yaw axis (attitude sample i)* *Not present* Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles/YAW
$a_y'(t_i)$	*Average rotation speed around the yaw axis (attitude sample i)* LEADER EPHEMERIS/ATTITUDE #1005-1009,…, #2445-2449 Dimap_Document/Data_Strip/Satellite_Attitudes/Raw_Attitudes/Aocs_Attitude/Angular_Speeds_List/Angular_Speeds/YAW
(b_i)	*Six (6) coefficients of polynomial B* LEADER MODELIZATION #577-592..#657-672 Dimap_Document/Data_Strip/Models/OneB_Model/B/c
c_0	*Value of on-board clock at the date t_0* LEADER EPHEMERIS/ATTITUDE #3397-3408 Dimap_Document/Data_Strip/Satellite_Time/CLOCK_VALUE

c(l)	*On-board clock value of the image line l*			
	IMAGERY	IMAGE DATA	#21-24	
	Not present			
(i_i)	*Six (6) coefficients of polynomial I*			
	LEADER	MODELIZATION	#337-352..#417-432	
	Dimap_Document/Data_Strip/Models/OneB_Model/Direct_OneB_Model/I/c			
(j_i)	*Four (4) coefficients of polynomial J*			
	LEADER	MODELIZATION	#433-448..#481-496	
	Dimap_Document/Data_Strip/Models/OneB_Model/Direct_OneB_Model/J/c			
(l_i)	*Six (6) coefficients of polynomial L*			
	LEADER	MODELIZATION	#689-704..#769-784	
	Dimap_Document/Data_Strip/Models/OneB_Model/Reverse_OneB_Model/L/c			
l_0	*First line of the raw scene in the raw segment*			
	LEADER	MODELIZATION	#25-32	
	Dimap_Document/Data_Strip/Data_Strip_Coordinates/FIRST_LINE_RAW			
L_0	*First line of the level 1B scene in the level 1B segment*			
	LEADER	MODELIZATION	#41-48	
	Dimap_Document/Data_Strip/Data_Strip_Coordinates/FIRST_LINE_1B			
l_1	*Line of absolute attitude* $[(a_p)_1, (a_r)_1, (a_y)_1]$ *at the beginning of the scene*			
	LEADER	EPHEMERIS/ATTITUDE	#3001-3008	
	Not present			
l_c	*Line containing the scene center*			
	LEADER	HEADER	#117-132	
	Dimap_Document/Data_Strip/Sensor_Configuration/Time_Stamp/SCENE_CENTER_LINE			
l_i	*Line of average rotation speed* $[a_p'(t_i), a_r'(t_i), a_y'(t_i)]$			
	LEADER	EPHEMERIS/ATTITUDE	#1001-1004,…, #2441-2445	
	Not present			
l_m	*Mean value of lines l in 1A segment*			
	LEADER	MODELIZATION	#257-272	
	Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/LINES_L_M			
L_m	*Mean value of lines L in 1B segment*			
	LEADER	MODELIZATION	#193-208	
	Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/LINES_l_M			
lsp	*Line sampling period*			
	LEADER	EPHEMERIS/ATTITUDE	#947-958	
	Dimap_Document/Data_Strip/Sensor_Configuration/Time_Stamp/LINE_PERIOD			

M_{MCV}	*Look direction change matrix (Matrice de changement de visée)* *Not present* Dimap_Document/Data_Strip/Sensor_Configuration/Mirror_Position/MCV_Matrix/MCV_Matrix_Row_List
(p_i)	*Four (4) coefficients of polynomial P* LEADER MODELIZATION #785-800..#833-848 Dimap_Document/Data_Strip/Models/OneB_Model/Reverse_OneB_Model/P/c
$P(t_i)$	*Satellite position coordinates (ephemeris sample i)* LEADER EPHEMERIS/ATTITUDE #21-56, #121-156, …, #821-856 Dimap_Document/Data_Strip/Ephemeris/Points/Point/Location
pms	*Pointing mirror step (pms=3..93)* LEADER HEADER #677-692 Dimap_Document/Data_Strip/Sensor_Configuration/Mirror_Position/STEP_COUNT
p_0	*First pixel of the raw scene in the raw segment (p_0=1)* LEADER MODELIZATION #25-32 Dimap_Document/Data_Strip/Data_Strip_Coordinates/FIRST_PIXEL_RAW
P_0	*First pixel of the last line of the level 1B scene in the level 1B segment* LEADER MODELIZATION #33-40 Dimap_Document/Data_Strip/Data_Strip_Coordinates/FIRST_PIXEL_1B
p_m	*Mean value of columns p in 1A segment* LEADER MODELIZATION #289-304 Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/COLUMNS_P_M
P_m	*Mean value of columns P in 1B segment* LEADER MODELIZATION #225-240 Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/COLUMNS_J_M
t_0	*UTC reference date* LEADER EPHEMERIS/ATTITUDE #3385-3396 Dimap_Document/Data_Strip/Satellite_Time/UT_DATE
t_c	*Scene center date* LEADER HEADER #581-612 Dimap_Document/Data_Strip/Sensor_Configuration/Time_Stamp/SCENE_CENTER_TIME
t_i	*Universal times cooresponding to the positions $P(t_i)$ and velocities $V(t_i)$* LEADER EPHEMERIS/ATTITUDE #93-120, #193-220, …, #893-920 Dimap_Document/Data_Strip/Ephemeris/Points/Point/TIME
t_i	*Universal times cooresponding to the attitudes $[a_p(t_i), a_r(t_i), a_y(t_i)]$* *Not present* Dimap_Document/Data_Strip/Satellite_Attitudes/Corrected_Attitudes/Corrected_Attitude/Angles/TIME

t_i	*Universal times cooresponding to the attitudes [$a_p'(t_i)$, $a_r'(t_i)$, $a_y'(t_i)$]* *Not present* Dimap_Document/Data_Strip/Satellite_Attitudes/Raw_Attitudes/Aocs_Attitude/Angular_Speeds_List/Angular_Speeds/TIME		
$V(t_i)$	*Satellite velocity coordinates (ephemeris sample i)* LEADER Dimap_Document/Data_Strip/Ephemeris/Points/Point/Velocity	EPHEMERIS/ATTITUDE	#57-92, #157-192, …, #857-892
Δl	*Interval half-width for lines l in 1A segment* LEADER Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/LINES_DELTA_L	MODELIZATION	#273-288
ΔL	*Interval half-width for lines L in 1B segment* LEADER Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/LINES_DELTA_I	MODELIZATION	#209-224
Δp	*Interval half-width for columns p in 1A segment* LEADER Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/COLUMNS_DELTA_P	MODELIZATION	#305-320
ΔP	*Interval half-width for columns P in 1B segment* LEADER Dimap_Document/Data_Strip/Models/OneB_Model/Coordinate_Normalization/COLUMNS_DELTA_J	MODELIZATION	#241-256
ω	*On-board clock period* LEADER Dimap_Document/Data_Strip/Satellite_Time/CLOCK_PERIOD	EPHEMERIS/ATTITUDE	#3409-3420
$(\psi_X)_1$	Along-track look angle for the first pixel LEADER Dimap_Document/Data_Strip/Sensor_Configuration/Instrument_Look_Angles_List/Instrument_Look_Angles/Look_Angles_List/Look_Angles[1]/PSI_X	EPHEMERIS/ATTITUDE	#3077-3088
$(\psi_X)_p$	Along-track look angle for the pixel number p (p=1..N) *Not present* Dimap_Document/Data_Strip/Sensor_Configuration/Instrument_Look_Angles_List/Instrument_Look_Angles/Look_Angles_List/Look_Angles/PSI_X		
$(\psi_X)_N$	Along-track look angle for the last pixel (N=3000 or N=6000) LEADER Dimap_Document/Data_Strip/Sensor_Configuration/Instrument_Look_Angles_List/Instrument_Look_Angles/Look_Angles_List/Look_Angles[N]/PSI_X	EPHEMERIS/ATTITUDE	#3065-3076

$(\psi_Y)_1$	Across-track look angle for the first pixel LEADER EPHEMERIS/ATTITUDE #3101-3112 Dimap_Document/Data_Strip/Sensor_Configuration/Instrument_Look_Angles_List/Instrument_Look_Angles/Look_Angles_List/Look_Angles[1]/PSI_Y
$(\psi_Y)_p$	Across-track look angle for the pixel number p (p=1..N) *Not present* Dimap_Document/Data_Strip/Sensor_Configuration/Instrument_Look_Angles_List/Instrument_Look_Angles/Look_Angles_List/Look_Angles/PSI_Y
$(\psi_Y)_N$	Across-track look angle for the last pixel (N=3000 or N=6000) LEADER EPHEMERIS/ATTITUDE #3089-3100 Dimap_Document/Data_Strip/Sensor_Configuration/Instrument_Look_Angles_List/Instrument_Look_Angles/Look_Angles_List/Look_Angles[N]/PSI_Y

APPENDIX D - EXAMPLE OF SPOT AUXILIARY DATA

Scope of this appendix is to provide a concrete example of numerical values found in a real SPOT product to illustrate the range of expected values and the expected geometric modelization.

D.1 SPOT4 – 126-265/2 – 29/09/1998 – Ancillary data

Some of the main information retrieved within the product are given in the table here below. Ephemeris and attitude data are presented in the next sections.

satellite	"SPOT4"
instrument	"HRVIR2"
observation date	29/09/1998
observation time	08:00:17.105469
path	126
row	265.2
orbit	112
product Id	"S4H2980929080015"
line number	3000
pixel number	3000
altitude	830640 m
viewing angle	**-20.4 °**
sun elevation	43.4°
sun azimuth	158°
look direction – first CCD	$(\psi_X)_1$ = 00°00'33.78" $(\psi_Y)_1$ = -22°18'50.69"
look direction – last CCD	$(\psi_X)_{3000}$ = 00°00'27.79" $(\psi_Y)_{3000}$ = -18°11'12.39"
spot product level	1A
spot spectral mode	XS
level1AsceneLineShift	9092
level1AscenePixelShift	1
level1BsceneLineShift	9066
level1BscenePixelShift	263
level1BpolynomA	5 1.00000000000000000e+00 4.10099833970889449e-04 -3.67122993338853121e-06 -2.67009994558975450e-07 4.79999984204226848e-10
level1BpolynomB	6 0.00000000000000000e+00 -1.98755919933319092e-01 -2.55642551928758621e-03 2.17189608520129696e-05 2.19140005697227025e-07 -1.99999999200839440e-11
level1BpolynomL	6 6.04648485023062676e-05 1.00000000000000000e+00 -6.04766610194928944e-05 -1.17000000798839210e-08 1.18000000881579581e-08 -3.60000002030957944e-10
level1BpolynomP	4 -1.91947389394044876e-02 1.24773263931274414e+00 2.57239025086164474e-02 2.91100121103227139e-04
line1BMean	8.57831542968750000e+03
line1BHalfInterval	8.57731542968750000e+03
line1AMean	8.60250000000000000e+03
line1AHalfInterval	8.60150000000000000e+03
pixel1BMean	2.18587011718750000e+03
pixel1BHalfInterval	2.18487011718750000e+03
pixel1AMean	1.50050000000000000e+03
pixel1AHalfInterval	1.49950000000000000e+03

D.2 SPOT4 – 126-265/2 – 29/09/1998 – Ephemeris

The eight (8) values of the ephemeris (position and velocity) found in the product are given within the table here below. As shown in fig. 33and also checking the dates of these ephemeris, none of these 8 values has been measured within the range of image acquisition. As a consequence, only eight values (and not nine) are provided.

position (m)			velocity (m/s)				
X	Y	Z	X'	Y'	Z'	date	time
2.792669e+06	3.337747e+06	5.733024e+06	5.117309e+03	3.205796e+03	-4.349242e+03	29/09/1998	07:57:00
3.109538e+06	3.509992e+06	5.461204e+06	4.942003e+03	2.964689e+03	-4.708289e+03	29/09/1998	07:58:00
3.415970e+06	3.666109e+06	5.168379e+06	4.745643e+03	2.713851e+03	-5.049282e+03	29/09/1998	07:59:00
3.710650e+06	3.805595e+06	4.855673e+06	4.528894e+03	2.454420e+03	-5.370903e+03	29/09/1998	08:00:00
3.992304e+06	3.928026e+06	4.524285e+06	4.292508e+03	2.187560e+03	-5.671904e+03	29/09/1998	08:01:00
4.259704e+06	4.033051e+06	4.175488e+06	4.037328e+03	1.914460e+03	-5.951117e+03	29/09/1998	08:02:00
4.511674e+06	4.120398e+06	3.810622e+06	3.764280e+03	1.636326e+03	-6.207455e+03	29/09/1998	08:03:00
4.747094e+06	4.189870e+06	3.431089e+06	3.474369e+03	1.354376e+03	-6.439923e+03	29/09/1998	08:04:00

fig. 33displays the SPOT image, a piece of its trajectory overlaid on a vectorial chart of the World in orthographic projection. Image has been previously geocoded and matches the border of the Caspian Sea. The eight position points have been projected on the ground to coincide with the nadir of the satellite.

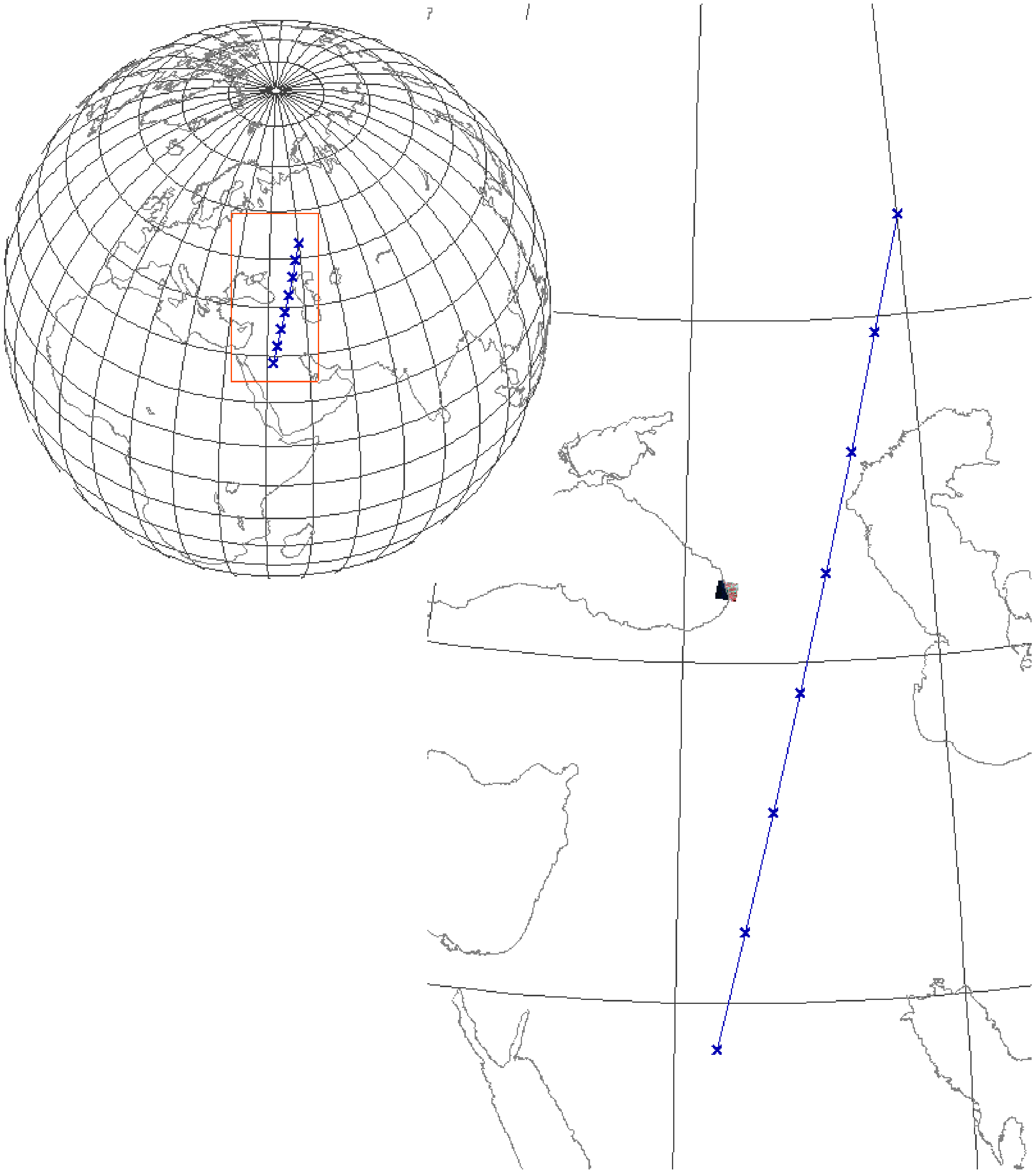


fig. 33 SPOT4 – 126-265/2 – 29/09/1998 - Location of the scene and satellite position

D.3 SPOT4 – 126-265/2 – 29/09/1998 – Attitudes

Table here below contains the 73 attitude data.

- first column provides the line number found in ancillary data,
- "yaw", "roll" and "pitch" columns contain the average rotation speeds found in ancillary data and expressed in 10^{-6} degrees per second,
- columns "date" and "time" have been computed from the line number using equation 1 ("Line dating"),

Average rotation speeds

Yaw, roll and pitch average rotation speeds are shown in fig. 34. Yaw and roll values are in the range [-1.10^{-4} deg/s, + 1.10^{-4} deg/s], while pitch values are in the range [-3.10^{-4} deg/s, + 3.10^{-4} deg/s].

Absolute rotation angles

Initial attitude values found in ancillary data are:

line	yaw	roll	pitch
0001	+0037	-0311	-0046

Starting from these values given in 10^{-6} degrees and integrating the average rotation speeds of the 72 samples (all the 73 attitudes of the table here below except the first one) leads to attitude profile shown in figure fig. 35.

We may note the magnitude of pitch angle in range [0°, 4.10^{-3}°] that leads to displacements on Earth until 58 metres.

Attitude list 73 samples

line	yaw	roll	pitch	date	time
0001	+0000	+0000	+0040	29/09/1998	08:00:12.599609
0042	+0000	+0020	-0020	29/09/1998	08:00:12.722656
0084	+0000	+0000	+0000	29/09/1998	08:00:12.849609
0125	+0000	+0060	+0040	29/09/1998	08:00:12.972656
0167	-0080	+0000	+0020	29/09/1998	08:00:13.099609
0208	+0060	+0000	+0000	29/09/1998	08:00:13.222656
0250	+0000	-0020	-0100	29/09/1998	08:00:13.347656
0292	+0140	-0100	-0120	29/09/1998	08:00:13.474609
0333	+0020	-0040	-0080	29/09/1998	08:00:13.597656
0375	+0100	-0100	-0080	29/09/1998	08:00:13.724609
0417	+0100	+0020	-0060	29/09/1998	08:00:13.851562
0458	+0100	-0080	-0120	29/09/1998	08:00:13.974609
0500	+0000	+0000	-0040	29/09/1998	08:00:14.099609
0541	+0000	+0000	-0040	29/09/1998	08:00:14.224609
0583	-0040	+0060	+0020	29/09/1998	08:00:14.349609
0625	-0020	+0000	+0000	29/09/1998	08:00:14.476562
0666	-0060	+0060	+0020	29/09/1998	08:00:14.599609
0708	-0080	+0060	+0000	29/09/1998	08:00:14.726562
0749	-0100	+0100	+0080	29/09/1998	08:00:14.849609
0790	-0120	+0120	+0080	29/09/1998	08:00:14.972656
0832	-0040	+0040	+0040	29/09/1998	08:00:15.099609
0873	-0060	+0080	+0040	29/09/1998	08:00:15.222656
0915	-0020	+0000	+0000	29/09/1998	08:00:15.349609
0957	+0000	+0000	+0080	29/09/1998	08:00:15.474609
0998	+0060	+0000	+0000	29/09/1998	08:00:15.597656
1040	+0020	-0020	-0080	29/09/1998	08:00:15.724609
1082	+0140	-0060	+0000	29/09/1998	08:00:15.851562
1123	+0060	-0140	+0040	29/09/1998	08:00:15.974609
1165	+0080	-0060	+0020	29/09/1998	08:00:16.101562
1206	+0080	-0060	+0000	29/09/1998	08:00:16.224609
1248	+0000	-0060	+0000	29/09/1998	08:00:16.349609
1290	+0000	+0000	+0040	29/09/1998	08:00:16.476562
1331	+0080	-0040	-0060	29/09/1998	08:00:16.599609
1373	+0040	-0040	-0040	29/09/1998	08:00:16.726562
1414	+0000	+0000	-0040	29/09/1998	08:00:16.849609
1455	-0040	+0000	-0020	29/09/1998	08:00:16.972656
1497	-0040	-0040	-0100	29/09/1998	08:00:17.099609
1538	-0060	+0000	-0060	29/09/1998	08:00:17.222656
1580	-0040	+0000	-0100	29/09/1998	08:00:17.349609
1622	+0000	+0000	-0180	29/09/1998	08:00:17.474609
1663	+0000	+0080	-0120	29/09/1998	08:00:17.599609
1705	+0060	-0040	-0260	29/09/1998	08:00:17.724609
1747	+0040	-0020	-0220	29/09/1998	08:00:17.851562
1788	+0080	-0080	-0220	29/09/1998	08:00:17.974609
1830	+0000	-0080	-0260	29/09/1998	08:00:18.101562
1871	+0060	-0060	-0260	29/09/1998	08:00:18.224609
1913	+0000	-0100	-0200	29/09/1998	08:00:18.351562
1955	+0040	-0020	-0180	29/09/1998	08:00:18.476562
1996	-0060	+0040	-0240	29/09/1998	08:00:18.599609
2038	+0020	+0000	-0240	29/09/1998	08:00:18.726562
2079	-0060	+0020	-0160	29/09/1998	08:00:18.849609
2120	-0060	+0060	-0100	29/09/1998	08:00:18.972656
2162	-0080	+0040	-0140	29/09/1998	08:00:19.099609
2203	-0100	+0020	-0120	29/09/1998	08:00:19.222656
2245	-0060	+0080	-0060	29/09/1998	08:00:19.349609
2287	-0140	+0000	-0080	29/09/1998	08:00:19.476562
2328	-0060	+0140	-0020	29/09/1998	08:00:19.599609
2370	-0120	+0060	+0000	29/09/1998	08:00:19.724609
2412	-0040	+0040	+0040	29/09/1998	08:00:19.851562
2453	-0020	+0020	+0000	29/09/1998	08:00:19.974609
2495	-0040	+0000	+0100	29/09/1998	08:00:20.101562
2536	+0000	+0000	+0040	29/09/1998	08:00:20.224609
2578	-0040	+0000	+0140	29/09/1998	08:00:20.351562
2620	-0020	+0000	+0180	29/09/1998	08:00:20.476562
2661	-0020	-0020	+0120	29/09/1998	08:00:20.601562
2703	+0000	+0000	+0220	29/09/1998	08:00:20.726562
2744	-0060	+0060	+0320	29/09/1998	08:00:20.849609
2785	+0040	+0000	+0280	29/09/1998	08:00:20.974609
2827	-0040	-0020	+0240	29/09/1998	08:00:21.099609
2868	+0000	-0060	+0260	29/09/1998	08:00:21.222656
2910	+0020	+0020	+0240	29/09/1998	08:00:21.349609
2952	+0040	-0100	+0220	29/09/1998	08:00:21.476562
2993	+0020	+0020	+0220	29/09/1998	08:00:21.599609

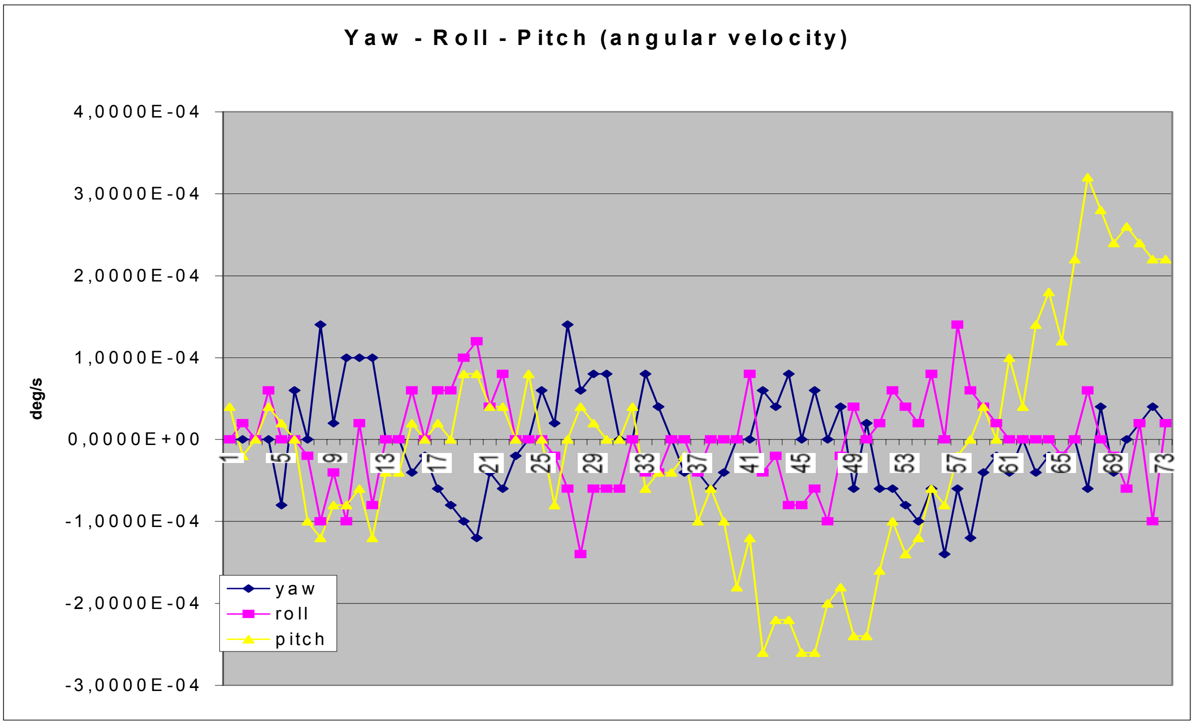


fig. 34 SPOT4 – 126-265/2 – 29/09/1998 – Angular velocities of attitude angles

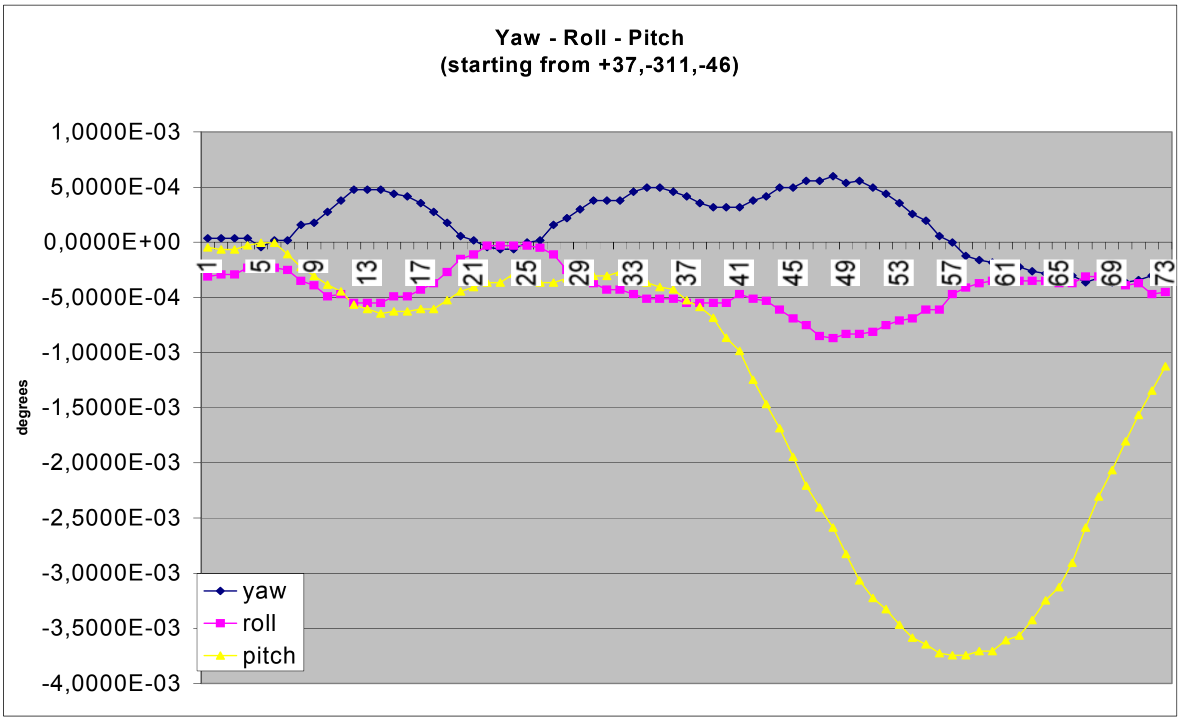


fig. 35 SPOT4 – 126-265/2 – 29/09/1998 – Attitude angles after integration