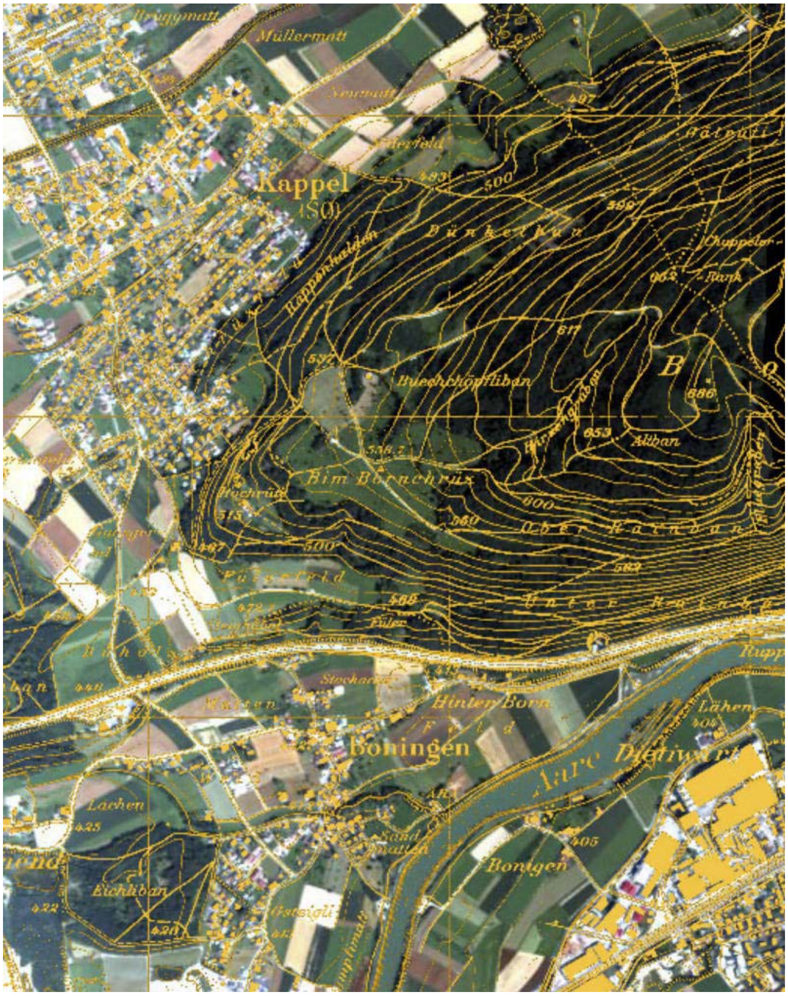
PARAMETRIC GEOCODING

Orthorectification for Airborne Scanner Data

User Manual, Version 3.3



ReSe Applications Schläpfer

PARGE User Manual, Version 3.3

Software and manual are made in Switzerland.

PARGE software produced and maintained by ReSe Applications Schläpfer. Original development was supported by the Remote Sensing Laboratories (RSL) of the University of Zurich. Year of the first publication of manual and software: 2000.

PARGE user manual and PARGE software authored by Daniel Schläpfer, Dr. sc. nat.

Front cover:
Orthorectified HYMAP imaging spectroscopy data, acquired in 2004 in Switzerland. Overlay with digital topographic map. Processing done by PARGE; data courtesy of RSL, University of Zurich.

Table of Contents

Chapter 3:

Typical Workflow

Chapter 4:

PARGE Programming Interface

Chapter 1:

Introduction

Airborne imaging spectrometers and optical scanner systems are of increasing importance for operational remote sensing applications. The geometric stability of such systems is always worse than for spaceborne platforms. Geometric distortions occur due to variations of the flightpath as well as of the attitude (given by roll, pitch and heading angles) of the plane. Even mechanically stabilizing platforms built into the carrier can not remove these effects completely as residual movements of the stabilizing systems remain.

These distortions become relevant if pixel- or even sub-pixel accuracy is required and cannot be corrected by polynomial rectification. Such ground control point based traditional georeferencing procedures are inadequate to this problem since the movements cannot be approximated satisfactorily by polynomial transformations of the image. For non-stereographic imagery (as acquired by most of the current imaging spectrometers), a physically exact reconstruction of the scan geometry can only be obtained by a direct georeferencing, parametric geocoding approach as implemented in the PARGE software. A pixelwise calculation has to be performed to account for the position and attitude of the scanning instrument during the scanning process with respect to the terrain.

1.1 Goals of PARGE

With the PARGE software, a complete georeferencing and orthorectification procedure has been implemented based on a parametric approach, allowing for sub-pixel accuracy even in steep terrain. PARGE bases on unique algorithms having specific benefits for parametric geocoding of airborne scanner data preserving a maximum of the radiometric quality. It precisely reconstructs the scanning geometry for each image pixel using position, attitude, and terrain elevation data. An important advantage of PARGE is the provision of an environment for synchronization and recalibration of auxiliary data. Ground control points based procedures have therefore been included to recalibrate the boresight offsets of attitude and to correct offsets in the navigation data.

The whole application is kept as flexible as possible. Its consistent internal data format contains image, sensor, and DEM description as well as all sets of the auxiliary data and processing options. The data structure helps documenting and repeating the processing steps with the capability to store intermediate status reports. Some minor viewing and analysis capabilities for the DEM and the image data were introduced too, but are not considered a key part of the package. The internal data format of the algorithms is constructed in a way that it can be used for all common sensors – pushbroom or whiskbroom. The procedure has been tested on AVIRIS, DAIS, ROSIS, MIVIS, HyMap, Probe-1, CASI, AISA, SAMSON, HYPER, and HYSPEX, APEX, and Headwall Hyperspec imaging spectrometry data. The geocoding results are of reliable accuracies of down to 0.2 pixels in comparison to digital topographic data. The quality of the procedure was found to be within the accuracy of the input data and as the input data quality increased recently, pixel-accurate processing results have become a standard.

1.2 Functionality

The PARGE application is a flexible system for direct parametric orthorectification of airborne scanner data. The following features are included:

a) Compatibility

- consistent and open data structure for any airborne imaging instruments,
- support for whiskbroom, pushbroom, and any non-standard or tilted sensor model,
- import filters for standard hyperspectral file formats and supported sensor systems,
- import and synchronization capabilities for attitude and GPS navigation data,
- DEM standard import, resizing, and resampling options,
- Support for global SRTM DEM data import
- fully IDL (Interactive Data Language, Harris Inc.) based and therefore operating system independent application,
- compatible to the free IDL virtual machine, and
- outputs to standard ENVI™ (Exelis Inc. [20]) file formats.

b) Auxiliary data functions

- consideration of the attitude and navigation data by image line or by pixel,
- coordinate conversion (includes standard conversion to common coordinate systems and local datums),
- exact correction of roll, pitch, and true heading variations,
- ground control point (GCP) based algorithms for auxiliary data offsets estimation and boresight calibration,
- options for flightpath approximation, roll/pitch interpolation, or drift correction from GCPs,
- options for roll compensation, and roll-based post-synchronization, and
- automatic GCP detection based on vicinity image matching.

c) Processing options

- orthorectification algorithm for flat or rugged terrain,
- output to initial DEM geometry or to arbitrary resolution, rotation, and extent.
- optional output for data acquisition geometry based radiometric processing,
- inverse transformation capability (for DEM transformation to scan geometry),
- variety of resampling options for cartographic cube generation (selection of nearest neighbour, bilinear interpolation, or compromise solutions),
- batch processing support on the basis of an open data format and scripting environment,
- external text file based call batch processing option
- optional scan angle outputs of sensor zenith, sensor azimuth, and pixel distance for BRDF processing and atmospheric correction,
- ENVI image geometry map (IGM) output,
- RGB/grayscale visualization and image creation (TIFF/JPEG/ENVI), and
- Mosaicking and spectral merging of results.

d) Supporting abilities

- comprehensible workbench interface,
- ENVI file analysis tools,
- history tracking and optional job logging,
- processing status management with the ability to store, restore and view the current progress of work,
- on-line help system and and hyper-linked PDF manual,
- web-based support and upgrading option,
- various image preview and display options, and
- interactive tools for quality control.

1.3 Limitations

PARGE has been developed in view of processing of airborne hyperspectral imagery. It is limited to the following restrictions:

- Memory: 4-8 times the size of the DEM is required as RAM for optimal performance of the procedure.
- Disc storage space: the geometric processing tends to require huge amounts of disk space. You can expect 2-3 times the size of the original data cube for a geocoded image, if the information loss shall be kept minimal. This problem may be avoided by keeping only the geometry information besides of the original imagery and geocoding processing results directly.
- The processing time scales with the number of lines and pixels per line. So, the package is not best suited (though applicable) for digital photogrammetry sensors having more than 5000 pixels per scan line due to performance issues.

- The accuracy of the results depends on the provided auxiliary data. If neither DGPS flight-path or an accurate attitude measurement is available per image line, standard geometric processing will most probably lead easier to a certain accuracy.
- The program is not suited for orthorectification of satellite imagery since it is based on an orthogonal geometry and does not consider curvature of earth surface.
- PARGE-internal coordinate conversion needs to defined by the end user or has to be done externally if non-standard coordinate systems are used.
- The standard image processing capabilities of PARGE are limited – a full image processing license (such as ENVI) is recommended in parallel to a PARGE license.

1.4 Future Extensions

The PARGE application is under continuous improvement. The following features are options to be potentially included in future versions of the software (depending on demand):

- dedicated import filters for new airborne instruments (upon request),
- support for central perspective frame/video data processing, and
- self consistent fully automatic image-to-image boresighting.

Such features are implemented based on specific requests of licensed end users. Please contact the software distributor, if you have new ideas or wishes to the software or if you'd like to contribute suited IDL-based tools to be included in the processing system.

1.5 Terminology

Below is a short compilation of major process terminology as used in throughout this manual:
- *attitude angles* are the angles to measure the sensor axis in space
- *DEM* is a digital elevation model (i.e., the altitude of the terrain ground)
- *DTM* is a digital terrain model (synonymously used to DEM).
- *DSM* is a digital surface model (i.e., the altitude of the visible surface)
- *direct geocoding* is a geocoding process where the pixel coordinates are calculated directly from parameter
- *geocoding* is the process of assigning geographic coordinates to image pixels
- *geocorrection* is the process to bring an image to a georectified state with whichever means.
- *georeferencing* is almost synonymous to geocoding, but may also refer to assigning coordinates to a whole image instead of individual pixels.
- *georectification* is the process of creating rectified maps from geocoded imagery.
- *navigation data* is the ensemble of GPS-derived information, i.e., mainly the position of

the aircraft.
- *parametric geocoding* is a geocoding process where flight geometry (navigation and attitude) and sensor parameters are used
- *rectification* is a resampling approach which removes distortions from raw imagery.

1.6 Organisation of this Manual

This manual shall lead the user for appropriate use of the PARGE software. Knowing that the parametric, direct geocoding approach follows completely different principles than traditional geocoding approaches, a review of this manual is highly recommended prior to using the software.

The following chapters are organized as follows: first, the theoretical principles covering the physics and algorithms within PARGE are described. This chapter mainly is based on scientific publications and gives a thorough review about the theoretical background. The chapter 'Typical Workflow' gives guidelines how to work with PARGE in interactive mode. It summarizes tips for working with standard sensor data and how to deal with special cases. The chapter 'Working with Ground Control Points' gives important hints, how ground control points can be incorporated into PARGE for improvement of the final results.

The PARGE batch mode capabilities are described in chapter 'PARGE Programming Interface'. This includes a full internal data format description as well as the usage of all essential programming functions for creation of PARGE batch scripts. All information in this chapter is required if the PARGE functionality shall be used in a fully automated environment.

The last chapter 'Functions Reference Guide' gives a description of every function of the PARGE user interface and on the usage of the interface functions. Finally, in the appendix the bibliographic references as well as an index of topics can be found.

Some conventions in the manual:
- menu commands are given as **>File:Save Status<** p.147, with a link to the description page,
- PARGE variables are written in italic (eg., *demarr*), and
- Batch routines and calls on the IDL prompt are written in monotype, e.g., `gc_cleanup.`

Please check out the warning signs on the side-bars carefully and do not hesitate to contact the software support through email before you run into serious troubles.

Chapter 2:

Theoretical Background

Orthorectification of airborne scanner data can be solved using various approaches. Traditional polynomial 'rubber sheet' approaches (e.g., [3], [22]) require a large number of tie-points to account for sensor movements and do not achieve satisfying accuracies for airborne data [44]. Although piece wise polynomial models offer a certain potential even for airborne data, they can not solve high frequency distortions in the image (except if linear features of the image are employed [42]). On the other hand, correlation matching algorithms ([9], [18], [49], [67]) can deliver adequate results if the image is highly structured. The raw image is fit to stereographic or orthophoto imagery but usually requires human interaction for correction of mis-matching points. However, the latter approach only leads to satisfying results if an adequate orthophoto database is available [24].

The parametric direct geocoding approach is a favourable solution for imaging spectrometry data since it does not require any referenced image data sources. Parametric solutions have been implemented for satellite imagery taking into account the respective sensor models and orbit parameters initially for Landsat MSS by Bähr [3] and later by Sawada [54], then for SPOT HRV [34], and generic orbits [53]. With all these approaches, the number of Ground Control Points (GCPs) could be significantly reduced in comparison to the pure 'rubber-sheet' approaches while increasing accuracy. The achieved accuracies for the satellite data as described in the above papers was between 0.5 and 1.5 pixels. The envisaged geometrical accuracy for airborne imaging spectrometry is pixel-accuracy after geocoding. This requirement is defined as an absolute pixel position accuracy being better than 50% of the actual pixel size. At a nominal spatial resolution of actual instruments between 2 and 20 meters, the generic accuracy requirement is in the range of 1 to 10 meters.

Parametric models have already been implemented for various airborne scanners: e.g., AISA ([4]); AVIRIS ([8], [10]) ; CASI ([2], [72], [68]), and Daedalus ATM AADS-1268 ([7], [46]) and have become popular with the invention of photogrammetric line scanner imagery, where they are known as 'direct georeferencing' approaches ([14], [15], and [52]). These solutions are partially tied to proprietary systems, and some of them do not allow to include the topographic information of a DEM. An in-depth evaluation of these methods is beyond the scope of this manual. The current status and detailed methodology used in the mentioned methods may be obtained directly from the respective authors.

The motivation of PARGE was to implement a flexible generic methodology for parametric orthorectification with special emphasis on the requirements of currently used airborne imaging spectrometry data (such as AVIRIS, HyMap, AISA, or HYSPEX). Hence, it has to deal with the intrinsic data quality of these sensor systems and its associated auxiliary data. The "Parametric Geocoding" procedure (PARGE, [62]) reconstructs the scanning geometry for each image pixel using position, attitude, and terrain elevation data. A first version of the model has been created at the Remote Sensing Laboratories (RSL) of the University of Zürich starting in 1992 [45] as an AVIRIS specific tool. It then evolved to a generic application including extensive GCP based calibration and boresight alignment procedures ([56], [57], [62]), and was finally operationalized under the hood of ReSe Applications Schläpfer. The current implementation allows a complete automation of the geocoding and rectification process as long as the auxiliary data are provided calibrated in space and time to the resolution of an image pixel.

A sophisticated methodology is required to achieve highest precision with the direct parametric processing approach. The basic geometric model used for the processing is explained in this chapter. The implementation including processing workflow and the functionality of the individual modules are shown subsequently.

2.1 Input Data

By definition, any parametric approach relies on physically measured auxiliary data. The respective entities, their preparation as well as their definitions are given herafter. The requirements for the indispensable input data are summarized in Table 2.1.

2.1.1 Auxiliary Data Entities

The critical parameters for the procedure include the sensor model and the synchronization uncertainty in addition to the six classical orientation parameters.

a) Aircraft Position and Attitude Data

These data consist usually of aircraft GPS position (longitude, latitude, height) and attitude data (roll, pitch and true heading, in some cases yaw) stored for each line (or even each pixel) of the scanner image. The raw GPS data needs to be converted to an orthographic coordinate system for use with PARGE.

b) Digital Elevation Model

The DEM has to be provided in the same metric orthographic coordinate system as the airplane navigation data. The spatial resolution is chosen based on the nominal pixel size of the image. The DEM sampling has to be equally spaced in x and y direction. The DEM may be used to initialize the final geometry of the geocoded image. If no DEM is available, a flat 'dummy' DEM needs to be created or the processing is done on the basis of an average ground altitude.

c) Image/Sensor Data Acquisition Information

The geometric sensor model (i.e., the interior orientation of a sensor) is given by the position of each detector element in relation to the optical axis of an instrument. For a whiskbroom scanner, this can be defined by exact information on FOV (field of view) and IFOV (instantaneous field of view) of the sensor. For synchronization purpose, the scanning frequency, starting time, missing lines, and dimensions of the image are required in addition to the geometric information. Additionally, the so-called boresight misalignment (the angles between the axes of the INS (Inertial Navigation System) and the camera) should be known for each data acquisition.

2.1.2 Auxiliary Data Handling

For use with PARGE, all data are read using a common user interface. The coordinates of the DEM and the flightpath have to be available in consistent metric rectangular geodetic coordinates to make use of the formulation of the geometric model. The spatial resolution of the DEM is ideally chosen based on the nominal pixel size of the image. Depending on the processing workflow, the DEM resolution may initiate the final geometry of the geocoded image.

Table 2.1: Parametric geocoding data entities, accuracy requirements, and potential accuracy based on standard hyperspectral sensor characteristics and currently available technology.

Group	Parameter	Description	Accuracy Requirement[1]	Potential Real Accuracy	Position Error Δp_i
Sensor	Sensor model	Theoretic view angle θ_v per pixel center	0.1 mrad	~0.05 mrad	0.10 pixel
	Sync	Accuracy of synchronization	8 ms	~5 ms	0.14 pixel
	Boresight	Angular offset to INS	0.1 mrad	~0.05 mrad	0.10 pixel
DGPS	x/y	Aircraft coordinates	0.8 m	0.1 m	0.03 pixel
	z	Aircraft altitude	2 m	0.3 m	0.02 pixel
	Transform	Transformation to local coordinates	0.1 m	0.01 m	see text
Attitude	roll/pitch	Attitude per image line	0.1 mrad	0.15 mrad	0.30 pixel
	heading	True heading to direction north	0.6 mrad	0.4 mrad	0.13 pixel
DEM / DSM	altitude	Surface accuracy	2 m	0.1 m	see Table 2.2
	position	in alignment to the flightpath	0.5 m	0.1 m	see Table 2.2

1. for IFOV=0.5 mrad / Resolution = 4m / FOV = ± 15 degree / Frequency 25 Hz / Flight altitude 5 km

Note that a digital surface model (DSM) should rather be used than a DEM to allow maximum accuracy.

The auxiliary information needs to be exactly synchronized to the image data per scan line. Measurements of these data for each image pixel is an option which may be included in future PARGE versions for slow rotating whiskbroom-type instruments. Such calculation will only make sens highest quality attitude measurements are available. A consistent data format is used containing image and DEM descriptions as well as all sets of the auxiliary data. The format helps to keep track on the processing steps and to store intermediate status reports. It is designed in a generic way such that it is easily adaptable to any airborne scanners.

The minimal accuracy requirement as displayed in Table 2.1 for each parameter is derived equivalent to a pixel accuracy of one fifth of the pixelsize under standard geometric conditions. Furthermore, the estimated technically feasible accuracy for most parameters is listed (ESA, 2000). The goal of this limit is to obtain pixel accuracy of the final results. The most critical parameters are sensor model, synchronization, and the three attitude angles. Since these parameters are physically approximately independent, the overall accuracy ΔP is

$$\Delta P = \sqrt{\Sigma(\Delta p_i^2)} = 0.37 \quad \text{pixels}\,, \tag{2.1}$$

which is well below the theoretically required ±0.5 pixels as required for 'pixel-accuracy'. The individual deviations Δp_i are the horizontal pixel position errors as given in Table 2.1.

Sometimes, parts of the auxiliary data are not known exactly and must be estimated or interpolated from external sources. This can occur even in generally well documented test sites, which were flown with high performance sensors. Details on procedures to correct such artefacts are given below in Section 2.4 on page 32.

2.1.3 Parameter Definitions

The parameters as used in PARGE are defined following technical standards for attitude measurement. Internally to the program, all parameters are stored and used in standard radians and meters. Anyhow, the angles are usually displayed/entered in decimal degrees. For calculation purpose, the parameters are converted internally to the units and sense appropriate for the respective calculations. Below are the definitions of the parameters which apply for data import and storage of the attitude information:

roll

Right wing up is positive: this definition is according to navigational standards. The roll is temporarily inverted by PARGE in course of internal calculations to be in line with the positive rotation sense as defined from mathematical point of view.

pitch

Aircraft nose up is positive: this definition is consistent with calculation and common definitions.

heading

Right turn from north is positive: according to the common definition of geographical directions. This direction is temporarily inverted internally to fulfill the requirements for the equation given below, e.g. Eq. (2.6).

yaw

Left turn from north is positive: according to INS definitions. Note that the yaw parameter is only used in special situations by PARGE and is mostly not of any further relevance.

FOV

Scanning from left to right is positive. The FOV can be set negative in PARGE to simulate a sensor scanning from right to left in flight direction. It also may be included explicitely.

Navigation Data

Coordinates and altitude are metric. A strictly rectangular and metric coordinate system is assumed for all calculations within PARGE. Any geographical coordinates (Longitude and Latitude in degrees) will not produce acceptable results. Non-standard units such as feet, miles or kilometers are not accepted.

2.2 Geometric Algorithms

Fundamental equations describing the airborne scanning geometry have been derived by Derenyi and Konecny [17] and further refined by Konecny ([28], [29]). For PARGE a calculation approach has been chosen which does not make use of the collinearity equations directly [28] but is capable to use the collinearity condition on each parameter individually to perform a boresight calibration. The outline of the chosen core processing procedure is given first whereas the related algorithmic steps are described in more details thereafter.

2.2.1 Main Processor

The following steps are performed during the core processing algorithms:

a) Calculate the current observation geometry (see Figure 2.1)

The theoretic scan vector ($\vec{L_0}'$) is calculated between the airplane position and a supposed 'flat' DEM, using the instruments sensor model information. This vector is then transformed to the effective scan vector ($\vec{L_0}$) using equation (2.6).

b) Find the intersection point on the surface

The intersection point is found as described in Section 2.2.5. The exact intersection point position is usually written to an ENVI image geometry file (IGM).

c) Find the image coordinates

The pixel coordinates of the image (pixel and line number) are written to an array in DEM geometry at the intersection point position. The result of this procedure is a 'mapping array' (MAP, GLT) which contains the indices of the raw image coordinates, mapped to the correct positions on the DEM. The MAP and the image geometry map (IGM) describe the relation between image and orthorectified output in an unambiguos way.

d) Complete Mapping Array

The mapping array is filled using nearest neighbour techniques (expanding or triangulation). Any originally missing pixels are marked and the background mask is created. In parallel, the optional scan angle output may be written. This step may be omitted if final processing is based on IGM only.

e) Final Processing

The final processing step performs the effective production of orthorectified images. It is separated from the main processing algorithm to increase performance and to allow the separation between the geocoding and the georectification step. The 'mapping array' or the 'image geometry map' is applied as an index directly to the original image data to perform the final geocoding. Furthermore, optional interpolation and rotation steps may be applied only at this stage. This step is applied band by band which increases the speed of the processing.

The concept also allows one to process any image in original scan geometry. This has the advantage, that higher processing levels could be performed in scan geometry and only the final results are resampled to map geometry.

2.2.2 Scan Process Model

The scan process can be geometrically described by a linewise data acquisition with one auxiliary parameter set per image line for position (*x / y / z)* and attitude (roll ϕ / pitch θ / heading ψ). This description includes two approximations: first, the motion of the camera within the aircraft can be neglected if the attitude angles are measured directly on the sensor head, and the DGPS information has been corrected for the distance between GPS antenna and sensor head (the latter is a standard procedure for modern inertial navigation systems). Secondly, the scan period for whiskbroom scanners between the first and the last pixel in an image line is neglected if only one auxiliary data set per image line is available. Furthermore, it is assumed that the sensor optical axis is aligned to the IMU system within 2 degrees (compare accuracy considerations regarding the boresight calibration in Section 2.7.3 on page 39). Such approximations are allowed in order to stay within the required accuracy for airborne imaging spectrometry data as given above (± 1-10 meters).

2.2.3 Geometric Sensor Model

The geometric model applied in PARGE starts with the scan vector $\vec{L}_0'$ as seen in the sensor intrinsic coordinate system. The corresponding initial coordinate system is taken such that it corresponds directly to image coordinates, having the *x'* ('pixel') direction across track and the *y'* ('line') direction along track (see Figure 2.1). Note that this coordinate system is not in line with the common definitions used in classic photogrammetry. It rather has been chosen specifically for the line scanner situation such that the x-coordinate corresponds to the scan movements.

The sensor model for a standard scanning system assumes that all across track scan vectors lie in a co-planar system. The initial scan vector per pixel for a whiskbroom instrument can then be derived directly from the pixel scan angle as:

$$L_{0,x}' = s\tan(\Phi_i + \Delta\Phi)\,,\ L_{0,y}' = s\tan(\Theta_i + \Delta\Theta)\,,\ \text{and}\ L_{0,z}' = -1\,, \tag{2.2}$$

where Φ_i is the relative scan angle between the scan vector of pixel i and nadir in across track direction, ranging from $-(FOV)/2$ to $+(FOV)/2$ at monotonically increasing angular

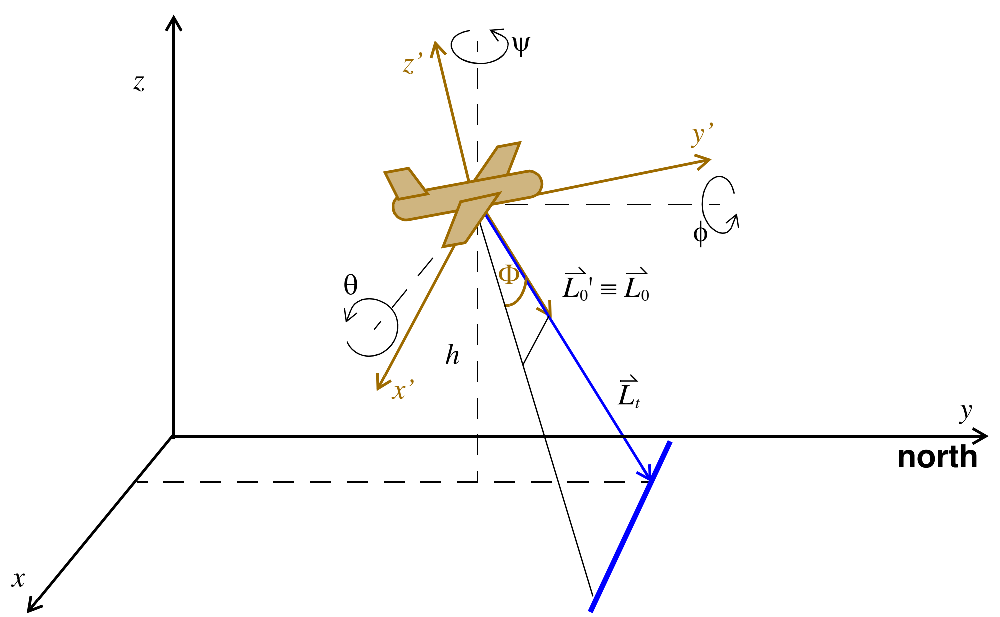


Figure 2.1: Rotated sensor coordinate system (*x',y',z'*) in space and the orientation of the attitude angles (ϕ, θ, ψ). The scan vector $\vec{L}_t$ is derived by transformation of the coordinate system.

intervals. Θ_i is the relative scan angle in along track direction, and s is the sign (±1) being negative for left hand side (in flight direction) pixels. The parameter Θ_i is given by the delay during the movement of the whiskbroom sensor systems across the field of view. The parameters $\Delta\Phi$ and $\Delta\Theta$ are the across track and along track tilt of the sensor to the horizontal position of the navigation coordinate system.

For an ideal pushbroom system, the initial vector is given as:

$$L_{0,x}' = s\tan(\operatorname{atan}(x_{dist}/x_{max}) + \Delta\Phi)\,,\ L_{0,y}' = s\tan(\Theta_{cent} + \Delta\Theta)\,,\ \text{and } L_{0,z}' = -1\,, \quad (2.3)$$

$$\text{with } -x_{max} \le x_{dist} \le x_{max} \quad (2.4)$$

where x_{dist} is the monotonically spaced distance of the projected pixel from nadir and x_{max} is the maximal projected distance given as $\tan(FOV/2)$.

Instead of using the two standard sensor models as described above, the angles to each pixel may also be defined explicitely as output of the geometric sensor calibration and used as such in the processing sytem.

2.2.4 System Transformation

The sensor coordinate system is now rotated by the attitude angles according to the measurement order of the three angles. This rotation leads to new coordinates of the scan vector $\vec{L_0}$ in the cartographic coordinate system (as defined by the target DEM):

$$\vec{L_0} = \mathbf{R} \cdot \mathbf{P} \cdot \mathbf{H} \cdot \vec{L_0}'\,, \quad (2.5)$$

which is explicetely written as:

$$\begin{bmatrix} L_{0,x} \\ L_{0,y} \\ L_{0,z} \end{bmatrix} = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} L_{0,x}' \\ L_{0,y}' \\ L_{0,z}' \end{bmatrix} \quad (2.6)$$

$$= \begin{bmatrix} \cos\phi\cos\kappa - \sin\phi\sin\theta\sin\psi & \cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & -\sin\phi\cos\theta \\ -\cos\theta\sin\psi & \cos\theta\cos\psi & \sin\theta \\ \sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi & \sin\phi\sin\psi - \cos\phi\sin\theta\cos\psi & \cos\phi\cos\theta \end{bmatrix} \cdot \begin{bmatrix} L_{0,x}' \\ L_{0,y}' \\ L_{0,z}' \end{bmatrix}$$

where R, P and H are the coordinate transformation matrices for the attitude angles. The attitude angles are read into to PARGE as Euler rotation angels roll (ϕ), pitch (θ), and true heading (ψ), respectively [5]. The dimensions of the parameters ϕ and θ are interchanged in the above equations in comparison to standard definitions [31] due to the scanner-compatible coordinate system, which had been chosen within PARGE. When importing, the true heading ψ is transformed to an orientation in counter clock-wise sense to direction north in order to account for the geographic (and navigation) standards.

The scan vector length has finally to be constrained to the aircraft altitude h in order to obtain the real scan vector $\vec{L}_t$:

$$\vec{L}_t = \frac{-h}{L_{0,z}} \vec{L}_0 . \tag{2.7}$$

The above equations describe, how the sensor is virtually turned from the aircraft system to the actual position ([31], [16]). This forward transformation is contrary to rotation methods as often applied in digital photogrammetry (e.g., [73], [5]). Also note that this rotation uses the orientation angles roll, pitch, and true heading as input, which are defined in relation to the earth centric navigation coordinate system. The transformation above ends in a cartographic coordinate system. The related approximation neglects effects of earth rotation and curvature, which is appropriate for airborne imaging spectrometry, at spatial resolution between 1 and 20 meters. Note that the photogrammetric triple omega, phi, and kappa cannot be used synonymously to phi, theta, and psi which are the finally used PARGE-internal angles for transformation.

2.2.5 Ray Tracing

An orthorectification process has to consider the terrain elevation in a precise manner using a ray tracing algorithm and a DEM, starting at the aircraft position $\overrightarrow{P_{air}(t)}$. The ray tracing intersects the effective view line with the DEM (see Figure 2.2). First the view line within the altitude range of the DEM (minimal height h_{min}, maximal height h_{max}) is created. Its starting point $\vec{P}_S$ and ending point $\vec{P}_G$ are defined as

$$\vec{P}_S = \overrightarrow{P_{air}} + \frac{h - h_{max}}{h} \vec{L}_t \text{ , and } \vec{P}_G = \overrightarrow{P_{air}} + \frac{h - h_{min}}{h} \vec{L}_t , \tag{2.8}$$

where h is the aircraft altitude above ground (compare Figure 2.2). The new intersection line $\overline{\mathrm{SG}}$ is defined between these two points. Its vertical projection on the DEM surface is now calculated by creating a profile along the trace of the vector, interpolating the profile altitude values from DEM. The highest intersection point $\overrightarrow{P_{pix}}$ between $\overline{\mathrm{SG}}$ and the profile is now found by comparing the altitudes of both lines on an equally spaced basis. The horizontal coordinates of $\overrightarrow{P_{pix}}$ can then be stored as final result of the geocoding process (while its altitude is now available directly from the DEM). The exact pixel position is finally determined by linear interpolation of the closest profile points.

There are various possibilities to intersect a vector with an irregular plane such as the DEM.

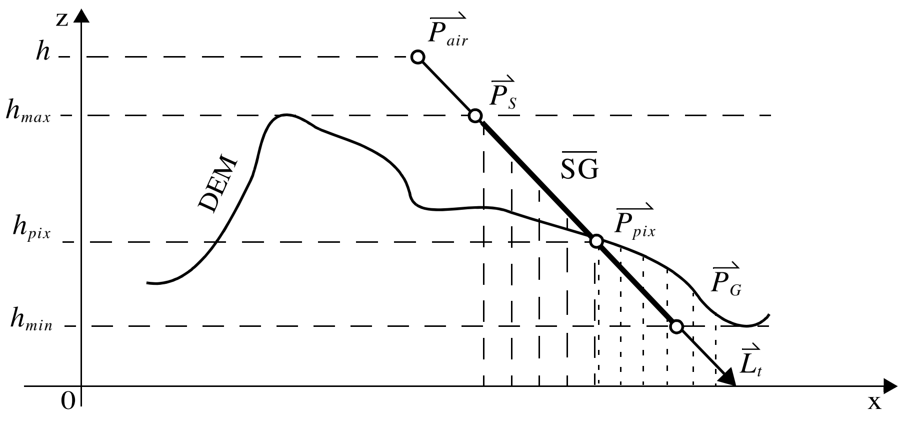


Figure 2.2: Intersection procedure of the real scan vector with the DEM.

Meyer [45] used a minimizing procedure of the angle between ($\overrightarrow{L_t}$) and a number of surrounding test vectors, while iterative solutions are used in other algorithms. The selected forward ray tracing procedure provides a sub-pixel-accurate intersection point position. It is specifically suited for airborne scanning situations where backward intersection (propagation from the DEM) is less efficient due to the linewise changing geometry. Moreover, the procedure is able to solve for the problem with multiple intersections of the view line with the DEM since the highest intersection point is selected first.

2.2.6 Final Map Production

In the final processing step, the previously stored image geometry is used to create a rectified cartographic image. The spatial resolution of the target image-map has to be taken slightly higher than the nominal resolution of the original image data. If the same spatial resolution would be taken, roughly 20-30% of all image information might get lost due to double mapping. However, it can not be avoided that some of the image data will be multiply mapped, while other parts will be lost because of aircraft motion. If the instabilities are relatively high, a high number of pixels in the final image will not be mapped by the scanner.

The resampling may be done on various stages of the processing. Originally, an oversampling method was proposed by Meyer [45] and used a temporary DEM with up to 16 times the original number of pixels. This method has been omitted due to performance problem. Currently, a variety of suited resampling methods to resolve this problem may be practiced using PARGE, as described in more detail in Section 2.3.

The positions of the original pixels in DEM geometry are stored in the 'mapping array' which

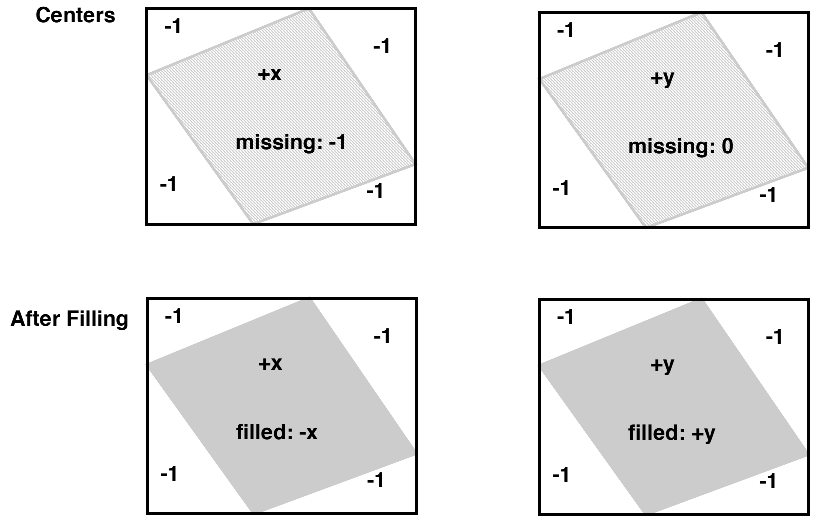


Figure 2.3: Structure of the two layers in the PARGE mapping array before (top) and after filling of center pixels (botteom). The grayed area depicts the image area whereas the white area is not covered by the image.

also contains information about the image borders and so called ‘center pixels’. Center pixels are pixels in the DEM which can be directly related to original image pixels by their position. The structure of the resampled mapping array as used in PARGE is given in Figure 2.3. This file is equivalent to the Geolocation table (GLT) as used in ENVI. Alternatively, the x/y coordinates of each pixels are stored in an ‘Image Geometry Map’ (IGM), which can be used to interpolate to any target geometry in a very flexible way.

The geometric panorama effects (larger pixel footprints at off-nadir positions) are corrected satisfyingly by the resampling techniques. Radiometric panorama effects do not need to be considered because the sensor characteristics are constant for every pixel if an electro-mechanical ‘whiskbroom’ scanner is employed for data acquisition. For hyperspectral pushbroom instruments such as APEX [27], the IFOV (Instantaneous Field of View) may slightly vary across track and can therefore cause radiometric panorama effects. Such influences will have to be measured by the laboratory calibration procedure and be corrected in the data calibration process [60].

2.2.7 Link to Radiometric Processing

The scan angles output provides all the linking layers for later radiometric processing of the geocoded image (cf. [59]). It includes three data layers in DEM resolution describing the relative geometry between airplane and observed pixel: the scan zenith (being negative for pixels on the right hand side of the nadir), the absolute scan azimuth to direction north, and the altitude of the aircraft (see Figure 2.4). The scan angle output can (e.g.) be directly fed into ATCOR 4 which is the airborne version of the atmospheric correction software ATCOR ([50], [51]). The angles are defined according to Figure 2.5. The scan azimuth follows a fan distribution whereas the zenith angle only changes its sign at the nadir position.

The scan azimuth is required for the correction of BRDF effects [41] and the aircraft altitude is provided for completeness of the geometric description. Further radiometrically relevant geometric parameters which only depend on the DEM such as slope, aspect, and elevation are cal-

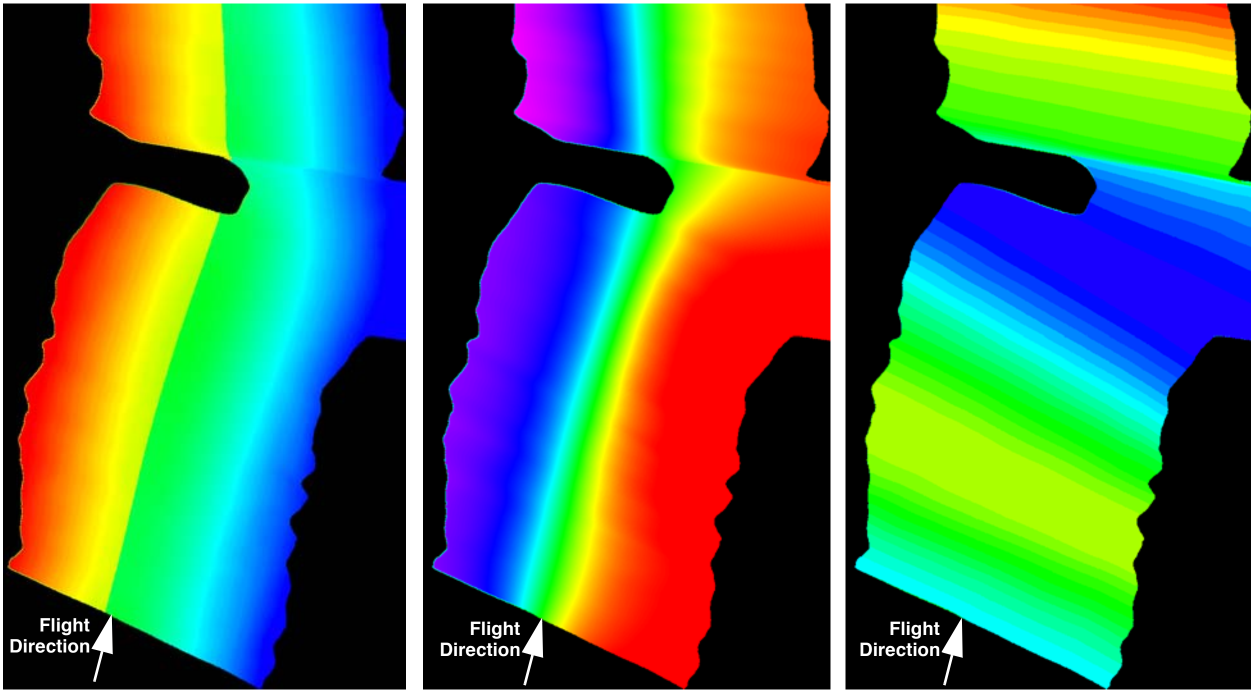


Figure 2.4: Example of the linking layers for radiometric processing of airborne scanner data (AVIRIS): scan zenith (left), azimuth (middle), and aircraft altitude (right). The nadir line appears prominently in the scan zenith due to the negative coding of flight-right hand side pixels. The color coding used is blue for small values, green for intermediate values, and red for large values.
The azimuth angle changes from left to right looking over the pitch angle which induces a north looking azimuth at nadir angles.

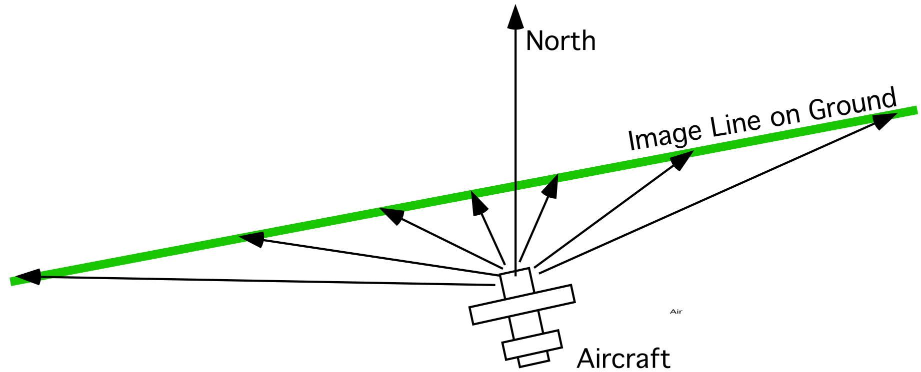


Figure 2.5: View angle derivation for each pixel for positive pitch values.

culated separately prior to the radiometric correction.

A pixel distance layer, storing the distance between aircraft and each pixel is added to this file for certain applications such as atmospheric processing or the combination with laser scanning data (compare Section 4.1.3 on page 79 for a short description of all respective layers).

2.3 Spatial Resampling Methods

The orthorectification calculates the exact positions for each imaged pixel. This output has to be resampled to achieve a regular grid (compare [64]). This step is required at the latest if data are required in map geometry. The kind of resampling has already lead to extensive discussions for the geocoding of multispectral satellite imagery. Cubic resampling has been found to lead to the spatially most accurate results for such data, while having a negative impact on the radiometric integrity [64]. The optimal choice of the resampling algorithm thus depends on the goals of the data acquisition and processing algorithms to be applied to the data.

2.3.1 Supported Resampling Methods

Resampling options are available at three places of the PARGE processing workflow: the geolocation mapping array needs to be resampled to form a continuous map, and the radiometric data are resampled if a geocoded output needs to be produced or the basis of the geolocation map or on the image geometry map (compare Section 3.2.3 on page 47). Hereafter, a description of the resampling methods supported by PARGE are given:

a) Nearest Neighbor Methods

The interpolated pixel value is transferred from the nearest pixel position as:

$$L_i(x_p, y_p) = L(x_p + \Delta x, y_p + \Delta y) \ , \tag{2.9}$$

where $L_i(x_p, y_p)$ is the re-located radiance at the final pixel position x_p, y_p, L is the original radiance value, and Δx and Δy are the distances to the selected nearest neighbor, where $\sqrt{\Delta x^2 + \Delta y^2}$ is the minimal distances to the selected position. Two implementation options are available for nearest neighbour resampling:

- The *expanded center pixel coding* (see Figure 2.6) is a simple technique which expands each final pixel by a surrounding cross in map geometry. If an adjacent pixel is already occupied, no replacing will occur. This growing technique is fast, and yields satisfactory results as long as the spatial accuracy is not of highest priority and the DEM-pixels are of the approximate size of the original image. The method is recommended for creation of mapping arrays.
- In the *triangulated nearest neighbour method*, the center pixel locations are triangulated to remap the missing pixels based on a gridding procedure (see Figure 2.6). The triangulation is a true nearest neighbor technique for filling all occurring gaps between the center pixels. The produced TIN (Triangular Irregular Network) could be used to achieve whatever image final resolution is required. Another advantage of this method is its independence from final product resolution.

b) Natural Neighbor Interpolation

Each interpolant is a linear combination (weighted average) of the three vertices of its enclosing Delaunay triangle and their adjacent vertices. The weighting of the natural neighbours is done on the basis of the area influenced by the respective neighbor [1]. Therefore, it is well suited for radiometric interpolation (as also radiometry is closely coupled to the area rather than to the

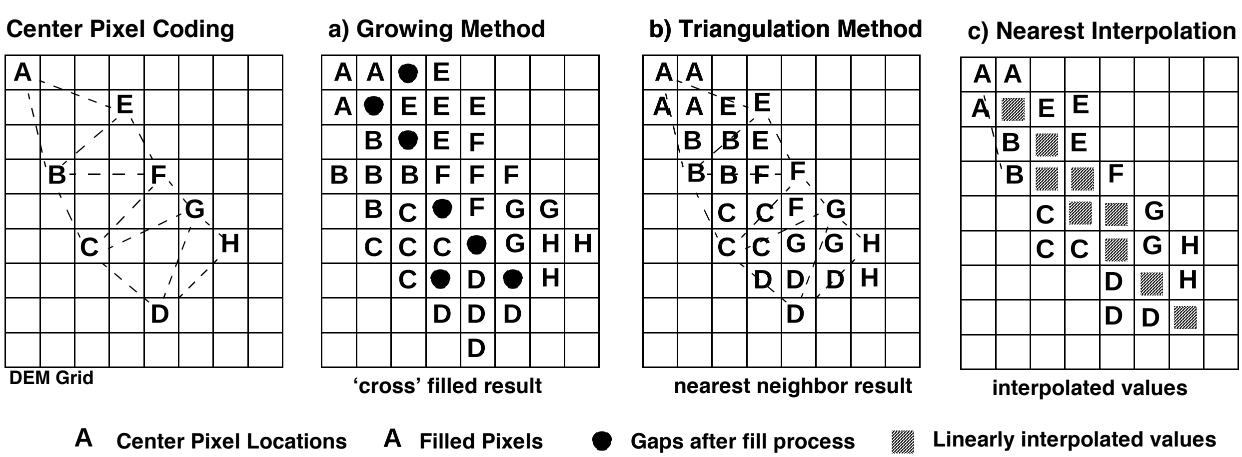


Figure 2.6: Resampling ('Gap filling') methods for slightly oversampled output maps, used when resampling in map geometry

distance. However, the computational requirements of this method inhibit the extensive use of the method on large data sets.

c) Linear Interpolation Methods

A computationally fast linear interpolation can be performed by averaging the linear interpolates of the two spatial image dimensions such that:

$$L_i(x_p, y_p) = \frac{L(x)_{x_p} + L(y)_{y_p}}{2} , \tag{2.10}$$

where $L(x)_{x_p}$ and $L(y)_{y_p}$ are the linearly interpolated continuous functions of radiance in across track and along track directions, respectively. This approach does not include a full consideration of the relative distances of the pixels and thus may lead to artefacts whenever the interpolation distance is more than the size of one pixel.

Two variations of this method are available in PARGE:

- *Nearest bilinear interpolation*: A straight-forward compromise between linear interpolation and nearest neighbour interpolation is implemented by this selective procedure. It only applies a linear interpolation where direct neighbours are available to a missing pixel. The center pixels remain untouched.
- *Across/along track interpolation*: It was shown, that the spectral accuracy of resampled pixels is increased if spatial interpolation is applied to direct neighbours. This method preserves the original spectra in the center pixels while all other pixels may be replaced in an explicite direction.

Note: the latter method is not further recommended and is therefore not available through the PARGE GUI since it tends to artefacts in the results. The option is only available in batch mode and is only recommended for special cases.

d) Bilinear Interpolation

An accurate bilinear interpolation is done on the basis of a triangulation before interpolating. The interpolation is afterwards done by gridding the triangulated surface to a uniform target raster, where the interpolated value is found as:

$$L_i(x_p, y_p) = L(x, y)_{triang}\big|_{x_p, y_p} , \tag{2.11}$$

which represents a two-dimensional (i.e., bilinear) interpolation from the three closest original data points. Two options are supported:

- A true *bilinear interpolation* can be done by using the original image geometry map (*_igm, containing the pixel locations) instead of the center pixels for triangulation. This method provides highest spatial accuracy, while all radiometric values are the result of an interpolation.
- *Bilinear interpolated gaps*: This method makes an overlay of the bilinear interpolation with

the center pixels in map geometry. The originally measured spectra are thus preserved in the center pixels, whereas the gaps are interpolated to the maximum possible spatial accuracy. This method is a good compromise between preserving the original spectra while improving spatial accuracy. However, it may produce inconsistent results if the target resolution is significantly higher than the original resolution.

e) Spatial Binning

High resolution instruments often are flow in such a way that the image data is oversampled in either across track or along track dimension if compared to the target resolution of the image map. The spatial binning option available in PARGE allows to average a number of originally measured spectra depending on the relation between target resolution and the average along/across track pixel distance. The optional binning is performed whenever this relation exceeds a factor of 2 and is adjusted in full numbers to the target resolution. The spatial binning is a step preceding the final resampling and may be combined with most of the methods as described between a)and d).

f) Other Methods

A number of more sophisticated resampling methods have been proposed and applied to remote sensing data in the last decades ([35], [36], [1]) such as cubic convolution [37], radial basis functions [38], spline interpolations [39], or neural network based processing [40]. In such methods, not only the closest pixels but also more distant pixels are used to construct a smooth surface. The closest pixels of a remote sensing image is radiometrically directly related to the searched pixel, while pixels further off may already cover a completely different surface type. Extensive validation results for the various resampling strategies can be found in the above-mentioned references. Any of these higher order interpolations (IDL, [26]) are not included in PARGE due to their tremendeous requirements in processing ressources and their minor advantages in radiometric accuracy if compared to the proposed resampling options.

2.3.2 Nearest Neighbor Resampling or Data Interpolation ?

For airborne imaging spectrometry, a paradigm so far was to leave the spectra 'as is' in order to avoid interpolated spectra in final data products – any interpolation would lead to never measured 'unphysical' spectra. The arguments for such nearest neighbor based resampling are twofold. Firstly, the spectro-radiometric accuracy and spectral uniformity is often higher weighted than the spatial accuracy. Using nearest neighbour approaches, spectral integrity is preserved. Secondly, the interpolation processing of full image data cubes may be very time consuming if, e.g., cubic convolution would be applied to such data. However, it can be shown [61] that spatial interpolation should be preferred for the resampling of direct neighbour pixels, if more priority is put on geometric accuracy than on spectral integrity.

Both, nearest neighbor as well as bilinear interpolation is best done on the basis of a Delaunay

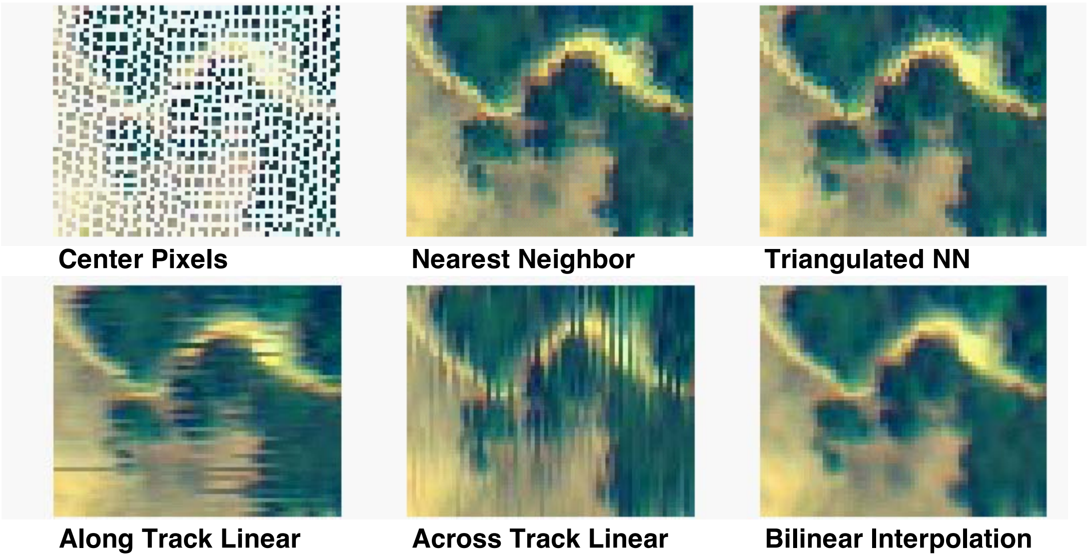


Figure 2.7: Spatial effects of resampling methods in an oversampled grid.

triangulation [26] on the pixel positions. For better performance, one may also apply fast buffering algorithms based on center pixels in the target grid. The advantage of the triangulation is a higher accuracy if more than singular final resolution pixels need to be filled. The produced TIN (Triangular Irregular Network) can be used to transform the imagery to whichever image final resolution.

2.3.3 Qualitative Evaluation of Resampling Methods

Using an exemplary set of AVIRIS data it has been tested how some of the proposed interpolation methods affect the spatial accuracy of the data. The low altitude AVIRIS data set stems from the Navarro River Watershed, Mendocino County, California collected in July 2000 (provided by UC Davis). The results of various interpolation methods are depicted in Figure 2.7 and Figure 2.8. Artefacts are evident if the initial geocoding procedure leads to an undersampled image as given in Figure 2.7. The gaps between the 'original' pixels need to be interpolated by the best suited resampling procedure. However, in order to take respect to the argument not to change radiometric measurements, the initial spectra at the center pixel positions are forced to remain unchanged. This oversampled situation is often preferred since it keeps the data loss minimal and usually leads to higher spatial accuracy. If the data amount must be kept minimal while preserving most of the original data, the output grid resolution is taken according to the original image resolution (see Figure 2.8). Hence, only few pixels are missing after geocoding and need to be replaced.

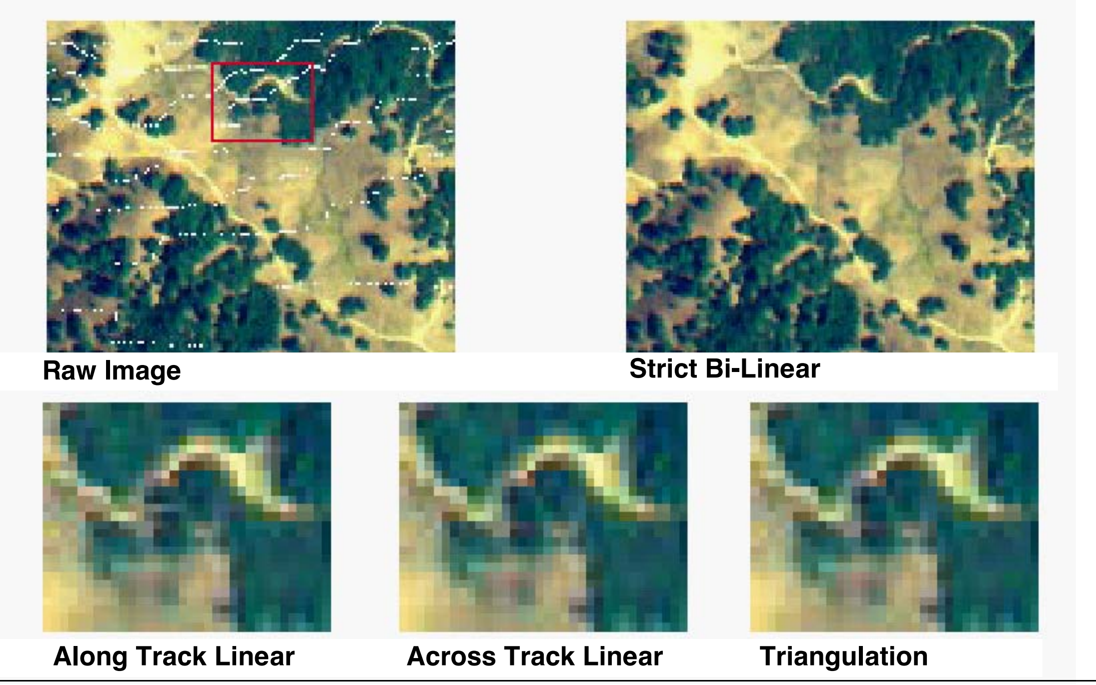


Figure 2.8: Spatial effects of interpolation methods in an equally sampled grid (on AVIRIS data).

For the undersampled output grid, it is obvious from Figure 2.7 that all linear interpolation methods fail to properly reconstruct a realistic spatial pattern in the image. Only a triangulated or bilinear interpolation can solve the issue satisfyingly. The spatial pattern is also kept if nearest neighbour resampling is applied, but the texture suffers from this kind of resampling by introducing non-realistic 'crispy' artefacts. The triangulated nearest neighbor still shows such artefacts although being more accurate by considering triangulated distances between the original pixels. If only few pixels have to be replaced as given in Figure 2.8, the triangulated interpolation looses its advantage over linear interpolations. A simple and fast linear interpolation can lead to well comparable results.

The effect of binning is shown in Figure 2.9. Bilinear interpolation bears the risk of losing to much information whereas simple nearest neighbour interpolation leads to higher noise in the data.

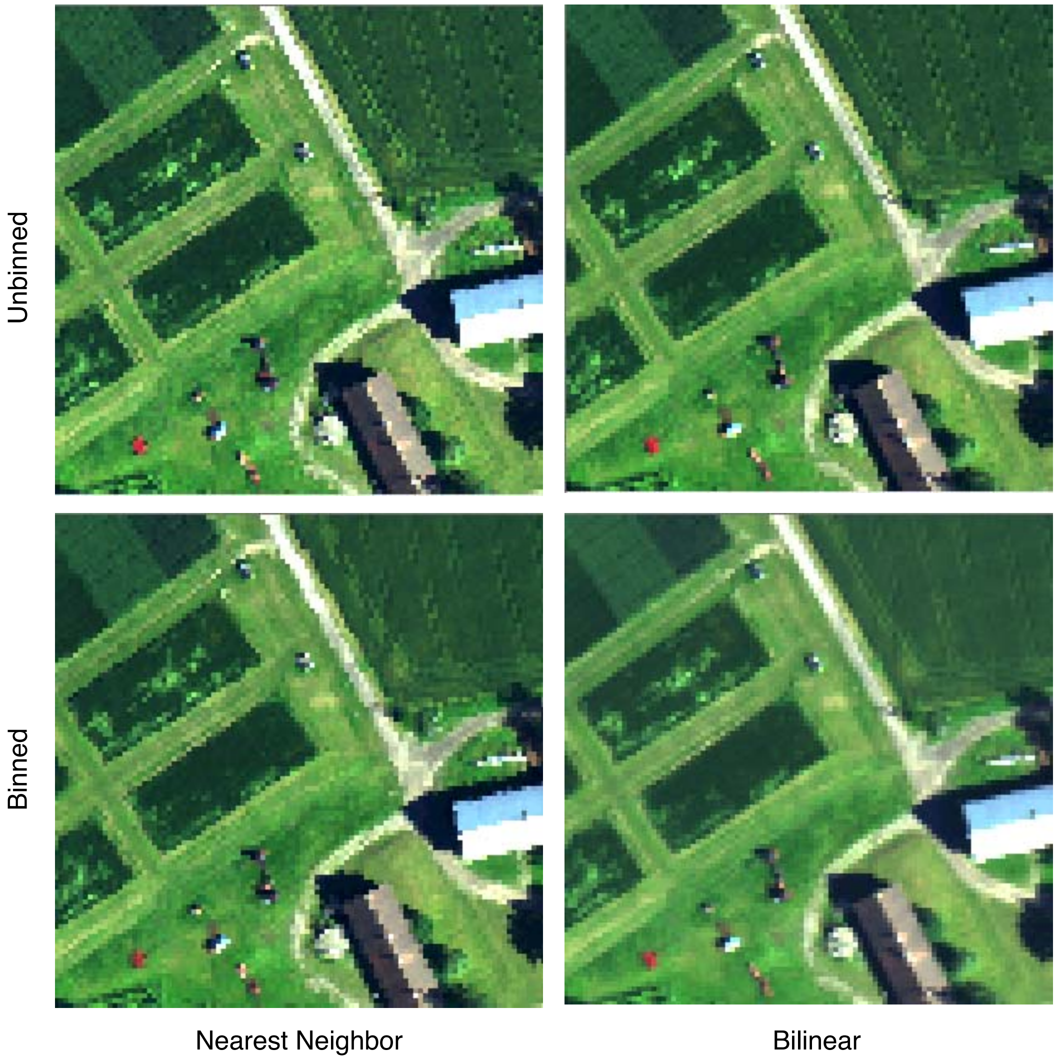


Figure 2.9: Effects of bilinear interpolation and spatial binning on data quality (here: 3x binning in across track direction applied).

2.4 Parameter Offset Determination - Boresight Calibration

A ground control points (GCP) based offsets estimation and boresight calibration tool was developed for the PARGE application. The inversion of the georeferencing algorithm allows the calculation of the airplane position for each GCP. The transformed view vector is upward-followed from the GCP position to the aircraft altitude:

$$\vec{P_a}' = \overrightarrow{P_{GCP}} - \vec{L_t}\frac{h_a - h_{GCP}}{h(\vec{L_t})}, \tag{2.12}$$

where $\vec{P_a}'$ is the estimated aircraft position and $\overrightarrow{P_{GCP}}$ is the position vector of the GCP, with their absolute heights h_a and h_{GCP}. The horizontal differences between the estimated positions $\vec{P_a}'$ and the real navigation data $\vec{P_a}$ are analyzed to statistically obtain the offsets. Auxiliary data offsets can be calculated for roll, pitch, heading, x-y-navigation and aircraft altitude. The angular and distance offsets for a number of GCPs are evaluated statistically to obtain the corresponding offset estimates as follows (see Figure 2.10):

- *roll*: average of the angular offsets in scan direction,
- *pitch*: average of the angular offsets in flight direction,
- *heading*: minimum correlation of the angular offsets in flight direction (pitch) to the pixel distances from nadir,
- *x-offset*: average of the distance offsets in longitudinal direction,
- *y-offset*: average of the distance offsets in latitudinal direction,
- *aircraft altitude*: minimum correlation of the angular offsets in scan direction (roll) to the pixel distances from nadir (also related to the sensor field of view).

For heading offset estimation, the correlation between pitch offset and nadir distance is minimized by iteratively adjusting the true heading average. An analogous procedure is used for the height with the roll offset as indicator. Because each offset potentially depends on the others, iterations may be done between them; e.g. the heading offset may be iterated together with the pitch offset over sloped terrain. Note that the first three offset parameters of the above list are equivalent to what is known as the boresight calibration of an instrument. There's a methodological difference as the boresight angles are applied to the attitude angles instead of adding them to the sensor model (compare Figure 2.14) It can be shown that this approximation is good as long as the roll/pitch boresights remain within 2 degrees. If larger boresights are detected, the angles should be applied as a fixed offset in the sensor model.

In some cases, the sensor field of view (FOV) is used to define the geometric sensor model and may not be known precisely. In that case, an aircraft altitude offset can be interpreted as a FOV offset and vice versa. Each is derived from the other by the equation:

$$FOV_1 = 2\,\mathrm{atan}\left(\frac{h_1 \tan\left(\frac{FOV_0}{2}\right)}{h_0}\right) \quad \text{from} \quad h_1 = h_0 \frac{\tan\left(\frac{FOV_1}{2}\right)}{\tan\left(\frac{FOV_0}{2}\right)}, \tag{2.13}$$

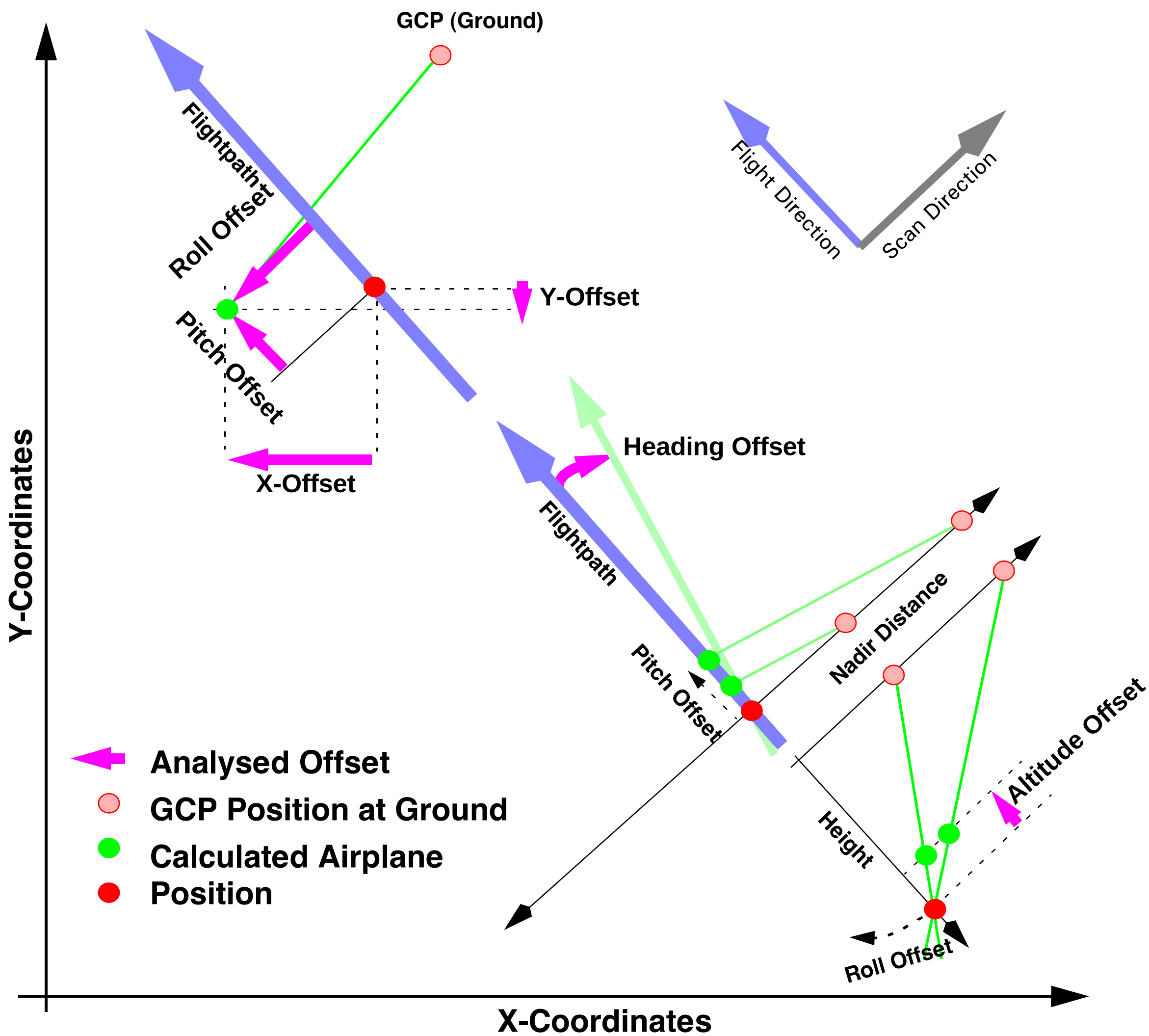


Figure 2.10: Top: offset analysis for roll, pitch, and X/Y-coordinates based on one single GCP. Bottom: offset analysis for true heading and average height, respectively, based on the offset statistics of multiple GCPs.

where h_0 is the original flight altitude, FOV_0 is the original FOV while h_1 and FOV_1 are their corrected counterparts. Note that this method can only estimate the average height offset for an entire flightpath.

2.5 Parameter Interpolation Algorithms

The flightpath trajectory information should be provided with each data set. If no such path is available, the PARGE flightpath reconstruction procedure may be applied based on a number of GCPs: the xy-position of the plane is determined for each GCP, and an average flight height is derived from the statistics of additional GCPs. The following assumptions allow such a flightline reconstruction based on a number of GCPs:

- the flight altitude is constant within the required accuracy,
- the flight velocity is constant within GCP distances, and
- the flight path is more or less straight without any hard turns.

The aircraft position then is calculated using a cubic spline interpolation between the position points (see Figure 2.11). Errors may be introduced into this procedure if the height is not constant during the overflight and if the GCP accuracy is lower than the resolution of the resulting image. The procedure needs approximately one or two GCPs for 100 image lines for flightpath calculation and another GCP for the offsets determination within the same area.

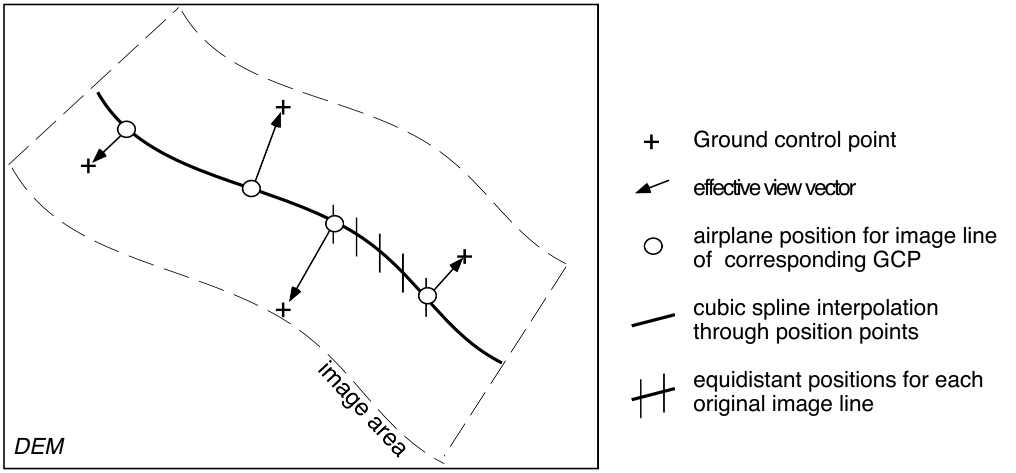


Figure 2.11: The flightpath reconstruction procedure.

Vice versa, roll and pitch can be interpolated if the flightpath is known exactly and the registration of roll and pitch failed (or was not available). Due to the (quasi) non-continuous characteristics of the attitude data, it is more appropriate to linearly interpolate the values derived per GCP. Spline interpolation only is useful, if the number of GCPs is very low. If the true heading is missing, its interpolation can not been easily done based on GCPs but has rather to be estimated from the first derivative of the flightpath.

2.6 Image Based Synchronization

If image and auxiliary data stream is out of sync, the direct geocoding will fail. Post-synchronization is required in such cases. The PARGE preprocessing system provides a procedure for that purpose.

The optically most obvious distortion in airborne imagery are usually roll-induced across track line shifts. This distortion can be derived from the imagery itself by systematic line-by-line correlation analysis. The offset between the lines in across track direction is attributed to the roll movements. It is derived by correlating the scan lines in along track direction at an accuracy of 0.5 pixels within up to 4 pixels shift. The best correlation shift is calculated and stored in a first roll estimate. Low frequency variations are removed and finally filtered using a high pass filter set to 250 image lines (this parameter may be changed in batch mode). This roll estimate is

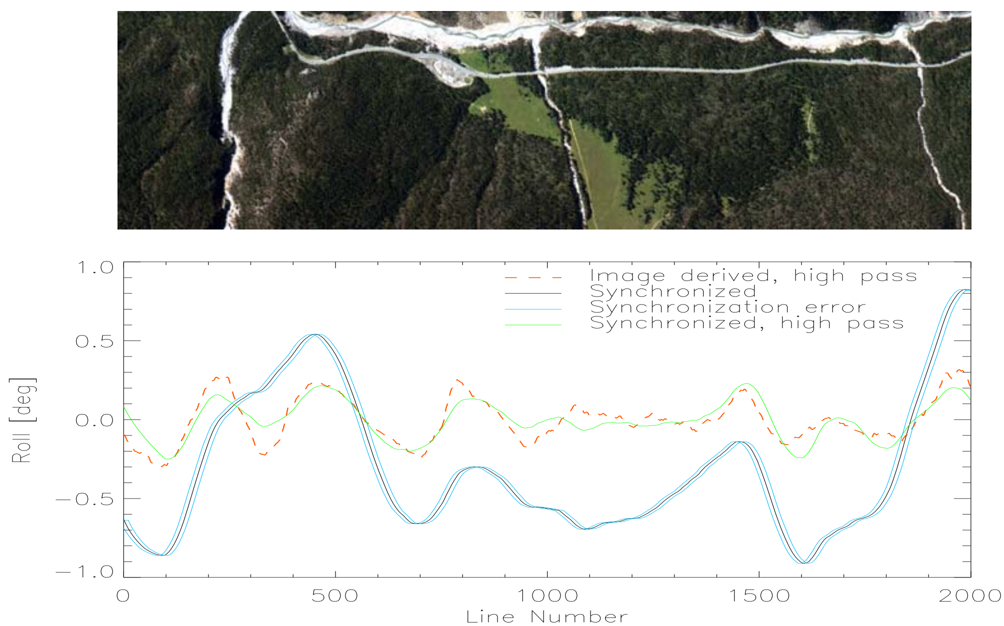


Figure 2.12: Synchronization of ROSIS roll parameter (dashed) to the image based roll estimate, raw image given at upper left. Best correlation is found within ±5 image lines (given by the added additional lines).

then correlated to original roll parameter to find the best synchronization offset within ±50 lines at an accuracy of 0.5 lines. The method has been developed and successfully applied for the ROSIS imaging spectrometer system [19] as shown in Figure 2.12. However, the method may fail for systems flown on a well stabilizing platform or on natural targets without visible spatial structures.

2.7 Accuracy Considerations

An increasing number of remote sensing applications such as GIS based modeling require pixel accuracy for imported information layers. High spatial accuracy is also required if in-flight calibration experiments have to be performed [55]. The PARGE approach bears the theoretical potential to achieve this goal even for high resolution airborne data. However, a number of errors given below have to be minimized.

2.7.1 DEM/DSM Related Errors

DEM resolution and positional accuracy are major error sources in the orthorectification process. Considering the residual height of vegetation and buildings, the inaccuracies increase since standard DEMs seldomly represent the observed surface. Vertical accuracy can be easily related to horizontal accuracy depending on the scan zenith angle. A short analysis of the DEM-related horizontal error is shown in Table 2.2.

Table 2.2: Horizontal accuracy in relation to the off-nadir scan zenith angle and the DEM vertical error Δh.

Off-Nadir Scan Angle	$\Delta h = 2m$	$\Delta h = 5m$	$\Delta h = 10m$	$\Delta h = 20m$	$\Delta h = 50m$
10°	0.36	0.9	1.8	3.6	8.8
15°	0.54	1.3	2.7	5.4	13.4
20°	0.73	1.8	3.6	7.3	18.2
30°	1.15	2.9	5.8	11.5	28.8
40°	1.68	4.2	8.4	16.8	41.9

Airborne imaging spectrometers typically have a FOV (Field of View) between ±15 degrees and ±30 degrees and spatial resolutions between 1 and 20 meters. Thus the critical surface elevation accuracies to achieve pixel accuracy are between 2 meters (e.g., HYSPEX case) and 50 meters (e.g., AVIRIS high altitude case). These errors are equivalent to the typical height of buildings or trees. For high quality results in standard European landscapes it is therefore advantageous to use high accuracy digital surface models for orthorectification instead of ground elevation based DEMs.

Notes on Forests

The preprocessing of the imaging spectrometry data over the forested area proves to be a very challenging endeavour. The underlying problem is that the exact surface characteristics for the optical response of a forest canopy can not easily be related to a DSM product (e.g., from a laser scanner). The multiple scattering processes within the volume of a plant canopy and the geometrical shape of the tree crowns lead to surface characteristics which can not be completely described by any 2.5-dimensional surface model. The vertical forest structure rather requires a true 3-dimensional view of the problem. Figure 2.13 tries to illustrate how the shape of a virtual 'crown-surface' may vary significantly depending on spatial resolution, sampling characteristics, and the shape of the crowns. An artificial optical reference surface could be defined on a top-crown level, on an average optical response basis within the canopy, or on the terrain surface. All these approaches are no satisfying solutions. Ideally, an illumination-equivalent layer would be calculated using a ray tracer combined with the 3D structure of the forest canopy. In such a way, the radiometric behaviour might be parametrized to a level suitable for further radiometric correction. Note, that any reference surface is a model far from the realistic 3D structure and may be useful only for specific and limited processing steps.

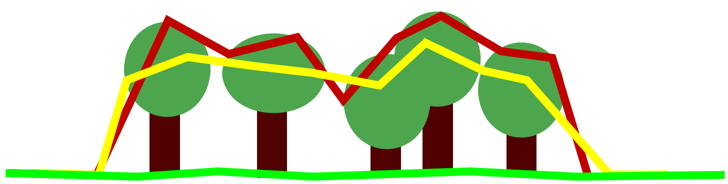


Figure 2.13: Digital surface description within a forest canopy.

Also, for subsequent radiometric processing, not a tree crown model should be used but rather the terrain model whereas the DSM is still used for geometry. Any radiometric variability within the forest is to be handled by three dimensional radiative transfer modeling rather than applying DSM-based correction approaches.

2.7.2 Auxiliary Data Error Sources

The requirements to the auxiliary data have already been given in Table 2.1 on page 15. Some critical details to the individual parameters are given below because each of these parameters can lead to corrupt results if it is erroneous.

GPS/DGPS

The flightpath of the aircraft as well as ground control points are usually measured with GPS/DGPS systems. The required accuracies as given in Table 2.1 can be achieved with current

DGPS systems. Special care has to be taken in the cartographic transformation of the GPS coordinates to the geodetic coordinate system of the DEM, where systematic errors up to hundreds of meters may occur (e.g., if a wrong datum has been selected for coordinate transformation). If non-differential GPS is used for data acquisition, the altitude becomes inaccurate to 10 to 50 meters and potential offsets and drifts in horizontal position have to be investigated. A further error source is the lever arm correction between GPS antenna position and sensor head. This correction is not done within PARGE but is expected to be done in the INS data preprocessing software.

Roll and Pitch

The attitude can theoretically be measured to sub-pixel accuracy [25]. However, the absolute calibration is often not accurately known or even absolutely unknown and thus the offsets need to be reconstructed using GCPs. All attitude parameters may be affected by drifts, particularly within long image runs (i.e., > 2000 lines). An automatic drift correction for roll or pitch can therefore lead to an improvement of results for long data acquisitions.

Heading

The true heading (i.e., the angle between aircraft orientation and DEM direction north) may suffer from the aircraft heading knowledge and from the coordinate conversion where the conversion of the heading angle is usually not included in the calculation. Furthermore, the definition of the north direction has to be carefully selected – while the cartographic north is relevant to the DEM and the final maps, the heading is usually given with respect to the geographic GPS north direction. The approximation of one parameter set per scan line for whiskbroom scanners introduces another small systematic error to the heading in the range of 0.1°. These effects can be mostly removed by correcting the heading offset from a number of GCPs.

Synchronization

The synchronization of the auxiliary data to the image lines is difficult if the instrument master clock is not exactly registered for each image line during data acquisition. Synchronization errors may lead to apparent drifts in the data due to data acquisition frequency changes or to unknown offsets between the auxiliary data stream and the image data. Exact knowledge of the registration for the instrument master clock and the steering of the auxiliary data registration is required to solve this problem. Thus, consideration of the measurements per image pixel (rather than per line) would only be advantageous if highest quality and high frequency (200 Hz and more) attitude measurement units are available perfectly synchronized.

Sensor Description

A final error source is the description of the sensor itself. The exact knowledge of the geometric sensor model is crucial. Equally spaced scan vectors across track are assumed for standard scanners as described in Section 2.2.3. Note that the intrinsic geometric parameters may substantially differ from the ideal situation and should be measured in laboratory for an accurate representation.Further parameters which may be sources of errors are the sensor tilt and boresight angles. If the sensor boresight angles are large than 2 degrees, the sensor model should be

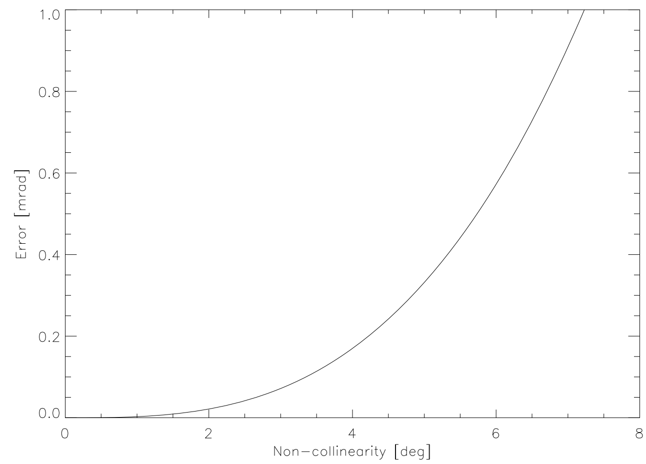


Figure 2.14: Error within the PARGE approach with respect to misalignement between sensor model and the attitude measurement unit. The angular difference between the exact solution (offsets directly in the sensor model) and the approximative solution with offset in the roll/pitch angles is depicted.

adjusted accordingly as otherwise errors of the scan vector angle as depicted in Figure 2.14 are to be expected.

2.7.3 Accuracy Assessment

The accuracy of the orthorectified results can be assessed using various approaches. The most common method is the consideration of residuals from independent GCPs [44] while correlation analysis of the results to reference images is a more robust approach [13].

In summary, the accuracy of the parametrically geocoded imagery is assessed using four approaches:

- calculate the location residuals of ground control points which were not used for the prior calculation,
- correlation analysis of the results to the DEM based illumination map (as required for atmospheric correction),
- correlation analysis of digital maps to the geocoded results,
- correlation analysis of co-registered imagery taken at different days (compare Section 3.8.2 on page 64).

The spatial correlation analyses are made based on a regular sampling grid between the result image and a reference image of exactly the same resolution and extent. The herewith retrieved accuracies are attributed directly to the accuracy of the engaged input data since the algorithm itself consists of rigid analytical expressions. PARGE currently supports such a method for correlation analysis between a processed and a reference image of the same dimensions. For further analysis, it is recommended to use standard imaging processing software (e.g., PCI, Erdas, ENVI) for such analyses.

Chapter 3:

Typical Workflow

The process workflow of PARGE had originally been implemented based on the requirements of well-known hyperspectral sensors such as AVIRIS [71], HyMap [12], or DAIS 7915 [11]. In a later stage, the capability for operational processing of modern pushbroom sensors such as AISA, CASI, Hyspex, or APEX has been added. A major goal was to create an interactively usable application workbench with a full set of flexible features between raw input data and image output. The interactive elements are required due to the experimental character of certain data formats and the mediocre quality of some of the currently available INS/IMU data. If auxiliary data quality is not an issue, all computing steps are available in a scripting environment for integration in fully operational processing chains (see the detailed description in Chapter 4).

The knowledge about the status of processing is very important to keep track of the ongoing work. It is therefore given special attention in the first section below. Typical workflows and rules for selection of workflow variations are given thereafter in the subsequent sections.

3.1 Geocoding Status Management

A geocoding report including all relevant variables is used to view and store the information on the status of the geocoding efforts (**>File:Save/Show/Restore Status<** [p.146]). This function is of high relevance for the management of the processing workflow.

3.1.1 The Geocoding Status

The status saves all available information about the current state of the geocoding session and can be saved at any time using the function **>File:Save Status<** [p.147]. It includes all parameters as described in Section 4.1 on page 69, where the internal data format is given in detail. Its contents are divided in the following topics:

- image data definitions,
- sensor model definition,

- DEM data definitions,
- auxiliary data definitions,
- result and processing options definitions.

For status handling the corresponding functions in the **>File<** p.125 menu should be used. Additionally, PARGE automatically saves a status file (`parge_temp.gcs`) for security reasons and to make the command **>Edit:Undo<** p.151 functional (note: this file may be used to recover PARGE sessions after system failures).

Two complete images are temporarily held in system memory: one single band and the DEM. Since the calculation of their statistics may require some time, this function is only optional within the display of the status. These two images are restored to memory if **>File:Restore Status<** p.146 is chosen in the file menu. A report can be displayed which shows the meta-data available in the geocoding status (see Figure 3.1) - the displayed ASCII text is not the status itself but rather an informational description.

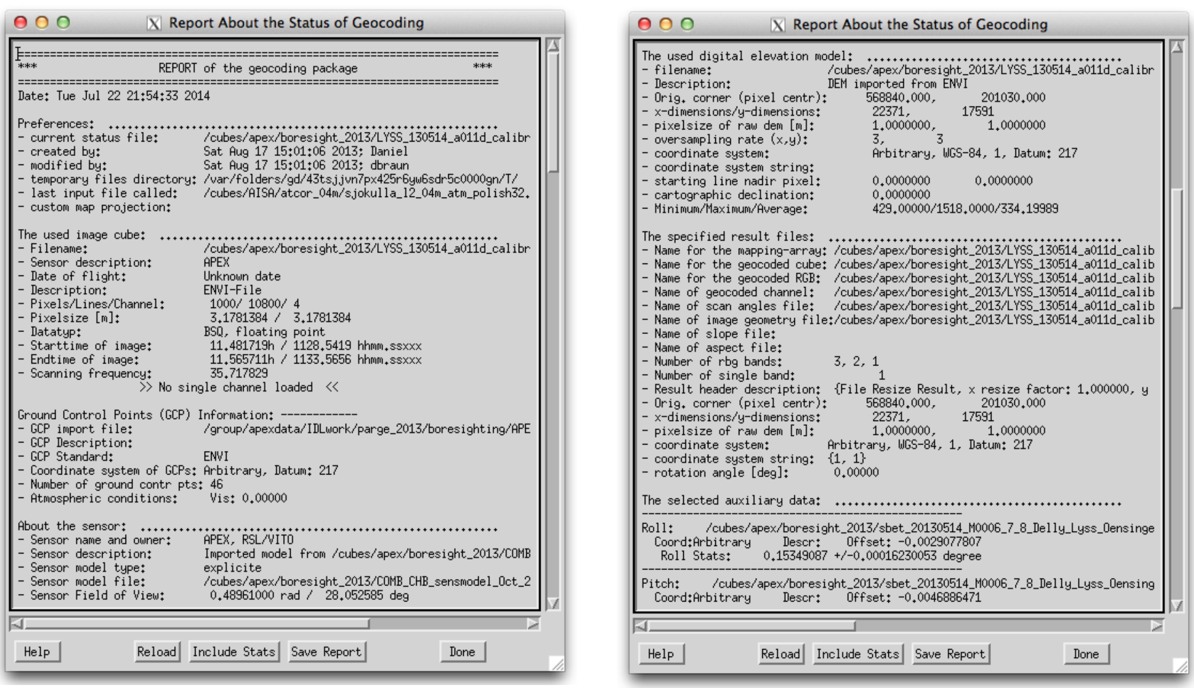


Figure 3.1: Report about the geocoding status as displayed by >File:Show Status<.

3.1.2 Accessing the Status from IDL

For users having access to an IDL/ENVI developer license, the status parameters can be made available by using the "`restore`" command on the IDL prompt:

```
IDL> parge,/norun
IDL> restore, "your_status.gcs"
```

Invoking this command, all variables and structures as stored from within PARGE will be available within the IDL session. The variable names and definitions are given in Section 4.1 on page 69.

3.2 Processing Workflow

3.2.1 Overview

Although PARGE was developed for a number of specific sensors, it is easily adaptable to a generic sensor type. The workflow structure for such processing is depicted in Figure 3.2. The first steps of the workflow are concerned with the data preparation of image, sensor, auxiliary, and DEM data. Interactive viewing and filtering capabilities support this section of the process.

The boresight alignment is a procedure which usually is done once for each data take to account for the absolute orientation of the sensor in the aircraft. A number of GCPs is required for validation and recalibration of the auxiliary parameters. The procedure to perform this recalibration has already been described in Section 2.4 on page 32, whereas its implmentation details are given in Section 3.5 on page 56. The boresight alignment procedure is necessary for all currently available airborne imaging spectrometry systems and is therefore part of the 'standard' processing flow.

In practice, some of the auxiliary data may be faulty or missing at all. Such parameters can be interpolated using GCP-based algorithms , as long as a good estimate of the remaining parameters exists. Such interpolations need an increased number of GCPs in the range of 1 GCP per 100-200 lines (for optimal quality). Section 3.6 on page 59 describes these algorithms.

A consistency check concludes the data preparation, giving clues to missing or non-compatible parameters and data. The whole auxiliary data status can now be saved for documentation and later re-iteration of the auxiliary parameters preparation. The main processing algorithm is the next step. It calculates the geometry for each image pixel and stores the results either in DEM geometry ('*_map.bsq') or in acquisition geometry ('*_igm.bsq'); compare Section 3.2.3.

The effective production of geocoded image maps is the last step to be performed. The mapping array or the IGM are directly applied to the original image data. This process is used to create RGB images for visualization and accuracy assessment as well as for full orthorectified image cubes. The mapping array or the IGM may also be applied to processed thematic results in order to avoid the large memory overhead of geocoded imaging spectrometry data.

3.2.2 Processing Steps

A number of common steps is to be done for a generic processing. The number of required steps decreases significantly as soon as PARGE-supported sensors are used. A complete list of potential steps is given below. Before using the PARGE software, the input data shall be prepared and compiled. A complete input data sets contains the following entitites:

- image data cube (should be in ENVI or in raw binary format),
- geometric sensor model (pushbroom/whiskbroom/FOV/explicite)
- digitial elevation model or ground altitude for flat areas,
- inertial navigation system auxiliary data (sensor specific INS Files),
- flightpath DGPS converted to ortho-metric coordinate system, and
- ground control points (about 10 per scene) or boresight alignment information/angles.

The data of the DEM, the DGPS flightpath and the GCPs should have the same geodetic reference (projection, geodetic datum).

a) Importing the scanner data

- **>File:Import<** p.131: Command for the import of standard imagery and INS data of supported sensors. If auxiliary files are partly missing, an error may be displayed and the data will only be partially available, or
- **>File:Define Raw Data Cube<** p.152: reads a generic image data cube if no standard import filter is available.

The raw imagery is preferably transformed to a band sequential (BSQ) data format in the beginning to ease further geometric processing – only limited support for BIP imagery is included in PARGE whereas BIL is not supported. One single band is read and may displayed to check the correctness of the data definition using **>Edit:Display Single Channel<** p.155.

b) Checking the auxiliary data

The auxiliary data is checked by using one or more of the functions:

- **>Auxiliary:Plot Image Area<** p.192
- **>Auxiliary:Plot Flightpath<** p.193.
- **>Auxiliary:Plot Attitude<** p.193,

If some of the auxiliary data streams looks strange, e.g. unsteadiness, outliers, noise, these effects can be corrected by using **>Auxiliary:Filtering<** p.190.

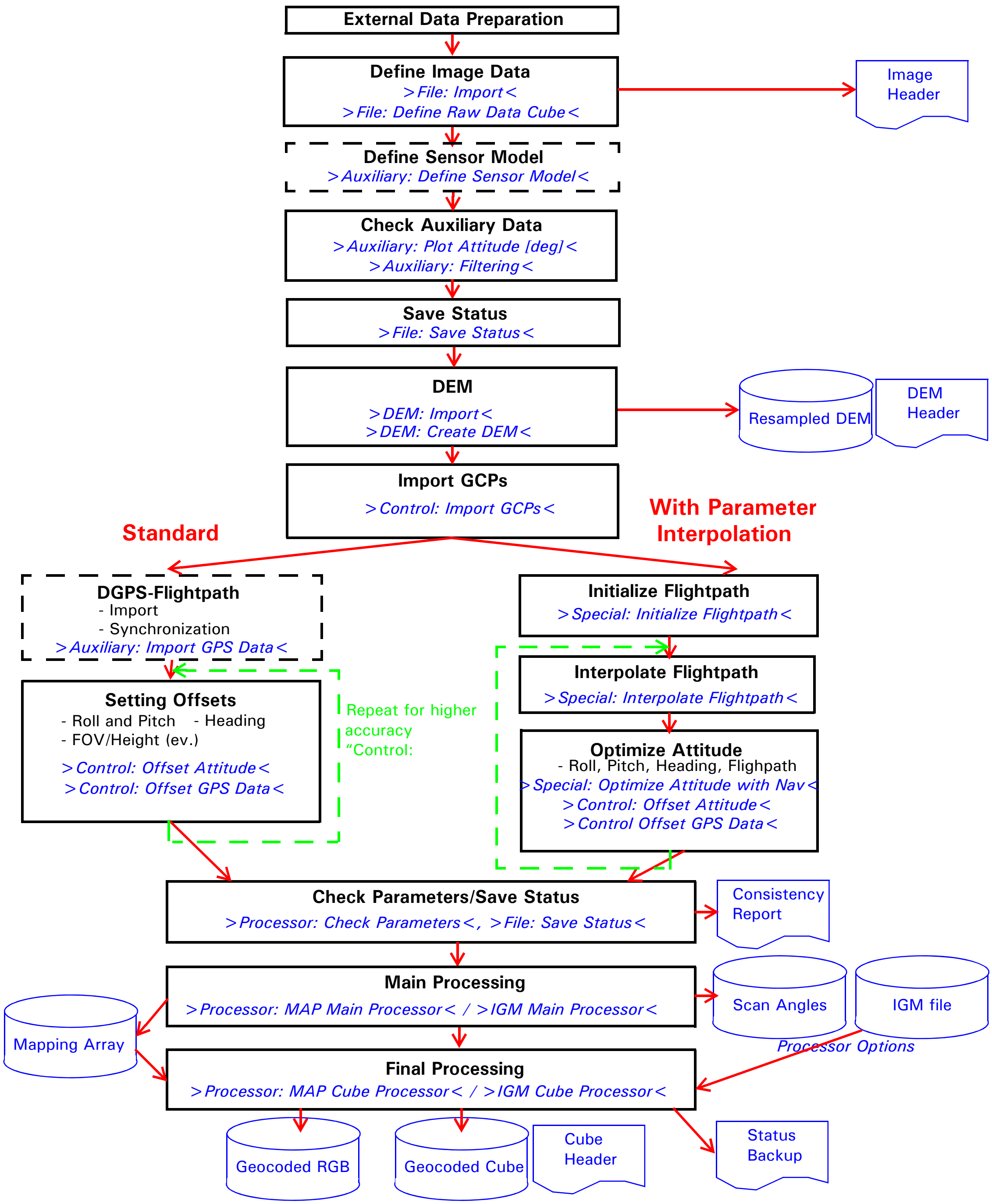


Figure 3.2: Workflow for parametric geocoding: standard and with interpolation of missing parameters.

c) Saving the current status

The status helps tracking the work. It can be saved and restored at any time of your work. (see Section 3.1). The extension for saving PARGE status files is '*.gcs')

d) Defining the DEM

A raster digital elevation model can be read from raw, ENVI, PCI, ARC GRID, GEOTIFF or other file formats (**>DEM:Import<** [p.163]). If no DEM is available, it has to be created using the available GCPs or a flat area has to be assumed (**>DEM:Create DEM<** [p.170]). Afterwards, it is cut and resampled to the desired dimensions of the final image using **>DEM:Resize DEM<** [p.174]. A security range of about 50 pixels on each side of the image borders is recommended for optimal performance of the ray tracing procedure.

e) Input GCPs

If a boresight alignment or parameter interpolation shall be performed, the ground control points have to be read from a PCI segment, an ENVI '*.pts' - file, or from an ASCII format file (**>Control:Import GCPs<** [p.196]). Alternatively, they may be defined from within PARGE. They must be made available in the same coordinate system as the DEM and the sensor GPS information.

A third option to define GCPs is to use the function **>Control:Auto GCP <** [p.203] to determine GCPs automatically by matching analyses to a digital orthophoto

f) Setting the offsets for roll and pitch and heading

These offsets (boresight angles) should be set if they are not absolutely calibrated **>Control:IOffset Attitude<** [p.204]. Many of the current INS systems are not able to provide attitude data at absolutely calibrated accuracy which would be required for pixel accuracy. Thus, these offsets have to be set at least once per flight or even per scene. If the alignment once is calculated, it may be inherited to subsequent flights of the same configuration using the function **>Control:Import Offsets<** [p.208].

g) GPS data and altitude offsets

If there is still no trust in the data, the offset to aircraft altitude or even its position can eventually be set using **>Control:Offset GPS Data<** [p.207].

h) Checking the parameters

If no major error is displayed in the **>Processor:Check Parameters<** [p.231] procedure, all parameters have been provided consistently and the current status may be saved for documentation purpose. For higher accuracy the steps e) to g) may need to be repeated.

i) Running the main processor

Calculates the geometry and resamples the output to the DEM grid using **>Processor:IGM Main Processor<** [p.237] or **>Processor: MAP Main Processor<** [p.246]. Various resampling options are available within these steps which are described with the respective modules. For later postprocessing, scan angle file or image geometry map may be produced as a side output for this step.

j) Creating orthorectified results

Fully geocoded cubes may be created for further use in models and GIS systems using **>Processor:IGM Cube Processor<** [p.238], **>Processor:IGM Rotation Processor<** [p.240], or **>Processor:MAP Cube Processor<** [p.249]. For visualization purposes, the creation of single band or RGB TIFF/JPEG files is supported in **>Processor:Process Band or RGB<** [p.249].

k) Creating a mosaicked output

The georectified outputmay be merged into one file by the function **>Processor:Integrated Processing<** [p.243] (in case the data came in two separate files). Mosaicking of an arbitrary number of scenes can be accomplished using the function **>Processor:Mosaicking<** [p.256].

3.2.3 Resampling Workflow

The resampling of hyperspectral imagery is an issue which is discussed conversely. All current image processing systems rely on data which is resampled to a regular grid having constant pixel sizes in both spatial dimensions. Resampling in PARGE may be done at two places: in the main processor, a resampled map may be created for later (fast) nearest neighbor resampling, whereas in the final processor, the radiometric values may be resampled directly to map geometry.

Two distinct workflows are available in PARGE which handle the resampling completely differently. The first (original-traditional) workflow resamples the radiometric data directly to target geometry in an early stage of processing. Vice versa, the acquisition geometry based workflow tries to avoid any resampling of spectral data as long as possible, targetting a resampling of final data products only. Table 3.1 gives an overview of the main differences between the two workflow approaches.

Table 3.1: Differences between acquisition and map geometry based processing

	image/acquisition geometry	**DEM/map geometry**
DEM	resampled to image	resampled to map resolution
Spectra	original/uniform	resampled or relocated duplicated and/or omitted values
Statistics	distorted statistics	area-dependent spatial statistics
Memory	acquistition memory	up to 5x original memory required
Geometry	image geometry map (IGM)	Cartographic mapping array (MAP)
Resampling	Nearest Neighbor, Natural Neighbor, Bilinear Interpolation	Nearest Neighbor, Bilinear Interpolation, Nearest Bilinear, Interpolated Gaps
Output Resolution	arbitrary output resolution, rotation, and image size.	output ruled by DEM resolution and size.

a) Acquisition geometry (IGM processing)

In this workflow, the coordinates are written to the image geometry map and no resampling is done in the initial processing. Therefore, the original spectra are preserved. Also scan angle map is stored in acquisition geometry. Georectified results may be produced by interpolation to an arbitrary output grid, including rotated outputs. When proceeding to radiometric processing or atmospheric correction, the DEM related layers need to be transformed to the acquisition geometry using the function **>Processor:IGM Inverse Processor<** p.242. Only final results suitable for an end user are to be transformed to map geometry.

b) Cartographic geometry (MAP processing)

First, the image pixel coordinates (original pixel and line number) are written to an array in DEM geometry at their geocoded position what results in a 'mapping array' which is stored as main output of the processing. In parallel, the viewing geometry per image pixel is saved to a separate file which is described in detail below. It to be remembered, that the data product will have exactly the same spatial format as this DEM (a smaller DEM may thus be used to process a subset of the image only). The information loss in the geocoding process is minimized by resampling the DEM (defining the final image) to a slightly higher spatial resolution than the nominal resolution of the original image data.

3.2.4 Outputs

A number of outputs are created in the course of the processing workflow:

- *Image Header:* An ENVI™ format header to the raw imagery if none has been available.
- *DEM:* The resampled and resized DEM is stored in ENVI™ format.
- *Consistency Report:* An ASCII report about the consistency of the auxiliary data to the image.
- *Status Backup:* Status information of the geocoding session.
- *Mapping Array:* A two-layer array containing the raw pixel and line coordinates for each DEM element.
- *Image Geometry Map (IGM):* A two-layer array containing the exact x/y coordinates for every original pixel.
- *Geocoded Cube:* Ortho-rectified hyperspectral image in ENVI™ format.
- *Scan Angles:* Data layers with scan zenith, scan azimuth angle, and the absolute distance from aircraft to the pixel for each geocoded pixel.

The latter provides all required linking layers for later radiometric processing of the geocoded image. Parameters which only depend on the DEM such as slope, aspect, and elevation are derived separately. A more detailed description about the PARGE outputs can be found in Section 4.1.3 on page 79.

3.3 Importing Standard Sensor Data

A growing number of hyperspectral sensors are supported by PARGE with dedicated filters and procedures. Additional sensors can be added in the future as soon as specific requests from data providers or users arise. In principle, any kind of line scanner can be introduced into PARGE as long as its data format and the technical specifications are available. Below, only three exemplary sensor import workflows are described. The similar principles and conventions as described for these sensors apply for any other system.

3.3.1 AISA

The Finnish company ‘Specim’ (www.specim.fi) has developed the AISA airborne imaging spectrometer system, which has become popular for operational hyperspectral remote sensing. AISA is pushbroom-type imager and therefore, some specialities need to be kept in mind:

- There are three types of AISA systems on the market - PARGE principally could handle all of them - currently the AISA eagle and the AISA+ system are implemented through the GUI (any other AISA type instrument will be included upon request).
- The geometric sensor model is calculated as standard pushbroom model based on the fore-optics focus width (to be selected).
- The 1000 across track pixels of the AISA Eagle system and its high scan frequency lead to high memory requirements. Therefore the acquisition geometry based workflow is recommended for this type of sensor.

More details about AISA import options is given in **>File:Import:AISA<** p.132.

AISA

3.3.2 AVIRIS

The command **>File:Import:AVIRIS<** p.134 is an interface to read most of the various AVIRIS data formats as they have been delivered by JPL up to now. Currently, formats until 2003 are supported. Note that the formats in 1999 and 2000 may need some conversion prior to the data import.

There are various points which need to be considered while working with AVIRIS data:

- It is recommended to transform the raw file to BSQ format first to achieve better performance. BSQ format can be produced directly and efficiently through the procedure built into the PARGE application (**>Special:Convet BIP to BSQ<** p.211). The name of this file (without extension) must be the same as for the auxiliary data files.
- The selected year of data distribution must be the year when JPL processed the data and delivered it. This is quasi independent of the year the data has been taken!
- Starting in 1998, low altitude flights were carried out with differing parameters from the

high altitude data.

- Original raw data always is in 16 bit 'raw unsigned integer'. Anyhow, after atmospheric correction or other transformations 'signed integer' values may be created.
- The parameters should be smoothed by default factors for old data acquisitions in order to remove noise in the auxiliary data stream.
- AVIRIS data are normally delivered as rectified imagery, using a proprietary software [8] – such data is not suitable for PARGE. Please ask for the raw imagery recorded in acquisition geometry (614/677 pixels across track).

Please contact the data provider, if you can not find the appropriate auxiliary data files with your data.

3.3.3 HYMAP / PROBE-1

The commands **>File:Import:HYMAP<** p.139 **>File:Import:Probe-1<** p.144 provide interfaces to read the HYMAP and PROBE-1 dataformats (which are very closely related, stemming from the same manufacturer). The filters are based on the formats as delivered by HYVISTA (South Africa), DLR (Germany), and Earth Search Sciences (USA) starting in 1998 for various worldwide campaigns.

Some remarks about those sensors (more detailed information about the import procedure is given on page 139):

- The raw data cube is assumed to be in BSQ format.
- Default smoothing: the procedure smoothes all parameters by default factors. This option can be switched off for inspection of the raw data.
- Maybe some of the auxiliary data files have to be renamed to import them properly (they only should vary by the extension).
- The application opens the HYMAP file first and assigns the number of lines autmatically (even if no ENVI header is available).
- This procedure has been first developed on Integrated Spectronics Hymap data only (as delivered by Hyvista) and a specific import filter for the PROBE-1 instrument has been included, while other Hymap-type instruments such as ARES may be added later.
- If the image is inverse to the flight direction (right-to-left scanning), the FOV can be set negative to solve the problem (**>Auxiliary:Define Sensor Model<** p.182).
- The auxiliary file *.log (if required) needs to be of original format and has to be named like the raw image file.

3.3.4 HYSPEX

Hyspex is a modern and well calibrated imaging spectrometer distributed by the norwegian Norsk Elektro Optiks. Its standard data may be imported using the function:

>File:Import:HYSPEX< [p.142]. Details see there.

3.3.5 Other Sensors

Additional sensors have been included into PARGE (ie., MIVIS, ROSIS, DAIS, AIS, CASI) while further sensors will be added on demand. Please refer to the respective online help sections or the specific instructions given by the PARGE provider for detailed instructions how to use the software with your specific sensor. However, principles and some of the details remain the same as for the above described exemplary imaging spectrometers.

Attention: The data format, even of supported sensors are subject to change on a yearly basis. If a PARGE import filter fails on your data set, please do not hesitate to contact technical support and prepare a complete test data set to allow an update of the import filters.

3.4 Importing Auxiliary Data

3.4.1 DEM

For orthorectification, the digital elevation model (DEM) is an essential basis. The projection and geodetic datum of the DEM define the geodetic reference of the geocoded result. In the map based workflow, the dimensions of the DEM define the size of geocoded outputs. The DEM is stored as an ENVI formatted file and is read to memory completely for.

Details

- The major DEM formats as supported by PARGE are ENVI or PCI format DEM (must be a DEM in a metric orthographic coordinate system), but also GEOTIFFs and ARC GRID format may be imported.
- Units of original data are ‘m’ or ‘dm’; if the unit ‘dm’ is chosen, the data is converted to floating point meters.
- Center referencing (all coordinates at pixel center) is the default for and within PARGE, corner point referencing of pixels (such as used in ENVI) are thus converted.
- The origin corner coordinate are the lowest coordinates in x and y direction, corresponding to the referencing type.
- A pre-defined or arbitrary projection can be set manually (for information purpose mainly). The arbitrary projection can be set if the real world projection is unknown but all geometric data are given in consistent coordinates.

3.4.2 DGPS Flightpath

The aircraft navigation (location) data are usually imported and optionally synchronized on the image lines with the standard import filters. Again, it is important that the coordinates are in the same geodetic system (projection, ellipsoid, geodetic datum) as the DEM and the Ground Control Points, both in position and height! The output will be a synchronized flightpath in the variable *navarr*, while the uncorrected raw GPS data are kept in *gpsarr* (if available).

If no DGPS flightpath is available it can be interpolated using the GCPs (see **>Help:Flightpath Troubles<** and Section 3.6 on page 59). The GPS data may be converted to the local coordinate systems from within PARGE or externally as decribed in Section 3.4.3 below.

Synchronization

There are three possibilities of synchronization (depending on the input data):

- Synchronization using *image start time* and *scan frequency*:
  start time: This is the absolute time of first image line (same time system as for DGPS flightpath).
  scan frequency: This is the average frequency of the whole image scan.

- Synchronization using respective time for each image line: In this case the PARGE variable *syncarr* is used for synchronization. It has to be known from the standard import filter for the data set.
- Synchronization by line number: The auxiliary data is already tagged by an absolute line number corresponding to the number of image lines, which is used directly for synchronization.
- Post-Synchronization: if the data is out of sync, the post-synchronization option in PARGE may be used to optimize this parameter.

3.4.3 Remarks on Coordinate Conversion

PARGE currently supports the conversion of geographic WGS-84 Longitude/Latitude to a few coordinate systems including UTM (see Section 5.1.6 on page 124 for an overview of this procedure). Any unsupported coordinate conversion needs to be pre-defined according to IDL conventions (**>File:Define Map Projection<** p.160) or it may be calculated using external dedicated conversion tools.

The following workflow is recommended if external coordinate conversion is preferred:

1) Import the raw imagery by standard filters, but choose >NONE< as coordinate conversion option.
2) If the GPS information comes in a separate file which is not covered by the standard filter, use the function **>Auxiliary:Import GPS Data<** p.186 to import and possibly synchronize the GPS data file.
3) Plot the GPS navigation data by **>Auxiliary:Plot Raw GPS<** p.192 to make sure that the raw GPS information it is in Longitude/Latitude geographical coordinates (should be in decimal degrees now, see also the variable *gpsarr*)

4) Export the navigation data using the function **>File:Export:Raw GPS<** p.149. An ASCII file containing three columns is created. First column: x-coordinate (longitude, deg), second column: y-coordinate (latitude, deg), third column: aircraft altitude (meters).
5) Convert this ascii file externally to your metric coordinate system of choice using ENVI (Function: Utilities > Map Projection Utilities > ASCII Coordinate Conversions), Blue Marble's geographic calculator (*http://www.bluemarblegeo.com*), the SeisSoft Coordinate file converter (*http://www.connect.net/jbanta*) or any other coordinate conversion tool.
6) Make sure the output of the procedure comes in three columns, ASCII, meters
7) Import the converted coordinates through the function **>Auxiliary:Import GPS Data<** p.186 function back into PARGE
8) Check if the calculated flightpath fits to your DEM (e.g. by plotting it or by using the Function **>Control:Display Flighpath on DEM<** p.202.)

Attention: The conversion of the flightpath coordinates is one of the major errors sources in parametric geocoding. Please make sure that you have the correct information about your local coordinate system available (i.e., projection, ellipsoid, local datum) before converting the data. The raw GPS information usually comes in generic geographical coordinates based on the WGS-84 ellipsoid - anyhow, a different ellipsoid may have been set during recording, or some preprocessing may have been applied to the original recordings.

Typically x/y offsets of 100-1000 meters may occur, if the coordinate conversion was incorrect. This may be cross-checked with the GCPs to avoid systematical errors. Before applying high x/y-offsets to the navigation data, make sure its conversion has been correct.

3.4.4 Attitude Data

Reading and preparing the attitude data allows to read roll, pitch and heading with the same synchronization options as described in Section 3.4.2. The perfect synchronization is even more important for attitude data due to their high temporal variations.

The most common way to import the attitude data into PARGE is by using the supplied import filters for the supported sensor systems. If no filter is available, an user interface is provided to import the data for each of the three attitude data streams from separate data files under the function **>Auxiliary:Import Attitude Data<** p.184.

Handling Attitude Problems

It is possible that there will appear some problems working with attitude data. In the following, a few details are discussed if you are in troubles with those data:

a) Heading

The true heading is defined as airplane nose direction per image line, where north is zero degree and East is 90 degree. Its average should be coincident with the flight direction.

- **heading offset correct:** **>Control:Offset Attitude<** p.204 will correct the heading. Offsets are given in radians and calculated from a series of ground control points which optimally are distiributed across track. Theoretically, a minimum number of 2 GCPs is required to calculate a heading offset. In practice 6 to 15 GCPs should be used to achieve accurate averaged results. The ground control points should be well off nadir for good accuracies.
- **no heading available in auxiliary data:** The heading can be calculated as first derivative of the flightpath. The function **>Special:Prepare Heading<** p.216 will do such a calculation. This function requires the navigation data to be already read and transformed to metric coordinates. If there is also a yaw (angle between airplane nose and effective flight direction) defined at this point, that will be added to the flight direction – one of the rare situation where yaw is used in the PARGE process.
- **heading and flight direction differ by large amount:** There are two possible reasons for this mistake: 1) the heading data is offsetted by a factor of 2*pi or has negative values. 2) the heading definition is wrong in the auxiliary data; in this case it has to be calculated from the flightline as described above.
- **heading and flight direction differ by small amount:** The heading parameter may have been defined to geographic north and should be converted to cartographic map north when importing. This may have been wrong and needs to be corrected by improving the import filters or by offsetting heading using the coordinate conversion function `gc_coordf` as described on page 101.
  If a magnetic heading mesaurement is performed it even might be that the magnetic declination may not have been considered. It has always to be remembered that for most parts of the american continent, the declination is about 10-15 degree between magnetic and cartographic north. Such differences may also occur if there were high side winds during data acquisition or if the sensor was not aligned to north correctly after installation in the airplane.

b) Roll:

A standard roll parameter should vary in a range of +/- 5 degree with an average at zero. If the variations are higher, the image should show heavy distortions. If the average differs strongly from zero (above 5 degree) there may be a tilted sensor situation or the navigation coordinates are wrong in roll (across track) direction.

- **roll offset correction:** **>Control:Offset Attitude<** p.204 can be used to correct the roll. Offsets are calculated as average from a series of ground control points. One single ground control point per image scene would be theoretically enough to correct this offset.
- **roll creation:** The function **>Special:Interpolate:Roll/Pitch<** p.218 offsets the roll (and pitch) by an interpolated function through the roll-offsets of a number of selected Ground Control Points. The flag of these Ground Control Points must be '2' meaning 'For Offset

and Interpolation Use'. The other Ground Control Points afterwards may be used for validation purposes.
A second option to derive a missing roll parameter is through image-based roll determination in >Special:Roll Compensation< [p.219].

- **drift in the roll data:** For certain instruments, the roll gyros may drift along track. This effect can be corrected by the function as described above, but using only a small number of Ground Control Points (2-4) with one at the beginning and one the end of the flightline (>Special:Drift Correction< [p.218]).

c) Pitch:

An usual pitch varies in the range of +/- 3 degree with an average at zero to 5 degree. If the variations are higher, the image should show heavy distortions. If the average differs strongly from zero (above 5 degree) there may be a tilted sensor situation or the navigation coordinates are wrong in pitch (along track) direction. In such cases, the sensor model may need adjustment in >Auxiliary:Define Sensor Model< [p.182]instead of putting too large offsets on pitch.

- **pitch creation and correction:** The description for roll correction above can be applied analoguous for the pitch data.

d) Yaw:

A correct yaw is usually not required for rectification. It only may be used for derivation of the heading if necessary. Thus, there is no problem if you enounter weird yaw values.

3.4.5 Sensor Model

Normally, the geometric sensor model (i.e. the sensor's interior orientation) is defined analytically from FOV, sensor type (whiskbroom or pushbroom), and number of across track pixels (see Section 2.2.3 on page 19). The sensor model may also be read explicitely from external files which give the angular position for each across track pixel with respect to the main geometrical sensor axis (see function >Auxiliary:Define Sensor Model< [p.182].

Note that the geometric optical axis is defined as the axis perpendicular to the absolute horizontal layer as given by the INS system.

3.5 Working with Ground Control Points

Using PARGE, you are able to work with ground control points (GCPs) in a similar way as they are engaged in non-parametric geocoding applications. The points are used to calibrate the geometric parameters to the geometric model as described in Chapter 2. It uses GCP statistics to calculate individual offsets for roll, pitch, or heading independently. The GCP coordinates are ray traced backwards to the (theoretical) aircraft position which results in an offset to the real aircraft position. Statistical and analytical calculations afterwards lead to the offsets of the individual parameters.

In the following sub-sections, some background information about the philosophy behind this 'semi-parametric' approach is given. The descriptions of the individual function can be found in Chapter 5.

3.5.1 Importing, Creating, Exporting GCPs

Using the PARGE functions, the Ground Control Points (GCPs) can be imported, edited and exported for further offset calculations. A minimal number of GCPs is required as soon as the offsets of the airplane attitude data are unknown (e.g. unknown cartographic north or horizontal direction). A table of all GCPs can be displayed, where single points can be edited, added, and deleted manually.

Various import features are supported in **>Control:Import GCPs<** p.196:

- PCI GCP segment (recommended tool for GCP collection is PCI's GCPworks),
- ENVI GCP file (collected within ENVI and exported: Save GCPs w/ map coords).
- PARGE text file, saved from within PARGE, e.g. in another session

The GCP coordinates have to be converted to the same geodetic system (projection, and geodetic datum) as the DEM and the flightline, both in position and height. The list of the ground control points is stored in the PARGE variable *gcparr*.

It's principally enough to take one single accurate GCP to estimate either x/y or roll/pitch-offset values. More GCPs are required, if heading or aircraft altitude offsets have to be estimated. Moreover, one may want to have additional GCPs as independent control points. In the special case of flightpath reconstruction, more GCPs are necessary, distributed evenly over the image scene.

3.5.2 Searching GCPs using ENVI

For an accurate GCP collection using ENVI, the following procedure is recommended (please refer to the help about image to map registration in Envi):

- open the cartographic reference image and the raw image side by side,
- choose in ENVI “Map:Registration:Select GCPs - Image to Map”,
- Select the non-georeferenced image as the Input Display and enter the target projection information (this projection is not relevant to PARGE at this point; choose ‘Arbitrary’ if you’re unsure). Click OK.
- the ‘Ground Control Points Selection’- window appears.
- In the Image window of the georeferenced image, right-click and select Pixel Locator to open the Pixel Locator dialog

Now you may start searching for GCPs:

- select a point in the map,
- push the 'Export' for the location to appear in the GCP Selection window,
- select the corresponding point in the image, and
- push the 'Add Point' button.

After repeating this for all points, use the function “File:Save GCPs w/ map coordinates'” from the GCP Selection panel to export a ‘*.pts’ -file. This file may be imported to PARGE afterwards.

3.5.3 Attitude and Navigation Offsets

Two major tools are provided by PARGE for offsetting attitude and navigation data. They allow for calculation of offsets for all parameters based on the Ground Control Points as described in Section 2.4 on page 32. The functionality is depicted in Figure 3.3. A list of gcps is displayed, while offsets for both navigation and attitude parameters are calculated.
A typical workflow if using GCPs may work as follows:

1) A set of Ground Control Points is imported or edited. The points can be defined by **>Control: Import GCPs<** [p.196] or **>Control: View and Set GCPs<** [p.200]. The standard remote sensing image processing packages offer valuable tools for GCP collection. It is thus recommended to search the points externally and import them.

   Attention: PARGE references all GCPs and the DEM by the center points of every pixel, starting at position (0,0). Be aware of potential shifts, if an external GCP list is used.

2) The position of the GCPs on the raw image may be checked using the function **>Control: View and Set GCPs<** [p.200].

3) Using the function **>Control: List GCPs<** [p.198], the list of all GCPs with the respective residuals is displayed.
   A flag is set for each GCP, which may be changed between:
   0: excluded (only marked, not deleted),

1: use for offset calculation, but not for parameter interpolation,
2: use for both offset calculation and flightpath reconstruction or parameter-interpolation.

4) The statistics of all GCP residuals are now used for parameter offset determination using the functions **>Control:Offset Attitude<** p.204 and **>Control:Offset GPS Data<** p.207 . Usually, first the three attitude parameters are offsetted iteratively (using the 'Optimize All' function). Second, the aircraft altitude may be offset.

Attention: The X/Y coordinates of the flightpath should only be offsetted, if no DGPS flightpath was available or it is assumed that the flightpath has to be shifted (for example, due to uncertainties in projection/datum transformation of the DGPS coordinates).

The heading offset is calculated based on the selected Ground Control Points. An optional heading offset is related to the correlation of pixel-nadir distance and pitch offsets of the GCPs. This correlation is minimized iteratively while adjusting the heading offset. The aircraft

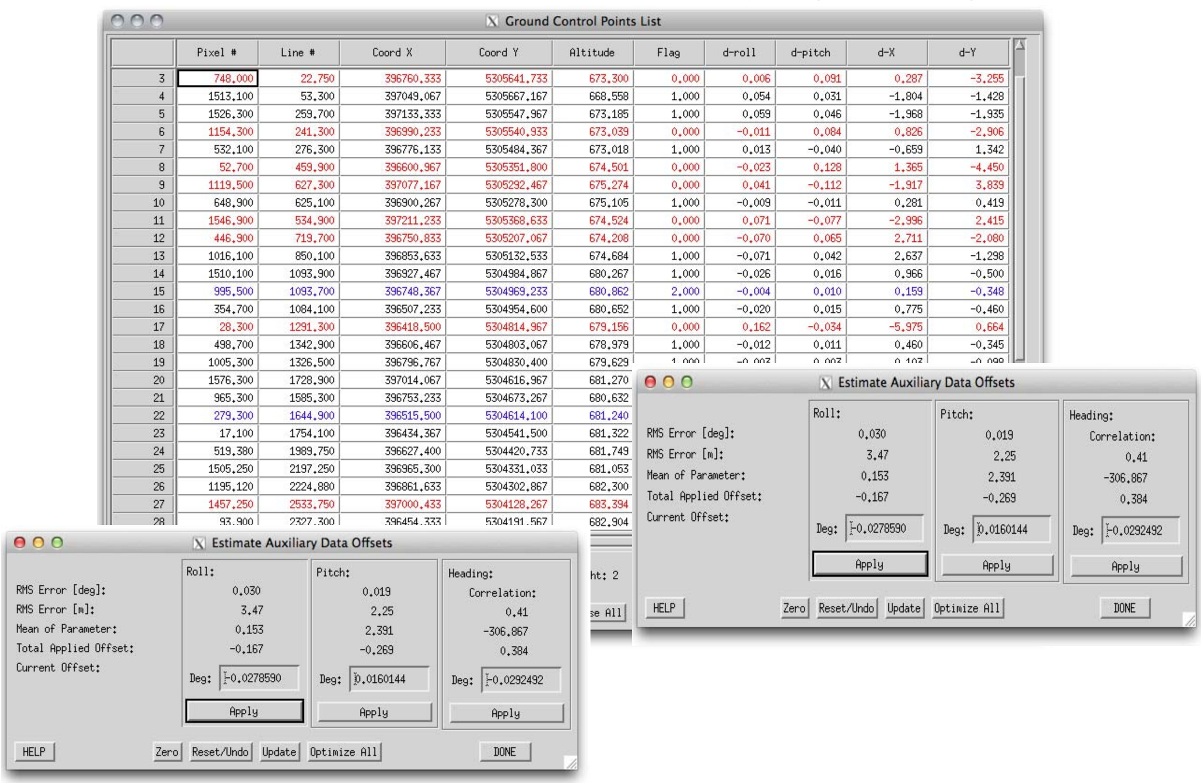


Figure 3.3: Auxiliary Data Offsets Tools.

altitude offset is related to the correlation of pixel-nadir distance and roll offsets of the GCPs. This correlation is minimized iteratively while adjusting the altitude offset. Moreover, the height offset is directly related to the FOV offset (geometrically). Hence, the sensor model may need adjustment if systematic altitude offsets are observed throughout all data of a sensor system.

3.6 Auxiliary Data Interpolation

If one of the core parameters is missing, a reconstruction procedure on the basis of GCPs is required. A number of iterations between parameter interpolation and the offset algorithms (up to 5) are required to converge to a satisfying solution. Cubic spline interpolation is preferred for the flightpath reconstruction process due to the continuous characteristics of the flight. On the other hand, the discontinuous angular roll and pitch movements of the aircraft are better approximated by linear interpolation. Heading and altitude variation can not be easily interpolated using such algorithms since their respective offsets both can only be approximated if two or more GCPs are available for a quasi–stable number of contiguous scan lines.

Using a spline interpolation and a number of Ground Control Points the flightpath can be approximately reconstructed (see Figure 3.4, function **>Special:Interpolate Flightpath<** p.214). Prior to interpolation an initial flightpath is required (Function **>Special:Initialize Flightpath<** p.213 . The output will be an updated flightpath (*navarr*) and an updated Ground Control Point-List.

Details

- By calculation and plotting a new track, the flightpath is calculated and a plot of the previous and the new flightpath against ground control point positions is given.
- After updating, the navigation data is updated to the currently calculated flightpath – original navigation data is lost after applying this function
- An initial, ideally straight, flightpath is required to start the iterative interpolation process.
- Parameter offsets are to be calculated prior to any interpolation. After interpolations have been applied, the parameters should no longer be changed.

PARGE also offers the interpolation of roll or pitch parameters if one of them may be missing in the dataset. A similar approach as for the flightpath is used in that case using the function **>Special:Interpolate Roll/Pitch<** p.218.

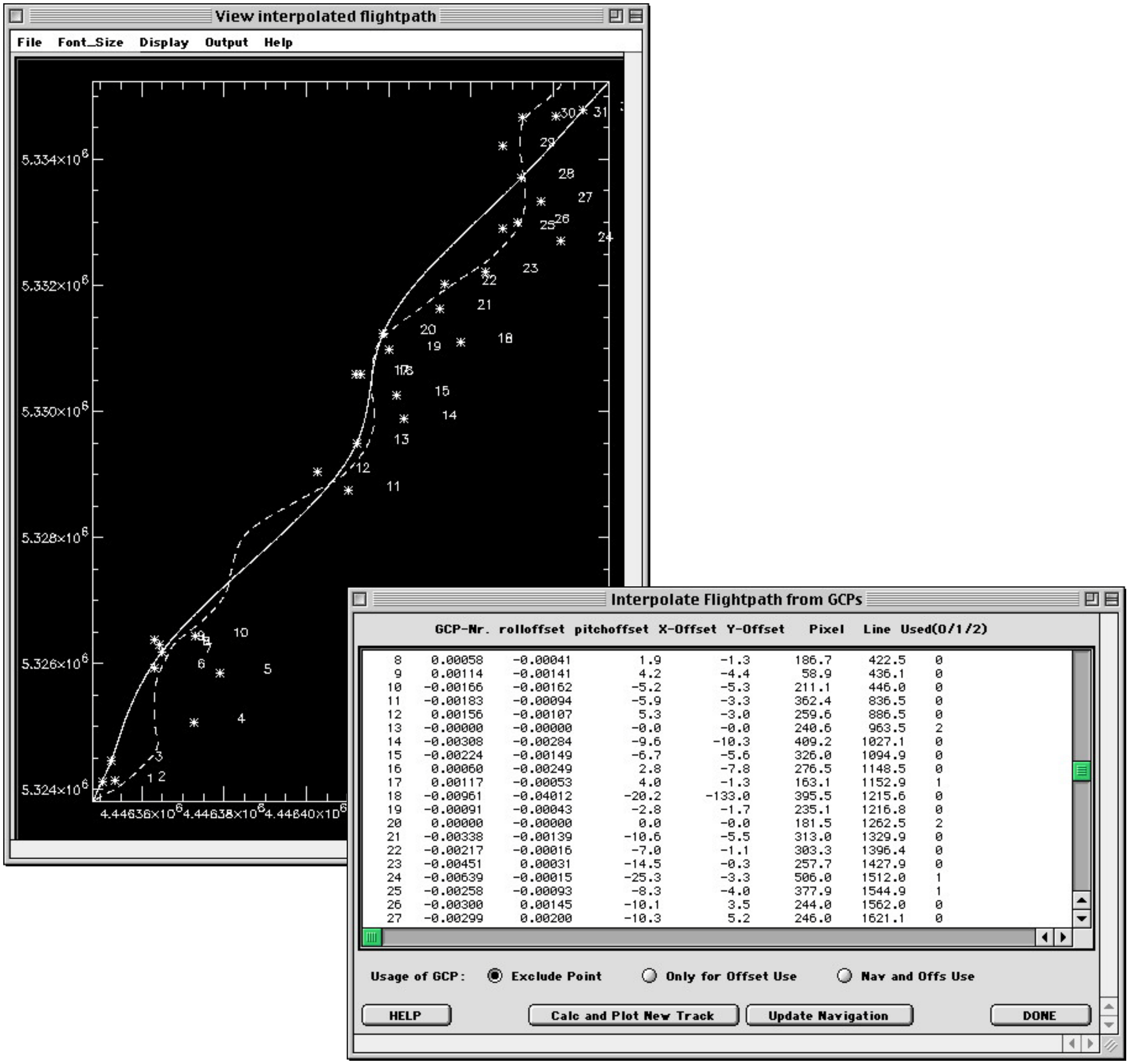


Figure 3.4: Flightpath interpolation from GCPs.

3.7 Operational Processing of Large Data Sets

For the operational processing, the use of the graphical user interface with its intrinsic flexibility is not an optimal solution. It is recommended to use the PARGE batch capabilities for that purpose. Also, you should decide which resampling workflow to follow.

The following procedure is recommended to setup an operational processing of a big data acquisition:

a) Prepare the DEM

Make your DEM(s) ready for the whole data acquisition. This might be one DEM for a series of image runs or one DEM for each run individually. If only one big DEM is selected, the memory requirements may increase drastically as PARGE uses the dimensions of the whole DEM as basic data structure. The DEM needs also to be resampled to the target resolution of your geocoding process.

b) Do the boresight calibration

Follow the procedures given in the previous section to retrieve the boresight offsets of the attitude parameters. Mostly, it can be assumed that the alignment is stable within one data collection (ie., one flight day or one campaign). Therefore, ground control points only need to be searched for one calibration runs with well known geometric references. Save the status for later reference

c) Setup a script

Write a script as described in Chapter 4 to process the runs automatically. Here, some IDL knowledge may be required. Use the function “`gc_impoffs`” to import the offsets from the calibration data set. An example script is also given in Appendix D: Example Batch Program on page 275.

d) Do the processing

Define a loop to go through all your image scenes/runs and process one after the other automatically.

e) Mosaic your results

Use the function “`env_mosaic`” to create a mosaic of all files in batch mode. Alternatively, you may use ENVI or any other standard processing software to do a mosaic of the resulting images.

Attention: Note that a full IDL or ENVI developer license and some basic knowledge about IDL programming is required for successful setup and testing of a processing script.

3.8 Example Results

The PARGE procedure has been applied to a broad variety of airborne imaging spectrometry data such as HyMap, AVIRIS, AISA, HYSPEX, and others. Many sensor have their own problematic specialities which has lead to the high degree of flexibility in the current application.

Some example results are given in the next sections. The first two examples are representatives for two kind of data: The AVIRIS data of 1998 are of experimental character with respect to their geometric quality. This data set is mainly used to test the flexibility and limitations of the PARGE software for critical imaging spectrometry data. The HyMap data on the other side are processed on a regular basis using the PARGE application. The respective results correspond to the real-world accuracies which can be achieved on a truly operational basis. Thereafter, two examples of for modern pushbroom-type imaging spectrometers are shown for the SAMSON and for the HYSPEX imaging spectrometers. PARGE is used for both systems to do the operational geometric processing.

3.8.1 AVIRIS

Already the initial development of the PARGE-precursor application had been based on AVIRIS data. It has been the low altitude option flying AVIRIS on a Twin Otter aircraft which pushed the demand for parametric georectification again. AVIRIS registers the earth surface at a FOV of 31.2° and 1mrad IFOV from altitudes between 3 and 21 km. The AVIRIS data providers at JPL introduced a fully parametric standard rectification procedure which however does not include a true orthorectification. Additional geometric processing would be required to retrieve orthorectified images from these standard products. Hence, the PARGE mehod has been adapted and tested on standard raw AVIRIS data products in order to provide a 'one-step' orthorectification procedure for this instrument. The workflow differs between high altitude data and low altitude data due to the distinct GPS/INS configurations and the difference in aircraft characteristics.

For high altitude data, only the ER-2 aircraft information for position and attitude is available, synchronized to the AVIRIS data stream. The heading is taken as derivative of the (non-differential) GPS-flightpath. The low frequency pitch variations are interpolated over best-known GCPs (since pitch is not measured so far for the sensor head) and overlaid to the high frequency pitch measurements. Both parameters require some smoothing filtering to remove measurement artefacts. Another problem is the software roll compensation which has been applied to the standard AVIRIS high altitude data: the provided aircraft roll parameters are no longer relevant for processing. Analysis using a number of GCPs however have shown, that significant residual roll variations within 0.5 degree remain in the data. The roll variations have therefore to be interpolated from GCPs. All the above assumptions are applicable since the aircraft is flying at 20 km altitude under very stable conditions and the data do not suffer from large high frequency distortions.

For low altitude data acquisition, a Boeing C-MIGITS II IMU has been mounted on the AVIRIS sensor head starting in 1998. This system is comparable in accuracy to the unit used on the HyMap sensor (see next section) and measures the position and attitude to at best one pixel accuracy. The use of this auxiliary data for currently available AVIRIS data sets can be very time-consuming; specifically since the synchronization offset between attitude measurements and image lines is not known exactly and since the IMU data formats are inconsistent between the years. The external GPS and attitude data have to be synchronized using (an often varying) offset to the AVIRIS master clock.

An experimental data set has been investigated with both high altitude data (flown at 21 km altitude) and low altitude data (at 3.7 km altitude) acquired over the Ray Mine site, Arizona, in summer and fall 1998 [43]. A standard USGS DEM with a horizontal resolution of 30 meters is used as surface reference. It is resampled to the required output resolutions for both data sets, and registered to the UTM (North american datum, 1927) coordinate system. The processing has been done using the standard PARGE methods, with 28 GCPs for the high altitude data and 18 GCPs for the low altitude scene. The results of co-registration of the two AVIRIS data sets is shown on the title page of this manual. The relative accuracy between the two images has been assessed by correlation analysis from image to image and from images to the illumination shaded DEM (as described in the section on accuracy assessment on Section 2.7.3 on page 39). All given accuracies are of relative character since no absolute geometric standard could be established for the test area. Overall relative accuracy turns out to be stable throughout the high altitude and low altitude flightlines which consist of 1478 and 4487 contiguous scan lines, respectively. The observed relative shifts between the three co-registered data layers were between 15 and 40 meters. This accuracy is low when comparing to the low altitude pixel size (3.75m), but is within the estimated accuracy for the high altitude data where no accurate flight parameters were available.

The residuals of the GCPs indicate an error in the range of 10 -15 m (3-5 pixels) for the low altitude scene while the high altitude residuals are in the range of 20-30 meters. The low altitude data accuracy suffered from the incomplete system integration and synchronization, the low DEM resolution, and from drifts on the pitch parameter which needed to be corrected. The accuracy still bears the potential for improvement if higher quality DEMs or surface models would be used. Anyhow, the results are within an acceptable margin given the fact that only a small number of GCPs had to be used. Tests of the software capabilities on more recent AVIRIS data containing completely synchronized attitude and DGPS information will be done as soon as such data are available.

3.8.2 HyMap

Orthorectification of HyMap data had been envisaged as the data of this Australian sensor had become available to a broad community of users in Europe. Having a large FOV of 61.44° at an IFOV of 2 mrad, orthorectification is a must for this sensor if the radiometry shall be corrected in dependence of the geometry. An IGI Inertial Navigation System had first been mounted on the HyMap sensor head for dedicated campaigns in Europe in 1998 and 1999. Although the IGI system is of high accuracy, this setup was flawed by the missing system integration and cumbersome synchronization to the HyMap imaging spectrometer. In mid 1999, a Boeing C-MIGITS system was finally integrated to the HyMap system as standard measurement unit. The accuracy of this system is in the range of 1 mrad for roll and pitch and down to 3 mrad for heading [66]. It decreases with longer flightlines, since it may be affected by slight drifts. Hence, at best it was possible to achieve pixel-accurate geocoding using this INS data.

One main goal of the campaign in Barrax 1999 (organized and funded by the European Space Agency [6]) was to measure the spectral variability of agricultural areas under varying geometric flight conditions. Six overlapping data sets has been taken over the same area. These data sets have been co-registered using the PARGE application under operational conditions and based on a flat, artificial DEM. The achieved accuracy as derived from the residuals of 15 GCPs was between 7 and 10 meters.

Table 3.2: Cross correlation offset analysis between six co-registered HyMap Images of varying heading (west or south) and day-time. RMS values of the image-to-image offsets in pixels at a pixelsize of 5 m after geocoding.

	south, 12 UTC	**south, 15 UTC**	**west, 9 UTC**	**west, 12 UTC**	**west, 15 UTC**
south, 9 UTC	1.75	1.50	2.11	1.56	1.37
south, 12 UTC	0	2.16	1.94	1.56	1.66
south, 15 UTC		0	2.16	1.84	1.36
west, 9 UTC			0	2.29	1.44
west, 12 UTC				0	1.60

An example of the differences between two co-registered images is shown in Figure 3.5. The strong appearance of some field borders on the other hand can be clearly attributed to the co-registration errors between the two images. The RMS co-registration errors have been assessed by cross-correlation analysis of all six images. The horizontal offsets have been searched by systematic correlation analysis on a regular raster of 100 pixel width, spread over the overlapping image size. Up to 10% of the test points have been excluded from the offset calculation if no correlation maximum could be found within 10 pixels vicinity. This has been the case if the correlation is searched within mostly homogeneous agricultural fields (e.g.) or if the surface

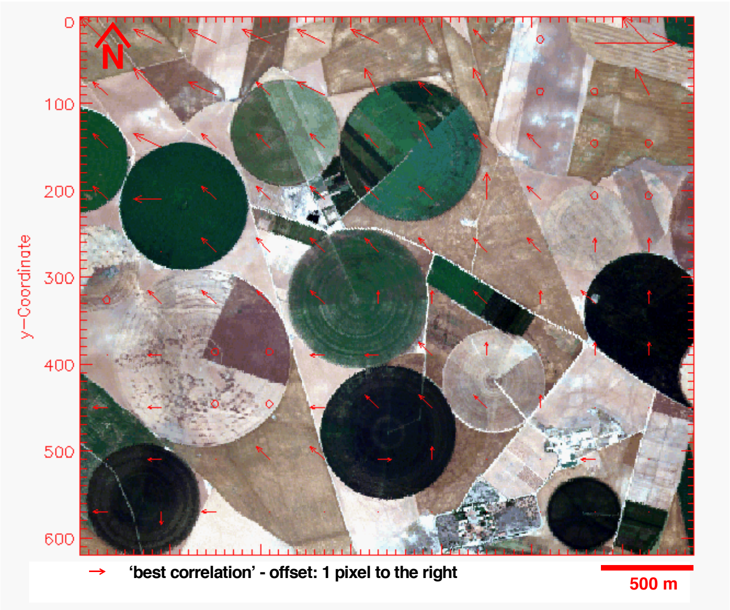


Figure 3.5: Co-registered HyMap data. Raw data true color image with overlaying offset-arrows at co-registration errors of 1-2 pixels (3.52 x 3.10 km).

characteristics changed between two acquisition dates. The results are summarized in Table 3.2 and an example is shown in Figure 3.5. According to the table, the co-registration error for this HyMap imagery is between 1.3 and 2.2 pixels which corresponds to 6.5 to 11 meters. This result confirms the residuals as derived from the GCPs.

The observed error can be attributed to the non-availability of a digital surface model and the moderate quality of the IGI INS system integration (alignment/synchronization) with HyMap used in 1998. Also, the number of GCPs was restricted to 15 per scene and the processing had been done using PARGE in an operational environment at the German Aerospace Centre (DLR) without much room for extensive re-iteration efforts. The results could be further optimised by adding in-field measured GCPs in relevant areas and by optimising the parameter offsets to fit the best GCPs.

Results from more recent HyMap scenes showed GCP residuals better than 3 meters. A number of 30 and more GCPs had been used for boresight calibration. The achieved accuracies are considered being practical accuracy limit for this sensor/INS combination. An example result of mosaiced HYMAP data on a digital elevation model is depicted in Figure 3.6 (compare the Figure on title page of this manual for another example result).

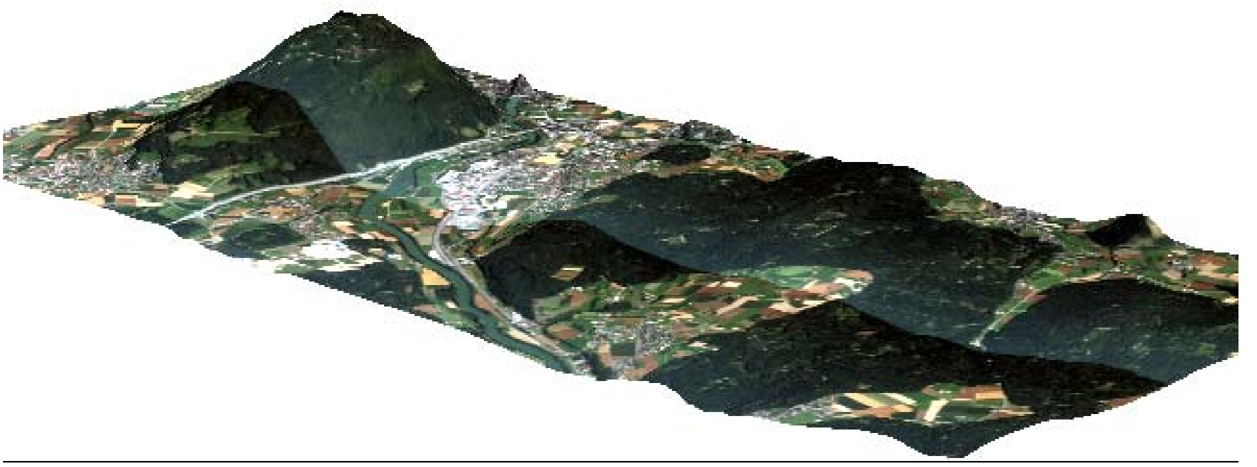


Figure 3.6: Mosaiced HYMAP-data (2004) overlaid to a digital elevation model. (Data courtesy of RSL, University of Zurich)

3.8.3 SAMSON

With recent data sets from the SAMSON system, sub-pixel accuracies below 0.2 pixels could be proven in relation to orthophoto data basis, using high accuracy and well integrated IMU systems. The example images of Florida Environmental Research Institute's Spectroscopic Aerial Mapping System with Onboard Navigation (SAMSON) sensor (Figure 3.7 and Figure 3.8) shows an excellent agreement on the basis of 1m spatial resolution.

Figure 3.7: Mosaic of 6 hyperspectral flighlines as processed with PARGE. (© Florida Environmental Research Institute's Spectroscopic Aerial Mapping System with Onboard Navigation (SAMSON) sensor)

Figure 3.8: Overlay of two SAMSON flightlines with a digital orthophoto at 1 m spatial resolution

3.8.4 HYSPEX

The Hyspex system is being built by the Norwegian company Norsk Electro Optics (NEO) and is increasingly popular as a stable off-the shelf imaging spectrometer. PARGE has been selected as the standard method for the processing of Hyspex imagery. Highest accuracy on the 1m level could be achieved in combination with a laser scanner providing the digital surface models. Example results are depicted in Figure 3.9.

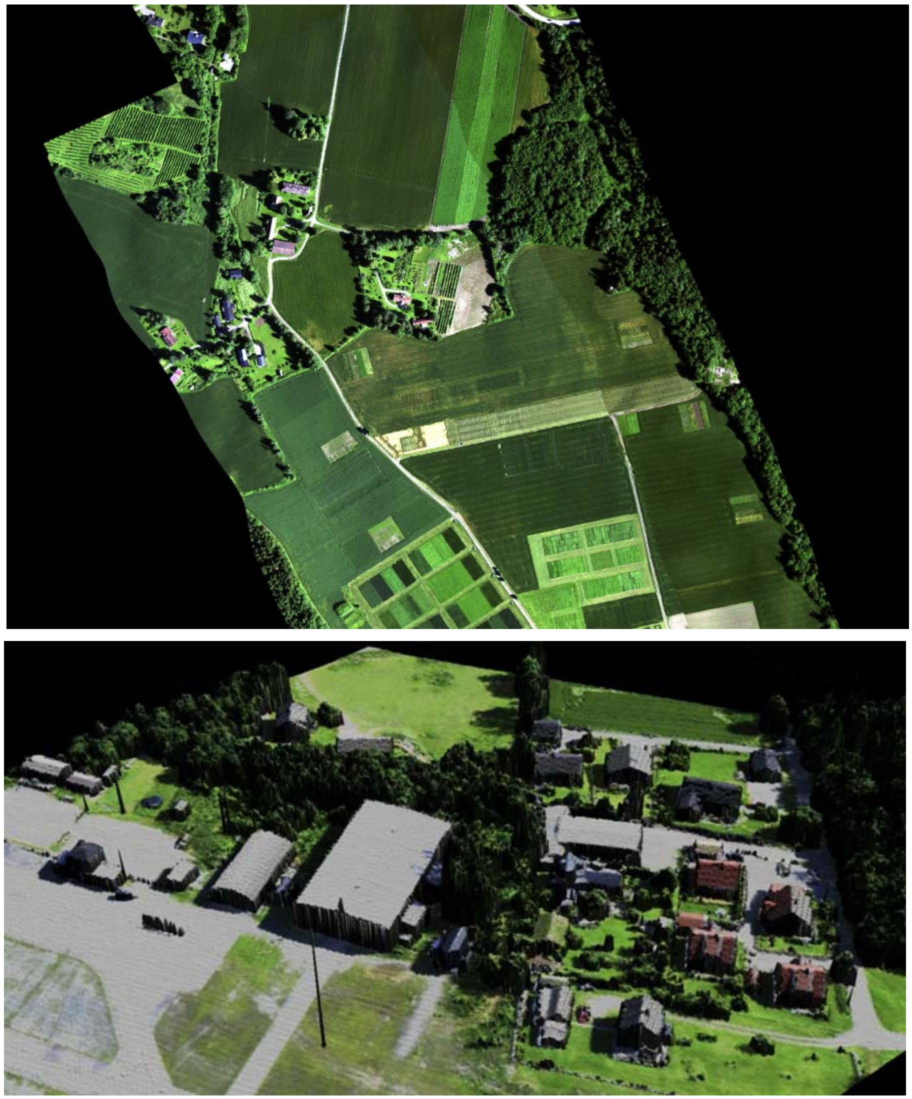


Figure 3.9: Processing results for the HYSPEX imaging spectromete. Top: mosaic of two adjacent flight lines, Bottom: overlay with laser scanning data. Images courtesy of Norsk Elektro Optik (NEO).

Chapter 4:

PARGE Programming Interface

4.1 Internal Data Format

All essential parameters of PARGE are stored in a proprietary data format. They can be saved and restored using the **>File:Save Status<** [p.147] and **>File:Restore Status<** [p.146] commands of the File menu. The restore command additionally reads the DEM and one single image channel from the specified files to memory while they are not read with **>File:Restore Variables<** [p.147].

There are three ways to make the variables accessible on the IDL prompt if PARGE is run from a licensed IDL installation (these methods are not available on the IDL virtual machine):

1) Restore Status on the IDL prompt:
   `IDL> restore, ‘mystatus.gcs’`
2) Function **>Special:Interrupt<** [p.225] : The variables will be available as long as no *retall* command is performed. `IDL> '.c'` takes you back to the PARGE application.
3) IDL prompt: exit PARGE and type on the IDL prompt:
   `IDL> common c_geovar`
   `IDL> common c_geostruc`
   and the variables will be available as long as the session lasts. Typing 'parge' again will allow for saving and further use of the manually set parameters.

The two common blocks of shared variables which are used for PARGE are described in the section below.

Important: The internal units for all measures are radians and metres.

4.1.1 Variables

The variables of variable dimensions are defined in an IDL-common block as follows:

COMMON **c_geovar**, m_geo, rollarr, pitcharr, yawarr, headarr, navarr, gpsarr, demarr, sensarr, onearr, gcparr, syncarr, chflag

Explanation of the variables:

- **m_geo**: (obsolete) widget-ID of the group leader. Default: -999 (now wid.m_geo)
- **rollarr**: array of the roll data per image line.
  Default: dblarr(1,image.n_l)
- **pitcharr**: array of the pitch data per image line.
  Default: dblarr(1,image.n_l)
- **yawarr**: array of the yaw data; is NOT used for the PARGE main processor.
  Default: dblarr(1,image.n_l)
- **headarr**: array of the true heading data.
  Default: dblarr(1, image.n_l)
- **navarr**: Array of the navigation data, containing x-coordinate, y-coordinate, and altitude in meters per line
  Default: dblarr(3, image.n_l)
- **gpsarr**: array of the original navigation data, containing longitude, latitude (geographic coordinates degree), and altitude (meters) per line.
  Default: dblarr(3, image.n_l)
- **demarr**: array of the DEM data [unist: meters; floating point or integer values], defines the dimensions of the output.
  Default: -1
- **sensarr:** array defining the sensor model by pixel number, across track position, along track position relative to absolute nadir (0,0);
  Default: dblarr(3,image.n_p) OR dblarr(4,image.n_p); the last column may contain a flag for use of the sensor model. If the flag is not 1, the column is not used.
- **onearr**: array containing a single channel test image in raw dimensions.
  Default: -1
- **gcparr**: Ground Control Point stack for offset calculations (imgpix, imglin, demx, demy, demh, flag).
  Default: dblarr(6, 1)
- **syncarr**: synchronization time per scanline, starting with value of image.starttime; unit: decimal hours.
  Default: double(-1)
- **chflag**: flag-array for the treatment of each channel: 0= channel is dismissed; 1= channel is calculated. Default: chflag = -1 (this variable currently not used)

4.1.2 Auxiliary information about input data

The following lists describe the four major structure-variables of the PARGE common block:

COMMON **c_geostruc**, image, sensor, dem, aux, result, pr, wid

The structures are accessed withing IDL by the syntax 'strucname.tagname'; e.g. the image name is given in the variable 'image.name'. The single variables are categorized by their importance regarding the geometry processor with the following types:

Main	Used for the main processing algorithm
Auxiliary	Auxiliary information, which normally is not required for the processor
Offset	Data used for recalibration of flight parameters
Sync	Data for synchronisation of external auxiliary data streams

a) Image

Describing the raw data cube in bip/bsq format and the fixed sensor parameters.

Table 4.1: Image data structure

image.x	Default	Description	Type
name	""	File name of original, not geocoded bip-cube	Main
sensor	""	Name of the sensor type	Auxiliary
date	""	Date of the image taking	Auxiliary
descr	""	Description string for this image	Auxiliary
n_p	0L	Number of pixels of cube	Main
n_l	0L	Number of lines of cube	Main
n_c	0	Number of bands of the cube	Main
h_off	0	Header offsets (bytes)	Main
psize	[0.d,0.d]	Nominal pixel size [m] (pix/line-direction)	Auxiliary
datyp	0	Datatyp of the image data; 0: BIP signed integer, 1: BIP unsigned integer; 2: BSQ signed integer, 3: BSQ unsigned integer; 4: BSQ byte (8bit), 5: BSQ floating point.	Main
calfile	""	Name of file containing radiometric calibration	Auxiliary
smilefile	""	Name of file containing smile calibration	Auxiliary
frownfile	""	Name of file containing frown calibration	Auxiliary
synfile	""	Name of file containing sync signal	Sync
starttime	0.0d	Time of first image line HHMM.SSS	Sync
endtime	0.0d	time of first image line [decimal hour]	Sync
scanfreq	0.0d	scanning frequency (lines per second)	Sync
gcpfile	""	name of imported GCP File	Offset

Table 4.1: Image data structure

image.x	Default	Description	Type
gcpdescr	“”	description of GCP File	Offset
gcpstand	0	standard of GCP File	Offset
gcpcoord	‘Arbitrary’	coordinate system description of GCPs standards. Syntax (comma separated string): 'projection, zone-info, North/South,...']	Offset
gcpdatnum	-1	geographic datum number of GCPs	Offset
ngcp	0	Number of actual GCPs	Offset
atmo	“”	ATCOR atmospheric file code	Auxiliary
visibility	0.	estimate on atmospheric visibility	Auxiliary
zen	30.	solar zenith angle	Auxiliary
azi	180.	solar azimuth angle	Auxiliary
aux1	“”	ignore data value for pixels (ENVI header ‘`data ignore value`’)	Auxiliary
aux2	“”	auxiliary variable 2	Auxiliary
aux3	“”	auxiliary variable 3	Auxiliary

b) Sensor

The sensor structure stores all relevant information about the sensor definition and its physical status.

Table 4.2: sensor data structure

sensor.x	Default	Description	Type
name	“”	Name of Sensor	Auxiliary
owner	“”	Name of Sensor owner	Auxiliary
descr	“”	Description of Sensor / Calibration status etc.	Auxiliary
IMU_type	“”	Description of Sensor INS/IMU	Auxiliary
GPS_type	“”	Description of Sensor INS/GPS	Auxiliary
model	“whisk”	Type of Sensor Model: 'whisk','push','explicite'	Main
fov	0.0d	FOV (swath) of the sensor edge-to-edge [rad]	Main
fovtilt	dblarr(2)	x/y tilt offset of the FOV to 'original' FOV	Main
ifov	dblarr(2)	Instantanous FOV of the pixels [rad] (across track (0) and along track (1))	Main

Table 4.2: sensor data structure

sensor.x	Default	Description	Type
leverarm	dblarr(3)	x/y/z distance of GPS antenna from sensor	Main
file	""	Name of sensor model file	Auxiliary
calfile	""	Name of file containing radiometric calibration (from labatory)	Auxiliary
smilefile	""	Name of file containing smile calibration (from lab)	Auxiliary
frownfile	""	Name of file containing frown calibration (from lab)	Auxiliary
missfile	""	Name of file describing the missing pixels (from lab)	Auxiliary
borefile	""	File containing boresight angles (may be *gcs)	Auxiliary
boresight	dlbarr(3)	default boresight angles (radian)	Auxiliary
scanp	0.0d	Scan period as partial of ifov(1)	Auxiliary
aux1	""	Type of sensor for Import dialog	Auxiliary
aux2	""	auxiliary variable 2	Auxiliary
aux3	""	auxiliary variable 3	Auxiliary

c) DEM

The Digital Elevation Model has to be resampled to a rectangular metric coordinate system (like UTM or Swiss L+T) and subsetted to the required image area. Its contents are stored in the variable *demarr*. The DEM itself is saved in ENVI data format by default.

Note: PARGE references the DEM, and also GCP pixels always to the center of each pixel; this is in contrast to the referencing which is used by ENVI, where the coordinates are referred to the upper left corner of the pixel. This issue is taken care of as long as the standard import fileters of PARGE for ENVI files are used.

Table 4.3: DEM structure tags

dem.x	Default	Description	Type
name	""	file name of original DEM	Main
descr	""	description of auxiliary DEM origin	Auxiliary
SE_Lon	0.0d	longitude of origin-corner: SE or SW; corner of the image, depending on referencing system; referenced to centre of the pixel)	Main

Table 4.3: DEM structure tags

dem.x	Default	Description	Type
SE_Lat	0.0d	latitude of origin-corner	Main
n_x	1	number of pixels in X-direction	Main
n_y	1	number of pixels in Y-direction	Main
psize	[0.0, 0.0]	pixel size [m] (x-direction/y-direction)	Main
overs	[1, 1]	oversampling rate (x-direction/y-direction); used to create oversampled DEM - not used currently.	Auxiliary
coord	Arbitrary	DEM projection description, according to ENVI; standards. Syntax (comma separated string); "system[, zone-number, area]"; e.g. dem.coord = "UTM, 12C, North"	Auxiliary
datnum	-1	geographic datum number of auxiliary data	Auxiliary
proj	""	holds optional projection information for custom map_proj.	Auxiliary
nad_x	0.0d	X-coordinates of starting line nadir pixel [in (sub-) pixels of DEM]	Auxiliary
nad_y	0.0d	Y-coordinates of starting line nadir pixel [in (sub-) pixels of DEM]	Auxiliary
decl	0.0d	magnetic declination of INS heading to cartographic north for the scene [rad]	Auxiliary
mean	0.0d	mean altitude of the whole DEM (meter)	Auxiliary
aux1	""	name of raw DEM (after smoothing)	Auxiliary
aux2	""	ENVI projection system string (PROJCS), see etc/projcs.sav	Auxiliary
aux3	""	auxiliary variable 3	Auxiliary

d) Auxiliary

These data are resampled to image dimensions and stored for each line; the preparation is performed by the data specific modules. Source of the engineering data could be INS or instrumental gyros. The orientation and the number of lines correspond to the raw image.
Up to 14 auxiliary data sets may be managed through this data structure, but PARGE usually only relies on the first seven elements (aux[0:6]).
The complete auxiliary data set contains 7 parameters available for georectification, where from the true heading and the yaw can contain redundant information. A second set is foreseen for the storage of alternative data in *aux*[7:13].

The numbers of the single data sets are assigned as follows:]

Table 4.4: AUX structure array numbering

Nr.	Description
0,1,2 (7,8,9 resp.)	roll, pitch, yaw of Instrument gyros or INS
3 (10 resp.)	true heading of aircraft or INS
4,5,6 (11,12,13 resp.)	long, lat, altitude as measured by GPS system

Table 4.5: AUX structure tags

aux(n).x	Default	Description	Type
name	“”	File name of the auxiliary data origin	Auxiliary
descr	“”	Description of auxiliary data origin	Auxiliary
coord	Arbitrary	coordinate xystem standards; syntax (comma separated string): 'projection, zone-info, North/South,...'	Auxiliary
datnum	-1	geographic datum number of auxiliary data	Auxiliary
proj	“”	holds optional output projection information for custom map_proj.	Auxiliary
standard	0	type of the auxiliary data flag depending on type.	Auxiliary
starttime	0.0d	time of the first measurement of file; (hhmm.sss)	Sync
sync	0L	synchronisation point of the measurement which corresponds to the first image line	Sync
freq	0.0	number of measurements per second	Sync
n_s	1	number of samples per image line; (for whiskbrooms, currently only 1 is supported)	Auxiliary
offset	0.0d	offset which was applied to the original data	Offset
used	0	flag if the parameter is used for the calculation (0: not used, 1: used)	Auxiliary
aux1	“1“	applied smooothing factor to data	Auxiliary
aux2	“”	auxiliary variable 2	Auxiliary
aux3	“”	auxiliary variable 3	Auxiliary

Example: The name of the roll data origin file can be accessed by the variable “aux[0].name”.

e) Result

The structure 'result' stores the names of the anticipated or created output files along with the single channel information.

Table 4.6: RESULT structure tags

result.xxx	Default	Description	Type
remaparr_name	'*_map'	save name of the remapping array.	Main
geoband_name	'*_gbd'	save name of a single band image	Auxiliary
georgb_name	'*_rgb'	save name of a rgb image	Auxiliary
geocube_name	'*_geo'	save name of the whole geocoded cube	Main
sca_name	'*_sca'	save name for scan angle file	Auxiliary
igm_name	'*_igm'	save name for ENVI igm file	Auxiliary
igm_cord	""	IGM coordinate system	Auxiliary
demslp_name	'*_slp'	save name for DEM slope file	Auxiliary
demasp_name	'*_asp'	name for DEM aspect name	Auxiliary
demsky_name	'*_sky'	name for DEM skyview name	Auxiliary
demshd_name	'*_shd'	name for DEM aspect name	Auxiliary
accuracy_name	""	name for accuracy assessment file	Auxiliary
accuracy	[0.,0.]	estimated accuracy in x/y direction	Auxiliary
geoband	1	number of the active single band	Main
rgbbands	[1,1,1]	number of the bands used for RGB processing	Auxiliary
bandrange	[1,1]	band range to be processed	Auxiliary
pixrange	[1,1]	range of pixels to be processed	Auxiliary
linerange	[1,1]	range of lines to be processed	Auxiliary
n_tiles	1	number of tiles used for processing	Auxiliary
mainopt	[0,1,0,0,0 ,100,0]	options set for main processor	Main
finopt	[1,1,0,1,3 ,0,0]	options set for final processor	Main
cubopt	[1,1,0,1,3 ,0,0]	options set for final processor	Main
rgbopt	[1,1,0,0,0 ,0,0]	options set for the RGB processor	Auxiliary
igmopt	[1,0,0,0,0 ,0,0]	options set for the IGM processor	Auxiliary

Table 4.6: RESULT structure tags

result.xxx	Default	Description	Type
accopt	[1,1,100, 100,12,0]	options set for the accuracy assessment / correlation analysis processor: refband1, refband2,resolution, patchsize, pixshift	Auxiliary
proctilt	[0.,0.]	parameters for tilted processing	Auxiliary
batchscr	“”	file name for batch script	Auxiliary
batchlog	“”	file name for (batch) log outputs	Auxiliary
descr	“”	description of result for ENVI header	Auxiliary
SE_Lon	0.0d	longitude of origin-corner: SE or SW; corner of the image, depending on referencing system; referenced to centre of the pixel)	Main
SE_Lat	0.0d	latitude of origin-corner	Main
n_x	1	number of pixels in X-direction	Main
n_y	1	number of pixels in Y-direction	Main
psize	[0.0, 0.0]	target pixel size [m] (x-direction/y-direction)	Main
coord	‘Arbitrary’	Output Projection description, according to ENVI; standards. Syntax (comma separated string); “system[, zone-number, area]”; e.g. result.coord = “UTM, 12C, North”	Auxiliary
datnum	-1	geographic datum number of result	Auxiliary
proj	“”	holds optional projection information for custom map_proj.	Auxiliary
rotation	0.0	output rotation [deg].	Auxiliary
aux1	“”	name of illumination file	Auxiliary
aux2	“”	coordinate system string of result (csyst; ENVI Projection String PRJCS)	Auxiliary
aux3	“”	internal binning x/y factor. '{x,y}' joined processing parameters, {splitwvl, nbands firstcube, nbands secondcube}	Auxiliary

f) Preferences

The structures 'pr' and 'wid' contain information about the PARGE preferences and widget identifiers. Some preference variables useful for batch processing are given below:

Table 4.7: PR structure tags

pr.xxx	Default	Description	Type
status_name	""	currently active status file	Auxiliary
date_created	""	status creation date	Auxiliary
creator	""	name of licensee	Auxiliary
date_modified	""	status modification date	Auxiliary
user	""	name of user (id)	Auxiliary
parge_vers	'3.0'	version of parge where file was saved	Auxiliary
batch	0	batch processing mode: 0: normal processing 1: batch processing 2: debugging mode (verbose)	Auxiliary
disk	0	batch processing mode: 0: normal processing 1: reduces RAM requirement, intermediate processing results are put on disk	Auxiliary
hsry	strarr(50)	last 50 steps performed on the data	Auxiliary
tmpdir	""	directory name where to store temporary data	Auxiliary
colors	[0,0,0]	color definition vector	Auxiliary
proj	""	Custom IDL projection definition: {IDL projection #, IDL ellipsoid number, keys}. (compare IDL reference guide for definitions)	Auxiliary
proj2	""	Alternative custom projection	Auxiliary
aux1	""	Default Web Browser	Auxiliary
aux2	""	auxiliary variable 2	Auxiliary
aux3	""	auxiliary variable 3	Auxiliary

4.1.3 Outputs Description

Some details about the output files as depicted in Figure 3.2 on page 45 are given below:

Status Files:

recommended extension: *.gcs.
Status information of the geocoding session.
Data format: regular IDL save file

BSQ Image

recommended extension *_bsq
This type of raw cube is created by PARGE if 'bip/bil to bsq' conversion is chosen.
Data format: Binary BSQ image with ENVI-type header

Image Header

recommended extension *.hdr
An ENVI™ format header to the raw imagery (whatever.bsq with whatever.hdr)
Data format: ENVI-type image header. Due to compatibility to the Windows extension conventions, PARGE recommends an 3-letter extension for all files - note that this may lead to confusion in ENVI, if the distinction of image files is done by extension only!

DEM

recommended extension *_ele.bsq with *_ele.hdr
digital elevation model / surface model; elevation in meters
usually stored in integer format (meter-accuray) may also be floating point.
Default format: demarr = intarr(dem.n_x, dem.n_y) + ENVI-type image header
Geolocation in ENVI: Magic Pixel coordinates, define on upper left edge of pixel

Slope and Aspect

recommended extension *_slp.bsq and *_asp.bsq
terrain slope and aspect derived from digital elevation model or surface model.
Format: single band ENVI file format, stored as
*_slp: 8bit Byte data of slope (in degrees; 0-90°)
*_asp: 16bit integer data of aspect (towards north, east=90 deg, 0-360°),in degrees
(derived from: *_ele - corresponding DEM in metres)

Skyview and Cast Shadow

recommended extension *_sky.bsq and *_shd.bsq
skyview factor and cast shadow map derived from digital elevation model or surface model.
Format: single band ENVI file format, stored as
*_sky: percentage of visible sky above pixel, floating point value [%]
*_shd: mask of cast shadow, 8bit Byte data of slope [degrees]
(derived from: *_ele - corresponding DEM in metres)

Report

recommended extension: *.txt
Status Report: text describing the current status.
Consistency Report: a report about the consistency of the auxiliary data to the image. The file sizes of all available data is checked agains the auxiliary data information. If inconsistencies are found hints are given to resolve them.
Data format: ASCII text

Ground Control Points

recommended extension: *.gcp (imported from ENVI: *.pts)
export of the ground control points for storage and later use in PARGE and other applications.
Data format: Columnar ASCII file, pixel/line/x/y/altitude

Mapping Array

recommended extension: *_map.bsq
A two-layer array containing the raw pixel and line coordinates for each DEM element (see description in Section 2.2.6 on page 22. It furthermore contains the information about the real pixel centers and the background mask.
Default format: remaparr = intarr(dem.n_x,dem.n_y,2), raw binary
(Attention: this array is currently not transferrable between different memory architectures)

Geocoded RGB Composite (*_rgb.tif/*_rgb.jpg)

recommmended extension: *_rgb.tif;/*_rgb.jpg
is written by P*rocessor: Process Band or RGB*, can be easily exported to image processing applications, but usually has lost the radiometric and absolute geometric position (is scaled to 1byte data)

Geocoded Image Cube

recommended extension: *_geo.bsq
Output of final processor for the defined channel range in ENVI data format (BSQ or BIP) of all or some channels (*_geo), maintaining the original radiometry
Format: Ortho-rectified geocoded hyperspectral image in ENVI™ format.

Single Channel Image

ENVI format (*bsq), TIFF format (*.tif), PICT Format (*.pct), JPEG format (*.jpg)

Scan Angles

automatic extension: *_sca.bsq
Data layers in defined output geometry or in raw geometry with scan zenith, scan azimuth angle, and the absolute distance from aircraft to the pixel for each geocoded pixel. The first band of this image is used as standard input for ATCOR-4 radiometric processing whereas the second band is only required for BRDF correction.
Format: 3 bands integer ENVI image.
1: Sensor zenith angle (degree * 100), 0° is downward looking,

- unkown/invalid values are greater than 9000 (asigned value: 9100),
- pixels to the right with respect to the flight direction have negative values.

2: Absolute azimuth angle to direction north (degree * 10, unknowns are lower zero)
3: Height of airplane above ground for each pixel [metres], unknowns are 0
4: Optional: pixel distance: distance in meters from aircraft to each pixel.
Optional: Relativ azimuth angle to flight direction (would be in channel 2 while shifting channels 2/3 to 3/4)

Image Geometry Map

extension: *_igm.bsq
ENVI standard image geometry description in raw image dimensions, storing the x/y pixel positions as double precision floating point value for each pixel in acquisition geometry.

Roll Compensated Image Cube

From the roll compensation routine (Recommended extension: *_rlc.bsq).
This output cannot be further processed by PARGE, since the internal geometry has been lost
Format: ENVI image cube

Temporary Files

The below two files can be found in the default temporary directory (see variable pr.tmpdir or function **Edit:Preferences** p.161.'

parge_temp.gcs:
temporary status saved during processing
This file may be used for recovery of PARGE session after system failure, using a simple ‘restore status’ command.

parge_temp0.gcs:
old temporary status (used for undo command).

Log Files

extension: *.log
Log file which is written after initialization in the dialog **Edit:Preferences** p.161. The log file is continuously updated during processing or interactive work

4.2 Batch Process from Parameter File

PARGE may be started as a batch process using a ASCII parameter file from prompt or from within the application using the function **Processor:Batch Processor** p.236. From the prompt, the call is on Windows:

```
xx\idlrt.exe -rt="...\parge.sav" -args batchfile [logfile]
```

and on Unix, it is:

```
xx/idl -rt=".../parge.sav" -args batchfile [logfile]
```

with:

`xx`	path to the idl installation (also IDL VM)
`batchfile`	name of ASCII batch file to be used as parameter file for processing
`logfile`	name of logfile to be created/used during the processing.

The processing defaults to the UTM coordinate system.

4.2.1 The Batch Parameter File

The parameters used in a batch process may be written to the batch parameter file in arbitrary order using a syntax of the form "`parameter = value`". Lines without equal sign in the file are ignored. The allowed parameters tags can be found in Table 4.8.

Table 4.8: Parameters supported in parameter file.

Parameter	**Explanation**
sensortype	sensor and sensor configuration in a syntax "sensor-x", e.g. HYSPEX-0
image file	Input image file
navigation file	Navigation GPS/IMU data file:
sensormodel file	Sensor model file (optional)
gcp file	GCP file (optional)
elevation file	DEM elevation file (optional)
sync file	Synchronization file (optional)
output file	Output File Name
resampling	Processing Parameters: Resampling (nearest|nearbilin|bilinear)
smoothing	Auxiliary data smoothing factor
sensor FOV	Sensor FOV [rad] (optional if sensor model is set)
red	red default band for RGB preview
green	red default band for RGB preview
blue	green default band for RGB preview
n_tiles	Number of tiles for processing
pixelsize	Output pixel size [m]
utmzone	Output UTM zone (default: -99)
mounting of IMU	Mounting of IMU for arbitrary sensor (default r-p+h+)

Table 4.8: Parameters supported in parameter file.

Parameter	Explanation
gcpheight	GCP height source (dem|data)
boresightfile	Boresight angles file (nomally a *.GCS file)
altitudeoff	apply altitude offset (yes|no)
firstline	range of lines to process: first image line (starting at 1, default: 0/ omit)
lastline	range of lines to process: last image line
firstband	range of bands to process: first spectral band (starting at 1)
lastband	range of bands to process: last spectral band

Note: the number of supported sensor systems by this batch process is limited. Please contact ReSe if a sensor system is to be added.

4.3 Setting up a Batch Process within IDL Programming Language

Batch processing is foreseen with PARGE by accessing the intrinsic IDL functionality and the data format of the program. There are two approaches to set up a batch process. A complete batch process can be set up for standard data formats of sensors which are fully supported by PARGE while partial batch processing is done as soon as some user interaction is necessary. The way to set up a batch process highly depends on the operating system. We only describe the IDL-Call commands for Unix systems, while PC or Mac systems require their respective batch mechanisms.

A batch process is set up by first compiling a batch script in a pure ASCII text file as series of IDL and PARGE commands. The file can be started on Unix or on Mac OSX using the command:

```
unix> nohup idl batchscript > batchfile.log &
```

or to be kind to other users:

```
unix> nice nohup idl batchscript > batchfile.log &
```

on csh/tcsh the command might require consideration of the error output with :

```
tcsh> nohup idl batchscript >&! batchfile.log &
```

or on bash:

```
bash> nohup idl batchscript > batchfile.log 2>&1 &
```

The .log-file will then contain all diagnostic outputs of the process, including eventual error messages if the process is interrupted. Make sure that the log-file does not yet exist prior to starting up the process

If using the IDL workbench, the batchfile can be used by typing at the IDL prompt:

```
IDL> @batchscript
```

Now the diagnostics and the errors will be printed directly on screen, but the process will not run in the background if the IDL sesssion is finished.

4.3.1 Full Batch Process

A complete batch process starts with the original image data as stored after data acquisition, imports the data, starts a main process and finally performs a final process to create a geocoded image cube. An example script based on AVIRIS data would look as follows:

```
;; full batch script for AVIRIS data geocoding
;; -------------------------------------------
print,'PARGE Batch Process Log File'
print,'============================'
print,'Starting geocoding at: '+systime()
;
parge,/norun    ; allocates the PARGE variables
                ; and restores the program
gc_rdenvdem,'dem_ele.bsq'; reads ENVI DEM file
av_imp98,'aviris_cube'; imports 1998 standard AVIRIS data
gc_main,/expand ; expanded center pixel coding
gc_cube         ; geocodes raw image to cube
;
print,'Geocoding finished at: '+systime()
exit            ; exit the IDL session
```

This script assumes a perfect data set where no tuning and offset validation is necessary. Such a situation is not very likely with most of the current data sets and will therefore very seldomly lead to sufficient accuracies. Thus, the full batch mode might remain a dream for currently available data sets.

4.3.2 Partial Batch Process

The partial batch process is based on a combination of menu-driven set-up of the input parameters and the subsequent execution of a batch script for the main processing steps. It is based on a previously saved geocoding status (here: “status.gcs”) and could look as follows:

```
;; batch script for PARGE geocoding
;; --------------------------------

parge,/norun    ; allocates the PARGE variables
gc_restore,'status.gcs',/full; full restore on the status
gc_maini, /expand; main geocoding process
gc_cube_tile    ; final cube generation process
exit            ; exit the IDL session
```

A more detailed example batch script is given in Appendix D: Example Batch Program and can also be found in the `etc` directory of the PARGE distribution.

4.4 Batch Functions Reference

The complete PARGE functionality is available at the IDL prompt and executable in batch mode as soon as the application is restored with the command '`parge,/norun`'. A list of the functions suited for batch processing is given below, in the order of appearance in the PARGE GUI.

4.4.1 Session Management, File Menu

`parge, [/norun, /reset]`

`/norun`	initialize the data structure of parge without starting the GUI
`/reset`	resets the initial data structure to default values, use if multiple image lined need to be processed

Builds the internal data structure as described in Section 4.1 on page 69.

`parge_startbatch, parsfile, logfile`

`parsfile`	batch process paramter file (compare Section 4.2.1).
`logfile`	log file to be created or used.

Starts a batch process based on predefined parameters.

`gc_restore, statusfile [, /full]`

`statusfile`	PARGE statusfile to be restored
`/full`	restores the variables and loads the indicated DEM and raw data cube

Restores the status of a former session into the internal data structure.

`gc_savestatus, statusfile`

`statusfile` PARGE statusfile to be stored

Stores the status of a session from the internal data structure.

`gc_report:` **`repstr = gc_report([, /stats])`**

`stats` Include statistics of image/DEM in status report

Creates an ASCII report about the current status of a session (as shown in the function 'Show Status').

`gc_cleanup`

Deletes the temporary variables / files and leaves the PARGE session.

`gc_control:` **`ctrl = gc_control([, /ctrl])`**

`ctrl` Returns a flag (1: ok; 0: Error) instead fo a sttring

Performs a consistency check on the internal data and creates a report or optionally a flag about the consistency of the current variables.

`gc_export([, /atti,/nav,/gpx,/sens,/gcp])`

`/atti`	attitude data
`/nav`	converted navigation data
`/gps`	raw GPS data in WGS-84 system
`/sens`	sensor model [radian]
`/gcp`	ground control points

Exports the respective parameters as space delmited ASCII *.txt file.

`gc_initresult, infile [,/noaux]`

`infile`	raw image data cube
`/noaux`	inhibits the procedure from resetting the auxiliary data

Initializes the result data structure (see internal data structure) for an arbitrary input file.

`gc_initbatchlog, logfile`

`logfile` log file to be initialized

Initializes a log file and enters a default header to the initialized file. Further log entries are written to the file in the course of the processing.

gc_mapproj: **proj = gc_mapproj([info=..])**

`/info` returns an informational string about the currently selected projection

Returns an IDL-standard projection definition on the basis of the parameters stored in the variable `pr.aux1`. Basically this is the same as using the IDL standard function '`map_proj_init`' when working in IDL (compare IDL reference manual for further information on the way IDL handles projections).

gc_hsry, logentry

`logentry` text to be entered in log file and/or in history and console

Tracks what has been done.

gc_undo

Restores the last tem porary status (parge_temp0.gcs) to undo the last command.

gc_writebatch, batchfile

`batchfile` batch processing ASCII file to be created

Saves the currently known paramters to a batchfile for later batch processing.

4.4.2 Image Data Import

For generic data, no special function is necessary for raw image data import. In the batch process, the variables of the structure '*image*' have to be assigned for the appropriate values to let the procedure run as long as no import filter is available – of course it is possible to write an own import filter, which assigns the necessary variables valid for a specific sensor to the PARGE variables.

gc_renvi, envfile

`envfile` Name of the ENVI formatted image file to be opened.

Feed all available data from ENVI formatted image files to the PARGE data structure. Note that the critical sensor model parameters (i.e. the `sensor` structure and `sensarr`) are not available after reading an ENVI file, because they are not defined in the ENVI header structure. The sensor model thus has to be set separately using the function '`gc_sensmodel`'.

rd_onech: **onearr = rd_onech(chnr)**

`chnr` band number to be read from file

Reads one band from image file (after the file has been successfully defined) to memory. This band is kept as a reference in the variable ‘onearr’.

a) AISA functions

aisa_imp, aisafile [,fov=..]

`aisafile`	Standard AISA data cube
`fov`	FOV setting for the data cube.

Reads the standard AISA/AISA+ data. The INS-file should be named with extension *.txt. Eventual BIP data are converted to BSQ on the fly.

aisa_rdhead, aisafile,[,fov=..]

`aisafile`	Standard AISA data cube
`fov`	FOV setting for the data cube, for definition of the pushbroom model

Reads the standard AISA/AISA+ header and defines the sensor model.

aisaea_imp, aisafile, snsfile, navfile, synfile [freq=..]

`aisafile`	standard AISA data cube
`snsfile`	sensor model file
`navfile`	INS data stream (Applanix standard format)
`synfile`	synchronization information per image line
`freq`	frequency of data acquisition [Hz], default: 125 Hz.

Reads the AISA Eagle data (ISBE version). The INS-file should be standard Applanix format. Eventual BIL data are converted to BSQ on the fly.

aisaea_rdhead, aisafile, snsfile

`aisafile`	standard AISA data cube
`snsfile`	sensor model file

Reads the standard AISA Eagle header and defines the sensor model.

aisahs_imp, aisafile, oriheader, navfile[,fov=..]

`aisafile`	Standard AISA data cube
`oriheader`	AISA standard header file

`navfile`	INS data stream (Applanix standard format)
`fov`	FOV setting for the data cube.

Reads the AISA Eagle data (OGS version). The INS-file should be standard Applanix format. Eventual BIL data are converted to BSQ on the fly.

`aisahs_rdhead, aisafile [,fov=..]`

`aisafile`	Standard AISA data cube
`fov`	FOV setting for the data cube, for definition of the pushbroom model

Reads the standard AISA Eagle header and defines the sensor model (OGS and OHB).

`aisaohb_imp, aisafile, navfile, fov [freq=.., offset=..]`

`aisafile`	standard AISA data cube
`navfile`	INS data stream (Applanix standard format)
`fov`	FOV setting for the data cube, for definition of the pushbroom model
`freq`	frequency of data acquisition [Hz], default: 125 Hz.
`offset`	offset in seconds between image and navfile.

Reads the AISA Eagle data (OHB version). The INS-file should be standard Applanix format. Eventual BIL data are converted to BSQ on the fly.

b) APEX functions

`apex_imp, apexfile, navfile, housfile,sensfile, [/nowander]`

`apexfile`	APEX raw imagery (level 1).
`navfile`	navigation file (sbet)
`sensfile`	sensor model file
`/nowander`	do no correction of applanix wander angle

Reads an APEX data set.

`apex_rdhead, apexfile`

`apexfile`	APEX raw imagery.

Reads and initializes the default values from APEX header information.

c) AVIRIS Functions

A bunch of procedures is available for AVIRIS data import which depend on the year and the altitude of the sensor (the year in the function name stands for the first year applicable to the sensor while the 'l' designs a low altitude sensor):

```
av_imp91, avfile
av_imp95, avfile
av_imp96, avfile
av_imp98, avfile
av_imp98l, avfile
av_imp00l, avfile
av_imp01, avfile
av_imp02, avfile
```

`avfile`	Name of the AVIRIS image. We propose to convert the AVIRIS image to band sequential (BSQ) byte interleave for use in the PARGE application for performance reasons.

The data parameters including the navigation and the attitude is read into PARGE. The corresponding auxiliary data files should be named *.eng, *.nav, or *.gps respectively having the same name as the AVIRIS raw file. The GPS coordinates are read as raw geographic coordinates and may be converted to a metric system using the function `gc_coordtf`.

av_rdhead,avfile[,/write_head]

`avfile`	Name of AVIRIS image
`/write_head`	Write an ENVI standard header for this AVIRIS file

Makes the initialization for AVIRIS standard and auxiliary data. Attitude and GPS information have to be read separately after this procedure.

d) DAIS Functions

dais_imp, daisaux, [/ar, /combi, /oldtype,$ /nosmooth, /useftime]

`daisaux`	Name of DAIS auxiliary file
`/ar`	ARINC data only is taken for navigation
`/combi`	Combination of ARINC and DAIS gyros
`/oldtype`	Data was acquired befor 1.1.1999 (having 13 s of GPS time offset and pitch negative)
`/nosmooth`	Prevents the import filter from smoothing the auxiliary data as it is done by default.

`/useftime`	Use the real ftime parameter for synchronization instead of a reconstruction from start and frequency.

Reads and assigns the default values for a DAIS bsq-image including all available data on attitude, navigation and synchronization. The filename has to be the same as the auxiliary data file, except for the extension which is *.aux and *.bsq respectively.

dais_imp_igi, daisfile

Reads a DAIS data set acquired in conjunction with the DLR IGI navigation system. Requires a *.aux and *.pos file for data import.

dais_rdhead, daisaux

`daisaux`	Name of DAIS auxiliary data file

Assigns the image parameters according to the standard auxiliary data streams and the DAIS raw data file (*.bsq).

e) Headwall Functions

hyperspec_imp, hyspexfile, navfile, synfile, [timeshift=.., /wide]

`hyspexfile`	Hyspex raw imagery.
`navfile`	navigation file
`synfile`	sync file
`timeshift`	shift between IMU and image in hours (default 4)
`wide`	use wide FOV sensor definition

Reads a headwall hyperspec data set.

f) HYMAP Functions

hym_imp98, hymfile

Import HYMAP data until spring 1999, without accurate gyros, needed file: *.log.

hym_imp99, hymfile, igifile

Import HYMAP data of special DLR Munich configuration including the IGI system, needed files: *.log, *.pos.

hym_imp99a, hymfile

Import HYMAP data for configuration including the CMIGITS navigation system, where the parameters are read directly from the HYMAP LOG file.
needed files: *.log.

hym_imp00, hymfile [, gpsfile]

Import HYMAP data for configuration including the CMIGITS navigation system, reading the standard .out file coming with HYMAP data since 2000.

hym_imp03, hymfile [, gpsfile]

Import HYMAP data for configuration including the CMIGITS navigation system, reading the standard synchronized.gps file coming with HYMAP data since 2003.

`hymfile`	Name of the HYMAP image. We propose to assure that the file is in band sequential (BSQ) byte interleave for use in the PARGE application (for performance reasons).
`gpsfile`	Standard hymap CMIGITS GPS file
`igifile`	Name of IGI file for special DLR configuration.

The data parameters including the navigation and the attitude is read into PARGE. The corresponding auxiliary data are imported.

hym_rdhead,hymfile[,/write_head]

`hymfile`	Name of HYMAP image
`/write_head`	Write an ENVI standard header for this HYMAP file

Reads and assigns the default values for HYMAP header information to a HYMAP data file.

g) HYSPEX Functions

hyspex_imp, hyspexfile, navfile, sensfile, [/ifms]

`hyspexfile`	Hyspex raw imagery.
`navfile`	navigation file
`sensfile`	sensor model file
`/ifms`	ifms specific options

Reads a HYSPEX data set.

hyspex_rdhead, hyspexfile

`hyspexfile` Hyspex raw imagery.

Reads and initializes the default values from HYSPEX header information.

h) MIVIS functions

mivis_imp, mivisfile, rollfact=..

`rollfact`	Factor to scale roll parameter (default: 1).
`headshift`	Shifting factor for heading (default: -175 lines)

Reads a MIVIS data set. Requires files *geo_tabxx, *gps_ptabxx, *gps_stabxx, and *headerxx.

mivis_rdhead, mivisfile [,/write_head]

`mivisfile`	Name of MIVIS image
`/write_head`	Write an ENVI standard header for this HYMAP file

Reads and assigns the default values for MIVIS header information to a MIVIS data file.

i) PROBE-1 functions

prob_imp, probefile [,/rightleft]

Reads a PROBE-1 data set acquired with no additional INS (parsing the *.log file).

prob_impcmig, probefile [,/rightleft]

Reads a PROBE-1 data set acquired with additional CMIGITS/INS data stream, parsed from *.log file directly (no additional file).

prob_imp04, probefile, gpsfile501, [,/rightleft]

Reads a PROBE-1 data set acquired with additional CMIGITS/INS data stream, parsing the CMIGITS *.501 file.

prob_rdhead, probefile [,/write_head, /rightleft]

Reads the header parameters of a ROSIS data set acquired in conjunction with the DLR IGI navigation system. Requires a valid ROSIS directory with files _bsq, _nav, _aux, _header for data import.

`probefile`	Standard Probe-1 data cube
`gpsfile501`	full *.501 data stream for Probe-1.
`/write_head`	Write an ENVI standard header for this PROBE file

`/rightleft`	Scan from right to left instead of left to right - use if raw data set is mirrored.

j) ROSIS functions

`rosis_imp, rosisfile`

Reads a ROSIS data set acquired in conjunction with the DLR IGI navigation system. Requires a valid ROSIS directory with files _bsq, _nav, _aux, _header for data import.

`rosis_rdhead, rosisdir [,envfile=..]`

`rosisdir`	Valid ROSIS data directory
`envfile`	Valid ENVI formatted ROSIS data cube (pre-converted).

Reads the header parameters of a ROSIS data set acquired in conjunction with the DLR IGI navigation system. Requires a valid ROSIS directory with files _bsq, _nav, _aux, _header for data import.

k) UAV spectrometers / Various import

`uav_imp, infile, navfile, syncfile, [/wide, /inverse, fov=.., smoothing=.., mount=..]`

`infile`	UAV raw imagery.
`navfile`	navigation file
`syncfile`	sync file name (gps file)
`/wide`	wide: wide FOV is used (49 deg); default is 16 deg.
`mount=..`	IMU mounting direction; format 'r+p+h+' of form roll/pitch/heading and sign
`fov=..`	total fov is entered as parameter to the import routine [deg]
`smoothing=..`	smoothing factor for auxiliary data (default: 5)

Reads a generic UAV data set and also some of the headwall hyperspec data formats based on standard data format as used for headwall photonics instruments.

`uav_rdhead, uavfile,`

`uavfile`	UAV spectrometer raw file.
`fov=..`	total fov is entered as parameter to the import routine [deg]

`/wide`	wide: wide FOV is used (49 deg); default is 16 deg.
`/inverse`	FOV is take right left instead of left right scanning for wide option.

Reads and initializes the default values from HYSPEX header information.

`various_imp, infile, navfile, syncfile, [fov=.., sensorfile=.., smoothing=.., mount=..]`

`infile`	UAV raw imagery.
`navfile`	navigation file
`syncfile`	sync file name (gps file)
`mount=..`	IMU mounting direction; format 'r+p+h+' of form roll/pitch/heading and sign
`fov=..`	total fov is entered as parameter to the import routine [deg]
`smoothing=..`	smoothing factor for auxiliary data (default: 5)
`mount=`	mounting of IMU, defauot: r-p+h+

Reads a generic data set and also some of the headwall hyperspec data formats based on standard data format.

4.4.3 DEM Import

`gc_rdarcdem, arcdem[, /nosave, /oldfmt, cord=.., missing=.., datyp=..]`

`arcdem`	Digital elevation model stored in arc/info GRID ASCII format.
`/nosave`	do not create an ENVI file after import (only to variables)
`/oldfmt`	Old ARCGRID format (no header)
`cord=..`	coordinate system to be assigned to file
`missing=..`	value to be assigned to missing pixels
`datyp=..`	output data typ (default: floating point); supported types: 1,2,3,4,12

Reads an ARC GRID (ASCII) formatted DEM to the variable *demarr* and assigns the structure *dem*; allows to reformroemat and save the result to ENVI DEM.

`gc_rddted, dtfile, outfile, /nosave, cord=..,missing=..`

`dtfile`	Digital elevation model stored DTED format
`outfile`	name of output envi file to be created
`/nosave`	do not create an ENVI file after import (only to variables)

`cord=..` coordinate system to be assigned to file
`missing=..` value to be assigned to missing pixels

Reads a DTED formatted DEM to the variable *demarr* and assigns the structure *dem*.

gc_rdenvdem, envidem, [/geographic]

`envidem` Digital elevation model stored in ENVI format
`/geographic` Will read geographic DEMs with unequal pixel sizes

Reads the DEM to the variable *demarr* and assigns the structure *dem*.

gc_rderdasdem, erdasfile, outfile

`erdasfile` digital elevation model stored in standard (uncompressed) ERDAS imaging file
`outfile` name of output envi file to be created

Reads the ERDAS-formatted DEM to the variable *demarr* and assigns the structure *dem*.

gc_rdglobdem, longrange, latrange, outdem, [globdem =.. /srtm, psize = .., /overwrite, utmzone = ..

`longrange` range in longitude direction; pixel edges
`latrange` range in latitude direction; pixel edges
`outdem` name of dem to be written
`globdem=..` name of global DEM (default: etc/GMTED2010.j2w)
`psize=..` output pixel size (default: 50m)
`/overwrite` overwrite the output DEM if it already exists.
`utmzone=..` force reading of global data to utmzone; southern zones are negative values.
`/srtm` force the use of the SRTM DEM etc/DEM_SRTM30_WORLD.DEM

Reads from the global DEM provided by parge (in etc/ directory of installation).

gc_rdgtifdem, gtifdem

`gtifdem` Digital elevation model stored in GEOTIFF standard format

Reads the DEM to the variable *demarr* and assigns the structure *dem*; allows to reformat the result to ENVI DEM.

rd_asciidem: dem=rd_asciidem(asciidem)

`asciidem`	digital elevation model stored as x/y/z triples
`/nosave`	do not create an ENVI file after import (only to variables)
`resol=..`	target resolution
`missing=..`	value to be assigned to missing pixels
`datyp=..`	output data typ (default: floating point); supported types: 1,2,3,4,12

Reads the DEM to the variable *demarr* and assigns the structure *dem*; allows to reformat the result to ENVI DEM.

rd_ltdhm: **dem=rd_ltdhm(ltdem)**

`ltdem`	Digital elevation model stored in proprietary Swiss L+T format

Reads the DEM to the variable *demarr* and assigns the structure *dem*; allows to reformat the result to ENVI DEM.

rd_pcidem: **dem=rd_pcidem(pcidem)**

`pcidem`	Digital elevation model stored in PCI data format (*.pix)

Reads the DEM to the variable *demarr* and assigns the structure *dem*; allows to reformroemat the result to ENVI DEM.

rd_usgsdem: **dem=rd_usgsdem(usgsdem [,head,chead])**

`usgsdem`	Digital Elevation 7.5’ model stored in USGS data format (*.dem), as distributed until 1998 (non-sdts data)
`head`	output structure which will contain the DEM header information
`chead`	output structure which will contain the DEM line header information

Reads the DEM to the variable *demarr* and assigns the structure *dem*; allows to reformat the result to ENVI DEM. Currently the 7.5 Minutes DEMs are distributed in the sdts format, which can be handled by ENVI and other shareware utilities. This format is not supported by PARGE directly.

gc_wdem, demname [, image, /general, bdnam=.., /export, /overwrite]

`demname`	name of ENVI DEM file to be written
`image`	2-dimensional array to be written (default: *demarr*)
`/general`	write a generic output file instead of a DEM
`bdnam=..`	enter ENVI name of the band to be written

`/export`	don't update the current DEM file name
`/overwrite`	overwrite a file without asking

Writes a DEM or a geocoded single band output to an ENVI formatted file.

4.4.4 DEM processing

`gc_crdem, resol, cord, hgs, xr, yr, demname = ..,`
`/auto, /gcps]`

`resol`	Target resolution of the DEM to be written.
`cord`	Coordinate structure for the new DEM.
`hgs`	Heights of the four corners given as four-element vector.
`xr`	Two-element vector containing the x-range of the new DEM
`yr`	Two-element vector containing the y-range of the new DEM
`demname`	name of DEM to be created
`/gcps`	use GCP heights for DEM creation
`/auto`	determine range of new DEM automatically from image extent.

Uses the available GCPs for creation of a DEM based on the given *dem* structure information. If no corners are available, all corner heights are set to the average height of the GCPs.

`gc_cutdem,xr,yr,resol,[demname,ele_in=ele_in])`

`xr`	two-element vector containing the x-range of the new DEM
`yr`	two-element vector containing the y-range of the new DEM
`resol`	target resolution of the resized DEM to be written.
`demname`	name of DEM file to be written.
`ele_in`	name of DEM file to be read from (if none is available in memory).

The current DEM is cut to the given range and simultaneously resampled to the new pixel size.

`gc_demtf, cord[, indem, resol, /interp, thresh=..,`
`/s_bilin, expand=.., coordinfo=.. outfile=..]`

`cord`	cordinate definition of output
`indem`	input dem (default: dem.name) in geographci Lat/Lon format
`resol`	resolution in meters (default 10m)
`/interp`	interpolate the missing values (see keywords below)

`thresh`	threshold value above which pixels should be interpolated (default: -1)
`/s_bilin`	use fast nereast bilinear resampling (note: s_bilin and expand ; can not be used at the same time)
`expand`	use fast nearest neighbor expansion by the given amount of pixels
`coordinfo`	will contain the information about the target coordinate information system.
`outfile`	name of ENVI DEM file to be written

Transforms a DEM in geographic lat/lon coordinates (WGS-84) to a metric system (e.g. UTM). Uses a triangulated bilinear interpolation by default.

`gc_doshade, sun, dl, [, shading=..]`

`sun`	Position of the sun: 3 dimensional unit vector [SunX,SunY,SunZ] or 2 dimensional vector [azimuth,elevation] in degrees.
`dl`	pixel size of DEM, eg., dem.psize[0].
`shading`	return an estimate of the illumination to a variable.

Calculates the cast shadow of the DEM using a fast cell gradient algorithm - only works with quadratic pixels!

`gc_gcpdem [, corners, /nogcp]`

`corners`	Height of the four corners given as four-element vector. The single elements are: [1] : origin, [2]: origin + delta-x, [3]: origin + delta-y, [4]: origin + delta-x/delta-y
`/nogcp`	Only the corner heights shall be used for DEM creation, corners have to be given, while the GCPs are ignored.

Uses the available GCPs for creation of a DEM based on the given *dem* structure information. If no corners are available, all corner heights are set to the average height of the GCPs.

`gc_makeilu, slpfile, zen, azi, ilufile`

`slpfile`	slope file (georectified, north oriented; ATCOR convention *_slp.bsq, together with *_asp.bsq)
`zen`	zenith angle [degrees]
`azi`	azimuth angle [degrees]
`ilufile`	name of output illumination file

calculates the ilumination using the equation:

ilu = cos(zen)*cos(slp)+sin(zen)*sin(slp) *cos(azi-asp).

and writes it to a file.

gc_prepele, in_ele, igmfile, dist [, out_ele, zen, azi, /fixed, /skyview, /shadow]

`in_ele`	Input DEM
`igmfile`	IGM fitting to DEM.
`dist`	smoothing distance for more accuarte slope/aspect calculation (in pixels)
`out_ele`	name of transposed elevation file
`zen`	Sun zenith angle (deg) for shadow option only
`azi`	Sun azimuth angle (deg) for shadow option only
`/skyview`	calculate skyview factor
`/shadow`	calculate cast shadow (requires zen/azi)

Prepares a DEM for atmospheric correction in IGM-based geometry. By default, slope and aspect files are calculated and inverted to raw geometry. Skyview and shadow files are produced on demand.

gc_skyview, aziint, eleint, dl, overs=..

`aziint`	interval in azimuth direction (degrees)
`eleint`	interval in elevation direction (degrees)
`dl`	pixel size of DEM, eg., dem.psize[0].
`overs`	oversampling factor to be applied to reduce processing time.

Calculates the skyview factor (*_sky) of a dem DEM - only works with quadratic pixels and may take some time for large DEMs (use an appropriate oversampling factor in that case).

gc_slpasp [, dist=..]

`dist`	Distance in pixels for slope/aspect calculation

Calculates the slope and the aspect for the currently selected DEM and stores the two files in ENVI format (*_asp and *_slp). Attention: existing slp/asp files of a DEM are overwritten - only works with quadratic pixels!

4.4.5 Auxiliary Data Handling

_att_filt:_ filtarr = att_filt(arr[, fac])

`arr`	parameter to be filtered (e.g., rollarr, pitcharr, navarr(0,*),...)
`fac`	factor for increased sensitivity to inconsistencies in the data. Default:1

Filters the attitude data from discontinuties. The higher the factor is, the more discontinuties are removed. The artefacts are found by comparing regular local variations in the parameter to the line-to-line variations.

_gc_borders:_ border = gc_borders(altitud=..)

`altitude`	average height of the ground in meters a.s.l.
`border`	double precision floating point array (3,n_l,2).

Calculates an estimate of the border coordinates of the whole image (for display of the 'Plot Area' function). An array with left, nadir, right border x/y coordinates of the image is returned. The minimum and maximum of this array may be used to determine the extent of an image

_gc_coordtf:_ coords= gc_coordtf(longlat, cord [, /headoff, coordinfo=..])

`longlat`	Geographic coordinates to be converted. Array of the dimensions [3,*], containing three columns for [longitude, latitude, altitude] (e.g. the array 'gpsarr'.
`cord.xx`	Structure defining the current coordinate system. The tags of 'cord' are:
`*.dtnum:`	datum number according to internal list of PARGE datum numbers (See "C: List of Datums" on page 266).
`*.proj`	Projections code: 0: Arbitrary/no conversion, 1: UTM 2: Gauss-Krüger 3: Gauss-Boaga 4: US State Plane 5: Swiss 6: Custom projection as defined by gc_mapproj
`*.utmzone:`	zone number - valid for proj=1,2,3,4 ; set to '' for default zone.
`*.ellnum:`	ellipsoid number according to internal list of PARGE ellipsoid numbers default: -1. (See "A: List of Ellipsoids" on page 264)
`*.datum:`	datum structure as read from the internal definition (tags: num, name, ell, dx, dy, dz, loc)

default: ‘’ (empty string)

`/headoff:` offsets the heading parameter by the difference between cartographic and geographic north.

`coordinfo=..:` Returns an informational string about the target coordinate system.

Converts geographic coordinates (longitude/latitude/altitude) to local orthometric coordinate system, and optionally fixes the heading parameter to local north.

gc_getcord: cord = gc_getcord(name, number, proj, zone=.., /parge])

`name`	name of projection (dem.coord for PARGE keyword); arbitrary if /parge keyword is not set.
`number`	number of projection (see Appendix C: List of Datums)
`proj`	parge projection number (see Appendix B: List of Projections)
`zone`	zone number (if required)
`/parge`	triggers usage of PARGE-internal naming.

Returns the ‘cord’ structure as compatible to gc_coordtf, cw_coord, etc. Compare Appendix for a list of valid datums and projections.

gc_headfix: headarr = gc_headfix(headarr)

`headarr`	aircraft true heading parameter

Fixes potential problems in heading parameters if heading is surpassing 360 degrees (generating a coninuous heading parameter).

gc_prepeng, attfile, parms, cols [, sync = sync, head=..]

`attfile`	Generic ASCII file containing attitude data - must be in columnar ASCII format (header rows followed by pure numeric data columns)
`parms`	5-element vector containing flags for the dimensions to be imported and updated; sync/roll/pitch/heading/yaw. e.g., dim = [1,1,0,1,0] updates roll and heading variables (`rollarr, headarr`) as well as syncarr if synchronization is performed
`cols`	column numbers in original file referring to the parameters which should be imported (numbering starts at 1, negative numbers refer to second option in GUI,i.e., for roll: left wing up, pitch: nose downs, heading: west positive)
`sync=..`	Post-synchronization keyword. Time has to be provided as one column in the raw data if this keyword is set.

	0: no synchronization, data are given exactly per line 1: synchronization by existing syncarr (time per image line) 2: synchronization by line number
`head=..`	Number of header rows in raw ASCII file (default: 0)

Reads and synchronizes external auxiliary data from a GPS receiver or after external coordinate transformation.

gc_prepnav, gpsfile, parms, cols [, sync = sync, llutm=.., head=..]

`gpsfile`	Generic GPS-file containing navigation data - must be in columnar ASCII format (header rows followed by pure numeric data columns)
`parms`	4-element vector containing flags for the dimensions to be imported and updated; sync/x/y/height. e.g., dim = [1,1,0,1] updates x and height dimensions of the variable navarr as well as syncarr if synchronization is performed
`cols`	column numbers in original file referring to the parameters which should be imported (starting at 0)
`sync=..`	Post-synchronization keyword. Time has to be provided with raw data if this keyword is set. 0: no synchronization, data are given exactly per line 1: starttime-frequency synchronization using decimal time per line 2: synchronization by existing syncarr (time per image line) 3: synchronization by line number
`llutm=..`	Direct conversion of the coordinates to the UTM system given the parameters in llutm (see function `gc_coordtf`)
`head=..`	Number of header rows in raw ASCII file (default: 0)

Reads and synchronizes external auxiliary data from a GPS receiver or after external coordinate transformation.

gc_sensmodel: **sensarr = gc_sensmodel([fov, ifov, scanp=.., /push, /cenfov, tilt=[..]])**

`fov`	total Field of View (FOV), default: sensor.fov
`ifov`	instantaneous Field of View, default: sensor.ifov (only required if keyword /cenfov is not set)
`scanp`	scan period of whole line as partial of along track sampling interval
`push`	standard pushbroom sensor model (default: whisckbroom model)
`tilt`	across and along track tilt of sensor model (two element array).

Creates a standard whisk- or pushbroom sensor model. This model should be loaded explicitely to the variable sensarr if known exactly.

gc_smoothaux, smfact

`smfact`	Factor so apply with smoothing

Smoothes all auxiliary data by given smoothing factors. Navigation data (navarr) is smoothed by 2 x smfact.

4.4.6 Ground Control Points Import

Most ground control tasks usually require user interaction for validation. Batch processing is therefore only possible if accurate ground control points are provided and if the sensor system is of high stability. A number of tools are given below:

gc_gcpimp: gcparr = gc_gcpimp(gcpfile [,/pci, /raw, /tnt, /txt, cord=..])

`gcpfile`	ENVI GCP file (*.pts)
`pci`	read pci-formatted data segment from .pix
`raw`	read exported PARGE-formatted file
`txt`	read pci-formatted ASCII file (stored as PLXY)
`tnt`	read tnt-formatted file
`cord`	coordinate conversion definition

Imports GCPs from a file into PARGE; doubled GCPs are omitted while existing GCPs are added to the imported new GCPs. By default, ENVI formatted GCP-file is assumed.

gc_flipzone, borefile, cord

`borefile`	boresight status file
`cord`	cord: current coordinate system.

Fix heading parameter if boresight alignment was done in different coordinate system.

gc_autogcps, cube, ortho , [mapfile=.., pixshift=.., magni=.., refbands=[cr,or], resol=.., patchsize=..]

`cube`	rectified image cube (with some offset to reality)
`ortho`	orthophoto file at similar location
`mapfile=..`	map file to obtain x/y pixel coordinate from.
`pixshift`	maximum distance to be considered

`magni`	magnification for subpixel accuracy (maximum: 5)'; recommended: 1 (pixel accuracy).
`resol`	patch size in pixels (default:100)
`refbands`	reference bands to be used for image matching (numbering starting at 1); default: green bands are taken at 560 nm. Two-element arry for the two files
`patchsize`	size in pixels of patches to analyse for each GCP (default=resol)

Find gcps from image mapping to (close) orthophoto.

Subroutines to gc_gcpimp:

gc_getenvgcp: gcparr = gc_getenvgcp(gcpfile)

`gcpfile`	ENVI GCP file (*.pts)

Imports ENVI GCPs from a file to a variable.

gc_getpcigcp: gcparr = gc_getpcigcp(pcidsk, dseg)

`pcidsk`	PCI/Geomatica *.pix file
`dseg`	data segment number containing the GCPs (default is 2).

Imports PCI GCPs from a file to a variable.

gc_gettndgcp: gcparr = gc_gettntgcp(gcpfile)

`gcpfile`	TNT/MIPS GCP file (*.txt)

Imports TNT/MIPS GCPs from a file to a variable.

gc_gcptf: gcparr = gc_gcptf(gcparr, cord)

`gcparr`	currently loaded GCPs
`gcparr`	currently loaded GCPs

Converts the loaded GCP coordinates from long/lat to metric coordinate system as defined in 'cord'.

gc_guesspos: coord = gc_guesspos([pix,lin])

`pix`	pixel position of GCP
`lin`	line position of GCP

Guesses the possition of one GCP based on its image coordinates (pixel/line) and the currently defined DEM and attitude parameter.

4.4.7 Ground Control based Offsetting

gc_impoffs, statusfile[, cord, /atti, /navi, /hedi]

`statusfile`	previous status file of same data acquisition

`cord`	current coordinate system definition
`/atti`	update attitude parameter
`/navi`	update navigation parameter
`/hedi`	update heading parameter with offset due to differing zone/coordinate system between calibration run and scene to process.

Import the offset from other status file and apply to currently imported parameters (ie., roll/pitch/heading offsets). Use this function to process a series of flight lines of the same data acquisition.

gc_fetchoffs: **`gc_fetchoffs(offs, boresightfile[, /text, /hedi])`**

`boreightfile`	boresight status file
`/text`	load boresight angles from text file with coding roll=.., pitch, head, yaw, xnav, ynav, alt for all boresights.
`/hedi`	load gcord into offset structure for later heading conversion.

Import the offset from other status file.

`gc_optatt, [/nohead , niter = xx]`

`/nohead`	only optimize roll and pitch
`niter=xx`	number of iterations (default: 2)

Optimizes the offsets of roll/pitch and heading for the currently available GCPs and apply directly to the respective parameters.

gc_opthigh: **`dhigh = gc_opthigh(gcplist)`**

Calculates an estimate of the altitude offset based on the currently selected GCPs - this offset may be applied to navarr(2,*) subsequently.

gc_opthead: **`dhead = gc_opthead(optstep, gcplist=..)`**

Calculates an estimate of the heading offset based on the currently selected GCPs - this offset may be applied to headarr subsequently.

`gc_resetoffs, [/atti , /navi]`

`/atti`	apply to attitude data
`/navi`	apply to navigation data

Resets the currently applied offsets to the original state by adding to the parameters and setting the offset meta information to zero.

gc_testgcps: `offs = gc_testgcps(gcplist)`

Calculates the roll/pitch/x/y offsets of all current GCPs and returns an array of all of them (including coordinates) as displayed in the offsets dialog.

4.4.8 Special

The special menu commands are not of standard character. Anyhow they may be used in some cases in batch scripts. The main tools are

`bip_2_bsq, infile, outfile`

`infile`	Input ENVI image file (must be in BIP format; Band Interleaved by Pixel)
`outfile`	Output file name; ENVI header is created automatically.

Transforms an ENVI BIP file to ENVI BSQ format. This function may be very time consuming, but is highly recommended if BIP data needs to be processed.

`bil_2_bsq, infile, outfile`

`infile`	Input ENVI image file (must be in BIL format; Band Interleaved by Line)
`outfile`	Output file name; ENVI header is created automatically.

Transforms an ENVI BIL file to ENVI BSQ format. This function is required if BIL data needs to be processed by PARGE.

`gc_cutraw, xr, yr, cr [, rawname]`

`xr`	pixel range in x direction
`yr`	pixel range in y (line) direction
`cr`	range in band direction (spectral subset)
`rawname`	output file name of the resized image cube (default: *_sub).

The raw data is cut to the desired size, while resizing all parameters including the sensor model and the attitude data streams in the structure of the geocoding package.

gc_pathinit: `navarr = gc_pathinit(startpt, endpt, n_l)`

`startpt`	starting point coordinates [line,x,y,z]
`endpt`	ending point coordinates [line,x,y,z]
`n_l`	total number of lines

Initializes a straight flightpath from two knwon points if navigation data are completely missing. Use before doing interpolations of the *navarr*-parameter.

gc_splinav: **navarr = gc_splinav()**

Interpolates a flightpath (x/y coordinates) from all GCPs with flag set to '2' using a cubic spline interpolation. The flight altitude needs to be predefined for that procedure.

gc_optpitch, step
gc_optroll, step

`step`	optimization step of roll/pitch in radians

Optimize the roll or pitch offset while interpolating the flightpath iteratively from all GCPs with flag set to '2'. The flight heading parameter is also updated.

gc_calcyaw [, mmd,/head,/getmmd,/decl]

`mmd`	side output of main mass direction (MMD)
`/getmmd`	only get mmd without updating yaw/heading parameters
`/head`	update the parameter headarr instead of yaw only
`/decl`	use the declination parameter (dem.decl) as offset for calculation

Read the yaw data and transform it to heading data using flightpath navigation data; if no yaw data are available, the heading is derived from the navigation data only.

gc_cresca, reffile, startpoint, endpoint, fov, rawdims scafile = scafile

`reffile`	Reference file for output coordinate system and dimensions
`startpoint`	first point of flightpath (x-y - pixel coordinates in output geometry; z altitude)
`endpoint`	last point of flightpath (x-y - pixel coordinates in output geometry; z altitude)
`fov`	total FOV of image
`rawdims`	n_p/n_l of raw image as two element array (default: image.n_p/ image.n_l)
`scafile`	name of scan angle output (*.sca)

Calculate a scan angle file from parameters, adding a 10% margin. Startpoint and enpoint may be outside of the output file dimensions.

gc_glttosca, gltfile, fov, heading, alt, scafile = scafile

`gltfile`	ENVI GLT or PARGE MAP file as a reference
`fov`	total FOV of image
`heading`	average heading of flight (degree)
`alt`	average altitude of aircraft above ground (meters)
`scafile`	name of scan angle output (*.sca)

Calculate a scan angle file from GLT or MAP.

`gc_rolpit [, /spline,/roll,/pitch]`

`/spline`	use spline instead of linear interpolation
`/roll`	interpolate the roll parameter only
`/pitch`	interpolate the pitch parameter only

Interpolate roll and pitch from gcps.

`gc_scalerp [,/roll,/pitch]`

`/roll`	scale the roll parameter only
`/pitch`	scale the pitch parameter only

Scales roll/pitch while minimizing the residual errors on the GCPs.

`gc_driftcor, auxid [,/spline]`

`auxid`	ID for the parameter to be corrected from drift influences:0:rollarr, 1:pitcharr, 2:navarr(0,*), 3:navarr(1,*)
`/spline`	spline interpolation is used instead of linear approach

Corrects a potential drift of the auxiliary data based on ground control points. Will not work if no GCPs are available.

`gc_reset [,/roll,/pitch,/heading,/yaw]`

`/xxx`	reset the respective parameter

Resets the respective parameters to zero and the correct dimensions.

`gc_getroll:` `newroll = gc_getroll(bandimg[, nsmooth=..])`

`bandimg`	single raw band image array to be used for roll calculation
`nsmooth`	number of lines to be used for smoothing of result (default: 250)

This function calculates an estimate for the roll parameter (`newroll`) using a line-by-line correlation analysis of one image band.

gc_rollcomp, height, chlist, outfile, /nav, /flip

`height`	average height of aircraft above ground
`chlist`	list of the channels/bands to be corrected
`outfile`	name of the output roll compensated image file
`/nav`	the navigation data is used for the height per line (if available)
`/flip`	flips the output vertically

Compensates for roll movements by shifting the single lines across track depending on the roll and height above ground.

gc_postsync, [nsmooth=.., /silent]

`nsmooth`	number of lines used for smoothing the roll parameter for correlation analysis.
`/silent`	apply the sync offset automatically to all data (without asking...)

Corrects a potential synchronisation offset using the image based roll parameter.

gc_shiftpars, pshift, /gps

`pshift`	shift in number of pixels (positiv: in flight direction)
`/gps`	also shift the navigation data (default: attitude only)

Shifts the attitude and optionally navigation parameters by a number of lines (or sub-lines) to fix synchronization problems.

4.4.9 Quality Control (Menu 'QC')

gc_sigtonoise, infile, outfile, maskfile, chlist

`infile`	image to be processed (ENVI format)
`outfile`	ASCII output containing SNR results
`maskfile`	ENVI file containing valid pixels (value greater than 0)
`chlist`	list of bands to be processed

Image based SNR estimate calculation based on high-pass filtering of homogeneous areas.

gc_checkigm: **stat = gc_checkigm(igmfile)**

`igmfile`	IGM file to be analyzed

The statistics of the IGM coordinate distances are calculated and stored in a data structure. All results are in meters:

`stat.psize_along:`	along track pixel size (mean of all distances)
`stat.sdev_along:`	standard deviation of along track pixel sizes
`stat.psize_across:`	across track pixel size (mean of all distances)
`stat.sdev_across:`	standard devaition of across track pixel sizes

gc_dataloss: **loss = gc_dataloss(rawfile,mapfile)**

`rawfile`	image to be processed (ENVI format)
`mapfile`	map-file corresponding to rawfile

The relative data loss due to nearest neighbour resampling is returned in a structure with number of pixels, number of lost pixels and percentages of lost pixels.

gc_imoffset, image1, bnd1, image2, bnd2 [,outfile, resol = .., meanoff = ..]

`image1`	reference image to be processed (ENVI format)
`bnd1`	band number within image1
`image2`	test image to be processed (ENVI format)
`bnd2`	band number within image2
`outfile`	file containing the offsets in x/y/average as IDL save set.
`resol=..`	patch size (default: image_size/8)
`meanoff=m`	will contain the overall mean offset for the two images.

Image correlation analysis of two images.

gc_chessboard, image1, bnds1, image2, bnds2 [,outimage, resol = ..]

`image1`	reference image to be processed (ENVI format)
`bnds1`	band numbers within image1
`image2`	test image to be processed (ENVI format)
`bnds2`	band numbers within image2
`outimage`	file containing the chessboard images (ENVI).
`resol=..`	chessboard patch size (default: image_size/8)

Calculates a chessboard style overlay image for evaluation purposes.

4.4.10 Processing

`gc_process infile, outfile, pixelsize, demfile=.., resample=.., lines = lines, bandrange = bandrange`

`omfile`	image to be processed (ENVI format)
`outfile`	name of output file to be created (orthorectified output *_geo)
`pixelsize`	output pixel size
`resample=..`	type of resampling: 'nearest' or 'bilinear'or 'nearbilin' - default bilinear
`demfile=..`	name of DEM to be used (default to global DEM).
`lines = .`	`r`range of the lines to be calculated int he form [first line, last line], num
`bandrange = ..`	range of the bands to be processed [first band, last band], num

This module is used to process an image directly after data import. It includes the handling of the DEM, the automatic size detection, the IGM calculation and the final cube processor and the quickview generation.

`gc_maini [, lines = lines, /atmo, /dist]`

`lines = ..`	range of the lines to be calculated int he form [first line, last line], numbering starts at 1.
`/expand`	expands the mapping array by 3 pixels in maximum
`/triang`	triangulates the central pixels (priority against expanding)
`/atmo`	writes the scan angles to a special file for atmospheric correction
`/dist`	include pixel distance in scan angle file as fourth layer

Direct geocoding of the single band image in raw acquisition geometry. Creates an image geometry map only. This is the recommended main processor.

`gc_main [, lines = lines, /expand, /triang, /atmo, /igm, /dist]`

`lines = ..`	range of the lines to be calculated int he form [first line, last line], numbering starts at 1.
`/expand`	expands the mapping array by 3 pixels in maximum
`/triang`	triangulates the central pixels (priority against expanding)
`/atmo`	writes the scan angles to a special file for atmospheric correction
`/igm`	write an ENVI image geometry map (*igm)
`/dist`	include pixel distance in scan angle file as fourth layer

Calculates the mapping array containing the pixel positions in the DEM geometry (*_map). This is the old default main processing step.

gc_mainl [, lines = lines, /direct, /expand, /triang, altitud = .., /atmo, igm=[1/2], /dist]

`lines = ..`	range of the lines to be calculated int he form [first line, last line], numbering starts at 1.
`/expand`	expands the mapping array by 3 pixels in maximum
`/direct`	geocodes the single band image directly without creating a mapping array first
`/triang`	triangulates the central pixels (priority against expanding)
`altitud=..`	average height of the ground in meters. if altitude is not set, the mean height as of the DEM (*demarr*) is taken as elevation value.
`/atmo`	writes the scan angles to a special file for atmospheric correction
`igm=..`	write an ENVI image geometry map (*igm). For igm=2, a flat geocoding is performed using the alitude parameter without the need of a DEM (see keyword *altitud* above)
`/dist`	include pixel distance in scan angle file as fourth layer

Performs the parametric geocoding on a flat surface (fast 'low level' processing).

gc_maind [, lines = lines, /expand, /triang, /atmo]

`lines = ..`	range of the lines to be calculated int he form [first line, last line], numbering starts at 1.
`/expand`	expands the mapping array by 3 pixels in maximum
`/triang`	triangulates the central pixels (priority against expanding)
`/atmo`	writes the scan angles to a special file for atmospheric correction

Direct geocoding of the single band image directly on the DEM. Produces the same output as `gc_main`, but does not create a mapping array first. This routine is not suited for linear interpolation.

gc_band: band = gc_band()

`band`	variable containing geocoded band

Geocodes the loaded single band image based on the actual mapping file (result.remapar-r_name). No interpolations can be applied using this procedure. Also, the routine is not functional with IGM based processing. Use gc_rgb with one band number given instead if interpolated results are required.

gc_cube [, chlist, ofile = .., mask=.., /nohdr, /flip, linear=.., /bilinear, /triang]

`chlist`	list of all channel numbers to be geocoded (default: no selection), integer array with channgel numbers starting at 1 (except a zero is given as part of the chlist).
`ofile`	output file name optional (default is result.geocube_name)
`mask`	value to assign to missing pixels outside of the image area
`/nohdr`	inhibits to write an ENVI header.
`/flip`	flips the output vertically; required for certain tiff representations
`linear=..`	linear resampling is applied to the geocoded images for the filling of missing values. 1: Along track linear resampling 2: Across track linear resampling
`/bilinear`	Nearest bilinear resampling is applied to the geocoded images for the filling of missing values.
`/triang`	Triangulation resampling is applied to the geocoded images for the filling of missing values (ATTENTION: this option is very memory intensive!)

Geocodes a whole cube to an ENVI formatted file, applying optional resampling.

gc_cube_ip [, chlist, ofile = .., mask=.., /nohdr, /flip, limits=.., /centers, /nearest, mapfile=.., /binning]

`chlist`	list of all channel numbers to be geocoded (default: no selection), integer array with channgel numbers starting at 1 (except a zero is given as part of the chlist).
`ofile`	output file name optional (default is result.geocube_name)
`mask`	value to assign to missing pixels outside of the image area
`/nohdr`	inhibits to write an ENVI header.
`/flip`	flips the output vertically; required for certain tiff representations
`limits=..`	6 elements vectors containing x/y coordinates of lower left and upper right edge of output and resolution
`/centers`	Center pixels are preserved and only holes are interpolated.
`/nearest`	Use gridded nearest neighbor interpolation
`mapfile=..`	Use a map-file as reference for dimensions; limits keyword has no effect if mapfile is set.
`/binning`	apply automatic spatial binning to the process for oversampled imagery.
`/triang`	Triangulation resampling is applied to the geocoded images for the filling of missing values (ATTENTION: this option is very memory intensive!)

Geocodes a whole cube to an ENVI formatted file, triangulation and true bilinear resampling.
Requires the IGM output for processing.

gc_cube_tile [, chlist, ofile = .., /flip, n_tiles=.., background=.., limits=.., exclude=.., /nearest, /natural, /binning]

`chlist`	list of all channel numbers to be geocoded (default: no selection), integer array with channel numbers starting at 1 (if a zero is given as part of the chlist, numbering starts at 0).
`ofile=..`	output file name optional (default is result.geocube_name)
`/flip`	flips the output vertically; required for ENVI outputs; not required for certain tiff representations.
`n_tiles=..`	number of tiles for the processing in along track direction (to reduce memory requirements)
`background=..`	lvalue for background pixels
`limits=..`	6 elements vectors containing x/y coordinates of lower left and upper right edge of output and resolution
`exclude=..`	lower limit for exclusion of large triangles (size in pixels)
`/nearest`	Nearest neighbour resampling is applied to the geocoded images.
`/natural`	Natural neighbour resampling is applied to the geocoded images (ATTENTION: this option is very memory/time -intensive!)
`/binning`	apply automatic spatial binning to the process for oversampled imagery.

Geocodes a whole cube to an ENVI formatted file, by default using bilinear resampling.
Requires the IGM output for processing.

gc_cube_ntile [, chlist, ofile = .., n_tiles=.., mask=.., /triang, /bilinear, /overwrite, /binning]

`chlist`	list of all channel numbers to be geocoded (default: no selection), integer array with channel numbers starting at 1 (if a zero is given as part of the chlist, numbering starts at 0).
`ofile=..`	output file name optional (default is result.geocube_name)
`n_tiles=..`	number of tiles for the processing in along track direction (to reduce memory requirements)
`mask=..`	lmask background pixels and assign this value to them.
`/bilinear`	nearest bilinear resampling is applied to the geocoded images for the filling of missing values. overwrite

`/triang`	triangulation resampling is applied to the geocoded images for the filling of missing values (ATTENTION: this option is memory intensive!)
`/binning`	apply automatic spatial binning to the process for oversampled imagery.
`/overwrite`	silently overwrites the output if existing

Geocodes a whole cube to an ENVI formatted file in full tiling mode, by default using maparr-based nearest neighbour resampling. The routine requires the IGM output and a valid map file for processing.

`gc_cubejoin, status1, status2, limits, outfile, [splitwvl = .., /near1, /near2, /mosaic, /clip, dist = .., n_tiles = ..`

`status1`	fully qualified status of the first cube to be processed.(all related inputs are taken from the status files)this files rules the dimensions
`status2`	fully qualified status of the second cube to be integrated with the first cube.
`limits=..`	6 elements vectors containing x/y coordinates of lower left and upper right edge of output and resolution
`outfile=..`	output file name optional (default is result.geocube_name)
`splitwvl=..`	wavelength to split the two cubes. It uses the ENVI header information to split; the same units as in ENVI header should be used here (normally: nm).
`/near1`	apply nearest neighbour resampling to first image (default: bilinear)
`/near2`	apply nearest neighbour resampling to 2nd image (default: bilinear)
`/mosaic`	create a mosaic instead of a joined cube (same number of bands required in both cubes!). The first image will be in front.
`/clip`	clips the extent of the images to the range covered by both images for a joined cube (is void if /mosaic is set)
`dist=..`	Distance of interpolation/gap filling in pixels
`n_tiles=..`	number of tiles for the processing in along track direction (to reduce memory requirements)

`gc_cube_rot [, chlist, angle, resolution, ofile = .., n_tiles=.., lines=.., mask=.., limits=.., /nearest, dist=.., /binning]`

`chlist`	list of all channel numbers to be geocoded (default: no selection), integer array with channel numbers starting at 1 (if a zero is given as part of the chlist, numbering starts at 0).
`angle`	Output rotation angle (deg) - main direction of image axis with respect to north (east: 90 deg); default: mean(headarr)
`resolution`	output resolution in meters

`ofile=..`	output file name optional (default is result.geocube_name)
`n_tiles=..`	number of tiles for the processing in along track direction (to reduce memory requirements)
`lines`	range of image lines/frames to process (from/to), starting at 1.
`mask=..`	lvalue for background pixels
`limits=..`	4 elements vectors containing x/y coordinates of lower left and upper right edge of output to be processed (in non-rotated coordinates)
`dist=..`	lower limit for exclusion of large triangles (size in pixels), distance in pixels to be considered
`/nearest`	Nearest neighbour resampling is applied to the geocoded images (default: bilinear)
`/binning`	apply automatic spatial binning to the process for oversampled imagery.

Geocodes a whole cube to an ENVI formatted file in a rotated geometry, by default using bilinear resampling. Requires the IGM output for processing.

gc_finalwarp, cube, splitband, ocube[, pixshift=.., magni=.., refbands=[vb,sb], offsets= .., resol=..]

`cube`	rectified image cube (with some offset to reality)
`ocube`	output cube name
`pixshift`	maximum distance to be considered
`magni`	magnification for subpixel accuracy (maximum: 5)'; recommended: 2 (0.25 pixels accuracy).
`resol`	patch size in pixels (default:100)
`refbands`	reference bands to be used for image matching (numbering starting at 1); default: NIR bands are taken at 850nm for VNIR detector and at 1050nm for SWIR detector. Two-element array for the two detectors

Warps the spectral part of an image (e.g. the SWIR part) onto the first part (e.g., the VNIR part).

gc_rgb, chlist, [, outfile, mask=.., /equal, /line, unilinear=.., /bilinear, /triang, /envi, /jpeg, /binning]

`chlist`	list of the 3 channels/bands to be corrected for RGB image creation
`outfile`	name of the output-file (default: `rgb_[chlist].tif`)
`mask`	value to assign to missing pixels outside of the image area
`/equal`	histogram equalization is used to scale the single channels of the image
`/line`	optimized linear scaling is used to scale the single channels of the image

`unilinear=..`	linear resampling is applied to the geocoded images for the filling of missing values. 1: Along track linear resampling 2: Across track linear resampling
`/bilinear`	Nearest bilinear resampling is applied to the geocoded images for the filling of missing values.
`/triang`	Triangulation resampling is applied to the geocoded images for the filling of missing values (ATTENTION: this option is very memory intensive!)
`/tribin`	Full bilinear interpolation is applied (set tribin to 2 to preserve center pixels)
`/binning`	apply automatic spatial binning to the process for oversampled imagery (only for /tribin opption available).
`/envi`	The file is written to an ENVI formatted image (default: TIFF)
`/jpeg`	The file is written to a JPEG formatted image

Geocodes one or three arbitrary bands out of the whole image to an image file. By default a 24 bit TIFF image is written.

gc_invert, demfile, igmfile, chlist, [ofile=.., /nearest]

`demfile`	name of file in DEM/Map geometry to be inverted
`igmfile`	name of IGM file used for inversion (same coordinate system as demfile)
`chlist`	list of bands to process
`ofile`	output file name (default: *_i.*).
`nearest`	use nearest neighbor instead of bilinear interpolation for resampling

Inverts data in cartographic map/DEM geometry to raw acquisition geometry.

gc_cremap, igmfile, mapfile, n_tiles=.., limits=.. expand=..

`igmfile`	name of IGM file
`mapfile`	name of MAP file to be created
`expand`	number of pixels to be expanded
`n_tiles`	number of tiles for the processing (to reduce memory requirements)
`limits=..`	6 elements vectors containing x/y coordinates of lower left and upper right edge of output and resolution

Creates a MAP (similar to GLT) from IGM (suitable for fast nearest neighbour resampling and used in conjunctions with gc_cube_ntile).

gc_maptoglt, mapfile, gltfile

`mapfile`	name of MAP file to be converted
`gltfile`	name of GLT file to be created

Converts a PARGE *_map file to a GLT file to be used in ENVI. The major difference being the coding of filled pixels and the numbering; ENVI files are startet with pixel 1/1 whereas PARGE uses 0/0 for the first pixel.

env_mosaic, filelist, outfile[, psize, [xr, yr,] chlist=.., background=..,/overwrite]

`mapfile`	name of MAP file to be converted
`outfile`	name of GLT file to be created
`filelist:`	list of files to mosaic
`psize`	pixel size of output
`xr`	range in x dimensions (edges of pixels, two elements [xmin,xmax])
`yr`	range in y dimensions (edges of pixels, two elements [ymin,ymax])
`chlist=..`	list of bands to mosaic (starting at zero 0).
`background=..`	value of background pixels (in and out)
`/overwrite`	overwrites the output without asking...

Used to mosaic a series of images to a common reference grid as declared. If yr is not set correctly, the full extent is taken for both xr and yr. If psize is not set, the minimum pixel size of all images is used.

env_resize, infile, outfile[, ncols, nrows, mtype, /pixels, missing=.., back=..]

`infile`	input file
`outfile`	output file
`ncols`	number of columns of output
`nrows`	number of rows of output

`mtype`	ENVI mtype coordinate string or (if keyword pixel is set), upper left coordinates in pixel coordinates.
`/pixels`	use pixel coordinates
`missing=..`	replace missing parts with this value.
`/back`	tacke care of background pixels value 0 properly

Resizes spatially to new dimensions by bilinear or nearest neighbour interpolation.

Chapter 5:

Functions Reference Guide

5.1 Generic Menu Elements

5.1.1 The PARGE Menu

The PARGE main menu is used for interactive operation of the software. It consists of 8 major parts (see Figure 5.4) which are described beginning with Section 5.2 on page 125. The workbench structure is such that menu tasks are usually applied from left to right and top-bottom where applicable.

5.1.2 Help System

PARGE features a context sensitive online help system, which can be accessed by using the >Help< Button in the respective function windows. An ASCII text describing the major elements of the current window will be displayed. The original files of this texts can be found in the directory 'parge_help' of the software distribution.

In addition to this help files, some generic help texts are accessible directly through the **Help** menu of the PARGE main menu. There, you find the information about data structure, licensing information, as well as generic workflow tips, but the tools to bring you installation up to date.

5.1.3 Text Editing

Any ASCII formatted data file or description may be edited directly through the PARGE built-in small text editor. The editing tool is a convenient way to browse and edit an ASCII file on the current working directory (e.g to look at an ENVI Header or at some ASCII auxiliary data), but also to check auxiliary data streams.

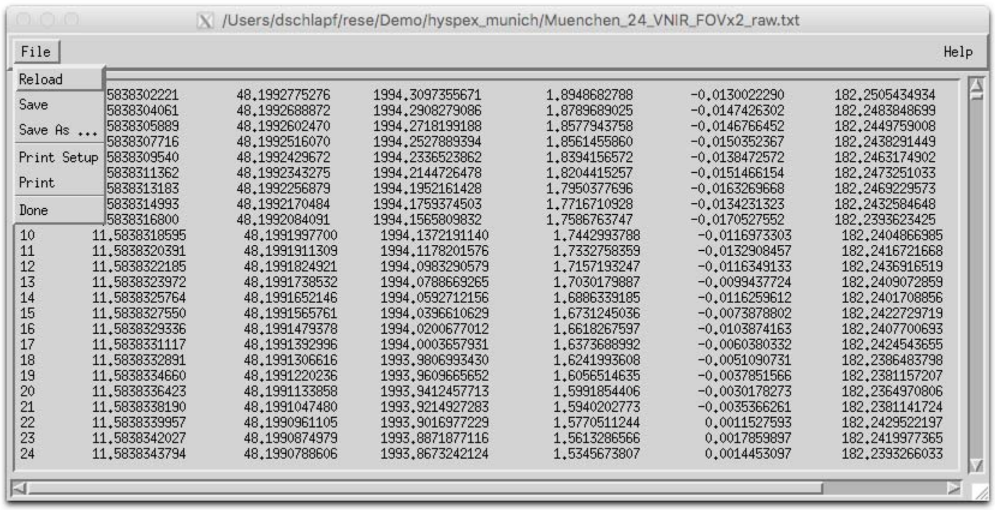


Figure 5.1: Simple PARGE text editor, invoked by 'Show/Edit Textfile'

Actions:

Reload:	Reloads the currently displayed file from disk,
Save:	Saves changes to the file
Save As:	Saves the file to a different name
Print Setup:	Sets up the printer depending on your operating system
Print:	Prints the file

Attention: While printing, files of multiple pages are separated into a series of print jobs with one page per print job. This may cause problems for large files since your printer queue may be overloaded. Please use dedicated text processing routines for printing large text files.

5.1.4 Plotting

The PARGE standard plots are displayed in a resizable and printable standard plot window. It offers the following options:

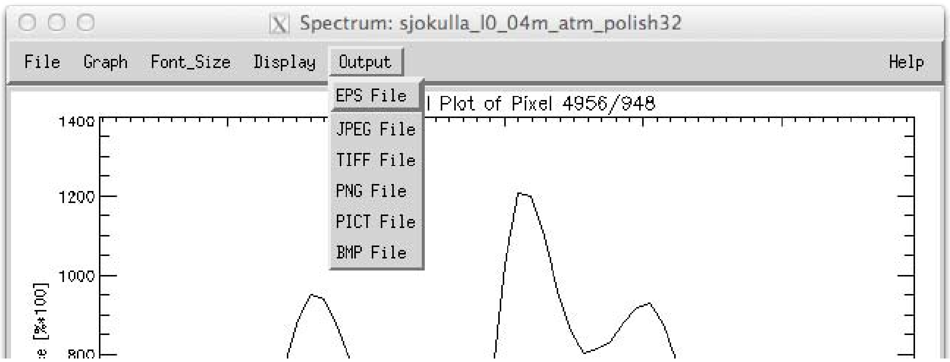


Figure 5.2: PARGE plotting window standardmenu.

Functional Details of the Menus

- *File*: display of underlying ASCII data, link to ascii file display, export to text (TXT) and spectral library (SLB) files, Printer Setup and Printing (colors may be inverted for black background)
- *Graph:* change title and range of axes of plot
- *Font_Size:* the display font size is changed to the selected number (approximately).
- *Display:* reloading the display will redraw the same plot, on the menu driven resizing the size of the plotting window may be set explicitely (in cm).
  Color tables may be loaded and adapted (applicable to the whole PARGE session).
- *Output*: The same plot as displayed can be written to a vector EPS file or to one of the available formats of rasterized files.

Note: The plot is redrawn from scratch after each resize action on the window.

5.1.5 Image Display

A standard routine for image display is used in PARGE as described in the function *File:Display ENVI file* on page 126 is provided within PARGE and also in the free accompagning software GLIMPS. It allows to display RGB images within PARGE, scaling and export of images, extracting spectra, and easy image manipulation. It is recommended to use standard image processing software systems for fully-fledged image analysis.

5.1.6 Coordinate Conversion

Whenever coordinates are provided to parge, therea re to be converted to a metric coordinate system. Finally, all coordinates have to be in the same geodetic system (projection, ellipsoid, geodetic datum) prior to processing the data. This is valid for the DEM, the GCPs, and the x/y/z flightpath. If longitude/latitudes are provided (in decimal degree) they need to be converted to a metric geographic coordinate system using the PARGE standard coordinate converter (see Figure 5.3) or by an external tool.

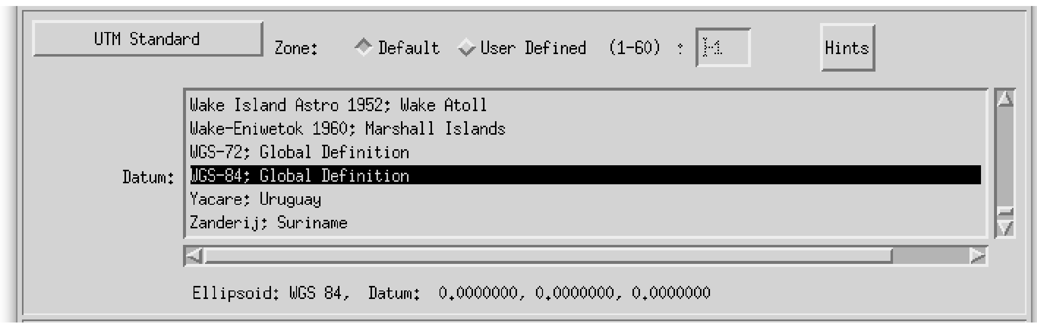


Figure 5.3: Coordinate conversion standard method.

Details

- If > NO CONVERSION< is selected in the first pull down menu, no coordinates are converted. This options is required if the raw data are already in converted format or if conversion shall be done elsewhere.
- If the dialog is meant for coordinate assignment, the 'Arbitrary' coordinate system may be selected for systems not supported by PARGE.
- Gauss/UTM: Principally, UTM, Gauss-Krueger, and Gauss-Boaga system follow the same principle despite the scaling constant.
- “> Custom Projection <“ uses the projection as set by the functioon **Edit:Define Map Projection** p.160 (out of the IDL-supported projections, compare page 160).
- Default Zone: By default the zone is calculated from the first data point of a flight path. (e.g., from the DEM if available) and then is updated from the value as entered in the data entry field 'Zone'.
- User defined zone: the zone number may be entered explicitely, e.g. for compatibility to adjacent data. This applies to UTM and Gauss only.

About the Datum:

A list of worldwide datums is available, choose the one as appropriate to your DEM and image data. The line below the datum selector gives you more information about a selected datum.

Formats Conventions

The input coordinates have to be in decimal degree (long/lat) for the conversion.

- East longitudes are positive, west longitudes are negative
- North latitudes are positive, wouth latitudes are negative

5.2 Menu: File

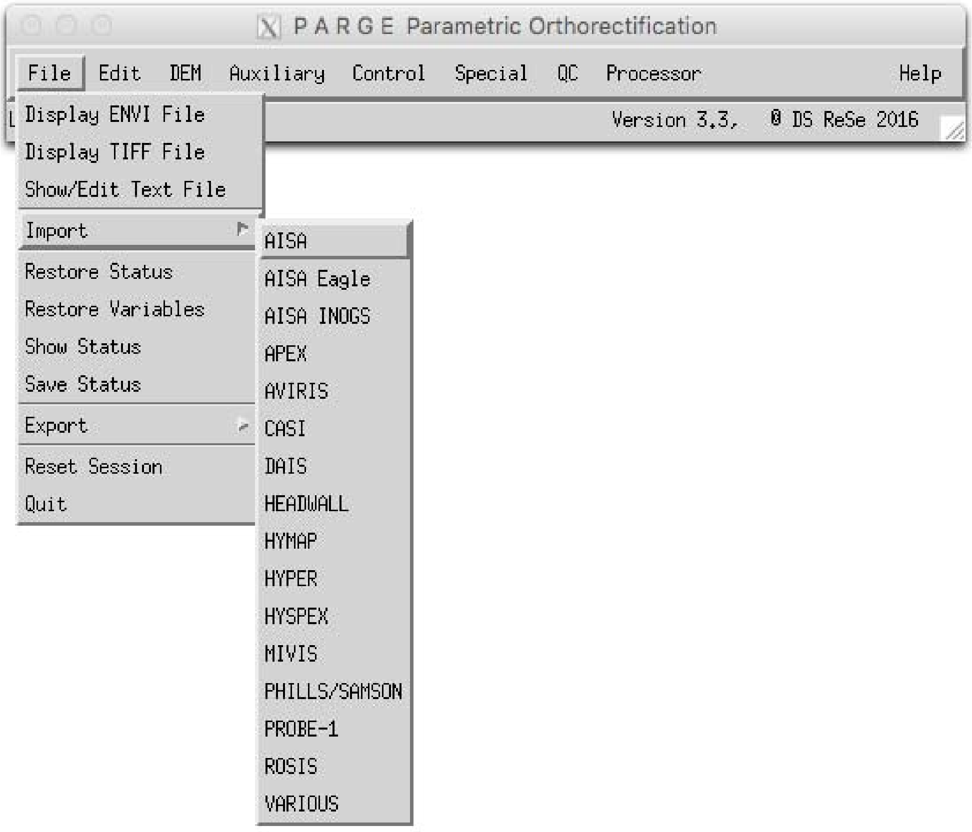


Figure 5.4: File menu, import function, and PARGE main menu.

The menu **'File'** deals with the raw remote sensing data import and image viewing. Both, predefined sensor filters as well as the import of data from new sensors is supported. If a predefined sensor is imported using the **'Import'** function, all auxiliary data are usually imported as well. If, alternatively, a raw data cube is assigned in the '**Edit**'-menu, the parameters (GPS/Attitude/Sensor Model) need to be defined manually later, using the main menu '**Auxiliary**'.

DISPLAY ENVI FILE

PARGE includes a complete ENVI file display routine for data browsing and extraction of spectra and profiles. The file display works for all interleave formats, i.e., BIL/BIP/BSQ. The below description applies also to similar display rotuines (e.g., the functions **Display DEM** p.176, **Display Single Channel** p.155, **Direct Display:Georectified RGB** p.253).

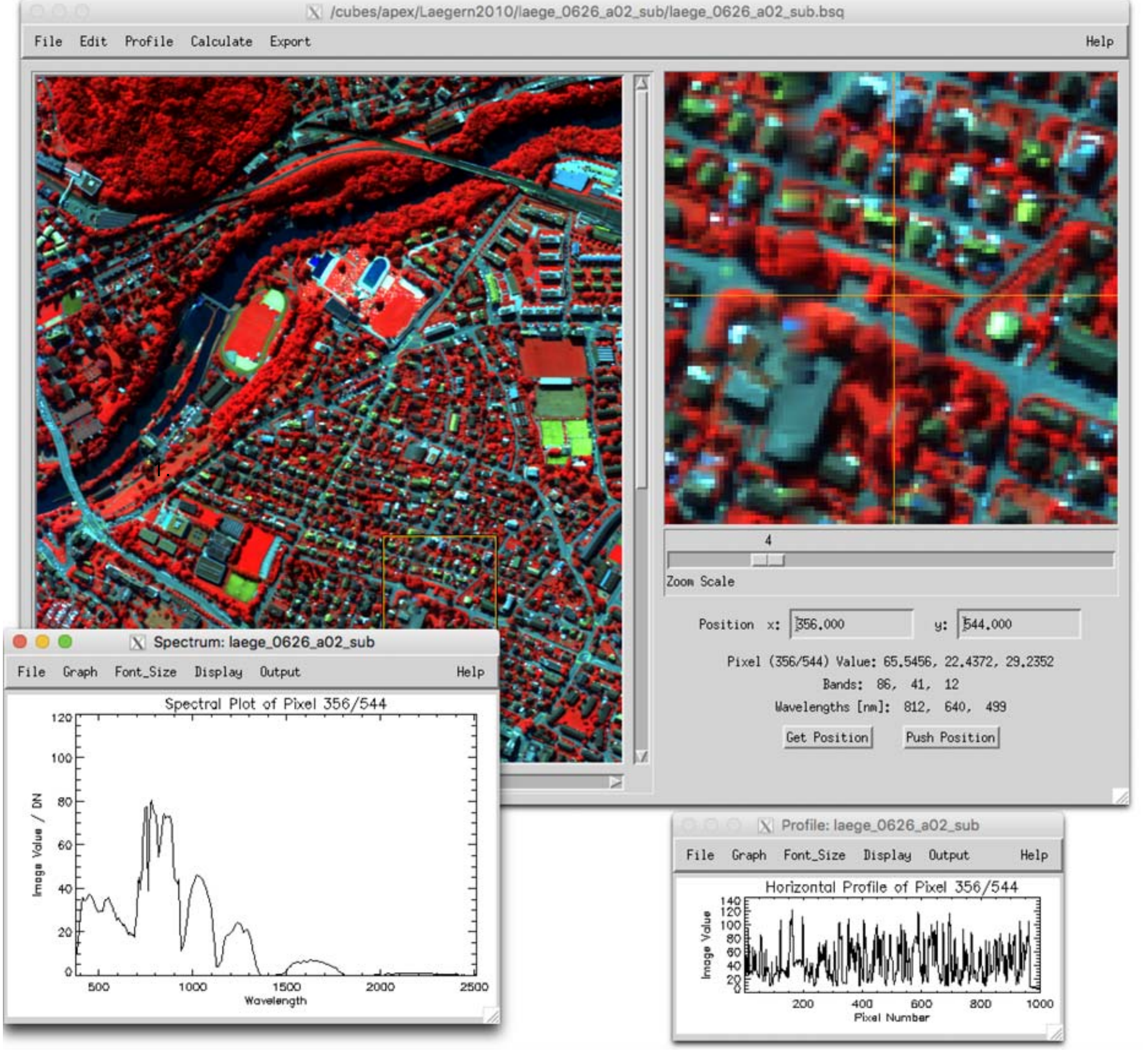


Figure 5.5: Envi file display including spectral and spatial profile plot..

One or three channels of an ENVI file can be displayed as seen in Figure 5.5. When starting up, a dialog box allows the selection of the bands to be displayed. This selection may be changed at a later stage using the menu command **File:Band Selection** (s. Figure 5.6). If the file does not contain more than 3 bands, the display will automatically switch to single band mode, whereas true color mode is started when more than one band is selected.

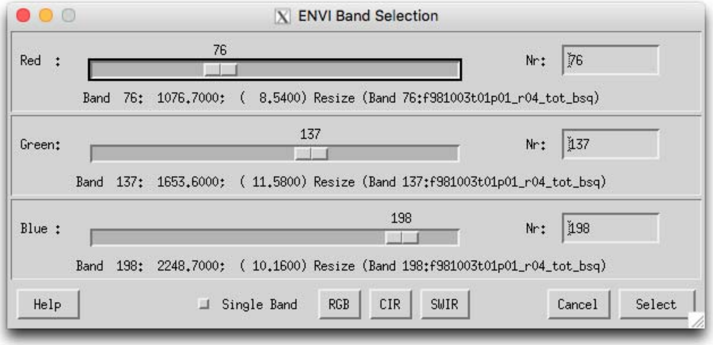


Figure 5.6: ENVI display band selection dialog.

Actions:

- Positioning the cursor in the zoom window will update the coordinate and DN values information in the text below the window.
- The position, either in pixel-centered image coordinates or in absolute geographic coordinates of the cursor is displayed in editable fields. And the pixel position and the value in the original image is shown below.
- If the coordinates are entered manually, the cursor will jump to the selected coordinates (only after the 'return'-key is hit).
- Clicking in the file overview updates the zoomed image portion; roaming is enabled if the mouse button is pressed.
- The motion of the zoom as well the pixel location can be fine-tuned using the arrow-keys from the keyboard.
- Change the zoom magnification by using the zoom-factor-slider (a magnification of 0 will downsample the data by a factor of 2).
- Clicking into the zoom windows displays the coordinates and image values below. If a spectrum is available, the spectrum is plotted to a separate window as soon as the middle mouse button is used for clicking. This function can also be started using the menu Profile:Spectrum.
- Middle-click or right-click in the Zoom window will show the spectral profile.
- The button 'Push' is used to copy the currently diplayed location into memory.
- The button 'Get' is used to copy the location from memory to the current display. This allows a link between two displays.

File Menu:

- **Show ENVI Header:** Displays the ENVI header of the current image in a new editable window. This allows to make changes to the ENVI header. Note that the file needs to be loaded from scratch if changes have been made.
- **Band Selection:** Allows to select a new spectral band combination for thecurrently loaded ENVI image (if available). The selected bands are then loaded in display. Use the buttons 'RGB', 'CIR', or 'SWIR' to default to true color, false color or SWIR default band configuration. If single bands are selected, only one band is loaded and the menu command changes to **File:Reload**.
- **Display ENVI Image:** Displays an additional ENVI image in a new window (sorry, no linking available).
- **Plot Spectral Library:** Allows to plot the spectra within an ENVI spectral library (*slb/sli) in a separate window.
- **Close:** Closes the file display window.

Edit Menu:

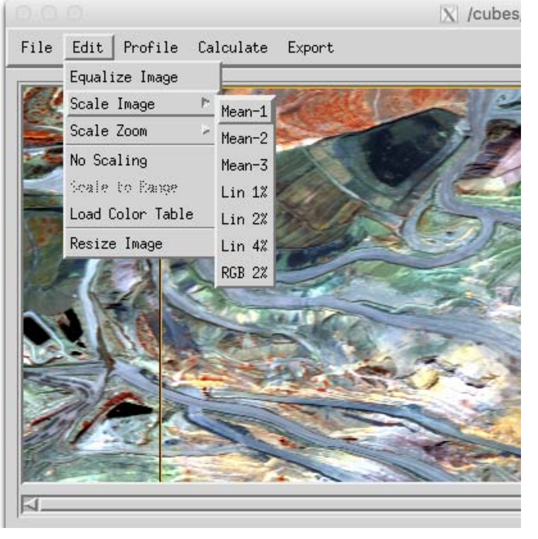


Figure 5.7: Image scaling menu.

- **Equalize Image:** uses the standard IDL `hist_equal` function for enhancement of the currently displayed bands.
- **Scale Image Mean 1/2/3**: allows a linear scaling of the currently displayed image from mean and standard deviation on 3 levels. The scaling applies best to gray-scale images, while awkward results are produced if special color tables have been loaded.
- **Scale Image Lin 1/2/4%:** Applies a linear scaling with a 1/2 or 4% cutoff limits of the histogram.
- **Scale Image RGB 2%**: Tries to scale to an optimal RGB display (this menu point only makes sense with true reflectance imagery).
- **Scale Zoom xxxx**: same functions as with image but using statistics from currenlty displayed zoom window.
- **No Scaling**: sets the display back to the originally loaded image.
- **Scale to Range**: allows to set a scaling range by minimum/maximum image value (only applicable in single band displays).
- **Load Color Table:** change the color table if a single image is loaded - not applicable for an RGB-Display (only avaible in single band mode).
- **Resize Image:** change the spatial dimensions of any ENVI file and store as a new file (see Figure 5.8). The upper left corner is to be given in coordinates or as column/row pixel loca-

tion. The resolution and the total number of pixel drives the size of the output. The resizing is done by bilinear interpolation.

- the Button 'Get UL Position from Cursor' copies the location stored by the 'Push' method int he image display routine
- the button 'Use Reference file' allows to fit the current file to the resolution and extent of a reference file.

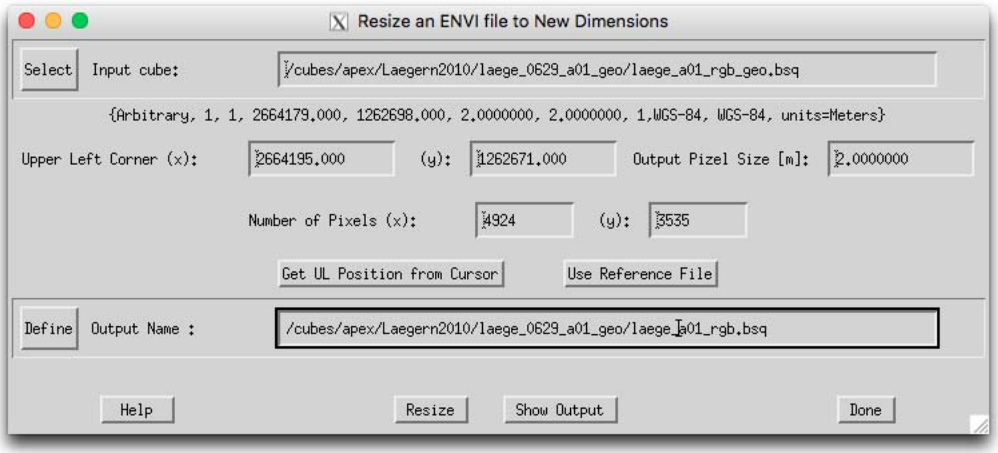


Figure 5.8: Envi file resize dialog.

Profile Menu

- **Horizontal**: opens a window for a horizontal profile through the image (of the first band only). The profile is updated for the cursor location in the zoom window whenever the zoom window is clicked.
- **Vertical:** opens a window for a vertical profile through the image (of the first band only).
- **Spectrum:** opens a window for a spectrum of the image (for images with 4 and more bands only); the spectrum may also be displayed by clicking the middle/right button in the zoom window.

Calculate Menu

This menu is meant for simple image analysis routines.

- **Band Index:** calculate a simple band index such as band ratio, NDVI or absorption depth from spectral imagery (see online help for further details).
- **Spectral Classification**: demonstrates a spectral classification using spectral angle mapping or best fit on the basis of ENVI spectral libaries. The spectral libraries may also be created from image by setting a cursor location probe (see online help for further details).
- **Statistics:** This will calculate the image statistics of all spectral bands of the currently selected image

Export Menu

The currently or last displayed image can be saved as colored GIF, PICT, PNG, TIFF, or JPEG (including its color table where applicable). The image is stored in its original resolution regardless of the currently selected zoom status.

- **Zoom:** exports the portion displayed in the zoom window only (at the resolution of the zoom window)
- **Image:** exports the full image in original resolution.

Note: For export of geocoded imagery it is rather recommended to use the Function **Processor:Process Band or RGB** p.251, where colored 3-band TIFFs or correctly indexed JPEG color files can be generated.

SPECTRA AND PROFILES

The pop-up menus used to display the spectrum or the profile from within the file display routine offer some additional options from its own menus. This includes adjustments to the graphical display and export of the graphics as well as export of the underlying spectra/profile data to a text file (compare Figure 5.5):

File

- Display the data as an ASCII table
- Opens a different ascii table for intercomparison.
- Save the data to an ASCII file
- Print the contents by postscript-printing

Graph

This menu lets you change some display parameters for the current plot, such as title, xy-range, xaxis unit (wavelength or band number).

Font_Size

Used to change the font size to one of the selected values.

Display

- Load color table/palette using the IDL methods
- Set the size of the display to given size in cm.
- Change the type of image display (for color tables, 256 colors are required)

Output:

The currently displayed spectrum may be saved either as EPS file or to any of the displayed rastered outputs. When saving the file to EPS, the same graphics as appearing on the screen is

saved to an EPS file, including font size and dimensions of the displayed image.
When printing, the xsize of the output is fixed to 16 cm, while the ysize is proportional to the size of the plot.

For any image non-eps format, a screenshot of the display is taken and saved as raster image to the chosen format.

DISPLAY TIFF FILE

This function is the same as the one to the display of ENVI files but does not contain the options to select the spectral bands.

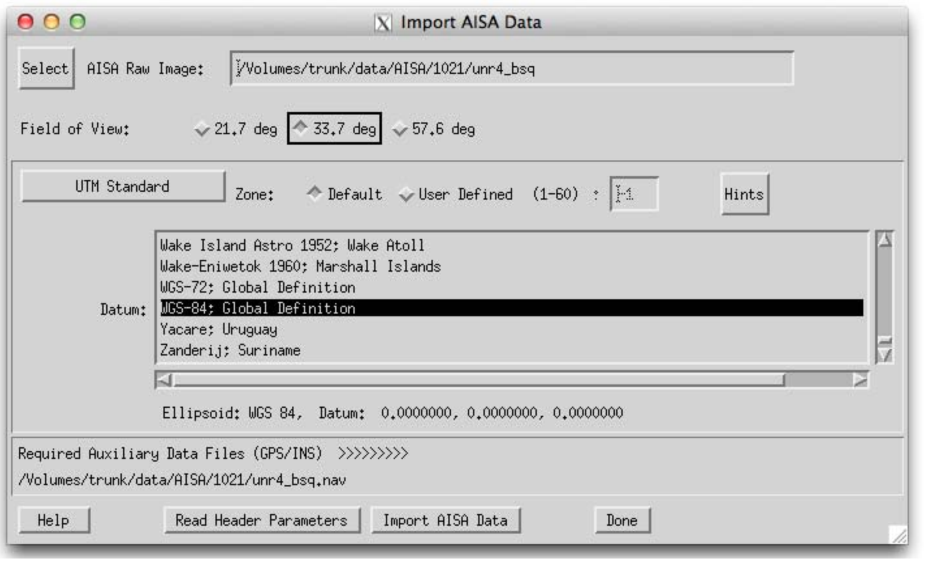
SHOW/EDIT TEXT FILE

This uses the standard text file editor to display an ASCII file. This is convenient to check the data on the current working directory without having to change applications (e.g to look at an ENVI Header or at some ASCII auxiliary data). See detailed description about text editing in Section 5.1.3 on page 122.

IMPORT

This function imports standard formatted and supported data types including all navigation/ attitude data. The import function is currently implemented for AISA, AISA Eagle, APEX, AVIRIS, CASI, DAIS, Headwall Hyperspec, HYMAP, HYPER, HYSPEX, MIVIS, PHILLS, PROBE-1, ROSIS, and various UAV based instruments (under VARIOUS). This listmay be extended for other sensor systems upon request. The options some of the supported sensor systems are described below. Please refer also to Section 3.3 on page 49 for further details about the data import.

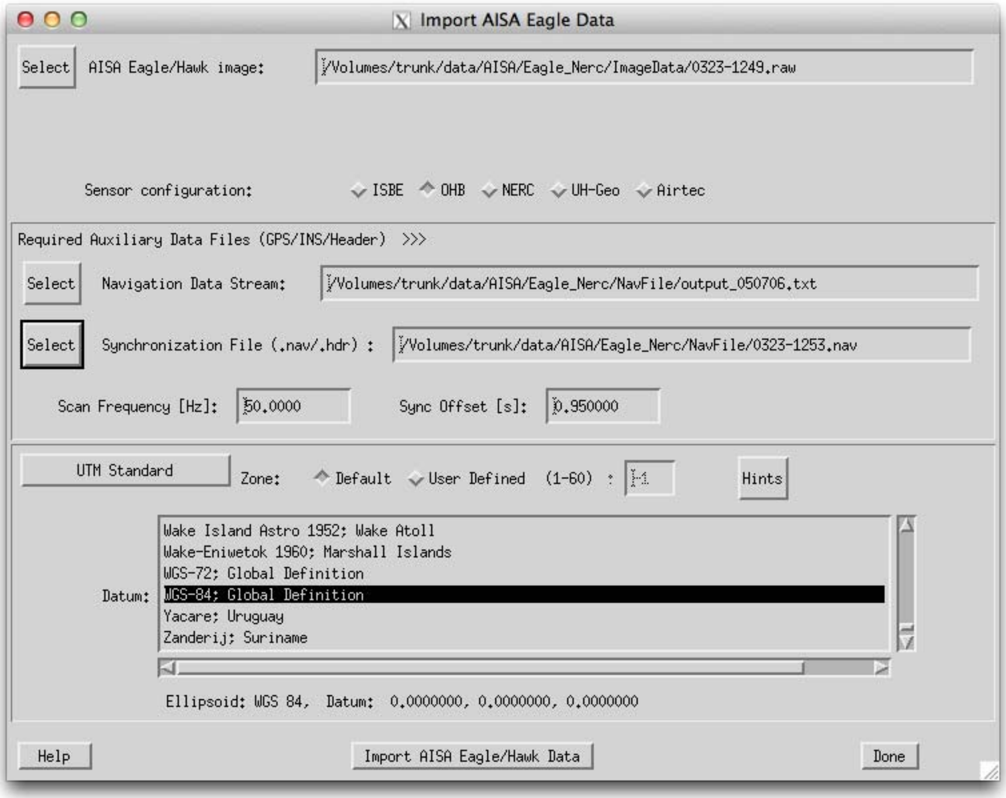
IMPORT AISA



Importing AISA data has once been tested with PARGE. A revision of the interface for support of more actual AISA data is under development. Please contact ReSe Applications if you plan to process actual AISA data and for further information.

IMPORT AISA EAGLE

This function supports the import of the advanced AISA instrument 'AISA Eagle' with 1000 across track pixels.


Import AISA Eagle Data
Select
AISA Eagle/Hawk image:
/Volumes/trunk/data/AISA/Eagle_Nerc/ImageData/0323-1249.raw
Sensor configuration:
ISBE
OHB
NERC
UH-Geo
Airtec
Required Auxiliary Data Files (GPS/INS/Header) >>>
Select
Navigation Data Stream:
/Volumes/trunk/data/AISA/Eagle_Nerc/NavFile/output_050706.txt
Select
Synchronization File (.nav/.hdr) :
/Volumes/trunk/data/AISA/Eagle_Nerc/NavFile/0323-1253.nav
Scan Frequency [Hz]:
50.0000
Sync Offset [s]:
0.950000
UTM Standard
Zone:
Default
User Defined
(1-60) :
-1
Hints
Datum:
Wake Island Astro 1952; Wake Atoll
Wake-Eniwetok 1960; Marshall Islands
WGS-72; Global Definition
WGS-84; Global Definition
Yacare; Uruguay
Zanderij; Suriname
Ellipsoid: WGS 84, Datum: 0.0000000, 0.0000000, 0.0000000
Help
Import AISA Eagle/Hawk Data
Done

IMPORT AVIRIS

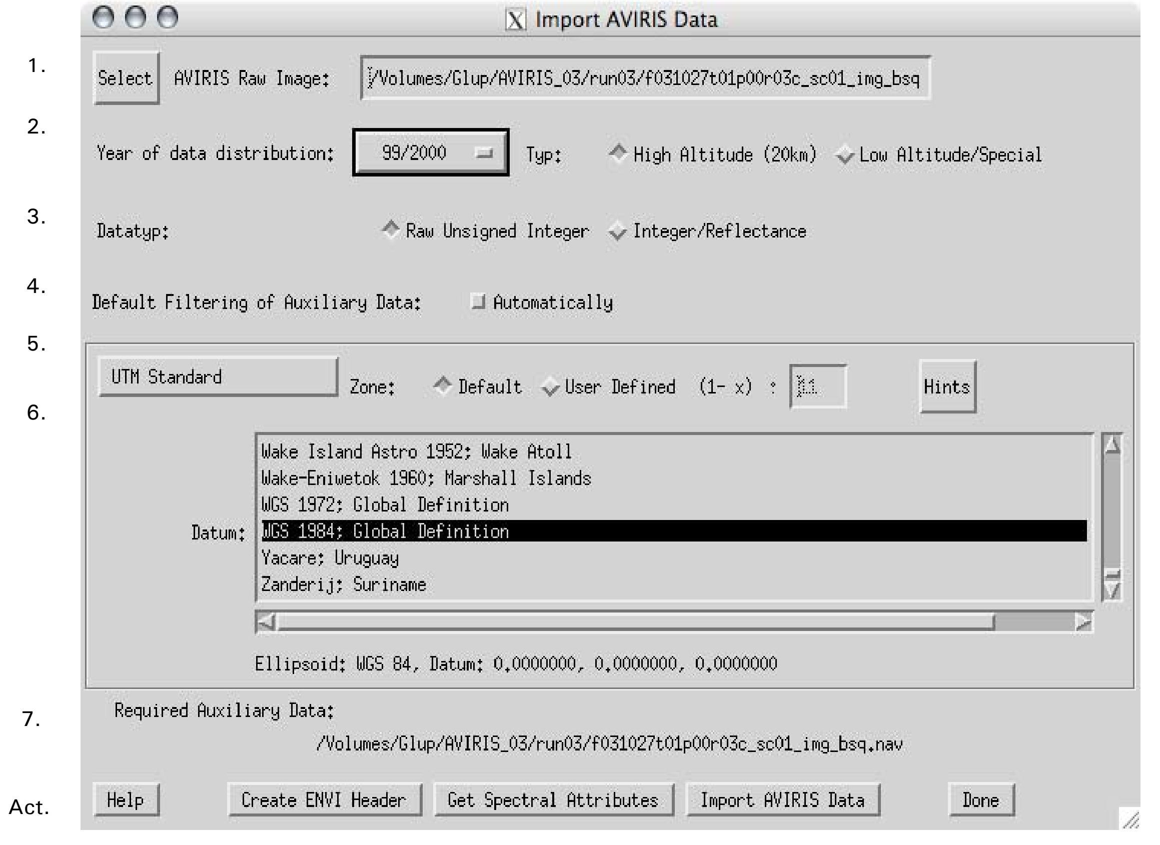


Figure 5.9: Import Raw AVIRIS Data

Inputs:

Nr.	Information
1.	Select AVIRIS Raw Image: Select ENVI formatted AVIRIS file in raw geometry (614 pixels across track). It is highly reommended to use a file in BSQ format for PARGE. The name of this file without extension must be the same as for the corresponding auxiliary data files. Please use the command >Special:Convert BIP to BSQ< if your raw image is in BIP (Band Interleaved by Pixel) format.

Nr.	Information
2.	Type of AVIRIS Data: defines the type of available auxiliary data, depending on the year of data delivery. This is quasi independent of the year the data has been taken! • Before 1994: .eng and .nav data of the ER-2 aircraft have been available but of mediocre accuracy. • 1995 through 1997: The formats of the .eng and .nav data files changed slightly between the years while their contents remained the same. • 1998/99: The first year with low altitude data, wherefrom a .gps file should be available for data import • After 2000: GPS accuracy and synchronization have been highly improved starting in 2000. This data therefore can be read at good accuracy. In 1999, the synchronization between the .gps file in the low altitude data and the image lines requires use of the Special:Post Synchronization function. If 1999 data shall be processed you may use the import filters for 1998.
3.	Low Altitude option: Starting in 1998, low altitude flights were carried out with differing parameters from the high altitude data. Specifically,a C-MIGITS INS system has been included for data acquisition resulting in an additional data stream. Another important difference between the high and the low altitude data was the roll compensation which has been switched off for Low Altitude data.
4.	Raw data type: AVIRIS data usually are delivered as unsigned integer raw binary data in BIP format. Optionally, you may also obtain atmospherically corrected reflectance data which are usually in BSQ integer format. Please use the second option to select this kind of data.
5.	Default smoothing of Auxiliary Data: smoothes all parameters by default factors. This smoothing has been required up to 1998 data. Use this option if you want a direct best-guess smoothing applied to the data(the smoothing may also be done subsequently by the commant *Auxiliary:Filtering*.
56	Coordinates conversion: The GPS information in the AVIRIS files comes in WGS-84 long/lat format. This coordinates need to be converted to a metric system. For north america the nad27 and the nad83 are the most widely used systems. See Section 5.1.6 on page 124 for more details about coordinate conversion.
7.	On these lines, the required auxiliary data files are displayed. Their name should be the same as the raw image despite their extensions (some renaming may be required).

Actions:

Action:	Information:
Create ENVI Header	Defines standard AVIRIS parameters and creates an ENVI *.hdr file corresponding to the raw dat
Import AVIRIS Data	Imports the AVIRIS data including its associated auxiliary data.

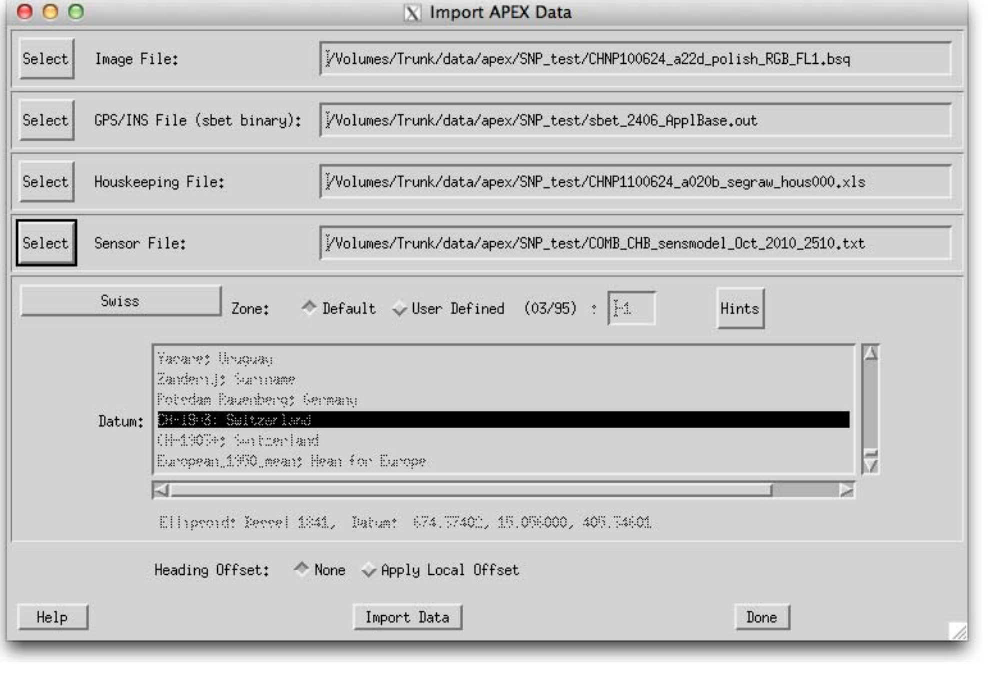
Outputs:

This function updates the input scene definitions. Furthermore, all attitude data is initialized and read where available.

The application tentatively opens the AVIRIS file first and assigns the number of lines if no ENVI header is available. An ENVI header is written if none has been available. Second, the navigation data is read (assumed to be WGS-84) and may be transformed (e.g.) to a generic UTM coordinate system, subtracting the NAD 27 datum (Clark 1866 ellipsoid) or any other applicable coordinate system. The heading is offset during coordinate conversion (compare Section 3.3.1 on page 49).

IMPORT APEX

Imports the standard APEX Level 1 formats.



IMPORT CASI

PARGE principally is well suited for processing of any Itres sensor such as CASI, SASI. Please contact ReSe Applications, if you plan to do so - an interface will then be included to the distribution upon request.

IMPORT DAIS

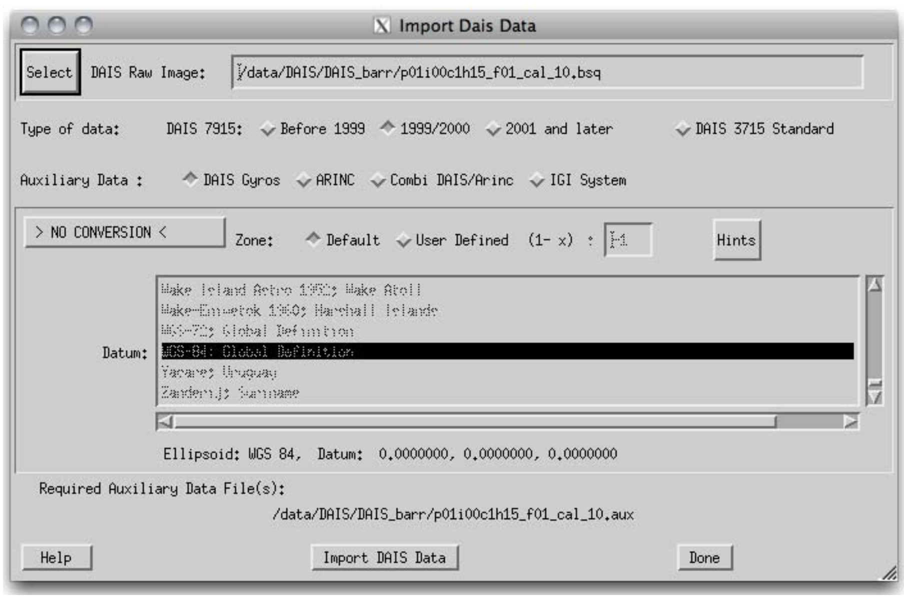


Figure 5.10: Import DAIS Data

Inputs:

Nr.	Information
1.	Select DAIS Raw Image: Select ENVI formatted raw DAIS file. The file has to be in BSQ format for PARGE. The name of this file (without extension) must be the same as for the corresponding auxiliary data files (.aux).

Nr.	Information
2.	Type of DAIS Data: defines the type of available auxiliary data, depending on the year of data delivery. The orientation of the 'aux'- data (i.e. the pitch) has changed between 1998 and 1999, furthermore the GPS time offset increased from 12s to 13s, which both is considered by the year setting.
3.	Attitude Source: Two different gyro reading are stored in the auxiliary data: The DO-228 arinc system and gyros mounted direct on the DAIS. Either of them can be chosen as input source for the angles. If the combination is taken, the low-passed filtered arinc-data is superimposed to the high pass filtered DAIS gyros.
4.	Avoid Smoothing: By default the DAIS data needs to be smoothed from artefacts. This chechbox bypasses that import filter.
5.	Coordinate Conversion: Compare Section 5.1.6 on page 124 for details.

Actions:

Action:	Information:
Import DAIS Data	Imports DAIS imagery and the corresponding attitude data and information about the line synchronisation.

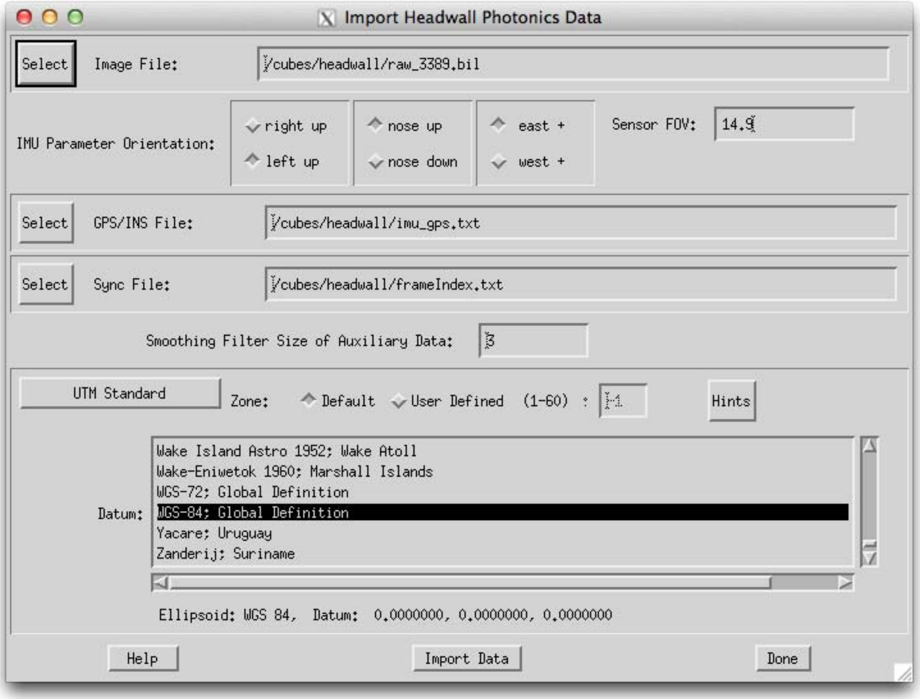
Details:

- No navigation data is available in the 'aux', it therefore has to be read from external DGPS sources using PARGE's import function.
- The synchronization time is taken directly from the 'aux' data, while the frequency is calculated from last and first line time. For synchronization of the GPS data use therefore 'Line time' and not starttime/frequency.

All attitude data is smoothed by default, since the DAIS gyro readings are somewhat noisy in the their original state.

IMPORT HEADWALL

Import the Headwall Photonics Hyperspec Data.



IMPORT HYMAP

Inputs

Nr.	Information
1.	Select HYMAP Raw Image: Select ENVI formatted raw HYMAP file. The file has to be in BSQ format for PARGE. The name of this file (without extension) must be the same as for the corresponding auxiliary data files. Please use the command 'Special:Convert BIL to BSQ' if your raw image is in BIL format.

Nr.	Information
2.	Type of HYMAP Data: defines the type of available auxiliary data, depending on the year of data delivery. • Before 1998: Only the log files were available, no attitude data has been stored and the DGPS data may have been measured externally. Such data will need some hand-work for optimal reconstruction of the parameters. • 1999 with IGI: DLR Munich mounted their own IGI system for one large campaign. This option is proprietary to this setup of Summer 1999. • after 1999 (*.log): A Boeing C-MIGITS INS has been mounted starting in 1999 for the Hymap flights. This option supports the reading of the original *.log file as provided with the data. • after 2000 (*.out): The *.log file is converted by HYVISTA to a columnar ASCII file (.out), which is delivered since 2000.This is an option instead of reading the *.log directly, useful specifically if the latter fails. • after 2003 (*.gps): A standard synchronized .gps file can be read.
3.	Convert Long/Lat to UTM:The original GPS data of the HYMAP system comes (usually) as geographical coordinated based on the WGS-84 ellipsoid. See Section 5.1.6 on page 124 for details about coordinate conversion.
4.	Default smoothing of Auxiliary Data: smoothes all parameters by default factors. Switch this option off, if you want to see the original data.
5.	INS Auxiliary Stream: for the options '1999 with IGI' and 'after 2000', the name of the INS auxiliary data stream may be explicitely set by this input. The default name is assigned as soon as an option and a raw data file have been selected.
6.	On this lines, the required auxiliary data file (*.log/*.pos) is displayed. Its name should be the same as the raw image despite the extensions (some renaming may be required).

The (optionally) required *.log data file appears below. Maybe you'll have to rename some of the files to import them properly (they only should vary by the extension).

Actions:

Action:	Information:
Read Header Parameters	Defines standard HYMAP parameters and writes a new ENVI header if none is available. This action doesn't read any auxiliary data.
Import HYMAP Data	Defines the ENVI header if necessary and reads all related attitude files. If one of the files fails, the reading process is aborted, but anyhow header parameters are already updated.

Outputs:

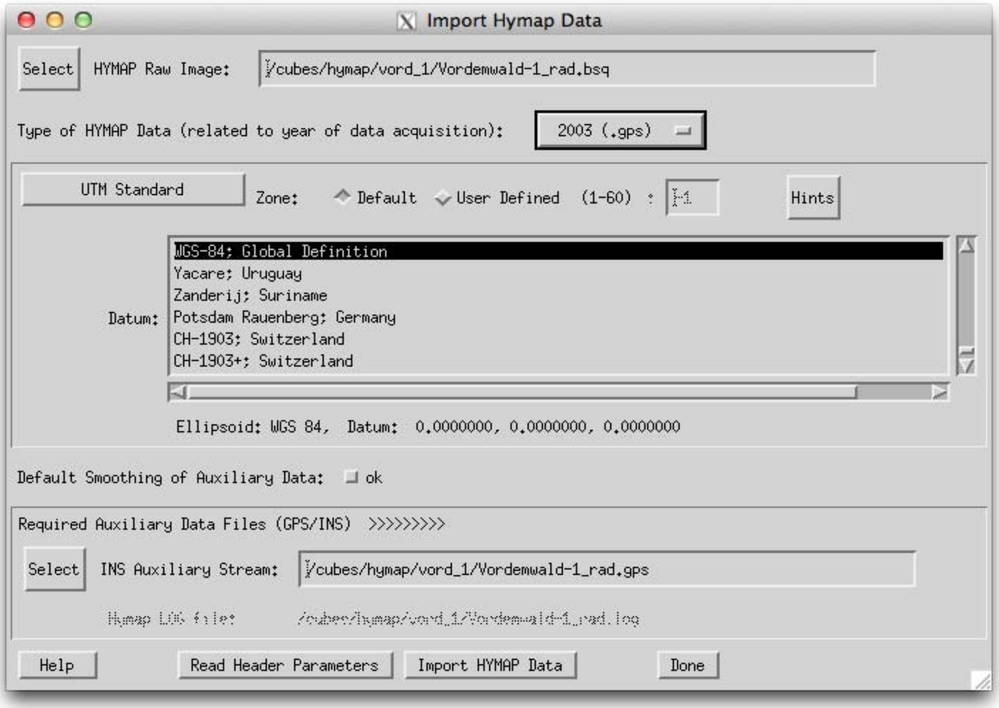


Figure 5.12: Hymap data import panel.

The input scene definitions are updated and all attitude data is initialized and read where available. If coordinate conversion is selected, the selected UTM zone appears at the bottom of the window after successful import.

(compare Section 3.3.1 on page 49).

IMPORT HYSPEX

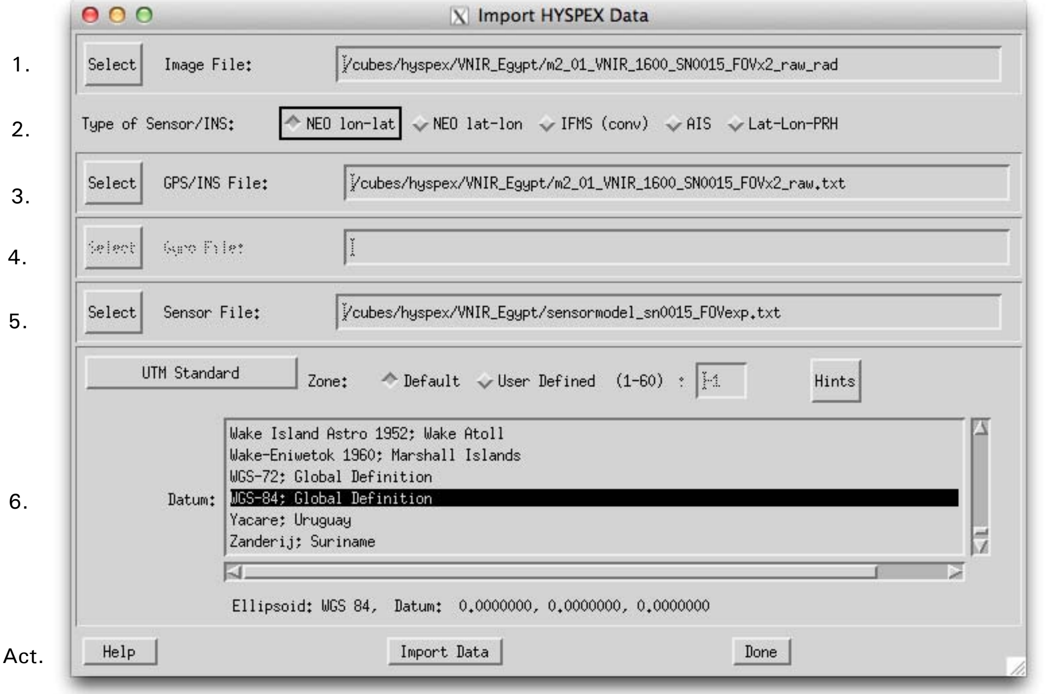
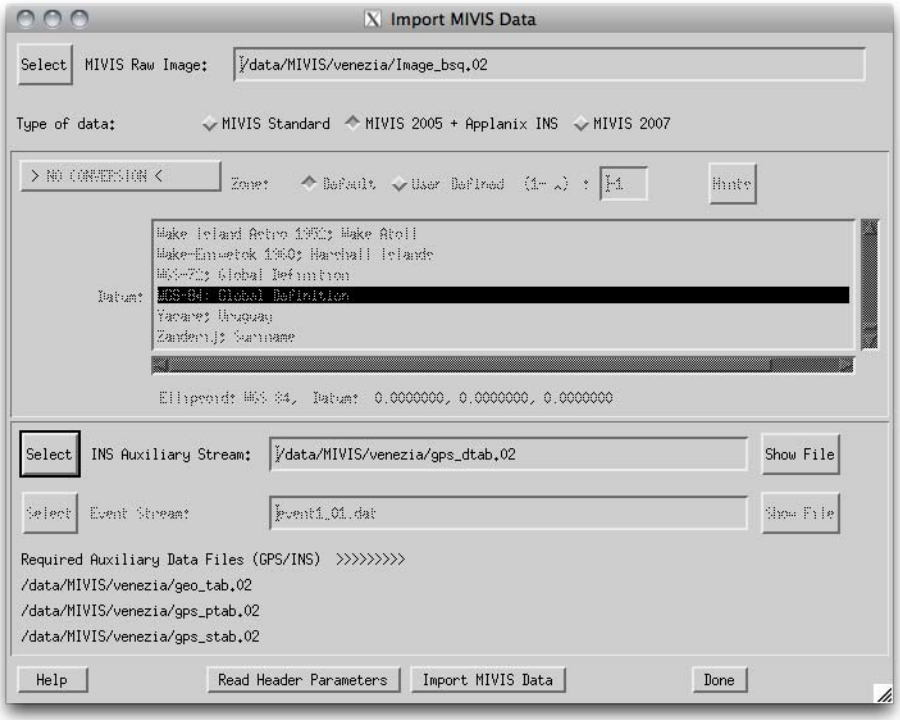


Figure 5.13: Import HYSPEX Data

Inputs

1.	Hyspex data cube.
2.	Sensor type and storage order lon/lat or lat/lon.
3	Name of GPS ASCII file
4	Optional attitude file
5	Sensor model file (explicite model required)
6	Standard coordinate conversion

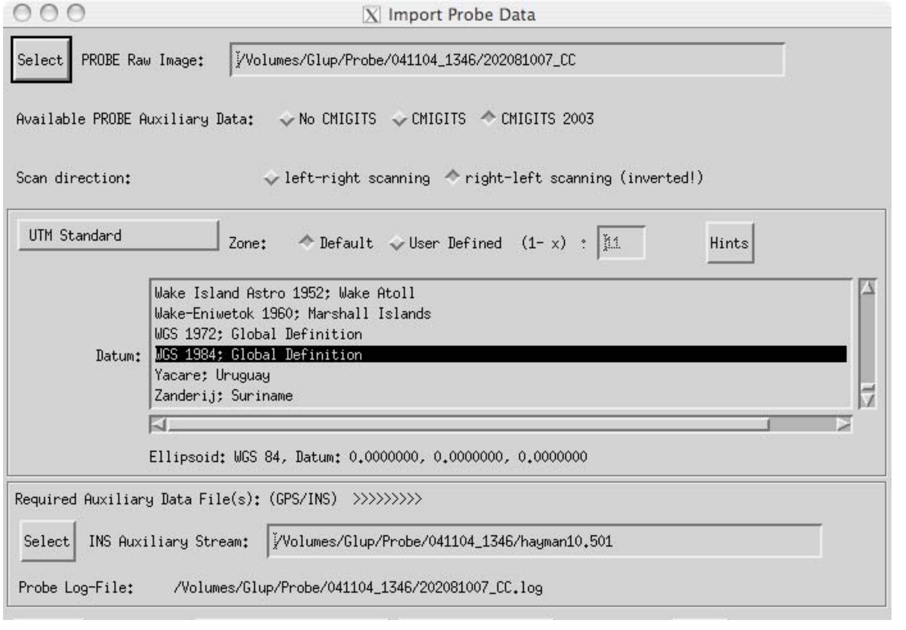
IMPORT MIVIS



Inputs:

Nr.	Information
1.	Select MIVIS Raw Image: Select file. The file has to be in BSQ format for PARGE. The directory this file must be the same as for the corresponding auxiliary data files.
2.	Coordinate Conversion: Compare Section 5.1.6 on page 124 for details.

IMPORT PROBE-1



Importing Probe data follows similar rules as described in the import Hymap data section. Some additional remarks:

- The scanning direction may change from flight to flight which results in mirrored imagery. Change this option, if your results are mirrored in flight direction
- The INS auxiliary data stream may be the same file for a number of flights.

IMPORT ROSIS

1.
2.
3.
Act.

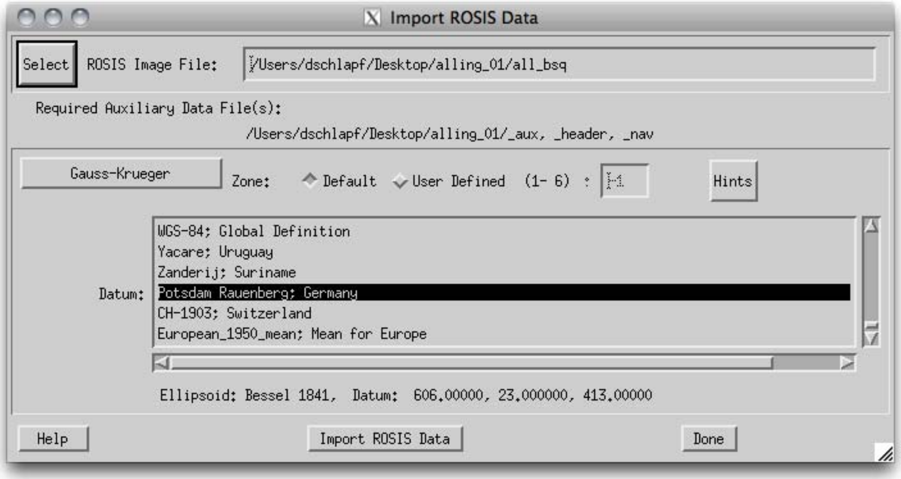


Figure 5.14: Import ROSIS Data

Inputs:

Nr.	Information
1.	Select ROSIS Raw Image: Select raw or ENVI formatted ROSIS file. The file has to be in BSQ format for PARGE. The name of this file (without extension) must be the same as for the corresponding auxiliary data files.
2.	Source of attitude data. The files for ROSIS usually are stored in one directory per data take with the default names as given in the list.
3.	Coordinate Conversion: Compare Section 5.1.6 on page 124 for details.

Actions:

Action:	Information:
Import ROSIS Data	Imports ROSIS imagery and the corresponding attitude data and calculates the pushbroom sensor model.

Details:

- ROSIS data are strictly formatted such that they reside in one directory per data stream.

IMPORT VARIOUS

This window is used for importing various systems store in their specific data format, mainly for UAV based applications. This includes the ARS-3 UAV system, the Resonon PIKA system, the ATSB and the WUR systems using headwall sensors. Additionally, a 'Var' option allows to import a generic simple system with known FOV and synchronized ASCII auxiliary data. The auxiliary data should be columnar ASCII with [line, x, y, z, roll, pitch, heading], no header row.

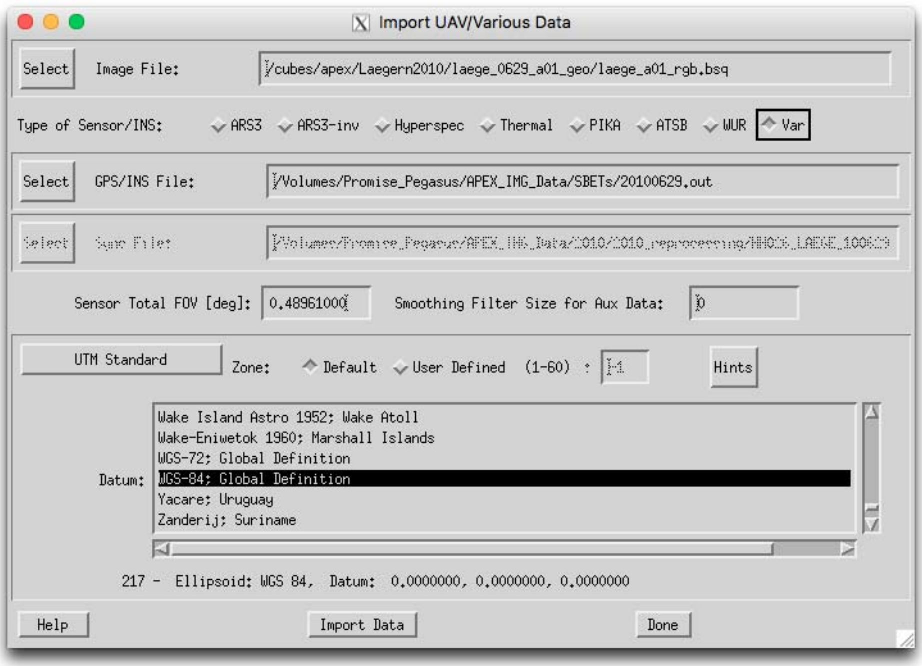


Figure 5.15: Import UAV / Various simple sensor systems.

RESTORE STATUS

This procedure restores all PARGE variables as described in Section 4.1 on page 69 from a *.gcs file. The DEM and one single channel of the imagery are read from the specified files into memory in addition to the structures. If these two files can not be found during restoring, the user is asked to provide locate them in his file system. More about the geocoding status can be found in Section 3.1 on page 41.

Use this function whenever you are about to proceed with normal processing of image data.

Inputs:
Enter the a save file name, recommended extension: '.gcs'

RESTORE VARIABLES

This function restores a previously saved geocoding status, including all attitude data, from '*.gcs' file without accessing any additional file. Thus, the effective data files and cubes are not required. This function is recommended to be used in the following situations:

- examine an old geocoding status,
- cross-comparison of geocoding status'.
- pure GCP-based analysis,

For processing of the data, the function *File:Restore Status*< needs to be selected.

SAVE STATUS

Saves all current variables of the geocoding session, excluding the image arrays. The save/restore mechanism helps to preserve the work status by dumping the current variables to '*.gcs'-IDL save-file at any time. More about the geocoding status can be found in Section 3.1 on page 41, while the definition of the saved variables is described in Section 4.1 on page 69.

SHOW STATUS

Some information about the current session is printed on screen (compare Section 3.1 on page 41).

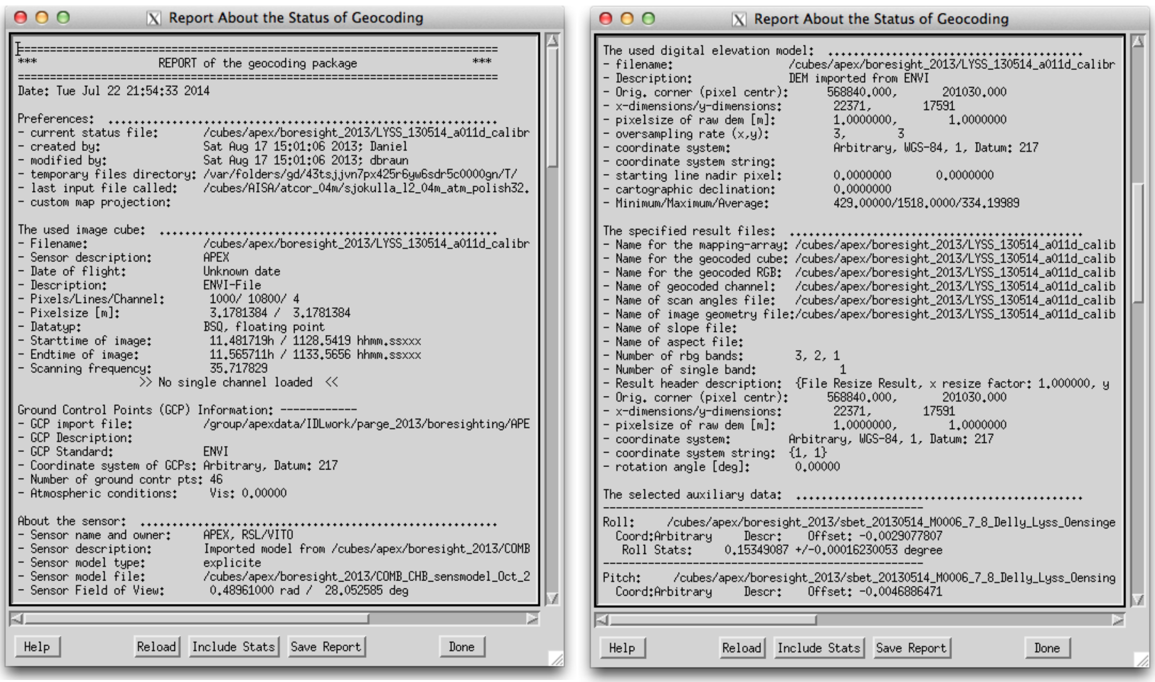


Figure 5.16: Show Status, first two pages(compare to Figure 3.1)

The general informations about the Geocoding Status are described in Section 3.1.1 on page 41.

Actions:

Action:	Information:
Reload	Redisplays the current status information (no automatic update!).
Include Stats	Includes statistical information about the single channel and the DEM (minimum, maximum, average). This operation may take some seconds for large DEMs.
Save Report	Saves the current status information as seen on the screen to a text file.

EXPORT

Allows to export any of the given auxiliary data to either ENVI files or ASCII formatted tables.

Export.Batch Process File

Uses the current parameters to write an ASCII batch processing file, which may be used to start up a batch process from prompt or by file list.

Export.Raw Single Band

Writes the currently loaded single band image to an ENVI formatted single band image, not georeferenced.

Export.DEM

Writes the DEM (variable *demarr*) to an ENVI formatted file including georeferencing header information.

Export.ENVI GLT

export a *_map.bsq file as created with PARGE to a GLT file as used with ENVI.

Export.GCPs

Exports the GCPs (variable `gcparr`) to an ordered list

Export.Navigation Data

Exports the navigation data (variable: `navarr`) to a columnar ASCII file

Export.Raw GPS Data

Exports the GPS navigation data (variable: `gpsarr`) to a columnar ASCII file

Export.Attitude Data

Exports the attitude data (variables: `syncarr, rollarr, pitcharr, headarr, yawarr`) to a columnar ASCII file

Export.Sensor Model

Exports the explicite sensor model (angle per pixel, variable:`sensarr`) to a columnar ASCII file.

RESET SESSION

Resets all PARGE internal parameters to default settings, useful if various runs are to be imported subsequently.

QUIT

Quits PARGE and jumps back to IDL or leaves the virtual machine, respectively. The command *Special:Interrupt* can be used to make the PARGE variables and structures available at the IDL prompt without quitting the program. While quitting, the current operation status may be saved to a PARGE status file (storing a status file is highly recommended if work is to be continued later).

5.3 Menu: Edit

Some generic editing functionalities are collected in the Edit menu:

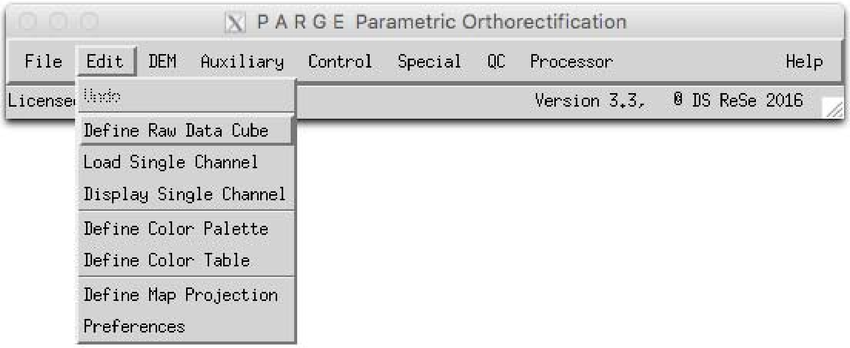


Figure 5.17: The Edit Menu.

UNDO

The undo command allows to undo the last step of the data processing which had effects on the internal PARGE data structure. For that purpose, a temporary status (parge_temp.gcs) is stored after each processing step.

Hint: The temporary status file 'parge_temp.gcs' may also be used to recover a session which was accidentally quit or after a system failure. Just use the command >File:Restore Status< and select the most actual temorary status file.

Attention: UNDO does not recover any accidentally overwritten files, nor does it take track of any display related commands (such as color table changes etc.).

vDEFINE RAW DATA CUBE

This tool defines the data cube dimensions and reads its header data from standards formats (e.g. ENVI,...).

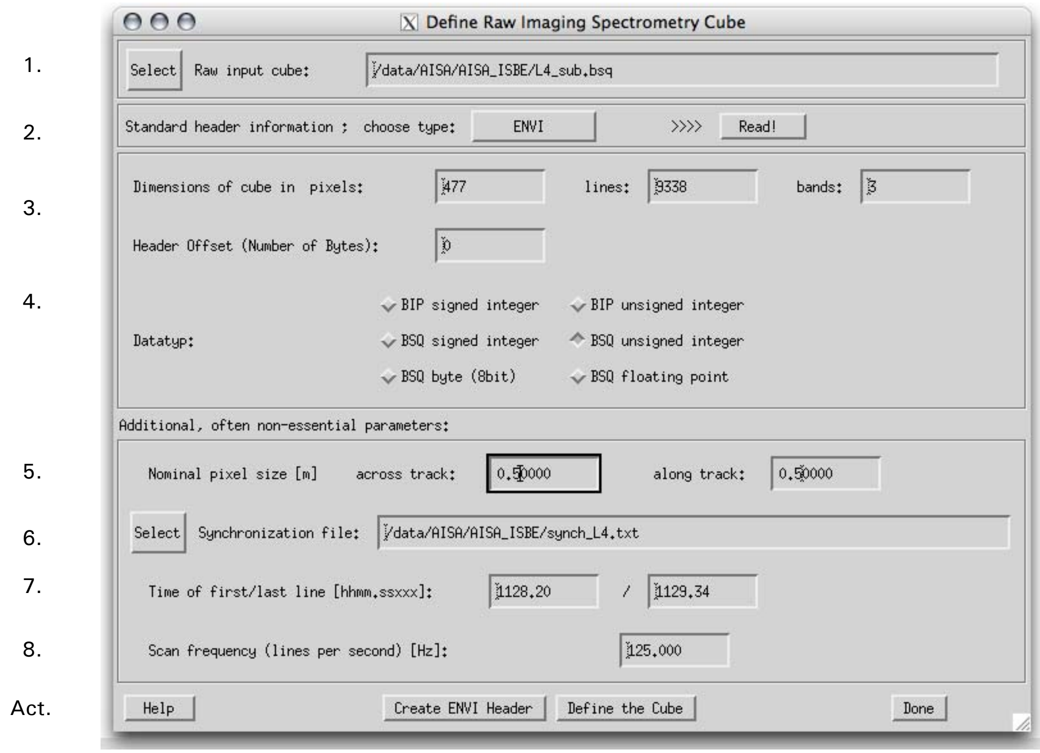


Figure 5.18: Define Raw Imaging Cube

The procedure is to be used if generic raw image data has to be read and its attributes have to be defined. Conversely, the functions in *File: Import* have to be used if auxiliary parameters should be read from standard sensor specific auxiliary data files.

The output of this function will be the stored image definition parameters.

Inputs:

Nr.	Information
1	The raw input cube should preferably be in ENVI format, but may also be raw binary.
2	The standard header information (ENVI, AVIRIS, ...) can be chosen by selecting the file type and activating the >Read< button to obtain all available information for the specified file type. If a sensor type is chosen, standard sensor specific parameters for FOV and IFOV and pixelsize are defined.
3	The dimensions of the cube is given in number of pixels (columns, across track dimensions), number of lines (rows) of whole scene and number of bands.
4	Datatype: preferred data format is Integer BSQ (signed or unsigned). But also 32bit floating point and 8bit Byte data may be used.
5	Nominal pixelsize: the size of the pixels across and along track at nominal flight altitude
6	Synchronization file containing one time entry for each image line (decimal hours or seconds).
7	Synchronization time for first and last image line; format hhmm.ssxxx.
8	The scan frequency (image lines per second) given in Hz.

Actions:

Action:	Information:
Read	Auxiliary information is read from header data and transfered to the image data structure. Additional attitude and navigation data is read too - if available.
Define the Cube	The cube is tested, if its size fulfills the specified parameters and all values in the widget fields are written to the image - data structure. This action has to be taken to accept eventual changes to specific dimension.

Details:

- The pixel size has only informational character. It is not used in the course of processing where the IFOV as given by the sensor model is the driver.
- Start/End time and frequency are only required if auxiliary data have to be synchronized with time and no explicite synchronization parameter per line is available.

LOAD SINGLE CHANNEL

This function reads one single channel from the image cube. The channel shall be used for quick data analysis and preview outputs of the geocoding results. It will remain present in IDL RAM during the work session for fast display.

Actions:

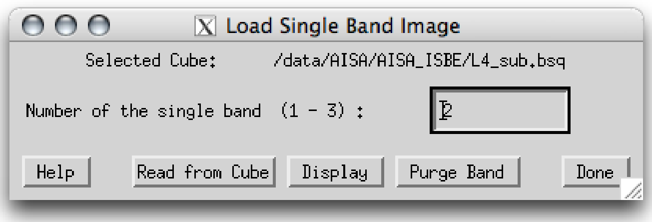


Figure 5.19: Load single channel to memory

Inputs:

The channel number to be read has to be defined starting with 1. The cube already must be successfully defined; its name appears at the top line of the window contents.

Actions:

Action:	Information:
Read from Cube	Reads the whole channel from cube to memory.
Display	Displays the currently active single channel image (same as File: *Display Single Channel* p.155.
Purge Band	Removes the currently loaded band from memory (to free memory)

Outputs:

The function loads a single channel to the geocoding-variable *onearr* in memory.

DISPLAY SINGLE CHANNEL

The selected single channel is displayed as grayscale image which can be scaled using the Edit menu. This function is a standard method for displaying images. It is also used to display geo-coded results and DEMs. Image related commands will apply to the displayed image. Please refer to the description for **File:Display ENVI File** p.126 for details.

DEFINE COLOR PALETTE

This IDL standard-function (xpalette) defines single colors out of 256 for specific plots. It interactively creates color tables using the RGB, CMY, HSV, and HLS color systems using the mouse, three sliders, and a cell for each color index. Single colors can be defined or multiple color indices between two endpoints can be interpolated.

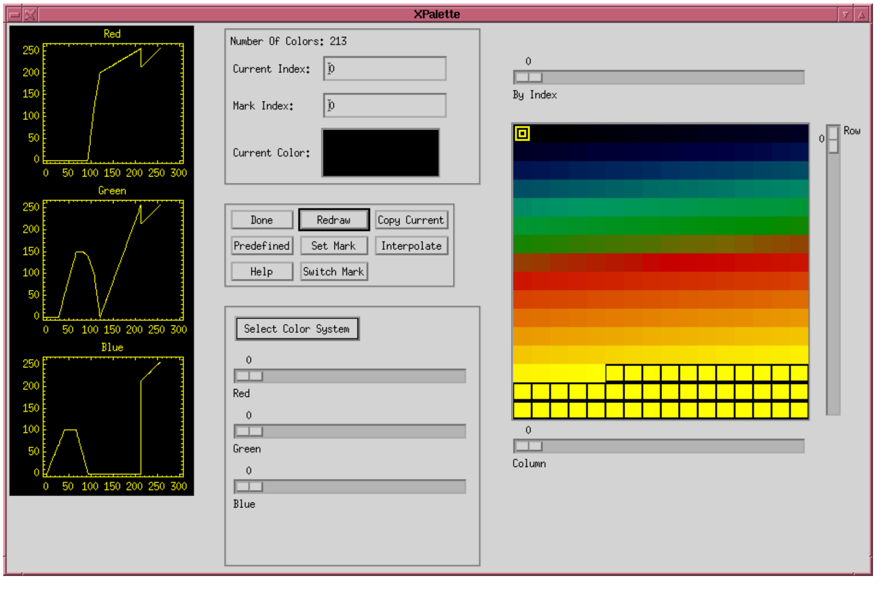


Figure 5.20: Color Palette

Use this function if you want explicitely change the display characteristics of one of the 256 colors. the last color is usually used for plotting, while the first defines the background of the plots.

for completeness, the IDL functionality is repeated here (IDL, ExelisVis Inc.):

Controls:

Row:	**Information:**
Left	Three plots showing the current Red, Green, and Blue vectors.
Middle	1) There is a status region containing the total number of colors, the current index, the current mark index and the current color. 2) The panel of control buttons do the following: - Done: Exits xpalette. - Predefined: Starts xloadct to allow selection of one of the predefined color tables. - Help: This text is displayed. - Redraw: Completely redraws the display using the current state of the color map. - Set Mark: Set the value of the mark index to the current index. - Switch Mark: Exchanges the mark and the current index. - Interpolate: The colors lying between the current index and the mark index are interpolated linearly to lie between the colors of two endpoints. 3) Three sliders (R, G, and B) that allow the user to modify the current color.
Right	A display which shows the current color map as a series of squares. There are 4 ways to change the current color: 1) Press any mouse button while the mouse pointer is over the color map display. 2) Use the "By Index" slider to move to the desired color index. 3) Use the "Row" Slider to move the marker vertically. 4) Use the "Column" Slider to move the marker horizontally.

DEFINE COLOR TABLE

The function *xloadct* is a widget based IDL standard utility for the interactive manipulation of color tables. It has been included in PARGE for display purposes of single band imagery. The color tables do NOT apply to RBG displays!

The color table maps the data values written to the screen to different colors and intensities. Its operation is similar to that of a photographic wedge. The slope and position of the wedge are manipulated to best display a particular dataset.

Actions:

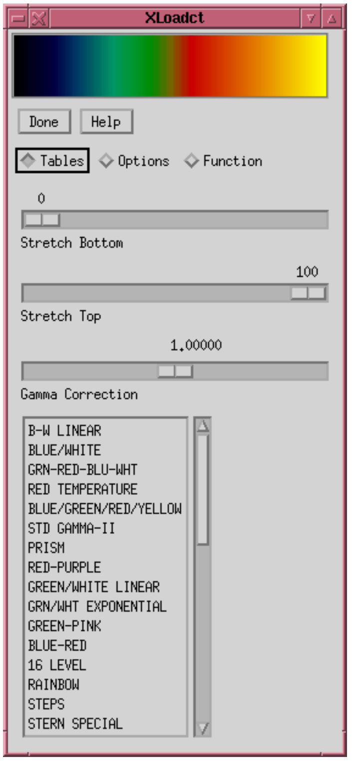


Figure 5.21: Define Color Table

Note: PARGE uses internally a 256 levels for some of its color displays. Please use an image processing software if you require true color capabilities for exploring image data (device,decomposed=0 is set).

Controls:

Action:	**Information:**
Tables	1) Stretch Bottom and Stretch Top Sliders: These sliders control the "contrast" of the color tables and are expressed in percentages of full intensity. 2) Gamma Correction Slider: The slider can be used to compensate for the characteristics of your monitor. 3) Pre-defined Color Table Buttons: Pressing one of these buttons loads the selected pre-defined color table.
Options	1) Gang Sliders: Connects the "Stretch Bottom" and "Stretch Top". Moving one slider moves the other. 2) Top: When set to CLIP, values larger than the Top are set to the largest color index. If set to CHOP, values larger than "Stretch Top" are set to color index 0. 3) Stretch: When set to the default of INDICES, manipulations affect the mapping between color indices and color table triples. When set to INTENSITY, the mapping controls the intensity of each color table entry. In this mode, the hue and saturation remain relatively constant for a given color index.
Functions	Selecting this mode allows interactive editing of the mapping of color table values to color indices by dragging control points on a plot of the map. Select and move a control point by clicking and dragging on its box.

DEFINE MAP PROJECTION

PARGE allows to define custom map projections according to the system as provided through the IDL development environment. A projection may be set per session and is stored as part of the processing status file. It can be accessed by selecting >Custom Projection< in any of the PARGE coordinate conversion dialogs.

See the IDL reference guide which can be found in the help directory of your IDL installation and search for the function “MAP_PROJ_INIT” to read about details of the system (cite: “The MAP_PROJ_INIT function initializes a mapping projection, using either IDL’s own map projections or map projections from the U.S. Geological Survey's General Cartographic Transformation Package (GCTP). GCTP version 2.0 is included with IDL.”, RSI [2004]).

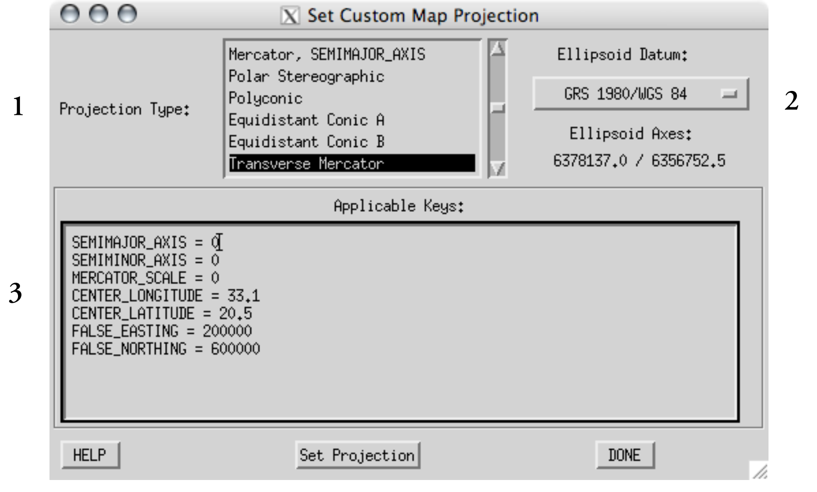


Figure 5.22: Setting custom map projection dialog.

Inputs

Nr.	Information
1	Choose one of the available projections. The allowed keywords will be displayed in the text window below.
2	Select the Ellipsoid applicable to your projection.
3	In this editable window, all parameters may be entered manually to the right of the “=”-sign. By convention, all keyword set to “0” are set to their defaults (exceptions see IDL reference guide).

EDIT PREFERENCES

Two basic preferences of PARGE can currently be customized using this panel

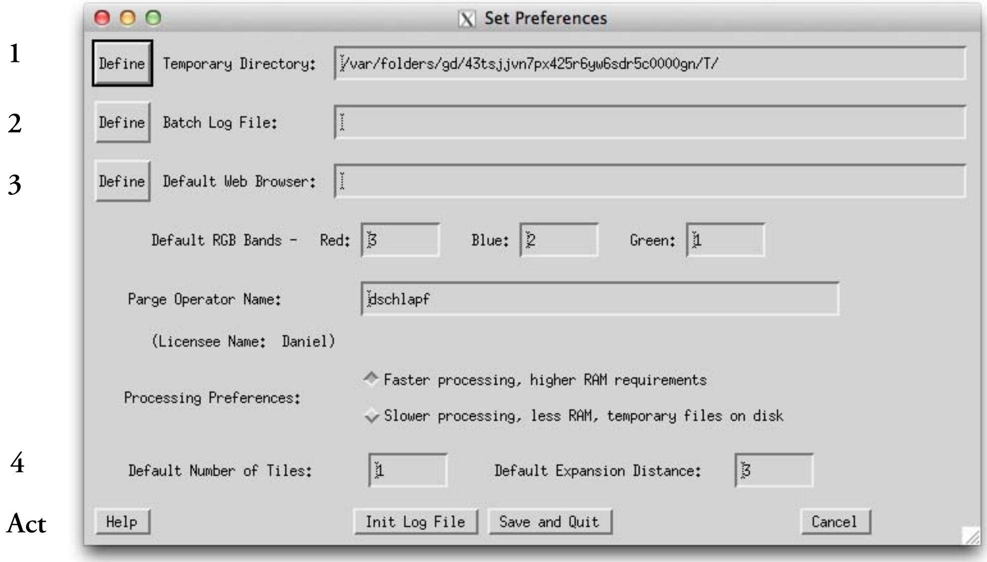


Figure 5.23: Setting the processing preferences.

Inputs

Nr.	Information
1	Temporary Directory: directory to be used for storing temporary files in memory intensive tasks (e.g, for the IGM processor). This directory also contains the save status in case of a system failure.
2	Selection of a log file to be used for history tracking of the PARGE session.
3	Location of executable of default web browser (in case it can be found by the operating system for display of items from the help menu)
4	Default RGB Bands used for display and ENVI headers
5	Operator Name
6	Selection for disk based processing (this preference applies mainly to tiled processing in the IGM processor). More temporary files of intermediate results are written to the temp-directory as defined above.
7	Default number of tiles and interpolation distance for rectification step

Actions

Action:	Information:
Init Log File	Initializes a (new) log file and purges all contents if the log file already exists. This starts logging to this file and writes all actions taken in parge to the log file including time and name of the routine.
Save and Quit	Applies the changes and initializes the log file if it does not yet exist.

5.4 Menu: DEM

The DEM menu is used for import, resizing and handling of the digital elevation/surface model to be used in PARGE. PARGE always needs a valid DEM, even if no real one is available. The DEM is referenced to its origin corner center pixel coordinate. The projection, reference ellipsoid (and geodetic datum) of the DEM defines the geodetic reference of the geocoded result.

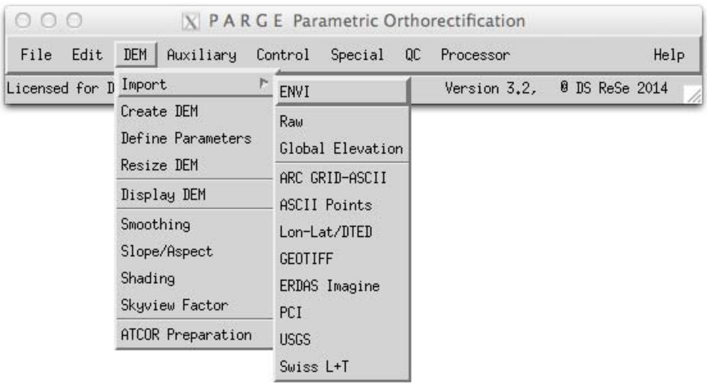


Figure 5.24: The DEM menu

PARGE keeps the ENVI formatted DEM file in the background for processing, while the elevation data themselves are always kept in memory during processing.

Attention about pixel referencing: In PARGE, the DEM pixels are referenced at the centers of the pixels, i.e. pixel 0/0 is the center of the origin pixel. This may cause problems while reading external data of not exactly known format.

IMPORT

Standard DEMs can be read in ENVI, PCI, GEOTIFF, USGS, or Swiss L+T Format and converted to ENVI format.

IMPORT ENVI

An ENVI formatted single band DEM is imported. The georeferencing information has to be provided in the ENVI header file.

IMPORT RAW

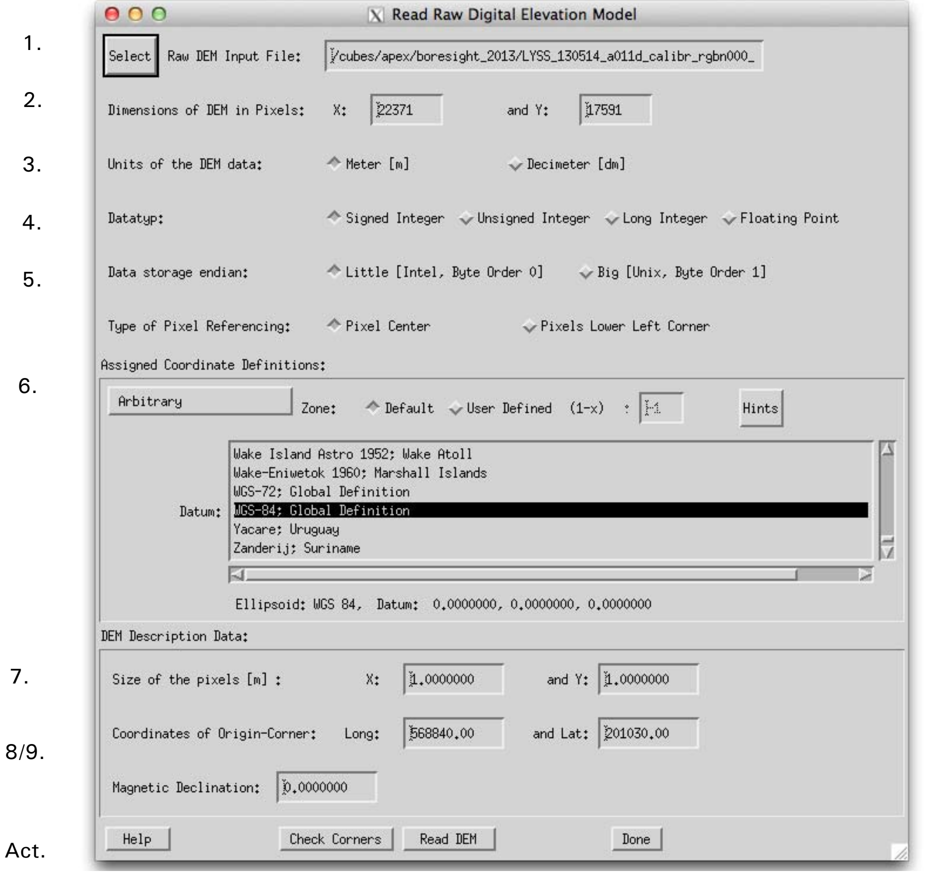


Figure 5.25: Import Raw DEM

This function allows to import a raw binary file and to store it to an ENVI file.

Inputs:

Nr.	Information:
1	Filename of the raw DEM to be read
2	Dimensions of DEM in x and y direction (number of pixels)
3	Altitude representation: meters or decimeters - decimeters will be converted to meters before saving the ENVI file.

Nr.	Information:
4	Data type of raw file: 16bit Integer or 32bit long integer/floating point Note that the file should be in the same endian order as the operating system you are working on.
5	pixel referencing (if pixel edge is selected, the coordinates are shifted by 0.5 pixels to the pixel center position).
6	Coordinate system definition as described in Section 5.1.6 on page 124.
7	Size of pixels - given as distance between pixel centers, only regular spacing is supported.
8	Origin corner coordinate: lowest coordinates in x and y direction, corresponding to the pixel center.
9	Magnetic Declination: non-mandatory parameter which describes magnetic offset of DEM.

Actions:

Action:	Information:
Check Corners	Calculates and displays all four corner positions based on the above definitions.
Read DEM	Reads the DEM based on the definitions; automatically the DEM is saved to a file in ENVI format (give filename) for further use and recovery of the DEM during status management.

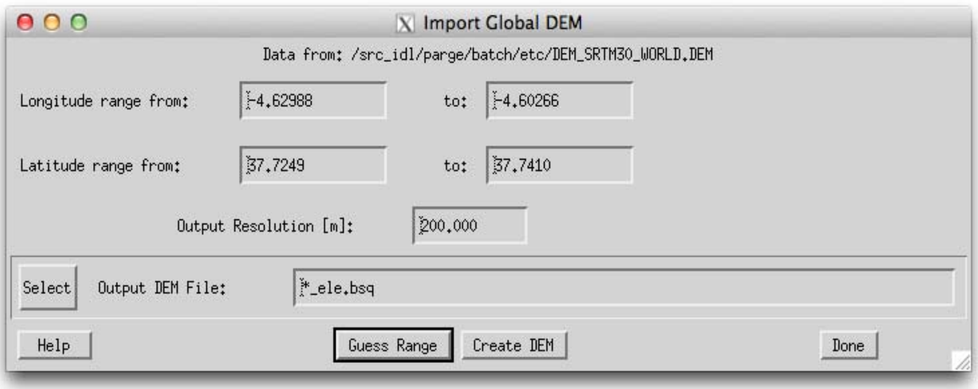
Outputs:

The function defines the DEM data structure and creates an ENVI formatted file from the raw file.

IMPORT GLOBAL ELEVATION

If no DEM information is available for processing, there is a fallback solution to use the low resolution global DEM data available from SRTM campaigns. Such a data set is provided with the PARGE software or is available for download upon request. Its native resolution is approx. 500m (i.e., 0.5 degrees).

The range can be estimated from GPS data and the data is converted to the UTM coordinate system.



IMPORT ARC GRID-ASCII

Reads an ASCII formatted arc GRID DEM, as it can be exported by the arc/info data package. Binary GRID data are not supported, i.e, they need to be converted to ASCII in ARC/INFO first. The button 'View Header' (see Figure 5.26) diplays the first part of the file containing dimensions and refence coordinates. Missing values are replaced by the value entered in the 'missing'field..

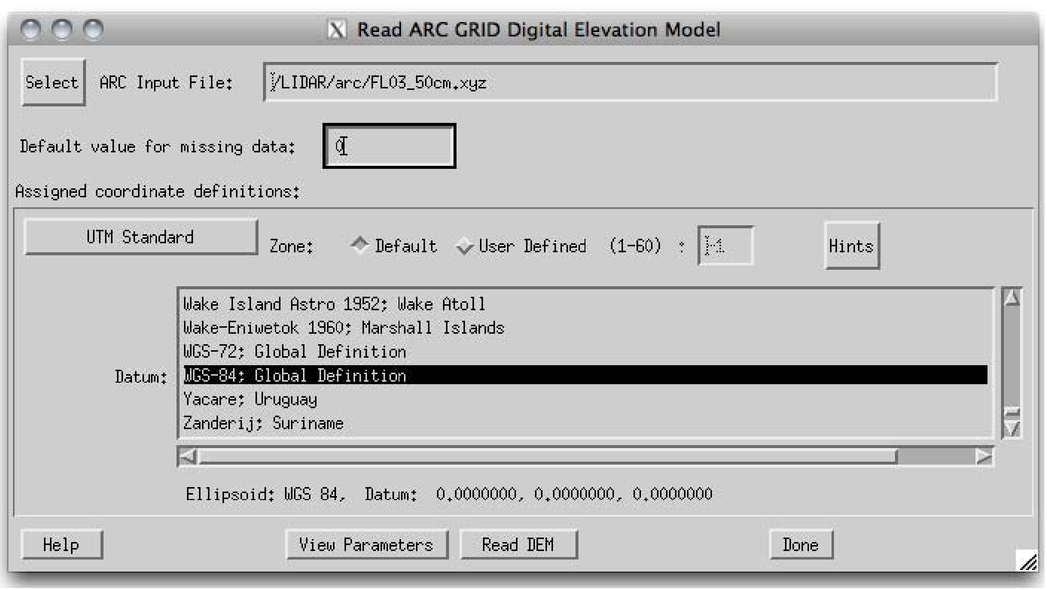


Figure 5.26: Import GRID ASCII data

IMPORT ASCII Points

Reads data points from a file containing pure x/y/z coordinates per line with no header information. The limits are found from the minimum and the maximum coordinates whereas the target resolution has to be entered manually.

IMPORT DTED/Lon-Lat

Imports the DTED format as used (e.g.) for distribution of the SRTM digital elevation models. The long/lat values are converted to the selected target coordinate system. Parameters to be given are the coordinate system, teh target resolution and the type of resampling method

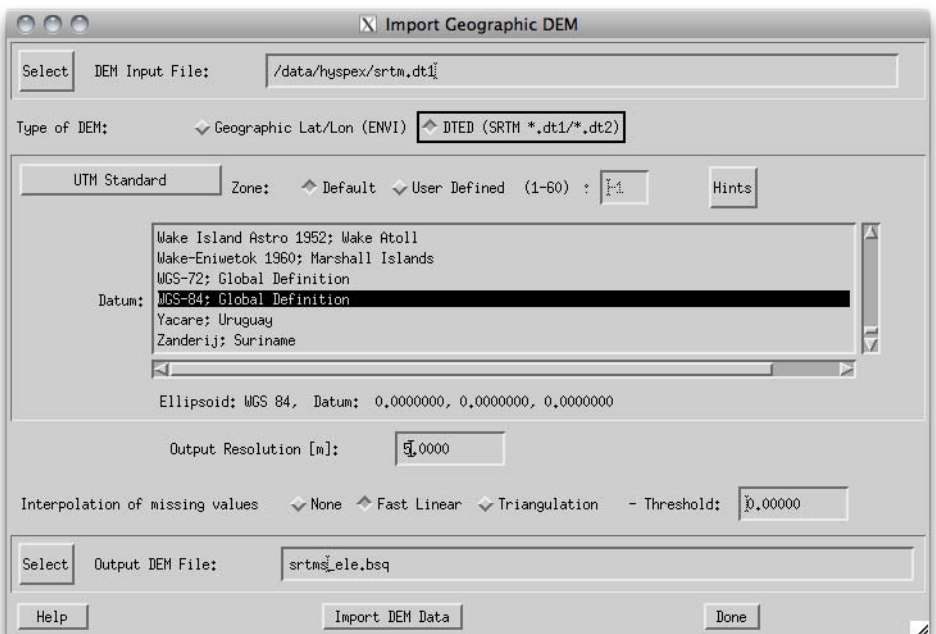


Figure 5.27: Import Lon/Lat or DTED data.

IMPORT GEOTIFF

A standard GEOTIFF formatted single band DEM is imported. The georeferencing information has to be provided in metric coordinates in the GEOTIFF header or the routine will fail.

IMPORT ERDAS Imagine

Reads a DEM from a standard, uncompressed single band ERDAS file.

IMPORT PCI

A DEM can be imported from PCIDSK formatted file (*.pix). The georeferencing information is read from the PCI georef segment, while the data itself has to be read from one of the available PCI channels.

IMPORT USGS

The USGS import function is given with a special interface: As an example for this *Read* Function the USGS Format is chosen.

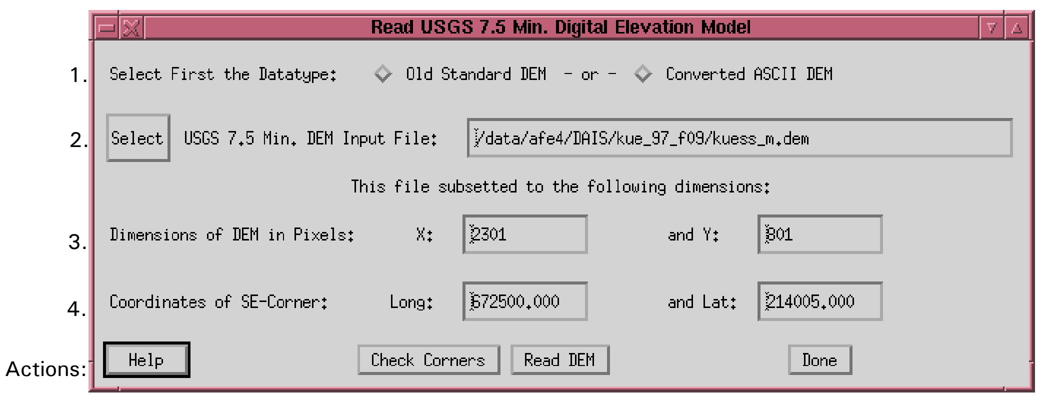


Figure 5.28: Read USGS 7.5 Min. Digital Elevation Model

This procedure reads a standard USGS DEM (7.5 Minutes, 30m resolution) into the PARGE data structure. The DEM is cut directly to the required dimensions on reading.

Inputs (Numbers 1 - 4):

Nr.	Information:
1	Filetyp: the Old Format button should be chosen for the original DEM format and the ASCII format for converted DDF data.
2	Filename: this is the name of DEM to be read.
3	Dimensions: in this tool the size of the desired output in pixels is given.

Nr.	Information:
4	SE-Corner: these are the origin corner coordinates of the DEM. The south-east corner for US DEMs has to be given.

Actions:

Action:	Information:
Select	The file name can be given and is instantly read to obtain the file dimensions (variables remain unchanged!). Use the read-button to store the data after selecting the file.
Check Corners	Calculates and displays all four corner positions based on the definitions of origin and dimensions (pixelsize is 30m).
Read DEM	Reads the DEM based on the given definition and stores it to a single channel ENVI formatted file (if required).

Procedure:

USGS distributes its DEMs nowadays as ISO 8211 formatted SDTS data. This format is not supported by PARGE. Please use the appropriate filters to transform the DDF-Files to '*.dem' files.

Filters can be found at: ftp://ftp.blm.gov/pub/gis/sdts/dem/zip/
The DEMs are available from:
http://edcwww.cr.usgs.gov/doc/edchome/ndcdb/ndcdb.html

The 7.5 min DEMs (30m resolution) are in a slightly tilted representation to UTM coordinates. They therefore can not be read fully without zeroes at some borders. Warning messages will be printed, which can be ignored as long as only the central part of the DEM is used.

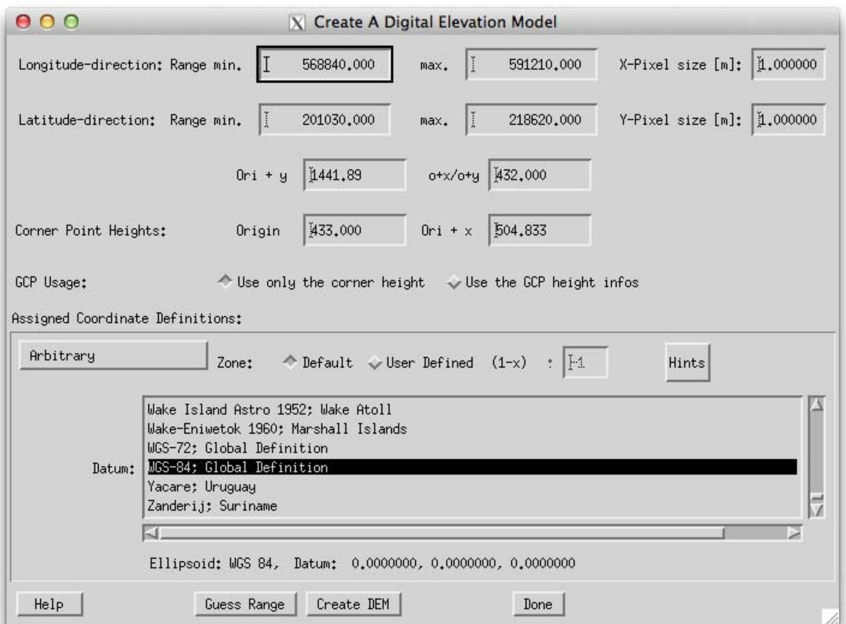
IMPORT Swiss L+T

A standard Swiss L+T DEM file is imported (standard as of 1998). Please ask if new formats are required.

CREATE DEM

The *Create DEM* function creates a flat or interpolated DEM from the corner heights and optional GCP height information. It only should be used if no other DEM is available.

1.
2.
3.
4.
5.



Actions:

Figure 5.29: Create DEM.

Inputs

Nr,	Information:
1 + 2	Dimensions: longitude and latitude range of DEM in meter coordinates (referenced to center of pixel) and pixel size in meters.
3	Corner Point Heights: heights of the four corner points (if the origin corner is entered, the others may be auto-filled by hitting the return-key).
4	GCP Usage: Defines if the GCPs (with flag 1 or 2) are used for DEM creation. The heights of the corner points are used in any case.
5	Coordinate system definition as described in Section 5.1.6 on page 124.

The DEM filename is requested, as soon as >**DEM:Create DEM**< is chosen.

Actions:

Action:	Information:
Guess Range	Guesses the range for the DEM from the given image dimensions and navigation/attitude data. This action is useless if the flightpath hasn't been read so far.
Create DEM	An artifical DEM is created which is based on the given geometric information and saved to an output file in ENVI format.

Output:

The PARGE variables relating to the DEM are updated (*dem.xxx, demarr*) and an ENVI formatted DEM-File is created, which contains the result.

DEFINE PARAMETERS

This function allows to change the DEM description parameters. The projection, reference ellipsoid (and geodetic datum) of the DEM define the geodetic reference of the geocoded result.

1.

2.

3.

4.

5.

Act.

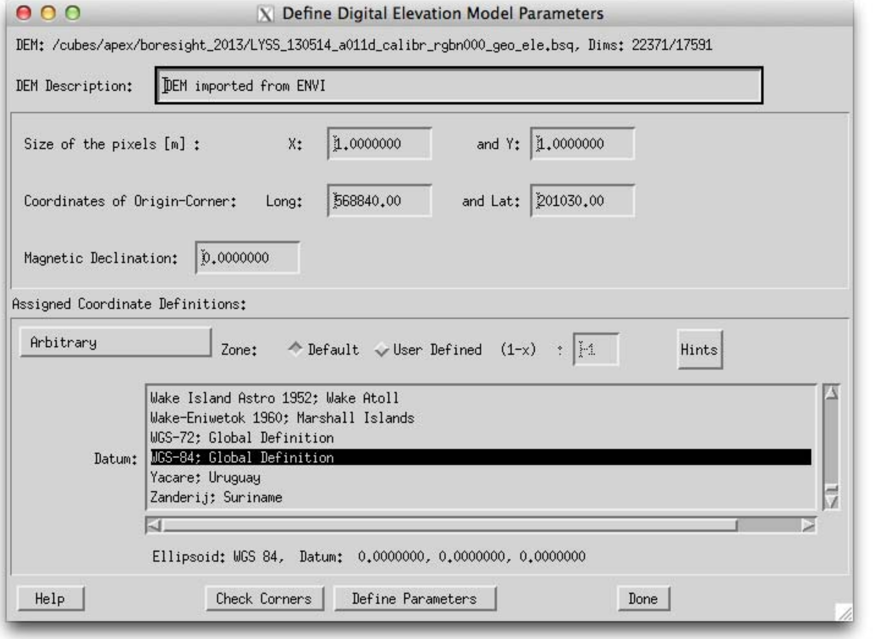


Figure 5.30: Define Raw DEM Parameters.

Inputs:

Nr.	Information:
1	Information about filename and dimensions of the current DEM and editable DEM description information
2	Spacing between pixel centres (x,y) (pixelsize)
3	Origin corner coordinate: lowest coordinates in x and y direction, corresponding to the pixel center.
4	Magnetic declination: in radians
5	Coordinate system definition as described in Section 5.1.6 on page 124.

Actions

Action:	Information:
Check Corners	Calculates and displays all four corner positions based on the above definitions.
Define Parameters	Assigns the definitions to the DEM parameters.

Outputs:

The function will update the DEM data structure.

RESIZE DEM

This function allows a resampling and a cutting of the DEM to the desired dimensions. Note that the size of the DEM affects the RAM requirements when processing, specifically in the MAP workflow.

1.

2.

3.

Actions:

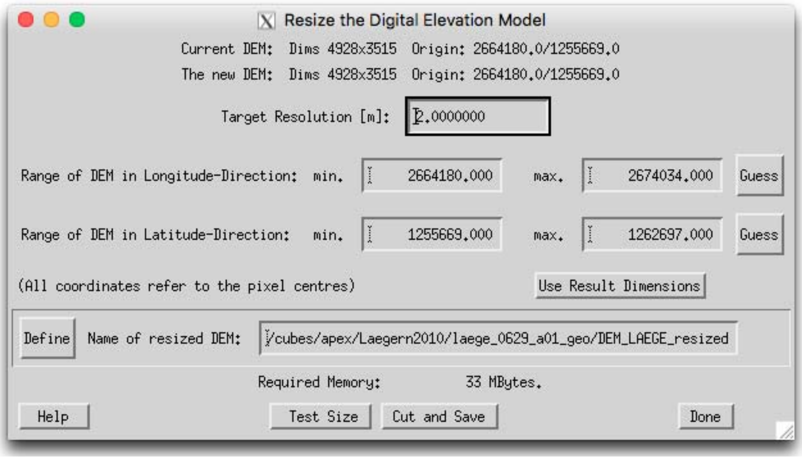


Figure 5.31: Resize DEM

Inputs:

Nr.	Information:
1	Target Resolution in meters (x/y resolution are symmetric)
2	Mimimal and maximal x/y coordinates (these are center pixel coordinates) of the output image.
3	Name of the resizes DEM.

Actions:

Action:	Information:
Use Result Dimensions	The currently defined extent of the result is used to resize the DEM - this helps to create pixel-to-pixel relation between DEM and PARGE output image..
Test Size	The DEM can be cut to test the memory requirements, but the changed variables (number of pixels and required memory for the new DEM are shown) should not be saved.

Action:	Information:
Guess	This button guesses the range of the DEM in x/y direction separately. The range is evaluated based on a preview geocoding to the 0 meter altitude level.
Cut & Save	Cuts again and afterwards saves the DEM.

Outputs:

The DEM (variable *demarr*) is cut to the desired size while setting all parameters in the DEM-structure of the geocoding package. The resized DEM is stored to ENVI format.

Procedure:

The oversampling uses a bilinear procedure (IDL REBIN-function); the dimensions are estimated and eventually rounded from the given boundaries (possible loss of pixels).

DISPLAY DEM

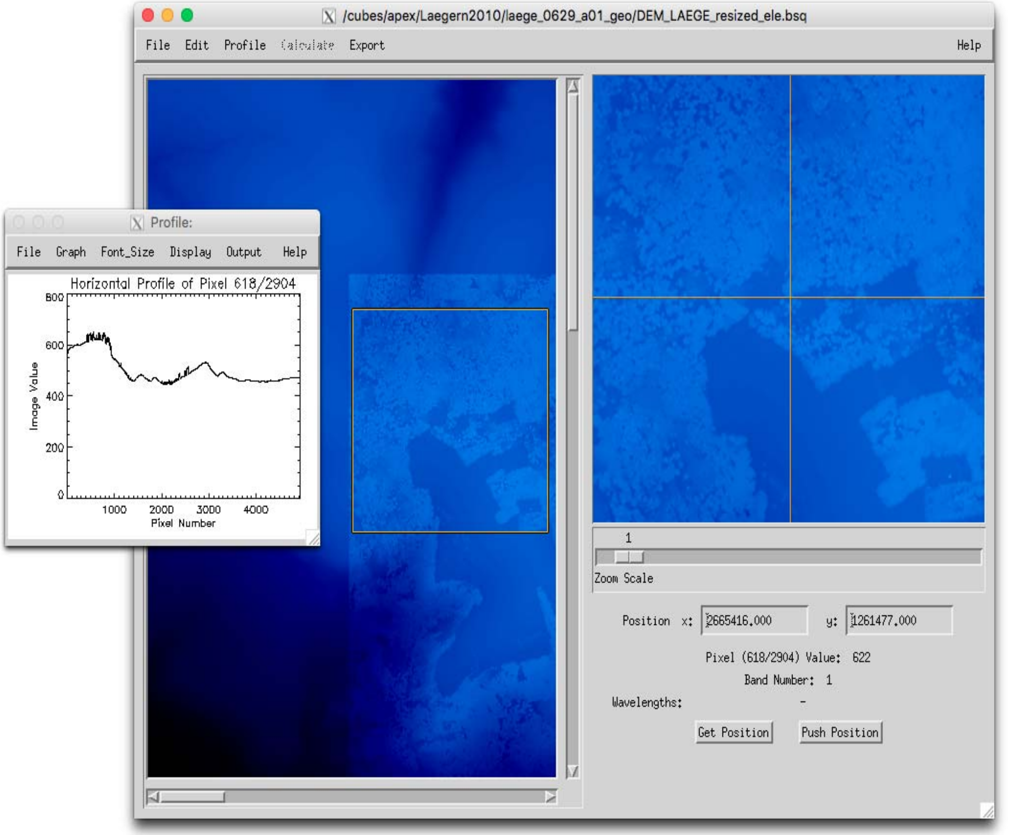


Figure 5.32: Display DEM.

This function displays the DEM as 2D image. The command **Control: Display Flightpath on DEM** p.202 can be used to display the DEM together with the flightpath and GCPs. The position of the cursor may be read for control purposes.

SMOOTHING

This program is used to smooth/filter the currently loaded DEM. A standard low pass smooting filter as well as a median filter is included.

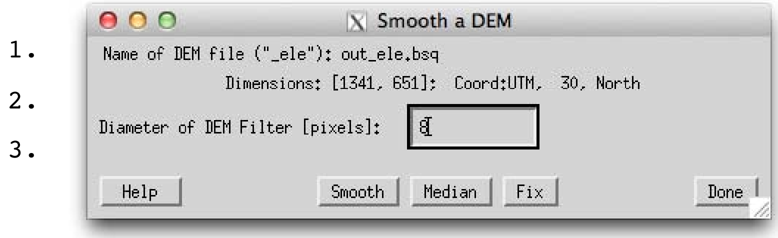


Figure 5.33: DEM smoothing and filtering function.

Inputs:

Nr	Description
1	The currently defined PARGE DEM (see status) is used as input data for calculation.
2	The description of the currently selected DEM is displayed on top - the information is updated by sliding over the name.
3	Diameter of DEM Filter: size of moving window for filtering given as diameter of a quadratic box in pixels.

Actions:

- Smooth: applies a smoothing low-pass filter to the DEM (usefull prior to slope/aspect calculation).
- Median: applies a median filter to the DEM (useful to remove pixel artifacts from DEM)
- Fix: removes artifacts from the image (i.e. single values beyond reasonable limits)

Outputs:

- The currently loaded DEM is filtered and can be stored to a new ENVI file.

Procedure:

Standard IDL smooth/median functions are employed for image filtering.

SLOPE & ASPECT

This program is used to calculate the slope and aspect angles from DEM elevation data for later radiometric processing.

1.
2.
3.

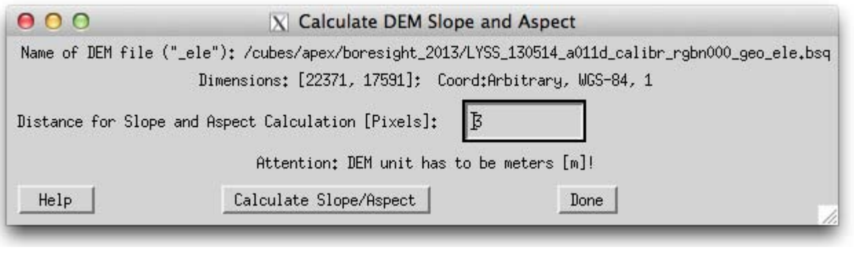


Figure 5.34: DEM Slope and Aspect calcuation

Inputs:

Nr	Description
1	The currently defined PARGE DEM (see status) is used as input data for calculation. The name of the DEM should end with *_ele (except for an optional extension) in order to be compatible to the ATCOR4 standards.
2	The description of the currently selected DEM is displayed on top - the information is updated by sliding over the name.
3	Calculation Distance: In the easiest case, only one pixel distance is required for slope/aspect calculation. Anyhow, this distance bears the problem of artefacts in relatively flat terrain. Higher distances will remove such artefacts while loosing some details.

Outputs:

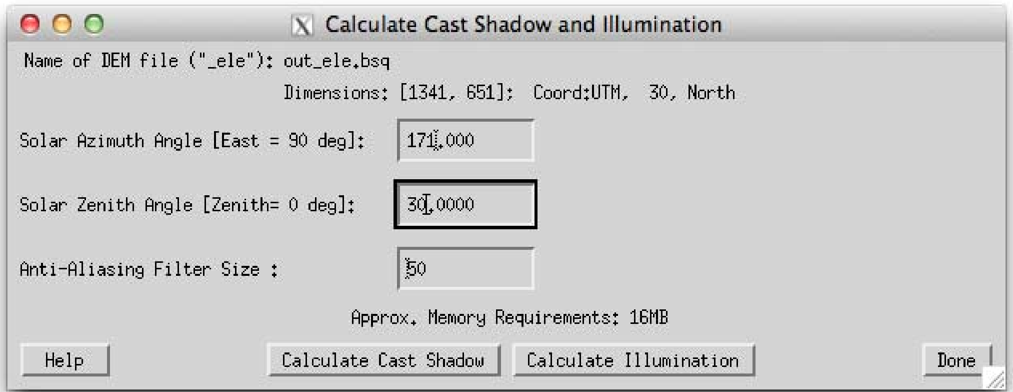
Two files are created from the input file:

- *_asp: file containing the aspect information
  'Aspect [deg/2]' - one band BYTE data
- *_slp: file containing the terrain slope information
  Slope [deg] - one band BYTE data

Procedure:

A cross is calculated over each DEM pixel and the cross product of the two crossing vectors is used for description of a normal vector on the DEM.

SHADING



Shading can be calculated on the DEM using this routine. This is for linking to the ATCOR package.

SKYVIEW FACTOR

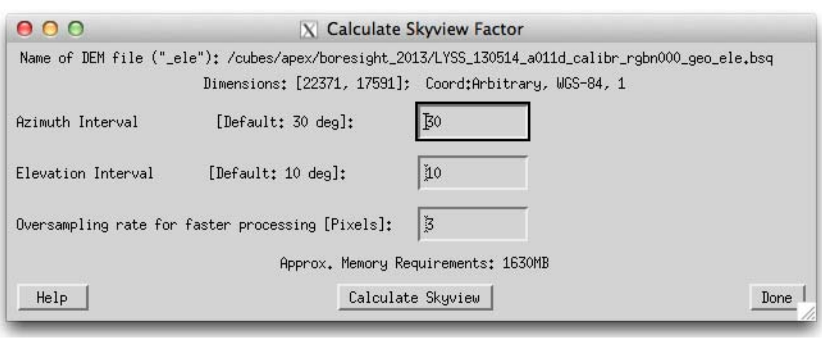


Figure 5.35: DEM skyview calcuation

The two topographic modelling routines have been included in PARGE to ease the link to the ATCOR4 radiometric / topographic correction method. See online help for more details.

ATCOR PREPARATION

For a fully IGM based processing workflow in atmospheric correction, the DEM and its derived products have to be provided in raw geometry, inverted from original elevation data.

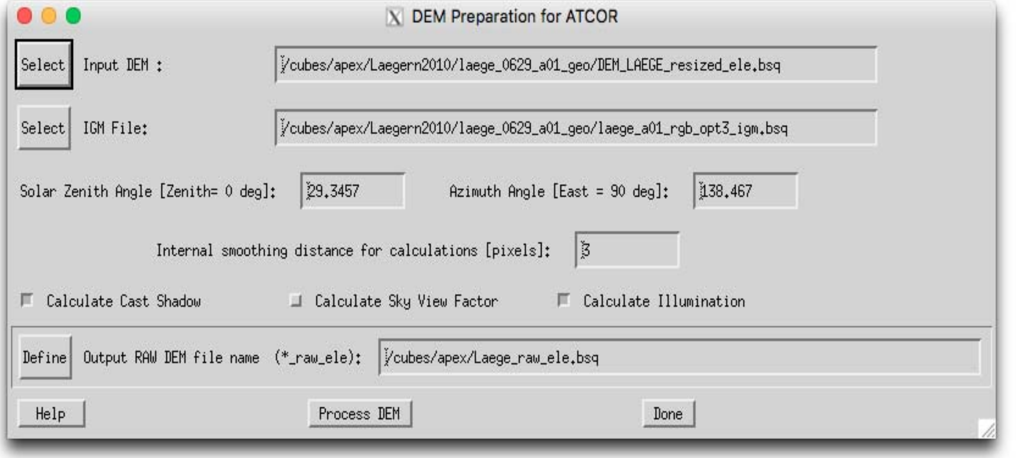


Figure 5.36: Prepare DEM for raw geometry based ATCOR processing

Inputs:

Nr	Description
1	An ENVI DEM to be inverted covering the area of the IGM.
2	IGM (in raw geometry) covering the DEM area.
3	zenith/azimuth angle (used for cast shadow calculation only)
4	internal smoothing distance of DEM to be processed (in pixels) smoothing is required to avoid artefacts in high resolution DEMs.
5	Options: create _shd, _sky and/or _ilu file in raw geometry as well.

Outputs:
A series of files is created in raw geometry
- *_ele.bsq: elevation data in raw geometry
  altitude a.s.l. meters - floating point data
- *_slp.bsq: file containing the terrain slope information
  slope [deg] - one band BYTE data
- *_asp.bsq: file containing the aspect information
  aspect [deg] - one band integer data

- *_shd.bsq: file containing the cast shadow information
  Cast shadow mask (0/255)
- *_sky.bsq: skyview factor
  percentage (byte data 0-100).
- *_ilu.bsq: illumination factor, scaled between 0 and 1 as cosine of incidence.

5.5 Menu: Auxiliary

The auxiliary menu serves as a menu to read, filter and check the auxiliary parameter data streams stemming from the instrument INS system.

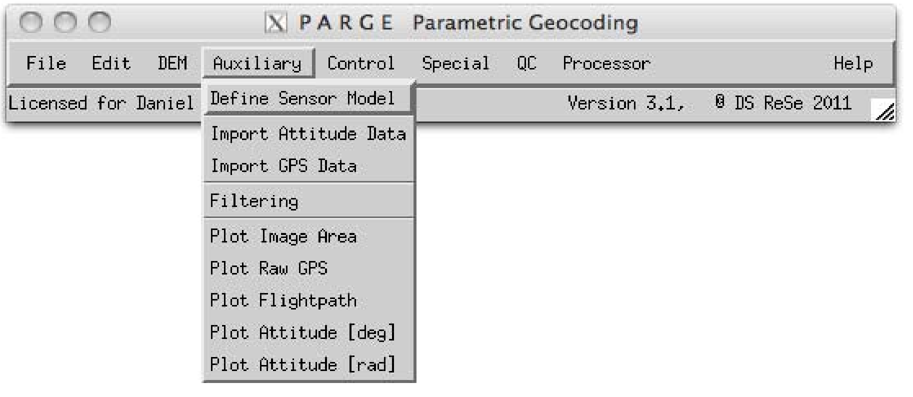


Figure 5.37: The Auxiliary data menu

DEFINE SENSOR MODEL

The purpose of this program is to provide an interface for the definition of pushbroom and whiskbroom standard geometric sensor models or to read an explicite sensor model from file. It defines the fixed geometry of any sensor included into PARGE.

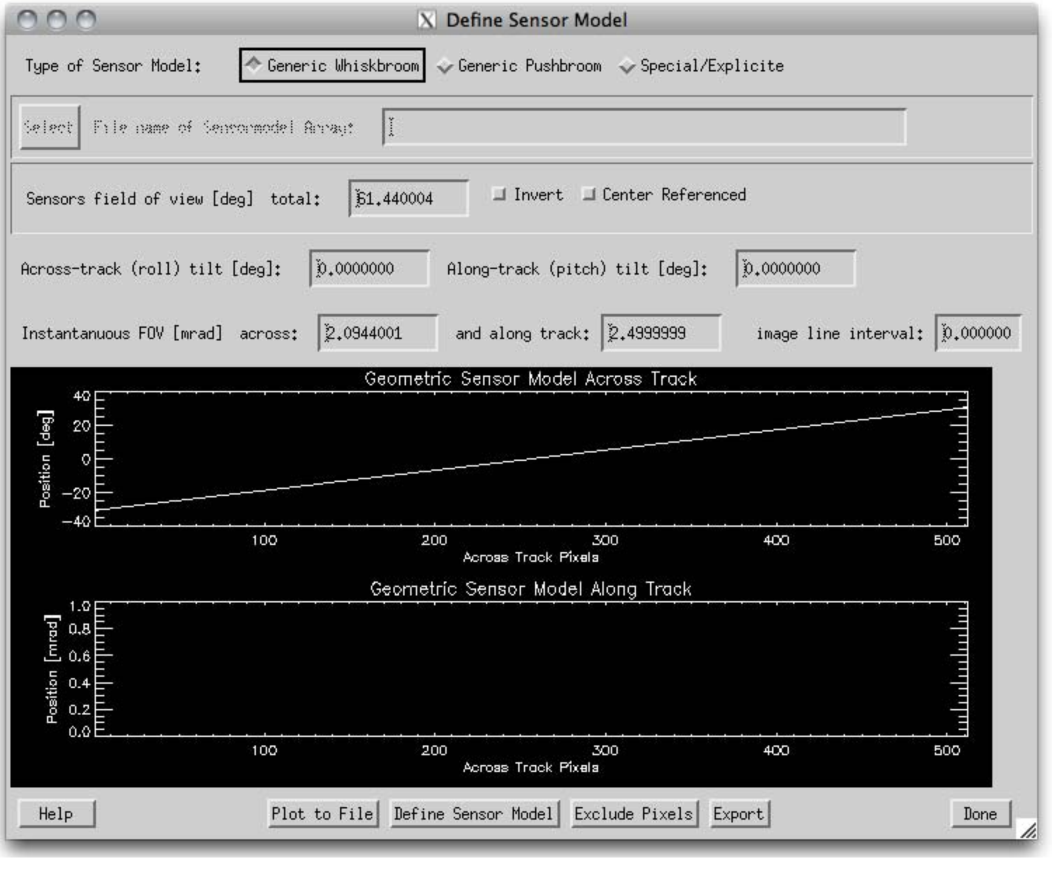


Figure 5.38: Sensor model definition

Inputs

Nr	Description
1	Type of Sensor Model: • Generic Whiskbroom sensor: The pixels are angular equally spaced across track. The rotating sensor leads to an along track delay of the pixels. • Generic Pushbroom Sensor: The projections of the pixels on a plane is equally spaced across track and all pixels are measured simultaneously. • Special/Explicite Sensor Model: The Sensor Model is read explicitely from a file
2	Sensor Model File for explicite definition: Three-columnar ASCII file has to be provided with 1 label/header row, with col1: across track pixel# (starting at 0), col2: across track position to nadir [rad], col3: along track position to nadir [rad]
3	Sensors field of view [deg] total: FOV from edge to edge of sensor outermost pixels in FOV • Invert: scanning from right to left instead of left to right, reflected by a negative FOV parameter • Center Referenced: FOV is given as angle between centers of outermost pixels in FOV (Default is edge to edge of outermost pixels).
4	FOV across-track tilt: offset in across track (roll) direction FOV along-track tilt: offset in along track (pitch) direction
5	IFOV Parameters: • Instantanuous FOV [mrad] across and along track: represents mean overall PSF, FWHM in both directions • image line interval: For whiskbroom scanners: relative difference along track due to scan process duration of one line, given as fraction of the along track sampling interval
6	plot window: displays current and updated sensor model

Actions:

Action	Description
Plot to File	allows to plot the current sensor model to graphics file
Define Sensor Model	assigns the new sensor model definition to the variable sensarr.
Exclude Pixels	allows to exlcude missing/faulty pixels from further processing
Exports	exports the sensor model definition to a columnar text file

Outputs: The PARGE variable `sensarr` and the structure `sensor` are defined

IMPORT ATTITUDE DATA

This is a program to read and prepare the attitude data. The data may be synchronized while importing by line number or by decimal time.

1.
2.
3.
4.
5.
6.
Actions:

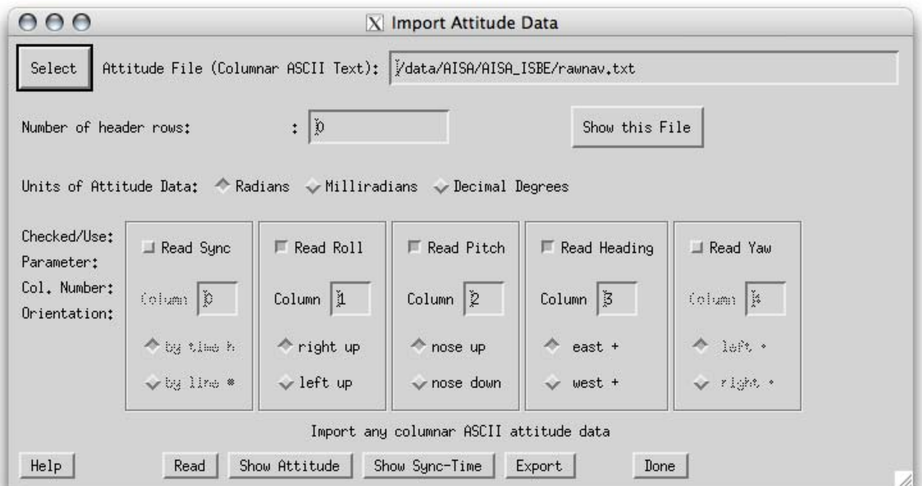


Figure 5.39: Import Attitude Auxiliary Data.

Inputs:

Nr.	Information:
1	Name of the file containing the attitude data - the data should be available as columnar ASCII numbers.
2	Number of header rows in the ASCII file to be ignored while reading. Use the button 'Show this file' to have a look at the selected raw data file.
3	Units of the raw data. The data are converted to radians internally in PARGE if milliradians or decimal degrees are provided
4	Dimensions to be read and updated internally in PARGE. Synchronization is only performed, if the 'Read Sync' option is chosen and the corresponding data set is read (see details below)
5	Columns of raw data file assigned to each parameter. The column numbering starts with '0' for the most left column in the raw data file.
6	Orientation of the parameters and options for the synchronization column.

After reading the data can be visually checked using the corresponding plot-buttons; the offset can be changed afterwards, while the calibration may be done using GCPs.

Actions:

Action:	Information:
Select	The file name is selected. For certain data types the definition for all angles is contained in the file given first.
Show this File	Displays the contents of the currently selected ASCII file in an editor window.
Read	Reads the defined data to the corresponding attitude variable (rollarr, pitcharr, headarr). See Chapter a).
Show Attitude	Displays the attitude parameters.
Show Sync Time	Plots the Synchronization time per image line (if available)
Export	Exports the currently available attitude data together with the synchronization information to a columnar ASCII file.

Details

sync: Synchronization of all imported attitude parameters is performed using the selected row if the box 'Read Sync' is checked.

- by line time: decimal hours should be available for both the attitude data and the image data (syncarr)
- by line #: the scan line number should be available for each attitude measurement line in the input file. The data is linearly interpolated to the number of lines of the image.

roll: The default definition for positive roll is right wing up (in flight direction). Inversion may be selected, e.g. for negative FOV (right to left) scanning systems.

pitch: The default definition for positive pitch is aircraft nose up; choose 'nose down' for the inverse definition.

head: The default definition for heading is positiv from north towards east (north is always zero). Choose 'West +' if positive is from north towards west.

yaw: The yaw parameter usually is not required for processing. It only may be used, if no heading is available and it needs to be reconstructed from the flight direction. Default definition of yaw is positive for a left turn from flight direction. Choose 'right +' for the inverse definition.

IMPORT GPS DATA

This program imports the raw GPS navigation (location) data and optionally synchronizes it to the image lines.

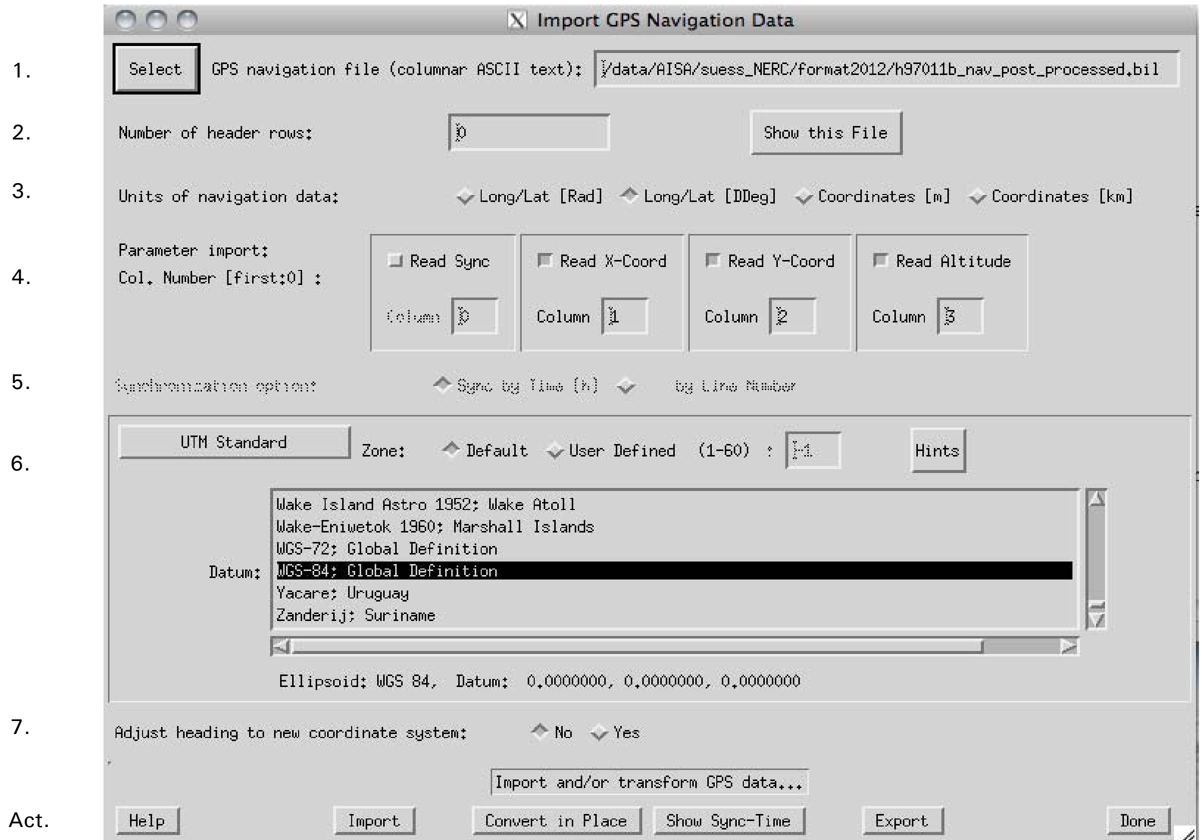


Figure 5.40: Import GPS Navigation Data window

Inputs

Nr.	Information
1	Input filename containing the AUX data.
2	Number of header rows in ASCII formatted input data. Use the button 'Show this file' to have a look at the currently selected auxiliary data file.

Nr.	Information
3	The units for raw data are decimal degrees, meters, or kilometers for Longitude or Latitude and always meters for altitude. If Long/Lat values are provided for the coordinates, it is required to convert them to a metric coordinate system. Enter the appropriate units of the raw file here: 'Long/Lat [DDeg]': Data is given as original GPS Longitude/Latitude in decimal degrees. Altitude is in meters 'Coordinates [m]': All Data to be read is given in meters, - only Navigation data will be assigned 'Coordinates [km]': All Data to be read is given in kilometers, - only Navigation data will be assigned
4	Dimensions: Only the selected dimensions will be replaced from file into the PARGE gpsarr/navarr variable(s). Synchronization is done on the fly if ‘Read Sync’ is selected. Uncheck this dimension if pre-synchronized data is available per image line.
5	Check one of the following buttons which provide two possibilities of synchronization (these options are only available if synchronization information is read from file): • 'Sync by Line Time': A time tag has to be provided along with the auxiliary data for synchronization.The PARGE variable syncarr containing a time tag per image line is used for synchronization. • 'Sync by Line Number': The data is synchronized to the number of lines in the image by interpolating by the line number. The line number must be present along with the navigation data.
6	see Section 5.1.6 on page 124 for details on coordinate convertion
7	Adjust Heading: If the flight heading has already been loaded beforehand, the northing is adjusted based on the chosen coordinate conversion - the reason being that GPS north is normally different from the north definition of the local coordinate system.

Actions

Action:	Information
Select	The file name is selected, usually containg all location information.
Import	Reads the defined data to the navigation variables (gpsarr and navarr); multiple dimensions may possibly read.
Convert in Place	The currently available Navigation data is converted in place instead of reading it from a file. Use this option, if the navigation data is available in decimal degrees after data import to transform it to a metric system.
Show Sync Time	Displays the currently available synchronization time per image line. This time tag is required for the option 'Sync:by Time h'.

Action:	Information
Export	Exports the currently available navigation data together with the synchronization information to a columnar ASCII file. This action is useful for later external coordinate transformation and import back into PARGE by this very same interface

Output:

The output is a synchronized flightpath in the PARGE-variables 'gpsarr' and 'navarr'

The coordinates are converted to the chosen zone and datum.
The heading is offset according to the coordinate conversion parameters. For this reason, the attitude data has to be read prior to coordinate conversion.

An optional output is the exported flightpath per image line as columnar ASCII file.

Formats:

The DGPS shall be available in standard columnar ASCII format.

Read Generic Auxiliary Data

This function may appear as sub-function when reading raw data within PARGE, specifically with certain functions in the 'Special' menu.

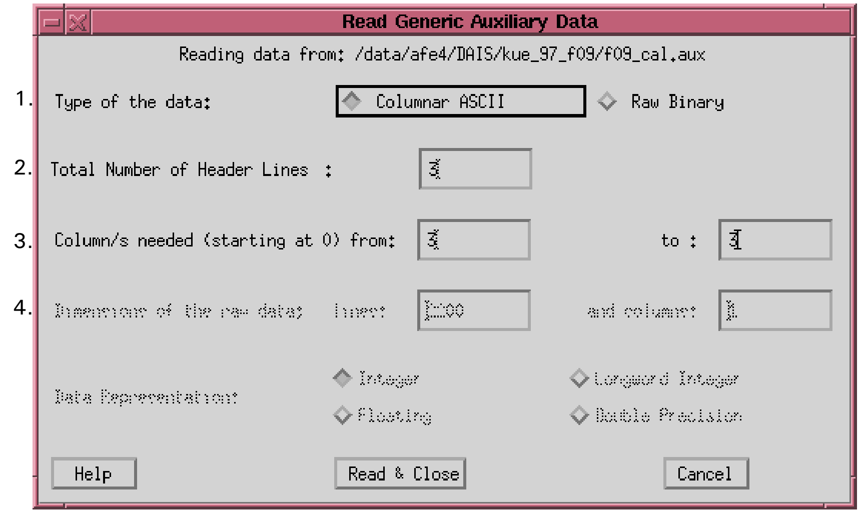


Figure 5.41: Read Generic Auxiliary Data

Inputs (Numbers 1-4):

Nr.	Information:
1	Selects if the file is binary or ascii formatted.
2	Defines the header size (bytes OR lines).
3	These are the needed columns. Usually only one column can be selected; in that case only the first field is used or the same number is typed twice.
4	The raw file specifications (for binary only) allows to read any file of the given formats as long as the whole filesize is defined.

FILTERING

Program to filter the auxiliary data and to apply offsets interactively. The filtered preview appears dashed in the plot window and replaces the original data as soon as the save button is pushed.

1.

Actions:

Actions:

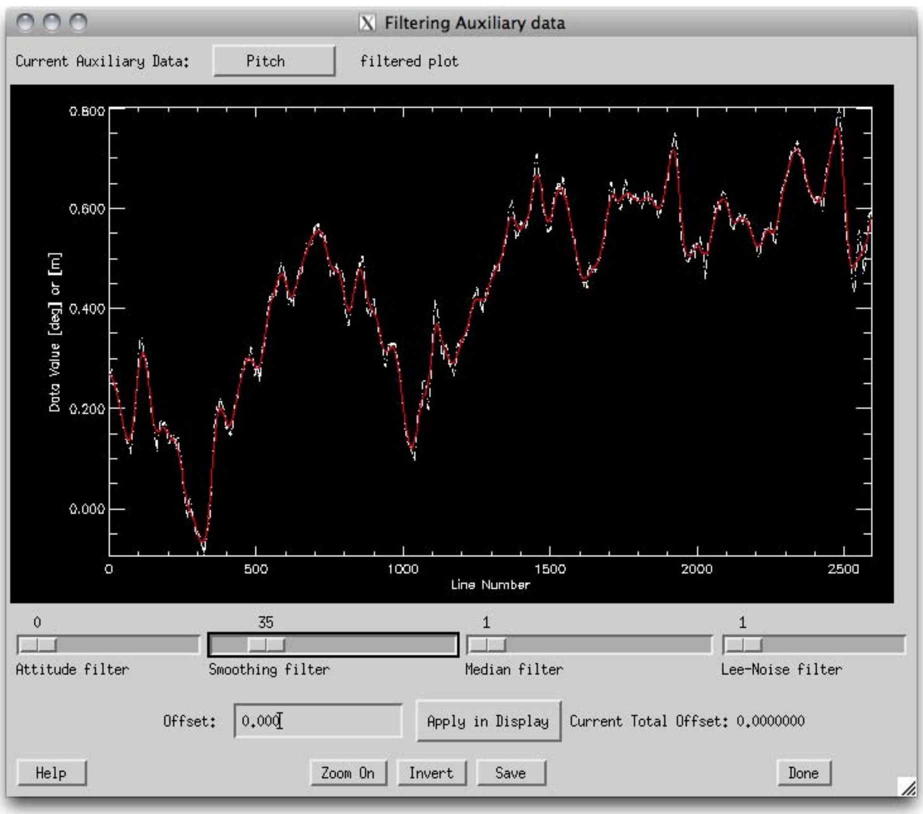


Figure 5.42: Filtering Auxiliary Data

Input:

The appropriate auxiliary data record can be selected (all auxiliary data can be chosen individually) – no filtering is available for the sensor model parameter.

Actions:

Action:	Information:
Attitude Filter	Corrects apruptly changing offsets given by inconsistency and spikes in the data. The slider value scales the relative strength of the filter; higher values filters scale even small inconsistencies.
Smoothing Filter	Allows correction using the number of elements of the data given by the slider.
Median Filter	Preserves the shape (usually better than smoothing) by the number of elements of the data given by the slider.
Lee-Noise Filter	Makes the noise reduction using an IDL standard method (see IDL help of function LEEFILT for details).
Offset: Apply in Displays	Offsets the dataset by the given value. The sum of all applied offsets is given to the right. Values for attitude data are given in degree: attention: the offset is only transferred to b
Zoom On/Off (also by click in graphics window)	Zooms to the 60 values close to the click position. This option is useful if a parameter is affected by noise and the effect of the various filters shall be previewed in detail. Another click brings you back to the full range display.
Invert	Mirrors the given data on the x-Axis (makes sense for attitude data, if orientation is unknown).
Save	Saves the filtered variable to the corresponding PARGE variable (i.e. rollarr, pitcharr, headarr, yawarr, navarr).

Outputs:

The save button saves the filtered result to memory; the unfiltered roll/pitch/heading or navigation data can only be restored from a previously saved status or by selecting the 'UNDO' function.

Note:

- A zoomed display of the data values is displayed when clicking in the required position of the graph.
- The filtered data is previewed whereas the original data is displayed as dashed line. The selected filter or offset is only applied when clicking the save button.

PLOT IMAGE AREA

A proximate guess of the image area on the surface is plotted. It uses both the navigation and attitude data and virtually extrapolates these position to a zero meter altitude level. Thus, in normal landscapes the image area may be smaller than the plotted borders (depending on surface altitude.

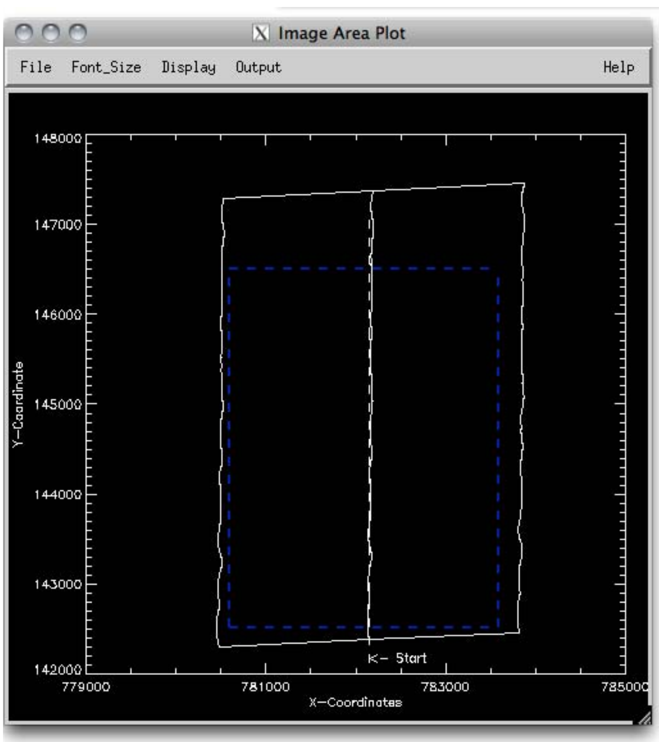


Figure 5.43: Image Area Plot

Use this functions to get an idea about the real extent of the imagery on the surface. The function may also be useful to cut the DEM to the desired size. The limits as given by this function are also used if 'guess range' is pressed while cutting a DEM.

The screen may be updated by using the 'reload' button from the Display menu.

The following lines are plotted:
•Guess of image border,
•Middle pixel line,
•Aircraft flightpath (dashed), and
•Area of DEM (thick dashed).

PLOT RAW GPS

The flightpath geographic coordinates (variable `gpsarr` containing longitude, latitude, altitude) are plotted individually, analoguous to the function **Auxiliary:Plot Flightpath** p.193.

PLOT FLIGHTPATH

The flightpath geodetic coordinates (variable `navarr`, x,y,z, x/y) are plotted individually. For a better comparison with the terrain the command **>Control:Display Flightpath on DEM<** p.202 can be used.

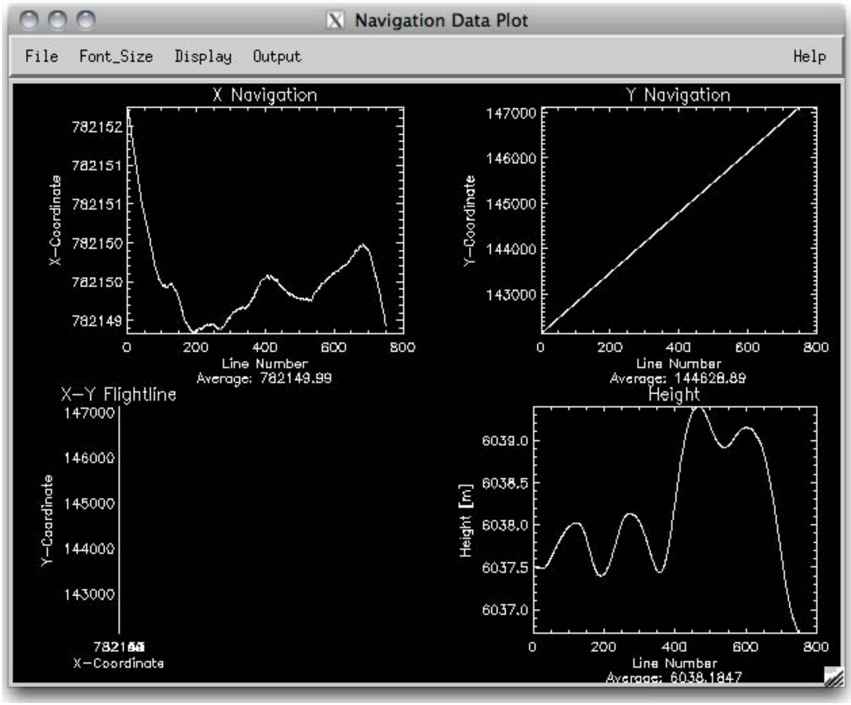


Figure 5.44: Plot Flightpath

PLOT ATTITUDE [DEG]

These are degree plots of the four attitude parameters Roll, Pitch, Yaw and Heading. The same plots are available in radians (*Auxiliary:Plot Attitude [rad]*).

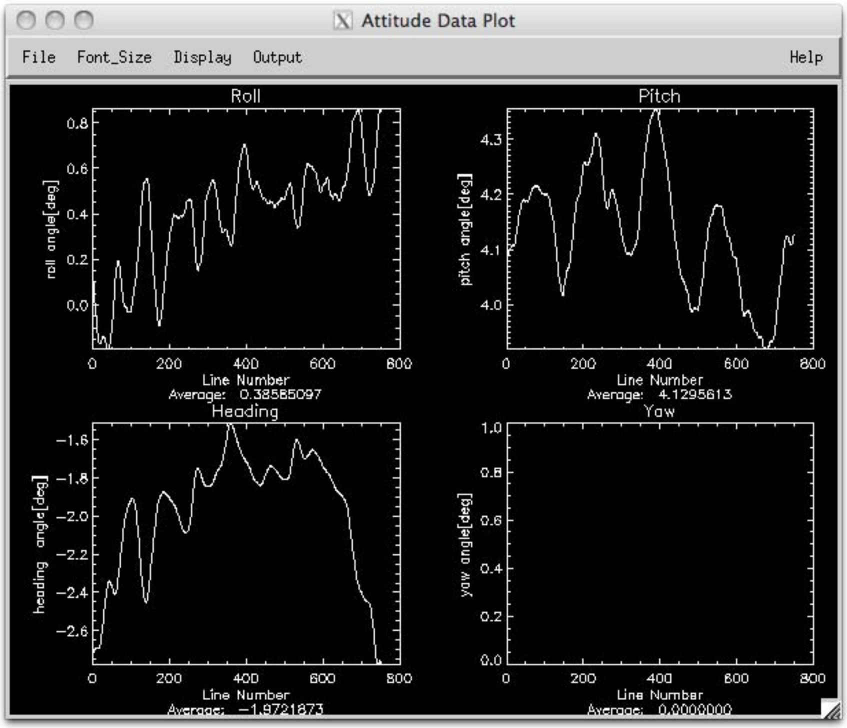


Figure 5.45: Plot Attitude [deg]

PLOT ATTITUDE [RAD]

This function will produce the same plots as shown in Figure 5.45 in radians (being the PARGE internal data values).

5.6 Menu: Control

A minimal number of GCPs is required as soon as the offsets of the airplane attitude data are unknown (e.g. uncalibrated cartographic north or horizontal direction).

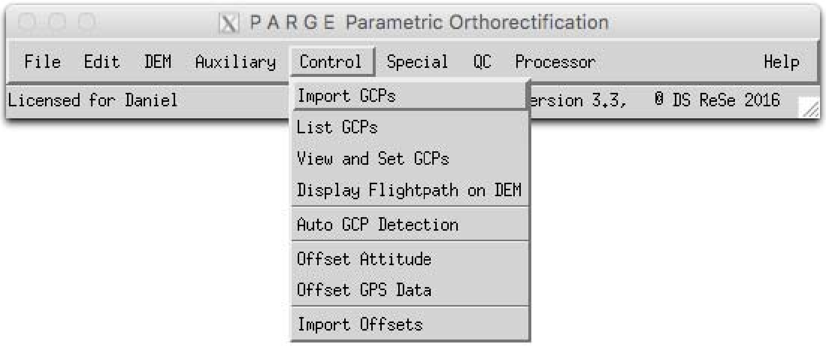


Figure 5.46: Control Menu.

It's theoretically sufficient to use one good GCP to estimate either x/y or roll/pitch-offset values. More GCPs are required, if heading or aircraft altitude offsets have to be estimated. In the special case of flightpath reconstruction more GCPs are necessary, distributed evenly over the image scene. Moreover, one may want to have additional independent GCPs for accuracy assessment.

The below functions may be used to work with ground control points and update aircraft parameters.

Note: Coordinates of GCPs have to be in the same geodetic system (projection, ellipsoid, geodetic datum) as the DEM and the flightline, both in position and height!

IMPORT GCPS

This function loads the Ground Control Points from ENVI, PCI or other standard formatted text files. The point are loaded either from PCI GCP segment (recommended tool for GCP collection is PCI's GCPworks) or from ENVI GCP file (collected within ENVI and exported: Save GCPs w/ map coords).

1

2

3

Act.

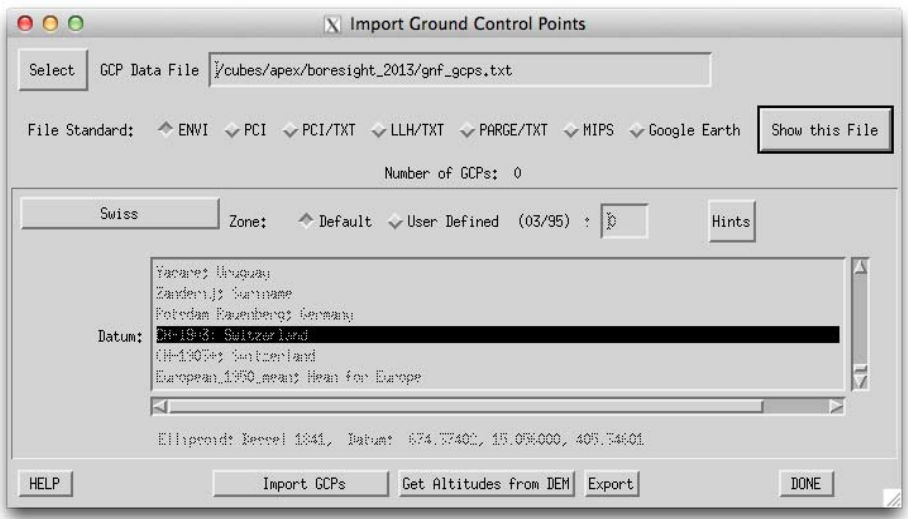


Figure 5.47: Import Ground Control Points.

Inputs:

Nr	Description
1	File Name for GCP import
2	Format of File • ENVI: GCPs saved with >Save GCPs w/ map coords< (*.pts file) • PCI: GCP stored in the regular GCP segment of a pcidsk file • PCI/TXT:GCPs stored as text from PCI using the 'XYPL' • formatting option (no header line) • LLH/TXT: GCPs stored as text using a 'Lat/Lon/Alt/Line/Pix' • formatting option (no header line) • TNT-Mips: GCP file saved by TNT-Mips • PARGE: GCPs saved from PARGE through > Export:GCPs < • Google Earth: GCPs saved from Google Earth as KML points • file (pixel coordinates are to be added manually) (the file may be displayed using the > Show < Button, except for PCI segments)

Nr	Description
3	Conversion: Ground Control Points Coordinates often come in GPS derived WGS-84 Geographic (long/lat) Coordinates. This coordinates need to be converted to a metric system. You may convert your coordinates to either of the available systems while importing. If 'No Conversion' is chosen, the data need to be converted and imported elsewhere. Use the function >Auxiliary:Import DGPS data< or any external software for that purpose. >> ATTENTION: Do not attempt to convert GCPs collected in metric coordinates again using this tool<<

Actions:

Action:	Information:
Import GCPs	Opens and reads the selected GCP file.
Get Altitudes from DEM	Calculates and stores the height above sea level of all GCPs by reading the DEM altitude at the given coordinates.
Export	Exports the currently given GCPs to a simple ASCII text file, which may be transferred to another image by the 'Get from PARGE' function.

Outputs:

The ouput is an editable list of the GCPs, given in the PARGE variable 'gcparr', with its description in the 'image' structure variable.

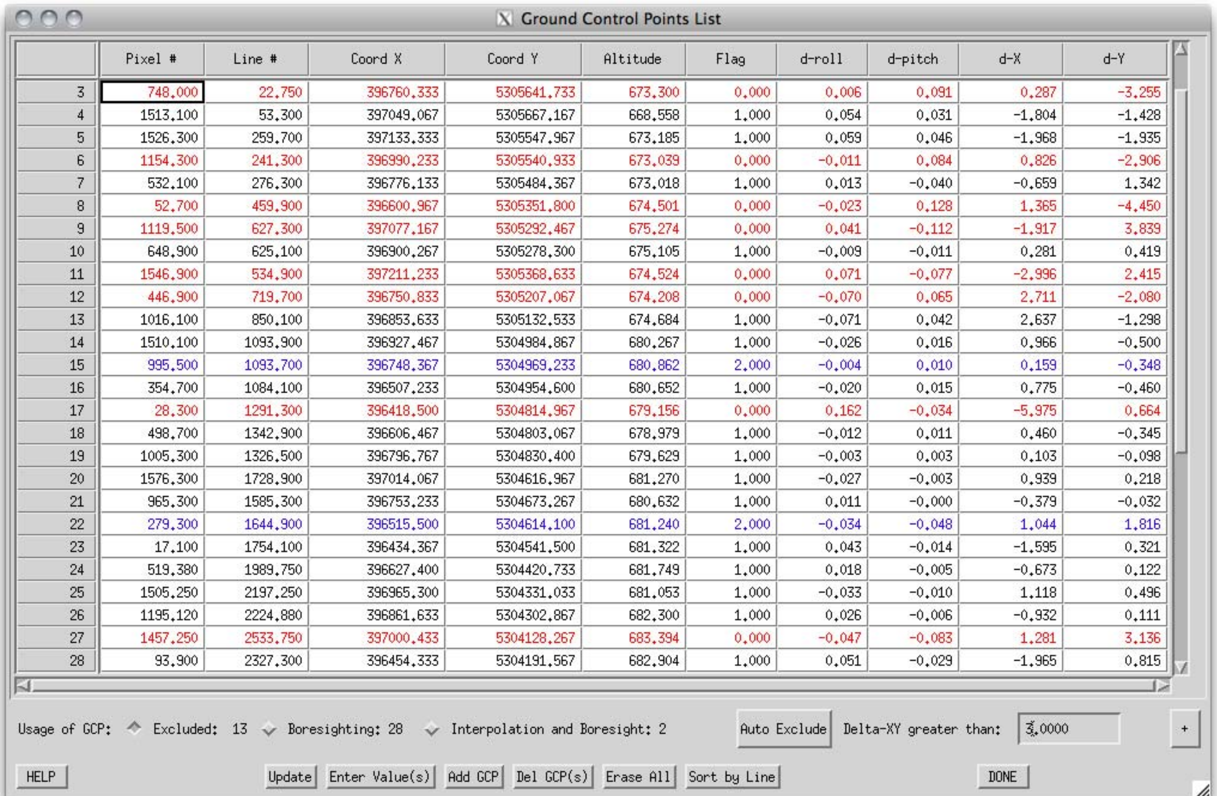
Details

- The procedure eads the GCPs from files. For ENVI formatted GCPs, the pixel referencing is shifted from upper-left pixel to the pixel center.
- If new GCPs are imported and their coordinates are the same as available GCPS, the new GCPs are ignored

Attention: If the GCPs were collected in ENVI, the xstart and ystart values have to be considered! (see ENVI-image-header) The import filter assumes that those variables have been set to 1/1.

LIST GCPS

This function displays the GCPs in a table where single points can be edited, added, and deleted manually.



	Pixel #	Line #	Coord X	Coord Y	Altitude	Flag	d-roll	d-pitch	d-X	d-Y
3	748,000	22,750	396760,333	5305641,733	673,300	0,000	0,006	0,091	0,287	-3,255
4	1513,100	53,300	397049,067	5305667,167	668,558	1,000	0,054	0,031	-1,804	-1,428
5	1526,300	259,700	397133,333	5305547,967	673,185	1,000	0,059	0,046	-1,968	-1,935
6	1154,300	241,300	396990,233	5305540,933	673,039	0,000	-0,011	0,084	0,826	-2,906
7	532,100	276,300	396776,133	5305484,367	673,018	1,000	0,013	-0,040	-0,659	1,342
8	52,700	459,900	396600,967	5305351,800	674,501	0,000	-0,023	0,128	1,365	-4,450
9	1119,500	627,300	397077,167	5305292,467	675,274	0,000	0,041	-0,112	-1,917	3,839
10	648,900	625,100	396900,267	5305278,300	675,105	1,000	-0,009	-0,011	0,281	0,419
11	1546,900	534,900	397211,233	5305368,633	674,524	0,000	0,071	-0,077	-2,996	2,415
12	446,900	719,700	396750,833	5305207,067	674,208	0,000	-0,070	0,065	2,711	-2,080
13	1016,100	850,100	396853,633	5305132,533	674,684	1,000	-0,071	0,042	2,637	-1,298
14	1510,100	1093,900	396927,467	5304984,867	680,267	1,000	-0,026	0,016	0,966	-0,500
15	995,500	1093,700	396748,367	5304969,233	680,862	2,000	-0,004	0,010	0,159	-0,348
16	354,700	1084,100	396507,233	5304954,600	680,652	1,000	-0,020	0,015	0,775	-0,460
17	28,300	1291,300	396418,500	5304814,967	679,156	0,000	0,162	-0,034	-5,975	0,664
18	498,700	1342,900	396606,467	5304803,067	678,979	1,000	-0,012	0,011	0,460	-0,345
19	1005,300	1326,500	396796,767	5304830,400	679,629	1,000	-0,003	0,003	0,103	-0,098
20	1576,300	1728,900	397014,067	5304616,967	681,270	1,000	-0,027	-0,003	0,939	0,218
21	965,300	1585,300	396753,233	5304673,267	680,632	1,000	0,011	-0,000	-0,379	-0,032
22	279,300	1644,900	396515,500	5304614,100	681,240	2,000	-0,034	-0,048	1,044	1,816
23	17,100	1754,100	396434,367	5304541,500	681,322	1,000	0,043	-0,014	-1,595	0,321
24	519,380	1989,750	396627,400	5304420,733	681,749	1,000	0,018	-0,005	-0,673	0,122
25	1505,250	2197,250	396965,300	5304331,033	681,053	1,000	-0,033	-0,010	1,118	0,496
26	1195,120	2224,880	396861,633	5304302,867	682,300	1,000	0,026	-0,006	-0,932	0,111
27	1457,250	2533,750	397000,433	5304128,267	683,394	0,000	-0,047	-0,083	1,281	3,136
28	93,900	2327,300	396454,333	5304191,567	682,904	1,000	0,051	-0,029	-1,965	0,815

Figure 5.48: Ground control points list.

Inputs

The values as displayed in the GCP list may be edited directly in the list (this option may not work on all operating systems as expected...). The flag may also be set by using the buttons below the list, which apply to the last selected GCP.

0: The point is not used for calculation of offsets nor for interpolation (Its individual parameters are displayed anyhow).

1: The point is used for offset calculation and potentially for drift correction (Functions: > File:Offset Attitude <, > File:Offset GCPs <, > Special:Drift Correction <)

2: The point is used as with flag=1, but also for parameter interpolation (Functions: > Special:Interpolate Flightpath <, and > Special:Interpolate Parameter <).

The GCPs are color coded, such that excluded points (flag 0) appear in red and interpolation points (flag 2) are blue.

Actions:

Action:	**Information:**
Auto Exclude	This function excludes all points whose horizontal offsets are greater than the given value in meters.
Update	Updates the list of GCPs and the Offset values (e.g. after applying an offset to one of the parameters).
Enter Value(s)	Enter a numeric value in all selected cells. This option is handy to set the flags of all GCPs to the same value.
Add GCP	Adds a new GCP to the table and manually inputs the pixel/line/coordinate values (new GCP is added after the last GCP).
Del GCP	Deletes selected GCPs (whole lines) in the table.
Erase All	Deletes all GCPs in LIst.
Sort by Line	Sorts the GCPs by the line number. This is a useful action to figure out possible drifts and systematic errors (e.g. missing lines) during data acquisitions.

Details

- The GCP list resizes dynamically only, if the window is closed and re-opened from the PARGE main menu.
- Sorting by line of the GCPs is recommended to find potential drifts or other systematic problems in aux-data.
- The ‘+’ Button serves to extend the length of the list.

VIEW AND SET GCPS

This is a crude interface to collect a GCP in the image and assign a coordinate. The function displays the GCPs on the top of the single channel image. Using a dedicated product for GCP collection is highly recommended.

Figure 5.49: View and Set Ground Control Points

Inputs:

Inputs:	Information:
Move cursor on zoom	Diplays the position of the cursor below the zoom image
Nr.1	Shows the coordinates of the ground control point in meters.

Inputs:	Information:
Mouse right	Gives the position of the ground control point within the image.
Mouse click left	Gives the position of the cursor.

Actions

Standard scaling options are given as described in Display Single Channel on page 155. Additional available actions are:

Action:	Information:
Edit: Flip Image Vertically	The image is flipped vertically (horizontal axis), i.e. the order of the image is changed from top-down to down-top.
GCP: Add Point	Saves the last selected point as diplayed on the rigt side of the image to the list.
GCP: Purge Point	Delete a GCP with the point number xx (interactive input of number).
GCP: Show List	Displays the currently active list of GCPs.
Guess Position	The position of the last selected point is estimated from the flight parameters.
Guess Altitude	The altitude of the point is estimated from the currently diplayed coordinates using the DEM altitude

Outputs:

The output is an updated list of GCPs (variable gcparr) and a plot of the GCPs on the image.

DISPLAY FLIGHTPATH ON DEM

This function plots the flightpath on top of the currently loaded DEM and indicates its flight direction.

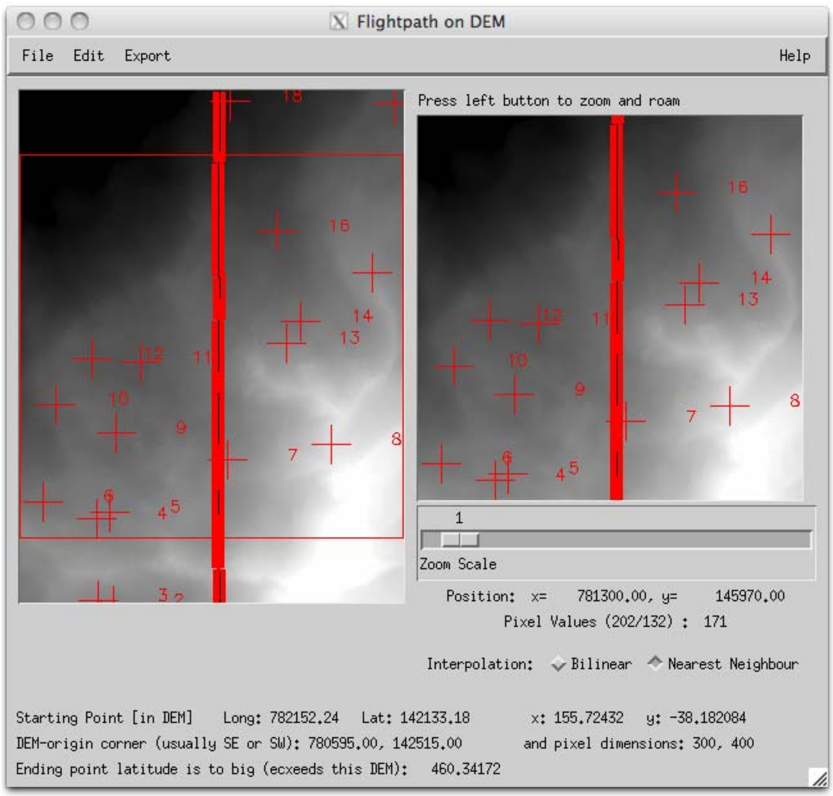


Figure 5.50: Display Flightline on DEM

Inputs:

The DEM and the DGPS aircraft flightpath data must have been read before.

Outputs:

The output is a plot of the flightline on the DEM the GCPs, and the image borders; there are also shown the coordinates of the flightline's start in the DEM, in pixel- and map-coordinates.

Procedure:

A click in the window shows the cursor location coordinates; the flightline should be checked visually and by examining the given coordinates (starting point and DEM corner).

AUTO GCP DETECTION

If an absolute image reference (e.g. an Orthophoto) is available, this routine may be used for automatic GCP retrieval provided the image can be rectified to a reasonable accuracy at first.

Preparation:

1) import the image data as usual
2) rectify an RGB or just the green band of the imagery to a preliminary georectified image
3) resize the ortho image to the same dimensions as the rectified image (you may use the image display routine resizing utility for this purpose)

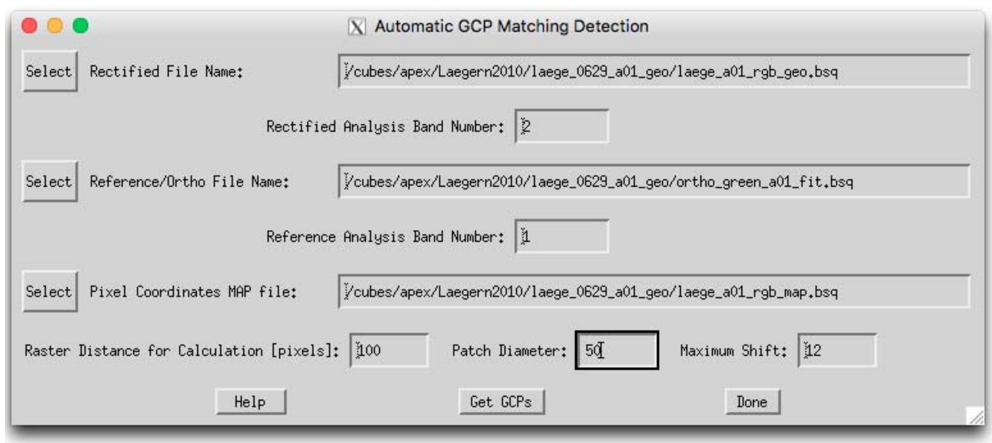


Figure 5.51: Auto GCP detection through image matching - panel.

Inputs:

Name of ENVI formatted input files (rectified file and geometric reference): both images have to be in BSQ storage order and the two files must have the same spatial dimensions in both location and resolution.

Bands to be used for correlation analysis (one band per file only); by default the green spectral band is used. Band numbering starts at 1.

Pixel coordinates MAP file: *_map file containing the pixel and line coordinates of the rectified image (as created in the preparation step)

Raster Distance for Calculation: Distance step at wich in both x and y dimensions correlation analyses shall be performed.

Patch Diameter: diameter of quadratic patch taken at raster points used for correlation analysis. This distance may need to be reduced in steep terrain (normally, the raster distance is a good choice at first).

Maximum Shift: Maximum shift in pixels around raster points to search for optimal correlation. Correlation is search in 1-pixel steps only here.

Actions:

> Get GCPs <

Calculates best correlations in proximity of the raster points and stores the coordinates of the raster points for use as GCPs in the Offsets estimation tools

Procedure:

The routine reads corresponding bands simultaneously and does a systematical correlation analysis. The results are stored as new GCPs for further use (as soon as the correlation coefficient is better than 0.6). After this analysis, the offsetting procedure (Control:Offset Attitude) and Exclusion process (Control:List GCPs) is to be performed and the parameter can be optimized to the ortho-referencing position.

OFFSET ATTITUDE

This tool may be used for offset of the attitude data using Ground Control Points (GCPs). A reasonable number of GCPs (i.e. 10-20) needs to be loaded for this analysis. If less than 5 GCPs are available, heading offsetting becomes very inaccurate to impossible.

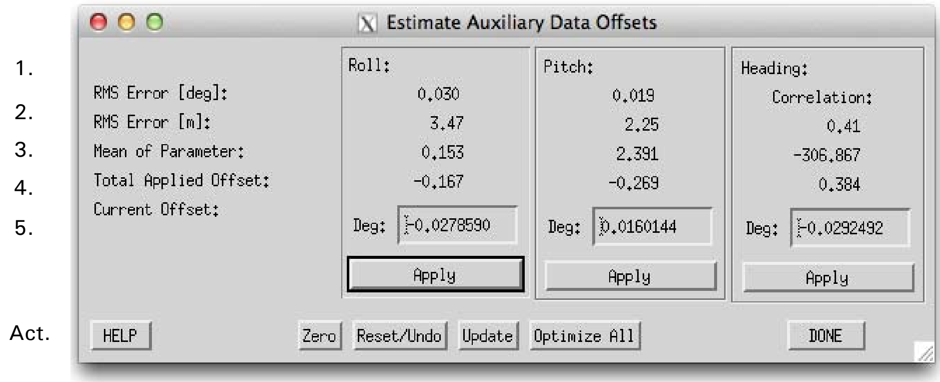


Figure 5.52: Estimate Attitude Data Offsets

Inputs:

Nr:	Description:
1	RMS of all GCPs against target position, degrees (n.a. for heading)
2	RMS of all GCPs against target position, meters in roll or pitch direction. For the heading parameter, the correlation factor between nadir position and offset value is shown (less correlation means better agreement of the model).
3	Average of the parameter (degree)
4	Offset of the parameters to the original raw data values
5	Current offset as calculated from GCPs - field may be edited for other offsets

Actions:

Action:	Information:
Zero	The boresight information is set to zero; this is used to set a reference point when starting a new boresighting (e.g. after heading offsets have been applied while importing GPS data).
Reset	The parameters are reset to their original state, ie. removing all applied offsets.

Action:	Information:
Apply	The displayed offset is subtracted from the respective parameter.
Update	The statistics are updated based on the current status of the GCPs and their flags..
Optimize all	Iterative offset calculation and reduction for all three parameters.

Outputs:

The outputs are offset values for roll, pitch, and heading, and offseted data rollarr, pitcharr, and headarr.

Procedure:

the variables rollarr/pitcharr/headarr are updated and offsetted according to the selected offsets. the aux(4:6).offset variable is updated.

Use the function >Control:List GCPs< to view the effects of the applied offfsets on the individual GCPs.

OFFSET GPS DATA

This tool may be used for correcting the GNSS(e.g. GPS) navigation data. Current GNSS technology is very accurate and it is therefore not recommended to offset the x/y coordinates. Since the altitude sometimes suffers from inaccuracies due to the GNSS or due to the geoid settings, it may be offset using this tool separately. Applying x/y offsets may only be useful, if the GPS data is ill-posed or if some data is of unknown geographic datum/coordinate system (with systematic linear offsets).

NOTE: A reasonable number of GCPs (i.e. 10-20) needs to be loaded for this analysis. If less than 5 GCPs are available, altitude offsetting becomes very inaccurate or may be impossible.

1.
2.
3.
4.
5.
Act.

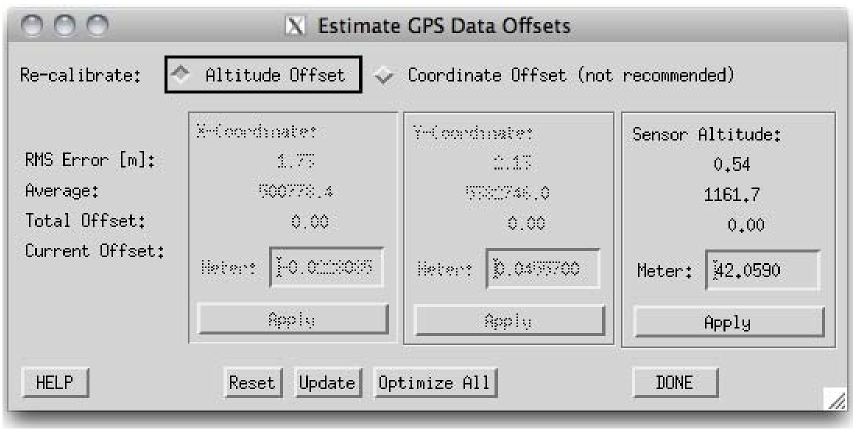


Figure 5.53: Offset GPS Data

Inputs:

Nr:	Description:
1	Select affected parameters. By default, x/y offsets are disabled but may be enabled using this button.
2	RMS of all GCPs against target position, meters in x/y direction, while for the altitude the correlation between altitude offset and relative pixel position is shown.
3	Average of the parameter (meters)
4	Offset of the parameters to the original raw data values
5	For each parameter, the estimate offset in meters is displayed. It also may be entered manually for special situations.

Actions:

Action:	Information:
Reset	The parameters are reset to their original state, ie. removing all applied offsets..
Apply	The displayed offset is subtracted from the respective parameter.
Update	The statistics are updated based on the current status of the GCPs and their flags..
Optimize all	The offsets of the active parameters are iteratively offset in three iterations - If z only is selected two iterations are done on the z offset (same as pushing the 'apply' button twice).

Outputs:

The output is a corrected aircraft altitude information, and possibly shifted flightpath positions.

Note: X/Y navigation should only be changed if no DGPS flightpath was available or it is assumed that the flightpath has to be shifted (for example, due to inaccuracies in projection/ datum transformation of the DGPS coordinates).

IMPORT OFFSETS

This functions import the offset from an external status file and applies them to the currently active parameters (ie., roll/pitch/heading and x/y/z offsets). Only the offsets will be imported from the external file and they are applied directly to the current parameters. All other data remains unchanged.

Use this function to process a series of flight lines of the same data acquisition, where a boresight alignment is typically done on a geometric calibration scene with good ground control points and any subsequent runs are taking the calibration from there. The GCS status file is restored and the offsets are read. They then are applied to the current data while tracking the updated offsets in the auxiliary data aux(*).offset.

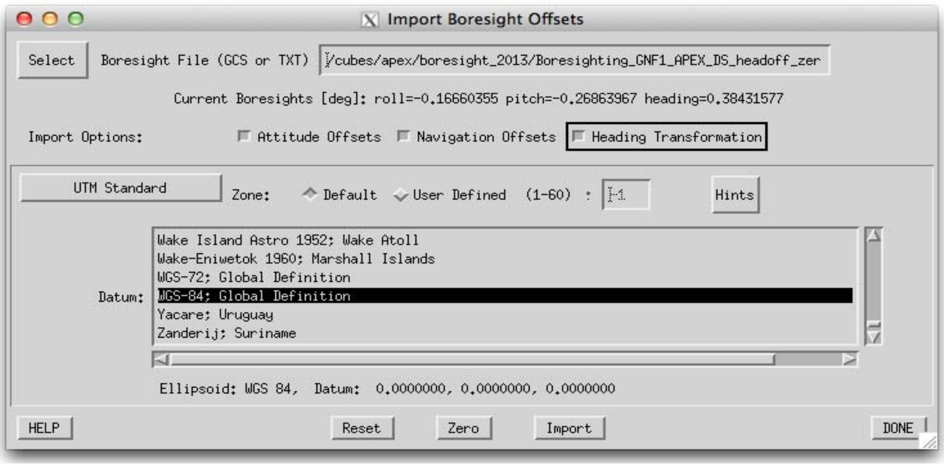


Figure 5.54: Boresight offset import dialog.

Inputs:

Nr:	Description:
1	GCS status file from original calibration campaign to be applied to the current data.
2	Select to which parameters the offsets shall be applied; heading transformation transforms the heading to the current coordinate system (as displayed in the coordinate definiton section of the panel).if e.g. the boresighting has been done in a different coordinate system, the heading transormation should be selected when importing the boresight offsets. However, if the 'north' definition is the same you shouldn't select this option.
3	Coordinate system to be applied (required for heading transformation only)

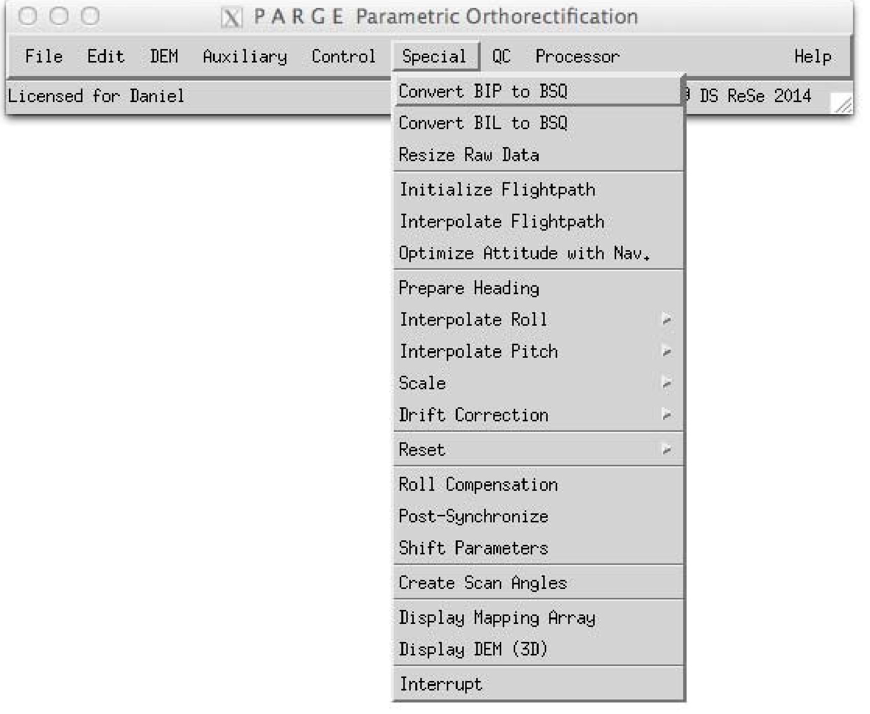
Actions:

Action:	Information:
Reset	Restores the original state of the selected data group by adding the formerly subtracted offsets.
Zero	Thissetting the offset meta data to zero while the GPS/IMU data themselves remain untouched (use this function before starting a boresighting)
Import	Imports the selected offsets and applies them to the data directly.

Attention: don't import the offsets multiple times as the auxiliary data is change each time

5.7 Menu: Special

The functions in the 'Special' menu can be very helpful as soon as some data are corrupted or missing. In regular processing, these functions are not required.



CONVERT BIP TO BSQ

This function converts BIP (Band Interleaved by Pixel) representation files to BSQ (Band Sequential), which is much more convenient to be treated by PARGE and also for atmospheric correction. This function is recommended for use with any standard data which is not in Band sequential order.

A dialog appears which let's select the input file. The output file and its name is created automatically and may be renamed on the Operating system level for further processing.

CONVERT BIL TO BSQ

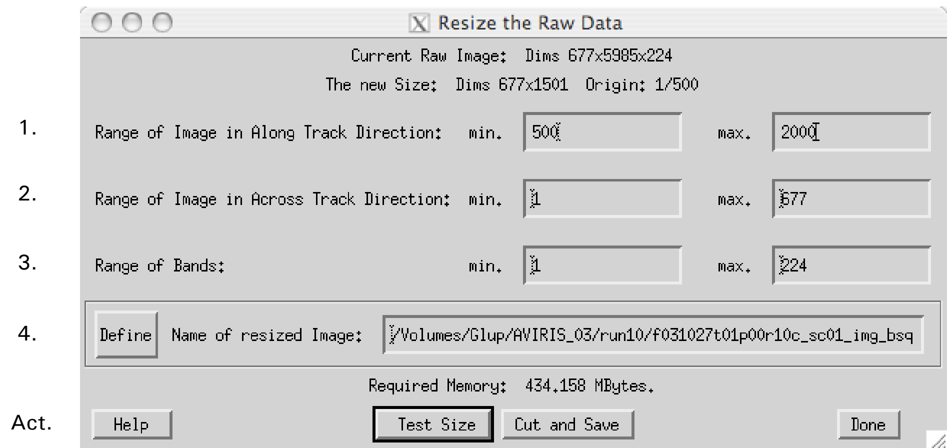
This function converts BIL(Band Interleaved by Line) representation files to BSQ, which is required to treat data by PARGE which originally is distributed in BIL format.

A dialog appears which let's select the input file. The output file and its name is created automatically and may be renamed on the operating system level for further processing.

Note on data representation:
BIP: Data is stored in dimension order [n_bands, n_cols, n_rows]
BIL: Data is stored in dimension order [n_cols, n_bands, n_rows]
BSQ: Data is stored in dimension order [n_cols, n_rows, n_bands]

RESIZE RAW DATA

Long data runs may be cut in handy pieces using this functions. It allow to cut the image cube and all associated auxiliary data to new dimensions, which remain self-consistent.



Inputs:

Nr:	Description:
1	Along track subset (number of lines), affects all INS parameters.
2	Across track subset, affects the geometric sensor model.
3	Band subset, affects the ENVI header and atmospheric correction options.
4	Output file name.

Actions:

Action:	Information:
Test Size	Tests the Size of the new image.
Cut and Save	Cuts to the desired dimensions and stores the new cube.

Outputs:

The output is a resized image file and resized interinal PARGE data structure. Note that after resizing, the original auxiliary data stream are no longer compatible to the resized data streams.

INITIALIZE FLIGHTPATH

With this function a straight flightpath can be constructed artificially. For the interpolation of a formerly unknown flightpath (page 214) a first guess of the flightline has to be estimated by setting a starting and ending point of the image data. This procedure creates a linear flightpath between two points. It is specifically useful, to start if a flightpath has to be interpolated from GCPs afterwards. This procedure should only be used, if no flightpath information is available. The former flightpath information given in the variable 'navarr' is overwritten!

1.

2.

Actions:

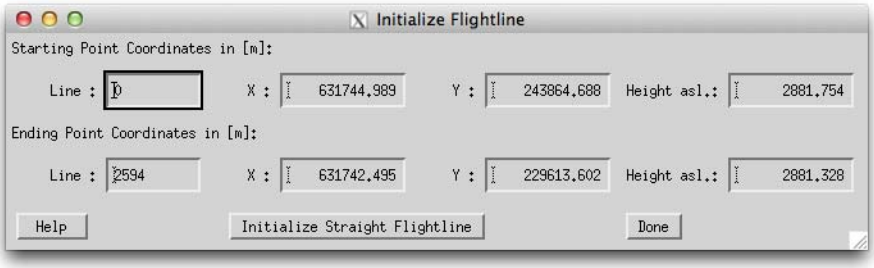


Figure 5.55: Initialize Flightpath

Inputs:

Nr.	Information:
1	Image line number and starting Point Coordintaes in meters; should correspond to the chosen DEM.
2	Image line number and ending Point Coordinates; should correspond to the chosen DEM.

Actions:

Action:	Information:
Initialize Straight Flightline	Creates a linear flight path through the two given points covering all scan lines.

Outputs:

The output is a linear flightpath between starting and ending point, which is stored in the variable 'navarr'; the function overwrites previous information of 'navarr'!

INTERPOLATE FLIGHTPATH

The unknown or inaccurate flightpath can be interpolated based on GCPs and a given altitude level.

Actions:

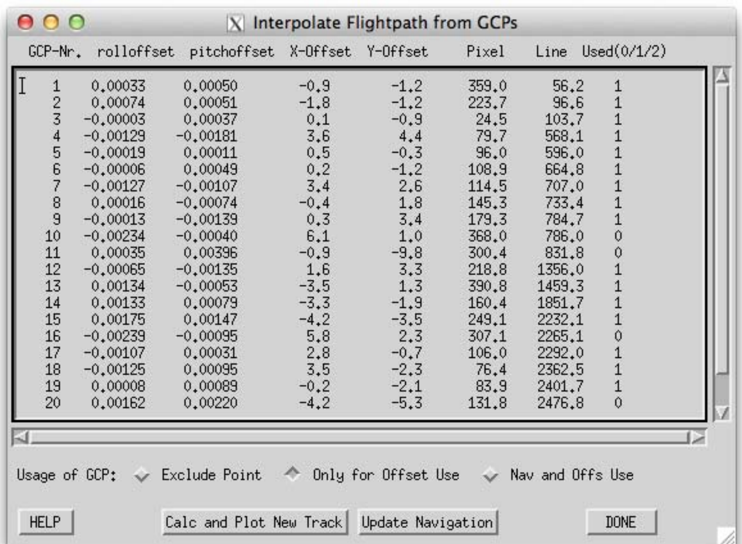


Figure 5.56: Interpolate Flightpath from GCPs

Inputs:

The input is a set of GCPs, given by the array 'gcparr'. The points used for the interpolation are marked by a flag value of 2 in the sixth column of the array.

How to change a point usage flag:

- A point from the list is selected by clicking it with the mouse,
- The desired usage can be selected:
  Only for offset: Point is used for offset calculations but not for flightpath reconstruction (flag=1)
  Nav and offs use: Point is used for flightpath reconstruction as well as for offset calculation (flag=2)

Actions:

Action:	Information:
Calc and Plot New Track	Calculation of the flightpath and a plot of the previous (dashed) and the new flightpath (line) against GCP positions (stars).
Update Navigation	The navigation data is updated to the currently calculated flight-path (the navigation data remains unchanged, if this button is not pressed).

Outputs:

The outputs are the updated flightpath (navarr) the updated GCP-List.

OPTIMIZE ATTITUDE WITH NAV.

The flightpath is interpolated iteratively, while roll and pitch offsets are adjusted. This step especially may be necessary over sloped terrain where heading and pitch offset can not be discriminated easily.

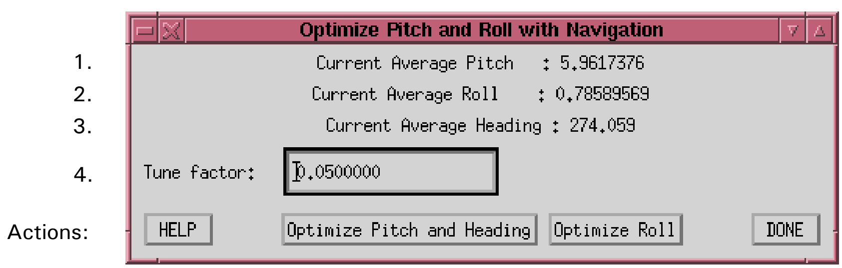


Figure 5.57: Optimize Pitch and Roll with Navigation

Inputs:

Nr.	Information:
1, 2, 3	Auxiliary data: initial navigation of roll, pitch and heading.
4	Tune factor: initial factor for the iterations (in radians for the angles); this value can be decreased to get higher accuracy of the results.

Actions:

Action:	Information:
Optimize Pitch and Heading	The pitch and the heading offset are optimized together with a contiguous recalculation of the flightpath.
Optimize Roll	The roll offset is optimized together with a contiguous recalculation of the flightpath.

Outputs:

The outputs are an optimized flightpath and new offsets for roll or pitch/heading.

PREPARE HEADING

This function calculates the true heading if only yaw data is available using flightpath navigation data; if no yaw data are available the heading is derived from the navigation data only.

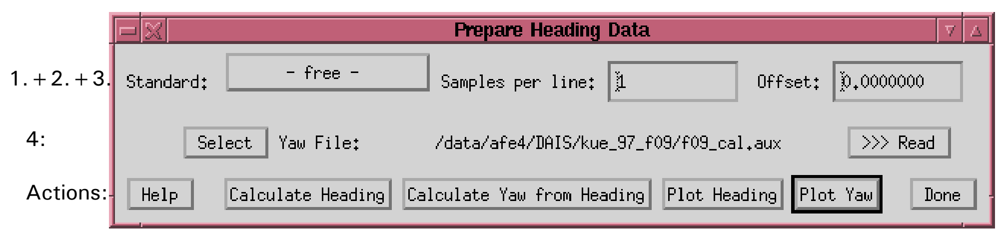


Figure 5.58: Prepare Heading Data

Inputs:

Nr.	Information:
1	Input file standard (format ’free’ can be used for non formatted binary or ascii data).
2	Number of samples per line.
3	Offset; can be updated by the ’return’ key.
4	File name selection for separate yaw data import

Actions:

Action:	Information:
Read	The yaw data is read to the yaw variable (yawarr).
Calculate Heading	The first derivative of the flightpath is joined with the yaw data to get a heading information. If no yaw is available, the heading can be estimated from the flighpath directly this way. If yaw seems to be erroneous *Reset Yaw* (Menu:Special) should be used to set yaw to zero.
Calculate Yaw from Heading	This is the inversion of the above procedure; can be used to check the heading (use 'Show Yaw').
Plot Heading	Plots the heading array in a separate window.
Plot Yaw	Plots the yaw array in a separate window.

Output:

The output will be a heading (or yaw) data per image line.

Hints:

- The PARGE calculation is based on the true heading of the airplane against direction north. Thus, the yaw data is only used for auxiliary information and does not have to be provided if heading is already available.
- The yaw parameter may be available in any format as given in the popup list to be read through this interface
- This functions is useful for resetting the heading parameter, be aware that first Yaw has to be reset to zero (by *Special:Reset:Yaw*) before the heading can be resetted using this function.

INTERPOLATE ROLL/PITCH

A number of GCPs with flag value 2 can be taken to interpolate the roll/pitch values for all lines. The interpolation results are added to the original roll/pitch values. Two interpolation options are supported:

- Smoothed Linear: Linear interpolation is performed over all image lines, using the offsets at the selected GCP s as Tie-points. The resulting curve is smoothed by 50 elements.
- Spline: A cubic spline is calculated through the selected GCP offsets.

Please use the reset function prior to interpolation if no original roll or pitch information had been available.

SCALE ROLL/PITCH

This function uses the GCPs with flag 2 to scale the selected parameter to the best fitting value. A simple linear scaling factor is derived iteratively from the ground control points offsets.

DRIFT CORRECTION

This function corrects drifts in the auxiliary data assuming a linear drift in dependence of the image lines. Drifts can be corrected for roll, pitch, x-coordinates, or y-coordinates. All GCPs having a flag of 1 or 2 are first used for offset determination. A linear drift model is then assumed to derive the drift parameters in relation to the image lines. The such derived drift is corrected while the parameter mean is kept constant.

RESET

Roll/pitch/yaw: the respective parameter is set to zero for all image lines.

For the heading paramter, the function **>Special:Prepare Heading<** [p.216] resets the heading to the flight direction (after setting yaw to zero).

For full reset of the PARGE session, please use the function **>File:Reset Session<** [p.150].

ROLL COMPENSATION

Roll distortions are often the most obvious effects in airborne image scanner data. This procedure allows a simple correction for such artifacts by shifting each line by an estimate of numbers of pixels, using the height above ground and the roll information from the image data or from INS measurements.

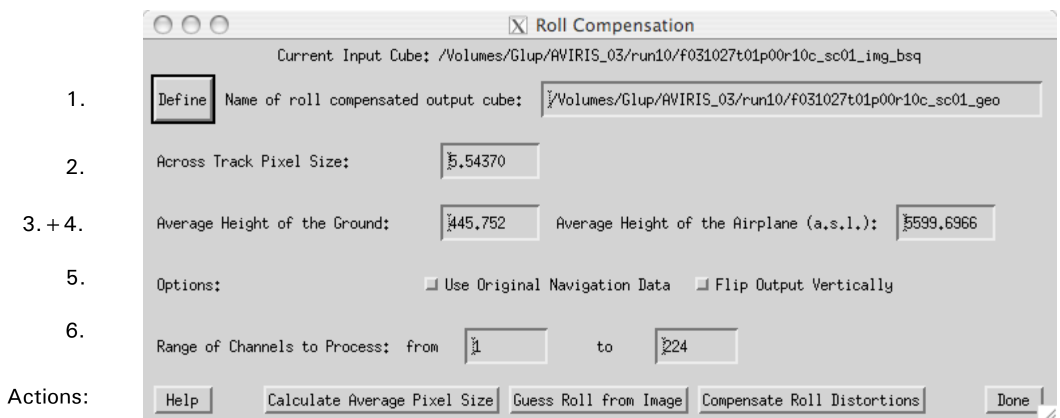


Figure 5.59: Roll Compensation

Inputs:

Nr.	Information:
1	Name of the roll-compensated output cube (cube file and ENVI header will be created at execution).
2	Across track pixel size in meters for average height above ground (not required for *Use Original Navigation Data* Option).
3	Average Height of the Ground: defaulted to average height of DEM.
4	Airplane Height: Average Height of the plane (taken from navigation data if available).
5	Options: 1) Use Original Navigation Data: uses the airplane height to calculate roll compensation more exactly. 2) Flip Output Vertically: check this box to flip output vertically.
6	Channel range that should be processed.

Actions:

Action:	Information:
Calculate Average Pixel Size	The pixel size is estimated from average altitudes and IFOV; this button can be used to recalculate pixel size after changing Average Height of Airplane or Ground.
Guess Roll from Image	Allows to calculate an estimate of the roll parameter from the apparent image distortions through line-to-line correlation analysis. ATTENTION: The roll-Parameter (rollarr) may be overwritten by this procedure.
Compensate Roll Distortions	A simple roll compensation is calculated and stored directly to file.

Outputs:

The output will be a roll compensated image cube.

Procedure:

The roll compensation pixel shift is calculated from the tan(roll) * height, divided by the cross track pixel size. The latter can be calculated directly from IFOV and average height above ground. Pixelsize and height is taken by line, if the *Use Original Navigation Data* button is selected.

ATTENTION: Roll compensated image cubes are not suited for any further geometric processing with PARGE.

POST-SYNCHRONIZE

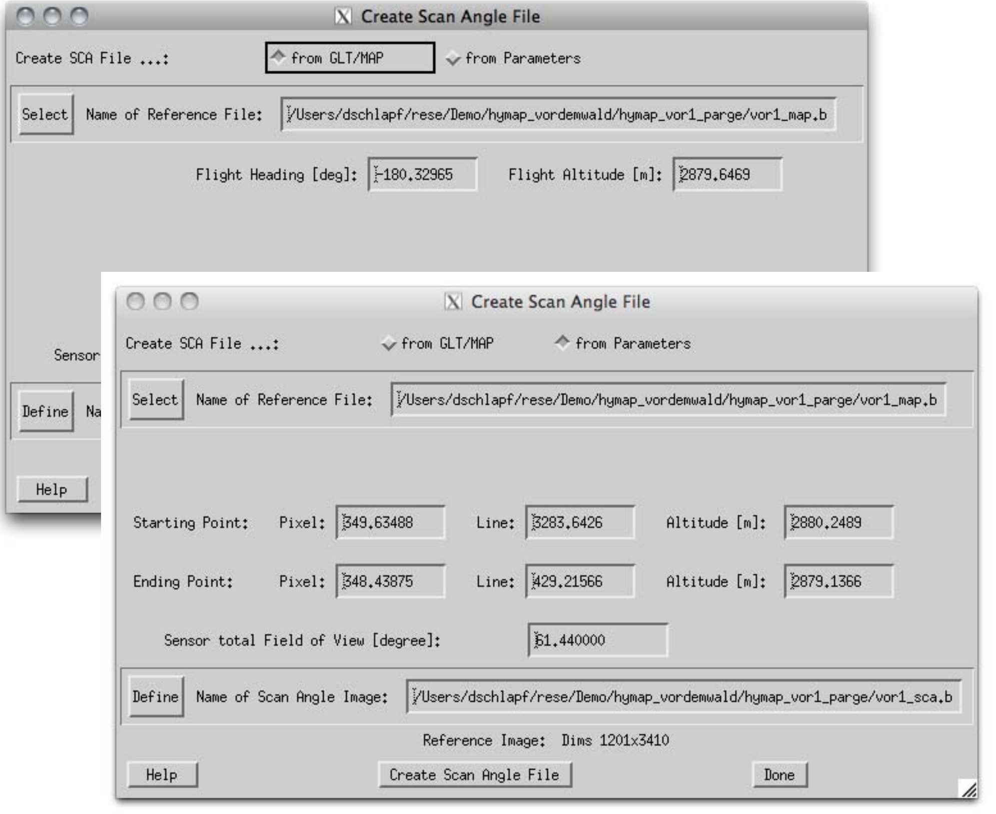
This function derives a synchronization offset from image- based roll and applies the offset to all parameters. The roll is determined by PARGEs built-in image auto-correlation algorithm, which determines a possible roll parameter by line by line correlation and subsequent filtering. (same function as in roll compensation). The thus derived roll is then fitted to the original roll parameter for determination of a potential synchronization offset. All parameters (attitude and GPS navigation) can then be shifted by the derived synchronization-offset value.

SHIFT PARAMETERS

This function shifts all auxiliary parameters (attitude/gps) by an arbitrary number of lines. Positive values are shifts in the positive time axis, while negative values are back in time. The shift is given by the fraction of number of lines. This function is useful if an unknown synchronization shift is observed and needs to be found.

CREATE SCAN ANGLES

This allows to calculate a scan angle file from GLT or simply from some parameters.



Inputs:

Nr.	Information:
	Select first if the scan angle file is to be created from MAP/GLT or from parameters
1	Reference file (ruling the output size, must be a valid MAP/GLT for the respective creation function)
2	Starting Point: first point of image in pixel coordinate of the reference file (lower left corner is (0/0)
3	Ending Point: last point of image in pixel coordinate of the reference file.

Nr.	Information:
4	Total FOV of sensor, used to determin absolute values of sensor zenith angle.
5	Scan angle file name to be created.

Actions:

Uses the MAP file and the FOV to calculate the view angles directly from the pixel positions (quite accurate..) or guesses the scan angles from the given parameters. For better coverage, a 10% margin is added in across track direction.

DISPLAY MAPPING ARRAY

In this function, the current mapping file is displayed as scaled image combination between the two mapping array layers. The display function overrides any negative values in the mapping array for display purpose, such that the filled pixels are displayed in the same manner as the originally referenced pixels.

DISPLAY DEM 3D

With this procedure the 3D surface of the DEM will be shown. There are several possibilities to customize the presentation by changing the plot parameters.

Attention: this may take some time to draw.

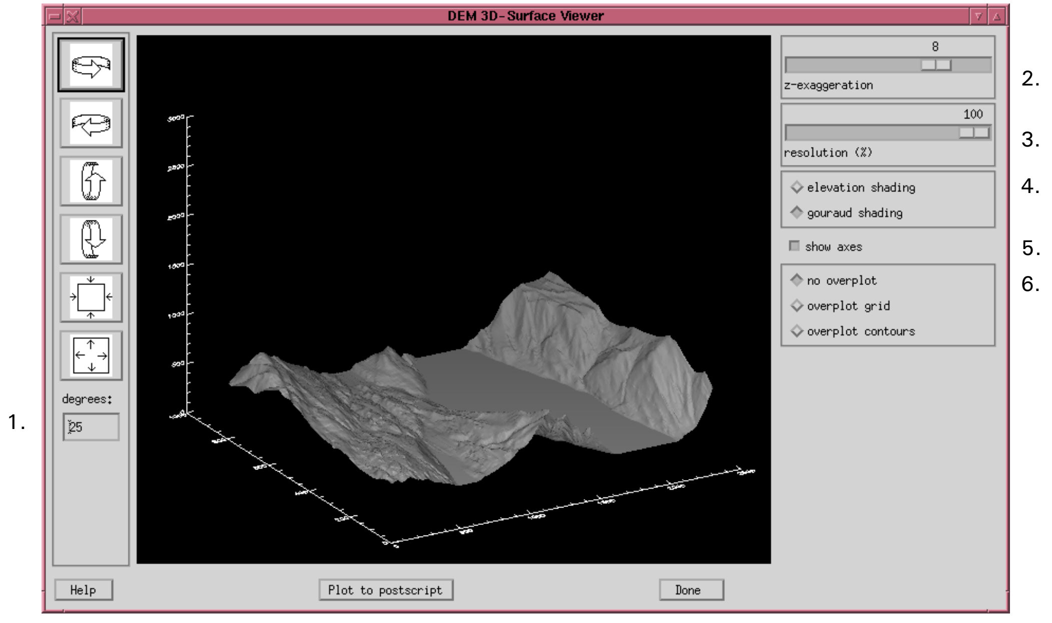


Figure 5.60: DEM 3D-Surface Viewer

Inputs:

Nr.	Information:
1	View angle (degrees).
2	Z-exaggeration.
3	Resolution (%).
4	Elevation shading.
5	Display of axes.
6	Overplot of grid or contours.

Actions:

Action:	Infomation:
Plot to Postscript	The 3D-image is saved to a postscript file (demsurf.ps).

Whenever a plot parameter is changed the presentation of the DEM is updated automatically.

Outputs:

The output will be a 3D-Display of the DEM.

Procedure:

The 3D-presentation can be customized using several parameters (see INPUTS); rotate (by 90 degrees) it vertically and horizontally using the first 4 buttons on the left side; zoom in and out using the last 2 buttons on the left.

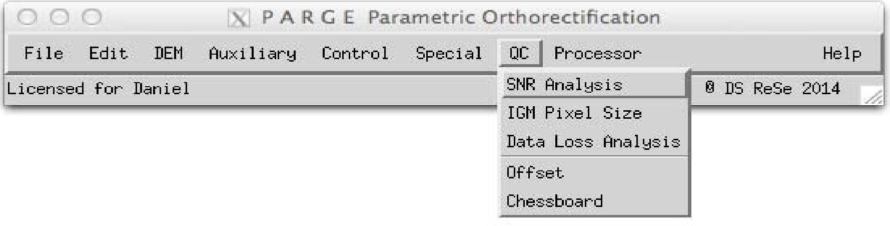
INTERRUPT

This function brings the user back to the IDL-prompt, while the variables still are active. Special changes to the parameters can be made using the IDL command line language. This function is meant for expert users, a description on how to access the functionality and data structure is given in Chapter 4.

To get back to the PARGE GUI, '.c' should be typed on the IDL/PARGE prompt.

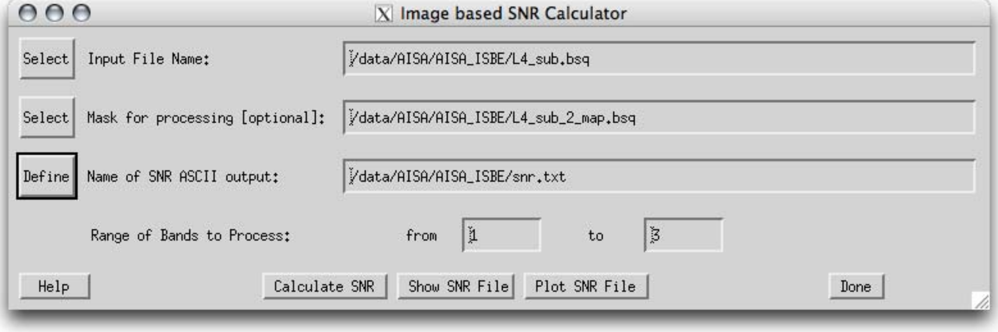
5.8 Menu: QC (Quality Control)

The quality control menu allows an interactive quality assessment of the imagery and the achieved results.



SNR ANALYSIS

Easy estimate calculation of SNR from image.



Inputs:

Nr.	Information:
1	Image data file for analysis
2	Mask file in same dimensions as image data.
3	Output ASCII file containing, Mean radiance, Noise equivalent delta radiance and SNR estimate for the data
4	Band range to be processed

Actions:

Action:	Infomation:
Calculate SNR	Calculates and writes the ASCII snr file.
Show SNR file	Displays the text file.
Plot SNR file	Creates a nice plot from the SNR file.

Procedure:

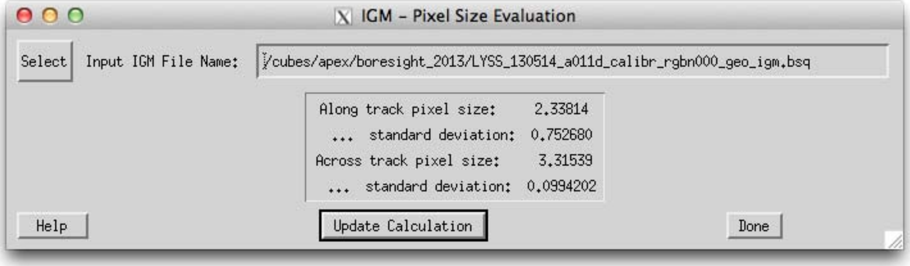
The algorithm applies a high pass filtering on a dark homogenous ROI and works as follows:

- search for the 7x7 pixels patch throughout the whole image in each band (=> ROI).
- calculate the mean of the whole image and the mean of the patch
- calculate the noise in the found patch after high pass filtering
- obtain SNR values as L_mean/Noise_roi or L_mean_roi/Noise_roi

NOTE: the obtained SNRs are a best-guess estimate and should not be taken as absolute quality measure for an imaging system.

IGM PIXEL SIZE

The optimal pixel size can be estimated by this tool from statistical analysis of the IGM file: The statistics of the IGM coordinate distances are calculated.

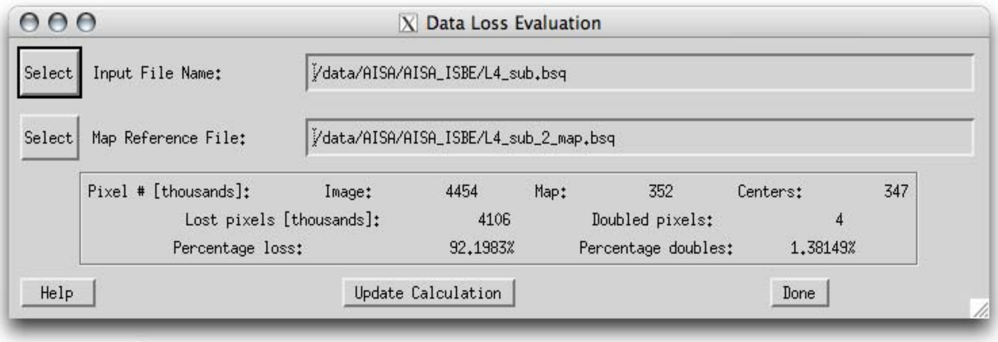


All results are in meters:

along track pixel size (mean of all distances)
.. standard deviation of along track pixel sizes
across track pixel size (mean of all distances)
.. standard devaition of across track pixel sizes

DATA LOSS ANALYSIS

Resampling results in data loss which can be calculated from the *map file by counting the doubled and lost pixels in relation to the total number pixels.



Inputs:

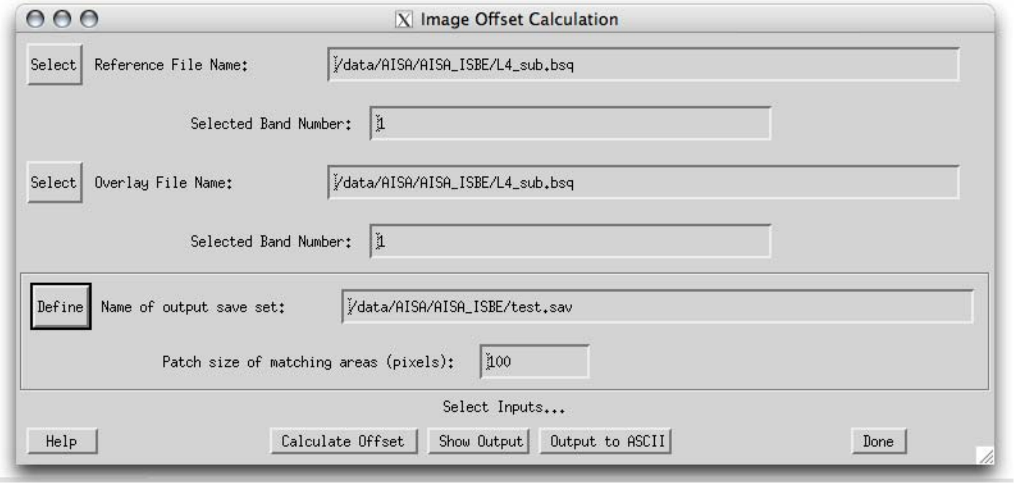
Nr.	Information:
1	Image data file for analysis
2	Map file corresponding to the image data.

Actions:

Action:	Infomation:
Update Calculation	Enters the values in the table on the basis of the map file.

OFFSET

Geometric offsets between two images are searched by correlation analysis.



Inputs:

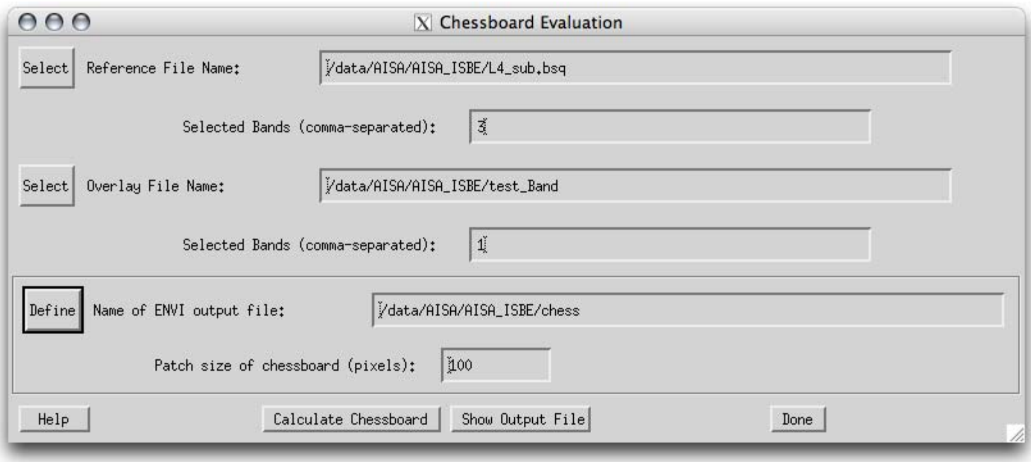
Nr.	Information:
1	Reference Image data file for analysis (e.g. topographic map) and band to be taken for the analysis from the image
2	Test image and the respective band number
3	output save set containing all calculation results (can be read from within IDL only)
4	patch size for systematic correlation analysis

Actions:

Action:	Infomation:
Calculate offset	goes through the whole image and gets the offset.
Show Output	creates a nice plot of the outputs.
Output to ASCII	writes the results to an human-readable file

CHESSBOARD

Visualization of the geometric differences by a chessboard image.



Inputs:

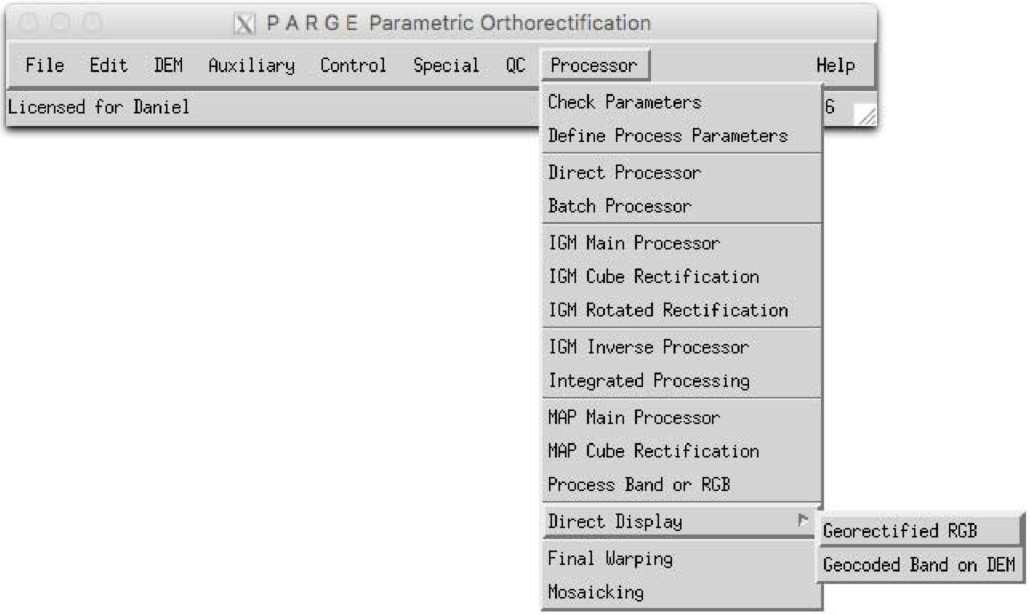
Nr.	Information:
1	Reference Image data file for analysis (e.g. topographic map) and band(s) to be taken for the analysis from the image
2	Test image and the respective band numbers
3	Output image file
4	patch size for the chessboard pattern

Actions:

Action:	Infomation:
Calculate chessboard	creates an ENVI file for display of the chessboard (scaled to byte data).
Show Output File	Displays the output.

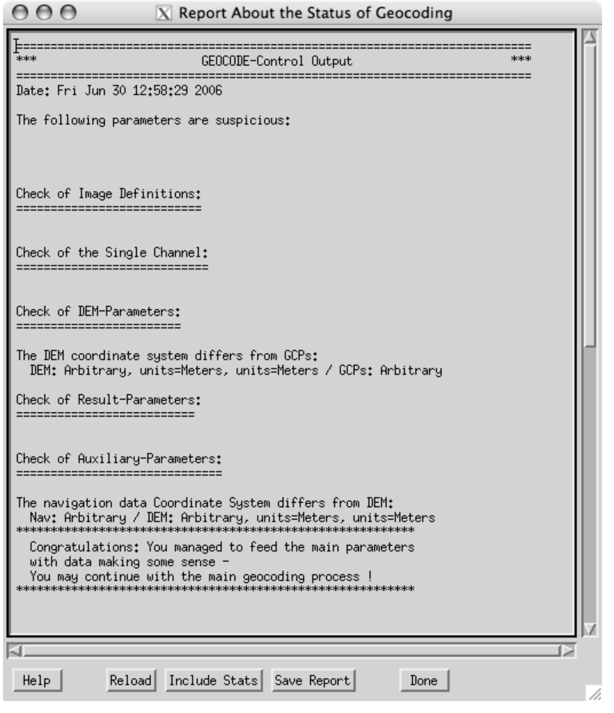
5.9 Menu: Processor

Two major steps are included in the processing part: first the main processor, which calculates the geometry for each pixel, and second the cube and RGB processor, which creates resampled imagery out of the calculated geometric information.



CHECK PARAMETERS

This report summarizes the main information on possible errors in the geocoding variables. The found problems are shortly described and should be solved by redifining the missing or wrong parameters. Its requirements have to be fulfilled to allow the time consuming processing of the final geocoding algorithm.



Information about errors is shown on the following topics:

1) Image data definitions
2) Single channel definitions
3) DEM data definitions
4) Auxiliary data definitions
5) Result definitions

Figure 5.61: Report about the Status of Geocoding

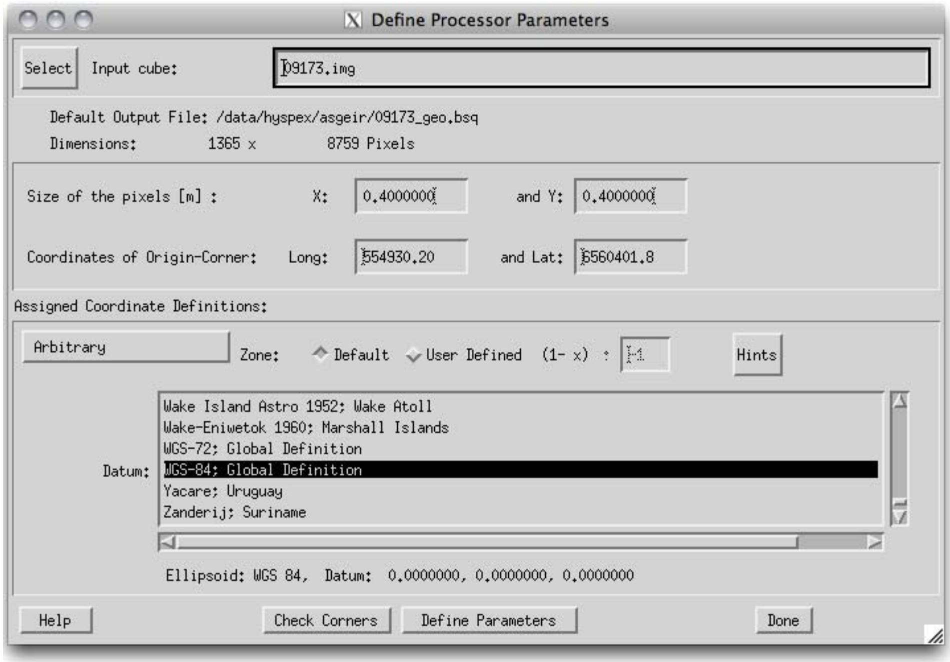
Actions:

Action:	**Information:**
Reload Report	Displays the current status information.
Save Report	Saves the current information as seen on the screen to a text file.

Note: Often, the coordinate system is unknown (i.e., 'Arbitrary') or not properly described from the various sources of the input data. This may result in a warning message after checking the parameters. This warning may be ignored as long as you are aware of the correct coordinate system for every available data source.

DEFINE PROCESS PARAMETERS

This function is used to set meta data information to be used for writting the correct result output headers and for selection of the input data cube to be processed.



Inputs:

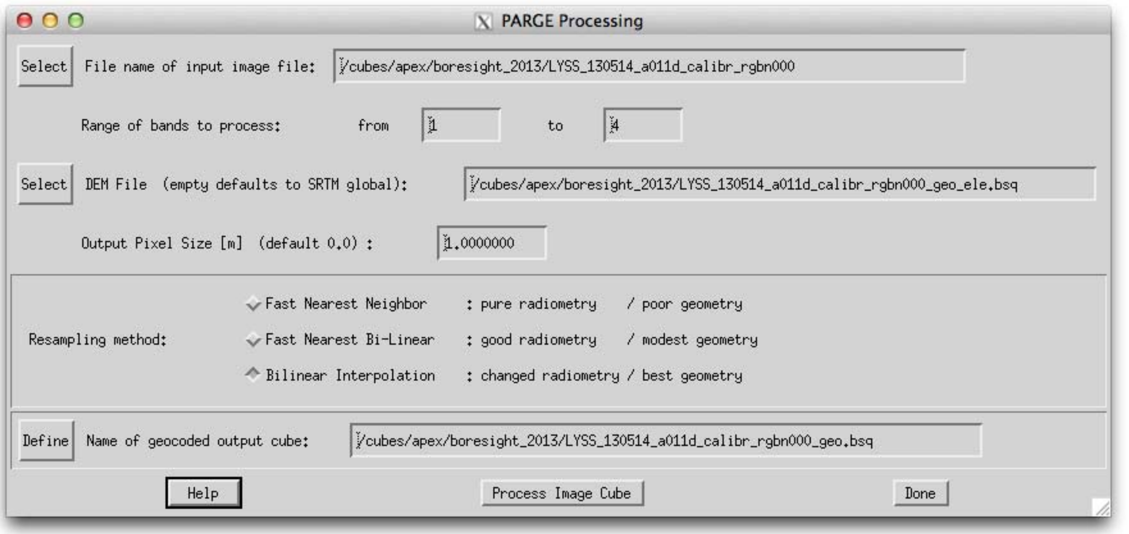
Nr.	Information:
1	Unrectified Image to be processed in the final processing
2	Default dimensions of result
3	Default coordinate systems of result

Actions:

Action:	Infomation:
Check Corners	displays the image corner coordinates of the outputs
Define Parameters	Changes the parameters accordingly.

DIRECT PROCESSOR

This Module is used to process an image directly after data import. It includes the handling of the DEM, the automatic size detection, the IGM calculation and the final cube processor and the quickview generation.



Inputs:

Nr.	Information:
1	Name of input image file - normally this is the input file as defined in the data import section. However, one may select a different input file of BSQ file format and with the same pixel dimensions as the original for the processing here.
2	Band range for cube processing: first and last band to process
3	DEM file: Name of the DEM file in ENVI format for the processing. If this field is empty or the file does not exist, the global elevation data from SRTM (in the /etc directory) is used for the processing.
4	Size of pixels: output pixel size - if value is 0.0 or below, the optimal output pixel size is estimated from the image parameters and rounded to 0.1 meters.

Nr.	Information:
5	Resampling Method: - Fast Nearest Neighbour: Fast standard way of resampling;does not introduce 'artificial' not measured spectra (highest spectral integrity) - Fast Nearest Bi-Linear: Combination of Nearest neighbour andinterpolated gaps. The gaps are interpolated in x and y direction only (no triangulation!), whereas the centers are preserved. - Bilinear Interpolation: Performs a full interpolation on the basis of an Image Geometry Mapp (IGM). This leads to highest geometric accuracy while loosing spectral integrity.
6	Output filename complete output cube (recommended: *_geo). This file may use lots of disk space (number of bands * size of DEM).

Actions:

>Process Image Cube<

A whole cube is created and stored in ENVI data format. This goes through the whole parge process from scratch.

Outputs:

- Pixel reference Map in target geometry (optional)
- IGM file
- elevation data resampled to image dimensions
- Geocoded ENVI file and header.
- log file

Procedure:

Uses the current status after data import to go through a streamlined processing. Some parameters are taken from defaults and from the PARGE preferences (e.g., number of tiles and default RGB bands).

For more options of the processing, it is recommended to use the IGM processing functions.

Attention: outputs are overwritten automatically if already existing at the given location.

BATCH PROCESSOR

This module is used to start a series of batch processes from text configuration files. The batch configuration files have to be prepared beforehand. A good starting point is to export these files using the function **File:Export:Batch Process File** p.149.

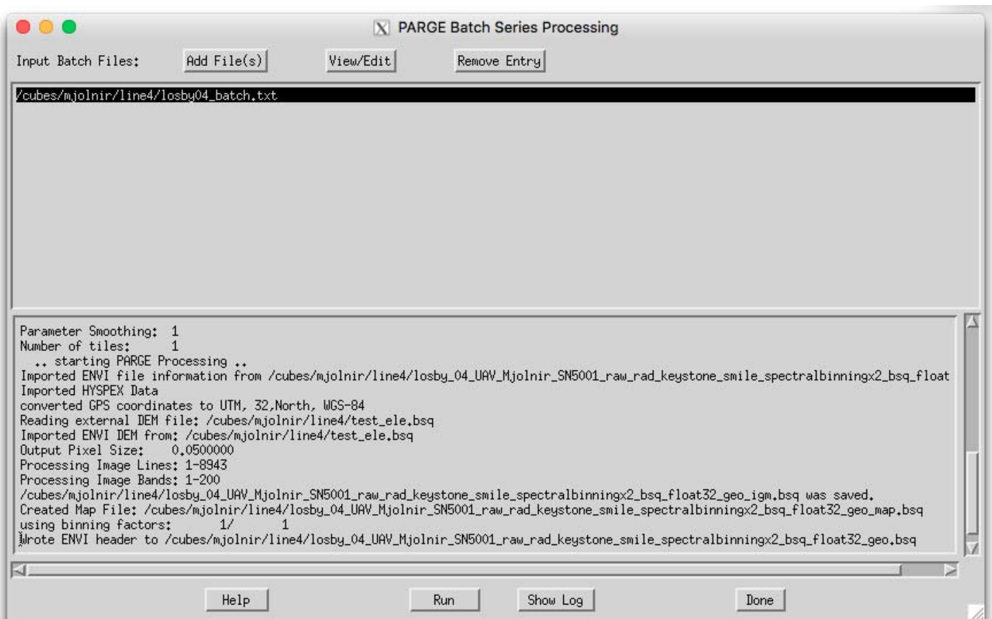


Figure 5.62: Batch Processor panel.

Inputs

input list of processing files, use the 'add/remove' button to update the list. All files will be processed sequentially when starting.

Console:

See what's happening - the console may also be used for logging any other parge process.

Actions:

>Run<

starts a process and runs all the batch files as of the list.

>Show Log<

displays the log of the currently selected processing file.

IGM MAIN PROCESSOR

This function is used to calculate the image geometry map (IGM) directly from raw data.The function does NOT create a PARGE *_map, whereas the scan angle file is created in raw scanning geometry rather than the target geometry.

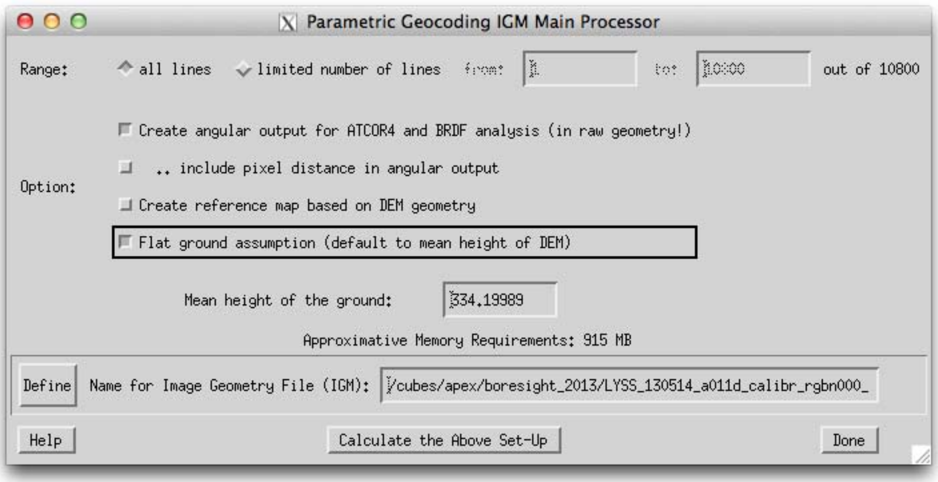


Figure 5.63: IGM processor window.

Inputs:

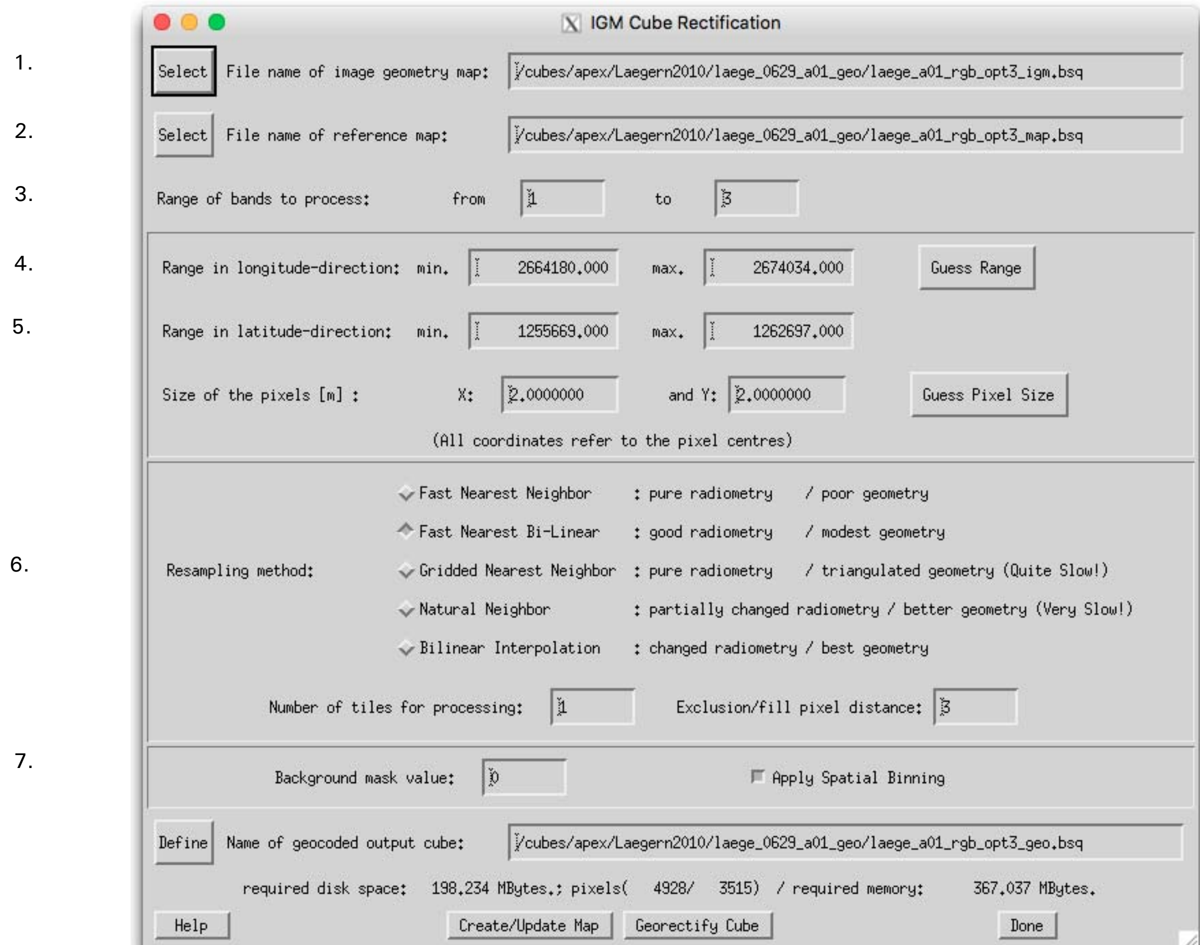
Nr.	Information:
1	range of lines/rows to be processed (starting at '1')
2	Options: A scan angle file shall be produced in raw scan geometry. This file will have same name as the IGM with *_sca appended. The pixel distance may be optionally included in scan angle file (3rd layer) For subsequent fast nearest neighbour resampling, creating the reference *.map file is recommended. Flat ground processing allows a fast process without having a DEM available.
3	Mean heigt of ground: if no DEM is available, this height is used for flat ground processing.
4	Name for Image Geomtry File: IGM filename of output.

Outputs:

- Image geometry map (*igm)
- Optional: Scan angle file (*sca) in raw geometry, Reference Map file (*map)

IGM CUBE RECTIFICATION

This function allows to process imagery to an arbitrary resolution using the image geometry map (IGM) which stores the coordinates for every pixel.



Inputs:

- IGM filename for bilinear interpolation options (*_igm).
- MAP file for nearest neighbour resampling only (*_map)
- Band range for cube processing (starting at ‘1’)
- Target image geometry, extent dimensions and ground sampling resolution the output may be entered in arbitrary values.

- ‘Guess’ button to find the extent of the image
- Resampling Method - nearest neighbour options, bilinear interpolation or combined processing. For combined resampling a map file is to be created (on the fly if required).
- Number of tiles for processing to reduce the memory requirements during resampling and triangulation.
- The exclusion/fill distance rules the extent of interpolation/extrapolation. If unwanted gaps occur in the image, the distance shall be increased.
- ‘Apply spatial binning’ option to bin the data automatically in the spatial dimension in case the pixel size is larger than two pixel IFOV.
- Output filename for complete output cube (recommended: *_geo)

Actions:

Action:	Information:
Guess Range	This function guesses the spatial limits of the output based on image borders and flightline GPS.
Guess Pixel Size	This function guesses a pixel size from IFOV and flight altitude.
Create/Update MAP	Creates a map reference file based on the given parameters (in arbitrary resolution/extent!)

Resampling Options

- Fast Nearest Neighbour: leaves radiometric values in place, bears the problem of data loss and changed image statistics, uses the *map for a fast process.
- Fast Nearest Bi-Linear: leaves radiometric values in place at the center pixels and interpolates the gaps with a simple bilinear approach.
- Nearest Neighbour: leaves radiometric values in place, bears the problem of data loss and changed image statistics.
- Natural Neighbour: Adaption of the nearest neighbour approach which performs a weighting of the individual pixels by their relative area (not suited for large datasets
- Bilinear Interpolation: Allows highest geometric accuracy but smoothes the radiometric values.

(More details about resampling can be found in Section 2.3 on page 25 of this manual)

Outputs:

A geocoded ENVI file with its header is created containing the new geometric reference information - note that this output is NOT compatible to the original DEM used for processing. In addition, a scan angle file is created automatically as long as _sca file in raw geometry had been calculated in the IGM main processor.

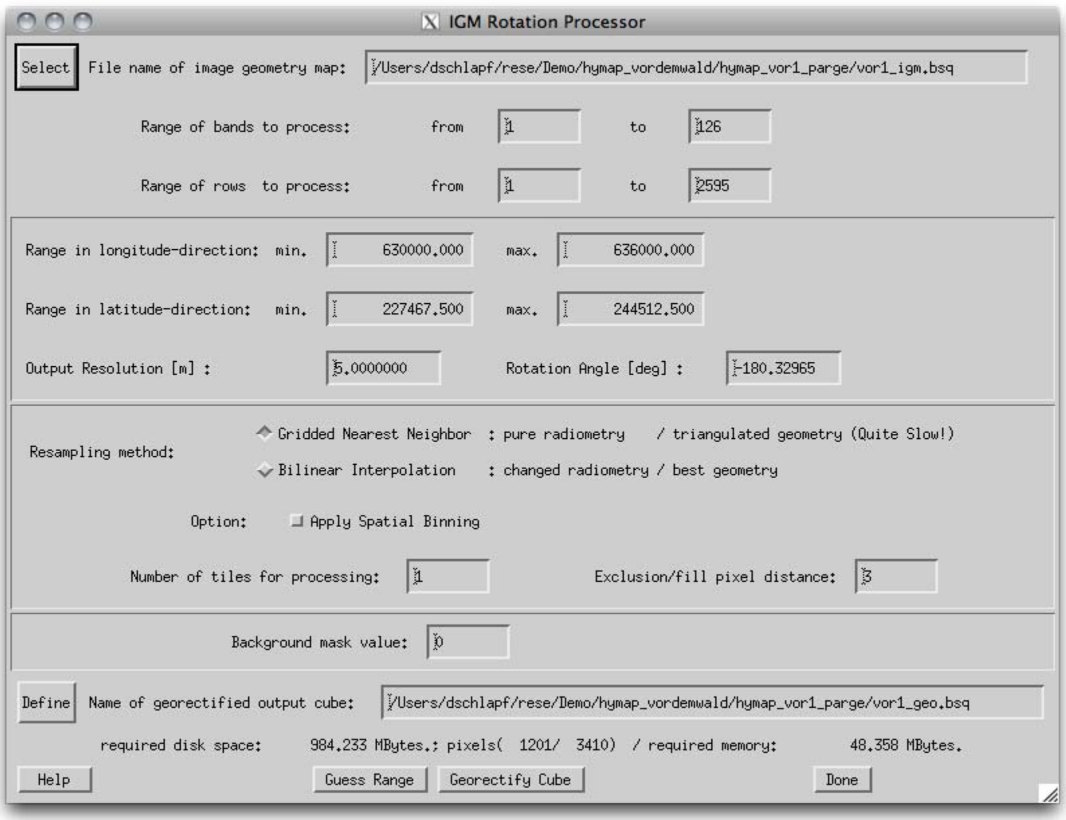
Procedure

Use a previously created IGM array ('IGM main Processor') to geocode the whole cube and interpolate and write the result - this may use quite some disk space and RAM.

IGM ROTATED RECTIFICATION

This function allows to process imagery to an arbitrary resolution and rotation using the image geometry map (IGM) which stores the coordinates for every pixel.

1.
2.
3.
4.
5.
6.
7.



Inputs:

Nr.	Information:
1	IGM filename (*_igm.bsq).
2	range of bands to process (starting at 1)
3	range of rows/lines to process (starting at 1)
4	range in non-rotated image to be processed
5	output resolution in meters and rotation angle (degrees) to north, east positive

Nr.	Information:
6	Resampling Method - nearest neighbour options or bilinear interpolation, includes option for automatic spatial binning of oversampled data.
7	Number of tiles for processing to reduce the memory requirements during resampling and triangulation. The exclusion/fill distance rules the extent of interpolation/extrapolation. If unwanted gaps occur in the image, the distance shall be increased.
8	Output file name definition and estimated processing memory requirements.

Actions:

Action:	Information:
Guess Range	This function guesses the spatial limits of the output.
Geocode Cube	creates a georeferenced rotated image cube.

Resampling Options

- Fast Nearest Neighbour: leaves radiometric values in place, bears the problem of data loss and changed image statistics, uses the *map for a fast process.
- Fast Nearest Bi-Linear: leaves radiometric values in place at the center pixels and interpolates the gaps with a simple bilinear approach.
- Nearest Neighbour: leaves radiometric values in place, bears the problem of data loss and changed image statistics.
- Natural Neighbour: Adaption of the nearest neighbour approach which performs a weighting of the individual pixels by their relative area (not suited for large datasets
- Bilinear Interpolation: Allows highest geometric accuracy but smoothes the radiometric values.

(More details about resampling can be found in Section 2.3 on page 25 of this manual)

Outputs:

A geocoded ENVI file with its header is created containing the new geometric reference information - note that this output is NOT compatible to the original DEM used for processing.

Procedure

Use a previously created IGM array ('IGM main Processor') and a rotation angle to geocode the whole cube and interpolate and write the result - this may use quite some disc space and RAM.

IGM INVERSE PROCESSOR

This function allows to invert imagery from geocoded map geometry to raw scan geometry. using the image geometry map (IGM) which stores the coordinates for every pixel. This function is useful if radiometric processing shall be performed in raw geometry.

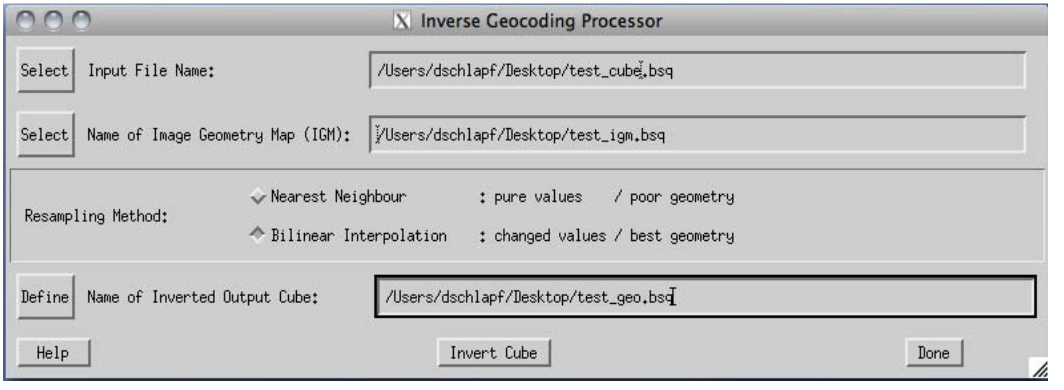


Figure 5.64: Inverse processing tool.

Inputs:

Nr.	Information:
1	Input file in map/dem geometry.
2	IGM filename for interpolation options (*_igm).
3	Resampling Method - nearest neighbour options or bilinear interpolation.
4	Output filename for inverted input cube.

Resampling Options

- Nearest Neighbour: leaves radiometric values in place, bears the problem of data loss and changed image statistics, well suited e.g., for aspect angle layers or shadow masks (with discontinuities).
- bilinear interpolation: well suited resampling for resampling of continuous data such as a DEM.

INTEGRATED PROCESSING

This routine performs the final rectification process for two data sets simultanously in order to create an integrated output product.

Inputs

The inputs for both files to be merged are given as follows:

- Name of status files: enter the names of valid status files of data after having performed the main processing. The status has to be saved after the IGM is created.
  Note: the first status serves as the master for the processing. Names of input cubes and igm files are read from the status file.

- Selection of resampling method:
  - Fast Nearest Bi-Linear: Combination of Nearest neighbour and interpolated gaps. The gaps are interpolated in x and y direction only (no triangulation!), whereas the centerare preserved.
  - Bilinear Interpolation: Performs a full interpolation on the basis of an Image Geometry Mapp (IGM). This leads to highest geometric accuracy while loosing spectral integrity.

- Range of output image in meters for longitude and latitude direction and pixel size in x/y.

- Output size of pixels, i.e. the target resolution (same size in x and y direction)

- Number of Tiles for processing: For reduction of the memory requirements, the triangulation can be calculated in tiles (tiled in along track dimension), increasing the number of tiles will reduce the processing speed.

- Exclusion /fill pixel distance: this is the maximum size of triangles to be used for interpolation larger triangles or distances are omitted and filled with the background value.

- Processing Options:
  - Mosaic: creates a mosaic from scratch from the IGM files using the selected resolution/limits. Mosaicing is only supported for files having the same number of bands.
  - Join: the two files are joined into one file 'as is', i.e. the bands of the second files are attached after the bands of the second file.
  - Join and clip: the two files are joined while only the common area is kept in the file; all other data is masked in order to have consistent spectra.

- Output filename of complete output cube; this is also the base name for the optional scan angle output.

Actions

> *Guess Range* < allows to guess the range of the merged imagery (button beneath the range fields) - the maximum extent of both IGM files is taken as the default.

> *Join Cubes* < a new integrated cube is created and stored in ENVI data format from the two input data cubes as specified in the status files.

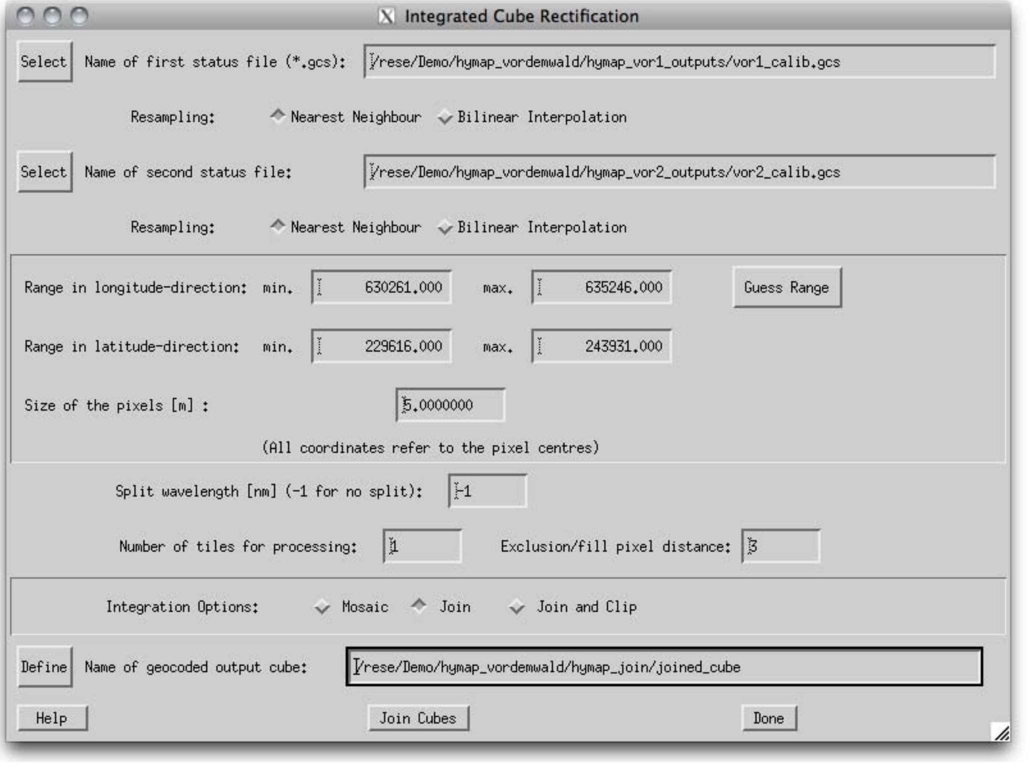


Figure 5.65: Integrated processing panel.

Outputs:

- Integrated Geocoded ENVI file and header
- as a side output, the rectified (and mosaiced) scan angle image fitting to the image output is created (if a scan angle file in raw geometry is available)

- the status files are updated to reflect the changes of the result.

Procedure

The routine will bin the original data to the selected resolution by default before applying the resampling method.

The scan angle file is processed automatically if available in rawformat.

The status files as selected are updated with the new parameters.

The rotuine uses the previously created IGM arrays ('IGM Main Processor') to geocode the whole cube and write the result - this may use quite some disc space. This output is independent of the original DEM geometry.

MAP MAIN PROCESSOR

This process performs the main geocoding. It stores the relative image coordinates to absolute DEM positions and fills the gaps in the final image mapping array.
Optionally, scan angles per image pixels are stored to a separate image file for use in atmospheric correction and BRDF analysis and an ENVI *.igm file can be created to store the explicite pixel coordinates.

1.
2.
3.
4.
5.
6.
Actions:

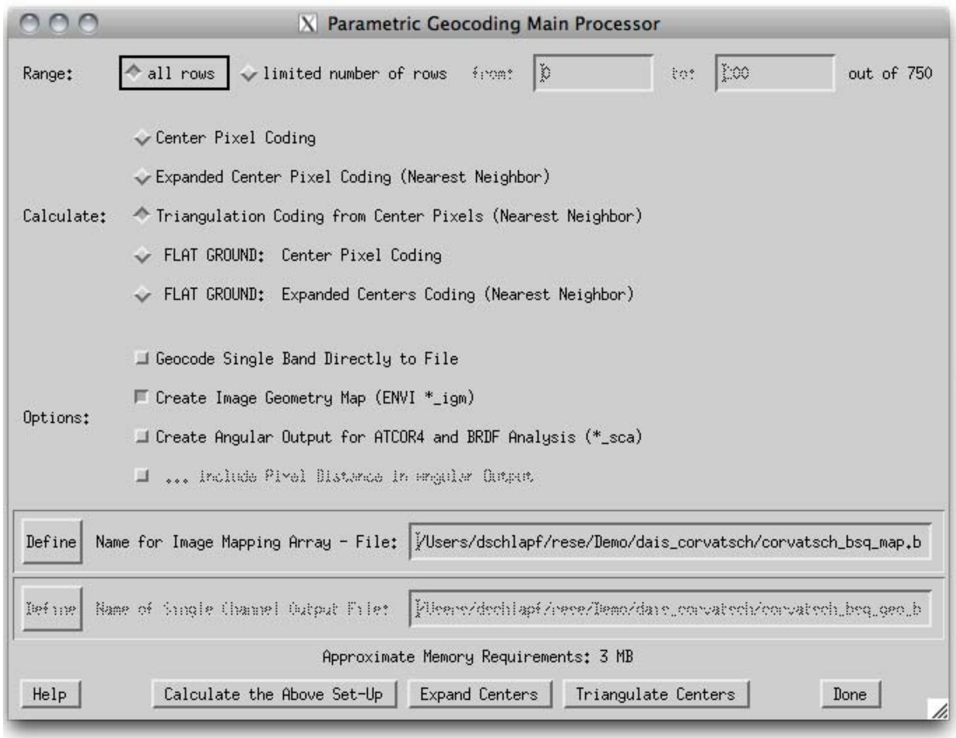


Figure 5.66: Parametric Geocoding Main Processor

Inputs:

Nr.	Information:
1	Number of lines to be processed (starting at 1). Recommended for test runs.
2	Types of map file resampling in view of nearest neighbour processing (see below).
3	Processing options (see below).
4	Output filename to store image mapping array (positions of the raw image pixels).

Nr.	Information:
5	Name of single channel file (required for direct coding to single channel).
6	Depending on the selected options an estimate on the required processing memory (RAM) is given.

Types of geocoding/gap filling:

- Center Pixel Coding:
  Only the centers of the pixels are mapped to the DEM-geometry. Choose this option, if subsequent interpolation is applied in the final processing step.
- Expanded Pixel Coding:
  The center coordinates are expanded, using a nearest neighbor method (function 'grow_pix') to maximum +/- 3 Pixels from center.
- Triangulation Coding:
  The gaps between the center pixels are filled, using a triangulation method - takes much longer than expanded coding, but is more accurate; memory requirements limit the applicability of such processing for large data sets.
- FLAT GROUND: Center Pixel Coding:
  The centers are geocoded directly to flat terrain. The height of the terrain and the aircraft are taken from the average of the DEM and the navigation data heights, respectively.
- FLAT GROUND: Expanded Centers Coding:
  Same as above, but with center expansion.

Options:

- Geocode Single Channel Directly to File:
  This option allows a direct-to-disc correction of one single channel (the currently selected band). Use this options e.g. for panchromatic imagery. No mapping array is created and no interpolation may be applied, what makes this kind of calculation very slow if more than 2 channels are geocoded.

- Create Image Geometry Map (ENVI *_igm)
  This options creates an ENVI-style IGM file, containing the absolute x and y coordinates for every pixel of the raw imagery. The igm is created in addition to the mapping file. It may be used from within ENVI for creation of geocoded results and has the spatial dimensions of the raw image.

- Create Angular Output for ATCOR4 and BRDF Analysis:
  PARGE may produce an angular output file (_sca), where the angles for each pixel are stored for optional further use by an atmospheric correction procedure (such as ATCOR4) or for BRDF correction algorithms. Check this box if you need such an

output.

- ... include Pixel Distance in Angular Output:
  adds the distance between aircraft and pixel as fourth layer in the scan angle output file. (this function is optional as the output allows to invert the DEM information).

Actions:

Action:	**Information:**
Calculate the Above Set Up	The image mapping-array is created or a single channel is geocoded directly.
Expand Centers	Expanding of center pixels: after a center pixel array was already defined. The nearest neighbours of the current center pixels are filled and saved to the mapping array..
Triangulate Centers	Calculates triangulation directly on center pixels. *Attention:* this operation is memory-intesive! It may take up to hours of calculation time depending on DEM size. Expansion and triangulation can also be performed after center pixel coding using these buttons.

Outputs:

1) Image mapping file:
This re-mapping file contains the indizes saying which pixel has to be transferred to what location on the DEM. The final processor can then be used to transfer whole image cubes from one geometry to the corrected coordinates, using this mapping-array.
Possibly two image mapping are created: one for center coordinates (extension '_c') and a second for the interpolated coordinates. The center coordinates array can be interpolated separately afterwards.

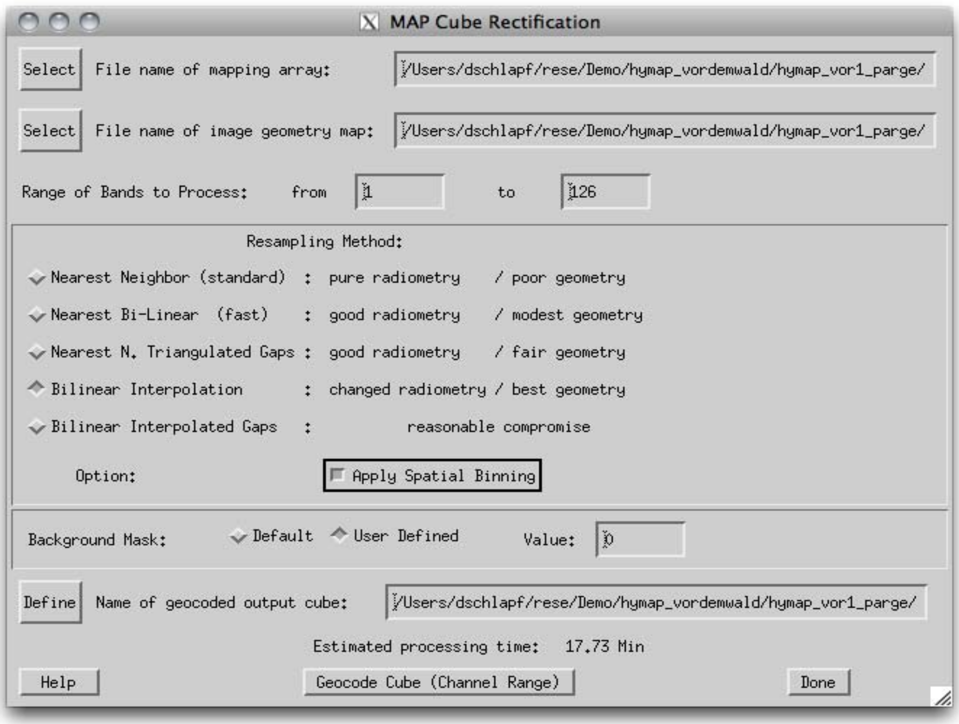
2) Geocoded single channel (Option):
if that option is selected, no image mapping array is created, but one single channel (the first of the input file) is geocoded directly to an output file in ENVI data format.

3) Side outputs:
Scan Angle file *_sca or Image Geometry Map *_igm, as described in the options section above.

MAP CUBE RECTIFICATION

This final geocoding step uses a previously calculated image mapping- file to create images of the final geometry. It is separated from the main processor for an independent use. This separation is useful for imaging spectroscopy applications if e.g. the results are geocoded instead of the raw imagery.

1.
2.
3.
4.
5.
6.
Actions:



Inputs:

Nr.	Information:
1	Name of the used image mapping array (usually *_map). This mapping file should be compatible to the selected cube dimensions and the DEM.
2	IGM filename for bilinear interpolation options (*_igm)
3	Channel range for cube processing (starting at ‘1’)

Nr.	Information:
4	Resampling Method (see below); includes option for automatic spatial binning of oversampled data.
5	Optional value for the uncovered background pixels. An arbitrary value appears in the background if no specific masking value is given by this function.
6	Output filename for complete output cube (recommended: *_geo)

Resampling Options

- Default/Nearest Neighbour: fast and convenient way of geocoding; does not introduce 'artificial' not measured spectra.
- Nearest Bilinear: combines two linear interpolations in x and y direction with nearest Neighbor resampling. The nearest linear interpolation is then selected of the two expressive directions.
- Triangulated Gaps: Interpolation of the gaps between center pixels (this requires extensive memory amounts up to about 10 times size of DEM).
- Bilinear Interpolation: Performs a full interpolation on the basis of an Image Geometry Mapp (IGM). Attention: High memory requirements!). This leads to highest geometric accuracy while loosing radiometric accuracy.
- Bilinear Interpolated Gaps: Performs a full interpolation on the basis of an Image Geometry Mapp (IGM).
  Attention: High memory requirements! Center pixels are replaced by the original measurements instead of doing an interpolation.

(More details about resampling can be found in Section 2.3 on page 25 of this manual)

Actions:

On execution, a whole cube is created and stored in ENVI data format. If an interpolation option is selected, each channel is interpolated separately to the output grid.

Outputs:

A geocoded ENVI file with its header is created containing the same geometric reference information as the currently selected DEM.

Procedure

Use a previously created mapping array ('MAIN Processor') to geocode the whole cube and interpolate and write the result - this may use quite some disc space.

ATTENTION: If the interpolation procedure fails due to memory allocation problems, use another more simple interpolation methods or assign more memory to your IDL session.

PROCESS BAND OR RGB

This final geocoding step uses a previously calculated image mapping- file to create TIFF , JPEG or ENVI images of the final geometry. Use this procedure for quality assessment and for the display of geocoding results.

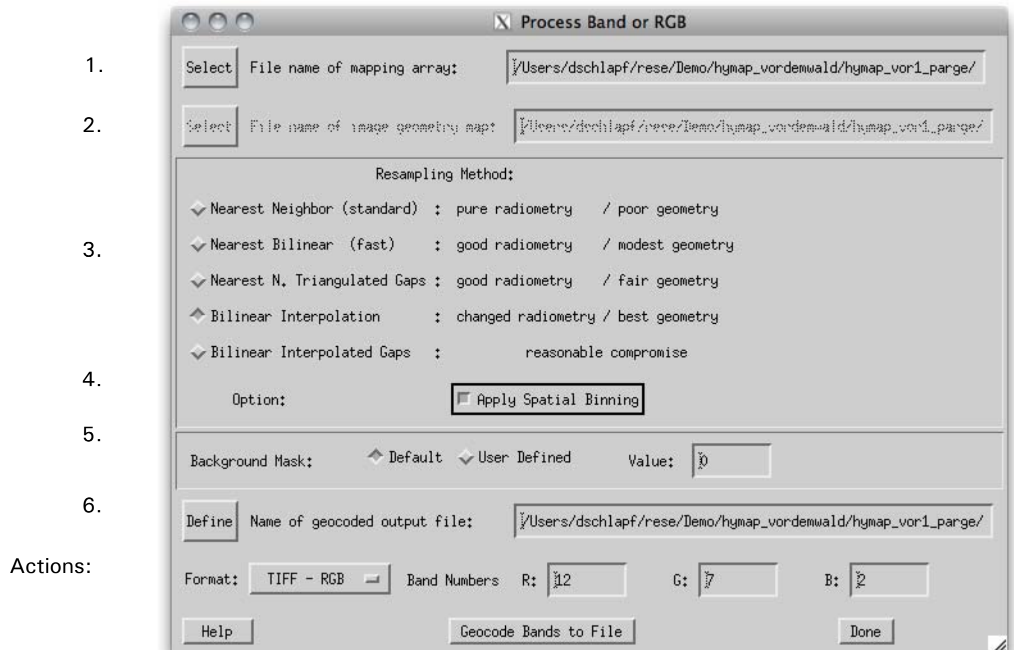


Figure 5.67: Geocoding Single Band and RGB Processor.

Inputs:

Nr.	Information:
1	Name of the image mapping array (usually *_map)
2	IGM filename for bilinear interpolation options (*_igm)
3	Resampling Method: compare comments in function: *Processor:Final Cube Processor* includes option for automatic spatial binning of oversampled data.
4	Optional Value for the background pixels. An abitrary value appears in the background if no specific masking value is given by this function.
5	Output filename for RGB (without extension - it will be added automatically for TIFF and ENVI files) or single band image file.

Nr.	Information:
6	Output file format (TIFF/JPEG/ENVI) and number of bands (1 or 3) for processing. The channel number shall be given as Red /Green Blue for the processing. For single band images, only the 'Red' channel number has to be given. The bands are numbered starting at '1'.

Actions:

Action:	Information:
Geocode Bands to File	An RGB or single band image is written for later use in standard image processing software. The appropriate extensions are written for JPEG and TIFF, and for the ENVI-formatted files, an ENVI header is created.

Outputs:

The output will be :

- RGB-tiff, RGB-jpeg, or three-band ENVI file, or
- single band -tiff, jpeg, or ENVI file.

Procedure

A temporary file is created first as would be done by regular processing. I then is converted to one of the supported image formats.

Attention (1): If this procedure fails due to memory allocation problems, use other interpolation method or assign more memory to your IDL session.
Attention (2): This procedure is only feasible if the resampled geometry has been created but not from IGM directly'; i.e. please create a *_map file first before using this function (available in the function **IGM Cube Rectification** [p.238].

DIRECT DISPLAY

This is a fast method for geocoded results. It reads the data from disk, geocodes it using the last created image mapping array and displays the result directly on screen.
There are three images that can be displayed directly:

Georectified RGB

One or three channels are georectified using the fast nearest neighbour approach and displayed directly on screen. The standard band selection allows to select the input bands. This routine is meant for a fast preview. As a side effect, an ENVI file (named rgb_x_y_z.bsq) is written automatically at the default output location containing the bands as selected.

Geocoded Band on DEM

The geocoded band is wrapped over a DEM. This procedure takes a lot of time and may be easier done from within ENVI.

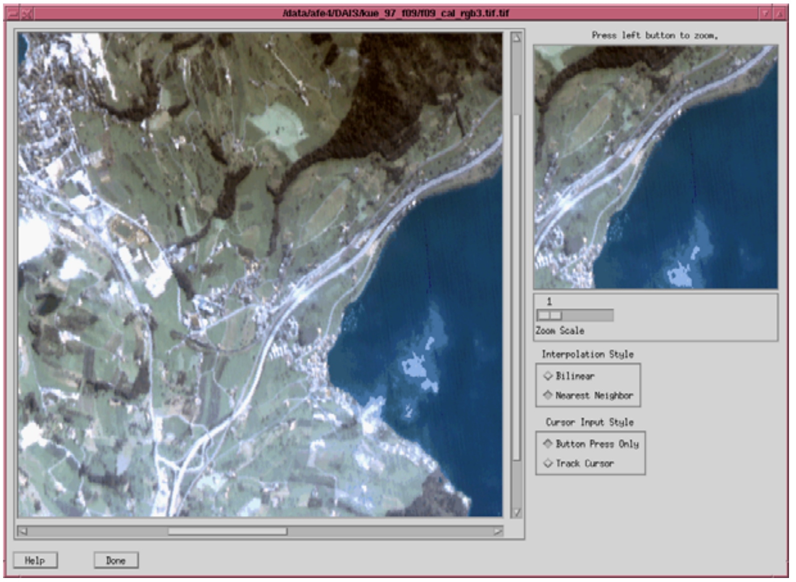


Figure 5.68: Display Geocoded RGB

Details:

This routine uses nearest neighbour coding only as stored in the mapping file. No interpolation is done on the displayed data.

FINAL WARPING

Sensor systems built upon multiple detectors sometimes are affected by coregistration offsets between detectors in subpixel range up to 2 pixels. The final warping routine will warp the second part (typically SWIR) of an output file onto the first detector (VNIR) by correlation analysis.

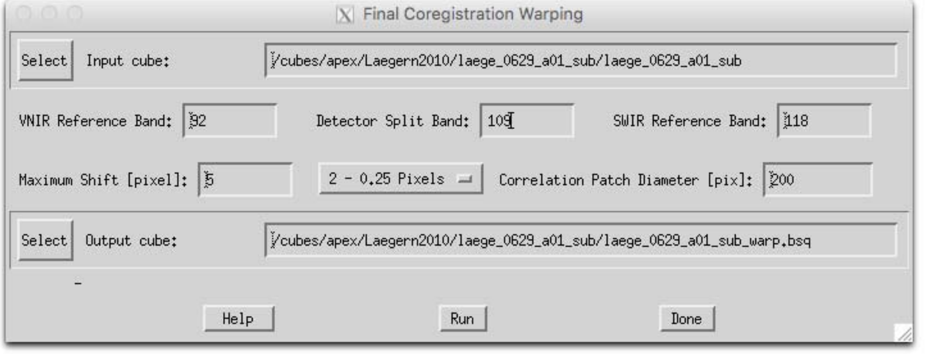


Figure 5.69: Final coregistration warping routine

Inputs:

Nr	Information:
1	Input cube: Joined Image cube, i.e. a cube, consisting of two spectral parts. The first part (usually the VNIR part) serves as a reference whereas the second part (usually the SWIR) is shifted by some pixels.
2	VNIR Reference Band: Band to be used as reference for correlation analysis (normally, a NIR band is recommended)
	Detector Split Band: First band of second detector (i.e., the first band to be warped)
	SWIR Reference Band: Band of second detector for correlation analysis, a band at 1050 is recommended for best correlation to a NIR band
3	Maximum Shift [pixel]: Maximum expected shift between the two parts of the cube
	Pixel Accuracy: Accuracy, to which the offsets should be calculated.

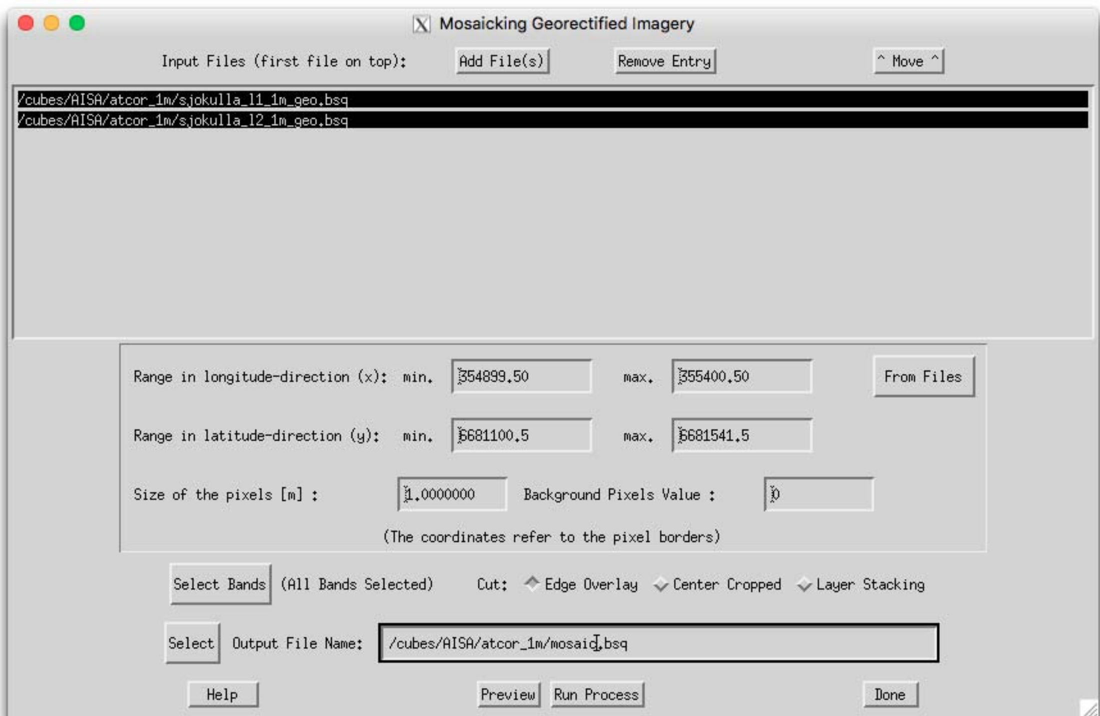
Nr	Information:
	Correlation Patch Diameter: Size of image patches for correlation analysis (in pixels)
4	Output: Ouptut file to be created.

Procedure:

The VNIR and the SWIR bands are analyzed first against each other to find average offsets. Afterwards, a transfer function is found by triangulation to transform the whole image to new position by bilinear interpolation.
Note: this function potentially decreases the sharpness of the second part of the image cube due to the interpolation.

MOSAICKING

Finally, the data may be mosaicked.



Inputs

Nr.	Information:
1	Input Files (first file on top): list of files to be mosaiced. The files are stacked in the order of appearance (i.e the first file in the list is on top)
2	File list buttons - Add: adds one or more new files - Remove: removes the selected file(s) - Move: moves the file one position up (or rotates if already at top)
3	Range range in x and y direction to edge of pixels of the mosaiced product should be entered. The coordinates refer to the pixel borders (edges) on either side of the image. Button: >From Files<: reads the maximum extent of all selected files of list.

Nr.	Information:
4	Pixel Size: Size of output pixels in meters Optional Value for the background pixels. An abitrary value appears in the background if no specific masking value is given by this function.
5	List of bands to process Button: >Select Bands< lets you select the bands to mosaic. Note: all files should have the same number of bands for mosaicing
6	Cut Options: *Edge Overlay*: The mosaicing is such that the first file is in the list is strictly on top *Center Cropped:* While mosaicing, the routine tries to find the middle of the overlap area between the new image and all the images mosaiced so far as a cut line. *Layer Stacking:* the files are stacked into one one file instead of mosaiced. The first file is taken as the reference for the stacking routine.
6	Output filename of mosaic

Actions:

> From Files <:

Calculates the outer borders for mosaicing and the spatial resolution from all currently selected files of the file list above.

> Preview < :

the first of the selected bands is mosaiced at a resolution reduced by a factor of 2 and the result is displayed.

> Run Process < :

the mosaicking is performed.

Procedure:

Note: the mosaicking uses a bilinear interpolation; no aggregation is done. This routines requires georeferenced data with:

- same coordinate system
- no rotation in ENVI header
- background coded with 0
- all files should have the same number of bands for mosaicing (but not for layer stacking)

The input resolutions of the imagery may vary and will be interpolated to the given resolution.

References

[1] Admiror I., 2002: Scattered data interpolation methods for electronic imaging systems: a survey. J. Electron. Imaging, **11(2)**: 157-176, (2002).

[2] Babey S.K., Anger C.D., Achal S.B., Ivanco T., Moise A., Costella P.R., and DeBliek J., 1999: Development of a Next-Generation Compact Airborne Spectrographic Imager: casi-2. 4th Int. Airb. R. S. Conf. and Exh., ERIM, Ottawa, CA, Vol. I:229-238.

[3] Bähr H.P., 1976: Geometrische Modelle für Abtasteraufzeichnungen von Erderkundungssatelliten. *Bildmessung u. Luftbildwesen*, 44(5):198-204.

[4] Bärs R., Watson L., and Weatherbee O., 1999: AISA as a Tool for Timely Commercial Remote Sensing. 4th Int. Airb. R. S. Conf. and Exh., ERIM, Ottawa, CA, Vol. I:239-246.

[5] Bäumker, M., Heimes F.-J., 2001: New Calibration and Computing Method for Direct Georeferencing of Image and Scanner Data Using the Position and Angular Data of an Hybrid Navigation System: In: Proceedings OEEPE-Workshop Integrated Sensor Orientation, Hannover, 17.-18. Sept. 2001. (available from: http://www.fh-bochum.de/fb5/baeumker/homemb4.htm).

[6] Berger M., Moreno J., Mü11er A., Schaepman M., Wursteisen P., Rast M., and Attema E., 2000: The Digital Airborne Imaging Spectrometer Experiment – DAISEX '99. IGARSS 2000, Hawaii, Vol. **VII**:3039-3041.

[7] Brewster S.B., 1999: Geometric Correction System Capabilities, Processing, and Applications. 4th Int. Airb. R. S. Conf. and Exh., ERIM, Ottawa, CA, Vol. I:162-169.

[8] Boardman J., 1999: Precision Geocoding of Low Altitude AVIRIS Data: Lessons learnt in 1998. R. Green (Ed.), 8th Ann. JPL Airb. Ea. Sc. Workshop, Pasadena, CA, JPL 99-17:63-68.

[9] Börner A., 1999: *Entwicklung und Test von Onboard-Algorithmen für die Landfernerkundung.* Inst. für Weltraumsensorik und Planetenerkundung, DLR Berlin, ISRN DLR-FB--1999-19, pp. 122.

[10] Clark R.N., Livo K.E., and Kokaly R.F., 1998: Geometric correction of AVIRIS Imagery Using On-Board Navigation and Engineering Data. Summaries of the Seventh JPL Airborne Earth Science Workshop, JPL, Pasadena (CA), published on WWW:pp 9.

[11] Chang S.-H. , Westfield M.J., Lehmann F., Oertel D., and Richter R., 1993: 79-Channel Airborne Imaging Spectrometer. Imaging Spectrometry of the Terrestrial Environment, Orlando, SPIE **1937**:164-172.

[12] Cocks T., Jenssen R., Stewart I., Wilson I., and Shields T., 1998: The HyMap™ Airborne Hyperspectral Sensor: The System, Calibration and Performance. Proceeding of 1st Earsel Workshop on Imaging Spectroscopy, EARSeL/RSL, Zürich, pp. 37-42.

[13] Congalton R.G., 1988: A Comparison of Sampling Schemes Used in Generating Error Matrices for Assessing the Accuracy of Maps Generated from Remotely Sensed Data. Photogr. Eng. and R. S., 54(5):593-600.

[14] Cramer M., Stallmann D. and Haala N., 2000. Direct georeferencing using GPS/inertial exterior ori-

entation for photogrammetric applications, IAPRS. International Archives on Photogrammetry and Remote Sensing IAPRS, Amsterdam, The Netherlands, July, Vol. XXXII/B, pp. 198-205.

[15] Cramer M. and Stallmann D., 2002. System Calibration for Direct Georeferencing, ISPRS Commission III Symposium. International Archives on Photogrammetry and Remote Sensing IAPRS, Graz, Austria, Vol. XXXIV, Com. III, Part A, pp. 79-84.

[16] Denker J.S., 2005: See How It Flies. Available from World Wide Web: <http://www.av8n.com/how/>, ISBN 7016405, chapter 19.

[17] Derenyi E.E., and Konecny G., 1966: Infrared Scan Geometry. *Photogramm. Eng.*, **32(5)**:773-779.

[18] Devereux B.J., Fuller R.M., Carter L., and Parsell R.J., 1990: Geometric Correction of Airborne Scanner Imagery by matching Delaunay Triangles. *Int. J. Remote Sensing*, **11(12)**:2237-2251.

[19] Doerffer R., Kunkel B. and Van Der Piepen H., 1989: ROSIS - An imaging spectrometer for remote sensing of chlrophyll fluorescence, Proc. *SPIE Conference 1070*, Los Angeles, pp. 9 (separatum).

[20] Exelis Visual Information Solutions (2016): ENVI The Environment for Visualizing Images, Version 5.3, Harris Geospatial, Exelis Visual Information Solutions, Boulder, Co, USA., http://www.harrisgeospatial.com/ProductsandSolutions/GeospatialProducts/ENVI.aspx (visited Sept. 2016)

[21] ESA, 2000: APEX Phase B Final Reports. Remote Sensing Laboratories, Univ. of Zürich, Switzerland, CD-Rom.

[22] Goshtasby A., 1988: Registration of Images with Geometric Distortions. *IEEE Trans. on Geosc. and R. S.*, **26(1)**:60-64.

[23] Green R.O., Chrien T.G., Sarture C., Eastwood M., Chippendale B., Kurzweil C., Chovit C.J., and Faust J.A., 1999: Operation, Calibration, Georectification, and Reflectance Inversion of NASA's Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Four- and Two-Meter Data Acquired from a Low-Altitude Platform. R. Green (Ed.), Proc. 8th Ann. JPL Airb. Ea. Sc. Workshop, Pasadena, CA, JPL 99-17:177-188.

[24] Gregory S., Hedges P., and Elgy J., 1999: The Geometric Correction of Airborne Line-Scanner Imagery. 4th Int. Airb. R. S. Conf. and Exh., ERIM, Ottawa, CA, Vol. I:178-185.

[25] Hutton J.J., Savina T., and Lithopoulos E., 1997: Photogrammetric Applications of Applanix's Position and Orientation System (POS). presented at the Third International Airborne Remote Sensing Conference and Exhibition, Copenhagen (DK), pp. 11.

[26] IDL, 1999: *Interactive Data Language (IDL)*, Version 5.3, Research Systems, Inc., Boulder, Colorado, USA.

[27] Itten K.I., Schaepman M., De Vos L., Hermans L., Schlaepfer H., and Droz F., 1997: APEX-Airborne Prism Experiment a New Concept for an Airborne Imaging Spectrometer. Proceedings of the Third International Airborne Remote Sensing Conference and Exhibition, ERIM International Inc., Copenhagen (DK), Vol. I, pp. 181-188.

[28] Konecny G., 1972: Geometrical Aspects of Remote Sensing. Int. Congress of Photogrammetry, Commission IV, Ottawa, Invited Paper, pp. 47.

[29] Konecny G., 1976a: Mathematical Models and Procedures for the Geometric Restitution of Remote

Sensing Imagery. XIII. Int. Congress of Photogrammetry, Commission III, Helsinki, Finland, Invited Paper, pp. 33.

[30] Konecny G., 1976b: Mathematische Modelle und Verfahren zur geometrischen Auswertung von Zeilenabtaster-Aufnahmen. *Bildmessung u. Luftbildwesen*, **44(5)**:188-197.

[31] Konecny G., and Lehmann G., 1984: *Photogrammetrie.* de Gruyter Lehrbuch, Walter de Gruyter, Berlin / New York, pp. 393.

[32] Konecny G., Kruck E., and Lohmann P., 1986: Ein universeller Ansatz für die geometrische Auswertung von CCD Zeilenabtasteraufnahmen. Bildmessung u. Luftbildwesen, 54(4):139-146.

[33] Konecny G., and Wu J., 1986: Analytische Streifen-Aerotriangulation für Stereo-Dreilinienabtaster. Bildmessung u. Luftbildwesen, 54(5):179-194.

[34] Konecny G., Lohmann P., Engel H., and Kruck E., 1987: Evaluation of SPOT Imagery on Analytical Photogrammetric Instruments. PE & RS, 53(9):1223-1230.

[35] Lam, S.-N., 1983: Spatial Interpolation Methods: A Review. The American Cartographer, **10(2)**: 129-149.

[36] Franke, R.,1982: Scattered Data Interpolation: Tests of Some Methods. Mathematics of Computation **38(157)**: 181-200.

[37] Park S. and R. Schowengerdt, 1983: Image Reconstruction by Parametric Cubic Convolution, Computer Vision, Graphics & Image Processing, **23**: 258-272.

[38] Buhmann M. D.,2000: Radial basis functions. Acta Numerica, **9(00)**, 1-38.

[39] Haber J., F. Zeilfelder, O. Davydov, and H.-P. Seider, 2001: Smooth approximation and rendering of large scattered data sets. T. Ertl et al. (Editors), In: *IEEE Visualization 2001*, (IEEE Computer Society, Vol. 571, pp. 341-347.

[40] Rigol J.P., C. H. Jarvis, and N. Stuart, 2001: Artificial neural networks as a tool for spatial interpolation. Int. J. of Geographical Information Science, **15(4)**, 323-343.

[41] Leckie D., Gougeon F., Jay C., Tam A., Leach R., and Walsworth N., 1999: A Flexible View Angle BRDF Compensation System using Aircraft Orientation Information. 4th Int. Airb. R. S. Conf. and Exh., ERIM, Ottawa, CA, Vol. I:115.

[42] Lee C., Theiss H.J., Bethel J.S., and Mikhail E.M., 2000: Rigorous Mathematical Modeling of Airborne Pushbroom Imaging Systems. *PE & RS*, **66(4)**:385-392.

[43] McCubbin I., and Lang H., 1999: Mapping Environmental Contaminants at Ray Mine, AZ. R. Green (Ed.),8th Annual JPL Airborne Earth Science Workshop, JPL **99-17**:287-290.

[44] McGwire K.C., 1996: Cross-Validated Assessment of Geometric Accuracy. *PE & RS*, **62(10)**:1179-1187.

[45] Meyer P., 1994: A Parametric Approach for the Geocoding of Airborne Visible/Infrared Spectrometer (AVIRIS) Data in Rugged Terrain. Remote Sens. Environ., Elsevier, 49(2):118 - 130.

[46] Mockridge W., and Leach M., 1997: Integrating a Deadalus 1268 ATM, Attitude GPS and AHRS to Achieve Scan Rate Geometric Correction. Proc. of the 3rd International Airborne Remote Sensing Conference and Exhibition, ERIM International Inc., Copenhagen (DK), Vol. **I**:158-165.

[47] Oertel D., 1994: The First Stage of the Digital Airborne Imaging Spectrometer DAIS-7915 Laboratory Calibration at DLR Oberpfaffenhofen. Proceedings of the First International Airborne Remote

Sensing Conference and Exhibition; Application, Technology and Science, Vol. II, Strasbourg (F), pp II-214 - II-224

[48] Reulke N., Reulke R., Schaepman M., Schläpfer D., Strobl P., and Müller A., 1997: *Geometric Correction of Hyperspectral Scanner Data with Simultaneously Acquired Stereo Data.* E.P. Baltsavias (Ed.), Joint ISPRS Commission III/IV, Vol. 32, Part 3-4W2, Stuttgart (G), pp 9-14

[49] Reulke N., Reulke R., Schaepman M., Schläpfer D., Strobl P., and Müller A., 1997: Imaging Surface Roughness from Hyperspectral Sensors: An Example from the GER 63 Channel Scanner Data. Proceedings of the Third International Airborne Remote Sensing Conference and Exhibition, ERIM International Inc., Copenhagen (DK), Vol. II:pp. 329-336.

[50] Richter R., 1998: Correction of satellite imagery over mountainous terrain. *Applied Optics*, Opt. Society of Americ, **37(18)**:4004-4014.

[51] Richter R. and Schläpfer, D., 2001: Geo-atmospheric correction of airborne imaging spectrometry data. Part II: Atmospheric and topographic correction. *Int. J. of R. S.,* 23(13):2609-2630.

[52] Sandau R. et al., 2000. Design Principles of the LH Systems ADS40 Airborne Digital Sensor. IAPRS, Vol. XXXIII: pp. 8.

[53] Salamonowicz P.H., 1986: An Analytic Correction Method for Satellite MSS Geometric Distortions. *PE & RS*, **52(4)**:491-499.

[54] Sawada N., Kidode M., Shinoda H., Asada H., Iwanaga M., Watanabe S., Mori K.-I., and Akiyama M., 1981: An Analytic Correction Method for Satellite MSS Geometric Distortions. *PE & RS*, **47(8)**:1195-1203.

[55] Schaepman M., Schläpfer D., Strobl P., and Mueller A., 1997: *Ground Spectroradiometric Measurements in Support of the Validation of the Calibration of DIGITAL Airborne Imaging Spectrometer (DAIS 7915) Data.* 3rd Int. Airb. Rem. Sens. Conf., ERIM, Copenhagen, Vol. I, pp 217-223

[56] Schläpfer D., 1998: Differential Absorption Methodology for Imaging Spectroscopy of Atmospheric Water Vapor. Remote Sensing Series , RSL, Zuerich, Vol. 32, pp. 131.

[57] Schläpfer D., Meyer P., and Itten K.I., 1998a: Parametric Geocoding of AVIRIS Data Using a Ground Control Point Derived Flightpath. Summaries of the Seventh JPL Airborne Earth Science Workshop, JPL, Pasadena (CA), 97-21, Vol. 1:pp. 367-372.

[58] Schläpfer D., Schaepman M., and Itten K.I., 1998b: PARGE: Parametric Geocoding Based on GCP-Calibrated Auxiliary Data. Descour, M.R. and Shen, S.S., Imaging Spectrometry IV, SPIE, San Diego, Vol. 3438:334-344.

[59] Schläpfer D., Schaepman M., and Itten K.I., 1998c: Level II Pre-Processing Concept for the Airborne PRISM Experiment (APEX). Proceeding of 1st Earsel Workshop on Imaging Spectroscopy, EARSeL/RSL, Zürich, 89-96.

[60] Schläpfer D., Schaepman M., Bojinski S., and Börner A., 2000: Calibration Concept for the Airborne PRISM Experiment (APEX). *Can. J. of R. S.*, **26**(5):455-465.

[61] Schläpfer D., Schaepman M., and Strobl P., 2001: Impact of Spatial Resampling Methods on the Radiometric Accuracy of Airborne Imaging Spectrometer Data. in Proceedings 5th Int. Airb. R. S. Conf. and Exh., VERIDIAN, San Francisco / Miami, CD-ROM, pp.8.

[62] Schläpfer D., and Richter R., 2002: Geo-atmospheric processing of airborne imaging spectrometry data. Part 1: Parametric Ortho-Rectification Process. Int. J. of R. S., 23(13):2609-2630.

[63] Schott J.R., Gallagher T., Nordgren B., Sanders L., and Barsi J., 1999: Radiometric Correction Procedures and Performance for the Modular Imaging Spectrometer Instrument (MISI). 4th Int. Airb. R. S. Conf. and Exh., ERIM, Ottawa, CA, Vol. I:328-335.

[64] Schowengerdt R.A., 1997: Remote Sensing: Models and Methods for Image Processing. Academic Press, 2nd Ed., pp. 522.

[65] Shukai L., Hongjian Y., Jianming L., Shaocuang L., Tong L., and Jingzhi M., 1999: An Air-to-Ground Positioning System Applied to Airborne Remote Sensing. 4th Int. Airb. R. S. Conf. and Exh., ERIM, Ottawa, CA, Vol. I:194-200.

[66] Skaloud, J., LI, Y.C, and Schwarz, K.P., 1997, Airborne testing of C-MIGITS II - low cost integrated GPS/IN. International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS97), Banff, Canada, pp. 161-166.

[67] Sörgel U., and Thönnessen U., 1999: Automatic geocoding of high value targets using structural image analysis and GIS data. EUROPTO, SPIE Vol. **3871**:10-18.

[68] Staenz K., Szeredi T., and Schwarz J., 1998: ISDAS - A System for Processing/Analyzing Hyperspectral Data. Can. J. of R. S., Vol. 24, No. 2:pp 99-113.

[69] Starck J.-L., Murtagh F., and Bijaoui A., 1998: Image Processing and Data Analysis: The Multiscale Approach. Cambridge University Press, pp. 286.

[70] Strobl P., Mueller A., Schlaepfer D., and Schaepman M., 1997: *Laboratory Calibration and Inflight Validation of the Digital Airborne Imaging Spectrometer DAIS 7915 for the 1996 Flight Season*.Proc. of SPIE Aero-Sense'97, Orlando, pp 8

[71] Vane G., Goetz A.F.H., 1988: *Terrestrial Imaging Spectroscopy.* R.S. of Env., Nr. 24, pp 1-29

[72] Wilson A.K., Mockridge W., and Robinson M.-C., 1997: Post-Processing to achieve Radiometric and Geometric Correction of ATM and CASI Data. Proc. of the 3rd Int. Airb. R. S. Conf. and Exh., ERIM International Inc., Copenhagen (DK), Vol. I:447-454.

[73] Zhang W., Albertz J., and Li Z., 1994: Rectification of Airborne Line-Scanner Imagery Utilizing Flight Parameters. Proc. First International Airborne Remote Sensing Conference and Exhibition, Strasbourg (F), Vol. II:447-456

A: List of Ellipsoids

Number	Name	Major SemiAxis	Excentricity
1	Airy 1830	6377563.0	0.0066705402
2	Australian National	6378160.0	0.0066945418
3	Bessel 1841	6377397.0	0.0066743721
4	Bessel 1841 (Namibia)	6377484.0	0.0066743721
5	Clarke 1866	6378206.0	0.0067686578
6	Clarke 1880	6378249.0	0.0068035112
7	Everest (India 1830)	6377276.0	0.0066378471
8	Fischer 1960 (Mercury)	6378166.0	0.0066934219
9	Fischer 1968	6378150.0	0.0066934219
10	GRS 1967	6378160.0	0.0066946051
11	GRS 80	6378137.0	0.0066943802
12	Helmert 1906	6378200.0	0.0066934219
13	Hough 1960	6378270.0	0.0067226700
14	International 1924	6378388.0	0.0067226700
15	Krassovsky 1940	6378245.0	0.0066934219
16	Modified Airy	6377340.0	0.0066705402
17	Everest (Modified)	6377304.0	0.0066378471
18	Modified Fischer 1960	6378155.0	0.0066934219
19	South American 1969	6378160.0	0.0066945418
20	WGS 60	6378165.0	0.0066934219
21	WGS 66	6378145.0	0.0066945418
22	WGS 72	6378135.0	0.0066943178
23	WGS 84	6378137.0	0.0066943802

B: List of Projections

Number	Name	Zones
0	Arbitrary	-
1	UTM	1-60
2	Gauss-Krueger	1-6
3	Gauss-Boaga	1-2
4	US State Plane	101-xxx
5	Swiss	-
6	Custom (Self defined)	-

C: List of Datums

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
1	Adindan	Clarke 1880	-118	-14	218	Burkina Faso
2	Adindan	Clarke 1880	-134	-2	210	Cameroon
3	Adindan	Clarke 1880	-165	-11	206	Ethiopia
4	Adindan	Clarke 1880	-123	-20	220	Mali
5	Adindan	Clarke 1880	-166	-15	204	MEAN FOR Ethiopia/ Sudan
6	Adindan	Clarke 1880	-128	-18	224	Senegal
7	Adindan	Clarke 1880	-161	-14	205	Sudan
8	Afgooye	Krassovsky 1940	-43	-163	45	Somalia
9	Ain el Abd 1970	International 1924	-150	-250	-1	Bahrain
10	Ain el Abd 1970	International 1924	-143	-236	7	Saudi Arabia
11	American Samoa 1962	Clarke 1866	-115	118	426	American Samoa Islan ds
12	Anna 1 Astro 1965	Australian National	-491	-22	435	Cocos Islands
13	Antigua Island Astro 1943	Clarke 1880	-270	13	62	Antigua (Leeward Islands)
14	Arc 1950	Clarke 1880	-138	-105	-289	Botswana
15	Arc 1950	Clarke 1880	-153	-5	-292	Burundi
16	Arc 1950	Clarke 1880	-125	-108	-295	Lesotho
17	Arc 1950	Clarke 1880	-161	-73	-317	Malawi
18	Arc 1950	Clarke 1880	-143	-90	-294	MEAN FOR Botswana/ Lesotho/ Malawi/ Swaziland/ Zaire/ Zambia/ Zimbabwe
19	Arc 1950	Clarke 1880	-134	-105	-295	Swaziland
20	Arc 1950	Clarke 1880	-169	-19	-278	Zaire
21	Arc 1950	Clarke 1880	-147	-74	-283	Zambia
22	Arc 1950	Clarke 1880	-142	-96	-293	Zimbabwe
23	Arc 1960	Clarke 1880	-160	-6	-302	MEAN FOR Kenya/ Tanzania
24	Arc 1960	Clarke 1880	-157	-2	-299	Kenya
25	Arc 1960	Clarke 1880	-175	-23	-303	Taanzania

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
26	Ascension Island 1958	International 1924	-205	107	53	Ascension Is land
27	Astro Beacon E 1945	International 1924	145	75	-272	Iwo Jima
28	Astro DOS 71/4	International 1924	-320	550	-494	St Helena Island
29	Astro Tern Island (FRIG) 1961	International 1924	114	-116	-333	Tern Island
30	Astronomical Station 1952	International 1924	124	-234	-25	MarcusIsland
31	Australian Geodetic 1966	Australian National	-133	-48	148	Australia/ Tasmania
32	Australian Geodetic 1984	Australian National	-134	-48	149	Australia/ Tasmania
33	Ayabelle Lighthouse	Clarke 1880	-79	-129	145	Djibouti
34	Bellevue (IGN)	International 1924	-127	-769	472	Efate & Erromango Islands
35	Bermuda 1957	Clarke 1866	-73	213	296	Bermuda
36	Bissau	International 1924	-173	253	27	Guinea-Bissau
37	Bogota Observatory	International 1924	307	304	-318	Colombia
38	Bukit Rimpah	Bessel 1841	-384	664	-48	Indonesia (Bangka & BelitungIds)
39	Camp Area Astro	International 1924	-104	-129	239	Antarctica (McMurdo Camp Area)
40	Campo Inchauspe	International 1924	-148	136	90	Argentina
41	Canton Astro 1966	International 1924	298	-304	-375	Phoenix Islands
42	Cape	Clarke 1880	-136	-108	-292	South Africa
43	Cape Canaveral	Clarke 1866	-2	151	181	Bahamas/ Florida
44	Carthage	Clarke 1880	-263	6	431	Tunisia
45	Chatham Island Astro 1971	International 1924	175	-38	113	New Zealand (Chatham Island)
46	Chua Astro	International 1924	-134	229	-29	Paraguay
47	Corrego Alegre	International 1924	-206	172	-6	Brazil
48	Dabola	Clarke 1880	-83	37	124	Guinea
49	Deception Island	Clarke 1880	260	12	-147	Deception Island/ Antarctia
50	Djakarta (Batavia)	Bessel 1841	-377	681	-50	Indonesia (Sumatra)
51	DOS 1968	International 1924	230	-199	-752	New Georgia Islands (Gizo Island)
52	Easter Island 1967	International 1924	211	147	111	Easter Island
53	Estonia/ Coordinate System 1937	Bessel 1841	374	150	588	Estonia
54	European 1950	International 1924	-104	-101	-140	Cyprus

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
55	European 1950	International 1924	-130	-117	-151	Egypt
56	European 1950	International 1924	-86	-96	-120	England/ Channel Islands/ Scotland/ Shetland Islands
57	European 1950	International 1924	-86	-96	-120	England/ Ireland/ Scotland/ Shetland Islands
58	European 1950	International 1924	-87	-95	-120	Finland/ Norway
59	European 1950	International 1924	-84	-95	-130	Greece
60	European 1950	International 1924	-117	-132	-164	Iran
61	European 1950	International 1924	-97	-103	-120	Italy (Sardinia)
62	European 1950	International 1924	-97	-88	-135	Italy (Sicily)
63	European 1950	International 1924	-107	-88	-149	Malta
64	European 1950	International 1924	-87	-98	-121	MEAN FOR Austria/ Belgium/ Denmark/ Finland/ France/ W Germany/ Gibraltar/ Greece/ Italy/ Luxembourg/ Netherlands/ Norway/ Portugal/ Spain/ Sweden/ Switzerland
65	European 1950	International 1924	-87	-96	-120	MEAN FOR Austria/ Denmark/ France/ W Germany/ Netherlands/ Switzerland
66	European 1950	International 1924	-103	-106	-141	MEAN FOR Iraq/ Israel/ Jordan/ Lebanon/ Kuwait/ Saudi Arabia/ Syria
67	European 1950	International 1924	-84	-107	-120	Portugal/ Spain
68	European 1950	International 1924	-112	-77	-145	Tunisia
69	European 1979	International 1924	-86	-98	-119	MEAN FOR Austria/ Finland/ Netherlands/ Norway/ Spain/ Sweden/ Switzerland
70	Fort Thomas 1955	Clarke 1880	-7	215	225	Nevis/ St. Kitts (Leeward Islands)
71	Gan 1970	International 1924	-133	-321	50	Republic of Maldives
72	Geodetic Datum 1949	International 1924	84	-22	209	New Zealand
73	Graciosa Base SW 1948	International 1924	-104	167	-38	Azores (Faial/ Graciosa/ Pico/ Sao Jorge/ Terceira)
74	Guam 1963	Clarke 1866	-100	-248	259	Guam
75	Gunung Segara	Bessel 1841	-403	684	41	Indonesia (Kalimantan)
76	GUX 1 Astro	International 1924	252	-209	-751	Guadalcanal Island
77	Herat North	International 1924	-333	-222	114	Afghanistan

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
78	Hermannskogel Datum	Bessel 1841 (Namibia)	653	-212	449	Croatia -Serbia/ Bosnia-Herzegovina?
79	Hjorsey 1955	International 1924	-73	46	-86	Iceland
80	Hong Kong 1963	International 1924	-156	-271	-189	Hong Kong
81	Hu-Tzu-Shan	International 1924	-637	-549	-203	Taiwan
82	Indian	Everest (India 1830)	282	726	254	Bangladesh
83	Indian	Everest (India 1956)	295	736	257	India/ Nepal
84	Indian	Everest (Pakistan)	283	682	231	Pakistan
85	Indian 1954	Everest (India 1830)	217	823	299	Thailand
86	Indian 1960	Everest (India 1830)	182	915	344	Vietnam (Con Son Island)
87	Indian 1960	Everest (India 1830)	198	881	317	Vietnam (Near 16?N)
88	Indian 1975	Everest (India 1830)	210	814	289	Thailand
89	Indonesian 1974	Indonesian 1974	-24	-15	5	Indonesia
90	Ireland 1965	Modified Airy	506	-122	611	Ireland
91	ISTS 061 Astro 1968	International 1924	-794	119	-298	South Georgia Islands
92	ISTS 073 Astro 1969	International 1924	208	-435	-229	Diego Garcia
93	Johnston Island 1961	International 1924	189	-79	-202	Johnston Island
94	Kandawala	Everest (India 1830)	-97	787	86	Sri Lanka
95	Kerguelen Island 1949	International 1924	145	-187	103	Kerguelen Island
96	Kertau 1948	Everest (Malay. & Sing)	-11	851	5	West Malaysia & Singapore
97	Kusaie Astro 1951	International 1924	647	177 7	- 112 4	Caroline Islands
98	Korean Geodetic System	GRS 80	0	0	0	South Korea
99	L. C. 5 Astro 1961	Clarke 1866	42	124	147	Cayman Brac Island
100	Leigon	Clarke 1880	-130	29	364	Ghana
101	Liberia 1964	Clarke 1880	-90	40	88	Liberia
102	Luzon	Clarke 1866	-133	-77	-51	Philippines (Excluding Mindanao)

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
103	Luzon	Clarke 1866	-133	-79	-72	Philippines (Mindanao)
104	M Poraloko	Clarke 1880	-74	-130	42	Gabon
105	Mahe 1971	Clarke 1880	41	-220	-134	Mahe Island
106	Massawa	Bessel 1841	639	405	60	Ethiopia (Eritrea)
107	Merchich	Clarke 1880	31	146	47	Morocco
108	Midway Astro 1961	International 1924	912	-58	122 7	Midway Islands
109	Minna	Clarke 1880	-81	-84	115	Cameroon
110	Minna	Clarke 1880	-92	-93	122	Nigeria
111	Montserrat Island Astro 1958	Clarke 1880	174	359	365	Montserrat(Leeward Islands)
112	Nahrwan	Clarke 1880	-247	-148	369	Oman (Masirah Island)
113	Nahrwan	Clarke 1880	-243	-192	477	Saudi Arabia
114	Nahrwan	Clarke 1880	-249	-156	381	United Arab Emirates
115	Naparima BWI	International 1924	-10	375	165	Trinidad & Tobago
116	North American 1927	Clarke 1866	-5	135	172	Alaska (Excluding Aleutian Ids)
117	North American 1927	Clarke 1866	-2	152	149	Alaska (Aleutian IdsEast of 180?W)
118	North American 1927	Clarke 1866	2	204	105	Alaska (Aleutian Ids West of 180?W)
119	North American 1927	Clarke 1866	-4	154	178	Bahamas (Except San Salvador Id)
120	North American 1927	Clarke 1866	1	140	165	Bahamas (San SalvadorIsland)
121	North American 1927	Clarke 1866	-7	162	188	Canada (Alberta/ British Columbia)
122	North American 1927	Clarke 1866	-9	157	184	Canada (Manitoba/ Ontario)
123	North American 1927	Clarke 1866	-22	160	190	Canada (New Brunswick/ Newfoundland/ Nova Scotia/ Quebec)
124	North American 1927	Clarke 1866	4	159	188	Canada (Northwest Territories/ Saskatchewan)
125	North American 1927	Clarke 1866	-7	139	181	Canada (Yukon)
126	North American 1927	Clarke 1866	0	125	201	Canal Zone
127	North American 1927	Clarke 1866	-9	152	178	Cuba
128	North American 1927	Clarke 1866	11	114	195	Greenland (Hayes Peninsula)

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
129	North American 1927	Clarke 1866	-3	142	183	MEAN FOR Antigua/ Barbados/ Barbuda/ Caicos Islands/ Cuba/ Dominican Republic/ Grand Cayman/ Jamaica/ Turks Islands
130	North American 1927	Clarke 1866	0	125	194	MEAN FOR Belize/ Costa Rica/ El Salvador/ Guatemala/ Honduras/ Nicaragua
131	North American 1927	Clarke 1866	-10	158	187	MEAN FOR Canada
132	North American 1927	Clarke 1866	-8	160	176	MEAN FOR CONUS
133	North American 1927	Clarke 1866	-9	161	179	MEAN FOR CONUS (Eastof Mississippi/ River Including Louisiana/ Missouri/ Minnesota)
134	North American 1927	Clarke 1866	-8	159	175	MEAN FOR CONUS (Westof Mississippi/ River Excluding Louisiana/ Minnesota/ Missouri)
135	North American 1927	Clarke 1866	-12	130	190	Mexico
136	North American 1983	GRS 80	0	0	0	Alaska (Excluding Aleutian Ids)
137	North American 1983	GRS 80	-2	0	4	Aleutian Ids
138	North American 1983	GRS 80	0	0	0	Canada
139	North American 1983	GRS 80	0	0	0	CONUS
140	North American 1983	GRS 80	1	1	-1	Hawaii
141	North American 1983	GRS 80	0	0	0	Mexico/ Central America
142	North Sahara 1959	Clarke 1880	-186	-93	310	Algeria
143	Observatorio Meteorologico 1939	International 1924	-425	-169	81	Azores (Corvo & Flores Islands)
144	Old Egyptian 1907	Helmert 1906	-130	110	-13	Egypt
145	Old Hawaiian	Clarke 1866	89	-279	-183	Hawaii
146	Old Hawaiian	Clarke 1866	45	-290	-172	Kauai
147	Old Hawaiian	Clarke 1866	65	-290	-190	Maui
148	Old Hawaiian	Clarke 1866	61	-285	-181	MEAN FOR Hawaii/ Kauai/ Maui / Oahu
149	Old Hawaiian	Clarke 1866	58	-283	-182	Oahu
150	Oman	Clarke 1880	-346	-1	224	Oman
151	Ordnance Survey Great Britain 1936	Airy 1830	371	-112	434	England
152	Ordnance Survey Great Britain 1936	Airy 1830	371	-111	434	England/ Isle of Man/ Wales

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
153	Ordnance Survey Great Britain 1936	Airy 1830	375	-111	431	MEAN FOR England/ Isle of Man/ Scotland/ Shetland Islands/ Wales
154	Ordnance Survey Great Britain 1936	Airy 1830	384	-111	425	Scotland/ Shetland Islands
155	Ordnance Survey Great Britain 1936	Airy 1830	370	-108	434	Wales
156	Pico de las Nieves	International 1924	-307	-92	127	Canary Islands
157	Pitcairn Astro 1967	International 1924	185	165	42	Pitcairn Island
158	Point 58	Clarke 1880	-106	-129	165	MEAN FOR Burkina Faso & Niger
159	Pointe Noire 1948	Clarke 1880	-148	51	-291	Congo
160	Porto Santo 1936	International 1924	-499	-249	3	Porto Santo/ Madeira Islands
161	Provisional South American 1956	International 1924	-270	188	314	Bolivia
162	Provisional South American 1956	International 1924	-270	183	-390	Chile (Northern/ Near 19?S)
163	Provisional South American 1956	International 1924	-305	243	-442	Chile (Southern/ Near 43?S)
164	Provisional South American 1956	International 1924	-282	169	-371	Colombia
165	Provisional South American 1956	International 1924	-278	171	-367	Ecuador
166	Provisional South American 1956	International 1924	-298	159	-369	Guyana
167	Provisional South American 1956	International 1924	-288	175	-376	MEAN FOR Bolivia/ Chile/ Colombia/ Ecuador/ Guyana/ Peru/ Venezuela
168	Provisional South American 1956	International 1924	-279	175	-379	Peru
169	Provisional South American 1956	International 1924	-295	173	-371	Venezuela
170	Provisional South Chilean 1963	International 1924	16	196	93	Chile (Near 53?S) (Hito XVIII)
171	Puerto Rico	Clarke 1866	11	72	-101	Puerto Rico/ Virgin Islands
172	Pulkovo 1942	Krassovsky 1940	28	-130	-95	Russia
173	Qatar National	International 1924	-128	-283	22	Qatar
174	Qornoq	International 1924	164	138	-189	Greenland (South)

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
175	Reunion	International 1924	94	-948	-1262	Mascarene Islands
176	Rome 1940	International 1924	-225	-65	9	Italy (Sardinia)
177	S-42 (Pulkovo 1942)	Krassovsky 1940	28	-121	-77	Hungary
178	S-42 (Pulkovo 1942)	Krassovsky 1940	23	-124	-82	Poland
179	S-42 (Pulkovo 1942)	Krassovsky 1940	26	-121	-78	Czechoslavakia
180	S-42 (Pulkovo 1942)	Krassovsky 1940	24	-124	-82	Latvia
181	S-42 (Pulkovo 1942)	Krassovsky 1940	15	-130	-84	Kazakhstan
182	S-42 (Pulkovo 1942)	Krassovsky 1940	24	-130	-92	Albania
183	S-42 (Pulkovo 1942)	Krassovsky 1940	28	-121	-77	Romania
184	S-JTSK	Bessel 1841	589	76	480	Czechoslavakia (Prior 1 JAN 1993)
185	Santo (DOS) 1965	International 1924	170	42	84	Espirito Santo Island
186	Sao Braz	International 1924	-203	141	53	Azores (Sao Miguel/ Santa Maria Ids)
187	Sapper Hill 1943	International 1924	-355	21	72	East Falkland Island
188	Schwarzeck	Bessel 1841 (Namibia)	616	97	-251	Namibia
189	Selvagem Grande 1938	International 1924	-289	-124	60	Salvage Islands
190	Sierra Leone 1960	Clarke 1880	-88	4	101	Sierra Leone
191	South American 1969	South American 1969	-62	-1	-37	Argentina
192	South American 1969	South American 1969	-61	2	-48	Bolivia
193	South American 1969	South American 1969	-60	-2	-41	Brazil
194	South American 1969	South American 1969	-75	-1	-44	Chile
195	South American 1969	South American 1969	-44	6	-36	Colombia
196	South American 1969	South American 1969	-48	3	-44	Ecuador
197	South American 1969	South American 1969	-47	26	-42	Ecuador (Baltra/ Galapagos)
198	South American 1969	South American 1969	-53	3	-47	Guyana

#	Name	Ellipsoid	Δ x	Δ y	Δ z	Country/Region
199	South American 1969	South American 1969	-57	1	-41	MEAN FOR Argentina/ Bolivia/ Brazil/ Chile/ Colombia/ Ecuador/ Guyana/ Paraguay/ Peru/ Trinidad & Tobago/ Venezuela
200	South American 1969	South American 1969	-61	2	-33	Paraguay
201	South American 1969	South American 1969	-58	0	-44	Peru
202	South American 1969	South American 1969	-45	12	-33	Trinidad & Tobago
203	South American 1969	South American 1969	-45	8	-33	Venezuela
204	South Asia	Modified Fischer 1960	7	-10	-26	Singapore
205	Tananarive Observatory 1925	International 1924	-189	-242	-91	Madagascar
206	Timbalai 1948	Everest (Sabah Sarawak)	-679	669	-48	Brunei/ E. Malaysia (Sabah Sarawak)
207	Tokyo	Bessel 1841	-148	507	685	Japan
208	Tokyo	Bessel 1841	-148	507	685	MEAN FOR Japan/ South Korea/ Okinawa
209	Tokyo	Bessel 1841	-158	507	676	Okinawa
210	Tokyo	Bessel 1841	-147	506	687	South Korea
211	Tristan Astro 1968	International 1924	-632	438	-609	Tristan da Cunha
212	Viti Levu 1916	Clarke 1880	51	391	-36	Fiji (Viti Levu Island)
213	Voirol 1960	Clarke 1880	-123	-206	219	Algeria
214	Wake Island Astro 1952	International 1924	276	-57	149	Wake Atoll
215	Wake-Eniwetok 1960	Hough 1960	102	52	-38	Marshall Islands
216	WGS 1972	WGS 72	0	0	0	Global Definition
217	WGS 1984	WGS 84	0	0	0	Global Definition
218	Yacare	International 1924	-155	171	37	Uruguay
219	Zanderij	International 1924	-265	120	-358	Suriname
220	Potsdam Rauenberg	Bessel 1841	606	23	413	Germany
221	CH-1903	Bessel 1841	674.37402	15.0560	405.34601	Switzerland

D: Example Batch Program

Below, an example batch script code is given as it might be used for operational processing of imaging spectroscopy data. It's source code can be found in the /etc directory of the parge distribution. Note that only an example setup is given here - a detailed description of all batch commands is given in Chapter 4.

```
;+
; PARGE batch program template
; ----------------------------
; version 3.1 of parge
;
; use this program for an IGM-based processing of PARGE data
; allocation of the parge variables should
; have been done before running this script and
; run this program by starting 'parge' or issuing the command 'parge,/
norun'
;
; replace statements starting with ** for your own data
;
; PARAMETERS
;  cube: image data cube to be processed
;  navfile: file containing the INS/IMU data
;  demfile:  DEM file in ENVI format
;  sensorfile: file containing the internal sensor geometry
;
;
; KEYWORDS
;  bore: boresight status
;  outpath: path to output directory
;  resol: resolution of output
;
; provided by ReSe, August 2011s
;-


PRO parge_batch, cube, navfile, demfile,sensorfile, bore=bore, $
    outpath=outpath, resol=resol

;; get PARGE common blocks
 common c_geovar
```

```
 common c_geostruc

;; get all datum definitions (required for generic
;; conversions, uses the file 'ellipsoids.inc',
;; which is also available in the etc directory on request.

 @ellipsoids.inc ;; include statement


;; define output resolution in meters
 if n_elements(resol) eq 0 then resol = 1.
 x_res=resol[0]
 y_res=resol[0]

;; prepare the output: start a batch log file
;;
IF n_elements(outpath) EQ 0 THEN outpath = filext(cube, /path)

gcsfile=outpath+filext(cube,/nam)+'.gcs'
log_file=outpath+filext(cube,/nam)+'.plog'

;; open and init a batch log file
gc_initbatchlog,log_file


;; Step1: DEM:Import:ENVI
;; ----------
;; read ENVI DEM file (named 'mydem_ele'
 gc_rdenvdem, demfile


;; Step2: File:Import:Hyspex
;; ----------
;; define coordinate system for output (compare contents of ellip-
soids.inc)
 dtnum= 138                    ;NAD 83
 proj = 1                      ;for UTM
 utmzone = -1                  ;default UTM zone
 cord = gc_getcord('WGS-84', dtnum, proj, zone=zone)

;; other example (may be deleted)
; dtnum= 220                    ;Potsdam
; proj = 1                      ;Gauss-K
; utmzone = 4                   ;default Zone
; cord = gc_getcord('Bessel 1841', dtnum, proj, zone=utmzone)

;; import the image data (use the filter function specific to your sen-
sor !)
 **sensor**_imp, cube, navfile, sensorfile, $ ;
```

```
                **further_arguments**


;; make coordinate conversion, apply heading offset
 navarr=gc_coordtf(gpsarr, cord, $
                  coordinfo = coordinfo, /headoff)

;; set comments for import
 aux(4:6).coord = coordinfo


;; Step3: Calibration of Attitude Data
;; --------------
;; import offsets from previous boresighting status
gc_impoffs,'**boresight_flight.gcs'

;; Step 4: IGM Creation (geocoding)
;; --------------------------------

;; set names for outputs - maybe some other standards would apply for
APEX

 result.igm_name = outpath + filext(result.igm_name,/nam, /ext)
 result.remaparr_name = outpath + filext(result.remaparr_name,/nam, /
ext)
 result.geocube_name = outpath + filext(result.geocube_name,/nam, /
ext)
 result.georgb_name = outpath + filext(result.georgb_name,/nam, /ext)
 result.sca_name = outpath + filext(result.sca_name,/nam, /ext)

;; creating IGM, Function 'Processor:IGM Main Processor'
 gc_maini,/atmo                 ; keyword /atmo creates the scan
                                ; angle file *_sca.bsq
                                ; geocodes raw image to cube
                                ; this assumes that the map file
                                ; and output file have default names



;; Step 5: Rectification
;; ---------------------
;; Function 'Processor:IGM Cube Processor'
;; default: limits/resolution driven by DEM definition (override: lim-
its-keyword)
;; ll:lowerleft edge, ur:upper right edge, psize: pixel size -
;; uncomment and put Numbers here in case you need specific dimen-
sions!!

 ;; create map array (based on DEM geometry)
```

```
 bords=gc_borders()  ;; get borders of the image on DEM

 ; info: limits=[ll_x,ll_y,ur_x,ur_y,psize_x,psize_y] **

 gc_cremap,result.igm_name, result.remaparr_name, n_tiles=4,expand=3,$
          limits=[floor(min(bords[*,*,0])),floor(min(bords[*,*,1])), $
                   ceil(max(bords[*,*,0])), $
                   ceil(max(bords[*,*,1])),x_res,y_res]


 ;; create RGB - TIFF by default; based on above geometry
 gc_rgb, result.rgbbands, result.georgb_name,  mask = 0, /line


 ;; process the full cube (optional)
 gc_cube_ntile, /bilinear       ; nearest bilinear neighbour

;; gc_cube_tile, limits=[ll_x,ll_y,ur_x,ur_y,psize_x,psize_y]
                                 ; alternative to above two statements:
                                  ; bilinear/gridded
                                  ; interpolation

;; Step 6: Save Status
;; --------
 gc_savestatus, datapath + '**status_File**.gcs'

 close,/all

END
```

This script is usually called by a script or from the IDL prompt. An example script as provided with the software distribution is given below:

```
;; script to be run for a full batch process
;; run this file from a prompt by the command 'idl parge_script.txt'
;; or from within IDL
;; names in <xx> brackets are to be adapted.
;; paths should always be the full explicite path to the data location
;; by  '@parge_script.txt'

.reset_session        ; clean all variables

;; initalize
parge,/norun

datapath='<path_to_your_input_data>'
```

```
outpath = '<path_to_your_output_directory>'

;; list of cubes and INS files to process
cubes     = datapath + ['cube1',$
                        'cube2',$
                        'cube3']

navfile   = datapath + ['nav1',$
                        'nav2',$
                        'nav3']

;; name of DEM (may also be a list as above)
demfile = '<your_dem_file>'

;; name of sensor definition file
sensorfile= '<your_sensor_file>'

;; boresight calibration status

borestatus = 'thun2011_02.gcs'

xparge_batch           ; run the parge batch script

FOR i =0,n_elements(cubes)-1 do $
   parge_batch, cubes[i], navfile[i], demfile, sensorfile,  $
   bore=borestatus,  outpath = outpath


exit                   ; exit IDL
```

E: List of Routines

F: Index