System Prototype Description (SPD)

Prepared by: Glaciers_cci consortium
Contract: 4000101778/10/I-AM
Name: Glaciers_cci-D3.1_SPD
Version: 0.7
Date: 22.08. 2013

Contact:
Frank Paul
Department of Geography
University of Zurich
frank.paul@geo.uzh.ch

Technical Officer:
Stephen Plummer
ESA ESRIN

Document status sheet

Version	Date	Changes	Approval
0.1	14.10.2012	Initial draft	
0.2	02.11.2012	Extended table of contents (OF)	
0.3	20.11.2012	Added Glacier Velocity part (OF)	
0.4	06.12.2012	Integrated all contributions	
0.5	13.12. 2013	Additional tables inserted, contents adjusted	F. Paul
0.7	22.08. 2013	Revised comments of TO amended	

The work described in this report was done under ESA contract 4000101778/10/I-AM. Responsibility for the contents resides with the authors that prepared it.

Author team:
Othmar Frey (Gamma), Frank Paul (GIUZ), Andreas Kääb, Christopher Nuth (GUIO), Thomas Nagler, Helmut Rott, Kilian Scharrer (Enveo), Andrew Shepherd, Francesca Ticconi (SEEL), Tazio Strozzi (Gamma)

Glaciers_cci Technical Officer at ESA:
Stephen Plummer

Table of Contents

Executive summary

This document is the deliverable D3.1 (System Prototype Description, SPD) of the Glaciers_cci project. The SPD provides a description of the processing part of the prototype systems for the three products glacier area, elevation change, and velocity. The focus is on a description of the current implementation of the system in a specific software and hardware environment. In general, the prototype systems described here are currently in use for product generation and follow closely the more theoretical and modular description of the processing workflow in the DPMv1. The document is written to be of practical use for system engineers rather than for the analyst generating the products (e.g. GUI related decisions are not shown).

1. Introduction

This document provides a description of the prototype production system for generation of the three data products:

- Glacier area
- Elevation change (from altimetry and DEM differencing)
- Velocity (from optical and microwave sensors).

According to the Statement of Work, the SPD should describe for each of these products:
- the components, functions, interfaces and modules that form the prototype ECV processing system
- the input and output dependencies
- the data flow between the processor components
- the data throughput
- the processing demand in terms of CPU load
- the processing volumes
- the time resources

The last five points provide a technical characterization of the system and are summarized for each product in a tabular form. All products are generated from specifically selected (individual) datasets on local computers having installed the software products required for the processing. The system as a whole has thus a rather low demand in terms of data storage requirements (gigabyte range), CPU load (minutes) and processing volumes (megabyte range). However, time resources are large for products requiring operator intervention, e.g. the manual correction of glacier outlines in the post-processing stage (this can take more than 90% of the total processing time). This expert work also drives the autonomy of the system, i.e. operator work (this must be a well-trained expert) is a key part of it. As the required work for the operator varies strongly with the region (e.g. the number of debris-covered glaciers with a difficult identification) and the experience of the operator, a-priori information on processing times are difficult to make. Examples are given based on the coding in a specific software. However, the software is not part of the system and software selection is on the analysts descretion. The required functionality is also available from several other software products, including public domain sources.

The entire prototype system includes more than the data processing described here. It also has a validation, dissemination and future developments (e.g. improved algorithms) component. All these components and their internal and external interfaces can currently only be described for the glacier area product (see SRD). For all products we here describe the main system architecture with its modules and the functionality of the prototype system. The latter is key to get the system working, as it relates the modules described from a system engineers perspective in the DPM to the Input/Output data described in the IODD and the practical implementation in a specific software and hardware environment. The SPD thus provides an example workflow for each product that can be practically followed. As the set-up for each of the products differs (e.g. in regard to the required operator interaction), there are also differences in the description of the prototype.

2. Glacier area

2.1 Prototype processing system

The prototype processing system as described in the following is one component of the overall system prototype (that also includes a validation, dissemination/archiving and a development component). The processing system is based on publicly available software scripts using long established procedures and guidelines to be interpreted by an operator. The individual processing steps are described in the DPMv1 (Glaciers_cci, 2013a) in a structured and modular way with the potential for implementation in various software products. In this document we describe one realization of the processing system for the glacier area product using a specific combination of software products. Scripts used in the modules are given in Appendix I.

2.1.1 Main system architecture modules and components

The general workflow of the processing system is shown in Fig. 2.1. It consists of five parts:

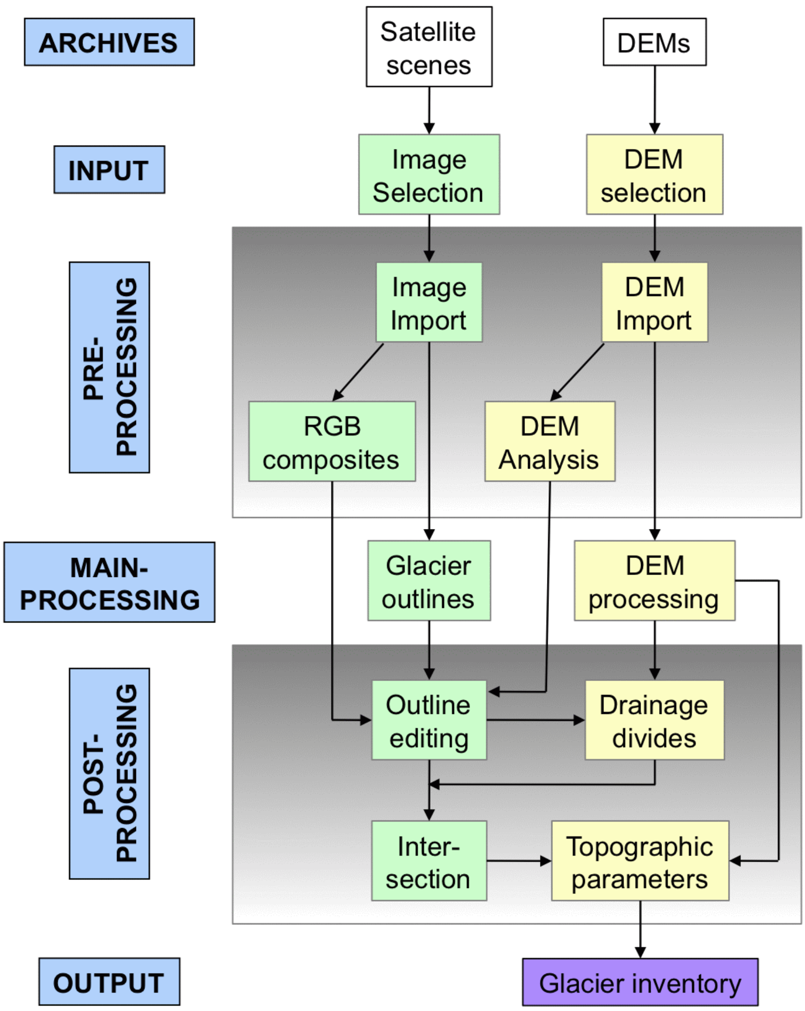

Fig. 2.1: The general data processing workflow for the glacier area product.

(1) Data input, (2) pre-, (3) main-, and (4) post-processing and (5) output generation. These steps are applied on two input data streams (satellite and DEM) using a mixture of (a) modules (that contain data processing scripts), (b) graphical user interfaces (to allow data manipulation and input by the analyst), and (c) a set of guidelines for (a) and (b) that support the analyst decisions (e.g. the best way to find the threshold selection for the ratio image used for glacier mapping). Product generation with the processing system requires to run (a) and (b) in a specific order of steps that are detailed in section 2.1.2 with reference to the terminology and module concept described in the IODD and DPM.

2.1.2 Functionality of prototype system

The processing steps performed for the glacier area product are described in the following. Steps (1a), (2a), (3a) and (4a) of the satellite data processing line are performed independent of the DEM processing line. Steps (1b), (2b) and (3b) of the DEM processing line are performed independent of the satellite data processing. Step (4b) needs the input from steps (3b) and (4a) to generate the output. Interaction by an operator (the analyst) is required for all steps. GAM1-4 refer to the Glacier Area Modules described in the DPM.

(1) Input

For the example discussed here, the EO data and DEMs required for product generation are available from the internet. The first step is the selection of an appropriate satellite scene from the archive (orthorectified L1T product from USGS) for the region of interest and download of the respective DEM tiles for the same region. The steps performed include:

(1a) Image selection
- find the region and browse through the available scenes
- select the best scene of all available scenes either based on the quicklooks or original resolution data. Criteria to be applied are: end of ablation period, no clouds hiding glaciers, no seasonal snow hiding the glacier perimeter, not too late in the year to avoid deep shadows
- if none of these criteria apply, chose the best scene available in regard to cloud free coverage. It might then be required to find additional scenes with locally better (e.g. snow) conditions to replace poor regions in the first choice
- downloaded the selected scene in the original format to the local disk

practical comments: the webpage usgs.glovis.gov used for download requires a Java applet, the scene selected for testing (from 7. Feb. 2003) has overall the best cloud conditions, but seasonal snow outside of glaciers is present in several regions; these regions have to be corrected using scenes with better snow conditions but less good cloud conditions.

(1b) DEM selection
- check which DEMs cover the region, go to their respective websites and download all required tiles along with any required meta-information
- in most cases the ASTER GDEM and the SRTM DEM might be available, but other sources should be checked as well (see DARD)

practical comments: the glacier-covered regions of South Georgia are included in only two tiles (S55W037/8); the also available ASTER GDEMII was not analysed

(2) Pre-processing

The pre-processing is converting the original image formats to the format of the software used and generates first products for later use.

(2a) Image import and creation of RGB composites
- convert original raster images (geotif) to the image format of the software used
- create contrast enhanced RGB composites with at least bands 3, 2, 1 and 5, 4, 3
practical comment: the second step is not really required but very practical.

(2b) DEM import
- convert the tiles to the format of the software used and mosaic them if required
- compute a hillshade from the available DEMs and if there are two, subtract them and colour code the difference for analysis
- decide which of the available DEMs is better suited for the selected region
- based on their specific advantages, it is possible that different DEMs are used for deriving drainage divides and topographic parameters
practical comment: the two SRTM tiles were mosaiced and analysis of the hillshade revealed a very good quality (in regard to artefacts and data voids)

(3) Main processing
(3a) GAM1 (outlines)
- apply GAM1 to calculate the ratio image using the pre-selected thresholds as a first approximation, spatial filtering and raster vector conversion
- visualize the results with GUI1 (e.g. a GUI of the GIS) and check if the threshold is ok, if not optimize and run the GAM1 module again
- chose the threshold in a way to minimize workload for post-processing, i.e. to get ice in shadow mapped as accurately as possible (debris has to be corrected anyway)
- convert the raster file resulting from the finally selected thresholds to a vector file after applying a 3 by 3 median filter for smoothing
- assign a new name to the vector file
practical comment: the two thresholds are 1.8 for the band3/band5 ratio and 95 for band 1.

(3b) GAM2 (DEM)
- apply GAM2 to calculate an aspect, slope, and the flowdirection grid from the DEM and a sine and cosine grid from the aspect grid

(4) Post-processing
(4a) Outline editing with GUI2
- correct the outlines (on-screen digitizing) in wrongly classified regions (water, debris, shadow, snow) using the RGB images created in step (2a)
practical comment: additionally, high-resolution images in Google Maps, a historic topographic map, and photographs available online were used to improve interpretation

(4b) Drainage divides from GAM3 (Divides) and topographic parameters from GAM4
- derive drainage divides from the edited outlines of step (4a) and the flow direction grid created in step (3b) using a script developed by Bolch et al. (2010)
- edit the resulting drainage divides with GUI3 (removing artifacts and basins without glaciers)

(4c) Derive topographic parameters for each glacier
- intersect the glacier outlines with the edited drainage divides using GAM4
- calculate topographic parameters for each glacier entity
- join the attribute tables to link the parameters back to the glaciers

(5) Output

- save the final file in shape file format

2.1.3 Processing software/tools used in the prototype

Generating the glacier area product from the raw satellite images and DEM data requires to operate on raster (e.g. satellite images, DEMs) and vector data (glacier outlines). These functionalities are provided by digital image processing software and Geographic Information Systems (GIS) that are both available as either commercial (COTS software) or public domain products. The following description of the prototype system for glacier area is based on the commercial image processing software ENVI (version 4.7) and ESRIs Arc/Info (v9.0) and ArcGIS (version 10.1) software. Furthermore, a short Fortran programme is used that is compiled with f90 (gnu) in a Unix environment. ENVI is only used for the pre-processing of the satellite images (creation of the RGB composites), while most of the other processing is done with the GIS in either command line mode (running the scripts of the four modules) or interactively using a graphical user interface (GUI) to visualize and edit the results. An internet browser (e.g. Safari) is used to select and download the satellite images and DEMs. Table 2.1 gives an overview of the software products used to perform the steps listed above. The processing prototype is illustrated on the example of a test region (South Georgia).

Step	Software	Details	Selection / *Commands*
1a	Safari 5.0.1	glovis.usgs.gov	Landsat 7 ETM+ (206-098 from 7. Feb. 2003)
1b	Safari 5.0.1	dds.cr.usgs.gov/srtm/version2_1/SRTM3/Islands	4 tiles: S55W036, S55W037, S55W038, S55W039
2a	Arc/Info 9.0 ENVI 4.7	AML script (map1) RGB image viewer	*imagegrid* *enhance zoom [square root], save image as*
2b	Arc/Info 9.0	AML script (dem1)	*imagegrid, con, setnull, hillshade*
3a	Arc/Info 9.0 ArcGIS 10.1	AML script (map1) ArcMap	*con, focalmedian, setnull, gridpoly* overlay of outlines (glmap) on RGB composite
3b	Arc/Info 9.0 ArcGis 10.1	AML script (dem2) ArcMap Toolbox (Spatial Analyst)	*slope, aspect, flowdirection, sin, cos*
4a	Arc/Info 9.0 ArcGIS 10.1	ArcEdit ArcMap	various (e.g. *add, select, reshape, delete, move*) vector editing tools
4b	ArcGIS 10.1	ArcMap Toolbox with Spatial Analyst	see separate processing script (Appendix)
4c	Arc/Info Fortran77	AML script (dem3) f90 compiler (aspect.f)	*zonalstats, joinitem* calculates mean aspect (degrees/sector)
5	Arc/Info 9.0	shapefile creation	*arcshape*

Table 2.1: Software used for the prototype processing of the glacier area system. The scripts in brackets (map1, dem1, dem2, dem3 and aspect.f) are compiled in Appendix I.

2.2 Input/output dependencies

The two input datasets required to generate the glacier area product are the satellite scene and a DEM. The DEM should cover the entire perimeter of the related satellite scene and should - at best - be acquired around the same year as the satellite image. This is in particular required for the rapidly changing topographic parameter minimum elevation. For the other parameters the changes are not too strong over a ten year time period for most glaciers. Most important, a detailed screening of the DEM using a hillshade version should reveal potential artifacts. They might have a much stronger influence on the results than the match of the DEM acquisition date.

The quality of the output also depends on a good coregistration of the DEM with the satellite image. This can be degraded when the DEM used for orthorectification differs from the DEM used for the topographic parameters. Geolocation mismatches as well as locally poor elevation data (e.g. poorly interpolated elevation values in the void-filled version of the SRTM DEM). Such problematic regions can only be identified by visual analysis (DEM hillshade and outline overlay) and might be difficult to overcome for individual glaciers. In this case it is recommended to exclude the affected glaciers from the calculation.

Required Input:
- Satellite image of the selected region (geotif)
- DEM covering the same region (geotif)

Output:
- Corrected glacier outlines with topographic parameters (shapefile)
- Uncorrected outlines (incl. lakes, excluding debris, shadow, etc.)
- Drainage basins (divides) for the entire region (shapefile)
- Various DEM-derived grids (e.g. slope, hillshade) (geotif)

2.3 Data flow within the prototype processor

The data flow between the processing modules is rather complex as shown in Fig. 3.2 of the DPMv1 (Glaciers_cci, 2013). For best results and fast processing, each step should be performed in the correct order. Once products are created, they should be converted to the GLIMS format as described in the PSD (Glaciers_cci, 2011) and uploaded to the GLIMS database (Raup et al., 2007). The prototype components not related to data processing are visualized in Fig. 3 of the SRD (Glaciers_cci, 2012).

2.4 Technical characterization of the system

In Table 2.2 typical technical characteristics of the production component of the system are given. They refer to the processing of one satellite scene and related DEM data. Roughly, the values scale linearly with the number of scenes processed. As the system has script-based and operator based processing parts, processing time values are given for both.

Step	Part	Value	Comments
Data throughput	Satellite	n/a	Driven by operator capacity and complexity of scene
	DEM	n/a	Driven by operator capacity and spatial resolution
CPU load	Satellite	1 h	Overall sum, often interrupted (Windows server)
	DEM	1/2 h	Overall sum, often interrupted (Windows server)
Processing volume	Satellite	< 1 GB	Landsat TM / ETM+ full scene
	DEM	< 1/4 GB	100 m resolution, (4 times more for 50 m)
Processing time	Satellite	10 min	*selection: 1/2 d, pre: 20', main: 20', post: 1 h - 1 week*
	DEM	10 min	*selection: 1/2 d, pre: 30', main: 10', post: 2 h - 2 days*

Table 2.2: Technical characteristics of the glacier area system (operator-based in italics).

3. Elevation changes from repeat altimetry

3.1 Prototype processing system

3.1.1 Main system architecture modules and components

The ECV production system for glacier elevation changes using Altimetry is shown schematically in Fig. 3.1.

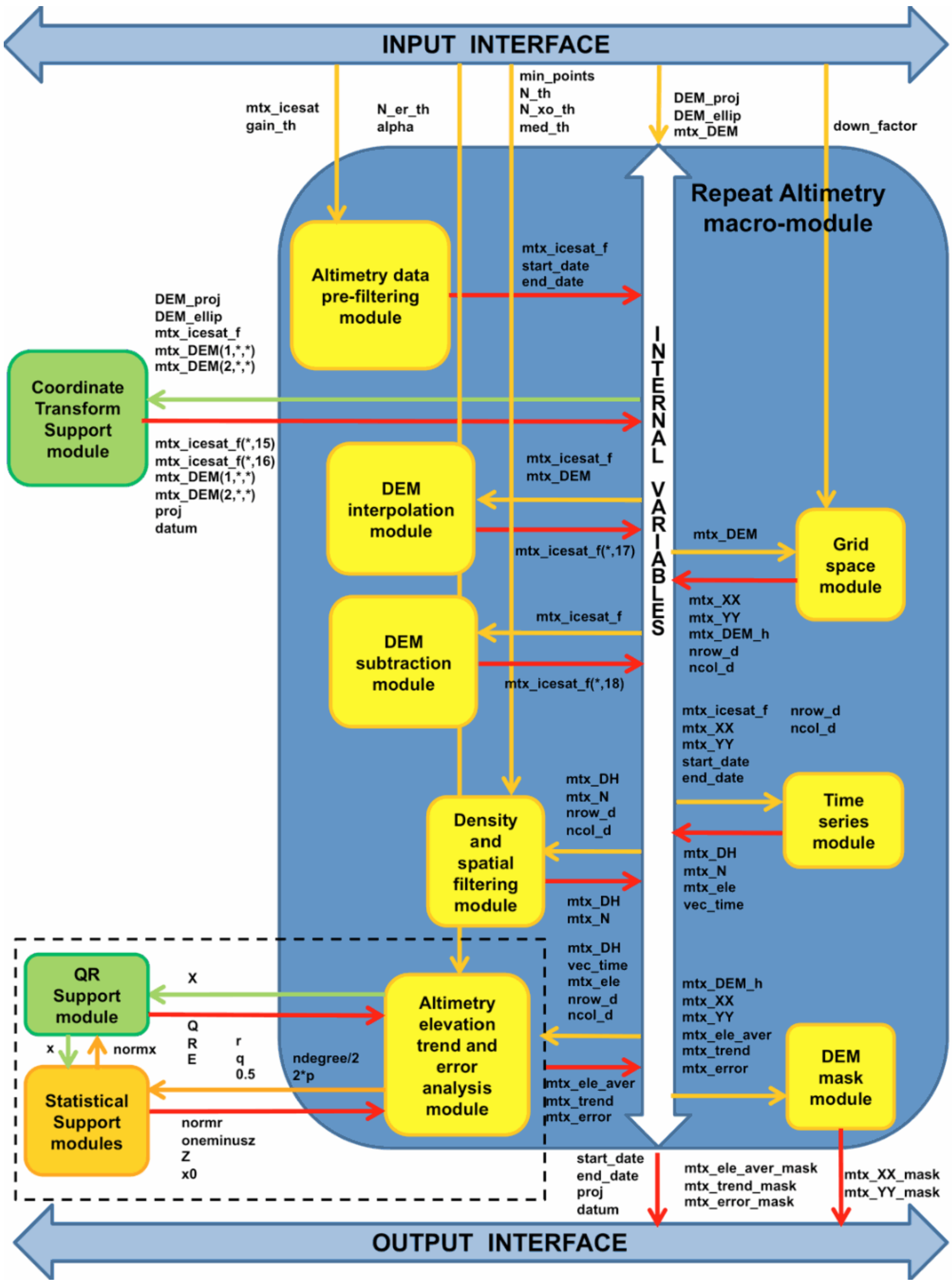

Fig. 3.1: Structure of the Repeat Altimetry macro-module.

It is represented as a macro-module which consists of the following modules:

- Altimetry data pre-filtering module;
- DEM interpolation module;
- Grid-space module;
- DEM subtraction module;
- Time series module;
- Density and spatial filtering module;
- Altimetry elevation, trend and error analysis module;
- Glacier mask and output format module.

The coordinate transformation module and the statistical modules constitute the support modules called by the macro-module. More details on the modules included in the Statistical Support modules block are shown in Fig. 3.2. For more details on each module refer to the DPMv1 (Glaciers_cci, 2013).

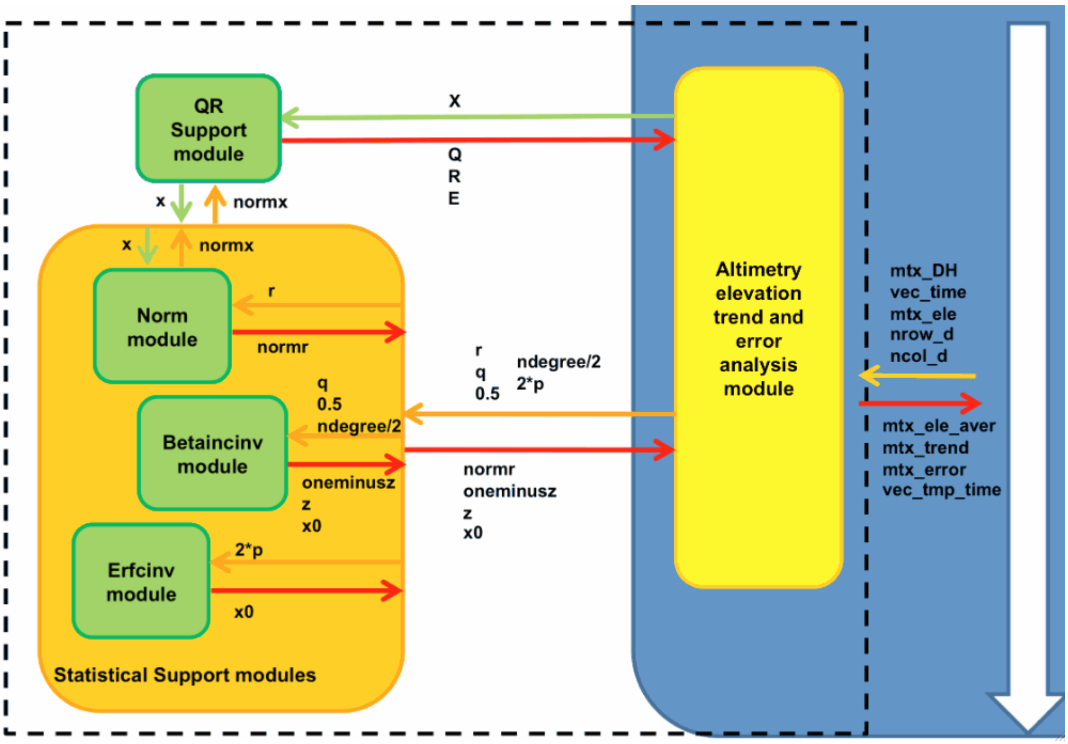

Fig. 3.2: Details on the Statistical Support modules block and its interaction with the Repeat Altimetry macro-module.

3.1.2 Functionality of prototype system

The Repeat Altimetry macro-module takes as user input the ICESat laser acquisitions, the DEM, a down scale factor for creating a grid space and other parameters needed for the density and spatial filtering of the data and for the determination of the confidence interval of the estimated elevation trend. More details are given in the IODDv1. It evaluates:

- the coordinates of the centre of each grid cell;

- the average of the elevation obtained from the measurements falling in each grid cell and over the whole epochs used for the estimation of the evaluation trend;
- the elevation change;
- the error.

The elevation change macro-module using Altimetry begins with an input interface module for the selection of the desired ICESat level 1B GLAS06 products (release v33) covering the user's area of interest and for the selection of a DEM relative to the same area.

As described in the IODDv1, the data which are delivered as scaled integer binary files need to be converted into the format specified in the IODDv1. The elevation provided by the ICESat level 1B GLAS06 product refers to the same ellipsoid as TOPEX/Poseidon and Jason-1. Hence, although the horizontal displacement caused by different ellipsoids is well below the GLAS accuracy in horizontal geolocation so that it can be ignored, the adjustment to elevation to account for different ellipsoids used with non-GLAS data (i.e. GPS, airborne LIDAR surveys, etc.) need to be ascertained before comparing those data to GLAS elevations.

After this pre-processing step on the ICESat data, the macro-module starts with a pre-filtering of the ICESat data, where measurements not suitable for the estimation of elevation changes are filtered out. Then, the macro-module uses a support module for checking the coordinate reference systems used by the two main datasets, i.e. the ICESat data and the DEM elevation. In case it is necessary, the coordinates will be transformed to a common reference system. After these processing, the elevations of the DEM are interpolated at the locations identified by the laser footprints, a 2-dimentional grid space for creating three-dimensional representations of data is created by simply down-sampling the DEM by the down-factor, which can be introduced as input or be fixed as an internal constant. The DEM interpolated elevations are then subtracted to the ICESat measurements. The macro-module then includes the creation of a time series which defines the time steps or seasons present within the ICESat full operational period. This module creates 3D arrays for representing the measured elevations and the difference between the latter and the DEM interpolated elevations in a grid format, where the third dimension represents the time steps or seasons defined. The main processing follows with the application of density and spatial filters and with the computation, for each grid cell, of the interpolation line for the determination of the elevation trend. Also the confidence interval associate with this estimation is computed and considered as a measure of the trend error. Finally, a glacier mask is applied for filtering out the acquisition outside the glacier region.

The order in which each module has just been mentioned follows the time sequence that can be seen in Fig. 3.1. It is given by the access of each module to the internal memory and from the consequent variable flow. Consequently the time order goes from the top to the bottom. The post-processing includes the preparation of the output data in the format specified in the IODDv1.

3.1.3 Processing software/tools used in the prototype

Pre-processing step

In the case of the repeat altimetry macro-module, the pre-processing step is applied to the input and auxiliary data to allow a correct representation of the laser altimetry data for the main processor (see Fig. 3.1). This step converts the ICESat level 1B GLAS06 scaled integer binary files into the format specified in the IODDv1. It is required to run the NGAT tool available at http://nsidc.org/data/icesat/tools.html for this purpose. For running the NGAT executable file, IDL (Interactive Data Language) Virtual Machine was installed. Furthermore, a batch file is used as an input to the NGAT tool. This file contains the initial parameter settings for the correct extraction of the information mentioned above (see IODDv1).

When only working with relative changes, a conversion of the ICESat elevation values (level 1B GLAS06 product) from the TOPEX/Poseidon and Jason-1 ellipsoid to another reference system is not required and was thus not applied here (otherwise a tool is available from NSIDC: ftp://sidads.colorado.edu/pub/DATASETS/icesat/tools/idl/ellipsoid). Finally, the imported auxiliary dataset (the DEM) was converted to UTM projection.

Main processing
The main processing is performed by a Matlab program which calls various functions and subroutines written in the same language.

3.2 Input/output dependencies

The Matlab program requires the ICESat/GLAS and DEM data in the format described in the IODDv1 (Glaciers_cci, 2013b) and an ASCII file containing other information.

Input:
- an ASCII file derived from the ICESat/GLAS data (directory path)
- the DEM in the format specified in the IODDv1 (directory path)
- a file with the other necessary parameters (directory path)

Output:
- a file containing the result (directory path).

3.3 Data flow within the prototype processor

The data flow for this product is very simple. After the input data have been properly formated in a pre-processing step, the main program performs all calculations and generates the output. Further user interaction is not required.

3.4 Technical characterisation of the system

The computational demands are very low (local machine), but software needs to be installed.

Step	Part	Value	Comments
Data throughput	Satellite DEM	n/a n/a	Not relevant Not relevant
CPU load	Satellite	n/a	Processing takes place on a local machine (dual core)
Processing volume	Satellite DEM	90 MB 2 MB	includes all points (604472) for Devon Island for Devon Island (depends on size of the region)
Processing time	Pre Main	30 min 10 min	running NGAT and the DEM conversion tool for Devon Ice cap (depending on the number of data points)

Table 3.1: Technical characteristics of the system for elevation change from altimetry.

4. Elevation changes from DEM differencing

4.1 Prototype processing system

Two prototype processing systems for the product glacier elevation changes from DEM differencing are currently implemented for data generation. After initial resampling and co-registration between the DEMs is performed, the result is an elevation difference matrix between two DEMs. The validation component in the system is part of the post-processing and is difficult to provide routinely mainly due to the lack of reference data sets, but also due to the seasonal and long-term variability of glacier surface elevations. Therefore, validation relies on internal verification over glacier-free terrain (assumed to be stable). While the entire production line can largely be automated, manual inspection is recommended at several stages of the processing to ensure meaningful and unbiased results.

4.1.1 Main system architecture modules and components

The general architecture of the main processing system consists of five parts: (1) Data input, (2) pre-, (3) main-, and (4) post-processing and (5) output generation (Fig. 4.1). Data input can be performed automatically or manually and should abide by the rules provided in section 4.2. The pre-processing module performs the import routines, resampling and co-registration. This includes the import of glacier masks and determination of glacier/non-glacier pixels for processing. The main processing applies the co-registration parameters, resamples the DEMs and derives the difference matrix. The post processing modules derives the quality indicator masks and the quality report based upon the non-glacier pixels.

An initial ‘basic’ system tool is provided as a manual implementation of the Co-Registration Module (Fig. 4.1) in Excel allowing individual users to apply their software of preference for the resampling and differencing. The second prototype system is scripting based and operates automatically with input of the DEMs and a glacier shapefile.

4.1.2 Functionality of prototype system

The prototype system has been developed under the assumption of globally available DEMs derived from very different sources and thus containing a mixture of accuracies and errors specifically dependent upon the DEMs. Thus the design of the system is functional for any DEM input and for comparisons between DEMs of different resolutions. A detailed description for all modules in Fig. 4.1 is provided in the DPM (Glaciers_cci, 2013a). The entire processing chain is fully automated without the need for user-interaction until the final validation of the generated elevation changes. Resampling is determined by the pixel sizes of the DEMs such that resampling always occurs to the largest pixel size to limit downsampling artifacts. The processing scheme provided so far involves the first order corrections required to derive DEM differencing, namely co-registration and re-sampling.

A second less automated prototype has been developed that only solves the main algorithms chosen for co-registration provided an input sample of elevation differences, slopes and aspects on stable terrain. This manual prototype is more flexible in that it allows the user to perform all the resampling and selection of stable terrain themselves using their preferred methods and software. This software is available upon request and provides more insight into the algorithms implemented in the Co-Registration Module.

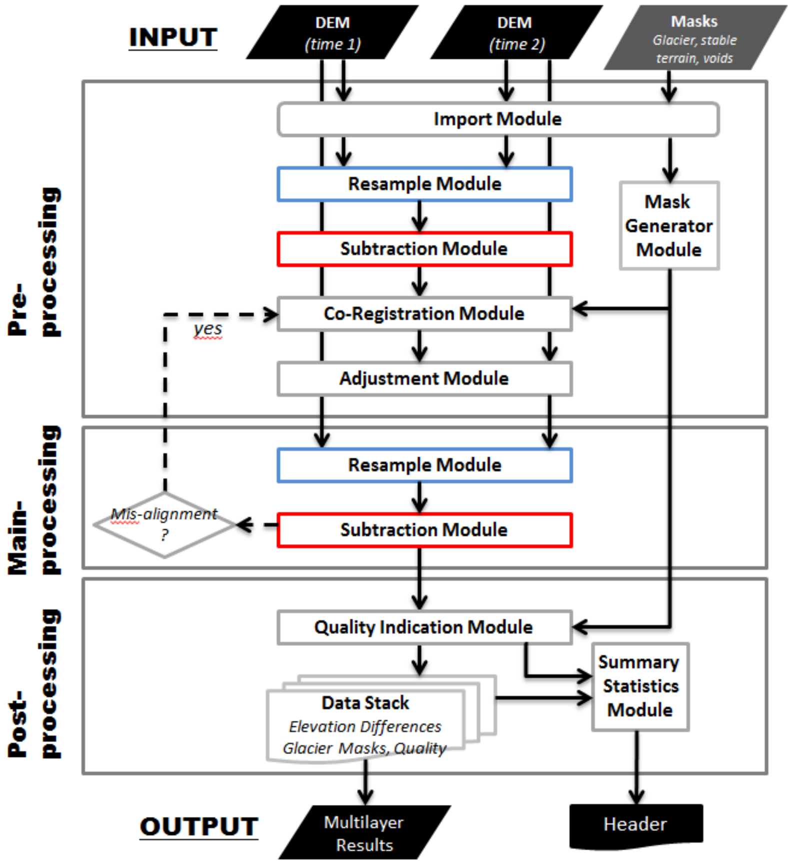

Fig. 4.1: Structure of the glacier elevation change by DEM differencing macro module.

4.1.3 Processing software/tools used in the prototype

For the basic protoype, the algorithm is implemented in an Excel spreadsheet (DEM_Co-registration_tool.xls). The Excel spreadsheet requires that the user installs an additional macro called "Solver" that is supplied by default with the software but requires activation. Instructions for the installation and operational procedures of the Excel spreadsheet are included with the tool, and is shown in Fig 4.2. This system tool requires the user to input manually (or import) into the spreadsheet a sample of non-glacier elevation differences, terrain slope and terrain aspect. These can be generated in the preferred software of the user. The user must then run 'Solver' in Excel following the instructions provided to solve for the co-registration parameters. The adjustment parameters are the output that the user then must apply manually in their preferred software. This is therefore a light processing system providing more flexibility for individual research and development.

The fully automated prototype system from the DEM differencing macro module (Fig. 4.1) is currently performed within Matlab software and operated by running one main script (*Main_DEMdifference.m*). A number of separate Matlab Toolbox licences are required to operate the prototype including the Mapping Toolbox, Statistics Toolbox, Curve Fitting Toolbox, and (optionally) the Distributed Computing Toolbox and Parallel Processing Toolbox. Furthermore, the mask generator module uses C/C++ scripting accessible through Matlab excutables (MEX) in order to improve the efficiency of the algorithm. For this purpose a C compiler must be installed (e.g. http://www.mathworks.se/support/compilers/R2012b/win64.html). The mask generator module is called *insidepoly.m* and uses open source code that is freely downloadable (http://www.mathworks.com/matlabcentral/fileexchange /27840-2d-polygon-interiordetection/content/InsidePolyFolder/insidepoly.m).

The file '*insidepoly_install.m*' must be run before operating *Main_DEMdifference.m* in order to produce the compatible Matlab – C interchange (*insidepoly_dblengine.c*) and this file must be placed in the same folder as the other scripts. An error will be returned if any of the toolboxes are not licenced, compilers are not installed or if Matlab is not able to find the paths for these scripts.

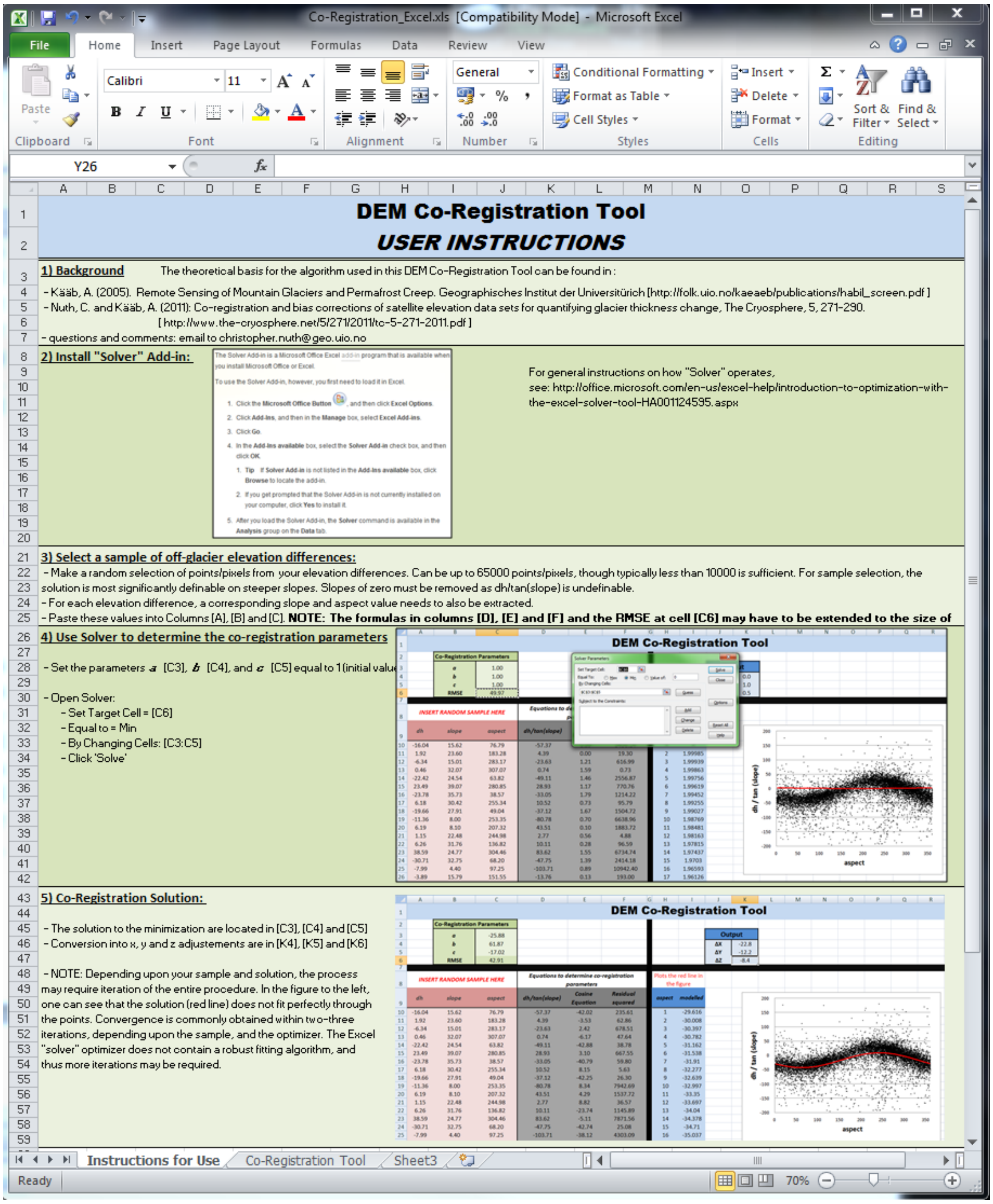

Fig. 4.2: Example of instructions for using the basic prototype version implemented in Excel [note: not fully automated implementation].

4.2 Input/output dependencies

The required input data for the elevation change from DEM differencing macro module are two multi-temporal DEMs in geotif format, and a shapefile for denoting glaciers. The DEMs should contain a long-enough time span between them such that elevation changes are larger than the associated errors of the combined DEMs. This in particular is site dependent, and as a general rule of thumb, a time span of five years or more should be considered. We limit the input data to automatically generated DEMs though in principle any DEM may be used within this processing scheme. The output quality indicators are dependent upon whether quality indicators are provided for the individual DEMs. For example, SPOT DEMs are provided with a quality mask representing the correlation between the stereo image pairs, but products from ASTER or the SRTM C-Band do not provide such information. Therefore, the prototype system provided here does not include optional correlation layers for the DEMs, and if they are to be used, the DEMs must be filtered using the correlation layers before input into the prototype system.

Required Input:
- Input directory path and file name of DEM from time one and two
- Input directory path and file name of shapefile that denotes glaciers

Output:
- Difference grid
- Quality mask
- Boolean glacier mask
- Table providing statistical results on and off glaciers.

4.3 Data flow within the prototype processor

Currently, the fully automated prototype processor, as seen from the operator, consists of one single Matlab script (*Main_DEMdifference.m*) that handles the complete processing sequence. The additional prototype system (an Excel sheet called *DEM_Co-registration_tool.xls*) is dependent upon user input to generate the processing sequence described in the instructions and run the system manually.

4.4 Technical characterisation of the system

The values below refer to the processing of two ASTER DEMs and the fully automatic prototype processor. Pure processing values scale about linearly, but operator based interactions have to be added.

Step	Part	Value	Comments
Data throughput	DEM	24 MB / 2 min	for two ASTER DEMs
CPU load	DEM	n/a	processing on a local machine (quad core i7CPU@2.2GHz)
Processing volume	DEM	< 100 MB	for two ASTER DEMs
Processing time	Pre Main Post	80 sec 40 sec 5 sec	without operator interaction for two ASTER DEMs without operator interaction

Table 4.1: Technical characteristics for elevation change from DEM differencing.

5. Glacier velocity

5.1 Prototype processing system

5.1.1 Main system architecture modules and components

The data processing system for glacier velocity consists of two separate production lines, the first using SAR data and the second using optical data as the source to determine displacement offsets by means of image matching of multitemporal acquisitions. In Figs. 5.1 and 5.2 the elements of the glacier velocity macro-module from SAR and optical data are given, respectively. Validation of glacier displacements from space is a difficult task due to (i) a lack of reference data sets, (ii) high temporal, and (iii) high spatial variability of the displacements. Therefore, validation of the displacement product has to be performed as a part of the processing system using quality parameters such as correlation coefficients and/or signal-to-noise ratios (SNR).

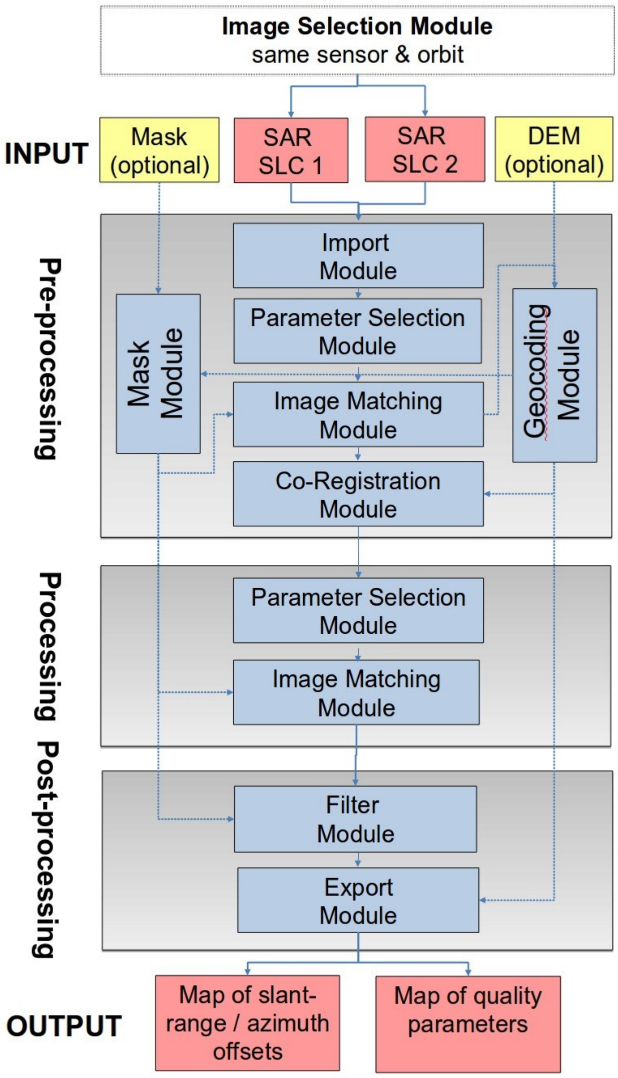

Fig. 5.1: Structure of the glacier velocity from radar data macro-module.

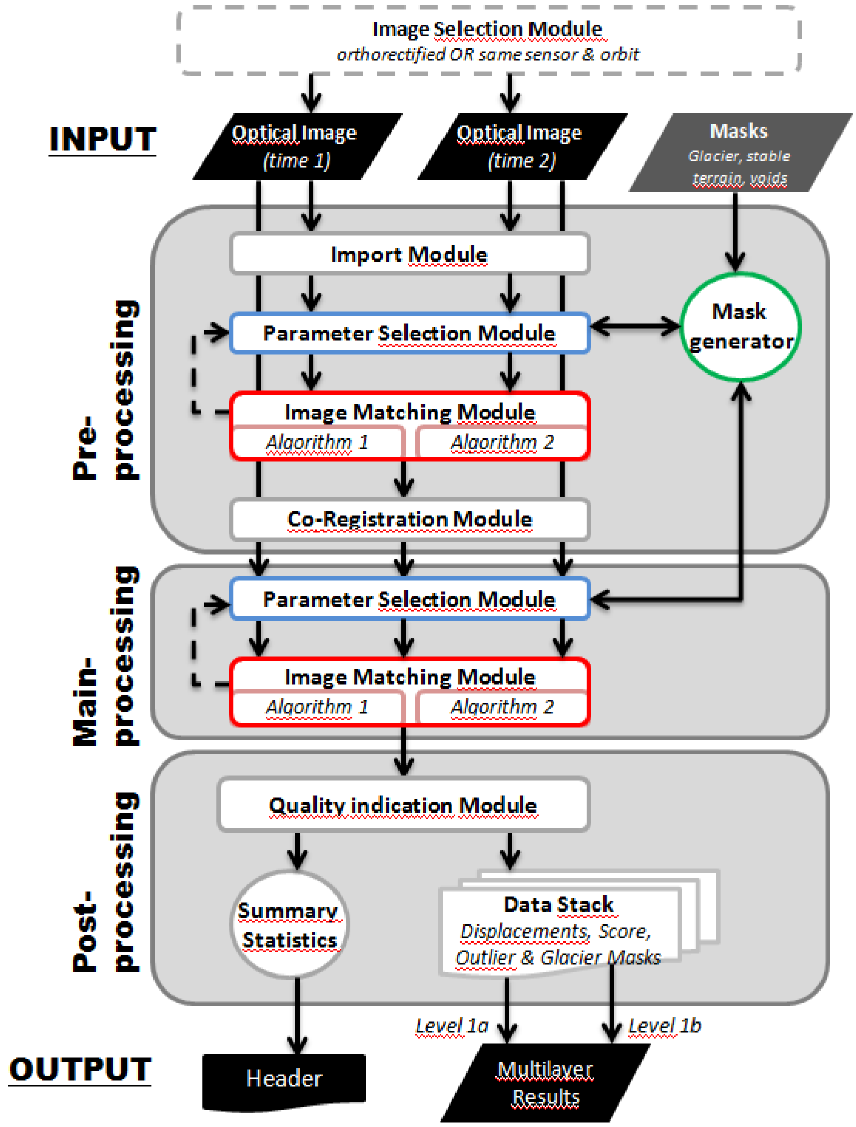

Fig. 5.2: Structure of the glacier velocity from optical data macro-module.

5.1.2 Functionality of prototype system

Displacements from SAR data

The ice-velocity macro-module for SAR images begins with an optional module for selecting suitable images, from for example ENVISAT ASAR, ALOS PALSAR or TerraSAR-X. The required input data are two multi-temporal images with associated headers, and optionally a stack of shapefiles for denoting glaciers and terrain assumed to be stable (e.g. no ocean, lakes, glaciers) and a DEM. The pre-processing entails data import, coarse scale matching procedures, co-registration of the two images as well as generating the optional binary masks or the optional geocoding. The main processing includes the multi-algorithm fine scale matching procedures, if selected only over glaciers. The post-processing includes the preparation and adjustment of the displacements, outlier filtering and preparation of the quality indicator(s) mask, summary statistics off/on glaciers, and data export. Output data are level 1a (raw) and 1b (filetered) multilayer data (geotif) and header information.

Displacements from optical data

The ice-velocity macro-module requires two multi-temporal optical images as input data and optionally a shapefile for denoting glaciers. Determination of the window sizes that the algorithm will use to match the multi-temporal images and the matching grid can be performed manually or (semi-)automatically. Alternatively to the matching grid, a list of locations to be matched can be imported (e.g. profiles). These parameters are dependent upon the image resolution and also algorithm dependent and multiple parameters may be required. The module has two processing modes: pre- and main-processing. At the pre-processing stage, the window sizes and grids may be coarser than within the main-processing mode. In the latter, the co-registration displacements are taken into consideration and passed through also to the image processing module. The post-processing includes the preparation and adjustment of the displacements, outlier filtering and preparation of the quality indicator(s) mask, summary statistics off/on glaciers, and data export. Output data include level 1-raw and 1-filtered multilayer data and header information.

5.1.3 Processing software/tools used in the prototype

Displacements from SAR data

The pre-processing, main processing, and the post-processing steps (see Fig. 5.1) of the glacier velocity from radar data macro module is currently performed by means of one single Unix csh shell script *run_offset* that makes use of various programs from different GAMMA software packages (Release Dec. 2011, or newer):

- GAMMA Differential Interferometry and Geocoding Software (DIFF&GEO)
- GAMMA Interferometric SAR Processor (ISP)
- GAMMA Display Software (DISP)
- GAMMA Interferometric Point Target Analysis Software (IPTA)
- GAMMA Land Application Tools (LAT).

In particular, the following command line tools provided as part of the GAMMA software are used; these programs need to be installed on the system, however, they are executed automatically from within the Unix csh shell script *run_offset* with the respective parameters assigned (no user interaction required):

- create_offset (ISP),
- init_offset_orbit (ISP)
- offset_pwr (ISP)
- offset_fit (ISP)
- offset_pwr_tracking (ISP)
- base_calc (DIFF)
- multi_look (ISP)
- multi_real (ISP)
- raspwr (DISP)
- cpx_to_real (DISP)
- rashgt (DISP)
- lin_comb (LAT)
- rasdt_pwr24 (DISP)
- average_filter (LAT)
- ratio (LAT)
- single_class_mapping (LAT)
- mask_class (LAT)
- replace_values (DISP)

- cc_wave (ISP)
- ras8_float (DISP)
- dis2ras (DISP)
- interp_ad (ISP)
- image2pt (IPTA)
- d2pt (IPTA)
- pt2d (IPTA)
- dis2hgt (DISP)
- gc_map (DIFF),
- geocode (DIFF),
- create_diff_par (DIFF),
- offset_pwrm (DIFF),
- offset_fitm (DIFF),
- gc_map_fine (DIFF),
- geocode_back (DIFF)
- rasshd (DISP)
- comb_hsi (LAT)
- data2geotiff (DISP)

Different processing steps performed within the script *run_offset*:
- Read data
- Initialize offset values
- Precise offset estimation
- Offset tracking
- Compute time interval
- Create power image
- Display results
- Noise filtering
- Filtering of azimuth streaks
- Geocode
- Prepare output files (geotif format)

Displacements from optical data

The prototype system for optical velocity is an IDL implementation and a fully automatic software with built-in GUI's that lead the operator through the entire processing procedure (Fig. 5.2). The prototype is executed by running the script '*cias.sav*' at the terminal window after opening IDL (typing CIAS and enter). Alternatively, if IDL-Virtual Machine is installed, a simple double-click on *cias.sav* file will run it. The software then opens through GUIs and asks the user for input data. The user can proceed with or without co-registration, and continue with image matching of selected points or point grids as derived from polygons imported by the user through a GUI. The input parameters (e.g. matching and search window sizes) are manually inserted by the user. After processing, the user can save the results as a table containing the displacements along with quality indicators for each point position.

A variation of this full prototype is a prototype that only implements one rather than two algorithms at the present time, and that does not provide a filtered output but rather provides the quality indicators. The user can then post-process them using their preferred methods. Furthermore, output is in table format rather geotif as in the full prototype.

5.2 Input/output dependencies

The required input data for the glacier velocity macro-module are two multi-temporal SAR or optical images with associated headers, and optionally a stack of shapefiles for denoting glaciers and terrain assumed to be stable (i.e. no ocean, lakes, glaciers etc.) and for SAR also a DEM. For best matching, the two SAR images have to be from the same sensor and orbit. Although image-matching is performed with intensity images and thus SAR data as multi-look precision images might be also considered, we limit the input products of the processing system to single look complex (SLC) digital images generated from raw SAR data using up-to-date auxiliary parameters. For optical matching, fully orthorectified images will be used as input (i.e. Landsat panchromatic) removing the requirement of a DEM. However, quality of the offsets is highly dependent upon the temporal consistency of the DEM used for orthorectification of the respective satellite scenes.

Displacements from SAR data

Required input:

- A string indicating the acquisition dates of the SLC pairs which are used for offset estimation according to the following naming convention: YYYYMMDD_YYYYMMDD (e.g.: 20070705_20070820)
- First and second single-look complex image (directory path)
- The two SLCs indicated by the identifier YYYYMMDD_YYYYMMDD, as well as the two associated GAMMA SLC parameter files must reside in the respective SLC directory
- Path to output directory where the results will be written to
- Path to directory where the DEM and its associated parameter file reside. The parameter file must conform to the GAMMA dem_par format

Output:

- slant-range offset (geotif)
- azimuth offset (geotif)
- SNR (geotif)

Displacements from optical data

Required input:

- First and second optical image (directory path and filename) as geotif or tif world (.tfw).
- Path and filename to a directory of input positions where image matching should be performed. [Note this varyies from the DPM in that the glacier shapefile needs to be converted into a list of locations that are to be matched.] Alternatively, the user can select point or polygon with a GUI within the system.
- Path to output directory where the results will be written to

Output:

- Offsets from Method 1
- Quality indication mask – correlation value
- Quality indication mask – correlation basement value

5.3 Data flow within the prototype processor

5.3.1 Displacements from SAR data

Currently, the prototype software, as seen from the operator, consists of one single Unix csh shell script (*run_offset*) that handles the complete processing sequence.

5.3.2 Displacements from optical data

Currently, the prototype software, as seen from the operator, consists of one single IDL script (*cias.sav*) that handles the complete processing sequence using user input through GUIs.

5.4 Technical characterization of the system

There are some differences when the product is derived from optical versus microwave data, and processing time is an issue for the former. All calculations are performed on local computers (Desktop).

Step	Part	Value	Comments
Data throughput	SAR Optical	~ 4 GB / h ~ 0.5 GB / 15 h	
CPU load	SAR Optical	n/a n/a	Code not for parallel processing Always on 1 CPU
Processing volume	SAR Optical	2 GB + 30 MB 0.5 GB	2 concatenated ASAR frames 2 Landsat ETM+ pan scenes
Processing time	SAR Optical	<1 h 15 h	on octa-core processor i7 CPU 950 @ 3.07GHz on 1 CPU

Table 5.1: Technical characteristics for glacier velocity.

References

Bolch, T., Menounos, B. and Wheate, R.D. (2010): Landsat-based inventory of glaciers in western Canada, 1985 - 2005. Remote Sensing of Environment, 114, 127-137.

Glaciers_cci (2011a): Product Specifications Document (PSD): Prepared by the Glaciers_cci consortium, 43 pp.

Glaciers_cci (2011b): Data Access Requirements Document (DARD): Prepared by the Glaciers_cci consortium, 46 pp.

Glaciers_cci (2012): System Requirements Document (SRD). Prepared by the Glaciers_cci consortium, 32 pp.

Glaciers_cci (2013a): Detailed Processing Model (DPM): Prepared by the Glaciers_cci consortium, 75 pp.

Glaciers_cci (2013b): Input Output Data Definition (IODD)): Prepared by the Glaciers_cci consortium, 25 pp.

Paul, F., Barry, R., Cogley, G., Frey, H., Haeberli, W., Ohmura, A., Ommanney, S., Raup, B., Rivera, A., Zemp, M. (2009): Recommendations for the compilation of glacier inventory data from digital sources. Annals of Glaciology, 50 (53), 119-126.

Raup, B. and Khalsa, S.J.S. (2007): GLIMS Analysis Tutorial, vers. 22/05/2007, Boulder, 15 pp. Online: http://glims.org/MapsAndDocs/assets/.

Abbreviations

ALOS	Advanced Land Observing Satellite
AML	Arc Macro Language
ASAR	Advanced Synthetic Aperture Radar
ASTER	Advanced Spaceborne Thermal Emission and Reflection radiometer
CCI	Climate Change Initiative
COTS	Commercial Of The Shelf
CPU	Central Processing Unit
DARD	Data Access Requirements Document
DEM	Digital Elevation Model
ECV	Essential Climate Variable
ERS	European Remote Sensing Satellite
ESA	European Space Agency
ESRI	Environmental Systems Research Institute
ETM+	Enhanced Thematic Mapper plus
GAM	Glacier Area Module
GDEM	Global DEM
GIS	Geographic Information System
GLAS	Geoscience Laser Altimeter System
GLIMS	Global Land Ice Measurements from Space
GPS	Global Positioning System
GUI	Graphical User Interface
ICESat	Ice, Cloud, and Elevation Satellite
IDL	Interactive Data Language
IODD	Input Output Data Definition
NGAT	NSIDC GLAS Altimetry elevation extractor Tool
PALSAR	Phased Array type L-band SAR
PSD	Product Specifications Document
RGB	Red Green Blue
SAR	Synthetic Aperture Radar
SLC	Single Look Complex
SNR	Signal to Noise Ratio
SPOT	System Pour l'Observation de la Terre
SRTM	Shuttle Radar Topography Mission
TM	Thematic Mapper
WGMS	World Glacier Monitoring Service

Appendix (Scripts)

Processing scripts used (for the area product)

(1) map 1

```
imagegrid b1.tif k1gr
imagegrid b3.tif k3gr
imagegrid b5.tif k5gr
grid
t = con((float(k3gr) / k5gr) > 1.8, 0, 255)
tt = con(t == 0 & k1gr > 95, 0, 255)
tmed = focalmedian(tt)
ttt = setnull(tmed == 255, 0)
q
gridpoly ttt glmap
```

(2) dem1

```
imagegrid S55E036.bil gr1
grid
gr2 = con(gr1 >= 32768, gr1 - 65536, gr1)
N33E076gr2 = setnull(gr2 == -32768, gr2)
kill gr1 all
kill gr2 all
q
```

(3) dem2

```
dem_slp = slope(DEM)
dem_asp = aspect(DEM)
dem_fld = flowdirection(DEM)
singr = sin(dem_asp div deg)
cosgr = cos(dem_asp div deg)
```

(3) dem3

```
intersect gl_outl basin gl_int

polygrid gl_int gl_label gl_int-id
25
n
310450, 6756075
8084, 7183

polygrid gl_int gl_0gr grid-code
25
n
310450, 6756075
8084, 7183

grid
gl_lab = gl_label + gl_0gr

gl_elev.stat = zonalstats(gl_lab, dem, all)
gl_slp.stat = zonalstats(gl06_lab, dem_slp, mean)
gl_sin.stat = zonalstats(gl06_lab, singr, mean)
gl_cos.stat = zonalstats(gl06_lab, cosgr, mean)
q

tables
sel gl_slp.stat
alter mean slp_mean
~
~
~
~
sel gl_sin.stat
unload sinstat.dat value mean columnar col
sel gl_cos.stat
unload cosstat.dat value mean columnar col
q
```

```
&system mean_asp.out

tables
define gl_asp.stat
value 4 10 b
asp_deg 4 8 f 1
asp_sec 4 4 I
~
sel gl_asp.stat
add from degstat.dat

sel gl_int.pat
dropitem gl_int.pat gl_int# gl_int-ID BASIN#
alter basin-ID value
~
~
~
calc value = gl_int-ID

sel gl_elev.stat
dropitem gl_elev.stat area range sum variety MAJORITY MINORITY
q

joinitem gl_int.pat gl06elev.stat gl_int.pat value
joinitem gl_int.pat gl06slp.stat gl_int.pat value
joinitem gl_int.pat gl06asp.stat gl_int.pat value
```

(4) aspect.f

```
        program

C       *** VARIABLES ***
        IMPLICIT none
        INTEGER i,v,s
        REAL f,x,y,d,n,w(2100)
        f=180./(4.*ATAN(1.))

C       *** OPEN FILES ***
        OPEN (51,file='sinstat.dat',status='old')
        OPEN (52,file='cosstat.dat',status='old')
        OPEN (56,file='degstat.dat')
        OPEN (57,file='secstat.dat')

C       *** READ INPUT AND WRITE OUTPUT ***
        DO 11 i=1,2057
           READ(51,*,end=11) v,x
           READ(52,*,end=11) v,y
           n=sqrt(x*x+y*y)
           d=MOD(360.+(ATAN2(x,y))*f,360.)
           s=(MOD(NINT(d/45),8))+1
           WRITE(56,101) v,d,s
           w(i)=d
C          WRITE (57,103) x/n,y/n
11      CONTINUE

        WRITE (57,102) (w(i),i=1,2057)

101     FORMAT(I5,F8.1,I3)
102     FORMAT(F7.1)
103     FORMAT(2F9.4)
        STOP
        END
```