Documentation – User's Guide

Interferometric SAR Processor - ISP

GAMMA REMOTE SENSING

Version 1.8 – June 2014

GAMMA *Remote Sensing AG, Worbstrasse 225, CH-3073 Gümligen, Switzerland*
tel: +41-31-951 70 05, fax: +41-31-951 70 08, email: gamma@gamma-rs.ch

Table of contents

List of acronyms

ALOS	Advanced Land Observing Satellite
AP	Alternating Polarization
ASAR	Advanced Synthetic Aperture Radar
ASF	Alaska SAR Facility
ASI	Agenzia Spaziale Italiana
CEOS	Committee on Earth Observation Satellites
CCRS	Canadian Centre for Remote Sensing
COSMO-SkyMed	Constellation of Small satellites for Mediterranean Basin Observation
DEM	Digital Elevation Model
DEOS	Department of Earth Observation and Space Systems
DIFF&GEO	Differential Interferometry and Geocoding Software
DISP	Display Tools
DLR	Deutsches Luft- und Raumfahrtzentrum
ENVISAT	ENVIronmental SATellite
EORC	Earth Observation Research Centre
ERS	European Remote Sensing (Satellite)
ERSDAC	Earth Remote Sensing Data Analysis Centre
ESA	European Space Agency
ESRIN	European Space Research Institute
FFT	Fast Fourier Transform
GCP	Ground Control Points
IECAS	Institute Of Electronics Chinese Academy Of Scences
IM	Image Mode
ISP	Interferometric SAR Processor
JAXA	Japanese Aerospace Exploration Agency
JERS	Japanese Earth Resources Satellite
JPL	Jet Propulsion Laboratory
KC	Kyoto and Carbon
MCF	Minimum Cost Flow
MLI	Multi-look Intensity
MSP	Modular SAR Processor
PAF	Processing and Archiving Facility (D = Germany I = Italy, UK =United Kingdom)
PALSAR	Phased Array L-band Synthetic Aperture Radar
PRI	Precision Image
RISAT	Radar Imaging Satellite
RSI	Radarsat International
SAR	Synthetic Aperture Radar
SCS	Single-Look Complex Slant
SIR	Shuttle Imaging Radar
SLC	Single Look Complex
SNR	Signal to Noise Ration
SRTM	Shuttle Radar Topography Mission
TanDEM-X	TerraSAR-X-Add-on for Digital Elevation Measurements
TCN	Track-Cross Track-Normal (reference system)
TIN	Triangular Irregular Network
UAVSAR	Uninhabited Aerial Vehicle Synthetic Aperture Radar
USGS	United States Geological Survey

1. Introduction

The Gamma Interferometric SAR Processor (ISP) encompasses a full range of algorithms required for generation of interferograms, height maps, coherence maps, and differential interferometric products. These steps include baseline estimation from orbit data, precision registration of interferometric image pairs, interferogram generation (including common spectral band filtering), estimation of interferometric correlation, removal of curved Earth phase trend, adaptive filtering of interferograms, phase unwrapping, precision estimation of interferometric baselines from ground control points, generation of topographic height, image rectification and interpolation of interferometric height maps. In addition, the ISP offers programs for the radiometric calibration of the SLC and for the conversion between slant-range and ground-range geometry.

In addition the ISP supports the processing of PRI/MLI data, expressing the magnitude information of a SAR image. PRI, which stands for Precision Image, images are obtained from multi-looking sub-bands of raw data in the frequency domain. PRI images are in ground range geometry. MLI, which stands for Multi-look Intensity, images are obtained from SLC data or from PRI images data by incoherently averaging in space over neighboring pixels in azimuth and in range. Depending whether the averaging was computed for a SLC or a PRI image, the corresponding MLI image is either in slant-range or in ground-range geometry. MLI images consist of floating point numbers with the image width and height determined by the number of range and azimuth looks selected.

SLC/PRI/MLI data can be either obtained from processing raw data using the Gamma MSP processor or delivered as such by a Processing Facility. Both formats are supported by the ISP package for interferometric processing. Interferometric processing will generate a number of image products: interferogram (complex valued), unwrapped phase image, coherence image and intensity images (all real valued). Format description for all data types can be found in the ISP Reference Manual.

At first data has to be prepared in order to cope with the format used by the GAMMA Software. To cope with the difficulty to decipher the variable CEOS format used for most SAR data products, the ISP (as well as all other GAMMA Software packages) uses a simple data structure for the metadata in the leader file accompanying the image data. Processing related parameters and SAR data characteristics are stored in a text file with system parameters referenced using simple keywords. The structure of the file can be initialized and updated using the ISP programs that write out files called the **ISP SLC parameter file**. Refer to the ISP Reference Manual for information on the structure of this file. Additional operations in the pre-processing step include manipulation of orbital state vectors and calibration. This last aspect is in particular relevant if PRI/MLI data will be analyzed. The ISP encompasses programs for the generation of an MLI image from an SLC dataset, for the radiometric calibration of SLC, PRI and MLI data, for SLC oversampling and for the conversion from slant-range to ground-range and vice versa.

After pre-processing, SLC data are ready for interferometric processing. At first co-registration of two SLCs forming an interferometric image pair is required. At this stage the ISP offset parameter file is generated, which keeps track of initial offsets and offset polynomial required for the resampling of one SLC to perfectly overlap with the reference SLC. Co-registration can also be performed for a dataset of more SLC with respect to a reference SLC in order to obtain a stack of SLCs to be used then for interferometric processing. Once images have been co-registered, common-band filtering in range and azimuth is applied and the interferogram is generated. In order to further utilize the interferogram, an estimate of the baseline has to be computed. The curved Earth trend can then be removed in order to obtain an interferogram including topographic and displacement phase only, as well as possible atmospheric distortions and phase noise. With this information it is possible to compute the interferometric coherence. Since the phase is wrapped in the interval $(-\pi,\pi)$, phase unwrapping is required to be able to correctly interpret it. The ISP offers two procedures for phase unwrapping based respectively on branch cut algorithm and Minimum Cost Flow techniques. In order to obtain from the unwrapped phase an elevation map, the baseline has to be refined first. After refinement the inversion from unwrapped phase to height is straightforward.

Figure 1 shows a typical flowchart for InSAR processing with the ISP package. Pre-processing characteristic parameters have to be determined from the CEOS leader file and/or the SLC image data. First of all the image data and the metadata are transcribed from the storage media to the system on which the data will be processed. Section 2 describes the different approaches used to import data files for pre-processing. Information on the main processing blocks is provided in Sections 3 to 11. The more general description of the available processing tools is followed by a number of Examples describing specific processing sequences. Example A describes the step for interferometric processing, i.e. generation of an interferogram, phase unwrapping and generation of an interferometric height map. Example B explains how to calibrate several types of SAR images. Example C is dedicated to offset tracking processing.

For details on individual programs please refer to the Reference Manual. Information on the parameters required by a specific program is also obtained by entering at the command line the name of the program.

It should be remarked that parameter values provided in the processing examples cannot be considered valid for all cases. It is possible that one or more values might have to be adapted to the specific case being processed. It is advised to look carefully at the messages printed on stdout when running each individual program. For assistance please get in contact with us (gamma@gamma-rs.ch).

It is recommended that a file name with a <scene_identifier> label be used. The scene identifier could be the orbit number or the date of the acquisition, such as (yy)yymmdd. In the following we will refer to <scene identifier> with the symbol “*”.

When using any of the ISP programs, a report will be printed on the screen (stdout) while running. The report contains various information and execution times. The report can alternatively be saved to ASCII text file. It is recommended that a file name of the type *.out be used. To redirect the report to the ASCII file, at the end of the command line the UNIX redirecting symbol “>” is used followed by the file name. If this information is not provided, the processing report will be printed on the screen (stdout).

Processing related parameters and data characteristics are saved as text files in ASCII format (e.g. variations of the baseline components in range and azimuth). These data sets can easily be imported in external software for analysis and visualization. Here we use the freely available public domain program *xmgrace*, which is available for all platforms for which the GAMMA Software is available.

The display of the final and intermediate products and the generation of easily portable images in SUNraster or bmp format are supported with programs available in the DISP package.

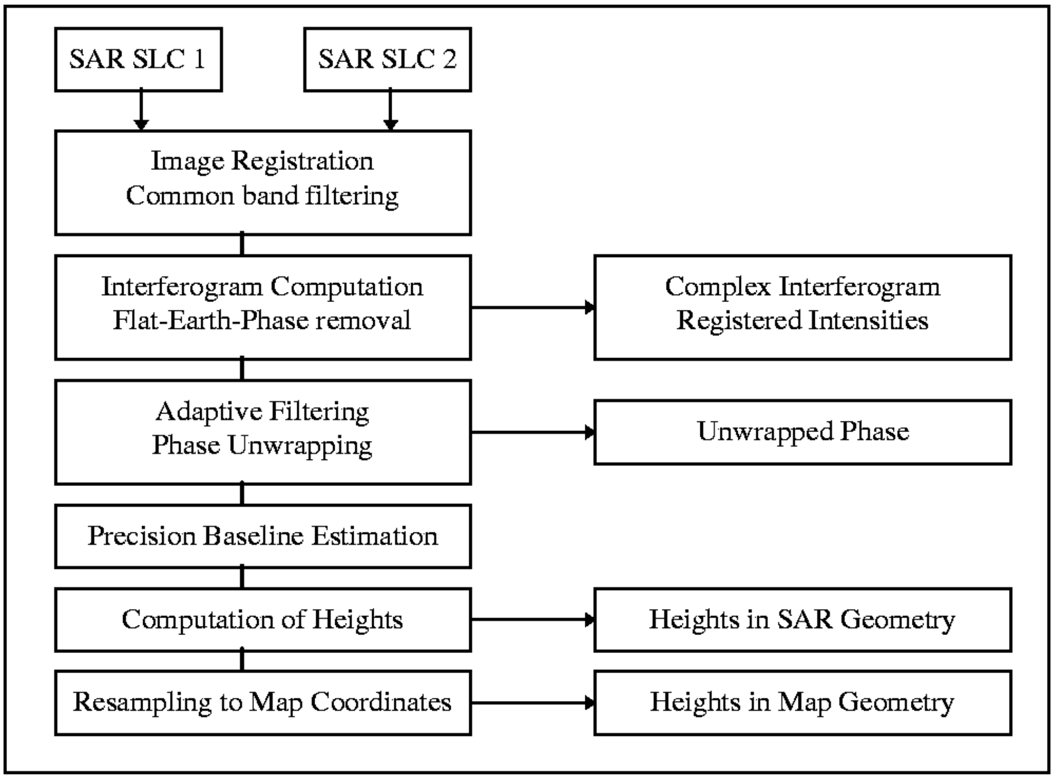


Figure 1. ISP flow chart.

2. Pre-processing

2.1. Data transcription and modification

SLC data obtained from processing the raw data with the MSP package are in complex floating point representation, each complex sample consists of either a pair of 4 byte floating point numbers (FCOMPLEX format) or a pair of 2 byte short integer numbers (SCOMPLEX format). There is no file header or zero padding of the data. Each record corresponds to a single range line.

The format of a SLC dataset generated from an external source is different depending on the processing facility. For example in case of ERS data provided by ESA PAFs real and imaginary part of the SLC data are in 2 byte integer format and are stored in the image data file. Each record corresponds to a single range line. Metadata is provided in the data leader file, following the typical CEOS format. ENVISAT ASAR data is instead provided in the form of a single file containing metadata and image data.

MLI (i.e. multi-look intensity) and PRI (i.e. ground range intensity) images can be either in floating point format (4 byte per pixel) or short integer (2 byte per pixel).

If data has been obtained from an external source, they are usually distributed on CD-ROM, DVD-ROM or ftp. Older data sets from ERS and JERS-1 might be stored on an Exabyte 8 mm tape. From the support media the CEOS format files need to be copied to a local directory on the machine with sufficient space. The space required for processing depends on the sensor.

Copying data from CD-ROM or DVD-ROM is straightforward. For ERS data stored on tape the GAMMA Software offers scripts for automatically reading in the files necessary for processing (see Table below).

Script name	Functionality
ERS_ASF_SLC	READ SLC data processed by the Alaska SAR Facility (ASF)
ERS_ESA_SLC	Read SLC data processed by the German-PAF (DPAF) or by ESA-ESRIN
ERS_ESA_PRI	Read PRI data processed by one of the ESA PAFs

For JERS CEOS level 1.2 SLC data the program ***copy_NASDA_SLC*** must be used to strip off a header from the image file and set the data in the format used by the GAMMA software.

For SIR-C SLC data the programs ***dcomp_sirc*** and ***dcomp_sirc_quad*** allow uncompressing Single Polarization Mode and Quad Polarization Mode data as provided by JPL.

Tip: If not done automatically by the script or the program, it is recommended that a file name of the type *.slc for an SLC and *.pri or *.grd for a PRI. For MLI images it is recommended that a file name of the type *.mli or *.grd be used if the image has been obtained from an SLC or a precision image respectively. In this way the geometry of the multi-look image is highlighted (ground-range). The scene identifier “*” could be the orbit number or the date of the acquisition, such as (yy)yymmdd. For the SAR leader file we suggest the extension .ldr.

2.2. Generation of ISP SLC parameter file

For SLC and PRI data all information concerning sensor and acquisition mode, geographical coordinates, acquisition time and SAR processing are stored in an ISP-specific parameter file, the ISP SLC parameter file. The ISP has a standard parameter file format to describe SLC, MLI and PRI image products. In this way data from many different processors can be used by the ISP. For SLC and PRI data, the parameter file is generated by programs that ingest the CEOS leader files, the image header file or the parameter file of the Modular SAR Processor (MSP), extract the appropriate parameters, query the user on the section of the image section to be processed, and set up the processing parameter file. For MLI images the parameter file is obtained when generating the MLI itself from the parameter file or the original SLC/PRI dataset.

The ISP Image parameter file is scene dependent and needs to be generated each time a new dataset is to be processed. Certain of the processing parameters given in the ISP SLC parameter file can be updated by inputs on the command line for some of the GAMMA Software programs.

The parameter file for a PRI images has the same format as the SLC/MLI parameter files used in the ISP. The indicated near center and far swath ranges correspond to ground-ranges, though, and not to slant ranges.

For SLC data processed with the Gamma MSP the ISP SLC/MLI parameter file is generated using the program ***par_MSP***. For SLC/MLI/PRI data not obtained with the GAMMA software, the programs required for the generation of an ISP SLC/MLI parameter file are grouped on a sensor basis. In this case, the creation of an ISP Image Parameter file is supported by the set of programs par_<facility>. Each of the programs creates the ISP SLC/MLI parameter file, as well as in some cases of the SLC/PRI image data, starting from the image data file and the CEOS leader file. It should be noticed that the CEOS leader file might be a separated file or embedded in the image file (e.g. in the case of ENVISAT ASAR).

par_<facility> reads the file(s) provided in input, searches for information needed for SLC image parameter file, and creates the SLC image parameter file. Below the programs that generate an ISP SLC parameter file for data obtained from external sources (i.e. not from the MSP) are grouped on a sensor basis.

Tip: If not done yet, for example because the image data is not yet in the format readable by the GAMMA software, it is recommended that a file name of the type *.slc for an SLC, *.pri or *.grd for a PRI, and *.mli for a MLI be used as output. The scene identifier could be the orbit number or the date of the acquisition, such as (yy)yymmdd. The same applies to the SAR leader file, for which we suggest the extension .ldr.

After running the appropriate par_<facility> program, an SLC image can be displayed with the DISP program ***disSLC*** or be saved to SUNraster / bmp format with the program ***rasSLC***. A PRI image can be displayed with the DISP program ***dispwr*** or be saved to SUNraster / bmp format with the program ***raspwr***. If the user prefers using a log-scale the programs to be used are ***dis_dB*** and ***ras_dB***.

2.2.1. ERS

The Table lists the programs supporting the generation of the ISP SLC parameter file for ERS data depending on archiving facility and/or SAR processor. More information on each program is provided in the ISP Reference Manual.

Program	**Image Type and facility**
par_ACS_ERS	SLC data from the ACS processor used by Indian PAF
par_ASF_91	SLC data from Alaskan SAR Facility, Fairbanks (1991-1996)
par_ASF_96	SLC data from Alaskan SAR Facility, Fairbanks (after 1996)
par_ASF_PRI	PRI data produced by the Alaskan SAR Facility, Fairbanks after 1996
par_ATLSCI_ERS	SLC data produced using the Atlantis APP processor (CCRS)
par_ESA_ERS	SLC data from the German, Italian, UK or ESRIN PAF, either VMP or PGS processed data[1]
par_PRI	PRI data processed by ESRIN/ASI/D-PAF
par_PulSAR	SLC data processed using the PulSAR SAR processor from Phoenix Systems
par_RSI_ERS	SLC data processed by RSI

[1] ERS data delivered up until about 2005 was delivered in the VMP format. The PGS format has been introduced in 2005, being rather analogous to the header format of ENVISAT ASAR SLC data.

2.2.2. ENVISAT ASAR

To generate the ISP SLC parameter file for ENVISAT ASAR data, the program ***par_ASAR*** must be used. This program generates both the SLC image parameter file and the image data file(s) for ENVISAT ASAR SLC (Alternating Polarization, Image Mode, Wide Swath) and PRI data. For AP data an ISP SLC parameter file is generated for each of the two polarimetric channels provided.

It should be considered that for strips of ENV ISAT ASAR Wide Swath mode data the program reads the orbital state vectors corresponding to the first section of the processed data by ESA. If the strip is formed by more sections, an extension of the orbital state vectors information in the ISP SLC parameter file is required. This is achieved by using the program ORB_prop_SLC (see Section 2.3.1).

2.2.3. JERS-1

The Table lists the programs supporting the generation of the ISP SLC parameter file for JERS-1 data depending on data type, archiving facility and/or SAR processor. In some cases the original SLC is reformatted to adhere to the GAMMA format. More information on each program is provided in the ISP Reference Manual.

Program	**Image Type and facility**
par_ASF_PRI	PRI data produced by the Alaskan SAR Facility, Fairbanks after 1996
par_EORC_JERS_SLC	JERS-1 SLC level 1.1 data processed by JAXA EORC
par_PRI_ESRIN_JERS	JERS-1 PRI data processed by ESRIN PAF

The program ***par_ASF_PRI*** generates the ISP image parameter file from the CEOS metadata for ground range detected images (PRI) produced by the Alaskan SAR Facility after 1996. The program also reformats the image data to be compatible with the GAMMA software.

The program ***par_EORC_JERS_SLC*** generates the ISP image parameter file and reformats the SLC (Level 1.1) image data from the CEOS format leader and data file provided by JAXA EORC. The program also calibrates SLC image. The magnitude of the SLC returns the SAR backscatter intensity image in sigma nought format.

The program ***par_PRI_ESRIN_JERS*** generates the ISP image parameter file for ground range detected images (PRI) produced by ESA ESRIN. The image data is compatible with the GAMMA Software format.

In addition the GAMMA Software supports geocoded SAR backscatter image products (Level 2.1) in CEOS format produced by EORC/JAXA. The program ***par_JERS_geo*** (part of DIFF&GEO module) generates the ISP image parameter file from the CEOS metadata and a DEM/MAP parameter file. The program also reformats the data to be compatible with the GAMMA software. The output is a geocoded (ellipsoid-corrected, zero height), calibrated SAR backscatter image in sigma nought format.

2.2.4. PALSAR

The Table lists the programs supporting the generation of the ISP SLC parameter file for PALSAR data depending on archiving facility and/or SAR processor. More information on each program is provided in the ISP Reference Manual.

Program	Image Type and facility
par_EORC_PALSAR	SLC Level 1.1 products produced by EORC/JAXA in CEOS format
par_ERSDAC_PALSAR	SLC Level 1.1 products produced by ERSDAC in non-CEOS, VEXCEL format
par_KC_PALSAR_slr	MLI products produced by JAXA EORC for the Kyoto and Carbon Initiative in slant range geometry

The program ***par_EORC_PALSAR*** generates the ISP image parameter file from the CEOS metadata and reformats the SLC data records by removing the 412 byte line header. In this way the SLC data is compatible with the GAMMA software.

The program ***par_ERSDAC_PALSAR*** generates the ISP image parameter file from the PASLL*.SLC.par metadata. The SLC data have no header and are big-endian float complex (4 bytes real, 4 bytes imaginary/sample).

The program ***par_KC_PALSAR_slr*** generates the ISP image parameter file from the Kyoto and Carbon parameter data. The image file is provided as little endian, multi-look amplitude image in short integer format. The steps required to obtain a calibrated SAR intensity image are described in the Reference Manual.

In addition the GAMMA Software supports MLI Level 1.5 products produced by EORC/JAXA in CEOS format. The program ***par_EORC_PALSAR_geo*** (part of DIFF&GEO module) generates the ISP image parameter file from the CEOS metadata and a DEM/MAP parameter file if the image has been provided in geocoded format. The program also reformats the MLI data. In this way the MLI data is compatible with the GAMMA software. The output is calibrated SAR backscatter image in geo-referenced (i.e. ground range) or geocoded (ellipsoid-corrected) format. The geometry has been chosen by the user when ordering the data.

2.2.5. RADARSAT-1

The Table lists the programs supporting the generation of the ISP SLC parameter file for RADARSAT-1 data depending on archiving facility and/or SAR processor. More information on each program is provided in the ISP Reference Manual.

Program	Image Type and facility
par_ASF_PRI	PRI data produced by the Alaskan SAR Facility, Fairbanks after 1996
par_ASF_RSAT_SS	Ground range detected SCANSAR images produced by the Alaskan SAR Facility
par_RSAT_SCW	Wide-swath SCANSAR images (8-bit/value).
par_RSAT_SGF	Stripmap path images SGF (ground range) and SCANSAR SCW16 data from RSI/Atlantis
par_RSAT_SLC	Stripmap SLC data processed by RSI, Atlantis APP SAR processor, or the ASF BPP processor.

2.2.6. SIR-C

To generate the ISP SLC parameter file for SIR-C SLC data from JPL or USGS EOC, the program ***par_SIRC*** must be used. The program reads in the CEOS leader file and generates the ISP SLC parameter file.

2.2.7. TerraSAR-X and TanDEM-X

The ISP currently supports reading and processing of the following formats: SSC (i.e. Single Look Complex) and MGD (i.e. ground range intensity). The GAMMA software also supports the EEC format (i.e. geocoded intensity) in the DIFF&GEO module. The Table below lists the programs supporting the generation of the ISP SLC parameter file. More information is provided in the ISP and DIFF&GEO Reference Manuals.

Program	**Image Type / output**
par_TX_GRD	MGD data as provided by DLR
par_TX_SLC	SSC data as provided by DLR ISP image parameter and SLC file in the format used by GAMMA software
par_TX_geo *(DIFF&GEO)*	EEC data as provided by DLR ISP image parameter file, DEM parameter file and SLC file in the format used by GAMMA software

Each program reads the data and the annotation file as provided by DLR and creates one (or two) parameter file(s) and the image data file in the format used by the GAMMA software. The Table below describes the output of each program more specifically.

Program	**Output**
par_TX_GRD	ISP image parameter file SAR intensity file in GRD format (ground range)
par_TX_SLC	ISP image parameter file SAR image in SLC format (slant range)
par_TX_geo *(DIFF&GEO)*	ISP image parameter file DEM/MAP parameter file (see User's Guide of DIFF&GEO module on Geocoding and Image Registration) SAR intensity file in geocoded format

For multi-polarization data the specific program has to be repeated for each single channel. The annotation file (*.xml in the main directory of the image data) is the same for all channels.

2.2.8. RADARSAT-2

The interface between a RADARSAT-2 SLC in the format provided by MDA and the format used by the GAMMA Software is the program ***par_RSAT2_SLC***. RADARSAT-2 SLC data are provided in an archive file consisting of a selection of annotation and data files. To import the data into GAMMA software format the main product annotation file in XML format and the corresponding GEOTIFF are needed.

The interface between RADARSAT-2 SGF and SGX data in the format provided by MDA and the ground range data format used by GAMMA software is the program ***par_RSAT2_SG***. RADARSAT-2 SGF and SGX data are provided in an archive file consisting of a selection of annotation and data files. To import the data into GAMMA software format the main product annotation file in XML format and the corresponding GEOTIFF are needed.

2.2.9. COSMO-SkyMed

The interface between a COSMO-Skymed SLC in the format provided by ASI and the format used by the GAMMA Software is the program ***par_CS_SLC***. The program reads COSMO-

SkyMed SCS (Single-Look Complex Slant) (Level 1A) data as provided by ASI and creates the ISP image parameter and SLC file in the format used by GAMMA software. The format of COSMO-SkyMed SCS data is HDF5. More information is provided in the ISP Reference Manual. Data obtained in tiff format can be imported with the program ***par_CS_SLC_TIF***.

COSMO-Skymed provided by ASI in Level 1C, i.e. in geocoded GEC format, is supported with the program ***par_CS_geo*** (part of DIFF&GEO module). The program creates the DEM parameter file, the ISP SLC parameter file and the image file in the format used by GAMMA software. For COSMO-SkyMed GTC data produced by ASI, the program ***par_CS_geo*** reads the image data file (in HDF5 format) and creates 1) the ISP SLC image parameter file, 2) the DEM/MAP image parameter file and 3) the image data file in the format used by GAMMA software. The data are already relatively calibrated. The program transforms the amplitude data to intensity and applies the calibration constant provided in the header information.

2.2.10. KOMPSAT-5

The interface between a KOMPSAT-5 SLC in the format provided by KARI and the format used by the GAMMA Software is the program ***par_KS_SLC***. The program reads KOMPSAT-5 SCS (Level 1A) (Single-Look Complex Slant) as provided and creates the ISP image parameter and SLC file in the format used by GAMMA software. The format of KOMPSAT-5 SCS data is HDF5. More information is provided in the ISP Reference Manual.

The program ***par_KS_DGM*** supports the import of Kompsat-5 DGM (Level 1B) images as provided by KARI and generates the image file and the SLC parameter file.

2.2.11. IECAS airborne SAR

The interface between a IECAS SLC dataset and the format used by the GAMMA Software is the program ***par_IECAS_SLC***. The program reads dataset generated by IECAS and creates the ISP image parameter and SLC file in the format used by GAMMA software. More information is provided in the ISP Reference Manual.

2.2.12. RISAT

The interface between a RISAT-1 dataset and the format used by the GAMMA Software is consists of a suite of program called ***par_RISAT_xxx***. Each program reads the image dataset generated by ISRO and the corresponding CEOS leader file together with additional metadata information stored in a second file to create the ISP image parameter and image data file in the format used by GAMMA software. The Table below summarizes the programs supporting RISAT-1 datasets. More information is provided in the ISP and DIFF&GEO Reference Manuals.

Program	**Image Type / output**
par_RISAT_SLC	SLC data as provided by ISRO
par_RISAT_GRD	GRD data as provided by ISRO
par_RISAT_geo *(DIFF&GEO)*	Geocoded data as provided by ISRO

2.2.13. UAVSAR

The interface between a UAVSAR SLC and MLC dataset and the format used by the GAMMA Software is the program ***par_UAVSAR_SLC***. The program generates the ISP image parameter file from UAVSAR annotation file. For already geocoded UAVSAR data products, the GAMMA Software offers the program ***par_UAVSAR_geo*** (part of the DIFF&GEO module). The data files themselves are without headers and in little-endian byte order. Hence, it is necessary to swap the byte order before proceeding with further processing. Byte swapping for a file is supported by the program ***swap_bytes.*** In case of multiple files, it is suggested to use the script swap_bytes_all. More information is provided in the ISP Reference Manual.

2.2.14. SETHI ONERA airborne SAR

The interface between a ONERA SLC dataset acquired by SETHI and the format used by the GAMMA Software is the program ***par_SETHI_ONERA.*** The program reads dataset generated by ONERA and creates the ISP image parameter and SLC file in the format used by GAMMA software. More information is provided in the ISP Reference Manual.

2.3. Manipulation of orbital state vectors

Manipulation of orbital state vectors means either adding state vectors to those provided with the data or improving the accuracy of the position and velocity information contained in the metadata accompanying the SLC/PRI/MLI data.

If the SLC/PRI/MLI has been obtained with the MSP, this operation should have already been carried out. If data is obtained from an external source, the ISP offers a set of programs for adding and updating state vectors.

2.3.1. Generation of additional state vectors

If the number of state vectors provided with the SLC/PRI data is too small or the state vectors are too coarse for accurate processing of the image data, additional state vectors can be computed. The program ***ORB_prop_SLC*** calculates additional state vectors using orbit propagation and interpolation.

2.3.2. Modification of state vectors

The orbit state vectors of ERS and ENVISAT ASAR data provided by the product processing facilities can be improved by substituting the available records with data provided by external sources. Depending on the source of the orbital data, the ISP offers specific programs to determine the state vectors from the external data files and update the state vectors in the ISP SLC parameter file.

For ERS there are basically two types of orbital information from external sources available: DELFT orbits provided by DEOS and PRC precision orbits provided by DLR. For ENVISAT there are two types of orbital information from external sources available: DELFT orbits provided by DEOS and DORIS orbits provided by ESA. For more information the user should refer to the Documentation on InSAR Processing Theory.

For JERS, RADARSAT-1 and SIR-C there are no external sources that allow improving the orbital state vectors provided with the data. The state vectors provided with PALSAR, TerraSAR-X and COSMO-SkyMed data do not require an update.

For RADARSAT-2 orbital state vectors are available from MDA.

2.3.2.1. DELFT orbits, ERS and Envisat

To improve the state vectors in the ISP SLC parameter file with orbital data provided by DEOS, use the programs ***DELFT_vec2***. This program extracts and interpolates Delft ERS-1, ERS-2, and ENVISAT state vectors to calculate or update the ISP SLC parameter file state vectors.

The DELFT orbit data are distributed as files (ODR.*) for ERS-1, ERS-2, and ENVISAT. It is suggested to download the ODR files and the arclist to a local directory, from which the orbit files required for processing can be retrieved. Once downloaded both the ERS-1 and ERS-2 ODR files from the website, they should be stored in separate ERS-1 and ERS-2 directories. If this is too much data to download, only the required ODR files for processing can be downloaded. To find the necessary ODR file, the *arclist* file contained in the same directory with the ODR files at DEOS should be used.

2.3.2.2. PRC Precision Orbits, ERS

To extract state vectors from ERS-1 and ERS-2 PRC precision orbit state vector files and store them in an ISP SLC parameter file, use the program ***PRC_vec***. The program reads ERS-1 and ERS-2 PRC precision state vector files provided by ESA and extracts a specified number of state vectors that bracket the raw data set. The state vectors are written into the ISP SLC parameter file along with the start time and time interval between state vectors. The default number of state vectors is 5 spaced at 30 second intervals. This is adequate unless multiple frames are processed. Up to 64 state vectors can be extracted.

It is suggested to download the PRC files from ESA ESRIN ftp site to a local directory, from which the orbit files required for processing can be retrieved.

2.3.2.3. DORIS Precision Orbits, Envisat

To extract state vectors from ENVISAT ASAR DORIS precision orbit state vector files and store then in an ISP SLC parameter file, use the program ***DORIS_vec***. The program reads ENVISAT ASAR DORIS precision state vector files provided by ESA and extracts a specified number of state vectors that bracket the raw data set.

It is suggested to download the DORIS files from ESA ESRIN ftp site to a local directory, from which the orbit files required for processing can be retrieved.

2.3.2.4. MDA precision orbits, RADARSAT-2

Update of the orbital state vectors information in the parameter file of a RADARSAT-2 image is possible with the program ***RSAT2_vec***. The program requires external orbital information provided by MDA. The state vectors in the parameter file are updated. The user can choose the number of state vectors to be replaced/inserted in the updated ISP SLC parameter file.

2.4. Calibration

If an SLC/PRI/MLI has been obtained with the MSP but has not been calibrated, it is strongly suggested to calibrate the image with the MSP programs in order to avoid confusions with calibration of data generated from a processing facility. Refer to the MSP User's Guide for more information.

The ISP supports the radiometric calibration with the programs ***radcal_SLC***, ***radcal_PRI*** and ***radcal_MLI***. In particular, *radcal_PRI* converts ESA processed short integer format PRI images to radiometrically calibrated ground range images (float).

Radiometric calibration of SLC/MLI images consists of:

- range spreading loss correction
- antenna gain correction
- normalization reference area correction
- correction for calibration constant (absolute calibration)

Correction for analog digital converter (ADC) saturation is not supported.

Depending on the sensor and on data type, images can be uncalibrated, relatively calibrated or absolutely calibrated. For more details it is referred to the following Sub-sections.

While the correction for range spreading loss and the normalization reference area can be set by choosing the desired flag value (see Reference Guide for more information), correction for antenna gain, if applied, requires an external file (provided with the software). The antenna diagram consists of a text file in which the antenna pattern for the specific acquisition mode is listed. Absolute calibration is obtained by providing the calibration constant. Typically this value is provided by the processing facility and is stored in the ISP SLC parameter file.

The output can be scaled with an additional scaling factor, as required, for example, for output in short integer format.

In case no information is provided for a sensor, please contact gamma@gamma-rs.ch

Tip: When calibrating a MLI image it is recommended to use an extension *.cmli. When calibrating a SLC image it is recommended to use an extension *.cslc. When calibrating a ground range image, e.g. a PRI image, it is recommended to use an extension *.grd.

2.4.1. ERS

The file containing the antenna gain correction is provided with the software. The correction for the calibration factor is provided by the processing facility and can be found in the ISP SLC parameter file under the keyword "calibration gain".

2.4.2. ENVISAT ASAR

For ENVISAT ASAR the program ***ASAR_XCA*** supports the interpretation of the ASAR external calibration data file (ASA_XCA) provided by ESA. This file contains calibration scale factors for all ASAR modes and products and the ASAR antenna diagrams for all the swaths. The program reads the ENVISAT ASAR external calibration data file, writes calibration factors for the different modes and data products to the screen and generates antenna diagram files for all the modes or one specified mode in the format used by GAMMA software.

The correction for the calibration factor is also provided by the processing facility and can be found in the ISP SLC parameter file under the keyword “calibration gain”.

2.4.3. JERS-1

All image products supported by the GAMMA software are calibrated (see Section 2.2.3).

2.4.4. PALSAR

Level 1.1 (SLC) and 1.5 (orthorectified intensity) data are already relatively calibrated. To achieve absolute calibration for SLC data the external calibration factor must be provided at the command line of ***radcal_SLC***. The value -115 dB should be given at the command line. For more details see Example B. For orthorectified intensity products absolute calibration is performed during re-formatting of the data with the program ***par_EORC_PALSAR_geo***. The calibration constant is in this case -83 dB. Hence no further calibration is required. For K&C PALSAR data the calibration constant of -83 dB shall be applied. Calibration is supported by the program ***radcal_MLI***.

2.4.5. TerraSAR-X and TanDEM-X

Ddata in SSC, MGD and EEC format are already relatively calibrated. Absolute calibration is done automatically by the corresponding programs for the generation of the ISP SLC parameter file (See Section 2.2.7), which add the calibration constant found in the xml annotation file. For TerraSAR-X data acquired during the commissioning phase (before 7 January 2008) an additional offset of 56 dB needs to be added. This can be done for example with ***radcal_MLI*** or ***radcal_SLC.***

2.4.6. Cosmo-SkyMed

SLC image products are calibrated to sigma nought format when imported with the program ***par_CS_SLC***. The calibration constant for absolute calibration is provided in the SLC parameter file.

2.4.7. RADARSAT-2

SLC image products are calibrated. The program ***par_RSAT2_SLC*** scales however the image with a 60 dB factor (see also processing of ERS SLC data). To obtain the correct magnitude, e.g. after multi-looking, a scaling factor of 10^{-6} needs to be applied.

2.5. Multi-looking

With multi-looking it is possible to average or oversample an image. Averaging reduces noise, decreasing at the same time the spatial resolution. Oversampling can be used to improve the spatial resolution of the image, e.g. in order to facilitate phase unwrapping in difficult regions such as those with dense fringes. The ISP offers several programs depending on the format of the input dataset (complex or real valued) and on the format of the output dataset (complex or real valued).

The program ***multi_cpx*** allows multi-looking and/or subsetting a complex image to obtain a new image in complex format. The program supports both averaging and oversampling. This program can be applied to average, oversample and/or subset a complex interferogram.

The program ***multi_real*** allows multi-looking and/or subsetting a real valued image to obtain a new real valued image. This program can be used to multi-look and/or subset a coherence or an unwrapped phase image. Oversampling can be applied, which can be useful in particular for further processing with the unwrapped phase. Together with the multi-looked real valued image a corresponding ISP offset parameter file is generated.

The program ***multi_look*** allows multi-looking and/or subsetting an SLC to obtain an intensity image. Together with the multi-looked intensity image a corresponding ISP SLC parameter file is generated.

The program ***multi_look_MLI*** allows multi-looking and/or subsetting an intensity image. Together with the multi-looked intensity image a corresponding ISP SLC parameter file is generated.

For all programs the user can specify

- the number of looks to average/oversample in range and azimuth,
- the coordinates of the subset to be extracted

Tip: It is recommended to use an extension *.mli for the output MLI image.

A MLI image can be displayed with the DISP program ***dispwr*** or be saved to SUNraster / bmp format with the program ***raspwr***. If the user prefers using a log-scale the programs to be used are ***dis_dB*** and ***ras_dB***.

Below we list multi-look parameters (range x azimuth) that allow obtaining an MLI image with roughly square pixels on ground from a corresponding SLC or ground range (e.g. PRI, MGD etc.) image. The values are obtained by comparing the pixel spacing in (ground) range and azimuth. Multi-look factors are arbitrary; nonetheless, it is always desirable to use values that restitute images where the dimensions along the range and the azimuth direction are similar. While the first spaceborne sensors had a fixed viewing geometry and therefore one set of multi-look factors (or multiples of it) was sufficient to characterize the multi-look operation, recent spaceborne SAR systems offer the possibility to acquire images in different modes, thus resulting in different combinations of range and azimuth pixel spacing and therefore of possible multi-look factors that could be used. Rule of thumb the multi-look factors can be obtained by considering that in the multi-looked version the pixel spacing in ground range and along azimuth should be similar. The pixel spacing in ground range, when starting from SLC data, can be obtained from the slant range pixel spacing by dividing for the sine of the incidence angle at mid-swath. The ratio between ground range and azimuth pixel spacing (i.e. 1 to 3, 1 to 5) is the basic relationship between the multi-look factors. Multiples can be taken if the pixel spacing of the resulting MLI image is too small with respect to the desired pixel spacing.

Below some multi-look factors are provided for standard imaging configurations:

- ERS, ENVISAT SLC (swath 2): 1x5 to obtain 20 m pixel spacing, 2x10 for 40 m pixel spacing etc. (valid for Swath 2)
- ERS, ENVISAT PRI: 2x2 to obtain 25 m pixel spacing (original data has 12.5 m pixel size, with 3 looks)
- JERS, PALSAR Fine Beam Single SLC: 1x3 for approximately 10 m pixel spacing or 2x6 for approximately 20-25 m pixel spacing.
- PALSAR Fine Beam Double SLC: 1x4 for approximately 15 m pixel spacing, 2x8 or 2x9 for approximately 25 m pixel spacing, depending on look angle. Because of the large range of look angles, the most suitable combination can be derived by dividing range pixel spacing (in ground range) and azimuth pixel spacing in order to obtain the ratio of pixel spacing.
- PALSAR Polarimetric SLC: 1x7 for approximately 25 m pixel spacing
- RADARSAT-1 SLC: 1x4 for approximately 20 m pixel spacing
- TerraSAR-X SLC: 5x5 for approximately 10 m pixel spacing
- TerraSAR-X MGD: 4x4 for 11 m pixel spacing (original data has 2.75 m pixel spacing)
- COSMO-SkyMed SLC: 5x4 for approximately 10 m pixel spacing

For further information, please contact gamma@gamma-rs.ch

2.6. Additional pre-processing tools

In this Section we provide a list of tools that can be used before proceeding with interferometric processing. Some of them can also be used with PRI/MLI data.

Additional pre-processing tools include programs that allow

- the extraction of a subset of an SLC (-> useful if working with small areas and large number of images)
- the retrieval of the geographical coordinates of the corners of the SLC/PRI (-> for identification of the area covered by the dataset)
- the computation of the azimuth spectra of the SLC (-> recommended if images present large variations of Doppler spectral properties, e.g. RADARSAT, ERS/ENVISAT)

2.6.1. Copy / subset of SLC image file

To copy user defined image segment of an input SLC the program ***SLC_copy*** must be used. Correspondingly extraction of a segment of a MLI image is supported by the program ***MLI_copy.***

Both programs also generate an updated ISP SLC parameter file, which describes the geometry of the new image file. Starting, center, and ending times, and the near, center and far slant ranges are calculated for the selected SLC segment. The output SLC together with its parameter file contains the full information needed for further processing including for example geocoding. Furthermore the Doppler polynomial is adjusted for the relative offset between the image segment and the original image.

2.6.2. SLC oversampling

Range oversampling has several advantages for interferometric processing. In situations where a range spectrum shift has occurred due to high deskew, oversampling is required to ensure that the baseband SINC interpolation used in the ISP programs for resampling an SLC image (***SLC_interp, SLC_interp_map, interf_SLC***) functions optimally. In any case, range oversampling prior to resampling the images for interferogram generation is expected to improve the performance of the SINC interpolation. To oversample an image in SLC format the program ***SLC_ovr*** must be used. The program generates a new SLC and the corresponding ISP SLC parameter file.

SLC oversampling allows mixed-mode interferometry of PALSAR data acquired in Fine Beam Single (FBS) and Dual modes (FBD). FBD data have half range bandwidth compared to FBS data, i.e. twice as much range pixel spacing. Oversampling by a factor 2 the FBD dataset allows obtaining an image with similar pixel size compared to FBS and hence the generation of mixed-mode FBS-FBD interferograms at the pixel spacing of the higher resolution dataset.

2.6.3. Retrieval of image corner coordinates

To retrieve the lat-long coordinates of the corners of an SLC/PRI/MLI image the program ***SLC_corners*** must be used. This program determines the latitude and longitude (decimal degrees) of the corners and center pixel of the image. The image can be in either ground-range or slant-range geometry.

2.6.4. Calculation of azimuth spectrum

To compute the azimuth Doppler spectrum and Doppler centroid from SLC data, use the program ***az_spec_SLC***. The azimuth Doppler spectrum is written to a text file for plotting, which can be plotted using a program such as *xmgrace*. This program only estimates the constant component of the Doppler centroid. For data with varying Doppler from near to far range (e.g. RADARSAT-1 or JERS-1) the user can vary the position of the estimate.

2.6.5. Resampling from ground to slant range (and vice versa)

The ISP offers two programs to convert a real valued image from ground to slant range geometry and vice versa.

- Ground range to slant range image transformation: ***GRD_to_SR***
- Slant range to ground range image transformation: ***SR_to_GRD***

The first program is useful to convert PRI or orthonormal images to slant range geometry. The second program is useful if an image in slant range coordinates shall be put in the ground geometry. Refer also to Section 11 for the geometric transformation of interferometric height images from slant range to ground range coordinates.

Tip: It is recommended to use an extension *.grd for images converted to ground range.

3. Co-registration

Interferometric processing of complex SAR data combines two single look complex (SLC) images s_1 and s_2 into an interferogram. This requires co-registration of the two images at sub-pixel accuracy; a registration accuracy of better than 0.2 pixels is required in order not to reduce the interferometric correlation by more than 5%. Co-registration consists of the computation of the offsets in range and azimuth between the two images forming the interferometric pair and resampling of one image (slave) to perfectly match with the other (reference) image. Computation of the offset between two SLCs consists of two steps: estimation of the local offsets for a number of small areas throughout the image and generation of a polynomial that allows the resampling of the slave image to match with the reference SLC.

This procedure can be applied to co-register two SLCs forming an interferometric pair or, in more general terms, to co-register a set of SLCs to a common reference. If N images are considered, the procedure will be repeated N-1 times. The advantage of having a set of co-registered images is that all possible interferometric combinations can be computed without having the need of having to deal with the co-registration of the specific interferometric pair.

The GAMMA software offers a further procedure to co-register SLCs. This procedure requires the availability of a DEM (e.g. SRTM-3 or better resolution) and is based on a lookup table which has the size of the reference image and contains at each position the coordinate of the corresponding pixel in the other image. This co-registration procedure makes use of programs of the DIFF&GEO module and is described in the DIFF&GEO User's Guide to Geocoding and Image Registration. It is recommended for images with long baseline and in general when the cross-correlation method does not provide sufficient co-registration accuracy.

3.1. Generation of offset parameter file

Similarly to the ISP SLC parameter file that contains all information concerning a SAR image, interferometric products are accompanied by a parameter file, the ISP offset/processing parameter file. The ISP offset/processing parameter file contains the information on the co-registration of the SLC image pair, the image section to be processed, and parameters used or determined during the processing. Refer to the ISP Reference Manual for the format description and description of this file.

This file is usually created using with the program ***create_offset***. The program reads the SLC parameter files and queries the user for parameters required to calculate the offsets using either the cross correlation of intensity or fringe visibility algorithms.

The ISP offset parameter file can be viewed/edited with a simple text editor.

Tip: it is recommended to use a file name of the type *.off for the ISP offset parameter file.

3.2. Estimation of offsets

The calculation of the offsets is based on the local spatial correlation function for a number of small areas throughout the image. The image offsets which maximize the local correlation can be determined either by cross correlation of the real valued image intensity or by optimization of the fringe visibility, i.e. based on the complex valued signals. So computed offsets are then used to define the coefficients of the polynomial for the transformation between the slave and the reference geometry.

To compute offsets between the two images several programs are available depending whether an initial estimate or a precise estimate has to be computed. The ISP offers also the possibility to define the region in which offsets shall be computed.

Initial offsets help in guiding the precise estimation of the offsets. An initial estimate of the range and azimuth offsets can be obtained with either the program ***init_offset*** or ***init_offset_orbit***. Both programs compute a constant offset in range and in azimuth, i.e. do not take into account variation of the offsets along the range and the azimuth direction. While *init_offset* uses cross-correlation between two images extracted from the SLCs to determine the initial estimate of the offsets, *init_offset_orbit* uses the orbital information in the state vectors provided with the images. When using *init_offset* the user can select size and position of the subsets as well as the correlation signal-to-noise (SNR) value. The correlation SNR is obtained by comparing the height of the correlation peak relative to the average level of the correlation function. It provides a measure of the confidence in the offset estimate. SNR values greater than 6.0 indicate co-registration better than 1 pixel between the images to be co-registered. Especially in the case of large registration offsets or any kind of problems with the registration the (additional) use of *init_offset_orbit* is recommended. It should be noted that for sensors with inaccurate orbital information (e.g. RADARSAT or JERS) this method might lead to mis-estimation of the initial offsets.

A field of estimates of the offsets can be obtained with either ***offset_pwr*** or ***offset_SLC***. A set of image patches is defined in each of which the offsets in range and azimuth are estimated. For both programs the user can define number of windows, i.e. patches, and the corresponding size in which the offsets in range and azimuth are computed. The programs distribute automatically the windows to cover uniformly the entire reference image. The two programs differ in terms of the algorithm used to compute the offsets. ***Offset_pwr*** considers the cross correlation of the real valued image intensity also known as intensity tracking. The capacity of this algorithm to correctly identify the offsets depends on some sort of image contrast within the local offset estimation window. ***Offset_SLC*** works instead by optimization of the fringe visibility, an algorithm also known as coherence tracking. As a consequence this method becomes unreliable in areas of low correlation because of the reduced fringe visibility.

The latter method of measuring offsets between SLC images is complementary to the intensity correlation method used in ***offset_pwr***. For identical search windows ***offset_SLC*** takes significantly longer to execute than *offset_pwr*, but has the advantage that the offset is actually based on the interferometric phase coherence. Another advantage is that the program can find offsets in areas with essentially no features at all such as snow fields where *offset_pwr* might fail.

If the user wants to specify size and position of the windows within the image, the ***offset_pwr_tracking*** and ***offset_SLC_tracking*** can be used. In this case the user can specify the spacing between windows in range and azimuth as well as in which part of the image shall the local offsets be computed. The program ***offset_pwr_tracking*** estimates range and azimuth offset fields for SLC images using intensity tracking, ***offset_SLC_tracking*** estimates range and azimuth offset fields for SLC images using coherence tracking. A specific example on the use of these programs is provided in Example B.

Tip: it is recommended to use a file name of the type *.offs and *.snr for the binary data files containing the local offsets and the SNR values respectively. For the text file including the offsets and extension *.offsets can be used.

3.2.1. Conversion of offsets to displacements

The offset computation procedure based on offset tracking can also be used as a basis for measuring displacements, e.g. in the case of glacier motion. This method of detecting motion from SAR data is called SAR offset-tracking (see Example C for details). Offset tracking is supported by the ISP program ***offset_tracking***. The program converts the input offset fields (*.offs) to an offset map (*.disp_map) which has the same dimensions as the original SLC reference image.

The output is in FCOMPLEX format. The real part corresponds to range displacements and the imaginary part to azimuth displacements. Different modes are supported. The offsets can be expressed as displacement in range and azimuth pixels (0), or as displacement in meters in slant range and azimuth direction (1), or as displacement in meters in ground range and azimuth direction (2, default mode).

3.3. Computation of offset polynomial

The local estimates of the offsets computed during the previous stage are then used to estimate coefficients of an offset polynomial for the range and the azimuth direction over the whole image. The image registration offsets are modeled as bilinear functions in range and azimuth i.e. of the type

range_offset = A0 + A1*range + A2*azimuth + A3*range*azimuth
azimuth_offset = B0 + B1*range + B2*azimuth + B3*range*azimuth

The coefficients A_i and B_i are determined via linear least-squares regression from the local offset estimates.

To generate the offset polynomials use the program ***offset_fit***. It computes range and azimuth registration offset polynomials from offsets estimated by one of the programs ***offset_pwr, offset_SLC, offset_pwr_tracking*** or ***offset_SLC_tracking***. As input the range and azimuth offsets and the corresponding SNR estimates (used a measure for the quality of the individual estimates) are provided. This program generates polynomial models of range and azimuth offsets using linear least-squares. The order of the polynomial can be chosen by the user. The approach implemented in ***offset_fit*** rejects offsets far from initial fit and iteratively improves the polynomial coefficients by using the culled offsets. The program also provides a measure of the accuracy of the co-registration error based on the estimated offset models.

Tip: it is recommended to use a file name of the type *.coffs for the binary data files containing the culled local offset. For the text file including the culled offsets and extension *.coffsets can be used.

3.4. Resampling

Once the offset functions are known the two SLC images can be co-registered. As this is done to the sub-pixel resolution precise resampling of one of the images is necessary. Appropriate interpolation methods are used to minimize interpolation errors.

The ISP offers two programs for co-registration: ***SLC_interp*** and ***SLC_interp_map***.

SLC_interp resamples the slave SLC to the reference geometry using 2-D SINC interpolation and the range and azimuth offset polynomials stored in the ISP offset/interferogram parameter file.

SLC_interp_map has the additional feature that a raster map of residual range and azimuth offsets can be taken into account in the resampling. This program is needed only in situations were there are non-linear deformations in the scene such as due to glacier motion, or large earthquakes.

The map of residual offsets is generated by a dense sampling of the offset field between the SLCs and is stored in the file of culled offsets generated with ***offset_fit***. The range and azimuth offsets are calculated as the sum of the contributions from the polynomials stored in the ISP offset parameter file and the local residual offsets. These residual offsets are measured between the reference SLC and the SLC that has been resampled to the geometry of reference SLC using the polynomials alone. This map contains the non-linear residual offsets between these two images. The sum of the residual offsets and the offset contribution determined from the polynomials are the offsets used to resample the slave SLC to the reference geometry.

Processing of data using ***SLC_interp_map*** requires first generating an SLC resampled to the geometry of the reference SLC with ***SLC_interp***. Offsets are then measured using a dense grid between the resampled SLC and the reference SLC. These measurements contain information on the non-linear offset function between the SLCs due to scene deformation (e.g. earthquake or glacier motion). Taking this deformation into account when forming the interferogram, improves the interferometric correlation.

Both programs can adapt to changes in the Doppler centroid along-track. This is especially applicable to the processing of long strips (>200 km) of RADARSAT data where the Doppler centroid changes by more than 100 Hz/frame.

Two co-registered SLC images can be displayed with the DISP program ***dis2SLC***.

Tip: it is recommended to use a file name of the type *.rslc and *.rslc.par for the resampled SLC and the corresponding ISP SLC parameter file.

3.5. Refinement of offsets

If the resampled SLC presents residual offsets compared to the reference SLC, the offsets can be improved by measuring the residual offset between the resampled SLC and the reference SLC with the programs described in Section 4.2 and 4.3. The residual offset model can then be added to the initial offset model and resampling can be newly performed. To add a polynomial fit of these residuals to the original polynomial the program ***offset_add*** must be used. This program is most useful for images that have large relative distortions. If these distortions are non-linear, then the program ***SLC_interp_map*** is required (see Section 4.4).

4. Baseline estimation

The next step is to determine an estimate for the baseline. The ISP offers several programs for the computation of the baseline, to convert from the TCN reference system to the parallel-perpendicular reference system and to improve an initial estimate to obtain a precise estimate. In the GAMMA software the baseline is defined as the baseline at the first line of the reference SLC.

For an initial estimation of the baseline use the program ***base_init***. This is a generic program that offers the possibility to estimate the baseline based on either the orbit state vectors, the SLC registration offsets or the interferogram fringe rate. It is usually best to combine two of these methods. The first two methods give the better estimates for the parallel baseline component. The third method gives very often the better estimate for the perpendicular baseline component. Therefore, it is recommended to combine either one of the first two methods with the third method. The included algorithms allow to estimate the interferometric baseline and the (along track) rate of change of the interferometric baseline. The output consists of baseline components in the TCN reference system at centre of the image and their rate of change along track. Refer to the ISP Reference Manual for format description of the ISP baseline file.

Tip: it is recommended to use a file name of the type *.base for the baseline parameter file.

The more specific program ***base_orbit*** can be used if the baseline has to be estimated from the orbital data only, e.g. when an interferogram is not available yet. It reads the state vectors in SLC parameter files, computes the interferometric baseline and generates a baseline parameter file (optional). While it is possible to obtain an initial guess of all baseline components with this program, the error affecting the estimates can be relevant in case on inaccurate orbital information (e.g. RADARSAT or JERS).

Tip: it is recommended to use a file name of the type *.base_orbit for the baseline parameter file.

To obtain an estimate of the perpendicular component of the baseline the program ***base_est_fft*** can be used if an interferogram is available. The program reads a section of the complex interferogram, and then calculates the two-dimensional FFT to determine the local fringe rate. From this an estimate of the perpendicular baseline is obtained. For the estimate to be accurate, it is necessary to have clear visibility of the fringes, i.e. to consider an area not affected by decorrelation.

Having available a baseline parameter file, it is possible to determine the parallel and perpendicular component with the program ***base_perp***. This program determines the two components both along-track and across-track for the whole image.

To obtain a precise estimate of the baseline, use the program ***base_ls***. In this case ground control points, unwrapped interferometric phase and elevation information are used as input. In order to run ***base_ls*** it is therefore necessary to have run the interferometric processing up to the phase unwrapping stage first. ***base_ls*** computes a precise estimate of the interferometric baseline using ground control points of known x and y image coordinates, the corresponding terrain height, and the corresponding unwrapped interferometric phase. A non-linear least-squares solution algorithm is used to determine the baseline parameter values. The computed baseline (in TCN coordinates) is written to the baseline file and is used to calculate heights and for differential interferometry (see Section 10). Statistics for quality control of the least-squares fit are computed and displayed.

To determine the baseline file for the subset of an SLC, use the program ***base_copy***. After SLC extraction (with the ISP program ***SLC_copy***, see Section 2), the baseline needs to be adjusted according to the geometric parameters in the reference SLC parameter file of the SLC subset. When extracting a subset from the SLC this reference position changes and therefore the baseline must be calculated for the segment. The baseline component rates from the original baseline file are used to calculate the baseline values for the center of the interferogram created using the SLC subset as the reference image.

5. Common band filtering and interferogram calculation

Knowing the perpendicular baseline the filter function for the common band filtering of the range spectrum can be determined. Similarly, the azimuth spectra, which differ due to non-identical Doppler centroids, are filtered, in order to include only those parts of the spectra which are common to the two spectra. From the two common-band filtered, co-registered images the normalized complex interferogram can be computed with cross-correlation. In order to improve the estimates of the interferometric phase and correlation multi-looking can be performed.

For common-band filtering and generation of an interferogram the ISP supports two different methods. The first method registers in a first step the slave SLC to the reference SLC. Then, the two registered SLC are used for the interferogram calculation. For this the program ***SLC_intf*** is used. The second method carries out the resampling and interferogram calculation in a single step. In this case the program ***interf_SLC*** is used.

The first method provides a simple way to register multiple SLC to the same geometry. The main advantages of the second method are the lower disk space used (no additional SLC files are generated) and the higher computational efficiency achieved by the combined application of the interpolation and spectral filtering in one step.

Both programs have the possibility to choose the multi-look factor and whether common-band filtering in range and azimuth be applied. While ***SLC_intf*** generates only the complex interferogram, ***interf_SLC*** generates also the multi-looked co-registered intensity images.

SLC_intf also supports generation of interferograms with SLC data acquired with sensors having different radar center frequencies such as ERS and ENVISAT. When the carrier frequencies of the two SLCs are different, ***SLC_intf*** applies a range phase correction proportional to the slant range.

A complex valued interferogram can be displayed with the DISP program ***dismph*** or be saved to SUNraster / bmp format with the program ***rasmph***. These programs display the interferometric fringes with the correlation image as background. To display the interferogram with the backscatter intensity as background, use the program ***dismph_pwr*** or the program ***rasmph_pwr*** to generate a SUNraster / bmp version.

Tip: it is recommended to use a file name of the type *.int for the complex interferogram. If also the multi-look intensities are generated, you can use for these the extension pwr1 and pwr2 or mli1 and mli2.

6. Interferogram flattening

The complex interferogram consist of several contributions (curved Earth, topography, noise and eventually atmospheric distortions and displacements). With the flattening operation azimuth and range phase trends expected for a curved Earth are removed from the interferogram. This is done in order to facilitate consecutive filtering, averaging, and phase unwrapping.

To remove the curved Earth interferometric phase trend from the complex interferogram use the program ***ph_slope_base***. The program generates a phase trend using a spherical Earth and a baseline model. The so generated phase is then subtracted from the original interferogram. This program allows also the reverse operation in which the curved Earth phase trend is added back to an interferogram.

The complex valued interferogram can be displayed with the DISP program ***dismph*** or be saved to SUNraster / bmp format with the program ***rasmph***. These programs display the interferometric fringes with the correlation image as background. To display the interferogram with the backscatter intensity as background, use the program ***dismph_pwr*** or the program ***rasmph_pwr*** to generate a SUNraster / bmp version.

Tip: it is recommended to use a file name of the type *.flt for the flattened complex interferogram.

7. Coherence estimation

To compute the coherence, i.e. the magnitude of the cross-product of the two co-registered SLCs, the ISP offers the program ***cc_wave***. This program estimates the interferometric coherence from the normalized interferogram and the co-registered intensity images. Alternatively the complex-valued interferogram can be used alone. Size of the estimation window and weighting options can be selected by the user.

A coherence image can be displayed with the DISP program ***discc*** or be saved to SUNraster / bmp format with the program ***rascc***.

Tip: it is recommended to use a file name of the type *.cc for the coherence image.

The LAT module offers an advanced coherence computation program (***cc_ad***) that makes use of an adaptive coherence estimation window size.

8. Interferogram filtering

Possible steps preceding the phase unwrapping are multi-looking and filtering. The objective is to reduce the phase noise and therefore make the phase unwrapping simpler, more robust and more efficient since in this way we reduce the number of residues (problematic location for phase unwrapping, see the Documentation on InSAR Processing Theory and Section 9 in this document).

Multi-looking as well as filtering of the complex valued interferogram are supported by the ISP.

Multi-looking of a complex interferogram can be done with the program ***multi_cpx***, which has been described in Section 3. The result is an interferogram with less phase noise and lower spatial resolution.

For interferogram filtering, two different filters are included in the ISP. The first one accounts adaptively for the local phase gradient. The second one uses an advanced filtering technique as presented by Werner and Goldstein (1997). The goal of the adaptive filtering step is to reduce phase noise thereby reducing the number of residues without having loss in spatial resolution.

The program ***adapt_filt*** does adaptive filtering (adaptive to the local slope, respectively fringe rate) of the interferometric phase. The program reads the complex interferogram, computes locally the slope, and averages the interferometric phase along the local slope. The smoothed (respectively) filtered interferometric phase is written out to a file.

The program ***adf*** does adaptive filtering with filtering function based on local fringe spectrum. The program reads the complex valued interferogram, computes locally the interferogram power spectrum, designs a filter based on the power spectrum, filters the interferogram, estimates the phase noise coherence value for the filtered interferogram and writes out the filtered interferogram and coherence map.

In case of low frequency phase distortions (residual trends or large scale atmosphere) which would require large filter windows, it is suggested to use the program ***fspf*** since it is computationally much more efficient. The program can be used for both complex valued point interferograms (e.g. pdiff obtained after subtracting of the simulated topographic phase, or re-wrapped residual phases from a regression analysis) or float data (residual phases, atmospheric path delays).

The ISP also offers the program ***bpf***, which implements a 2-D bandpass filter on an image. The filter is specified in terms of the normalized center frequencies and bandwidths in x (across the image) and y (down). The filter can be used not only with complex data such as interferograms but also with real valued images.

The filtered interferogram, as any complex valued interferogram, can be displayed with the DISP program ***dismph*** or be saved to SUNraster / bmp format with the program ***rasmph***. These programs display the interferometric fringes with the correlation image as background. To display the interferogram with the backscatter intensity as background, use the program ***dismph_pwr*** or the program ***rasmph_pwr*** to generate a SUNraster / bmp version.

Tip: it is recommended to use a file name of the type *.sm for the filtered image and *.sm_cc for the coherence derived from the smoothened interferogram (if created).

9. Phase unwrapping

From the complex interferogram the interferometric phase is only known Modulo 2π. In order to be able to relate the interferometric phase to the interferometric imaging geometry and finally to retrieve the topographic height (as well as displacement information) and true ground range the correct multiple of 2π has to be added. This is done in the phase unwrapping step. Phase unwrapping is a problematic step due to discontinuities (for example due to layover) and inconsistencies (residues as a result of high phase noise). Filtering and multi-looking may be performed to reduce the phase noise. Relatively flat areas of intermediate to high coherence are not problematic. More care has to be taken in more rugged terrain and for areas of low coherence. In case of unwrapping a differential interferogram, removal of as much topographic related phase as possible by subtracting first a simulated interferogram obtained from a DEM is strongly advised. The flatter the interferogram to be unwrapped the better the unwrapping performance.

The ISP supports two techniques for phase unwrapping. The first phase unwrapping technique is a branch-cut region growing algorithm similar to the one described in (Rosen et al., 1994). It is typically applied to the filtered interferogram. Critical areas such as areas of very low coherence or residues are identified and avoided in the phase unwrapping. The encoded phase unwrapping algorithm is reasonably reliable and runs time efficient.

The second phase unwrapping method uses minimum cost flow (MCF) techniques and a triangular irregular network (TIN). This technique is a global optimization technique to the phase unwrapping problem. Other advantages of this technique are that gaps in the input data (for example at locations of very low coherence) can be considered and the higher density of the triangular network. Masking, adaptive thinning, and patch processing, are used to permit efficient and robust unwrapping even of very large interferograms. A more detailed characterization of the phase unwrapping solutions supported by the ISP is given in (Werner et al., 2003).

9.1. Phase unwrapping with branch-cut algorithm

Phase unwrapping using the branch-cut algorithm consists of the following steps:

- Masking low correlation areas
- Generation of neutrons to exclude regions of layover by generation of dense cuts
- Determination of residues
- Connection of residues through neutral trees
- Unwrapping of interferometric phase

Additionally the unwrapping can be continued in areas disconnected from the already unwrapped area as follows:

- Construction of bridges between disconnected regions
- Unwrapping of disconnected areas

The same procedure can be repeated with new bridges to unwrap further disconnected areas.

9.1.1. Masking low correlation areas

Areas of low interferometric correlation are affected by high phase noise and therefore are excluded during phase unwrapping. To mask out areas of low correlation use the program ***corr_flag***. This program generates a flag file used throughout the various steps necessary for the unwrapping of the interferometric phase. The user can select the threshold for masking out areas of low correlation. Other programs (see below) will add further information to the flag file (see Sections 9.1.2 to 9.1.4).

The flag file can be displayed with the DISP program ***disbyte*** or be saved to SUNraster / bmp format with the program ***rasbyte***.

The flag file can be reset at this stage or at any later stage (before unwrapping) with the program ***clear_flag***.

Tip: it is recommended to use a file name of the type *.flag for the correlation mask.

9.1.2. Generation of neutrons

Neutrons effectively reduce the size of the search area around residues for generation of branches and create a dense network of branch cuts in the vicinity of actual residues. To identify neutrons use the program ***neutron***. It reads an intensity image and uses this information to generate neutrons to guide the generation of phase unwrapping trees (branch cuts). A neutron is placed in the flag file obtained with *corr_flag* if the intensity is times the average image intensity. Other programs (see below) will add further information to the flag file (see Sections 9.1.3 and 9.1.4).

9.1.3. Determination of residues

The presence of residues leads to inconsistencies in the phase unwrapping process, thus they have to be identified. The program ***residue*** marks such locations and updates the flag file by including the residues. Another possibility is to use the program ***residue_cc***, which contrarily to ***residue*** does not consider the phases of the low coherence areas. As a consequence much fewer residues are found, which will speed up the connection of the residues by branch cuts (see below). When using ***residue_cc*** the branch cuts should be determined with ***tree_cc*** rather than *tree_gzw* (see below for a description of these programs).

9.1.4. Connection of residues through neutral trees

Once the residues have been determined, they are connected generating a tree-like structure. The branch cuts (walls) are places that localize jumps of multiples of 2π such as in regions of lay-over. To generate the branch cut structure use one of the programs ***tree_cc*** or ***tree_gzw***. Use ***tree_gzw*** if also neutrons have been identified, otherwise ***tree_cc***. The programs use the marked low correlation areas, the residues and the neutrons in the flag file and construct the tree-like structure of branch cuts. Branch cuts are then written to the flag file.

9.1.5. Unwrapping of interferometric phase

Once the flag file has been completed with the branch cuts, the phase can be unwrapped. To do this use the program ***grasses***. This program reads in the complex valued interferogram and the flag file to generate the unwrapped phase image. The user can choose where to start unwrapping the phase; it is convenient to choose a point of high coherence in the middle of an area with valid points to be unwrapped. The unwrapping proceeds by region growing for the entire area connected to the starting location avoiding low correlation areas and without crossing neutral trees.

The unwrapped interferogram can be displayed with the DISP program ***disrmg*** or be saved to SUNraster / bmp format with the program ***rasrmg***. It is suggested to use a phase scaling of .333 or .2.

9.1.6. Construction of bridges between disconnected regions

Bridges to connect wrapped and unwrapped locations can be constructed to expand the unwrapping. Bridges are saved in an ASCII text file. This file needs to be generated manually by the user. When unwrapping, the multiple of 2π at the not unwrapped location will be estimated based on the reference unwrapped phase and the expected multiple of 2π offset between the two locations (usually 0 as locations with comparable topographic height will be selected).

To construct the bridges the wrapped and unwrapped areas together with the flag file are first combined to generate a SUNraster file using the DISP program ***rastree***. By visualizing this image for example with the DISP program ***disras***, the display shows which areas are already unwrapped (bright colors) and which ones are not (darker colors). At this stage the user can generate a text file of the type *.bridges with the following content:

col of unwrapped location	row of unwrapped location	col of wrapped location	row of wrapped location	phase offset in 2π
640	45	600	48	0
100	630	80	630	0

9.1.7. Unwrapping of disconnected areas

The bridges are then used to unwrap the phase in the disconnected areas. For this use the program ***bridge***. The program reads in the wrapped and the unwrapped image files, the flag file and the ASCII text file containing the bridges. The file containing the unwrapped phase is updated with the additionally unwrapped phases. For data security reasons it is recommended to backup the unwrapped phase file before running bridge.

The final unwrapped interferogram can be displayed with the DISP program ***disrmg*** or be saved to SUNraster / bmp format with the program ***rasrmg***. It is suggested to use a phase scaling of .333 or .2.

9.2. Phase unwrapping with the Minimum Cost Flow algorithm

Phase unwrapping using the Minimum Cost Flow algorithm consists of the following steps:

- Generation of phase unwrapping validity mask (optional)
- Adaptive sampling reduction for validity mask (optional)
- Unwrapping of interferometric phase
- Weighted interpolation to fill gaps in unwrapped phase image
- Use of interpolated unwrapped phase as model to unwrap the initial interferogram

Each step is described in detail in the following sub-sections.

9.2.1. Generation of phase unwrapping validity mask

Areas of low interferometric correlation are affected by high phase noise and therefore can represent a problem during phase unwrapping. To exclude these areas from further processing a validity mask can be generated. This is done with the program ***rascc_mask***. The program reads in the interferometric correlation and generates a SUNraster file where pixels with coherence and intensity below a given threshold are flagged as not valid for phase unwrapping.

The use of a validity mask is optional. It is mainly recommended for data with predominantly high phase noise, as in the case of long-term differential interferograms.

The flag file can be displayed with the DISP program ***disras***.

Tip: it is recommended to use a file name of the type *.mask.ras for the validity flag mask.

9.2.2. Adaptive sampling reduction for validity mask

The spatial sampling of the phase unwrapping validity mask can be reduced using the program ***rascc_mask_thinning***. This thinning of the points is done adaptively to the phase variation (noise, gradient). In areas of low phase variation sparser sampling is possible, which allows to increase the efficiency of the MCF phase unwrapping. This step is optional.

The masking and adaptive sampling reduction result in a significant reduction of the number of samples which have to be considered in the MCF optimization. Apart from the gain in efficiency this also allows to apply real global optimization (i.e. without patch processing, see below) for relatively large interferograms.

Tip: it is recommended to use a file name of the type *.mask_thinned.ras for the validity flag mask.

9.2.3. Phase unwrapping

Phase unwrapping of the filtered interferogram is done using the program ***mcf***. The program reads in the interferogram to be unwrapped, the interferometric correlation file, and the phase unwrapping validity mask, if considered, to generate the unwrapped interferogram. The interferometric correlation is used as weight for the MCF solution. The user can choose where to start unwrapping the phase; it is convenient to choose a point of high coherence in the middle of an area with valid points to be unwrapped.

The user can also choose whether to process in patches or not. Unwrapping the scene as a single patch will always be more reliable than patching. However this requires large memory. To unwrap a full frame of ASAR/ERS about 5 GB of RAM are needed if the image is coherent everywhere. If the interferograms is so large that it exceeds memory capacity, there are two ways to get around the memory requirement.

1) Tile the image. The image is divided into overlapping patches. Each patch is unwrapped and phases are matched at the overlapping tile boundaries. In general, the patches should be set to be as large as possible within the memory constraints of the computer. The user can see the memory usage when running the ***mcf*** program. Make sure that the machine is not swapping to disk while unwrapping. The patch size is determined by the parameters specified on the command line with the npat_r and npat_az parameters. Set the patch overlap parameter to be on the order of 250 pixels. For a 1 GB RAM machine it is suggested to start with 5 patches in range and 5 in azimuth. The primary difficulty with patch-based processing is that errors occur on the patch boundaries. Errors in the patch unwrapping propagate over the entire image and can be a problem.

2) Unwrap a smaller interferogram by first multi-looking in range and azimuth. First multi-look the interferogram and then unwrap the smaller multi-look interferogram. Use this unwrapped image as a phase model to unwrap the full-resolution complex interferogram. The advantage of this approach is that the unwrapping is for an image with more looks (lower phase noise) and is perhaps more reliable due to this. The downside of this approach is that very high frequency fringes can not be unwrapped because they get averaged away by the multi-looking. In general though, such high frequency fringes have low coherence and are in general difficult to unwrap correctly.

   In detail the steps for the multi-looking phase unwrapping approach are

   - Multi-look the wrapped interferogram using ***multi_cpx***. As multi-look factors 2x2 or 3x3 are suggested. In case of small RAM maybe 5x5 is required to stay within RAM resources. Also multi-look the MLI images using ***multi_real*** with the same multi-look factors. The multi-looked images can be displayed with ***dismph_pwr*** or saved to SUNraster file with ***rasmph_pwr***.
   - Estimate correlation with ***cc_wave*** using the multi-looked images so that the correlation map can be used for the mcf. Alternately, run ***adf*** on the multi-looked interferogram. This will also generate a coherence map.
   - Run ***mcf*** on the small image specifying only 1 patch in range and azimuth (npat_r=1, npat_az = 1) and display using ***disrmg*** or ***rasrmg***. Check to see if the unwrapping is ok. A phase scale factor of .3333 will reveal most phase unwrapping errors as sharp discontinuities in the phase where there are no strong topographic features.
   - Expand the unwrapped image using ***multi_real*** using negative values for the rlks and azlks parameters. Here be sure to use the offset parameter file of the output to be the original one of the full size interferogram. This way ***multi_real*** knows how large to make the expanded/interpolated image!

At this stage the phase has been unwrapped only at the points in the validity mask. The steps to get full spatial coverage are described in Sections 9.2.4 and 9.2.5.

The unwrapped interferogram can be displayed with the DISP program ***disrmg*** or be saved to SUNraster / bmp format with the program ***rasrmg***. It is suggested to use a phase scaling of .333 or .2.

Tip: it is recommended to use a file name of the type *.unw_init for the unwrapped interferogram.

9.2.4. Weighted interpolation to fill gaps in unwrapped phase data

The gaps in the unwrapped phase image can be filled by interpolation. To cope with the variable size of the gaps, an interpolator with an adaptive window size should be used. The ISP offers the program ***interp_ad***. The program uses a window of increasing size for increasing gap size. This allows avoiding unwanted filtering as would occur with a constant, and therefore relatively large, window size. These interpolated phases can be assumed to be close to the correct unwrapped phases, at least within an interval $(-\pi,\pi)$, and can therefore be used as phase model in the sense of the program ***unw_model*** (see below).

9.2.5. Phase unwrapping using model of unwrapped phase

To obtain unwrapped phases for the entire interferogram the ISP offers the program ***unw_model***. The unwrapped phases are recalculated from the complex valued interferogram (the filtered or unfiltered one can be used) assuming that the phase values in the model correspond to the correct unwrapped phase within the interval (-π,π). The resulting unwrapped phase meets the condition that re-wrapping of the unwrapped phase results in exactly the phase of the complex interferogram, except for the constant offset defined through the reference phase value (e.g. 0.0) indicated for the reference location.

This procedure is also well suited for the unwrapping of longer term differential interferograms with predominantly very low coherence, except for built up areas.

The final unwrapped interferogram can be displayed with the DISP program ***disrmg*** or be saved to SUNraster / bmp format with the program ***rasrmg***. It is suggested to use a phase scaling of .333 or .2.

9.2.6. Masking phase unwrapped images

Pixels of an unwrapped interferogram can be masked out with the program ***mask_data***. For this, a mask should be generate first (for example with the program ***rascc_mask***). Masking can be used to improve the display of unwrapped interferograms, for example by excluding areas of low coherence.

10. Precise baseline estimation

If the unwrapped phase is to be used for the derivation of a height map, it is fundamental that the baseline information is as accurate as possible. A first estimate of the interferometric baseline was determined from the orbit (track) data or the average interferogram fringe frequency. This estimate was sufficient for the subtraction of the curved Earth phase trend carried out to facilitate the filtering of the interferogram and the coherence estimation. This estimate is not accurate enough, though, to convert the unwrapped interferometric phase to topographic heights. Therefore, refined baseline estimation is done using least squares fit for a number of control points of known height. More specifically, the precise estimation of the baseline consists of the following steps:

- Generation of a SUNraster or bmp format image
- Selection of ground control points using the image created above as a guide
- Least square estimation of interferometric baseline

10.1. Generation of SUNraster of bmp format image

To select ground control points any 8-bit SUNraster or bmp format image of the analyzed scene can be used. A useful image is for example the display of the unwrapped interferometric phase together with the backscatter intensity. This can be obtained with the

DISP program *rasrmg*. Refer to the DISP reference manual for more information on possible images that can be generated.

10.2. Selection of ground control points

The ground control points can then be selected using the program ***gcp_ras***. The SUNraster file is displayed. Using the selection mouse button ground control points can be selected and the corresponding terrain heights (obtained from a map) written to the data input window. Ground control points in flat areas are preferred. Ground control points are best spread over the entire image. Selection of at least a dozen ground control points is recommended. Exiting the display window initiates the writing of the ground control points to standard output, respectively to the file indicated.

10.3. Least squares estimation of interferometric baseline

Once the ground control points are determined the corresponding unwrapped interferometric phases can be extracted from the unwrapped interferogram and written together with the GCP coordinates to the GCP data file using the program ***gcp_phase***.

A least squares estimate of the interferometric baseline is then obtained using the program ***base_ls***.

The least squares estimation is rather critical and does not always perform as desired. Therefore, it is recommended to observe the quality control information written to the screen. Reasons for non converting estimation are:

- poor initial baseline estimate
- poor selection of control points
- too many free parameters specified for least squares fitting
- too small area specified for phase averaging

All the four mentioned problems can be worked on by different use of the ISP.

11. Computation of heights / orthonormalization

The unwrapped interferometric phase together with the precision baseline is then used to derive the topographic heights and true ground ranges. Knowing the topographic heights in SAR image geometry allows then to transform the images from SAR coordinates (slant range, azimuth) to orthonormal coordinates. In addition to the height other products such as the backscatter intensity or the coherence map may be resampled to the orthonormal coordinates.

If any of these images should be provided in a specific map projection, another type of transformation is required. To transform images from the slant-range/azimuth geometry of from the provided orthonormal coordinates to an actual map projection the SAR geocoding software which is part of the Differential Interferometry and Geocoding Software (DIFF&GEO) can be used.

To generate a height map the ISP offers the program ***hgt_map***. This program reads in the unwrapped interferometric phase and the (accurate) baseline geometry to generate a height map and ground range map in slant range geometry.

To finally convert a height map image from slant range to ground range coordinates (i.e. orthonormalization) the program ***res_map*** must be used. This program uses the interferometrically derived terrain heights (in slant range geometry) and true ground ranges (in slant range geometry) to derive a rectified height map. Additionally an intensity image can be orthorectified. The user can define the pixel spacing as well as the number of samples for the least square estimate when converting from slant range to ground range geometry.

The height map can be displayed with the DISP programs ***dishgt*** or ***disshd*** (for shaded relief). To save the height map to an image file in SUNraster / bmp format use either the program ***rashgt*** or ***rasshd*** for shaded relief. For ***dishgt*** and ***rashgt*** the user can specify the height difference for the color cycle.

12. Point target tools

The ISP offers tools to estimate parameters of point targets or derive estimates of parameters from point targets. The program ***ptarg_SLC*** allows determining the point target response function. The programs ***ptarg_cal_SLC*** and ***ptarg_cal_MLI*** allow to derive a radiometric calibration factor for SLC images from the response of point targets in the image. For details on the algorithm implemented in the latter program, it is referred to the Reference Manual. Radiometric calibration based on point targets (with low temporal variability) is supported by the program ***radcal_pwr_stat***. The program calibrate a stack of SLC images from which the point targets have been extracted. For details on the selection of the candidate point targets, it is referred to the program entry in the ISP Reference Manual.

13. Tools for split-beam interferometry

Split-beam interferometry refers to the generation of interferograms using SLC data with only a part of the beam. The generation of two SLC images with different squint angles (i.e., one looking forward and the other looking backward) from an SLC dataset is supported by the program ***sbi_filt***. SLC data obtained with this program can then be used to form two interferograms, from the forward-squinted beam and the backward-squinted beam respectively. The difference in phase of these two interferograms can be shown to be related to the along-track (azimuthal) displacement (Bechor and Zebker, 2006).

A split-beam interferogram can be obtained and unwrapped with the tools of the ISP module after which the phase can be converted to azimuth offset with the program ***sbi_offset*** (see also Reference Manuel of this program for details). The offset is given in meters and represents a shift along the radar velocity vector. A positive offset represent as shift in the same direction as the radar velocity vector.

14. Additional tools

If a stack of co-registered 2-D data files in floating format is available the average image can be computed with the program ***ave_image***. This program can be applied to any kind of image type in floating point format (SAR intensities, amplitudes, unwrapped phases, interferometric DEM heights). The aim is to reduce noise with respect to the original individual images.

The program ***subtract_phase*** is a general-purpose program that allows subtraction of phase image from complex interferogram.

The program ***phase_slope*** can be used to determine the phase slopes in range and azimuth direction.

The program ***af_SLC*** allows obtaining an estimate of the along-track velocity. This information can serve as quality flag for the focusing of the SAR image.

For PALSAR ScanSAR (Wide Beam) mode, it is possible to mosaic beams together to form a single image with the program ***mosaic_WB***. The program requires a list with two columns, containing the names of the files to be mosaiced and the corresponding ISP SLC parameter files. The user has the option to feather the radiometry in areas of overlap between adjacent beams. This is helpful for analysis of SAR intensity images. Splitting of an image to fit the geometry of the PALSAR ScanSAR beams is instead supported by the program ***split_WB***. This is helpful for example for dividing a DEM into the different beams in order to allow differential interferometry on each beam.

To concatenate two SLC images and form a long SLC strip, use the program ***SLC_cat***. The program also generates the corresponding SLC parameter file. To obtain the parameters that guide the concatenation, an offset polynomial must be obtained at first. The offset polynomial information is obtained with the procedure described in the related Section in this document. Doppler centroid information in the ISP parameter files is used to perform the interpolation. The same approach applies to two MLI images that should be concatenated. The program that supports concatenation of MLI images is called ***MLI_cat***. For details on the processing sequence, it is referred to the entry of ***SLC_cat*** in the ISP Reference Manual.

References

Bechor, N., H. Zebker, "Measuring two-dimensional movements using a single INSAR pair", Geophsical Resarch Letters, Vol. 33, L16133, 2006, doi:10.1029/2006GL026883

Rosen, P. A., Werner, C. W. and Hiramatsu, A., Two-dimensional phase unwrapping of SAR interferograms by charge connection through neutral trees, Proceedings IGARSS'94, Pasadena, 8-12 August, 1994.

Werner, C., Wegmüller, U., Strozzi, T. and Wiesmann, A., Processing strategies for phase unwrapping for INSAR applications, Proceedings.EUSAR 2002, Cologne, 4-6 June, 2002.

Processing examples

In this part a list of processing examples dealing with interferometric processing and offset tracking are presented.

It should be remarked that parameter values provided in the processing examples cannot be considered valid for all cases. It is possible that one or more values might have to be adapted to the specific case being processed. It is advised to look carefully at the messages printed on stdout when running each individual program. For assistance please get in contact with us (gamma@gamma-rs.ch).

Example A – Interferometric processing
Processing sequence for the generation of an interferogram, phase unwrapping and generation of a height map

Example B – Generation of a calibrated SAR intensity image
Processing sequence for derivation of calibrated SAR intensity images

Example C – Offset tracking
Processing sequence for offset tracking using SLC images

A. Interferometric processing

In this Example we illustrate how to generate an interferogram from two SLC single look complex images.

Processing consists of the following steps

- Processing setup
- Copy data to disk
- Create the ISP processing parameter file
- Manipulation of orbits (optional)
- Computation of offsets between SLCs (initial and precise estimation)
- Co-registration of SLCs
- Computation of the interferogram
- Estimation of the baseline
- Curved Earth phase trend removal (“flattening”)

After this we show how to obtain the unwrapped phase from a complex valued interferogram.

The unwrapped phase will then be inverted to obtain an interferometric height map. Conversion from slant to ground range is also illustrated.

For the processing an ERS-1/2 tandem SAR data set acquired over Las Vegas on 23-24 May 1996 is used. The ERS-1 image was acquired along orbit 25394 on 23 May 1996. The ERS-2 image was acquired along orbit 05721 on 24 May 1996. This dataset is available on the DEMO CD-ROM. The processing from Section A.3 is also supported by an automated processing script in the DEMO-CD scripts directory (run_ISP_LasVegas). The list of commands in this script can be found in the file com_ISP_LasVegas.

The script should be considered as an introduction to scripting and can be used for developing own scripts based on the user’s particular needs. If the script is used for processing, it is strongly recommended to adapt it by selecting the programs actually needed for processing and by critically choosing the values of the parameters required by each individual program. For this purpose it is highly recommended to refer to the Reference Guide.

A.1. Processing setup

First choose a name to be used as scene identifier. This can be the orbit number, the date of data acquisition, track and frame or any name other name you might want to select. For interferometric processing we suggest to create a directory where interferometric processing is done as well as two additional directories where the SLCs eventually can be stored as back-up. To generate directories use the command *mkdir*.

As scene identifier in this example we use the orbit numbers. Interferometric processing will be run in a directory called insar.

```
mkdir insar ; cd insar
```

For the interferometric processing of an entire image frame the disk should have a minimum of 3-4 GB free space. There should also be 1-2 GB space available for the two SLCs.

A.2. SLC pre-processing / generation of ISP parameter file

The starting point for generation of an interferogram is 2 Single Look Complex (SLC) images. Pre-processing of the SLC data consists in copying the data to the working directory (if necessary), generation of the ISP SLC parameter file and, eventually, manipulation of the orbital state vectors.

SLC data can be read from CDROM (or tape) in processed form from a processing facility or generated locally using the GAMMA MSP module.

If the data is obtained through a processing facility, the dataset consists of a number of files. The most important are the image data itself (typically containing in the filename the string “dat” or “img”) and the leader file including all metadata describing the image data. This file typically contains in the file name the string “ldr” or “lea”. It should be noticed that ENVISAT ASAR data are provided as one file.

If the data is provided on CD-ROM copy them to the SLC directories.

If the data is provided on tape by a processing and archiving facility (ESA-DPAF for example), reading of SLC data is supported by a number of shell scripts (see Section 2). The procedure has to be repeated for each of the two images. Before reading the tape, be sure to be in the directory where the SLC has to be stored. For example for an ERS image with scene identifier 8069 (orbit number) the command line would look as follows

```
ERS_ESA_SLC 8059 /dev/rmt/0mn
```

If the SLC data has been obtained with the MSP module, copy them to the respective SLC directories.

The first step of processing is to put the original dataset in the format used by the GAMMA software. Depending on the sensor and on the data type, there are several programs to read in the original image dataset and to obtain the image data file and the corresponding ISP parameter file (see Section 2).

For SLC data processed with the Gamma MSP the ISP SLC parameter file is generated using the program ***par_MSP***.

The dataset considered in this example is already provided in the GAMMA format. In order to obtain it we used the program ***par_ESA_ERS***. Here we show the command line that allowed obtaining the SLC and the ISP SLC parameter file from the image data file (DAT*) and the SAR leader file (LEA*).

```
par_ESA_ERS LEA_01.001 25394.slc.par DAT_01.001 25394.slc
```

Once each SLC is prepared, we suggest copying the SLC file and the corresponding ISP SLC parameter file to the directory for interferometric processing and change directory to this one.

The ISP SLC parameter file can be viewed with a text editor or with the command more

```
more 25394.slc.par
```

The image consists of 2500 pixels in range and 9000 pixels in azimuth.

A.2.1. Manipulation of orbital state vectors

Typically the orbital data provided by the processing facilities are not fully accurate. State vectors can be improved for a number of sensors, i.e. ERS and ENVISAT, as discussed in Section 2. If the update is not possible but the number of state vectors is small (less than 3), it is possible to interpolate to increase their number.

Updating of the orbital state vectors in the MSP processing parameter file is required for correct geocoding and if the SLC will be used for interferometry.

This operation has already been done for the data we are working with. Below we illustrate how this is performed.

If the PRC precision orbits are preferred, the program ***PRC_vec*** must be used. The command line in this example looks as follows:

```
PRC_vec 05721.slc.par PRC_960519_05652_rev2 5

PRC_vec 25394.slc.par PRC_960521_25357_rev2 5
```

where the files PRC* contain the precise orbital information. These files must be available either in the working directory as in the example or in a directory on the computer. In this case the full path to the directory containing the file has to be provided. In this example the file was available in the working directory and we have chosen to generate 5 state vectors. The time spacing is 30 seconds.

If the DELFT precision orbital data records are preferred instead, the program ***DELFT_vec2*** must be used. In this example the command line would looks as follows:

```
DELFT_vec2 05721.slc.par ~/Delft/ers2/dgm-e04

DELFT_vec2 25394.slc.par ~/Delft/ers2/dgm-e04
```

where ~/Delft/ers1/dgm-e04 and ~/Delft/ers1/dgm-e04 indicate the path to where the ODR files are stored for ERS-1 and ERS-2 respectively. In this way the program reads the arclist file and automatically identifies the ODR files to be used for extracting the orbital information. Alternatively, the user could have defined directly at the command line the needed ODR files (including path), which requires that these have been identified by the user in the arclist file. The program will generate 7 state vectors (default).

A.3. Initial offset estimation

Two SLC forming an interferometric image pair present a relative shift. This can be viewed with the DISP program ***dis2SLC*** (important: the two images must have the same width and length)

```
dis2SLC 05721.slc 25394.slc 2500 2500
```

In order to combine the two images interferometrically, the relative offsets have to be estimated and one image must be resampled to the other.

In this example we will consider the ERS-2 image, 05721.slc, to be the reference image, to which the ERS-1 image, 25394.slc, will be resampled.

To compute the offsets between SLC, the first step consists of creating the ISP processing/offset parameter file using the program ***create_offset***. This file contains information used in the interferometric processing, such as file dimensions, geometric parameters, and the registration offset polynomials:

```
create_offset 05721.slc.par 25394.slc.par 05721_25394.off 1
```

The last input variable indicates the registration algorithm that will be used (1 for registration based on image intensity cross-correlation, 2 for fringe visibility method). This variable can also be omitted if the intensity cross-correlation algorithm is used, since this is the default method.

Initial range and azimuth offsets between the two SLC images are estimated either manually by the operator (To do this the operator is supported with the SLC image display program ***disSLC***) or automatically using the programs ***init_offset_orbit*** and ***init_offset***. With ***init_offset_orbit*** a first guess of the offsets can be obtained based on orbital information. This first guess can then be improved with ***init_offset*** which determines the initial offsets based on the cross-correlation function of the image intensities. In order to avoid ambiguity problems and still to achieve an accurate estimates ***init_offset*** may first be run with multi-looking, followed by a second run at single look resolution (in this example 1 and 5 are the multi-look in the first run). In this example the command list looks as follows:

```
init_offset_orbit 05721.slc.par 25394.slc.par 05721_25394.off

init_offset 05721.slc 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off
1 5

init_offset 05721.slc 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off
1 1
```

Each time the initial offset estimates are written to *.off file and are then used in following run as initial guess.

The user can specify position and size of the area used to compute the cross-correlation between images, from which the initial estimate of the offsets will be derived. The user can also set the threshold on SNR intensity correlation for accepting/rejecting the offset estimates. As default the subset is put in the middle of the image with a dimension of 1024x1024 pixels. If the SNR intensity correlation is below the threshold, the estimated offsets are not written to the ISP offset parameter file. It is possible that by changing position and/or size of the subset, an estimate corresponding to a SNR intensity correlation above the threshold is obtained. To identify the location of the subset, it is suggested to look at one of the SLCs and look for areas with contrast (e.g. topography, urban areas, fields).

Another possibility is to run ***init_offset*** prior to the creation of the ISP processing parameter file. In this case ***init_offset*** also creates the ISP processing parameter file. This has the advantage that the limits for the area to be used for the fine registration can be determined taking into account the already available initial registration offsets.

A.4. Precise estimation of offset polynomials

A.4.1. Estimation of offsets

The ISP offers two methods for the precision (sub-pixel accuracy) estimation of the range and azimuth offset polynomials.

The first method is based on the image intensity and is implemented in the program ***offset_pwr***. For a large number of image segments this method searches for the range and azimuth offsets resulting in the maximum level of intensity correlation. The method requires at least a minimum of image contrast. It does not depend on the level of coherence between the two SLC images and is computationally efficient. The more important parameters for the search are number and size of the offset estimation windows in range and azimuth, and the offset estimation threshold. In the example we report some general values.

ISP processing parameter:	**Example for appropriate values for use of offset_pwr**
offset_estimation_range_samples:	32
offset_estimation_azimuth_samples:	32
offset_estimation_window_width:	64
offset_estimation_window_height:	64
offset_estimation_threshold::	7.0

The second method is based on the complex valued data and is implemented in the program ***offset_SLC***. For a large number of image segments this method searches for the range and azimuth offsets resulting in the maximum level of coherence. The method requires at least for a number of locations in the image a sufficient level of coherence between the two SLC. The method also works for very low image contrast. Appropriate values for the more important parameters for this search are:

ISP processing parameter:	**Example for appropriate values for use of offset_SLC**
offset_estimation_range_samples:	16
offset_estimation_azimuth_samples:	16
offset_estimation_window_width:	16
offset_estimation_window_height:	16
offset_estimation_threshold::	3.0

In this example we use ***offset_pwr*** as follows

```
offset_pwr 05721.slc 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off
offs snr 128 128 offsets 1 8 8 7.0
```

If we had used ***offset_SLC***, the command line could have looked as follows:

```
offset_SLC 05721.slc 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off
offs snr 128 128 offsets 1 8 8 3.0
```

Both approaches determine a field of registration offsets (file *.offs) and a corresponding quality measure field (*.snr). The *.offs file is in FCOMPLEX format, the real and the imaginary part expressing for each image chip the range and the azimuth offset. Offsets can also be saved as text file for further analysis (file *.offsets). For simplicity the file name for the offsets files and the snr file has been omitted. See the ISP Reference Manual for further examples of command lines.

The number of image chips and their size has been adapted to the specific case considered here. The aim is to distribute uniformly all over the image the image chips and include enough features to have in each a sufficient contrast. The user can see what happens to the number of offsets accepted/rejected and the SNR values by increasing or decreasing the number of chips and their size, as well as by changing the SNR threshold.

If very few or even no estimates with good quality are achieved the registration will fail. This may be a result of

- poor initial estimate of the offsets
- too little contrast in the image (*for offset_pwr*)
- too large baseline (*for offset_SLC*)
- too much change (water, forest, high vegetation, long acquisition time interval)

A.4.2. Generation of offsets polynomial

Based on the field of registration offsets and the quality measure of the offsets, the bilinear registration offset polynomial is then determined using a least squares error method. This is implemented in the program ***offset_fit***. In this example the command line looks as follows:

```
offset_fit offs snr 05721_25394.off coffs coffsets 7.0 3 0
```

In ***offset_fit*** an iterative procedure is applied to reject offsets far away from the regression. Similarly to the offsets determined with ***offset_pwr*** or ***offset_SLC***, the culled offsets are saved to both a binary file in FCOMPLEX format (*.coffs) and a text file (*.coffsets). For simplicity the file names have been omitted in this example.

The resulting registration offset polynomials for range and azimuth offset are written to the ISP offset parameter file. The user can judge the quality of the registration between the images by looking at the estimated standard deviation of the offsets in range and azimuth. Typically co-registration errors of the order of 1/5th of a pixel or more lead to non-negligible decorrelation.

A.4.3. Improvement of offsets polynomial

One way to try to improve the co-registration polynomial is to re-run the sequence ***offset_pwr*** (or ***offset_SLC***) – ***offset_fit*** by varying one or more of the following parameters until a satisfactory result is obtained:

- the number of image chips,
- the size of the image chips,

- the threshold
- the number of coefficients in the co-registration polynomial

Notice that since the ISP offset parameter file is each time overwritten with the updated polynomial offsets, it is recommendable to keep a copy of the original offset file (and any other offset file generated during the improvement stage). In this example the command line for an iteration can look as follows

```
offset_SLC 05721.slc 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off
offs snr 4 4 offsets 2 24 24 3. 32

offset_fit offs snr 05721_25394.off coffs coffsets 3.0 4 0
```

The precision estimation of the registration polynomial is done using ***offset_SLC*** with many (24x24) small search windows (32x32 pixel chips, 4x4 pixel search window, 2x oversampling) .

A.5. Computation of the interferogram

Once the co-registration polynomial has been obtained, co-registration and generation of the normalized interferogram can be performed. The ISP supports two different approaches.

The first approach (first for historical reasons) directly calculates the normalized interferogram and the registered intensity images from the two (non-registered) SLC and the registration function contained in the ISP offset parameter file without writing the registered SLC to a file. This approach is implemented in the program ***interf_SLC***. It requires less disk space and is faster because the interpolation and common spectral band filtering can be applied in a single filtering step. In our example the command line looks as follows:

```
interf_SLC 05721.slc 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off
05721.mli 25394.rmli 05721_25394.int 1 5 - - - - 1 1
```

The two SLC images are first co-registered then filtered for common-band in range and azimuth direction (default values 1 and 1 given at command line), and then one image is multiplied with the conjugate complex of the second image and normalized. Multi-looking in range (1-look in the example given) and azimuth (5-looks) is applied in the interferogram computation. The resulting interferogram (*.int) consists of complex values with the magnitude corresponding to the 5-look interferometric correlation and the phase to the interferometric phase. In addition the registered intensity images (*.mli and *.rmli, where rmli stands for resampled MLI) are calculated applying the same multi-looking as for the interferogram. The filtering-functions are accounted for, so that the image intensities remain unchanged.

The second approach resamples in a first step the second SLC to the reference geometry (i.e. the geometry of the first SLC). The two co-registered SLC are then used to create the interferogram. This procedure is recommended.

Resampling is performed with the program ***SLC_interp***. In our example the command line looks as follows:

```
SLC_interp 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off 25394.rslc
25394.rslc.par
```

The advantage of using this approach is that the co-registered SLC (*.rslc) is saved to a file and therefore can be compared to the reference SLC. To appreciate the co-registration precision the user can display the two SLCs simultaneously with the DISP program ***dis2SLC***.

```
dis2SLC 05721.slc 25394.rslc 2500 2500
```

If the pair of SLCs is affected by residual offsets, the co-registration procedure in Section A4 can be run this time between the reference SLC and the resampled SLC. The residual offsets can then be added to the initial offsets to obtain a new file of refined offsets. This file is then used to co-register to the reference SLC the other SLC. The refinement procedure can be repeated in case. Below the sequence for improving offsets is reported (not applied in this example because the standard deviation of the offsets obtained with the first iteration has proven to be satisfactory). The first command renames the offset parameter file. Anew offset parameter file with the same extension .off will be generated at the end of the sequence (with the command offset_add). By renaming the initial offset parameter file it will be possible to compare the offset polynomial before and after the refinement.

```
mv 05721_25394.off 05721_25394.off0

create_offset 05721.slc.par 25394.rslc.par 05721_25394.off_res

offset_pwr 05721.slc 25394.rslc 05721.slc.par 25394.rslc.par
05721_25394.off_res offs_res snr_res - - - 2 - -

offset_fit offs_res snr_res 05721_25394.off_res coffs_res coffsets_res 7.0
3

offset_add 05721_25394.off0 05721_25394.off_res 05721_25394.off

SLC_interp 25394.slc 05721.slc.par 25394.slc.par 05721_25394.off 25394.rslc
25394.rslc.par
```

In this case the initial resampled SLC is overwritten.

For the generation of the interferogram the program ***SLC_intf*** is used. Common spectral band filtering is applied in the interferogram generation step. In this example the command line looks as follows:

```
SLC_intf 05721.slc 25394.rslc 05721.slc.par 25394.rslc.par 05721_25394.off
05721_25394.int 1 5 - - 1 1
```

The two SLC images are first filtered in range and azimuth direction (common spectral band filtering, variables 1 and 1 at the end of command line) and then one image is multiplied with the conjugate complex of the second image, and normalized. Multi-looking in range (1-looks in the example given) and azimuth (5-looks) is applied in the interferogram computation. The resulting interferogram consists of complex values with the magnitude corresponding to the 5-look interferometric correlation and the phase to the interferometric phase.

The registered multi-look intensity images (MLI) are calculated in a separate step applying the same multi-looking as for the interferogram. These will be used for the computation of the coherence. For this the program ***multi_look*** is used, separately for each image. In our example the command line looks as follows:

```
multi_look 05721.slc 05721.slc.par 05721.mli 05721.mli.par 1 5

multi_look 25394.rslc 25394.rslc.par 25394.rmli 25394.rmli.par 1 5
```

The interferogram can be displayed with the DISP program ***dismph*** as follows

```
dismph 05721_25394.int 2500
```

Another way to display the interferogram is to display the interferometric phase in color and the backscatter intensity of image 1 as image brightness. This can be obtained with the DISP program ***dismph_pwr***.

```
dismph_pwr 05721_25394.int 05721.mli 2500
```

SUNraster versions of the interferogram with the coherence and the backscatter intensity as background can be obtained with the program ***rasmph*** and ***rasmph_pwr*** respectively as follows

```
rasmph 05721_25394.int 2500 - - 4 4 - - - 05721_25394.int.ras

rasmph_pwr 05721_25394.int 05721.mli 2500 - - - 4 4 - - - 05721_25394.int_mli.ras
```

To obtain a SUNraster file of reasonable size, averaging factors of 4 and 4 in range and azimuth have been used.

Figure A1 shows the interferogram as obtained with ***rasmph***. One color cycle corresponds to 2π interferometric phase. The image brightness shows the interferogram magnitude, which corresponds to the 5-look estimate of the coherence.

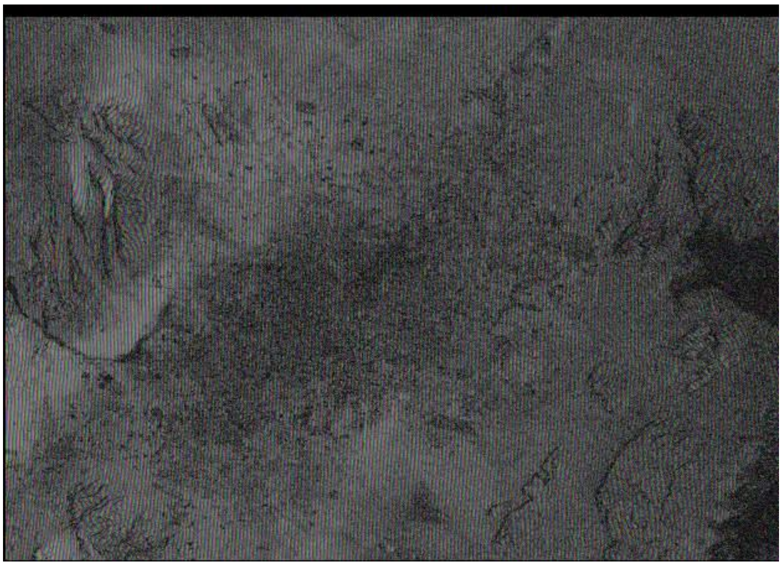


Figure A1. Interferogram with interferometric correlation as background. The image is 2500 pixels wide and 1800 pixels long, i.e. approximately 50 km wide and 36 km wide. The interferometric fringes have a high frequency due to the long baseline.

Figure A2 shows the interferogram display obtained with ***rasmph_pwr***. One color cycle corresponds to 2π interferometric phase. The image brightness shows the 5-look backscatter intensity.

Both images show that the "flattening" is not perfect since residual low-frequency fringes can be observed. This is due to small errors in the state vectors.

To display one MLI image use the program ***dispwr***. To display the two co-registered image use the program ***dis2pwr***.

```
dispwr 05721.mli 2500
```

```
dis2pwr 05721.mli 25394.rmli 2500 2500
```

SUNraster versions of each MLI image can be obtained with the program ***raspwr*** as follows

```
raspwr 05721.mli 2500 - - 4 4 - - - 05721.mli.ras
```

```
raspwr 25394.rmli 2500 - - 4 4 - - - 25394.rmli.ras
```

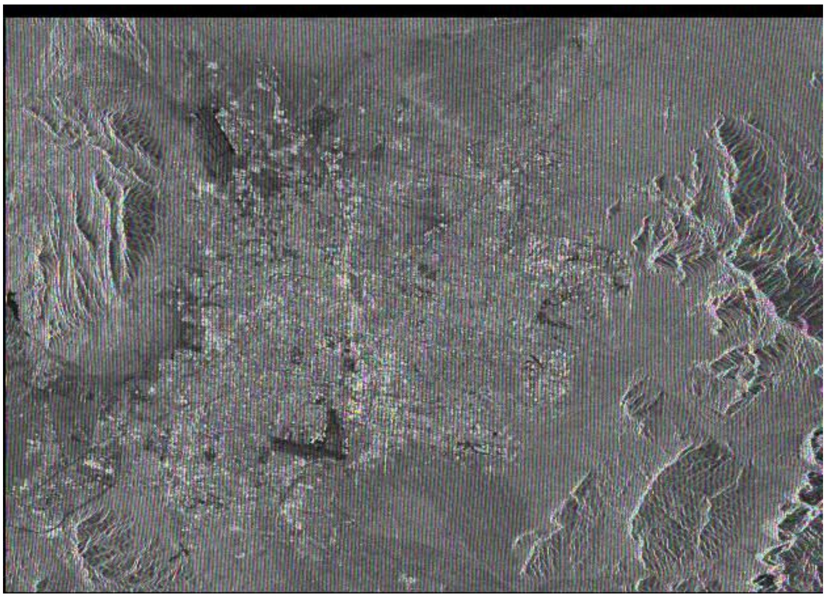


Figure A2. Interferogram with backscatter intensity as background. The image is 2500 pixels wide and 1800 pixels long, i.e. approximately 50 km wide and 36 km wide. The interferometric fringes have a high frequency due to the long baseline.

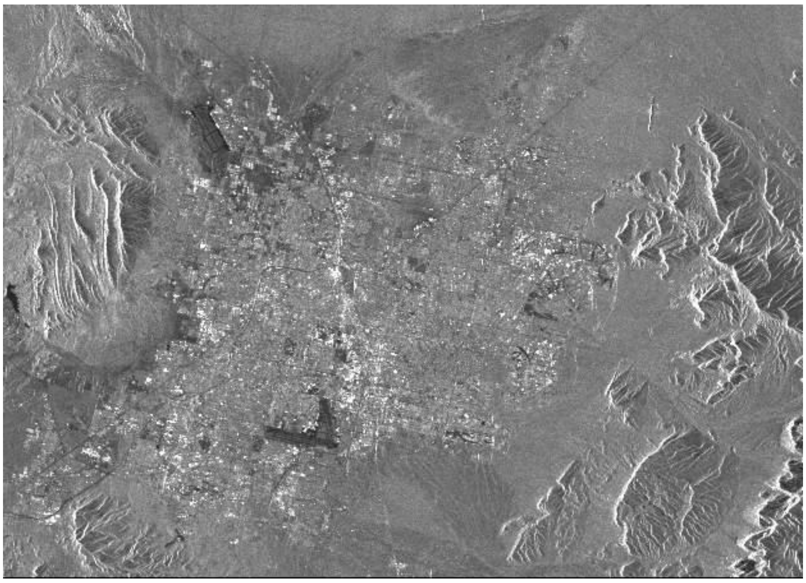


Figure A3. MLI image 05721.mli. The image is 2500 pixels wide and 1800 pixels long, i.e. approximately 50 km wide and 36 km wide.

Figure A3 shows the MLI image of the reference SLC (i.e. 05721.mli) as obtained with ***raspwr***. The city of Las Vegas can be well recognized in the middle of the image. Topography is visible on the right and on the left side of the image. The mountain ridges are affected by layover (very high backscatter intensity).

A.6. Estimation of interferometric baseline

There are two ways to estimate the interferometric baseline at this point.

The first method calculates the baseline from the orbital information, i.e. the state vectors given in the ISP SLC parameter file. This method is implemented in the program ***base_orbit***. The method looses in precision when the orbital state vectors are not accurate.

The second method estimates the baseline from the fringe rate of the normalized interferogram using FFT. This method is implemented in the program ***base_est_fft***. The method is applied in particular if no state vectors or state vectors of rather poor quality are available. The baseline should be estimated in an area where the interferometric phase consists purely of curved Earth phase.

The two methods are also available in the program ***base_init***.

All programs generate a baseline text file. Baseline coordinates and change rates (in TCN coordinates) are written to the first two lines of the baseline file.

In this example we use ***base_init*** as follows.

```
base_init 05721.slc.par 25394.slc.par 05721_25394.off 05721_25394.int 05721_25394.base 2 1024 1024
```

In this example we chose estimate the parallel and perpendicular component of the baseline using the orbital information and the fringe rate respectively (flag after baseline file set to 2). The FFT is estimated in a window of 1024x1024 pixels.

To obtain the parallel and perpendicular component of the baseline and see how they change along- and across-track, use the program ***base_perp***. The result can be printed to stdout (e.g. screen) or saved to a text file. In our example the command line for saving the information to file 05721_25394.base.perp looks as follows

```
base_perp 05721_25394.base 05721.slc.par 05721_25394.off > 05721_25394.base.perp
```

A.7. Curved Earth phase trend removal ("flattening")

The interferogram contains several phase components. The phase trend expected for a smooth curved Earth (ellipsoid, height constant) can be removed from the interferogram using the program ***ph_slope_base***. In our example the command line looks as follows:

```
ph_slope_base 05721_25394.int 05721.slc.par 05721_25394.off 05721_25394.base 05721_25394.flt
```

The phase of the resulting "flattened" interferogram (*.flt) does not change as quickly as the interferogram with the orbital fringes (*.int) facilitating filtering and averaging operations.

As for the interferogram in Figure A1, the flattened interferogram can be displayed with the DISP program ***dismph*** as follows

```
dismph 05721_25394.flt 2500
```

Another way to display the interferogram is to display the interferometric phase in color and the backscatter intensity of image 1 as image brightness. This can be obtained with the DISP program ***dismph_pwr***.

```
dismph_pwr 05721_25394.flt 05721.mli 2500
```

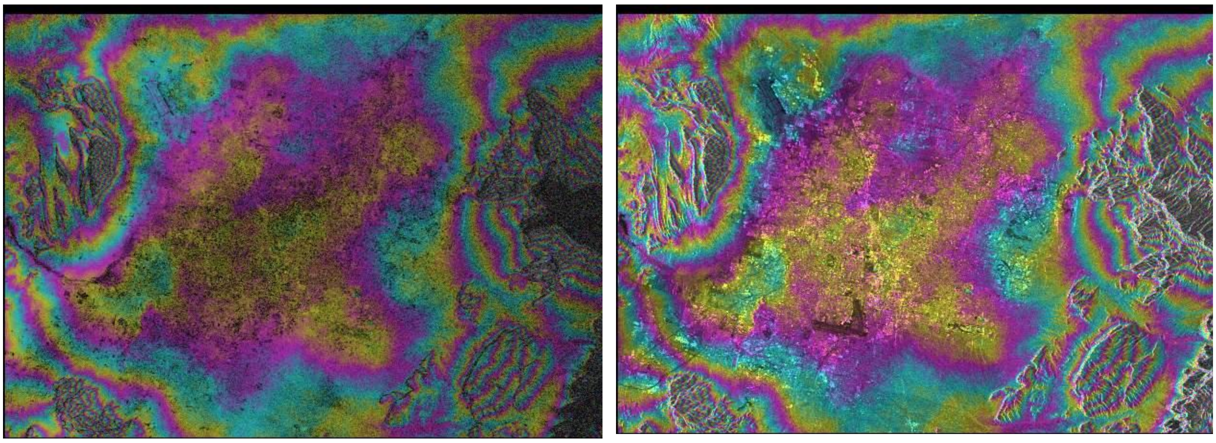
SUNraster versions of the flattened interferogram with the coherence and the backscatter intensity as background can be obtained with the program ***rasmph*** and ***rasmph_pwr*** respectively as follows

```
rasmph 05721_25394.flt 2500 - - 4 4 - - - 05721_25394.flt.ras
```

```
rasmph_pwr 05721_25394.flt 05721.mli 2500 - - - 4 4 - - - 05721_25394.flt_mli.ras
```

To obtain a SUNraster file of reasonable size, averaging factors of 4 and 4 in range and azimuth have been used.

Figure A4a shows the flattened interferogram as obtained with ***rasmph***. One color cycle corresponds to 2π interferometric phase. The image brightness shows the interferogram magnitude, which corresponds to the 5 look estimate of the coherence. Figure A4b shows the flattened interferogram as obtained with ***rasmph_pwr***. One color cycle corresponds to 2π interferometric phase. The image brightness shows the 5-look backscatter intensity. Figure A4 shows that the "flattening" seems to have worked quite well since no systematic fringes due to imperfect compensation of the flat-Earth fringes can be observed.



(a) (b)

Figure A4. Flattened interferogram with (a) correlation as background, (b) backscatter intensity as background.

A.8. Estimation of the degree of coherence

To estimate the degree of coherence (interferometric correlation), here simply referred to as coherence, the two MLI images and a phase correction term are needed. The most suitable image that can serve as phase correction term is the differential interferometric phase. This requires that a DEM is available. By subtracting the topographic phase component simulated from the DEM the differential interferometric phase can be obtained. If a DEM is not available, as phase correction term the "flattened" interferogram can be used.

Coherence is estimated using the program cc_wave. In this example the command line looks as follows

```
cc_wave 05721_25394.flt 05721.mli 25394.rmli 05721_25394.cc 2500 5 5 1
```

For the coherence estimation we used a 5x5 estimation window i.e. 75 looks are used taking into account that the MLI images have been obtained using a 1x5 multi-look factor. Using the program cc_wave the estimation is done with a uniform, linear or Gaussian weighting function. In this example the weight decreases linearly with increasing distance for the center pixel (variable value 1). Non-integer window sizes may be used (the program uses always odd integer value sized windows, but with the weights set to represent the non-integer window sizes.)

To display the coherence use the DISP program ***discc***. The program allows combining in color the coherence image with the backscatter intensity of image 1 (as image brightness). In our example the two displays are obtained as follows

```
discc 05721_25394.cc - 2500
```

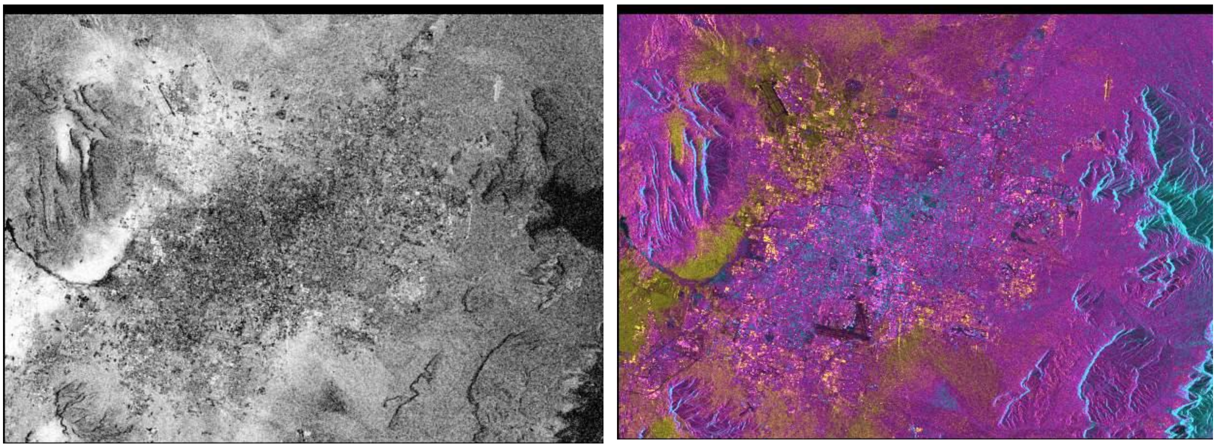
```
discc 05721_25394.cc 05721.mli 2500
```

SUNraster versions of the coherence image without and with the backscatter intensity as background can be obtained with the program ***rascc*** as follows

```
rascc 05721_25394.cc - 2500 - - - 4 4 - - - - - 05721_25394.cc.ras
```

```
rascc 05721_25394.cc 05721.mli 2500 - - - 4 4 - - - - - 05721_25394.cc.ras
```

To obtain a SUNraster file of reasonable size, averaging factors of 4 and 4 in range and azimuth have been used. Figure A5 shows the coherence as grayscale and combined with the backscatter intensity. Coherence is rather high except for some areas in the middle of the image (city of Las Vegas) and at the right side where strong decorrelation appears. This is due to vegetation and steep slopes. Thin dark features correspond to areas of layover with very low coherence.



(a) (b)

Figure A5. Interferometric coherence as grayscale (a) and combined with the backscatter intensity (b). Color corresponds to interferometric coherence, brightness to backscatter intensity.

At this point of the processing the following files have been obtained

File	**File name in example**
Reference SLC	05721.slc
SLC parameter file of reference SLC	05721.slc.par
Resampled SLC of file 2	25394.rslc
SLC parameter file of resampled SLC of file 2	25394.rslc.par
ISP offset parameter file	05721_25394.off
ISP baseline file	05721_25394.base
Normalized complex valued interferogram	05721_25394.int
Flattened complex interferogram	05721_25394.flt
Image intensity of reference file 1	05721.mli
Image intensity of resampled file 2	05721.rmli
Coherence image	05721_25394.cc

A.9. Interferogram filtering

Any interferogram has two central problems that need to be taken care of when moving to the most important step in SAR interferometry, i.e. phase unwrapping:

a) phase noise
b) discontinuity of the interferometric phase (caused for example by lay-over)

The first problem can be addressed by filtering. Caution has to be used if the interferometric phase cannot be expected to be constant over the area used for the filtering. The ISP includes several program to filter the interferometric phase, from simple band-pass filters to more advanced filters adaptive to the local slope (***adapt_filt*** and ***adf***). Multi-looking is another alternative which has the additional advantage that the number of samples is dramatically reduced. The big draw back in multi-looking is, of course, that the sampling is reduced. This is acceptable for areas with low phase gradients, but a problem in areas with large phase gradients. For multi-looking a complex interferogram use the program ***multi_cpx***.

In this example we illustrate the use of the program adf. The program ***adf*** applies a non-linear filter based upon the power spectrum to the data. Typical exponent parameters for ***adf*** lie in the range of .2 to .5. For phase unwrapping it is necessary to have an estimate of the local fringe quality after filtering. Because of the degree of filtering varies over the image, ***adf*** estimates the local phase standard deviation after filtering and converts this to an effective correlation. This 'correlation' file is specified on the command line.

The command line in our example looks as follows

```
adf 05721_25394.flt 05721_25394.flt_sm 05721_25394.smcc 2500 .5 32 7 8 0 0
.25
```

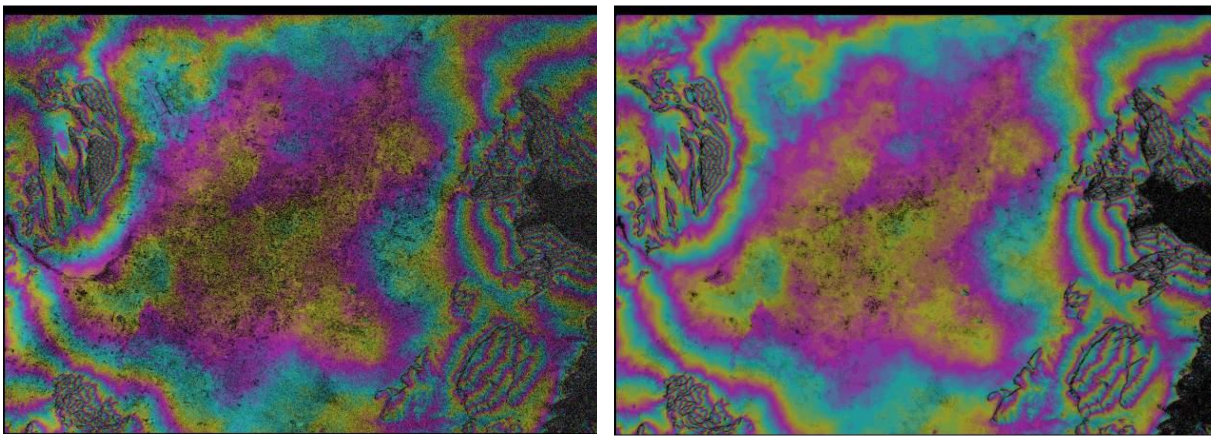
The filtered interferogram (*.flt_sm) has the same format as the unfiltered interferogram. In addition the correlation corresponding to the filtered interferogram is generated (*.smcc). To display the filtered interferogram the DISP program ***dismph*** can be used. To compare with the unfiltered interferogram the program ***dis2mph*** can be used.

```
dismph 05721_25394.flt 2500

dis2mph 05721_25394.flt 05721_25394.flt_sm 2500 2500
```

SUNraster versions of the flattened interferogram with the coherence and the backscatter intensity as background can be obtained with the program ***rasmph*** and ***rasmph_pwr*** respectively as follows

```
rasmph 05721_25394.flt_sm 2500 - - 4 4 - - - 05721_25394.flt_sm.ras

rasmph_pwr  05721_25394.flt_sm  05721.mli  2500  -  -  -  4  4  -  -  -
05721_25394.flt_sm_mli.ras
```

To obtain a SUNraster file of reasonable size, averaging factors of 4 and 4 in range and azimuth have been used. Figure A6 shows the flattened interferogram before and after filtering (with correlation as background). The filtered version is much less affected by noise. Close fringes in areas of steep slopes have been preserved.



(a) (b)

Figure A6. Unfiltered (a) and filtered(b) flattened interferogram.

A.10. Phase Unwrapping

In the normalized interferogram the interferometric phase is only known modulo 2π. As a consequence the interpretation of the interferometric phase has to be preceded by the determination of the correct multiple of 2π. This is done in the procedure called phase unwrapping.

Below we illustrate how to unwrap the phase of an interferogram with the two methods presented in Section 9, i.e. based on the branch-cut region growing algorithm and on minimum cost flow (MCF) techniques and a triangular irregular network (TIN).The later method is preferred in most cases since in requires significantly less user interaction.

The method based on the branch-cut algorithm is implemented in the DEMO-CD script, without bridges.

A.10.1. Phase unwrapping with branch-cut region growing algorithm

Phase unwrapping using branch-cut region growing phase unwrapping consists of the following steps

Processing steps	**Related program(s)**
Masking low correlation areas	***corr_flag***
Masking neutron areas (optional)	***neutron***
Determination of residues	***residue***
Connection of residues through neutral trees	***tree_cc*** or ***tree_gzw***
Unwrapping of interferometric phase	***grasses***

In the case that image sections are disconnected through areas of low coherence (as a result of a water surface, for example) or layover so called bridges have to be built between the disconnected areas in order to continue the phase unwrapping. This is done with the following sequence which may be repeated several times:

Processing steps	**Related program**
Display wrapped/unwrapped areas	***rastree***
Use "bridges" to continue phase unwrapping	***bridge***

In the following these steps are described in more detail.

A.10.1.1. Masking low correlation areas

Areas of low interferometric correlation (below a user-specified threshold) are not included in the phase unwrapping due to too high phase noise. Such areas of low correlation are masked using the corr_flag program. In this example the command line looks as follows:

```
corr_flag 05721_25394.smcc 05721_25394.flag 2500 0.3
```

With this program the phase unwrapping flag file is generated (*.flag). All pixels with correlation below the coherence threshold 0.3 are indicated with 0, the rest is flagged with 1. The file will be updated in the next processing steps.

To display the flag file we use the DISP program ***disflag***. The display indicates why areas have been masked out.

```
disflag 05721_25394.flag 2500
```

A.10.1.2. Generation of neutrons (optional)

To identify neutrons, use the program neutron. Adding neutrons based on image intensity will cause lay-over to be excluded from unwrapping. The lay-over area is detected simply by backscattering which is more than a user specified factor higher than the average backscattering. The neutrons are written to the phase unwrapping flag file.

Since in our example layover areas are already identified by the low coherence, this step is not considered.

A.10.1.3. Determination of residues

To eliminate residues, i.e. locations around which a closed integral of the phase differences leads to a non-zero result, from phase unwrapping use the program residue. The program marks such locations and updates the flag file. In our example the command line looks as follows:

```
residue 05721_25394.flt_sm 05721_25394.flag 2500
```

To display the updated flag file we run again ***disflag***:

```
disflag 05721_25394.flag 2500
```

A.10.1.4. Connection of residues through neutral trees

To “discharge” positive and negative residues through "neutral trees" either the programs ***tree_cc*** or ***tree_gzw*** can be used. In this example we use the program ***tree_cc***.

```
tree_cc 05721_25394.flag 2500 64
```

This program updates the flag file, which is now ready to be used in the phase unwrapping operation. The phase unwrapping flag file contains the information which area will be unwrapped. During the unwrapping of the interferometric phase no neutral trees will be crossed. Therefore, the result will be independent of the path chosen for the unwrapping.

To display the final flag file we run again ***disflag***:

```
disflag 05721_25394.flag 2500
```

Dark pixels will be unwrapped; pixels in any other color will not be unwrapped. By clicking on a pixel with any other color, it is possible to identify the reason why the pixel has been excluded. To generate a SUNraster version of the flag file, we can use the program ***rastree*** as follows:

```
rastree 05721_25394.flag - - 2500 - - - 05721_25394.flag.ras
```

Figure A7 illustrates the flag file.

Figure A7. Phase unwrapping flag file. Dark areas will be unwrapped. Other areas are excluded from unwrapping.

A.10.1.5. Unwrapping of interferometric phase

To unwrap the interferometric phase, the program ***grasses*** is used. The phase unwrapping is started at a user-defined location (default location is the image center) and continued by region growing for the entire area connected to the starting location avoiding low correlation areas and without crossing neutral trees. In our example the command line looks as follows:

```
grasses 05721_25394.flt_sm 05721_25394.flag 05721_25394.flt_sm.unw 2500 - -
- - 1200 1500
```

To start unwrapping the phase we chose the pixel at position (1200, 1500) which has high coherence and lies within an area of high coherence. The unwrapped interferometric phase (*.flt_sm.unw) is written to a real valued data file which can be displayed, for example, using ***disrmg***. The user can choose whether to display the backscatter intensity image in the background. In our example the command line looks as follows:

```
disrmg 05721_25394.flt_sm.unw 05721.mli 2500 - - - 1
```

We will see that areas excluded by unwrapping appear as dark. As phase scale factor we use 1 which means that consecutive fringes of the same color are separate by 2π radians. The user can set this scale factor to lower values to appreciate large-scale patterns or to higher values to appreciate the rate of change of the unwrapped phase. Contrarily to the flattened interferogram here the phase assumes values also outside of the range $(-\pi,\pi)$.

To generate the SUNraster version of the image use the program ***rasrmg*** (with backscatter intensity as background) and phase scale factor of 0.333 (default):

```
rasrmg  05721_25394.flt_sm.unw  05721.mli  2500  -  -  -  4  4  -  -  -  -  -
05721_25394.flt_sm.unw.ras
```

Figure A8 shows the unwrapped interferometric phase

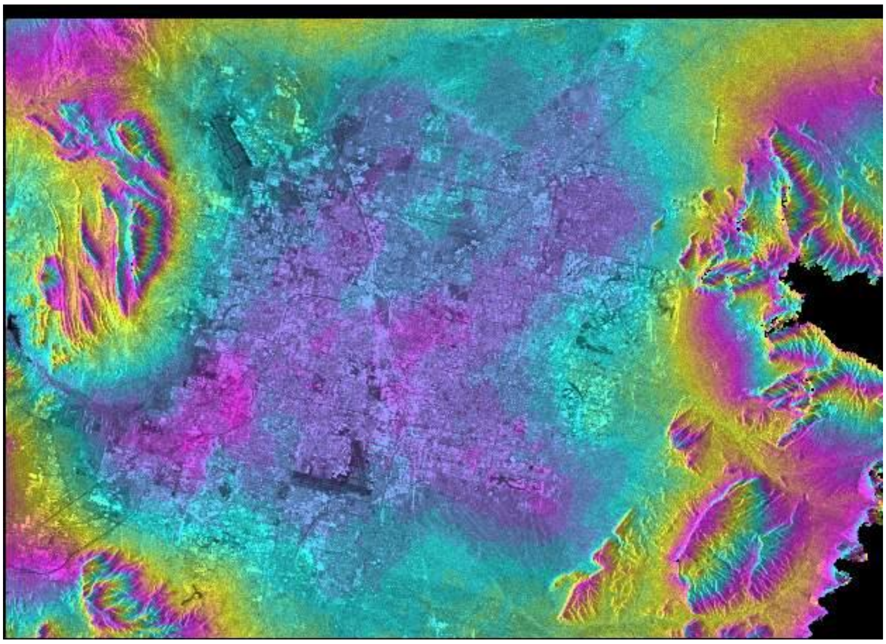


Figure A8. Unwrapped interferometric phase, with backscatter intensity as background. Areas of layover and significant phase noise have been excluded from the unwrapping. Phase display is approximately 6π per color cycle.

Figure A9 illustrates a zoom of the upper right part to show the unwrapped phase together with residues. As phase scale factor we used the default value 0.333. Two consecutive fringes of the same color have a difference of approximately 18 radians.

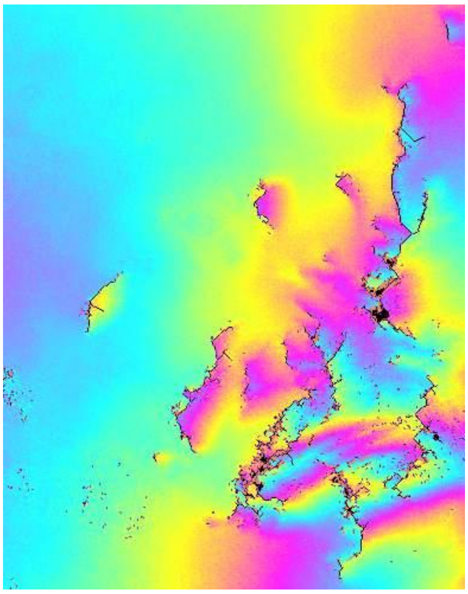


Figure A9. Unwrapped interferometric phase with approximately 6π per color cycle. Also shown are residues (+: red, -: blue) and branch cuts (black). Note that most residues come in +/- pairs. Chains of residues occur along regions of layover.

The full image from which the zoom has been extracted has been obtained with ***rastree*** as follows:

```
rastree 05721_25394.flt_sm 05721_25394.flt_sm.unw 05721_25394.flag 2500 - -
- 05721_25394.flt_sm.unw.tree.ras
```

A.10.1.6. Construction of bridges between disconnected regions

The unwrapping may be continued in areas disconnected from the already unwrapped area by constructing bridges between unwrapped and not unwrapped locations. To construct the bridges the wrapped and unwrapped areas together with the interferogram phase a SUNraster file is generated using ***rastree***:

```
rastree 05721_25394.flag 05721_25394.flt_sm.unw 05721_25394.flt_sm 2500 - -
0.3333 05721_25394.flt_sm.unw.tree.ras
```

It should be noticed that the phase scaling factor should be adapted to the rate of change of the phase, i.e. how much one wants to resolve consecutive fringes. The resulting SUNraster file is then displayed using the program ***disras***:

```
disras 05721_25394.flt_sm.unw.tree.ras
```

Using this display showing which areas are already unwrapped (bright colors) and which ones are not (darker colors) allows editing a text file *.bridges.

In this example there are no particular areas where the unwrapping was hindered, for this reason bridges are not constructed.

A.10.1.7. Unwrapping of disconnected area

The bridges are then used to unwrap the phase in the disconnected areas. For this use the program ***bridge***, which in this example would look as follows:

```
bridge    05721_25394.flt_sm    05721_25394.flag    05721_25394.flt_sm.unw
05721_25394.bridges 2500
```

The same procedure can be repeated with new bridges to unwrap further disconnected areas.

A.10.2. Phase unwrapping with Minimum Cost Flow (MCF) techniques

The basic sequence is very short:

Processing steps	**Related program(s)**
Generation of phase unwrapping validity mask (optional)	***rascc_mask***
Phase unwrapping algorithm using MCF and triangulation	***mcf***

but very often a sequence as the following will be preferred to better optimize robustness, efficiency, and accuracy.

Processing steps	Related program(s)
Generation of phase unwrapping validity mask (optional)	***rascc_mask***
Adaptive sampling reduction for validity mask (optional)	***rascc_mask_thinning***
Phase unwrapping algorithm using MCF and triangulation	***mcf***
Weighted interpolation to fill gaps in unwrapped phase image	***interp_ad***
Use of interpolated unwrapped phase as model to unwrap initial interferogram	***unw_model***

In the following these steps are described in more detail.

A.10.2.1. Generation of phase unwrapping validity mask

A phase unwrapping validity mask, i.e. a SUNraster or bmp file with NULL values for pixels which shall not be considered in the MCF phase unwrapping, is generated using the program ***rascc_mask***. As input file for example a coherence image is used and values below an indicated threshold are set to NULL in the output raster file. The use of a validity mask is optional. It is mainly recommended for data with predominantly high phase noise, as in the case of long-term differential interferograms.

In this example the command line to generate the phase unwrapping validity mask looks as follows

```
rascc_mask 05721_25394.cc 05721.mli 2500 1 1 0 1 1 0.3 0.0 - - - - -
05721_25394.mask.ras
```

Since the validity mask is in SUNraster format, use the program ***disras*** to display it.

```
disras 05721_25394.mask.ras
```

Figure A10 illustrates the phase unwrapping phase validity mask. Areas shown as dark have coherence below the threshold (see e.g. right part of the image). These areas will not be considered during phase unwrapping if the masking for low correlation areas is used.

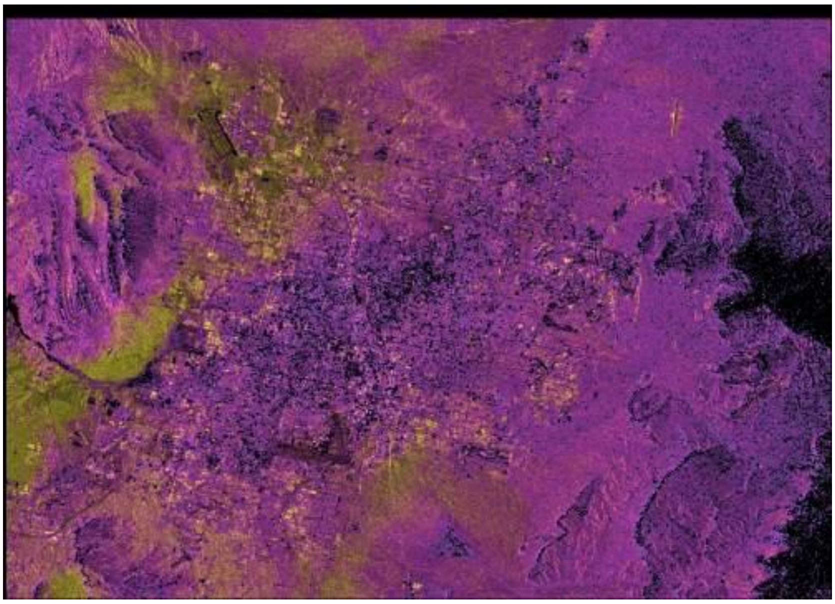


Figure A10. Phase unwrapping phase validity mask. Dark areas are excluded from phase unwrapping.

A.10.2.2. Adaptive sampling reduction for phase unwrapping validity mask

The spatial sampling of the phase unwrapping validity mask can be reduced using the program ***rascc_mask_thinning***. This step is optional. The main reason for such adaptive thinning of the phase unwrapping validity mask is to reduce the number of samples considered in the global optimization step of the Minimum Cost Flow (MCF) phase unwrapping technique. Adaptive thinning of the validity mask can be used if working with large images which have few areas excluded from unwrapping. Thinning a large number of samples for the unwrapping would speed up the unwrapping step without affecting the quality of the result.

Because we are working with a relatively small dataset, this step is not applied in this example. However, if applied, the command line could look as follows:

```
rascc_mask_thinning 05721_25394.mask.ras 05721_25394.cc 2500 05721_25394.mask_thinned.ras 3 0.3 0.4 0.5
```

The program reads the phase unwrapping validity mask and reduces the sampling adaptively to the provided coherence. In this case we have considered applying 3 thinning runs using as coherence thresholds the values 0.3, 0.4 and 0.5.

A.10.2.3. Phase unwrapping

Phase unwrapping of the filtered interferogram is done using the program ***mcf***. The user can set several parameters namely

- use of the coherence image as weight in the determination of the costs
- use of the validity mask to define areas that will not be unwrapped (and are not considered when computing the costs)
- type of triangulation
- subsetting the image to remove borders with zeroes or areas that can negatively affect the unwrapping
- number of patches

In this example we compute the unwrapped phase once with and once without the support of the validity mask. The command lines for unwrapping with and without the validity mask are as follows

```
mcf 05721_25394.flt_sm 05721_25394.cc 05721_25394.mask.ras 05721_25394.flt_sm.mcf.unw0 2500 1 9 18 - 1751 1 1 - 1200 1500

mcf 05721_25394.flt_sm 05721_25394.cc - 05721_25394.flt_sm.mcf.unw 2500 1 9 18 - 1751 1 1 - 1200 1500
```

For processing we have

- used a Delaunay Triangulation,
- offset out areas without values along the borders (range offset 9, azimuth offset 18, number of azimuth lines 1751) to avoid border effects during the unwrapping,
- used 1 patch. The program indicates the memory requirements for the patch. If the indicated memory requirement had been larger than the available memory of the

computer we could have reduced the size of the patches, otherwise the optimization might have been very time consuming (if swap space is used)

- used as starting point for unwrapping the position 1200,1500, i.e. a point in the validity mask (same as when unwrapping with the branch cut algorithm in order to make phase values comparable)

While the unwrapped interferogram in file 05721_25394.flt_sm.mcf.unw0 has only non-zero values at locations which have non-zero values in the validity mask, the unwrapped interferogram in file 05721_25394.flt_sm.mcf.unw does not present gaps.

For the unwrapped interferogram obtained by masking for low coherence, we added a “0” to the extension to indicate that this is not the final unwrapped phase because of the gaps in areas of low coherence. The steps to obtain full spatial coverage are described below.

The unwrapped interferometric phase is written to a real valued data file which can be displayed, for example, using ***disrmg***. To generate the SUNraster version use the program ***rasrmg***. Here the examples for the *.unw image is presented

```
disrmg 05721_25394.flt_sm.mcf.unw 05721.mli 2500

rasrmg 05721_25394.flt_sm.mcf.unw 05721.mli 2500 - - - 4 4 - - - - - -
05721_25394.flt_sm.mcf.unw.ras
```

The unwrapped interferogram not masked for low coherence is shown in Figure A11.

To compare the two unwrapped interferogram obtained with the MCF algorithm we use the DISP program ***dis2rmg***

```
dis2rmg 05721_25394.flt_sm.mcf.unw 05721_25394.flt_sm.mcf.unw0 2500 2500
```

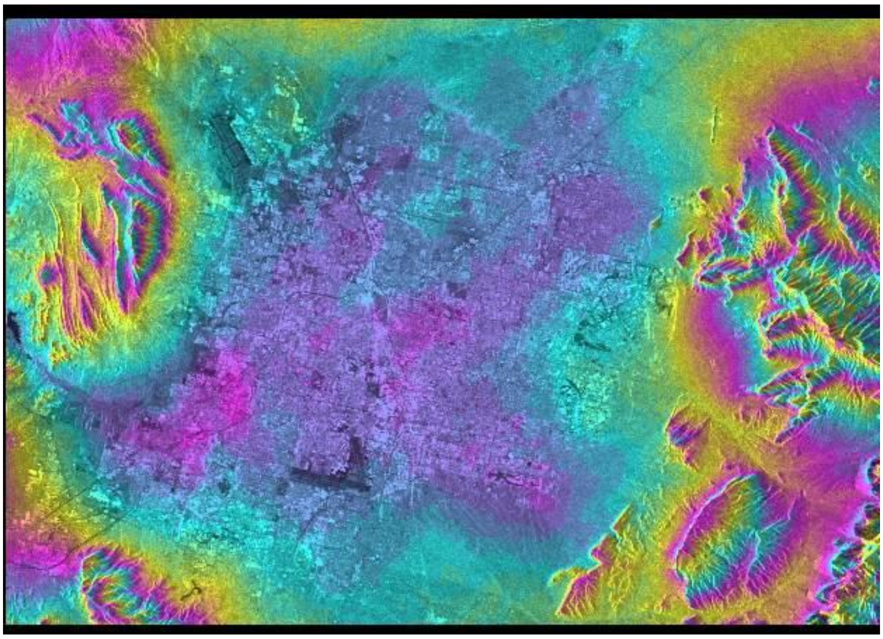
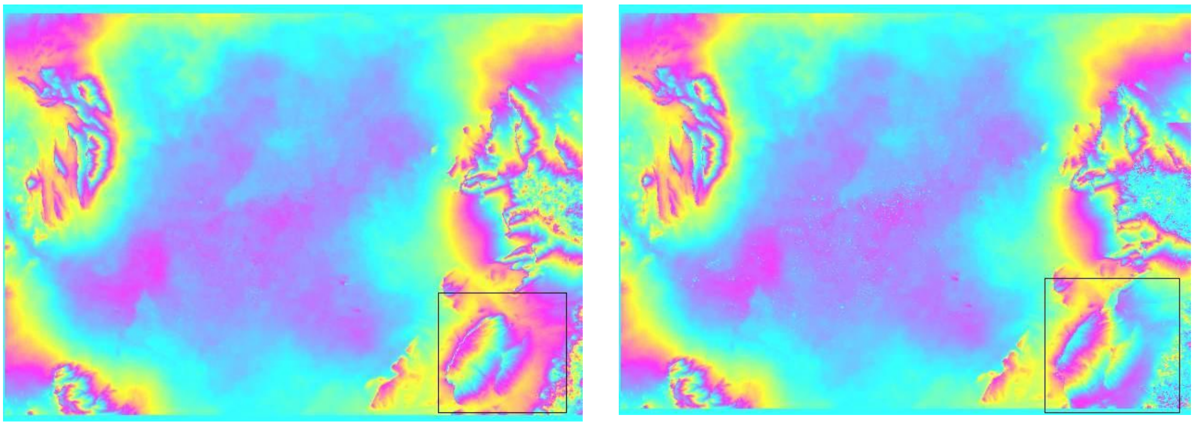


Figure A11. Unwrapped interferometric phase using the MCF algorithm with backscatter intensity as background. Phase display is 6π per color cycle.

Figure A12 shows the unwrapped interferograms. The patterns are similar almost entirely. However for the areas indicated by the boxes, while the unwrapped interferogram without phase validity mask (a) is characterized by a continuous trend in the phase, the other interferogram (b) presents two separate phase jumps of approximately 2π at the top and the bottom of the box. Although we expected that by masking out areas of low coherence, the unwrapping error would have decreased, we are actually experiencing the opposite. The reason for this result is that the area indicated by the box is almost entirely surrounded by layover and includes a relatively large percentage of pixels that have been masked out. The consequence is that the solution proposed by MCF for this area is in contrast with the unwrapping solution obtained for the rest of the image.

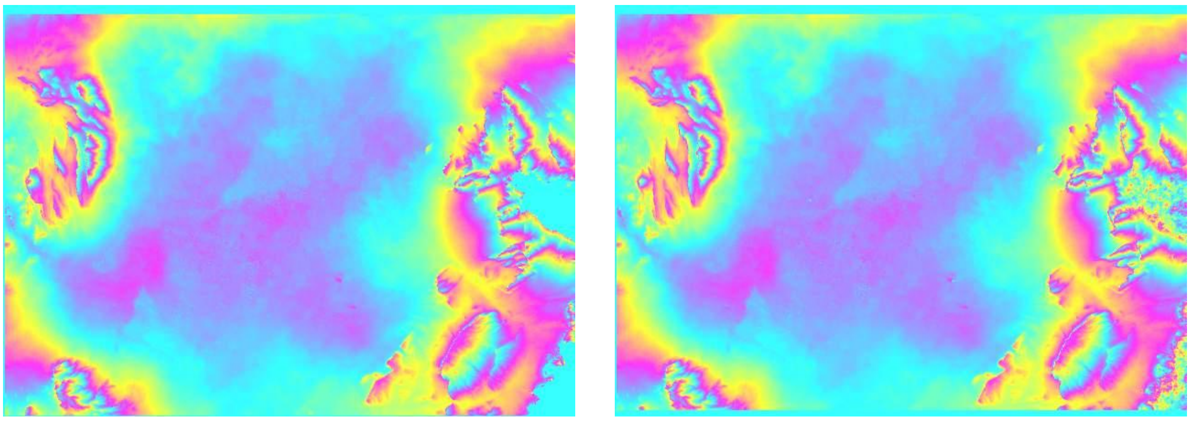
This result shows that when unwrapping with MCF one has to be extremely careful and closely look at the result. Changing one or more values of the ***mcf*** variables can improve the phase unwrapping.



(a) (b)

Figure A12. Unwrapped interferograms without (a) and with (b) phase validity mask.

Figure A13 finally illustrates the unwrapped interferograms using the branch-cut and the MCF algorithms. The main difference is that areas of low correlation have not been unwrapped in the “branch-cut” interferogram (compare right side of the interferograms).



(a) (b)

Figure A13. Unwrapped interferograms (a) branch-cut algorithm, (b) MCF algorithm.

A.10.2.4. Interpolation of gaps in unwrapped phase data

If the unwrapped phase image presents gaps, i.e. when phase validity mask has been applied, it can be filled by interpolation. An interpolator with a window size that adapts to the size of gap is preferred. This is implemented in the program ***interp_ad***.

Even though the unwrapped interferogram *.unw0 is not fully correct, we use it to show the result of interpolating and re-unwrapping.

For the unwrapped interferogram *.unw0, the command line in our example looks as follows:

```
interp_ad 05721_25394.flt_sm.mcf.unw0 05721_25394.flt_sm.mcf.unw0_interp 2500 32 8 16 2
```

The interpolated unwrapped interferometric phase is written to a real valued data file which can be displayed, for example, using ***disrmg***. To generate the SUNraster version of the image use the program ***rasrmg***.

```
disrmg 05721_25394.flt_sm.mcf.unw0_interp 05721.mli 2500

rasrmg 05721_25394.flt_sm.mcf.unw0_interp 05721.mli 2500 - - - 4 4 - - - - - 05721_25394.flt_sm.mcf.unw0_interp.ras
```

A.10.2.5. Phase unwrapping using model of unwrapped phase

Since we can assume that the interpolated phases are close to the correct unwrapped phases, at least within an interval $(-\pi,\pi)$, they can be used as phase model in the sense that the unwrapped phases are recalculated from the complex valued interferogram (the filtered or unfiltered one can be used) assuming that the phase values in the model correspond to the correct unwrapped phase within the interval $(-\pi,\pi)$. The command line of the program ***unw_model*** in this example looks as follows

```
unw_model 05721_25394.flt_sm 05721_25394.flt_sm.mcf.unw0_interp 05721_25394.flt_sm.mcf.unw 2500 1200 1500 -
```

It should be noticed that the same file name as for the unwrapped interferogram obtained without the phase validity mask has been used. Be sure to use different file names in case you are generating both unwrapped interferograms. For example the extension *.unw_mod could be used in such case.

The final unwrapped interferometric phase is written to a real valued data file which can be displayed, for example, using ***disrmg***. To generate the SUNraster version of the image use the program ***rasrmg***.

```
disrmg 05721_25394.flt_sm.mcf.unw 05721.mli 2500

rasrmg 05721_25394.flt_sm.mcf.unw 05721.mli 2500 - - - 4 4 - - - - - - 05721_25394.flt_sm.mcf.unw.ras
```

To compare the unwrapped interferograms with and without gaps we use the DISP program ***dis2rmg***

```
dis2rmg 05721_25394.flt_sm.mcf.unw0 05721_25394.flt_sm.mcf.unw 2500 2500
```

A.11. Least square estimation of interferometric baseline

To improve the estimate of the interferometric baseline we need to use ground control points (GCP).

The processing consists of 3 steps:

- Selection of ground control points
- Extraction of unwrapped phase values
- Least square estimation of the interferometric baseline based on the GCP height and unwrapped phase values.

For the estimation of the height we use the unwrapped phase obtained using the branch-cut algorithm. Since the file with GCP information is provided with the DEMO CD, the first Section is only demonstrative.

A.11.1. Selection of ground control points

To select GCPs an image file in SUNraster or bmp format is needed as guide in the selection process. A useful image is for example the display of the unwrapped interferometric phase together with the backscatter intensity. This can be obtained with the program ***rasrmg*** as e.g.

```
rasrmg 05721_25394.flt_sm.unw 05721.mli 2500 1 1 0 1 1 - - - - -
05721_25394.flt_sm.unw.ras
```

Notice that if the SUNraster version of the unwrapped interferogram has already been generated (as a quick-look version), it is advised to copy it to another file in order to avoid overwriting.

Using the image created above the ground control points can then be selected using ***gcp_ras***. In our example the command line looks as follows:

```
gcp_ras 05721_25394.flt_sm.unw.ras 05721_25394.gcp
```

The SUNraster file is displayed. Using the selection mouse button ground control points can be selected and the corresponding terrain heights (obtained form a map) written to the data input window. Ground control points in flat areas are preferred. Ground control points are best spread over the entire image. Selection of at least a dozen ground control points is recommended. Exiting the display window initiates the writing of the ground control points to standard output, respectively to the file indicated.

The data entered using ***gcp_ras*** are stored in a text format file that can be edited using a text editor if necessary. The file format consists of columns of numbers containing the GCP number, x (cross-track) pixel coordinate, y (along-track) line number, and the entered data. Each GCP has a single line entry in the file.

A.11.2. Extraction of ground control points unwrapped phase

Once the ground control points are determined the corresponding unwrapped interferometric phases can be extracted from the unwrapped interferogram and written together with the GCP coordinates to the GCP data file using the program ***gcp_phase***. In our example the command line looks as follows

```
gcp_phase 05721_25394.flt_sm.unw 05721_25394.off 05721_25394.gcp 05721_25394.gcp_ph
```

The GCPs provided in the file *.gcp are read and the corresponding unwrapped phase values are written to the *.gcp_ph file. This can be viewed or edited with a text editor.

A.11.3. Least square estimation of interferometric baseline

From the unwrapped phases and the height information a least squares estimate of the interferometric baseline is then obtained using the program ***base_ls***. The command line for this example looks as follows.

```
base_ls 05721.slc.par 05721_25394.off 05721_25394.gcp_ph 05721_25394.base 1 1 1 1 1 1
```

In this case we want to update all components and the rate of change of all components. The baseline file *.base will be updated.

A.12. Interferometric estimation of heights and ground ranges

Once the unwrapped phase and a good estimate of the baseline are available it is possible to estimate the heights and ground ranges. The estimation is done using the program ***hgt_map***. The command line in this case looks as follows:

```
hgt_map 05721_25394.flt_sm.unw 05721.slc.par 05721_25394.off 05721_25394.base 05721_25394.hgt 05721_25394.grd 1
```

The resulting height map in slant range / azimuth coordinates (*.hgt) may be displayed using the DISP program ***dishgt***. The SUNraster version of the height map can be obtained with ***rashgt***. The file *.grd represents the ground range map in slant range geometry for an ellipsoidal (WGS84) Earth model

```
dishgt 05721_25394.hgt 05721.mli 2500
```

```
rashgt 05721_25394.hgt 05721.mli 2500 - - - 4 4 - - - - 05721_25394.hgt.ras
```

It is possible to specify the color cycle in meters. In this case we used the default, i.e. 160 m / color cycle. Figure A14 shows the height map. To reduce the size of the SUNraster file we used an averaging factor of 4 and 4 in range and azimuth.

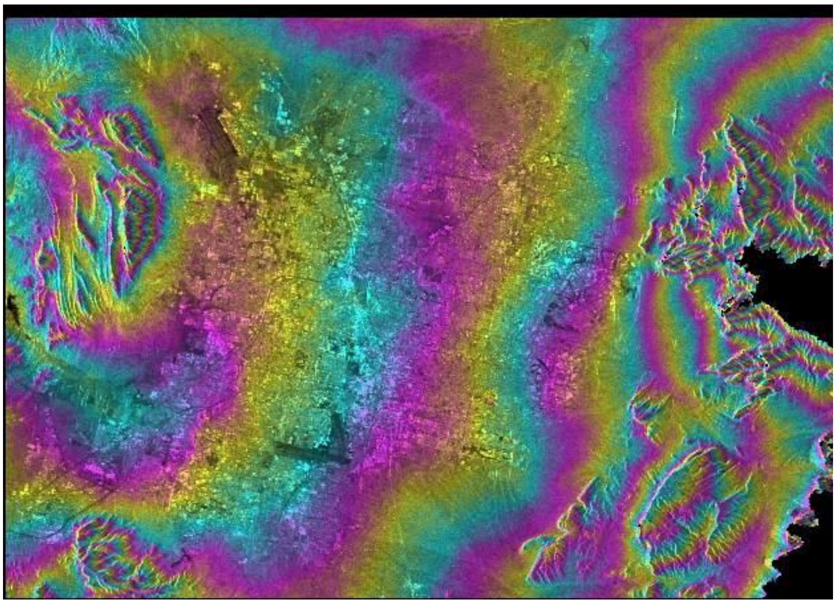


Figure A14. Interferometric height map with 160 m color cycle.

A.13. Resampling of interferometric height map to orthonormal coordinates

The heights and ground ranges given in SAR coordinates (slant range / azimuth) can be resampled to orthonormal coordinates (along track, cross track) using the geometric information of the interferometric heights and ground ranges. This can be done with the program ***res_map***. An additional data set, usually one of the two backscatter intensity images is also resampled to orthonormal coordinates:

```
res_map 05721_25394.hgt 05721_25394.grd 05721.mli 05721.slc.par 05721_25394.off 05721_25394.rhgt 05721.grd 7 7 20
```

The output consists of the resampled height map (*.rhgt) and the resampled MLI of the reference SLC (05721.grd). Images have been resampled to 20 m pixel size on the ground. To display the height map, one of the programs ***dishgt*** or ***disshd*** (for a shaded relief) can be used. It should be noticed that the size of the image in ground range coordinates has changed. In this case the width is 2630 pixels.

```
dishgt 05721_25394.rhgt 05721.grd 2630
```

```
disshd 05721_25394.rhgt 2630 20 20
```

A SUNraster file of the resampled height map may be generated using ***rashgt***:

```
rashgt 05721_25394.rhgt 05721.grd 2630 - - - 4 4 - - - - 05721_25394.rhgt.ras
```

A shaded relief in SUNraster format may be generated using ***rasshd***:

```
rasshd 05721_25394.rhgt 2630 20 20 - - 4 4 - - - 05721_25394.rhgt_shd.ras
```

Figures A15 and A16 illustrate the resampled height map with backscatter intensity as background and as shaded relief. Areas that have not been unwrapped clearly appear in the ground range geometry.

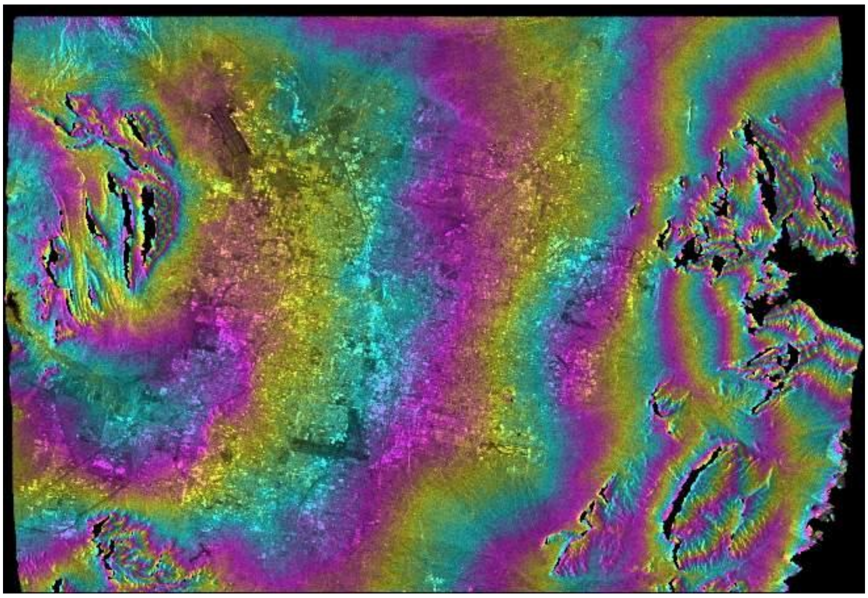


Figure A15. Interferometric height map in ground range geometry with 160 m color cycle.

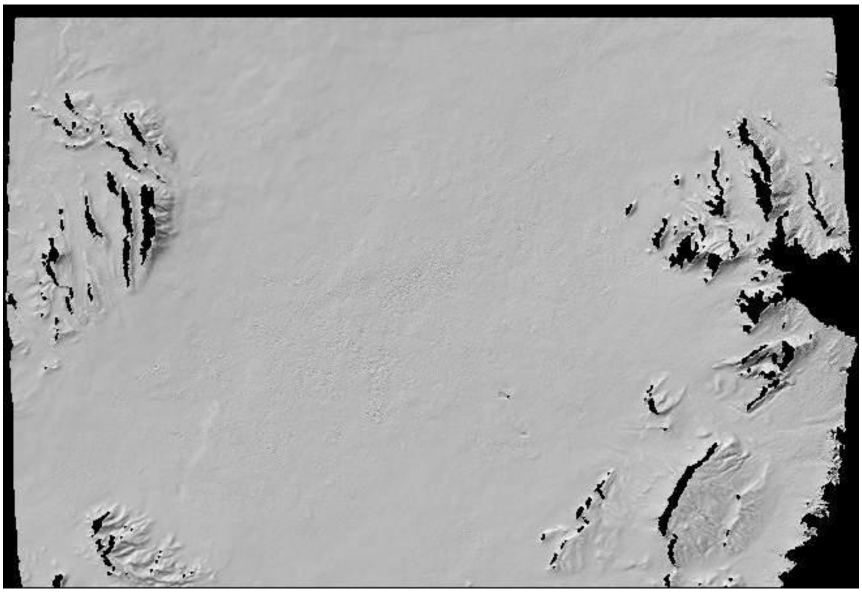


Figure A16. Interferometric height map in ground range geometry represented as shaded relief.

B. Generation of a calibrated SAR intensity image

In this Example we illustrate how to generate a calibrated SAR intensity image from a Single Look Complex SLC image obtained from a processing facility and a precision image (PRI). If the SLC has been obtained with GAMMA's MSP module correction for antenna pattern and absolute calibration have already been applied (see MSP User's Guide).

Before processing the following steps need to be performed

- Processing setup
- Copy data to disk
- Create the ISP SLC processing parameter file
- Manipulation of orbits (optional)

These steps have been described in Example A, Sections A and B.

Radiometric calibration (relative and absolute) is fairly simple and standard. Calibration requires the antenna pattern file and the absolute calibration constant factor. Depending on the sensor and the data format (SLC or PRI, float or short integer) some parameters may vary.

While the antenna pattern file is provided as an auxiliary file in most cases, the absolute calibration information delivered with the data can be found in the ISP SLC parameter file under the keyword "calibration gain". If the value 0.0 is indicated, it means that no calibration constant has been provided.

Calibrated images can be expressed as sigma nought or gamma nought. In the latter case compensation for varying incidence angle on flat terrain is applied. It shall be noticed that calibration does not take into account topographic related aspects. To compensate for local topography the output files from the program ***gc_map*** (in the DIFF&GEO module) must be used.

Below we present calibration procedures for a number of sensors and data types. As practical example we consider the ERS-1 SLC image used in Example A for interferometric processing. The image was acquired over Las Vegas on 23 May 1996 along orbit 25394. This is the scene identifier. The dataset is available on the DEMO CD-ROM.

B.1. Calibration of ERS SLC products

Both parameters depend on the sensor and on the processor used to generate the SLC. SLC provided by ESA can be in the VMP or PGS format. The PGS format replaced the VMP format during 2005 (for details see http://earth.esa.int/pub/ESA_DOC/SAR/VMP_PGS_products.pdf).

The file containing the antenna gain correction for the ERS satellites is provided with the software (file name: ERS1_antenna.gain). This should be copied to the working directory together with the SLC image 25394.slc and the ISP SLC parameter file 25394.slc.par.

To calibrate the SLC (sigma nought) use the program ***radcal_SLC***

```
radcal_SLC    25494.slc    25394.slc.par    25394.cslc    25394.cslc.par    4
ERS1_antenna.gain 1 1 1 59.61
```

Since the image in SCOMPLEX format a scaling factor of 60 dB is used (corrected for a small offset of 0.39). For details see the Reference Guide. Correction for range spreading loss and antenna pattern gain is also applied. Scaling for the absolute calibration constant has been applied automatically using the value provided in the ISP SLC parameter file (-48.13 dB).

If the SLC has been obtained with the PGS processor, the user must also specify the absolute calibration constant at the command line after the scaling factor. Contrarily to the VMP-formatted SLC, the absolute calibration constant is not provided for PGS products. For ERS-1 use -48.13, for ERS-2 use -49.7 as calibration constant.

To obtain the calibrated SAR intensity image, multi-looking with the program ***multi_look*** has to be performed. Because of the previous upscaling by 60 dB, a downscaling factor of 10-6 must be applied. The command line in our example looks as follows:

```
multi_look 25494.cslc 25494.cslc.par 25494.cmli 25494.cmli.par 1 5 - -
0.000001
```

From the calibrated SLC (*.cslc) the calibrate MLI image (*.cmli) is obtained. The calibrated SLC can at this stage be deleted.

Speckle filtering (see the LAT User’s Guide) and geocoding (see the DIFF&GEO User’s Guide on Geocoding and Image Registration) can then be applied.

B.2. Calibration of ENVISAT ASAR SLC products

In this Section we describe the radiometric calibration of ENVISAT ASAR SLC data for the Image Mode and the Alternating Polarization mode

At first the external calibration file provided by ESA has to be read in order to create the antenna pattern file with the program ***ASAR_XCA***.

To calibrate the SLC follow the same procedure as for the ERS VMP product. A scaling factor of 60 dB shall be used. The calibration constant is always provided. We recommend that you let ***radcal_SLC*** read the calibration constant automatically from the ISP SLC parameter file. In other words do not provide any value for this parameter at the command line. To obtain the calibrated SAR intensity image, multi-looking with the program ***multi_look*** has to be performed. Because of the previous upscaling by 60 dB, a downscaling factor of 10-6 must be applied.

B.3. Calibration of ERS and ENVISAT ASAR PRI products

PRI products are in short format and already relatively calibrated. To obtain an absolutely calibrated product and convert the image to float format, use the program ***radcal_PRI***. The user needs to specify the input and output data file names and corresponding parameter files. The absolute calibration constant is automatically read from the ISP SLC parameter file; hence a dash can be used at the command line. If a different calibration constant shall be used, this has to be specified at the command line.

Use then the program ***multi_look_MLI*** to obtain multi-looked version of the PRI image.

B.4. Calibration of PALSAR data

PALSAR detected data, no matter if SLC (i.e. Level 1.1) or orthorectified (i.e. Level 1.5), are already relatively calibrated, i.e. the elevation antenna pattern and range spreading loss have been corrected during the data processing.

The absolute calibration constant for SLC data is not provided along with the data (in the ISP SLC parameter file you will find the value 0.0 for the keyword calibration gain). The user must specify this value at the command line when running the program ***radcal_SLC***. The absolute calibration coefficient is -115 dB.

For orthorectified products (Level 1.5) the data is directly calibrated by the program ***par_EORC_PALSAR_geo*** when generating the image file and the ISP SLC parameter file. The absolute calibration constant is -83 dB. In other words reformatted Level 1.5 data are fully calibrated.

To provide an example on how to calibrate Level 1.1 PALSAR data we refer to the FBS sample data available at http://www.alos-restec.jp/sampledata_e.html. To obtain the images in GAMMA format use the program ***par_EORC_PALSAR***. As scene identifier we use the acquisition date 20061005.

To obtain the calibrated SAR intensity image (sigma nought) from the SLC image use the following command lines:

```
radcal_SLC 20061005.slc 20061005.slc.par 20061005.cslc 20061005.cslc.par 1
- 0 0 1 0 -115.0

multi_look 20061005.cslc 20061005.cslc.par 20061005.cmli 20061005.cmli.par
2 5
```

Radiometric calibration does not require correction for range spreading loss and antenna pattern since already applied. The data is FCOMPLEX format; hence no upscaling as for ERS and ENVISAT SLC data is required. Correction for absolute calibration of -115 dB is applied. Notice that if you working with polarimetric data, the procedure has to be repeated for each image.

C. Offset tracking

With offset tracking the registration offsets of two SAR images in both slant-range (i.e. in the line-of-sight of the satellite) and azimuth (i.e. along the orbit of the satellite) directions are generated. Applications requiring offset information can be classed into those that require only a global offset function, such as interferometry, multitemporal studies, and change detection, and those that require localized individual measurements, such as radar stereo and deformation mapping. Here we describe the use of offset-tracking for the estimation of surface displacement.

The offset fields are generated with a normalized cross-correlation of image patches of detected real-valued SAR intensity images. The location of the peak of the two-dimensional cross-correlation function yields the image offset. The successful estimation of the local image offsets depends on the presence of nearly identical features in the two SAR images at the scale of the employed patches. If coherence is retained the speckle pattern of the two images is correlated and tracking with small image patches can be performed to remarkable accuracy. In order to increase the estimation accuracy, oversampling rates are applied to the image patches and a two-dimensional regression fit to model the correlation function around the peak is determined with interpolation. The confidence level of each offset is estimated by comparing the height of the correlation peak relative to the average level of the correlation function to determine an effective correlation Signal-to-Noise Ratio (SNR). Coarse information on the slant-range and azimuth offsets is used to guide the search of the cross-correlation maximum. The estimated offsets are unambiguous values, which means that there is no need for phase unwrapping.

The image offsets in the slant-range and azimuth directions are related, in the absence of stereo offsets due to substantial topography and large orbit separation, to the different satellite orbit configurations of the two SAR images, the displacement occurring between the acquisition time interval of the image pair, and potential ionospheric effects. The estimation of surface motion requires the separation of these effects. The orbital offsets are determined \by fitting a bilinear polynomial function to offset fields computed globally from the SAR images assuming no displacement for most parts of the image. Detected ionospheric streaks on the azimuth offset maps may be high-pass filtered along the range direction.

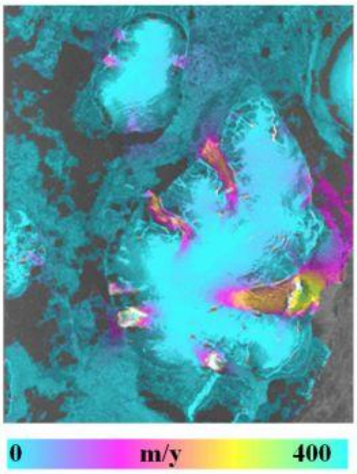


Figure C1. Horizontal displacement for Wilczek Land in Franz-Josef Land from JERS offset tracking between SAR images of January 6 and February 19, 1998 with a perpendicular baseline of 190 m. Image width is ~80 km.

Offset tracking consists of three processing steps

- Determination of the bilinear polynomial function
- Precise estimation of the offsets
- Computation of the displacement

The first step in offset tracking is the determination of the orbital offset between the two SLC images. The offset can be determined either by cross correlation of the real valued image intensity or by optimization of the fringe visibility globally from the SAR data assuming no displacement for large parts of the image. Polynomial coefficients for offsets in both range and azimuth direction over the whole image are determined.

Once the bilinear polynomial function is determined, many offsets are estimated in the area of interest. Recent developments were mainly dedicated to the image intensity cross-correlation algorithm, but the fringe visibility approach may be also applied. The bilinear polynomial function serves as indication for the positions where to precisely estimate the offsets. Typical size of the patch for offsets estimation is 64 x 256 SLC pixels, roughly corresponding to 1 km in ground range and azimuth with ERS or JERS SAR data.

The offsets estimated in range and azimuth direction are then transformed to displacements. The bilinear polynomial function determined all over the image is used for the separation of the orbital offsets from those of the area concerned with displacement. The displacement in the SAR geometry may be expressed in the range-azimuth or in the ground range-azimuth coordinate systems. Geocoding of the results is performed with the DIFF&GEO software module.

Below a processing example for measuring from the offsets the displacements of a glacier is given. For the processing two SLC images and the corresponding ISP SLC parameter files are needed. The ISP SLC parameter file can be obtained as described in Section 3. For a practical example see Example A, Sections 1 and 2. In this example we will use two images acquired by PALSAR on 20061206 and 20070211 separated by 46 days over a glacier on the island of Novaya Zemlya, Russia. The images are in SLC format and represent a subset of the original SLCs which have been obtained through JAXA's AO P1750001. For processing the image acquired on 20061206 will be used as reference. The dates of acquisition will be used as file identifiers. For self-processing the two SLCs and the corresponding ISP SLC parameter files have been provided on the DEMO CD-ROM.

C.1. Determination of the bilinear polynomial function

To compute the offsets between SLC, the first step consists of creating the ISP processing/offset parameter file using the program ***create_offset***. This file contains information used in the interferometric processing, such as file dimensions, geometric parameters, and the registration offset polynomials:

```
create_offset 20061206.slc.par 20070121.slc.par 20061206_20070121.off 1
```

The last input variable indicates the registration algorithm that will be used (1 for registration based on image intensity cross-correlation, 2 for fringe visibility method). The user can select number and size of the image chips that will be used when estimating the local offsets with the programs offset_pwr / offset_SLC (see Table on next page for some reference values). For this example with accept the default values suggested by the program.

Initial range and azimuth offsets between the two SLC images are estimated either manually by the operator (To do this the operator is supported with the SLC image display program ***disSLC***) or automatically using the programs ***init_offset_orbit*** and ***init_offset***. With ***init_offset_orbit*** a first guess of the offsets can be obtained based on orbital information. This first guess can then be improved with ***init_offset*** which determines the initial offsets based on the cross-correlation function of the image intensities. In order to avoid ambiguity problems and still to achieve an accurate estimates ***init_offset*** may first be run with multi-looking, followed by a second run at single look resolution. Each time the initial offset estimates are written to *.off file and are then used in following run as initial guess. The user can specify position and size of the area used to compute the cross-correlation between images, from which the initial estimate of the offsets will be derived. The user can also set the threshold on SNR intensity correlation for accepting/rejecting the offset estimates. As default the subset is put in the middle of the image with a dimension of 1024x1024 pixels. If the SNR intensity correlation is below the threshold, the estimated offsets are not written to the ISP offset parameter file. It is possible that by changing position and/or size of the subset, an estimate corresponding to a SNR intensity correlation above the threshold is obtained. To identify the location of the subset, it is suggested to look at one of the SLCs and look for areas with contrast (e.g. topography, urban areas, fields).

In this example we consider the use of ***init_offset_orbit*** only since PALSAR orbital data should be precise enough to provide a satisfactory initial estimate of the offsets. The command line looks as follows

```
init_offset_orbit 20061206.slc.par 20070121.slc.par 20061206_20070121.off
```

To obtain more precise range and azimuth offsets two methods can be pursued.

The first method is based on the image intensity and is implemented in the program ***offset_pwr***. For a large number of image segments this method searches for the range and azimuth offsets resulting in the maximum level of intensity correlation. The method requires at least a minimum of image contrast. It does not depend on the level of coherence between the two SLC images and is computationally efficient. The more important parameters for the search are number and size of the offset estimation windows in range and azimuth, and the offset estimation threshold. In the example we report some general values.

ISP processing parameter:	**Example for appropriate values for use of offset_pwr**
offset_estimation_range_samples:	32
offset_estimation_azimuth_samples:	32
offset_estimation_window_width:	64
offset_estimation_window_height:	64
offset_estimation_threshold::	7.0

The second method is based on the complex valued data and is implemented in the program ***offset_SLC***. For a large number of image segments this method searches for the range and azimuth offsets resulting in the maximum level of coherence. The method requires at least for a number of locations in the image a sufficient level of coherence between the two SLC. The method also works for very low image contrast. Appropriate values for the more important parameters for this search are:

ISP processing parameter:	Example for appropriate values for use of offset_SLC
offset_estimation_range_samples:	16
offset_estimation_azimuth_samples:	16
offset_estimation_window_width:	16
offset_estimation_window_height:	16
offset_estimation_threshold::	3.0

Both approaches determine a field of registration offsets (file *.offs) and a corresponding quality measure field (*.snr). The *.offs file is in FCOMPLEX format, the real and the imaginary part expressing for each image chip the range and the azimuth offset. Offsets can also be saved as text file for further analysis (file *.offsets).

If very few or even no estimates with good quality are achieved the registration will fail. This may be a result of

- poor initial estimate of the offsets
- too little contrast in the image (*for offset_pwr*)
- too large baseline (*for offset_SLC*)
- too much change (water, forest, high vegetation, long acquisition time interval)

In this example we use ***offset_pwr*** as follows

```
offset_pwr  20061206.slc  20070121.slc  20061206.slc.par  20070121.slc.par
20061206_20070121.off offs snr 128 128 offsets 2 24 24
```

Here we decided to use image chips being 128 pixels wide and long and accepted all other default values. The list of offsets is written to the binary file offs as well as to the text file offsets whereas the list of SNR values is written to the binary file snr.

Based on the field of registration offsets and the quality measure of the offsets, the bilinear registration offset polynomial is then determined using a least squares error method. This is implemented in the program ***offset_fit***. In this example the command line looks as follows:

```
offset_fit offs snr 20061206_20070121.off coffs coffsets
```

The binary files offs and snr are read and the resulting registration offset polynomials for range and azimuth offset are written to the ISP offset parameter file. The user can judge the quality of the registration between the images by looking at the estimated standard deviation of the offsets in range and azimuth. The offsets actually used for generating the registration polynomial are saved to the binary file coffs and the corresponding file in text version coffsets.

The procedure of estimating the offsets polynomials can be repeated to refine the polynomial. The offsets computed during the first run are used to guide the estimation procedure. In addition the user can choose whether to use a different number of image chips, their size and the SNR threshold. In this example we simply repeat the offset_pwr / offset_fit sequence used before:

```
offset_pwr  20061206.slc  20070121.slc  20061206.slc.par  20070121.slc.par
20061206_20070121.off offs snr 128 128 offsets 2 24 24

offset_fit offs snr 20061206_20070121.off coffs coffsets
```

C.2. Precise estimation of the offsets

Once the bilinear polynomial function is known, many offsets are estimated in the area of interest, again based on the image intensity cross-correlation or on the fringe visibility. If precise estimates shall be determined with the image intensity cross-correlation, use the program ***offset_pwr_tracking***. The method based on fringe visibility is supported by the program ***offset_SLC_tracking***.

The selection of one of the two supported methods (image intensity cross-correlation or fringe visibility) depends on the level of coherence between the two SLC images and on the image contrast, on one side, and on the computing time, accuracy and displacement velocity, on the other hand. For identical search windows, ***offset_SLC_tracking*** takes significantly longer to execute than ***offset_pwr_tracking***, but has the advantage that the offset is actually based on the interferometric phase coherence. Another advantage is that the program will find offsets in areas with essentially no features at all. If the displacement is very large, than ***offset_SLC_tracking*** may failed because of lack of coherence and ***offest_pwr_tracking*** is the only available source of information.

In both cases the bilinear polynomial function obtained at the previous stage serves as indication for the position where to estimate the precise offsets. The field of registration offsets here obtained is stored in a new FCOMPLEX file *.offs, where the real part corresponds to the offsets in range and the imaginary part to the offsets in azimuth. The corresponding quality measure field is stored in a new file *.snr.

If using ***offset_pwr_tracking*** a search window size of 64x256 pixels is appropriate for most of the cases. Oversampling rates of 1, 2 and 4 are supported, 2 is generally a good trade-off between accuracy and efficiency. For tracking the displacement of a glacier, a SNR threshold of 4.0 is preferred in order to obtain as many estimates as possible. If the noise is too high, estimates with SNR values larger than 4.0 can be rejected in the computation of the displacement (see Section C.3). The area where the offsets are to be estimated and the number of estimates can be specified on the command line.

For our example the command line looks as follows:

```
offset_pwr_tracking 20061206.slc 20070121.slc 20061206.slc.par
20070121.slc.par 20061206_20070121.off offsN snrN 64 192 offsetsN 2 4.0 12
36 1 2496 1 5976
```

The two SLCs, the corresponding parameter files and the ISP offset parameter file *.off area read, from which offset estimates in binary format (offsN), in text format (offsetsN) and offset estimation SNR in binary format (snrN) are obtained. The search region size has been set to 64 pixels in range and 192 in azimuth. In this way an almost squared area is considered. The quality threshold for the offset estimate is set to 4.0. Steps in range pixels are 12 pixels in range and 36 pixels in azimuth to reflect the ratio 1:3 of PALSAR pixel size in range and azimuth. The computation is limited to 2496 columns which corresponds to the rounded value of the width of the image (2500 pixels) divided by the multi-look factor in range (12). Similarly the computation is limited to 5976 lines which corresponds to the rounded value of the length of the image divided by the multi-look in azimuth (36). In this way the image of the offsets will match with the MLI obtained from the SLC using 12 and 36 as multi-look factors. This is of advantage when geocoding the maps of offsets since they are in the same geometry of the MLI image of the reference SAR image (see also Section C.4). Because of the dense sampling used in this example the program takes several minutes for generating the grid of offsets.

If using ***offset_SLC_tracking*** the search window size must be entered as a power of 2 (8, 16 or 32 are recommended). Oversampling rates of 1 (no oversampling), 2, 4, 8, and 16 are supported. The search chip interferogram size in non-oversampled pixels is set by default to 16. For tracking the displacement of a glacier, a SNR threshold of 2.5 is preferred in order to obtain as many estimates as possible. If the noise is too high, estimates with SNR values larger than 2.5 can be rejected in the computation of the displacement (see Section C.3). The area where the offsets are to be estimated and the number of estimates can be specified on the command line. The selection of the search window size, the oversampling factor, the search chip interferogram size and the SNR level depends on the size of the area of interest, on the number of estimates, and on the expected accuracy. Large oversampling factors (e.g. 4) with a relatively large search window size (e.g. 16) make the computation very slow and are not suitable for the determination of many offsets. If a large number of offsets have to be estimated, then it is preferred to reduce the search window size and/or the oversampling factor at cost of a reduced accuracy.

C.3. Computation of the range and azimuth displacements

The offsets estimated in range and azimuth direction are then transformed to displacements.

The bilinear polynomial function determined all over the image and stored in the *.off file is used for the separation of the orbital offsets from those of the area concerned with displacement. The range and azimuth offsets generated by ***offset_pwr_tracking*** or ***offset_SLC_tracking*** are transformed in range and azimuth displacements by means of the program ***offset_tracking***. The command line in this example looks as follows:

```
offset_tracking offsN snrN 20061206.slc.par 20061206_20070121.off coffsN
coffsetsN 2 4. 1
```

The output displacements are written to the binary file coffsN in FCOMPLEX format and (optionally) to a text file, which in this case is called coffsetsN. The displacements can be computed in range and azimuth directions in pixels or meters or in ground-range and azimuth directions (horizontal geometry) in meters. Points can be rejected based on a SNR thresholding. Offset estimates with a SNR value below the indicated threshold are not considered. In this case it was chosen to express the displacement in meters in ground range and azimuth direction (mode parameter set to 2), the SNR threshold to accept offset value was set to 4.0 and it was chose to subtract the polynomial trend from offset data (poly_flag parameter set to 1). If generated, the displacements for each image chip can be found in the last two columns of the displacements file in text format, i.e. in this example the file coffsetsN, respectively for the range and the azimuth direction.

C.4. Display of results

The MLI image is obtained with the program ***multi_look*** as follows

```
multi_look 20061206.slc 20061206.slc.par 20061206.mli 20061206.mli.par 12
36
```

The MLI image is 208 pixels wide and 166 pixels long. The ISP MLI parameter file can be used for geocoding the MLI and all offset tracking products in the same geometry.

The DISP program ***raspwr*** can be used to generate a SUNraster/bmp version of the multi-look intensity image.

```
raspwr 20061206.mli 208 - - 1 1 - - - 20061206.mli.ras
```

The multi-look image is displayed in Figure C2.

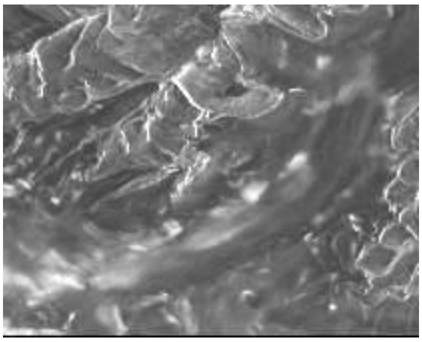


Figure C2. Multi-look image of 20061206.slc (208 pixels wide, 166 pixels long, pixel size approximately 120 m in both directions).The glacier of interest is located in the central part of the image along the valley stretching from the top right corner of the image to the bottom left corner.

The output file containing the displacements in FCOMPLEX format can be transformed in files of float with the program ***cpx_to_real*** (DISP module) in order to obtain the real or imaginary parts only (i.e. the range and azimuth displacements) or the maximal intensity or phase (i.e. direction) of the displacement. These files can be represented with the DISP programs ***dishgt*** or can be saved to a SUNraster / bmp image file with ***rashgt***.

In our example computation and display of the real and imaginary parts of the displacements file (i.e. range and azimuth displacements) are done as follows. At first the range and the azimuth component of the displacement are saved to separate files and then they are saved together with the SAR intensity image to two SUNraster files.

```
cpx_to_real coffsN coffsN_real 208 0

cpx_to_real coffsN coffsN_ima 208 1

rashgt coffsN_real 20061206.mli 208 1 1 0 1 1 30. 1. .35 1 coffsN_real.ras

rashgt coffsN_ima 20061206.mli 208 1 1 0 1 1 30. 1. .35 1 coffsN_ima.ras
```

Figure C3 shows these two components. The figures show the clear displacement of the glacier along both the ground range direction (i.e. horizontally) and the azimuth direction (i.e. vertically).

Computation and display of the maximal intensity of the displacement is achieved as follows. With ***cpx_to_real*** the magnitude of the complex file coffsN is obtained. With the DISP program ***rasdt_pwr24*** it is then possible to generate a 24-bit SUNraster/bmp file of float parameter (e.g. deformation) + intensity image:

```
cpx_to_real coffsN coffsN_mag 208 3

rasdt_pwr24  coffsN_mag  20061206.mli  208  1  1  0  1  1  30.  1.  .35  1
coffsN_mag.ras
```

Figure C4 shows the intensity of the displacement. The displacement of the glacier is clearly visible.

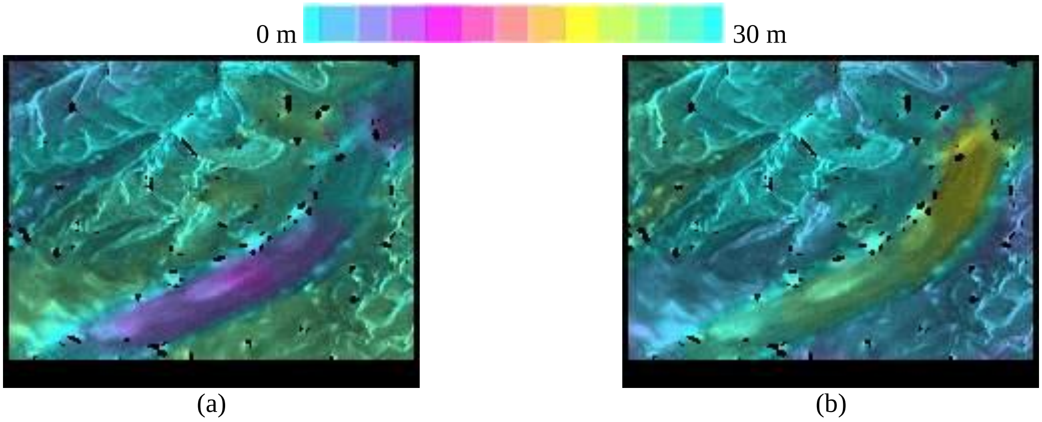


Figure C3. Real (a) and imaginary (b) component of the FCOMPLEX displacement file with SAR intensity of reference image in the background. The two images indicate respectively displacement in the ground range and the azimuth direction.

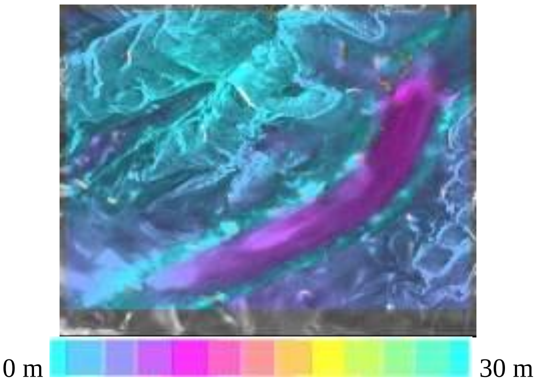


Figure C4. Intensity of displacement obtained from offset tracking file with SAR intensity of reference image in the background.

Further refinement is possible with filtering of noise and azimuth streaks (see Pritchard et al., 2005 and Wegmüller et al., 2006).

The MLI parameter file obtained with multi_look can now be used for setting up the geocoding lookup table so that data can be put into a specific map projection. Instructions for geocoding of SAR images can be found in the DIFF&GEO User's Guide on Geocoding and Image Registration.

C.5. Relevant publications on offset tracking processing

Pritchard, H., Murray, T., Luckman, A., Strozzi, T. and Barr, S., Glacier surge dynamics of Sortebræ, east Greenland, from synthetic aperture radar feature tracking, Journal of Geophysical Research, 110(F03005), doi:10.1029/2004JF000233, 2005.

Strozzi, T., Luckman, A., Murray, T., Wegmüller, U. and Werner, C., Glacier motion estimation using SAR offset-tracking procedures, IEEE Transactions on Geoscience and Remote Sensing, 40, 11, pp. 2384-2391, 2002.

Strozzi, T., Kouraev, A., Wiesmann, A., Sharov, A., Wegmüller, U. and Werner, C., Estimation of Arctic glacier motion with satellite L-band SAR data, Proceedings of IGARSS 2006, Denver, 31 July – 4 August, 2006.

Wegmüller U., Werner, C., Strozzi T., and Wiesmann, A., Automated and precise image registration procedures, in “Analysis of multi-temporal remote sensing images”, Bruzzone and Smits (ed.), Series in Remote Sensing, Vol. 2, World Scientific (ISBN 981-02-4955-1), pp. 37-49, 2002.

Wegmüller, U., Werner, C., Strozzi, T., and Wiesmann, A., Ionospheric electron concentration effects on SAR and INSAR, Proceedings of IGARSS 2006, Denver, 31 July – 4 August, 2006.

Werner C., Strozzi, T., Wiesmann, A., Wegmüller, U., Murray, T., Pritchard H. and Luckman, A., Complimentary measurement of geophysical deformation using repeat-pass SAR, Proceedings of IGARSS 2001, Sydney, 9-13 July, 2001.

Werner C., Wegmüller, U., Strozzi T., and Wiesmann, A., Precision estimation of local offsets between SAR SLCs and detected SAR images, Proceedings of IGARSS 2005, Seoul, 25-29 July, 2005.