Release Notes GAMMA Software, 20220701

Urs Wegmüller, Charles Werner, Christophe Magnard, Andreas Wiesmann,
Gamma Remote Sensing AG
Worbstrasse 225, CH-3073 Gümligen
http://www.gamma-rs.ch
1-Jul-2022

Introduction

This information is provided to users of the GAMMA software. It is also available online at https://www.gamma-rs.ch/uploads/media/GAMMA_Software_upgrade_information.pdf.

This release of the Gamma software includes new programs that provide new capability, additional features to existing programs and bug fixes.

Gamma Software on Linux, macOS, and Windows

The Gamma software has been compiled and tested on Linux (different distributions), Apple macOS BigSur (11.6.7) and Monterey (12.2.1), and Windows 10 and 11. Computationally intensive programs such as used in co-registration and resampling and geocoding have been parallelized using the OpenMP API built into the GCC compiler. Processing speed on Linux, macOS, and Windows systems is comparable.

Linux Distribution:

The Gamma software is developed on Ubuntu 20.04 LTS 64-bit Linux and is tested extensively with this distribution. The Gamma software is also available for Ubuntu 22.04 LTS.

Announcement: Support for Ubuntu 20.04 LTS will be available until the end-of-2023 upgrade.

Versions of the Software will also be uploaded for RHEL7 based on CentOS7 and RHEL8 based on CentOS8. Note that Red Hat has ended support for CentOS8 at the end of 2021. Consequently, the Gamma software built for RHEL8 / CentOS8 uses the final CentOS8 release.

For installation instructions for the binary LINUX distributions see the HTML file INSTALL_linux.html (provided with the distribution E-mail or found in the main directory of the distribution).

Apple MacOS Distribution:

The software in this version has been compiled using macOS BigSur (11.6.7) and macOS Monterey (12.2.1). You will need to install libraries such as GDAL using MacPorts. The build uses the GCC 11 compiler on macOS BigSur and GCC 12 compiler on macOS Monterey. A macOS Monterey (12.2.1) M1/M2 version is also available.

Announcement: The present upgrade is the last upgrade for macOS BigSur. MacOS Monterey will no longer be supported after the mid-2023 upgrade.

For installation instructions for the binary macOS distributions see the HTML file INSTALL_macOS.html (provided with the distribution E-mail or found in the main directory of the distribution).

Windows Distribution:

The Windows 10 and 11 version of the Gamma software is compiled with 64-bit support and multi-threaded. The build uses the MINGW64 GCC 12 compiler.

For installation instructions for the binary Windows distributions see the HTML file INSTALL_win64.html (provided with the distribution E-mail or found in the main directory of the distribution). Notice that installing the latest GAMMA_LOCAL_w64 version is mandatory because a new GCC compiler and new libraries were used to build the software. Furthermore, the .bashrc file needs to be updated following the installation instructions.

On Windows 11, it is now also possible to install the Windows Subsystem for Linux (WSL) and run a Linux distribution of the Gamma software on that environment. Instructions for this setup are available in the HTML file INSTALL_win11_wsl.html located in the main directory of the distribution.

Documentation and Program List

The Gamma documentation browser is an HTML based system for viewing the web pages and pdf documents. The documentation browser includes for each module a Contents sidebar on the right side of the screen and a search functionality.

The program *gamma_doc* facilitates the access to the documentation related to a given module or program:

gamma_doc	Opens the main page of the Gamma documentation browser and shows the program list.
gamma_doc DIFF	Opens the DIFF&GEO documentation.
gamma_doc gc_map2	Opens the reference manual web page for *gc_map2*.

Further information related to the GAMMA Software is available online:

General information:
gamma-rs.ch/uploads/media/GAMMA_Software_information.pdf
Technical reports, conference and journal papers:
gamma-rs.ch/uploads/media/GAMMA_Software_references.pdf
Release notes / upgrade information:
gamma-rs.ch/uploads/media/GAMMA_Software_upgrade_information.pdf

In case the program list is incomplete, run the python script `program_list.py` after successful installation of the Gamma Software in the main folder of the Gamma Software distribution:

```
./program_list.py Gamma_documentation_base.html Gamma_documentation_contents_sidebar.html -a
```

Python and Matlab wrappers

The Gamma Software is integrated into Python and Matlab through wrappers.

The *py_gamma* Python module permits a smooth usage of the Gamma Software within Python scripts as well as within a Python Interactive Development Environment (IDE) such as Spyder or PyCharm or using Jupyter Notebooks.

In the same way, the Matlab (and Octave) wrapper, composed of *mat_gamma* and *par_file* classes, permits a smooth usage of the Gamma Software within an interactive use of Matlab as well as within Matlab scripts.

Hardware Recommendations

Using multi-core processors (6 or more cores) will bring substantial improvement in processing speed due to parallelization of the code base. There should be at least 8 GB RAM available for each processor core with 16 GB per core recommended. Disk storage requirements for using the Gamma Software effectively depend on the amount of input data and data products that will be produced. Based on our experience we recommend considering at least 16 TB space, especially when working with stacks of Sentinel-1 or very high-resolution data (TerraSAR-X, Cosmo-Skymed) data. The current trend towards larger data products requires substantially increased storage capacities.

GAMMA Software Training Courses

A SAR/INSAR (MSP/ISP/DIFF&GEO/LAT) training at GAMMA (near Bern, Switzerland) is planned for late 2022. A PSI (IPTA) training is planned for late 2022. See also our website under http://www.gamma-rs.ch/courses/training-courses.html.

Significant Changes in the Gamma Software Modules since the end of 2021 Release

StriX (X-band constellation of the Japanese Company Synspective)

StriX-α and StriX-β are the first two X-band SAR satellites launched by the Japanese company Synspective. The SAR has a simple configuration while targeting a similar performance as bigger SAR satellites. It uses a 5m long passive antenna. It is planned that the orbits can be precisely controlled, in order to also support InSAR.

The GAMMA program *par_STRIX* is used to read the StriX SLC data. An example image is shown in Figures 1. For access to StriX data please contact Synspective through their web site (*https://synspective.com*).

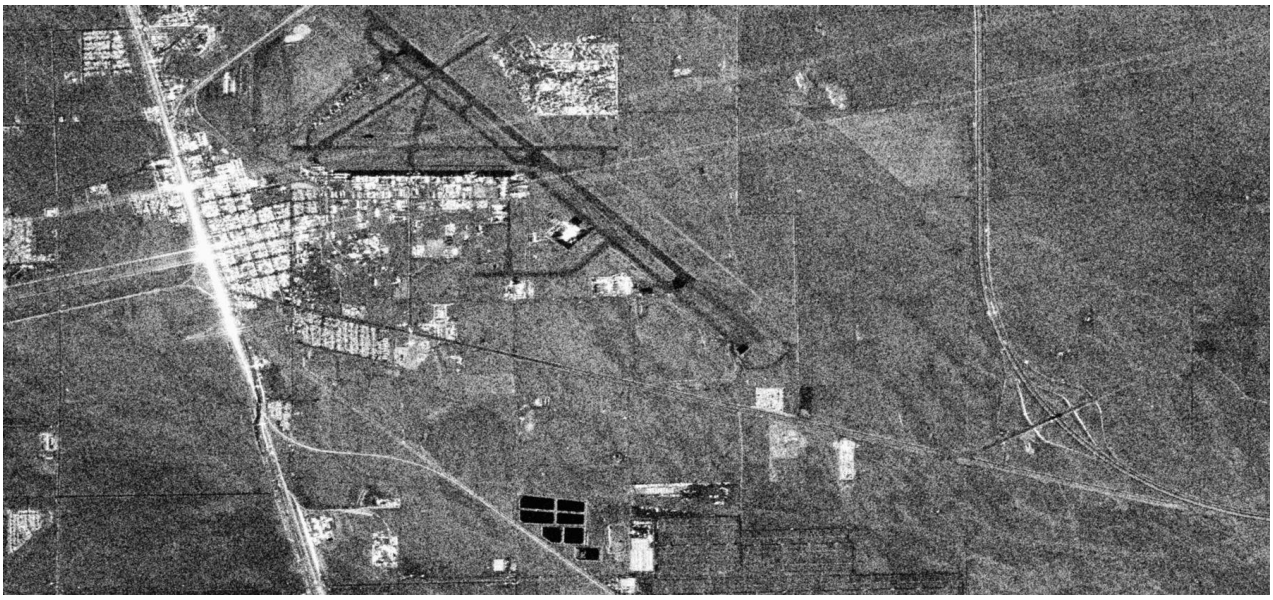
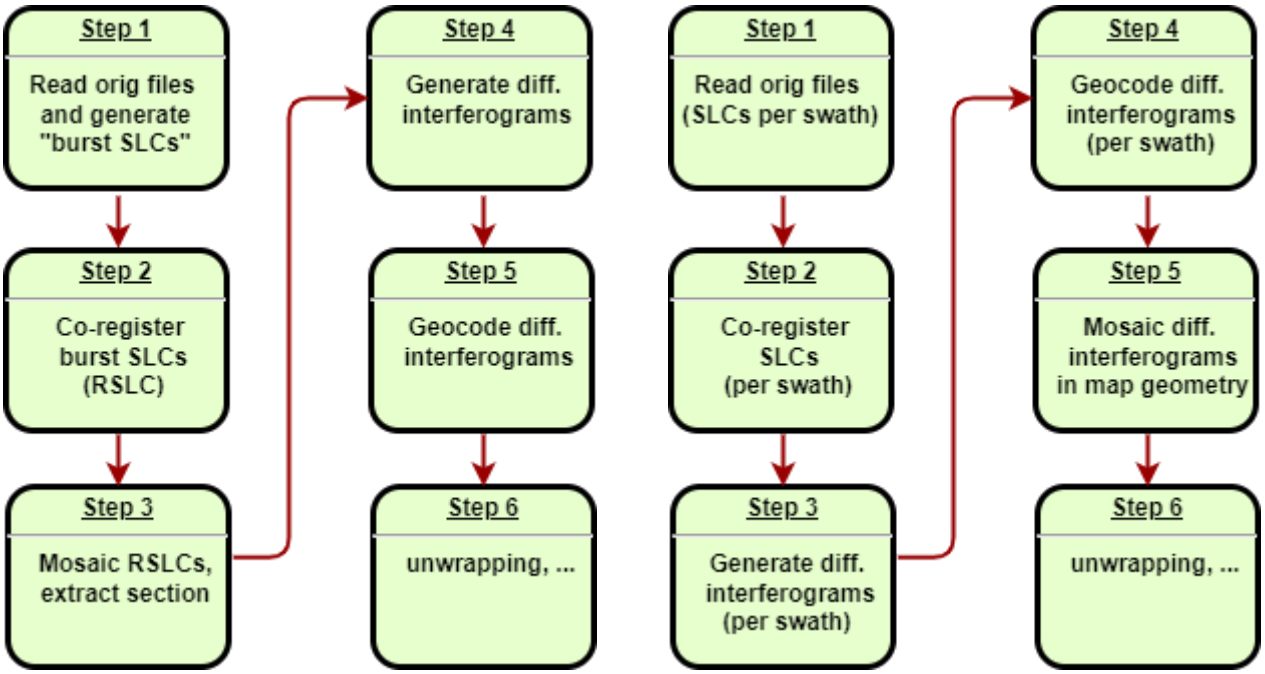


Fig. 1 Section of geocoded stripmap mode StriX-α VV-pol. X-band SAR backscatter on 16-Feb-2022 over Mojave, USA.

PALSAR-2 ScanSAR

In the mid-2022 upgrade, functionality to mosaic PALSAR-2 ScanSAR SLC data in the slant range geometry data was added. The new reader script *PALSAR_import_SLC_from_zipfile.py* imports PALSAR-2 full-aperture ScanSAR SLC data (in CEOS format) from the original data ZIP file and generates a "standard ScanSAR burst SLC data set" with an SLC_tab, SLC data, "par", and "TOPS_par" files. When several sub-swaths are imported, the script resamples the data to a common slant-range – azimuth grid. The format corresponds to a ScanSAR data set with several sub-swaths and a single burst in each sub-swath (this because of the full-aperture mode processing). ScanSAR co-registration and mosaicking programs can be used. As shown in Fig. 2a the data are co-registered in the slant range geometry, then SLC mosaics are calculated and used to calculate the differential interferograms (in slant range geometry). Alternatively, the already supported old procedure (Fig 2b) can be used. Here, the sub-swath SLCs are not resampled to a common grid. The SLC co-registration, interferogram calculation, and geocoding is done per sub-swath. The resulting sub-swath differential interferograms can then be mosaicked in the map geometry.



a) procedure with mosaic SLC b) procedure with mosaicking in map geometry

Figure 2 Flow charts for the two supported DInSAR sequences for PALSAR-2 ScanSAR data (starting from full-aperture SLC in CEOS format).

The new procedure results in a more straightforward processing sequence (corresponding closely to what is used for example with Sentinel-1 IWS data). Furthermore, band-pass filtering as required in the ionosphere mitigation can be done on the mosaic SLC in slant-range geometry, and therefore across multiple sub-swaths. The main disadvantage of the new procedure is the very large files generated. To mitigate this issue, an option permits to write the extracted PALSAR files in SCOMPLEX format instead of FCOMPLEX.

The old procedure with the per sub-swath processing up to the geocoded differential interferograms is still reasonable and also provides good results. The main advantage is the smaller maximum file size.

The New Zealand (Kaikoura) earthquake demo example was updated to demonstrate both the old and new procedures.

New and updated SAR data readers

New / updated SAR data reader	Short description
par_CSG_SLC	New program for generating parameter and image files for COSMO-Skymed Second Generation SCS data.
par_STRIX	New program to generate SLC parameter and image files for the STRIX Satellites of the Japanese company Synspective.
par_HISEA1_SLC	New program for generating SLC parameter and image files for the Chinese Hisea-1 SAR satellite.
PALSAR_import_SLC_from_zipfile.py	New script to import PALSAR full-aperture ScanSAR SLC data from a ZIP file. When several sub-swaths are imported, the script resamples the data as needed, to make sure that all sub-swaths share the same pixel spacing and are in a compatible grid. The output files include TOPS parameter files to enable use of the ScanSAR tools for mosaicking and co-registration.

Unwrapping correction with unw_correction_mb_pt.py

In a multi-reference stack with a certain redundance (e.g., selecting for each scene the pairs with the subsequent three scenes), *unw_correction_mb_pt.py* offers a process to check and correct the unwrapping of the unwrapped differential interferograms.

Running *mb_pt* on a multi-reference stack without smoothing (gamma = 0.0) results in a phase time series that follows the complete phases, including the atmospheric phase. In the case of consistently unwrapped differential interferogram phases the standard deviation of the original observations (= the input unwrapped differential interferograms) from the corresponding phases in the time series is very small (e.g., 0.1 radian). This is clearly different if one or several unwrapped differential interferogram have ambiguity errors (unwrapping errors). In this case the phase standard deviation gets significantly larger.

Aside from the phase time series, *mb_pt* also calculates simulated unwrapped differential interferogram phases for the pairs indicated. In the case of isolated unwrapping errors, this simulated phase tends to deviate the strongest from the phases with an ambiguity error. So, some unwrapping errors can be identified and corrected using that information.

IPTA regression programs to support data in 2D stacks

The regressions relative to the baseline and time difference are a core element of the interferometric times series analysis. In the IPTA module, these are supported by the programs *multi_def_pt* and *def_mod_pt*. Until now, the time series analysis in the IPTA module was always done using the data stacks in vector data format. In the mid 2022 upgrade, the programs *multi_def* and *def_mod* were added to support the same regression analysis for stacks of 2D differential interferograms or unwrapped differential interferograms.

Radio-frequency-interference filtering

The already available RFI filtering program *SLC_RFI_filt* for SLC data was complemented in the mid 2022 upgrade with the new program *SLC_RFI_filt2* that does an RFI filtering of SLC images using a band-stop filter. Furthermore, a demo example is now provided for the RFI filtering (*Gamma_demo_RFI_filtering.tar.gz*). Figure 3 shows an example of a Sentinel-1 SLC over Davos with RFI in the upper half of the section shown.

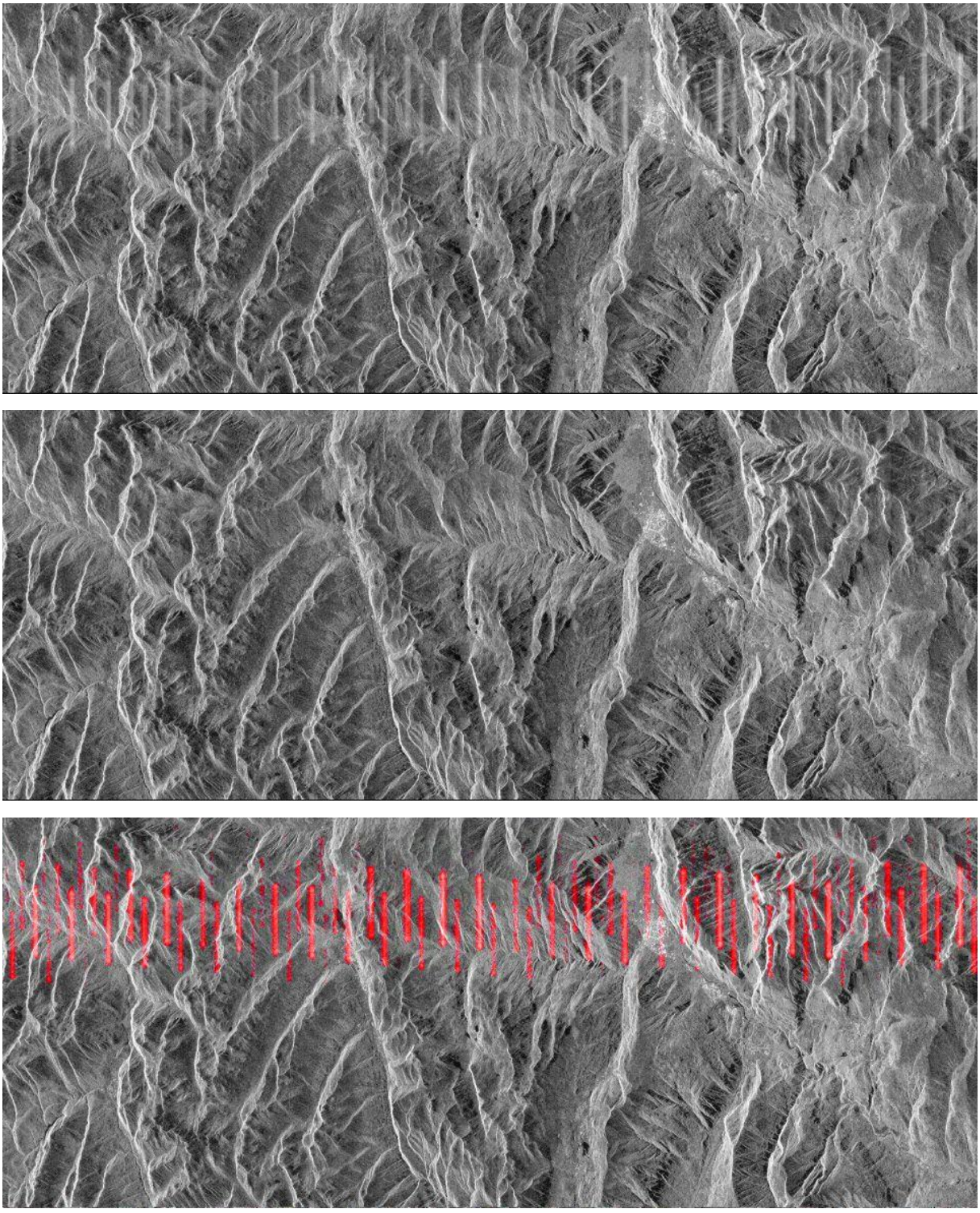


Figure 3 Sentinel-1 IWS VH pol. data over Davos with RFI effects (top), after RFI filtering with *SLC_RFI_filt2* (center) and color coded RFI visualization (high ratio unfiltered/filtered → red, indicates RFI, bottom).

Support for visualization of multiple co-registered images using the vis* display programs

The display programs *visbyte.py*, *viscpx.py*, *visdt_pwr.py*, *vismph_pwr.py*, *vispwr.py*, *visras.py* now support the display of multiple images (using alternate display), either using the new option "-a" to display a second image or using a pattern matching a set of raster images.

An example for the display of just 2 images is indicated here:

vismph_pwr.py 20191001_20191013.diff 20191001.rmli 2837 -a 20191001_20191025.diff -b &

To display a stack of images a pattern is indicated, such as:

vismph_pwr.py "20191001_????????.diff" 20191001.rmli 2837 -b -h &

or

vismph_pwr.py ".diff" 20191001.rmli 2837 -b -h &*

Notice that the pattern needs to be indicated in hyphens (" "). The display window looks then as shown in Figure 4. A simple interface was added to navigate through the multiple images. The option -h is used for a quicker display (disables preloading all images).

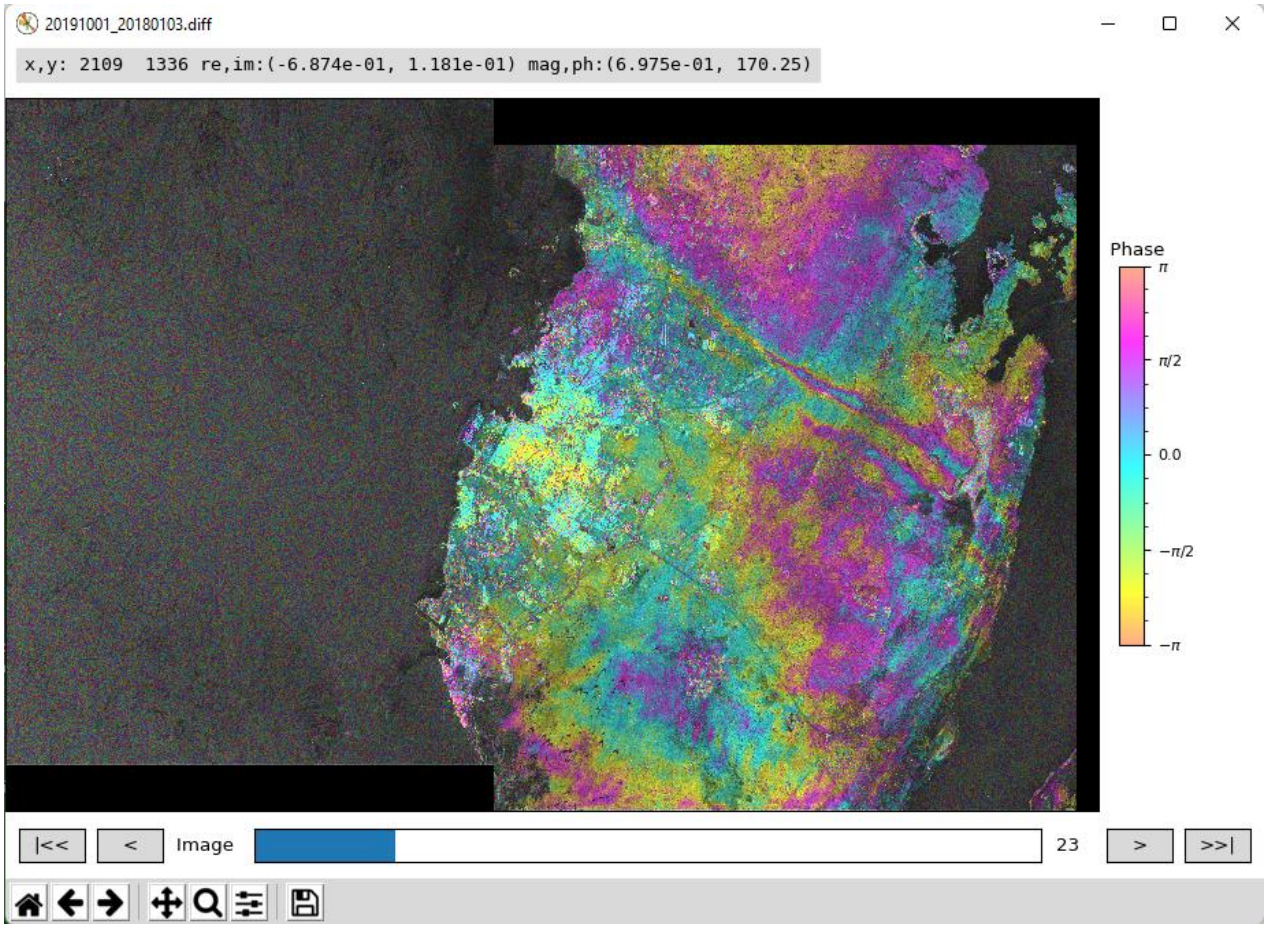


Figure 4 Visualization of multiple images using a pattern in a vis* command (see above). To move to the next or previous image the arrow can be used, to move several images a different location in the blue/white bare can be clicked.

Calculation of a weighted height in pixel_area

The program *pixel_area* was updated to also calculate, as an option, an area weighted terrain height written out in slant range geometry. For layover pixels, the different areas laid over contribute their heights with the contributing terrain area used as weight for the contribution. The terrain heights of the different areas contributing to a layover pixel may be very different, e.g., the valley ground with an elevation of 600m and sections of a mountain at 1400m and 2000m. Using this weighted height to calculate a differential interferogram may reduce some phase anomalies in differential interferograms but cannot be expected to perfectly resolve the phase interpretation in layover areas. Hence, layover masks should still be considered to avoid misinterpretation of the interferometric phase in layover areas.

Gamma Software Demo examples

In this period again some Gamma Software Demo examples were added/modified. Their access is limited to Gamma Software users with a valid license. The access information is provided with the software delivery.

New / modified demo example:	Contents
Gamma_demo_RFI_filtering.tar.gz	RFI filtering is demonstrated on 1 PALSAR and 2 Sentinel-1 SLCs. One example is the RFI shown in Figure 3.
Gamma_demo_PALSAR2_ScanSAR_NZ.tar.gz	Demo example on the generation of a large-area differential interferogram using PALSAR-2 ScanSAR SLC data (processed using the full-aperture processor, provided in CEOS format). Concerning the mosaicking between the swaths both procedures shown in Figure 2 are demonstrated. In addition, the co-registration is done with and without refinement with an offset field. Only the data of 2 sub-swaths are included in the zip files provided. Nevertheless, the total size of the demo files is 25 GB.

MSP

-

ISP

par_CSG_SLC: New program for generating parameter and image files for COSMO-Skymed Second Generation SCS data.

par_EORC_PALSAR: More precise near range distance. The program runs faster when only generating the SLC parameter file (no output reformatted PALSAR SLC). New option [dtype] to write data in FCOMPLEX (default) or SCOMPLEX format. New option [sc_dB] to scale data when writing them in SCOMPLEX format.

PALSAR_import_SLC_from_zipfile.py: New script to import PALSAR full-aperture ScanSAR SLC data from a ZIP file. It permits selecting which polarization and sub-swath(s) will be unzipped and imported. When several sub-swaths are imported, the script resamples the data as needed, to make sure that all sub-swaths share the same pixel spacing and are in a compatible grid. The output files include TOPS parameter files to enable use of the ScanSAR tools for mosaicking and co-registration. Include the option --scpx to write data in SCOMPLEX format to save disk space.

SLC_copy_PALSAR_ScanSAR.py: New script to copy an area from PALSAR full-aperture ScanSAR SLC data. The input data must have been generated using *PALSAR_import_SLC_from_zipfile.py*.

SLC_mosaic_ScanSAR: *SLC_mosaic_S1_TOPS* has been renamed to *SLC_mosaic_ScanSAR*. A new script permits calling the program using its previous name.

ave_image: Now opens only one file at a time and has been parallelized.

SLC_RFI_filt2: New program for RFI filtering of SLC images using a band-stop filter.

par_MSP, SLC_copy: adjusted *par_MSP* and *SLC_copy* for non-spaceborne cases in which the terrain height is larger than the height above ground.

par_STRIX: New program for generating parameter and image files for Synspective StriX SLC data.

par_HISEA1_SLC: New program for generating SLC parameter and image files for Hisea-1 SLC data.

par_TX_SLC: New option [dtype] permits writing FCOMPLEX data when input data are SCOMPLEX. For data written in FCOMPLEX format, the calibration factor (calibration gain) is now automatically applied to the SLC data.

DIFF&GEO

condition_offset_estimates: Now can also be used without access to the LAT package.

SLC_coreg.py: Replaces the script *SLC_coreg*, adds options to specify the polynomial order for the fit, the number of range and azimuth offset estimates, the size of the patches, the cross-correlation threshold, the interpolation method and order, and if intermediate files should be kept.

ionosphere_mitigation_S1: Adapted to also run for PALSAR-2 ScanSAR data in full-aperture mode, read using *par_EORC_PALSAR* v3.2 of 3-Feb-2022 or later.

dem_import, create_dem_par, DIFF_io.c: Support for DEMs and data spanning both sides of the date line has been improved when using EQA coordinates (lat/lon).

stacking: Updated program to read input data in blocks from one file at a time so that there is essentially no limit on the number of input data files that are used. Also added use of OPENMP to speed up computation by a factor of 2 or more.

atm_mod_2d: Update *atm_mod_2d* to permit setting the threshold svd_tol to accept singular values. The threshold is defined as svd_tol * max(w[]) where w[] is the set of singular values. The singular values for the model w0, and w1 can now be saved for analysis.

pixel_area: New option [hgt] to convert the DEM in RDC geometry using the same method as for the calculation of the gamma0 normalization area.

resamp_image, resamp_image_par: Updated to also support SCOMPLEX data.

DISP

visdt_pwr.py vismph_pwr.py: Raster output images in PNG format are now transparent if either the data are 0.0, or the background intensity image is 0.0.

visbyte.py, viscpx.py, visdt_pwr.py, vismph_pwr.py, vispwr.py, visras.py: Multiple images can now be displayed (using alternate display), either using the new option -a to display a second image or using a pattern matching a set of raster images. A simple interface was added to navigate through the multiple images. New option -h disables preloading all images when displaying multiple images, to spare memory.

swap_bytes: Modified to work with HUGE files, now using size_t for number of bytes in a file and the number of buffers.

html files: For some display programs individual reference manual html files were added.

LAT

-

IPTA

pwr_stat: Now also supports data in map coordinates.

IPTA_lib.c, ipta.h, ISP_io.c: Stacks of SLC parameter files are now text-based files. Previously generated binary files are still supported for reading and will be replaced by a text-based version when writing into them (using *SLC_par_pt*). Users do not need to change anything to their IPTA commands.

kml_ts_pt: Can now also use the displacement values of one record for the point colors.

mb, mb_pt: New [mode] option: when set to 0, valid unwrapped phase values are required in all layers for a point to calculate time-series; when set to 1 (default), the time-series is calculated even when unwrapped phase values are missing in some layers, requires network connectivity; when set to 2, the time-series is calculated even when unwrapped phase values are missing in some layers, does not require network connectivity.

vu_disp2d: New [zflag] option: permits interpreting 0.0 values either as "no data" values (default) or as valid values.

pt_density: New [n_max] option to specify a maximum number of points within the search radius. Program updated and parallelized using OpenMP, major processing speed improvement.

unw_correction_mb_pt.py: New script to correct phase unwrapping for a multi-reference stack using model produced by *mb_pt*.

atm_mod_2d_pt: Update *atm_mod_2d_pt* to permit setting the threshold svd_tol to accept singular values. The threshold is defined as svd_tol * max(w[]) where w[] is the set of singular values. The singular values for the model w0, and w1 can now be saved for analysis.

def_mod: New program to estimate deformation rate, delta height and residual phase over a scene for a 2D-image-based interferometric phase. Corresponds to the point-based version *def_mod_pt*.

multi_def: New program to estimate deformation rate, delta height and residual phase over a scene for a 2D-image-based interferometric phase using a multi-patch strategy. Corresponds to the point-based version *multi_def_pt*.

Python wrapper

py_gamma.py: The Python wrapper now also support Python scripts for GS-L and GPRI2. To enable this support, the environment variables "GSL_HOME" and "GPRI2_HOME" must have been defined.

py_gamma.py: When both "python" and "python3" calls exist, and "python" points to Python 2, the wrappers will execute Python scripts using the Python version specified in the shebang.

Matlab wrapper

mat_gamma_base.m: The Matlab wrapper now also support Python scripts for GS-L and GPRI2. To enable this support, the environment variables "GSL_HOME" and "GPRI2_HOME" must have been defined.

All packages

INSTALL_win11_wsl.html: Installation instructions for running the Gamma Software under WSL in Windows 11.

Gamma_documentation_contents_sidebar.html, DIFF_documentation_contents_sidebar.html, DISP_documentation_contents_sidebar.html, GEO_documentation_contents_sidebar.html, IPTA_documentation_contents_sidebar.html, ISP_documentation_contents_sidebar.html, LAT_documentation_contents_sidebar.html, MOCOM_documentation_contents_sidebar.html, MSP_documentation_contents_sidebar.html: Improved navigation experience in the documentation interface: forward and backward browser functionality now works throughout the documentation interface.