Release Notes GAMMA Software, 20241201

Urs Wegmüller, Christophe Magnard, Charles Werner, Othmar Frey, Andreas Wiesmann
Gamma Remote Sensing AG
Worbstrasse 225, CH-3073 Gümligen
http://www.gamma-rs.ch
1-Dec-2024

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 Sonoma (14.7) for both Intel and Apple Silicon processors, 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 22.04 LTS 64-bit Linux and is tested extensively with this distribution. The Gamma software is also available for Ubuntu 24.04 LTS.

Versions of the Software will also be uploaded for RHEL8 based on Rocky Linux 8, and RHEL9 based on Rocky Linux 9.

Note that Red Hat has ended support for CentOS8 at the end of 2021. Consequently, the Gamma software built for RHEL8 is now built on Rocky Linux 8.

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 Sonoma (14.7). You will need to install libraries such as GDAL using MacPorts. The build uses GCC 14 compiler. Version are available for Apple Silicon (M1/M2) and Intel x86_64 processors.

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 version of the Gamma software is compiled with 64-bit support and multi-threaded. The build uses the MINGW64 GCC 14 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).

The Gamma Plugin for ArcGIS is available in all Windows distributions that include the *GEO* or *ISP/DIFF&GEO* modules. Full functionality requires access to the *LAT* module.

On both Windows 10 and 11, it is also possible to install the Windows Subsystem for Linux (WSL2) and run a Linux distribution of the Gamma software on that environment. Instructions for this setup are available in the HTML file INSTALL_wsl.html located in the main directory of the distribution.

Announcement: With Windows 10 reaching end-of-support in October 2025, the Windows distribution will be tested exclusively on Windows 11 beginning with the mid-2025 release.

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 main Gamma documentation browser page *Gamma_documentation.html* is found in the main software directory.

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. Gamma Software program calls become Python / Matlab function calls where command line arguments can be used as function arguments, and system outputs can be stored in variables or written to log files. Binary images, point lists and data, parameter files, tab files, can be easily read, inspected, and written using additional functions provided with the 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. Using *py_gamma*, function arguments can be entered either as positional arguments or as keyword arguments, with the Gamma command line parameter names becoming the keyword names.

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.

Gamma plugin for ArcGIS

The Gamma plugin for ArcGIS permits using some Gamma software functionalities (tools) from ArcGIS Pro (Windows only) using a convenient interface. The Gamma plugin allows to perform the following operations:

- Reading SAR data from various sensors / formats
- Detection, radiometric calibration and geocoding of SAR data
- Co-registration of SLC and MLI SAR images in slant range / azimuth geometry
- Adaptive interferometric coherence estimation
- Multi-temporal processing and filters
- Spatial filtering of 2D SAR images
- Change detection in SAR images
- Polarimetric decompositions

Using ArcGIS ModelBuilder, it is possible to generate dedicated processing chains using the Gamma tools as building blocks. It is also possible to use each Gamma tool as a Python function (ArcGIS Pro is required). Note that the LAT module is required to be able to use all the tools; without the LAT module, only a subset of the tools is available.

See also gamma-rs.ch/uploads/media/2024-1_Gamma_Plugin_for_ArcGIS_presentation.pdf.

Hardware Recommendations

Using multi-core processors (8 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 5. – 9. May 2025.

A PSI (IPTA) training at GAMMA (near Bern, Switzerland) is planned for 1. – 4. Apr. 2025. See also https://www.gamma-rs.ch/software/training.

Significant Changes in the Gamma Software Modules since the Mid 2024 Release

Reader program for ALOS-4 PALSAR-3 SLC data: par_EORC_PALSAR

The reader program *par_EORC_PALSAR* was updated to also support reading ALOS-4 PALSAR-3 SLC data in CEOS format. Only initial testing with real data acquired during the commissioning phase was possible, so far.

Reader program for SWOT KaRIn SLC data: par_SWOT_SLC

A new reader program to support reading SLC data acquired by the KaRIn instrument of the SWOT mission. The **Surface Water and Ocean Topography (SWOT)** mission is a joint collaboration between the National Aeronautics and Space Administration NASA, USA, the French Space Agency CNES, the Canadian Space Agency, CSA, and the United Kingdom Space Agency, UKSA, and was launched on Dec. 16, 2022. One of the instruments on SWOT is a **Ka-band Radar Interferometer (KaRIn)**. As the primary instrument of the mission, KaRIn is a near-nadir viewing interferometric altimeter, consisting of two distinct 50 km swaths. It uses two SAR antennas, which are located at the opposite ends of a 10 m boom, enabling researchers to gather information along a roughly 120 km wide swath of the Earth's surface and to discern currents and eddies as small as 20 km across. KaRIn bounces radar pulses off the water's surface and receives the return signal with its antennas at the same time. While the nadir altimeter will point straight down and take data in one dimension, the KaRIn antennas are scanning the surface in two dimensions and collecting data with greater precision than the nadir altimeter alone. Further information on the mission concept is available from https://www.eoportal.org/satellite-missions/swot#spacecraft

An SLC product is available for the KaRIn instrument. This product is called the Level 1B Ka-band Radar Interferometer (KaRIn) high rate (HR) single look complex (SLC) data product from the Surface Water Ocean Topography (SWOT) mission. The KaRIn L1B_HR_SLC data product is defined in the related Product Description Document (JPL D-56410, Revision B, "SWOT Product Description Document: Level 1B KaRIn High Rate Single Look Complex (L1B_HR_SLC) Data Product," Jet Propulsion Laboratory Internal Document, 2023.), available at https://podaac.jpl.nasa.gov/dataset/SWOT_L1B_HR_SLC_2.0. SWOT KaRIn data can be accessed from https://search.earthdata.nasa.gov/search. Results over an area in Switzerland are shown in Figures 1 and 2. For more information see SWOT_Support_in_GAMMA (available at: https://www.gamma-rs.ch/uploads/media/2024-8_TR_SWOT_Support_in_GAMMA.pdf).

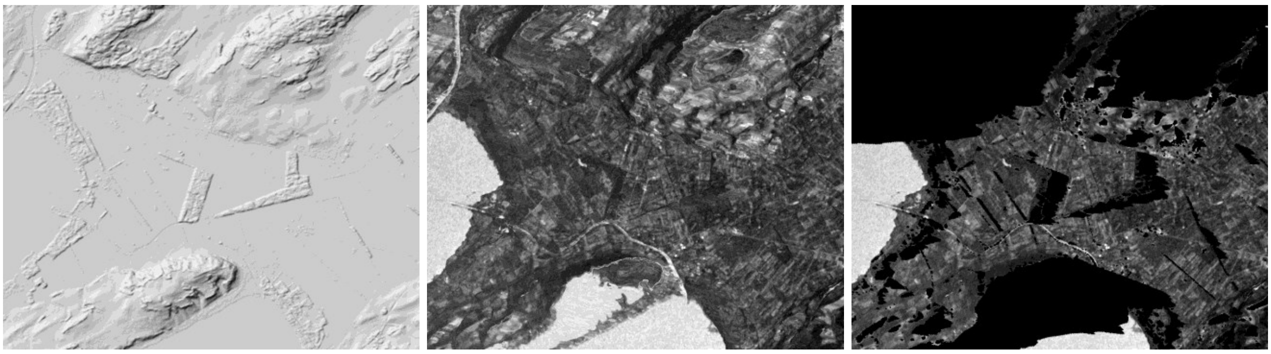

Figure 1 Section of geocoded SWOT backscatter over Ins, Switzerland. The image to the left shows the shaded relief of the area, the image in the center the geocoded SWOT backscatter without masking layover/shadow areas, and the image to the right shows the geocoded SWOT backscatter with masked layover/shadow areas.

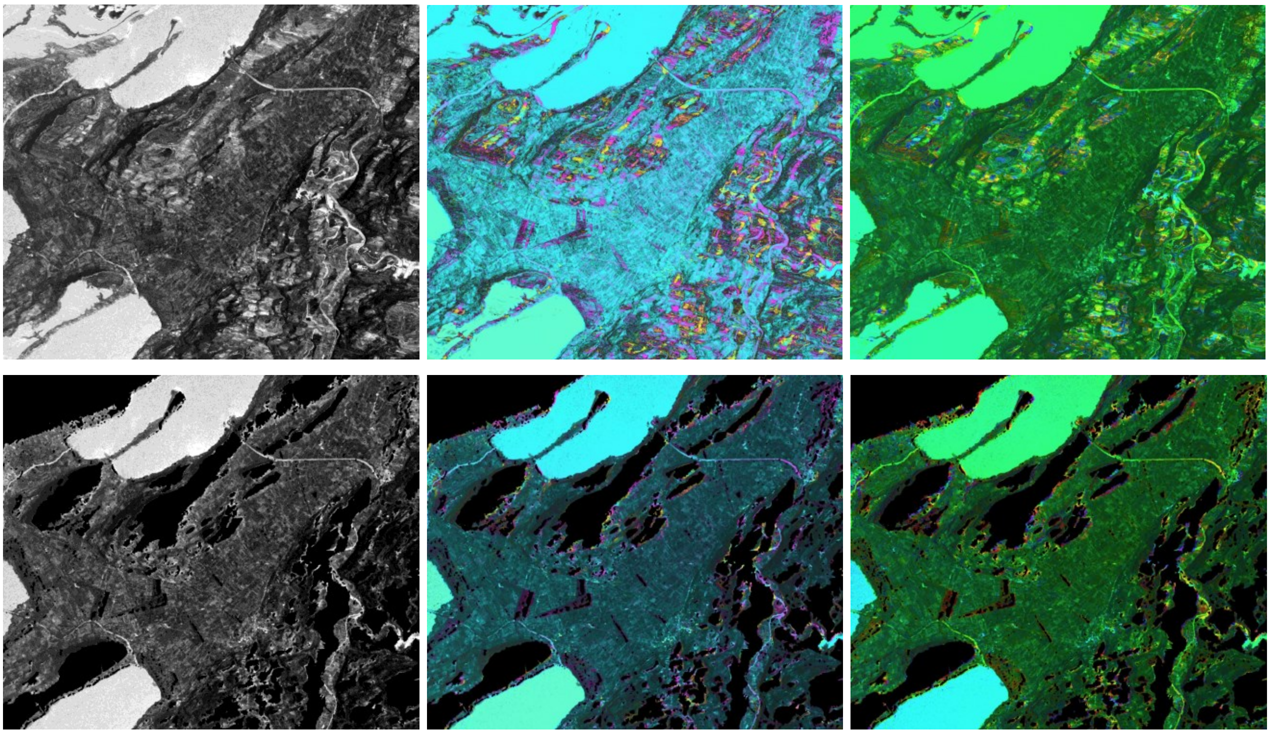

Figure 2 The SWOT backscatter (left), the single-pass interferogram (center) and height corrections (right, using a color scale between -15m (red), 0m (green), and 15m (blue)) estimated based on the unwrapped interferometric phase were geocoded into map coordinates using our normal geocoding procedure (top) and using the “increasing slant ranges” method (bottom).

Reader program for Capella data processed with the polar format processor: par_Capella_SLC

The Capella data reader program *par_Capella_SLC* was updated to also support reading data processed by the polar format processor. An example is shown in Figure 3 (see also README_Capella_pfa_Etna_demo).

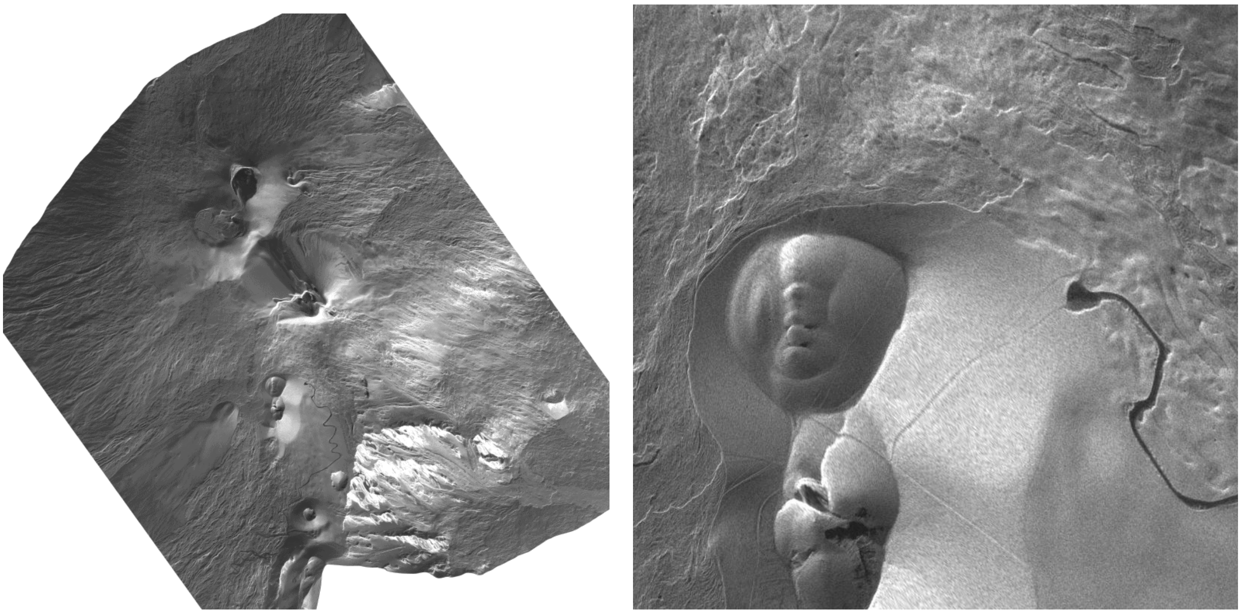

Figure 3 Capella data over Mount Etna, Italy, processed with the polar format processor was imported using *par_Capella_SLC* and then geocoded. The image to the left shows the overview over the scene and the image to the right gives a more detailed view on a small section of the same data set.

New scripts to co-register SLC and MLI data considering an offset refinement field

New scripts to automate the co-registration procedure in cases where ionospheric effects, ground motion, or SAR processing effects make it necessary to apply an offset field refinement step were added.

The demo example PALSAR3_iono_demo uses the new script SLC_coreg_refine.py to refine co-registration of two simulated PALSAR-3 scenes (prepared using PALSAR-2 data) affected by ionospheric effects.

The demo example README_Capella_pfa_Etna_demo uses the new script MLI_coreg_refine.py to co-register sub-apertures of a spotlight acquisition processed using a constant reference height..

New and updated SAR data readers

New / updated SAR data reader	**Short description**
par_SWOT_SLC	New reader program to generate SLC parameter and image files for SWOT level 1B KaRIn SLC data. See also: SWOT_Support_in_GAMMA (available at: https://www.gamma-rs.ch/uploads/media/2024-8_TR_SWOT_Support_in_GAMMA.pdf).
par_EORC_PALSAR	The reader program was updated, so that it now also supports reading ALOS-4 PALSAR-3 SLC data in CEOS format. Initial tests using real PALSAR-3 stripmap data could be done.
par_NISAR_RSLC	Supports reading NISAR RSLC data in HDF5 format. It produces one or multiple SLC parameter file(s) and SLC data file(s). The user can either read all data provided in the HDF5 file or select which data should be read, by selecting the radar band (L/S), the frequency in case of split beam imaging (A/B) and/or the polarization (HH, HV, RH, RV, VH, or VV). The user can also select to only write the parameter files. The following radiometric calibrations are available: none (output in as DN values), beta0, sigma0 (ellipsoid), and gamma0 (ellipsoid). The output format is FCOMPLEX. Note that the program was tested using simulated products (based on PALSAR 1/2 data).
par_Capella_SLC:	The reader program was updated to also support reading Capella data in PFA geometry
par_CS_DGM, par_CSG_DGM	New reader programs for Cosmo-Skymed and Cosmo-Skymed second generation DGM data
par_SICD_SLC	Updated to support RCS radiometric calibration of the data.

Gamma Software Demo examples

The access to the Gamma Software Demo examples is limited to Gamma Software users with a valid license. The access information is provided with the software delivery. A list of the Demo examples is available here:
https://gamma-rs.ch/uploads/media/README_Gamma_Software_demo.html.

New / modified demo example:	**Contents**
PALSAR3_iono_demo	This demo, added in mid 2024, was updated to facilitate some of the steps using new scripts such as SLC_coreg_refine.py.
README_Capella_pfa_Etna_demo	In this demo we reading and analyze Capella dwell mode data processed by the polar format processor.

MSP

-

ISP

SLC_cat, SLC_cat_list.py: New option to apply gain correction relative to first SLC using relative intensity of overlap areas, in addition to the method using calibration gain values in the parameter files.

par_Capella_SLC: Now also supports Capella data in PFA geometry.

*par_CS_DGM, par_CSG*_DGM: New programs to generate MLI parameter and image files for COSMO-Skymed and COSMO-Skymed Second Generation DGM data.

par_CS_DGM, par_CSG_DGM, par_CS_SLC, par_CSG_SLC: The HTML-based documentation and the program outputs now provide clearer information regarding radiometric calibration of CSK and CSG products.

par_CS_DGM, par_CSG_DGM, par_CS_SLC, par_CSG_SLC, par_NISAR_RSLC, par_ICEYE_SLC: Now reading the data block-by-block instead of line-by-line, speeding up considerably the reading process.

par_EORC_PALSAR: The program can now also read ALOS-4 PALSAR-3 SLC data in CEOS format.

SLC_deskew: Now takes into account the nominal terrain altitude. Robustness improved in case state vectors only tightly cover the acquisition and when the near range distance is shorter than the sensor altitude.

par_SWOT_SLC: New program to generate SLC parameter and image files for SWOT level 1B KaRIn SLC data.

par_SICD_SLC: Now also supports RCS radiometric calibration. In addition, in case no beta0 radiometric calibration polynomial is available, the program will try to perform the beta0 radiometric calibration using the RCS radiometric calibration polynomial and vice versa.

SLC_deskew: New options [sr0] and [sr2] to manually specify the output near and far range distances, instead of calculating these values from the input.

ISP_io, typedef_ISP.h: Added the INSTR_delay parameter to the GPRI_PAR data structure. Reading and writing INSTR_delay in seconds as a double value. It is the electronic delay in the GPRI-II due to the electronics, cables, and antennas. The reference component is the deramp mixer. The current best estimate of the delay is 4.22292145e-8 seconds equivalent to 6.33 meters.

DIFF&GEO

SLC_cat_all: New option to apply gain correction relative to first SLC using relative intensity of overlap areas, in addition to the method using calibration gain values in parameter files.

ScanSAR_coreg.py, ScanSAR_coreg_pol.py, ScanSAR_coreg_stack.py: New options --mask1 and --mask2 to use as an alternative to or in combination with the polygon areas. Option --mask1 will restrict the use of the intensity matching method to the non-zero areas in the mask, while option --mask2 will do the same for the spectral diversity method. The masks are in the same geometry as the reference MLI image.

base_calc, base_plot: Plot method modified from '-' to datablock following changes in GNUPLOT version 6.

geocoding.py: The file extension of the geocoded MLI image is now the same as the file extension of the input MLI image.

MLI_coreg_refine.py: New script to refine co-registration of an MLI using an offset field.

SLC_coreg.py: New option --n_ovr to define oversampling factor in offset estimation.

SLC_coreg_refine.py: New script to refine co-registration of an SLC using an offset field.

radcal.py: New script to perform radiometric calibration of multi-looked intensity (MLI) and single look complex (SLC) images between specified input and output radiometric calibrations. The following radiometric calibration types are supported: beta0, sigma0 ellipsoid, gamma0 ellipsoid, sigma0 terrain, gamma0 terrain.

par_TX_geo: Update to make sure values in the MLI parameter file are plausible. They now fully correspond to a multi-looked image in slant-range / azimuth geometry that might have been possibly used to produce the geocoded image.

2PASS_INT, 2PASS_UNW: These scripts now use *geocoding.py* for the geocoding step.

LAT

-

DISP

vis_colormap_bar.py, visbyte.py, viscpx.py, visdt_pwr.py, vismph_pwr.py, vispwr.py, visras.py: Backends (Matplotlib module) with exclusively lowercase names are now also supported. Improved robustness and error messages in case of already registered colormaps or in case of unknown colormaps.

IPTA

mb, mb_pt: Now supports a larger number of patterns. A warning is provided in case of a large number of patterns. An improved error message is provided if the maximum number of patterns is reached.

kml_ts_pt, disp_plot_pt, gras_dat, gras_disp, gras_disp2d, gras_ipta, SWAP_io: Plot method modified from '-' to datablock following changes in GNUPLOT version 6.

mssptr_pt, mcf_pt: Addition of a check to make sure no point has been omitted when unwrapping using *mcf_pt*.

npt: Added capability to print out the min and max of the point coordinates.

TDBP (Time-Domain Back-Projection)

The TDBP programs are available for Ubuntu 22.04 Linux and Windows 11 WSL2 with Ubuntu 22.04 Linux (and legacy 20.04 Ubuntu Linux and Windows 11 WSL2 versions).

The time-domain back-projection focusing program *az_proc_tdbp_gpu* employs parallelized TDBP imaging implemented in C/CUDA and therefore requires a Ubuntu Linux computer equipped with an NVIDIA GPU.

The TDBP module supports image focusing of SAR data from airborne platforms, UAVs, and mobile-mapping platforms such as cars, or trains, with high-quality 3-D geometry/motion compensation. Subsequent interferometric/tomographic processing and value-adding is supported. The TDBP module supports SAR image focusing of pulsed and FMCW SAR data with a generic sensor-agnostic data interface. In particular, SAR image focusing of FMCW SAR data acquired with our in-house Gamma SAR systems is also supported. See also the Gamma L-band SAR demo examples and our publications on ground motion/slope stability retrieval plus other SAR imaging examples obtained with Gamma SAR systems mounted on a car or a UAV and processed with the TDBP module: https://www.gamma-rs.ch/L-band-specific-Publications.

Python wrapper

py_gamma.py: New "rec_num" keyword parameter in ParFile constructor to read parameters for one record in a stack of parameter files (pSLC_par in the IPTA module).

py_gamma.py: Now checks for unsupported variable types when running a Gamma program.

Matlab wrapper

par_file.m: Fixed writing to parameter files.

GIS

With the end-of-2023 distribution the additional module *GIS* has been included. All the users with a valid license and running maintenance for the *GEO* or a combination of modules that includes the *ISP/DIFF&GEO* obtain the new additional module *GIS*. Notice that many of the programs of the *GIS* module require access to the *LAT*. Users without the *LAT* will not see the tools requiring *LAT* programs.

The *GIS* programs compose the "***Gamma Plugin for ArcGIS Pro***". This plugin supports running SAR data specific tools in the ArcGIS Pro environment (Windows only). The GIS module consists of several text files (README file, Change log file) and Python programs. The text files are in the main GIS directory, the Python programs and the related documentation (xml files that can be accessed within ArcGIS Pro) are located in the subdirectory scripts. The text file README_GIS is a listing of the contents of the GIS module.

arc_read_data.py: Support for COSMO-Skymed and COSMO-Skymed Second Generation DGM data has been added.

arc_read_data.py: Support for PALSAR-3 and SWOT SLC data has been added.

arc_read_data.py: The "RCS" radiometric calibration is now available for SICD data.

arc_geocoding.py: Radiometric calibration is now performed using the script "*radcal.py*".

All packages

HTML-based documentation: The HTML-based documentation has been updated to improve its consistency, for easier maintenance and for a nicer look.