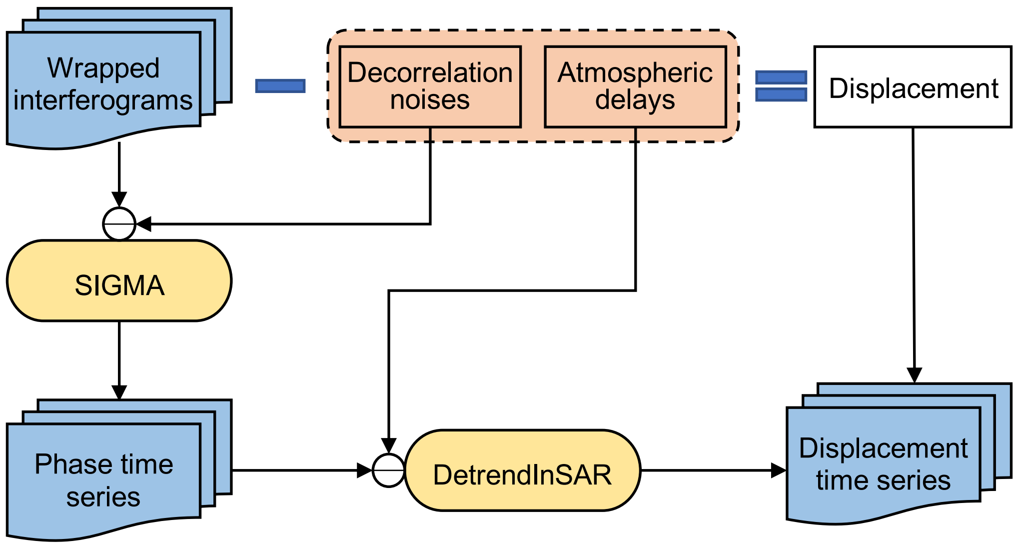
Manual for SIGMA & DetrendInSAR

Version 1.0



Jihong Liu, Jun Hu, Roland Bürgmann, Zhiwei Li, Sigurjón Jónsson

May 26, 2024

中南大學 CENTRAL SOUTH UNIVERSITY

جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

Berkeley UNIVERSITY OF CALIFORNIA

Contents

Chapter 1

Introduction

SIGMA, Strain-model based InSAR for Geo-hazards' Monitoring Approach, aims to decrease the decorrelation noise in DInSAR interograms based on strain model, and to obtain high-quality InSAR phase time series from short-baseline unfiltered differential interferograms.

DetrendInSAR, DEcrease both the TRENd and Dem-correlated components in InSAR time series, aims to decrease the atmospheric delays and orbital errors in InSAR phase time series (SIGMA output) based on the spatiotemporal characteristics of different components, and to obtain high-quality InSAR displacement time series.

These two methods serve as the post-processing procedure for InSAR displacement measurement based on unfiltered DInSAR interferograms (generated using other popular software). All processes are conducted in Matlab software, so users should be familiar with basic Matlab rules. Detailed introduction can be found in the following chapters.

Note that these two methods can be used independently.

The latest code can be downloaded from https://gip.csu.edu.cn/radar/xzzx/gkrj.htm, and please reference Liu et al. (2021, 2024):

Liu, J., Hu, J., Bürgmann, R., Li, Z., Sun, Q., & Ma, Z. (2021). A Strain-Model Based InSAR Time Series Method and Its Application to The Geysers Geothermal Field, California. Journal of Geophysical Research: Solid Earth, 126(8), e2021JB021939. https://doi.org/10.1029/2021JB021939

Liu J., Hu J., Bürgmann R., Li Z., & Jónsson S. (2024). Mitigating Atmospheric Delays in InSAR Time Series: The DetrendInSAR Method and Its Validation. Journal of Geophysical Research: Solid Earth, 129, e2024JB028920. https://doi.org/10.1029/2024JB028920

If you have any query, please feel free to contact me (liujihong@csu.edu.cn).

DEMO dataset can be found at https://dx.doi.org/10.5281/zenodo.11221799 (the latest version).

Figure 1-4 show examples of the performance of SIGMA and DetrendInSAR methods.

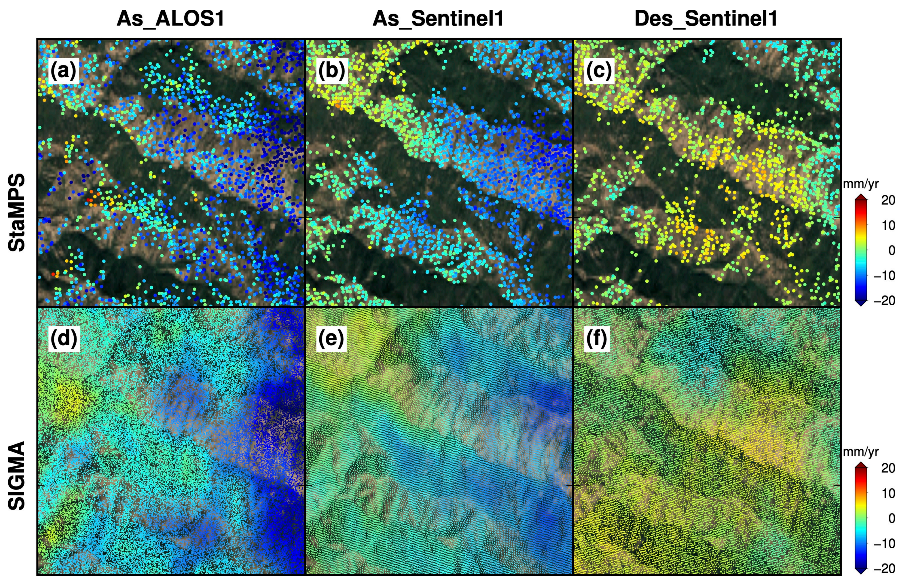


Figure 1. The displacement rate maps obtained from different methods and different datasets (Liu et al., 2021). It can be seen that SIGMA method can increase the density and accuracy of InSAR measurements compared with classical method.

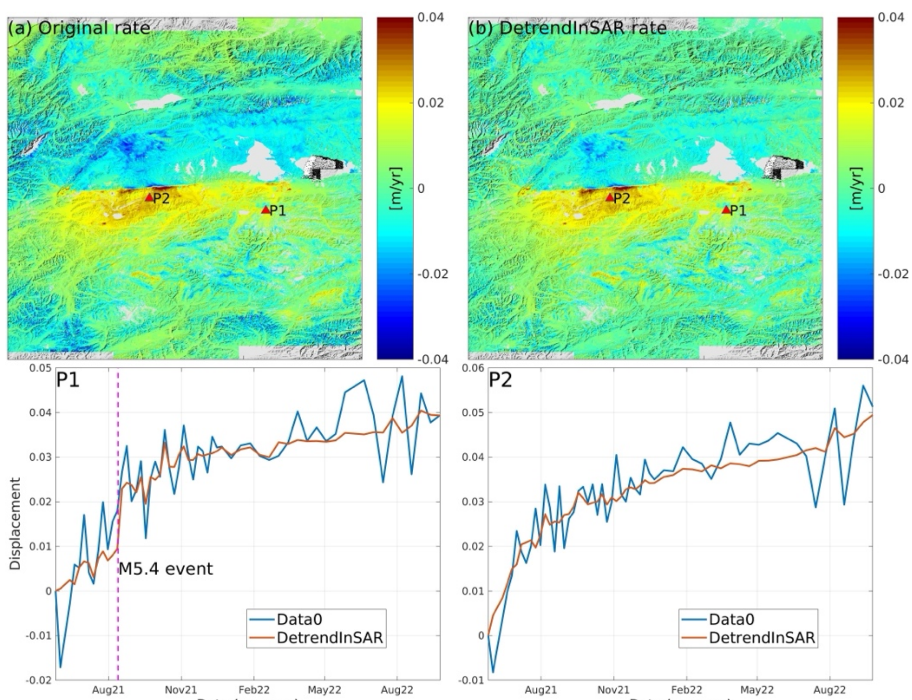


Figure 2. Displacements before and after DetrendInSAR correction for Maduo postseismic case. There was a magnitude 5.4 event after the main earthquake around P1, for the original time series, it’s hard to identify this event, but after the correction, it’s very clear to identify the occurrence and magnitude of this M5.4 event.

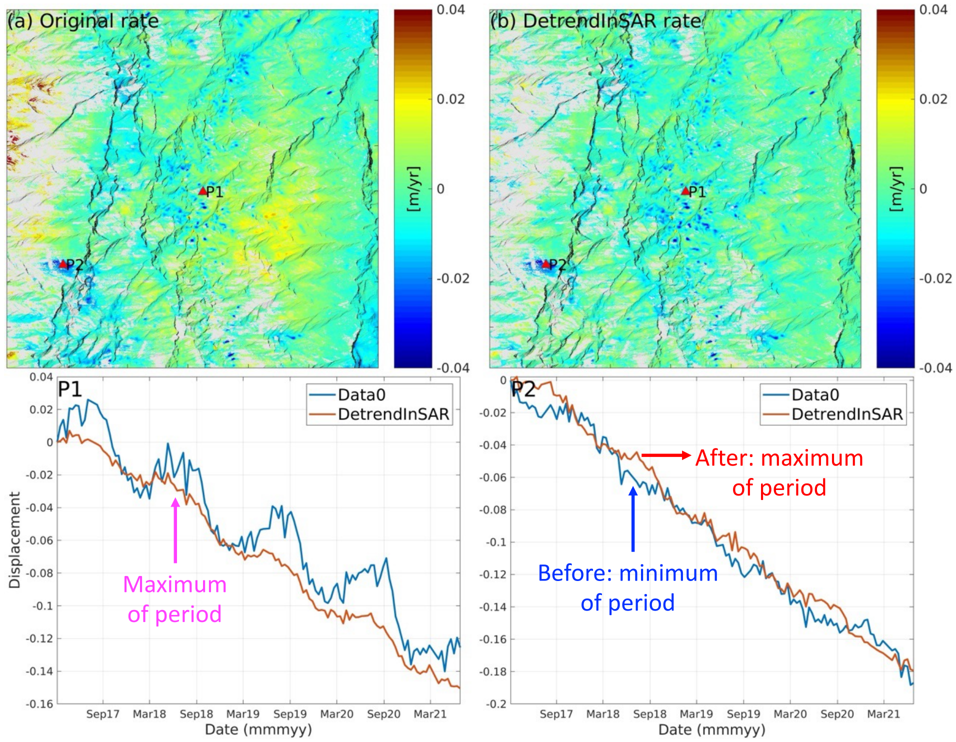


Figure 3. Displacements before and after DetrendInSAR correction for large-area landslides case. For P1, the magnitude of period signals is decreased after correction, where the maximum occurs around June. For P2, around June, the period displacement reaches it minimum for the original time series, while after correction, the occurrence of maximum is similar to P1.

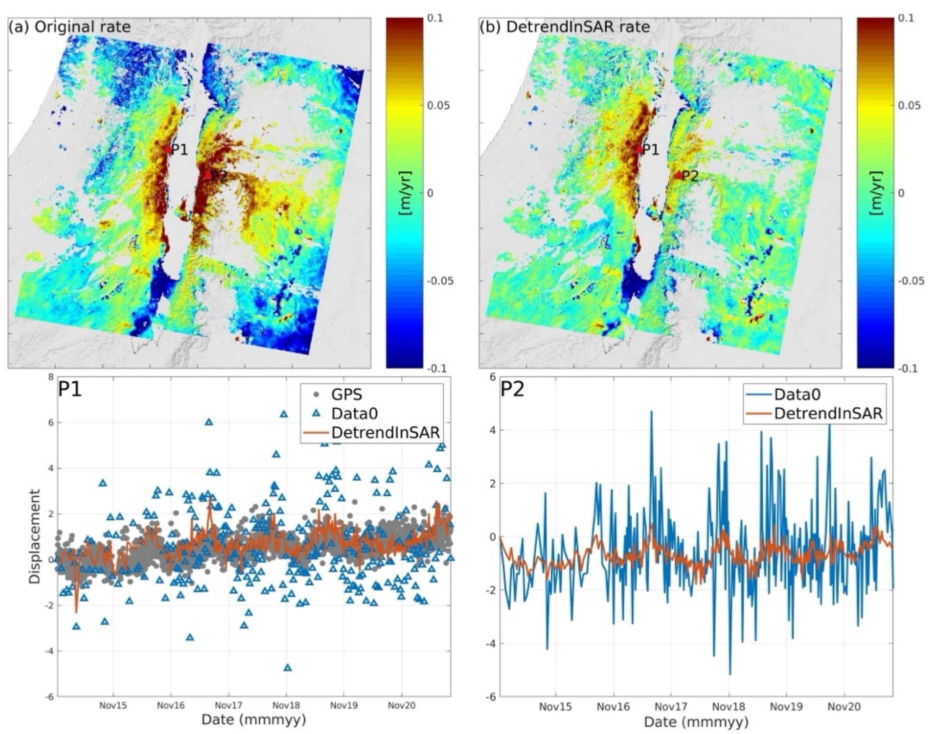


Figure 4. Displacements before and after DetrendInSAR correction for DeadSea rebound case. At P1, there is a GNSS station, as we can see, the corrected time series are consistent with GNSS even if the original time series are very noisy. The DEM-correlated delay in the right side can be well mitigated.

Chapter 2

SIGMA Method

2.1 Installation

The SIGMA package is implemented in Matlab language, and we recommend MATLAB 2018a (or higher version) to operate this package. When using the SIGMA, a new project folder should be created to include the folder `SIGMAInSAR_code` and the main script `SIGMAInSAR_main.m` (or `SIGMAInSAR_main_GUI.m`). The folder `SIGMAInSAR_code` includes the matlab functions of the SIGMA package, and the matlab script `SIGMAInSAR_main.m` (or `SIGMAInSAR_main_GUI.m`) outside the folder `SIGMAInSAR_code` is the main function to run the SIGMA method.

Users can directly run the main matlab script `SIGMAInSAR_main.m` (or `SIGMAInSAR_main_GUI.m`) to install this package, which is basically realized by using `addpath` function.

2.2 Input preparation

Here, we first introduce the input files used for SIGMA method, then share the shell code to generate these input files with GAMMA software.

2.2.1 Input files

(1) Multi-looked unfiltered DInSAR interferogram files in radar coordination

Each file indicates an interferogram, i.e., a `row*col` sized complex double matrix; In general, these interferograms are with short spatiotemporal baselines to keep high coherence;

(2) The coherence file for each interferogram

A `row*col` double matrix;

(3) Multi-looked SAR amplitude files

The amplitude of multi-looked SLC images, which is used to mask extreme low coherence area;

(4) The short baseline file

`nifg*4`, `nifg` is the number of short-baseline interferograms, `4` columns correspond to primary date, secondary date, spatial baseline, and temporal baseline (a text file);

(5) One SLC/MLI parameter file

A text file, includes some basic parameter used in the following analysis; if you don't have this file, just manually input the corresponding values in `SIGMAInSAR_main.m`.

(6) DEM file in radar coordination

A `row*col` sized double matrix;

(7) Longitude and latitude files in radar coordination

A `row*col` sized double matrix, to indicate the geographic coordinate of each pixel;

(8) Incident and azimuth angle file in radar coordination

A `row*col` sized double matrix, to indicate the look direction of InSAR data at each pixel;

These input files are also introduced in the main matlab script `SIGMAInSAR_main.m`. (1)-(3), (6)-(8) can be read/written using `freadbkB.m/fwritebkB.m` in folder `SIGMAInSAR_code`. To better illustrate the parameter setting when loading input file in `SIGMAInSAR_main.m`, Figure 5 shows the input file configuration during my preprocessing procedures (also see the DEMO dataset in section 2.5). You may use different pre-processing software from me, just import your own corresponding dataset into MATLAB, than use `fwritebkB.m` function to write the files required.

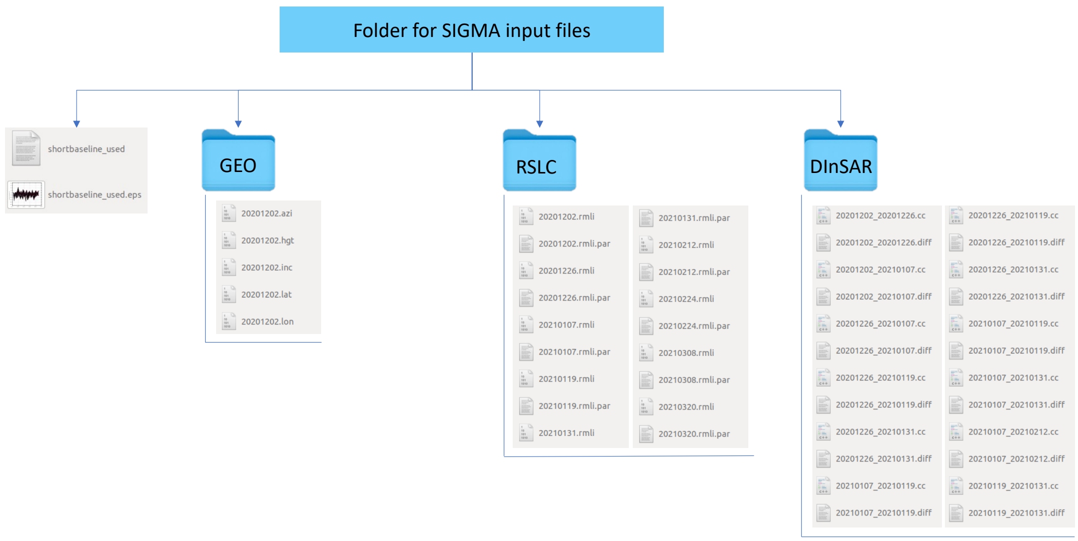


Figure 5. Example of the input files for SIGMA method

2.2.2 Prepare input files with GAMMA software

We share the shell code for generating the aforementioned input files based on the GAMMA software (i.e., Generate-Input-For-SIGMA-With-Gamma-2.0.0.zip, Figure 6). What users should prepare for using this code include coregistrated .rslc files, .rslc.par files, and the DEM files, like shown in the following screenshot (Figure 7).

The DEM files (i.e., srtm.dem and srtm.dem.par) can be obtained by using `gdal_merge.py` (can be found in the website) and `srtm2dem` (a GAMMA function) based on the usually used .tif format DEM data.

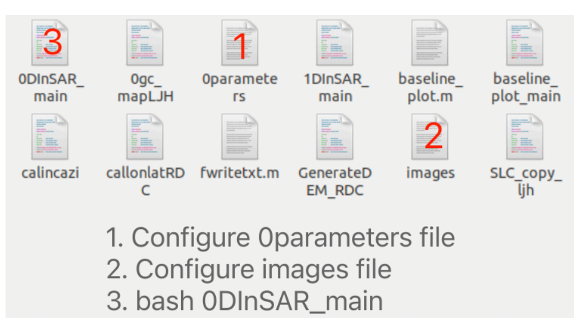


Figure 6. Screenshot of the shared shell code for generating interferograms based on GAMMA software.

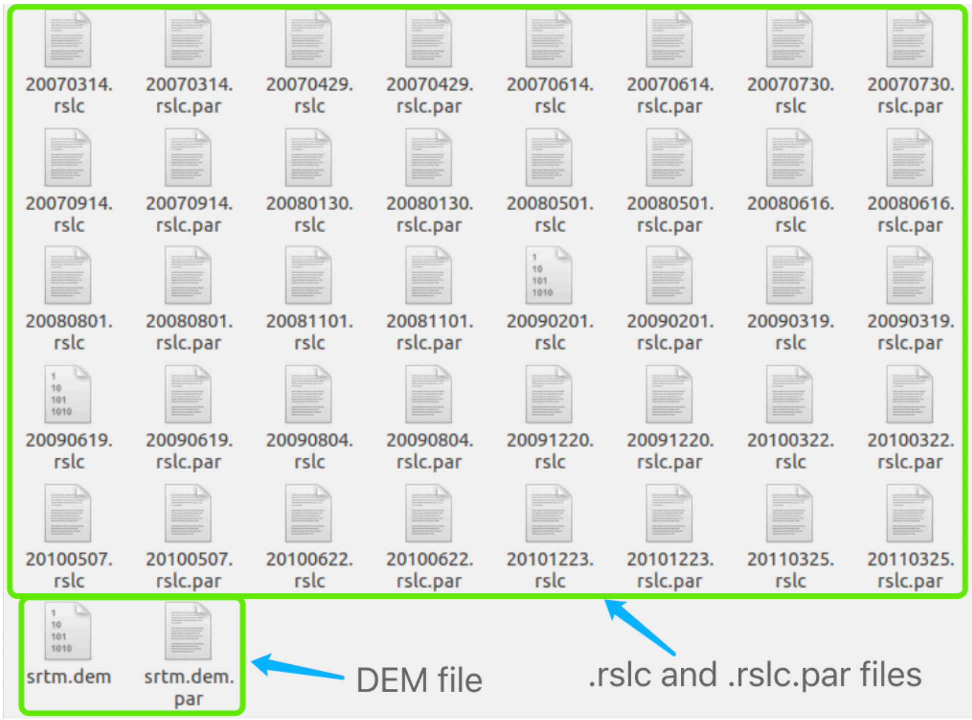


Figure 7. Example of input files for the shared shell code for generating interferograms based on GAMMA software.

After preparing the input files in Figure 7, you should also configure the `0parameters` file before using the shell code (as shown in Figure 8). Also, users should generate the `pimages` file, which indicates the SLC acquisition date. After preparing the aforementioned processes, just "`bash 0DInSAR_main`" to generate interferograms as well as other related files required for the SIGMA method.

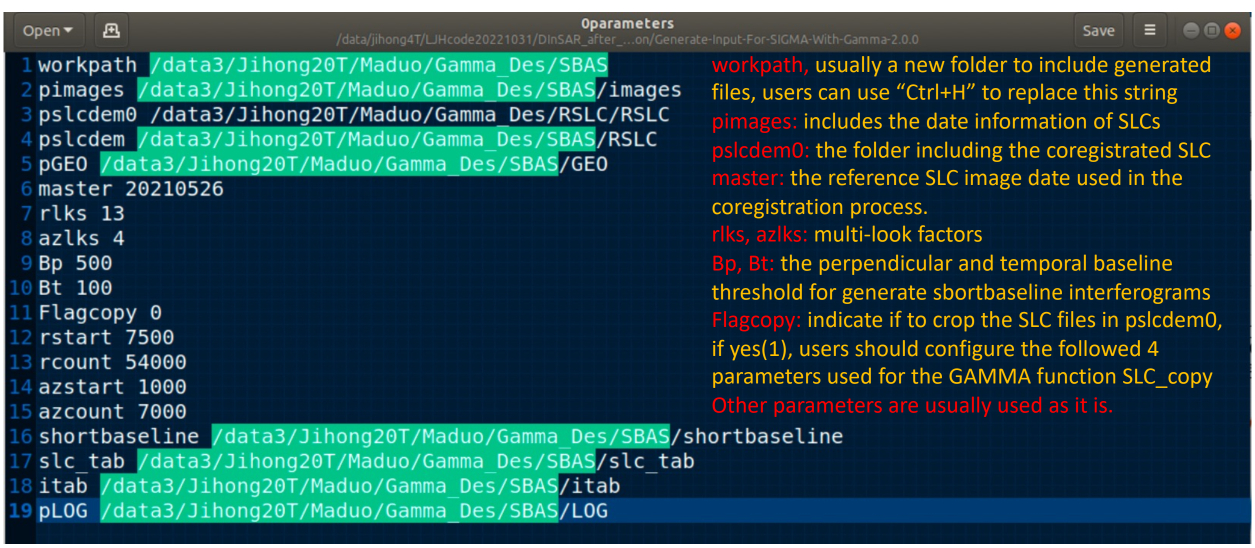


Figure 8. Explanation of 0parameters file for using the shell code

2.3 Run SIGMA

We provide two ways to run the SIGMA method, i.e., using the Matlab script code `SIGMAInSAR_main.m`, or run `SIGMAInSAR_main_GUI.m` to use a user interface.

2.3.1 Matlab Script

(1) Prepare the input as section 2.2;

(2) Configure the input parameters (Lines 16-74 in `SIGMAInSAR_main.m`, apart from the file path parameters, others usually use the default setting).

(3) Reset the threshold of `thrPS` in Lines 164-169, and iteratively run Lines 164-231 to select satisfied PS/DS candidate coverage. For example, if the selected points are too sparse, you can increase `thrPS.AD` or decrease `thrPS.cc`.

(4) After (3), just run the following code to obtain the final InSAR phase time series.

2.3.2 Matlab User Interface

Figure 9 shows the user interface of SIGMA method. In the following, all parameters shown in the interface will be introduced. Users can refer to Figure 5 to better understand this user interface.

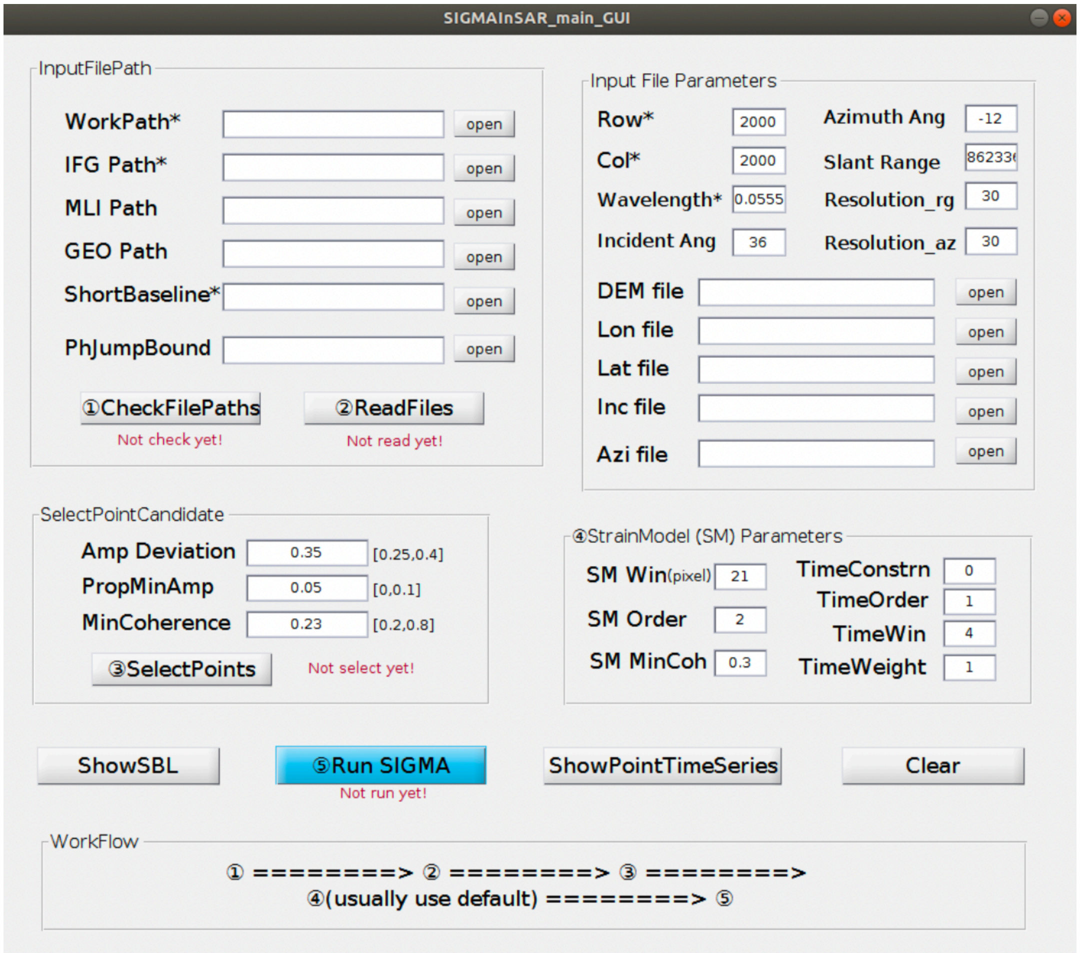


Figure 9. User interface for the SIGMA method

InputFilePath

WorkPath*: The folder path for SIGMA input files in Figure 5.

IFG Path*: The DInSAR folder path in Figure 5.

MLI Path: The RSLC folder path in Figure 5.

GEO Path: The GEO folder path in Figure 5.

ShortBaseline*: the shortbaseline_used file in Figure 5.

PhJumpBound: this is a text file, used to indicate the boundary where there may be phase jump, e.g., the fault rupture. An example of this file can found in "SIGMA_code/fault.xy".

Input File Parameters:

Row*: the number of row for the input interferograms and other files (unit: pixel).

Col*: the number of column for the input interferograms and other files (unit: pixel).

Wavelength: the wavelength of the used SAR data (unit: meter).

Incident Ang: the incident angle (unit: degree). (not used in SIGMA, but the key parameter for displacement interpretation).

Azimuth Ang: the azimuth angle (unit: degree). (not used in SIGMA, but the key parameter for displacement interpretation).

Slant Range: the distance between satellite to ground (unit: meter). (will be used if estimate the DEM residual)

Resolution_rg/az: the spatial resolution of one pixel in range and azimuth direction (unit: meter).

DEM, Lon, Lat, Inc, Azi file: the DEM, longitude, latitude, incident angle, and azimuth angle in RDC radar coordinate. These files with size of Row*Col are generated using the shell code in Figure 6.

SelectPointCandidate:

Amp Deviation: the threshold of amplitude deviation value to select PS/DS candidate. The larger this value is, the more pixels selected.

PropMinAmp: a proportion value, we sorted the all-pixel's amplitude value, and take those pixels with the smallest amplitude value in this proportion as the water area, and not processed in the following procedures.

MinCoherence: the coherence threshold to select the PS/DS candidate.

StrainModel Parameters

SM Win: the window size for optimize the central phase based on SM.

SM Order: the order used for SM, i.e., the number of modelled component $z=a_1x+ a_2y+ a_3xy+ a_4x^2+ a_5y^2$, default 2 means $z=a_1x+ a_2y$.

SM MinCoh: indicate pixels within the window with coherence smaller than this value are not included to establish the Strain Model.

TimeConstrn: if using additional temporal constraint? When the interferograms include more than one subset, it is impossible to directly obtain the time series based on the topology relationship between interferograms and time series. In this case, we can assume the time series smoothly change in the temporal domain, e.g., can be fitted using a time-dependent lineal model, then the time series from interferograms become solvable. Usually, this kind of constraint is not used, i.e., TimeConstrn=0.

TimeOrder: the assumed time-dependent polynomial order.

TimeWin: How many acquisitions around the target date are used to constrain the time series.

TimeWeight: how strong this kind of constraint added to the model.

Run SIGMA with user interface

(1) Using the open button to select the path of folders and files. If users prepare the input files as shown in Figure 5, then after you select the WorkPath*, other required paths of folders and files will be automatically filled. If not, you should select the folders and files one-by-one. <u>All fields with * are required for the SIGMA method, and others are optional.</u>
(2) Users should click the buttons as shown in the WorkFlow. There may be red alerts in the text boxes when click "CheckFilePaths", which are acceptable for running the SIGMA.
(3) When selecting points, users can adjust the three parameters to obtain a satisfied coverage of point candidates, then continue to run SIGMA.
(4) Users can click ShowSBL and ShowPointTimeSeries to plot the used short baselines and to select one pixel to show its time series, respectively.

Note

SIGMA method aims to decrease the decorrelation noises with spatially high frequency, thus the atmospheric delays with spatially low frequency will be reserved in the final

InSAR phase time series. In this case, the following DetrendInSAR method can be used to suppress the atmospheric delays and achieve InSAR displacement time series measurement. In the other hand, severe decorrelation noises will result in nonnegligible phase closures and fake temporal cumulation, which is especially obvious in areas with dense vegetation coverage. Further investigations on mitigating such heavy decorrelation noises are required.

2.4 Output

The main output is the InSAR phase time series `defoTS` stored in `SIGMA_output.mat` (here using "`defo`" term just means the unit being meter).

After running the SIGMA code, there will be several files generated:

Point_candidate.jpg: A picture to show the selected PS/DS candidates, the background is the average magnitude image. This is generated at Line 226 in `SIGMAInSAR_main.m`.

rate.png, rate.png.kml, and rate_colorbar.png: the calculated displacement rate, which can be imported into GoogleEarth software. This is generated at Line 305 in `SIGMAInSAR_main.m`.

SIGMA_output.mat: include the InSAR plase time series and other possible parameters used for further applications, e.g., the incident angle, azimuth angle…… This is generated at Line 313 in `SIGMAInSAR_main.m`.

SIGMA_ts_at_pixel_xxx_xx.png: show an example of the obtained phase time series. This is generated at Line 339 in `SIGMAInSAR_main.m`.

2.5 DEMO

Users can download the DEMO dataset at the link in page 1. Just unzip it, then run "`SIGMAInSAR_main.m`" or "`SIGMAInSAR_main_GUI.m`". The RAW Sentinel-1 SLC dataset are downloaded from ASF.

Chapter 3

DetrendInSAR Method

3.1 Installation

The DetrendInSAR package is implemented in Matlab language, and we recommend MATLAB 2018a (or higher version) to operate this package. When using the DetrendInSAR, a new project folder should be created to include the folder `DetrendInSAR_code` and the main script `DetrendInSAR_main.m`. The folder `DetrendInSAR_code` includes the matlab functions of the DetrendInSAR package, and the matlab script `DetrendInSAR_main.m` outside the folder `DetrendInSAR_code` is the main function to run the DetrendInSAR.

Users can directly run the main matlab script `DetrendInSAR_main.m` to install this package, which is basically realized by using `addpath` function.

3.2 Input preparation

The input of DetrendInSAR includes:

(1) the InSAR displacement time series `data`, i.e., a 3D matrix with each layer represents the cumulative displacement at the exact date, with size of `row*col*nslc`; this matrix could be the main output `defoTS` from SIGMA method;

(2) the Digital Elevation Model `dem`, i.e., a 2D matrix represents the DEM data, with size of `row*col`;

(3) the date (yyyymmdd) of the time series `SLC`, i.e., a 1D vector represents the date of each layer in `data`, with size of `nslc*1`;

(4) the relative perpendicular baseline of each date `perpBase` (optional, if no this data `perpBase=[]`), i.e., a 1D vector, with size of `nslc*1`;

(5) the gps time series used to assist the DetrendInSAR correction `gps` (optional, if no this data `gps=[]`), with size of `ngps`*(`nslc`+2), each row is [`row_gps`, `col_gps`, `gps displacement time series at each data acquisition`], [`row_gps`, `col_gps`] is the gps location in `dem` matrix, if no data for some acquisitions at gps station, fill the corresponding position with `nan`.

No matter what preprocessing software is used, the only input for the DetrendInSAR are the aforementioned five matrixes.

3.3 Run DetrendInSAR

We provide two ways to run the DetrendInSAR method, i.e., using the Matlab script code `DetrendInSAR_main.m`, or run `DetrendInSAR_main_GUI.m` to use a user interface.

3.3.1 Matlab Script

(0)Open `DetrendInSAR_main.m`.

(1) Prepare the input as section 3.2;

(2) Select a polygon and distributed points for non-zero and zero deformation assumptions, respectively.

a. Users should manually select a polygon to encompass the deformation area. See Line 34 in `DetrendInSAR_main.m` for how to select this polygon, and see section 3.5.1 for more information about this polygon. The information of this polygon will be stored as '`defozeropolygon.mat`' automatically after the first selection of this polygon, if you want to update the polygon information, please delete the file '`defozeropolygon.mat`' and re-run the code.

b. Users should also manually select distributed points to indicate that the deformation at these points is zeros. See Line 44 in `DetrendInSAR_main.m` for how to select these points, and see section 3.5.2 for more information about these points. The information of these points will be stored as '`defozeropoints.mat`' automatically after the first selection, if you want to update the points information, please delete the file ‘`defozeropoints.mat`’ and re-run the code.

(3) Just wait until the scripts finish (be patient, 500-1000 iterations are required for calculating the unknowns by `lsmr.m`).

3.3.2 Matlab User Interface

Figure 10 shows the user interface of DetrendInSAR method by running Matlab script `DetrendInSAR_main_GUI.m`. In the following, all parameters shown in the interface will be introduced.

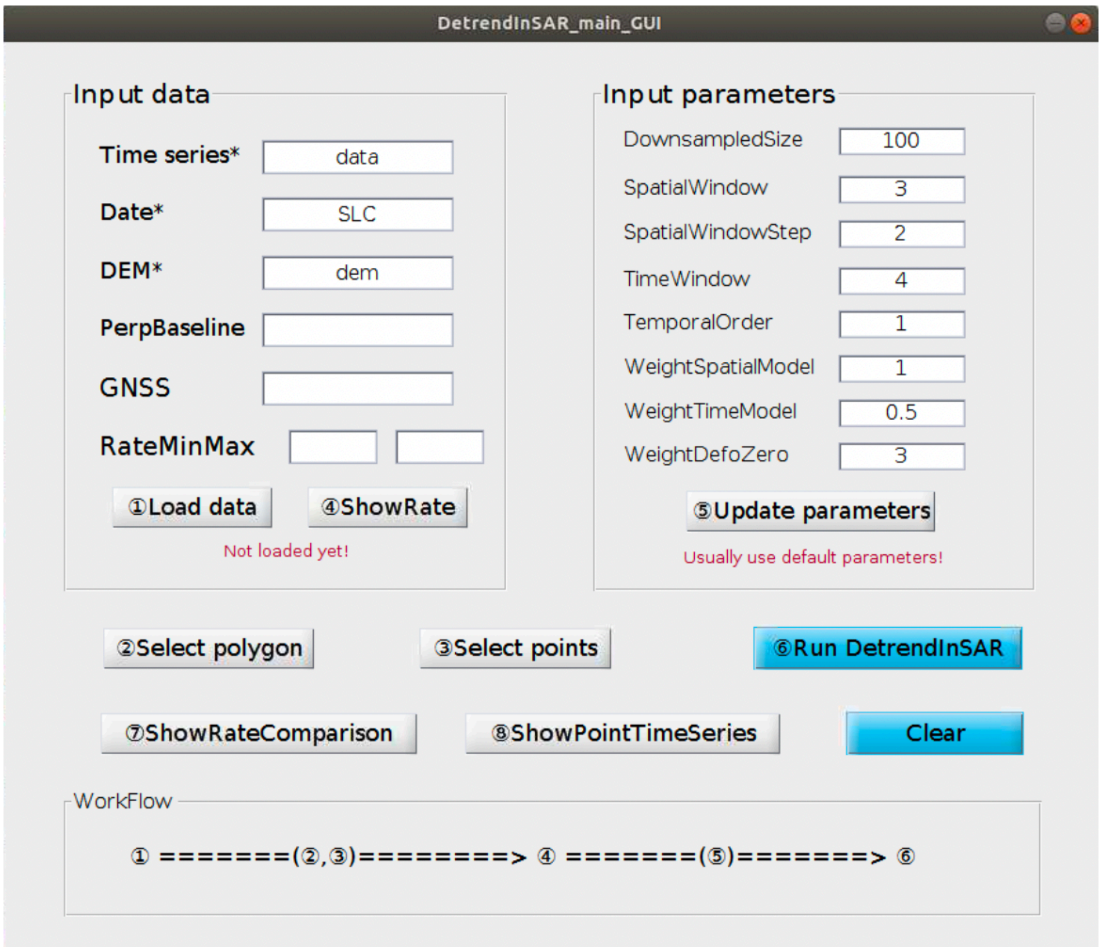


Figure 10. User interface for the DetrendInSAR method

Input data

Time series*, DEM*, Date*, PerpBaseline, and GNSS corresponds the input introduced in Section 3.2. Fields with * are required for DetrendInSAR, and others are optional.

RateMinMax: a limit corresponding to the colorbar range when plotting displacement rate map. These two values are automatically calculated when clicking Load data, users can also manually set these two values.

Input parameters: these parameters are recommended to use their default value.

DownsampledSize: Since the original InSAR time series have many pixels and it's hard to establish a large model for such a large number of pixels, we alternatively try to downsample original InSAR time series to a size of DownsampledSize* DownsampledSize. This parameter corresponds to the introduction in section 3.5.5.

SpatialWindow, SpatialWindowSetp: The spatial window used to model the central pixel, corresponding to the introduction in 3.5.3. For example, if SpatialWindow=3 and SpatialWindowSetp=2, this means that there will be (SpatialWindow+1)^2=49 pixels used to establish the model, while the maximum distance of these pixels from the central pixel is SpatialWindow* SpatialWindowSetp=6 pixel. We recommend to keep the ratio between the DownsampledSize and the maximum distance (SpatialWindow*SpatialWindowSetp). Larger SpatialWindow value mean heavy computational burden.

TimeWindow, TemporalOrder, and WeightTimeModel: as the instruction in section 3.5.4.

WeightSpatialModel: this is used to adjust the significance of the spatial model in the final DetrendInSAR model, usually using the value of 1.

WeightDefoZero: as the instruction in section 3.5.1, i.e., `weight_eq0`.

Run DetrendInSAR

(1) Before running the DetrendInSAR, users should first load the required matrix into the Matlab work space.
(2) Then, users can manually fill the corresponding blank with the variables name. Here, we set the default names as data, SLC, and dem. However, if users have

different variable name, they can manually change the string in the blank into their own variable name. For example, if my InSAR time series are stored in the Matlab as DefoTS, then just fill the blank of Time series* with “DefoTS”.

(3) After the required blanks are filled, click “Load data”. Users can click “ShowRate” to plot the initial displacement rate map, in which if there are existing information of defo!=0 area polygon and defo=0 points, these polygon and points will be plotted in the rate map.
(4) If users want to select defo=0 area polygon, click “Select polygon”; or to select distributed points, click “Select points”. See sections 3.5.1 and 3.5.2 for detailed information of the polygon and points.
(5) It is recommended to use the default Input parameters, but if users want try other values, please change the value and click “Update parameters”.
(6) Click “Run DetrendInSAR” to get the corrected InSAR time series.
(7) “ShowRateComparison” and “ShowPointTimeSeries” give users options to compare the displacement before and after the DetrendInSAR correction.

Note

It’s still a challenging task to correct atmospheric delays in InSAR, and our contribution aims to enrich the possible solutions in this topic. Of course, the current default parameters in DetrendInSAR cannot realize the optimal correction for all cases, hence there are some parameters in function `DetrendInSAR_downsample.m` that can be modulated according to different cases. Details about these parameters are introduced in Section 3.5, and if you have any questions, please feel free to contact me (make sure introducing yourself in the email).

3.4 Output

The main output is the corrected InSAR displacement time series `data1` stored in `DetrendInSAR_final.mat`.

After running the DetrendInSAR code, there will be several files generated:

Area_with_defo_masks.jpg: Background is the displacement rate, where the manually selected polygon and points are also presented. This is generated at Line 56 in `DetrendInSAR_main.m`.

Compare_DetrendInSAR_OriginalInSAR_at_pixel_xxx_xxx.png: An example to compare the displacement time series before and after the DetrendInSAR correction. This is generated at Line 179 in `DetrendInSAR_main.m`.

Compare_DetrendInSAR_OriginalInSAR_downsampled.jpg: Compare the downsampled displacement rate before and after the DetrendInSAR correction. This is generated at Line 102 in `DetrendInSAR_main.m`.

Compare_DetrendInSAR_OriginalInSAR.jpg: Compare the displacement rate before and after the DetrendInSAR correction. This is generated at Line 153 in `DetrendInSAR_main.m`.

Comparison_downSampled.jpg: Compare displacement time series at several date before and after the DetrendInSAR correction. This is generated at Line 480 in `DetrendInSAR_downsample.m`.

defozeropoints.mat: the information of distributed points with defo=0. This is generated at Line 50 in `DetrendInSAR_main.m`.

defozeropolygon.mat: the information of the selected polygon with defo!=0 inside. This is generated at Line 39 in `DetrendInSAR_main.m`.

DetrendInSAR_downsample.mat: the calculated error (trend and dem-correlated noises) and displacement components by DetrendInSAR method for the downsampled dataset. This is generated at Line 84 in `DetrendInSAR_main.m`.

DetrendInSAR_errors.mat: the calculated error (trend and dem-correlated noises) after the interpolation for the downsampled result. This is generated at Line 116 in `DetrendInSAR_main.m`.

DetrendInSAR_final.mat: the DetrendInSAR-corrected displacement time series. This is generated at Line 183 in `DetrendInSAR_main.m`.

DetrendInSAR_x.mat: the unknowns in the DertendInSAR model calculated by lsmr.m. This is generated at Line 437 in `DetrendInSAR_downsample.m`.

3.5 Main parameters

Here presents the main parameters that should be defined by users or would affect the performance of DetrendInSAR, and some tips of how to set them are also presented.

3.5.1 Select a polygon to include areas with deformation

To distinguish the non-deformation area from the deformation area, users should select a polygon to cover the area with significant deformations, i.e., the `selfigbypoly` function in `DetrendInSAR_main.m`. In this case, the area outside this polygon would be considered as non-deformation so that the constraint of "deformation=0" would be added to the DetrendInSAR model at those non-deformation pixels.

The selection of this polygon will make the final displacement result clear, especially for the cases with only one primary deformation region, like the postseismic deformation of one earthquake, the subsidence of one underground activity, the surface rebound around a big lake.

For the DEMOs, the polygon information exists and is saved as "`defozeropolygon.mat`", if user start a new case, please delete this .mat file, and it's required to select this polygon by yourself when running `DetrendInSAR_main.m`. After the first selection of this polygon, the polygon information is automatically saved as "`defozeropolygon.mat`", and you should delete this file if you want to update the polygon information.

Please do NOT select a very tense polygon to cover only the deformation area, which means that the polygon should be larger than the exact range of deformation area. This will make the "far-field" area "clear", and at the same time will not remove deformation signals.

Of course, it's also feasible for select the polygon as large as possible (e.g., to include all pixels of your data), in this case, the constraint of "deformation=0" is not used in DetrendInSAR process.

Another parameter to affect the non-deformation area result is the weight of the constraint "deformation=0". This parameter is `weight_eq0` in Line 82 of the function `DetrendInSAR_downsample.m`, which indicate how strong the constraint of "deformation=0" works in the DetrendInSAR model. Large values of `weight_eq0` will result in clearer field in the non-deformation area, i.e., the result value of non-deformation area will close to zero, and vice versa.

Figure 11 shows an example when you should select a polygon during running the code.

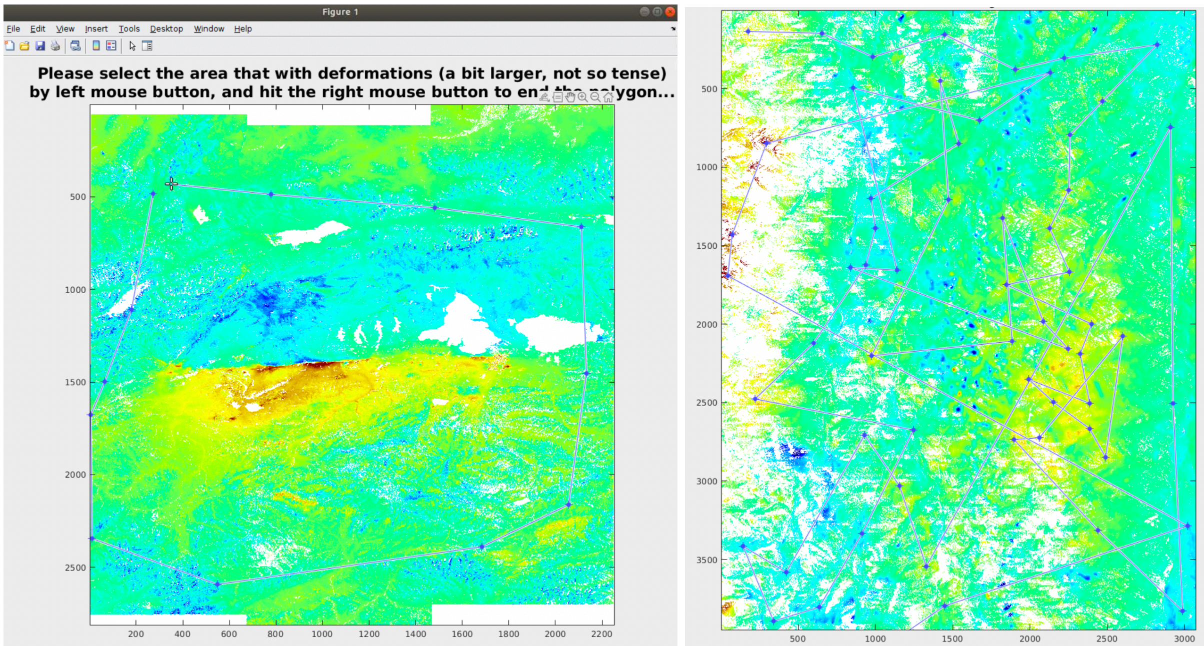


Figure 11. Left and right are examples to select the polygon and the distributed points.

3.5.2 Select distributed points to assume their deformation=0

Besides of the selected polygon in the previous section, the DetrendInSAR method also allows users to select distributed points with assumed deformation=0 to further constrain the correction. As shown in Figure 2(right), the vertexes are the selected points. The localized blue areas are potential landslides; therefore, the final displacement would be more reliable and clearer by assuming the deformations on these points are zeros.

3.5.3 The range for modeling trend and DEM-correlated components

The related parameters are in Lines 93-96 in "`DetrendInSAR_downsample.m`". `win2_ramp` means that there will be (`win2_ramp`+1)^2 pixels for modeling the central point, but the adjacent distance of these pixels used for modeling is `dsamp_win2_ramp`. Large value of `win2_ramp` will increase the computational burden.

3.5.4 The range for constraining temporal displacements

The related parameters are in Lines 106-108 in `DetrendInSAR_downsample.m`. Generally, it's recommended to use the default parameters.

`WT`: the weight of temporal constraint. Larger values mean smoother displacement time series, and vice versa.

`win2_tempPoly`: the number of acquisitions used for constraining the deformation of the central acquisition. Larger values mean smoother displacement time series, and vice versa.

`ord_tempPoly`: the order of temporal polygons. 1 for simple temporal behavior (e.g., linear, log) and 3 for complicated behavior (e.g., period).

3.5.5 Downsample degree

In order to accelerate the computation process of DetrendInSAR, the original data is downsampled in DetrendInSAR_downsample.m. The downsample degree is controlled by the parameter `ncount` in Line 55 of `DetrendInSAR_downsample.m`. Too large values mean much slow computation process. If you have a very powerful workstation, you can try any large value you want. For me, I prefer `ncount`=100.

3.5.6 Including GNSS data in DetrendInSAR model

The DetrendInSAR package is able to include GNSS data to assist the DetrendInSAR modeling. The required information of GNSS data is the position and the cumulative displacements at the SAR acquisition time, and see the parameter introduction of `gps` in `DetrendInSAR_main.m` for the detailed information.

One parameter to adjust how strong the GNSS data control the result is `weigh_gps` in Line 88 of `DetrendInSAR_downsample.m`. Large values of `weigh_gps` means strong constrain of the GNSS data to the result, vice versa. Users can use the default setting or try other values to see the performance.

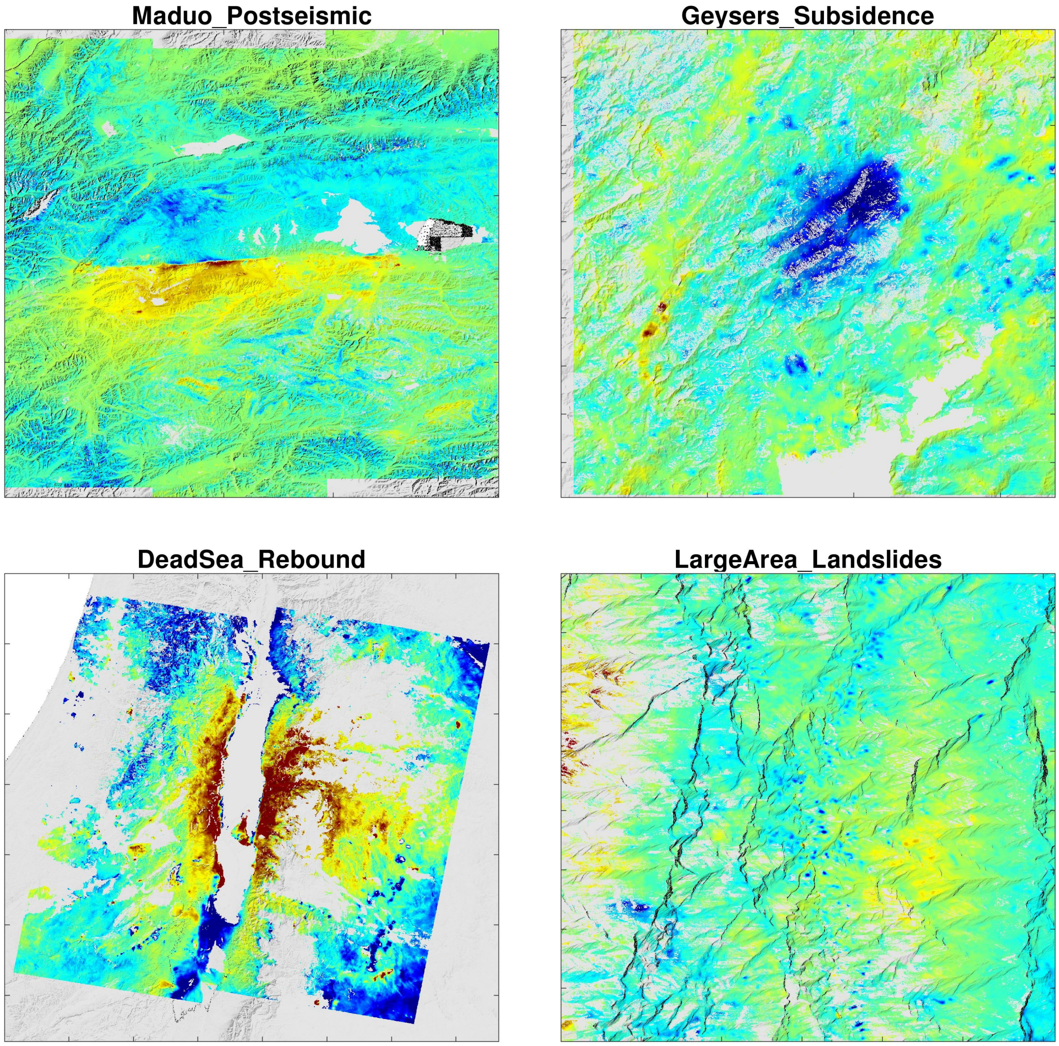


Figure 12. Displacement rate maps of four DEMO cases.

3.6 DEMO

For make users easily to understand the DetrendInSAR, we provide four DEMO cases, i.e., the postseismic displacement of the 2021 Maduo earthquake, the DeadSea rebound, the Geysers geothermal field subsidence, and the large area landslide identification (Figure 12), which can be downloaded at the link in page 1. ***Note that we use the same parameters for these four cases, and the displacement time series are plausibly reliable, indicating the robustness of these default parameter setting.*** The RAW Sentinel-1 SLC dataset are downloaded from ASF.

Chapter 4

Change history

4.1 Version 1.0

2024/04/29: Initial release of SIGMA and DetrendInSAR code.

2024/05/20: Simplified the `DetrendInSAR_downsample.m` function, provide more DEMO cases for the DetrendInSAR code.

2024/05/26: Design the user interface with Matlab; share the shell code for generating the input files of SIGMA method on GAMMA software; provide 4 DEMO cases for the DertrendInSAR method.