Small baseline InSAR time series analysis: Unwrapping error correction and noise reduction

Zhang Yunjun[a], Heresh Fattahi[b], Falk Amelung[a]

[a] Rosenstiel School of Marine and Atmospheric Science, University of Miami, Miami, Florida, USA

[b] Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, USA

Content of this file

Section S1. Supplemental figure S1 to S9 and table S1.

Section S2. Design matrices.

Section S3. Decorrelation noise simulation.

Section S4. Additional software features

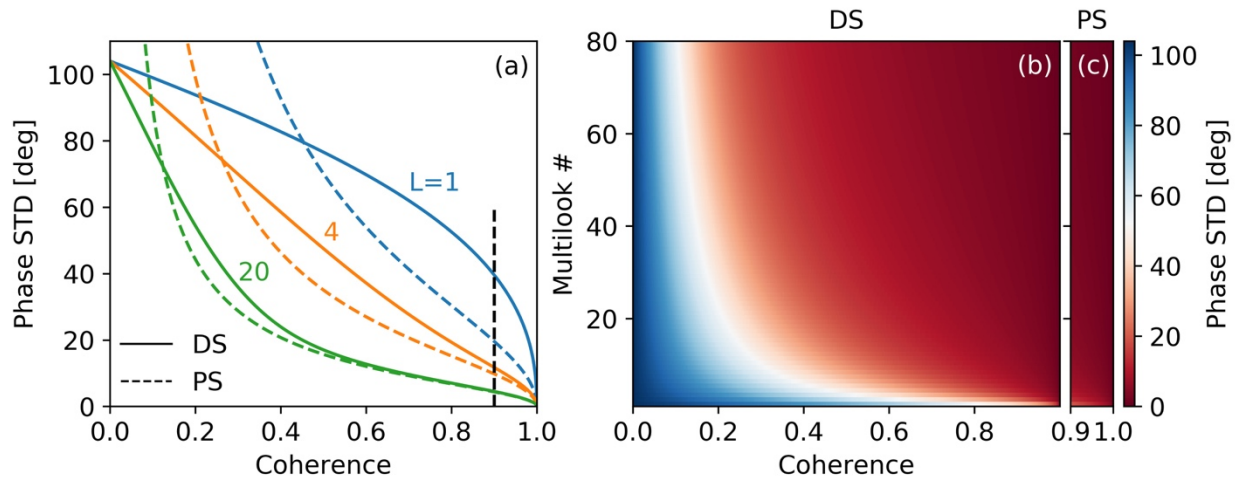Supplemental references.

S1. Supplemental figures and tables

This section provides figures S1 to S9 and table S1. Fig. S1 shows the standard deviation of the interferometric phase as a function of the spatial coherence and number of looks. Fig. S2 demonstrates the performance of four weighting functions in different temporal decorrelation settings using the mean RMSE of 10,000 realizations of the inverted phase time-series as a function of the number of looks. Fig. S3 demonstrates the simulation of the unwrapped

22   interferogram for unwrapping error correction with the bridging method, considering the ground

23   deformation, tropospheric turbulence, phase ramps and decorrelation noise. Fig. S4 shows the

24   output percentage of interferograms with unwrapping errors as a function of the LASSO

25   parameter to find its suitable value range. Fig. S5 demonstrates the necessity of adding the step

26   function during the topographic residual correction in the presence of displacement jump using

27   both simulated and read data. Fig. S6 shows the coherence matrix of Sentinel-1 dataset for GPS

28   stations within Sierra Negra. Fig. S7 shows the estimated residual phase time-series. Fig. S8

29   shows the coherence-based network modification for the Sentinel-1 data used in the discussion

30   of the network redundancy in section 6.3. Fig. S9 compares the displacement time-series from

31   the approaches in GIAnT and MintPy with and without unwrapping error correction and

32   weighted network inversion. Table S1 summaries the information of SAR data used in the paper

33   and their configurations for InSAR stack processing.

34

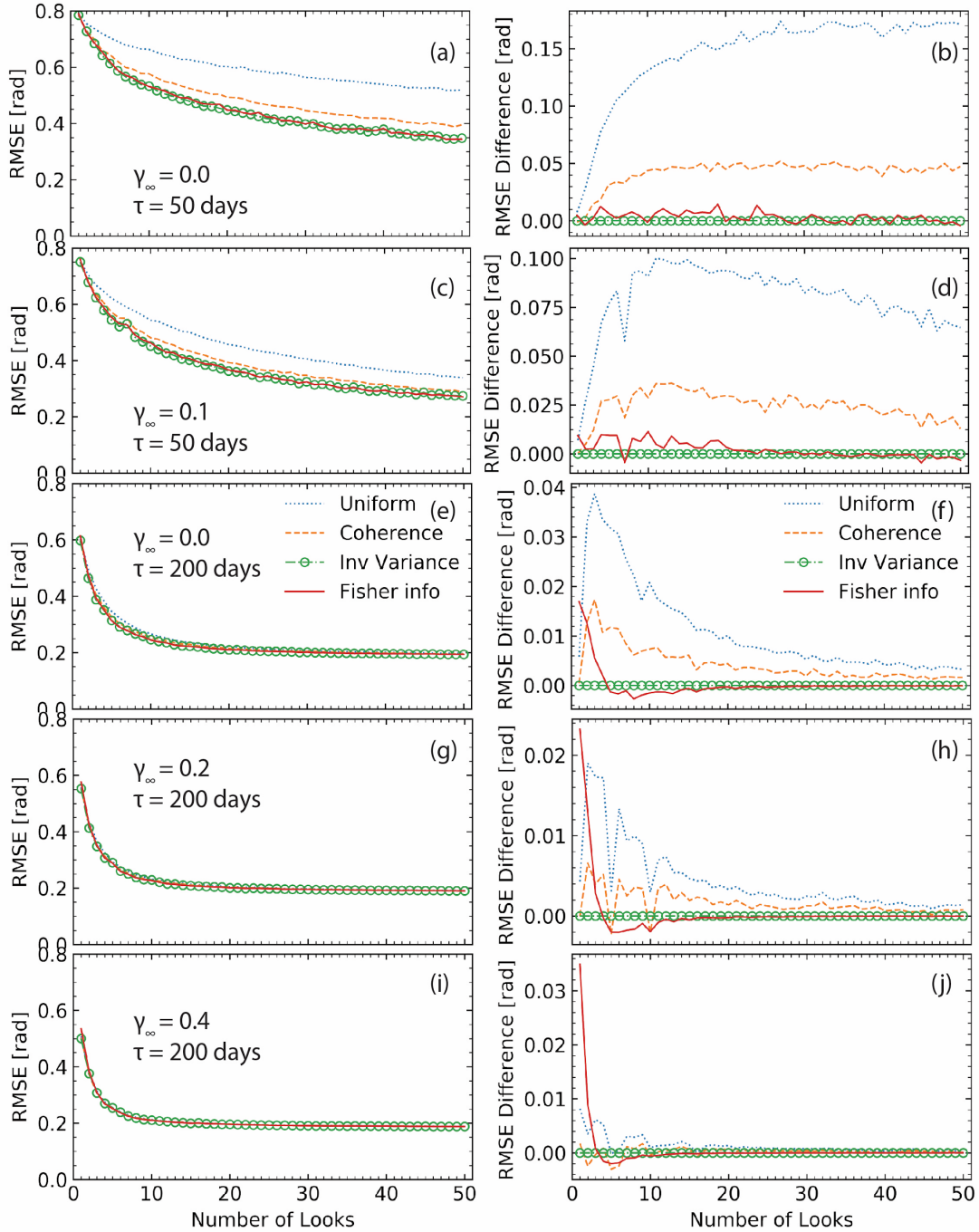**Figure S1.** Phase standard deviation versus spatial coherence for PS and DS. Related to equation (6). (a) Standard deviation of interferometric phase as function of coherence for DS (solid lines) and PS (dashed lines) with 1, 4 and 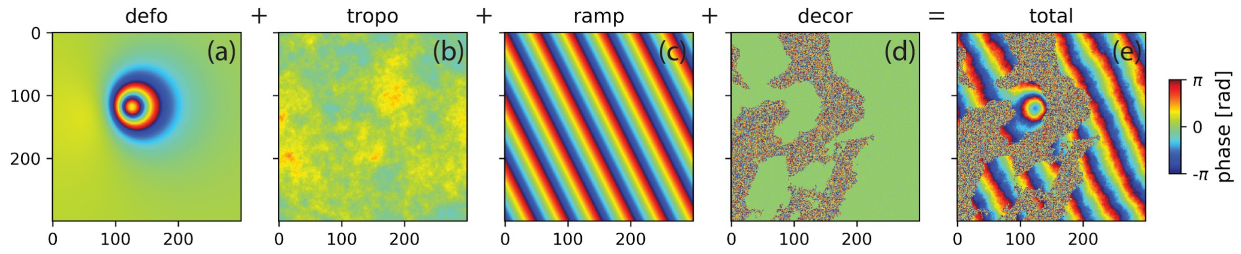20 looks. The black dashed line marks the effective boundary for PS ($0.9 < |\gamma| \leq 1$). (b) Lookup table to convert spatial coherence to phase standard deviation for number of looks in [1, 80].

Figure S2. Performance indicator for four weighting functions based on (left panel) the mean RMSE of 10,000 realizations of inverted phase time-series as a function of the number of looks. Related to Fig. 1, which uses $\gamma_\infty = 0.0$ and $\tau = 200$ days. Right panel: same as left panel but shown in differential RMSE with respect to inverse-variance weighting. From top to bottom for different temporal decorrelation settings.

Figure S3. Simulate interferogram for unwrapping error correction with the bridging method. Related to Fig. 2. We consider an area of 300 by 300 pixels with spatial resolution of 62 m in both directions, illustrated by radar echoes in a Sentinel-1-like geometry in descending orbit (with an incidence angle of 34 deg and heading angle of -168 deg). (a) Deformation phase caused by a Mogi source (x = 120 row, y = 120 col, z = 2 km under the free surface with a volume change of 10^6 m^3), (b) tropospheric turbulence modeled as an isotropic two-dimensional surface with a power law b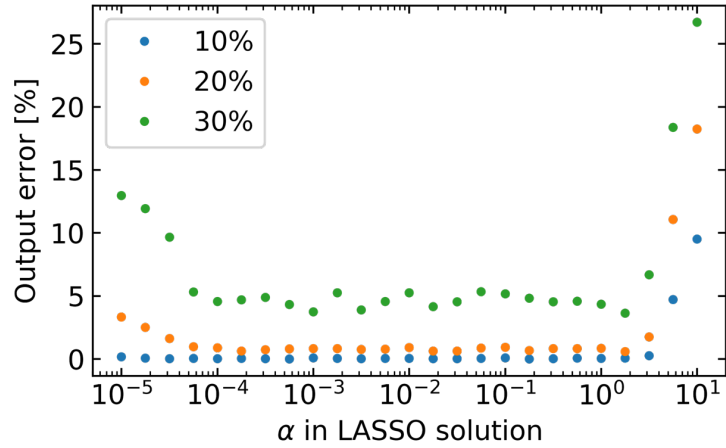ehavior (the multiplier of spectrum amplitude p0=1e-3, assuming a flat area without stratified tropospheric delay; Hanssen, 2001), (c) phase ramp modeled as a linear surface, and (d) simulated decorrelation noise (see section S3). The water body mask is rescaled from the real DEM in we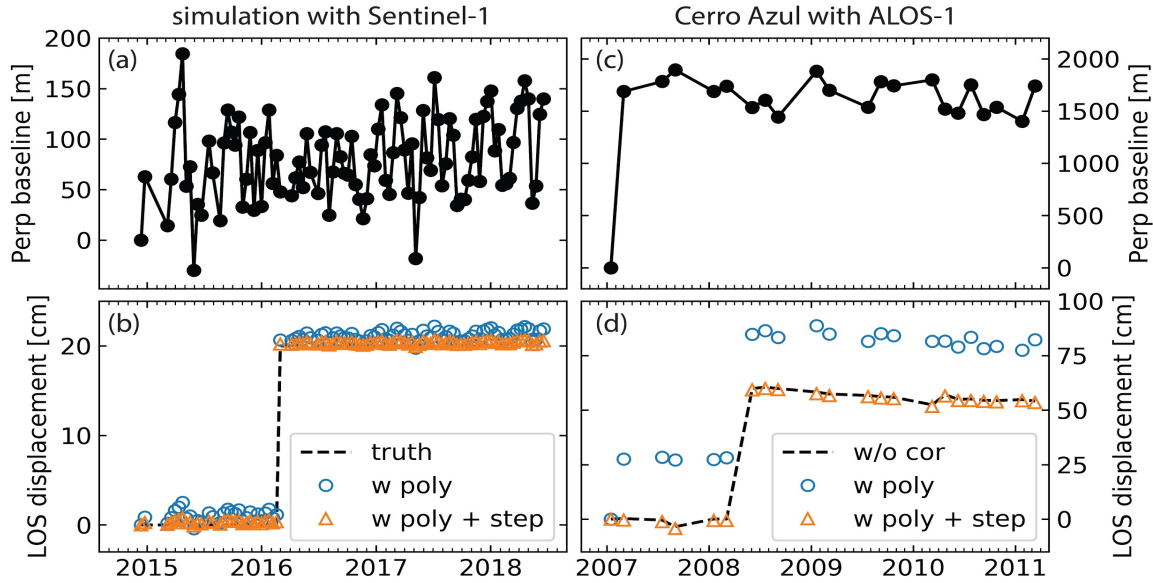stern Kyushu, Japan. We specify the spatial coherence of 0.6 and 0.001 for pixels on land and water respectively with the number of looks of 15 by 5.
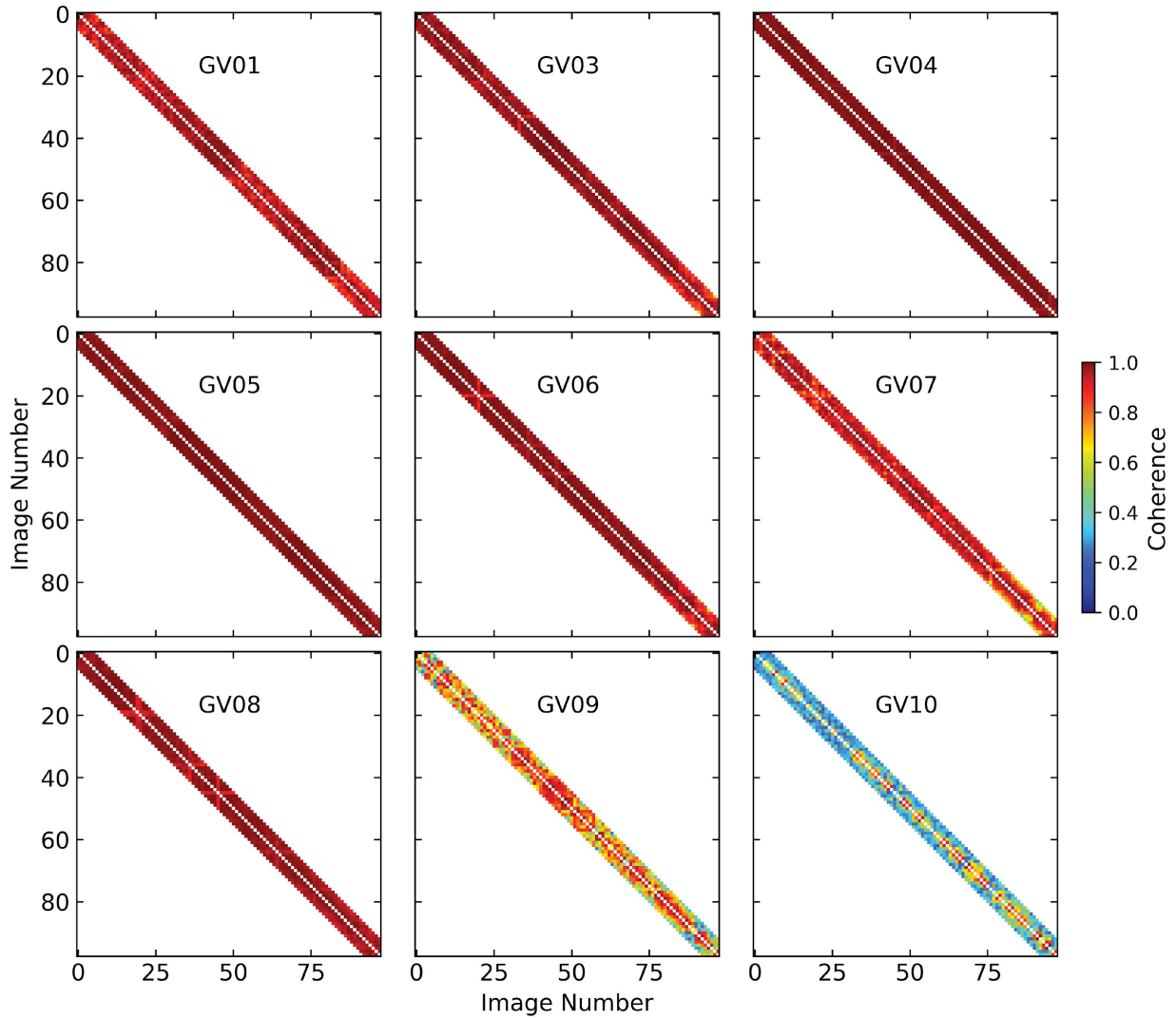
5

Figure S4. Simulation for the optimal LASSO trade-off parameter α. Related to equation (11) in section 3.2. Mean output percentage of 100 realization of interferograms with unwrapping errors after correction as a function of the nonnegative α value for different input percentage of interferograms with unwrapping errors. The network of interferograms is the same as Fig. 4a. The simulation result shows that any number of α in $[10^{-4}, 10^{0}]$ works. We choose 10^{-2} as default value.

Figure S5. Illustration of the step function in topographic residual correction in presence of displacement jumps. Related to equation (13) in section 4.8. (a and b) Perpendicular baseline history (from the Sentinel-1 data of section 5) and an arbitrary displacement time-series using simulated data (with a permanent displacement jump at 1 March 2016 with a magnitude of 20 cm, shown as the dashed black line in (b), in addition to the topographic residual contribution from a DEM error of 50 m). Blue empty circles and orange triangles represent displacement time-series after topographic residual correction assuming quadratic model without and with a step function, respectively. (c and d) Same as (a and b) but (i) using ALOS-1 data for one pixel on Cerro Azul located at [W91.270°, S0.928°] and (ii) the black dashed line for the displacement time-series without topographic residual correction. In both simulated and real data, the disagreement between the low-frequency quadratic model and the high-frequency displacement jump leads to biased estimation of the topographic residual (Du et al., 2007) and adding a step function could effectively eliminate this estimation bias. This estimation bias is amplified in the first ALOS-1 acquisition by its large perpendicular baseline (the difference between black dashed line and the blue empty circles in (d)).
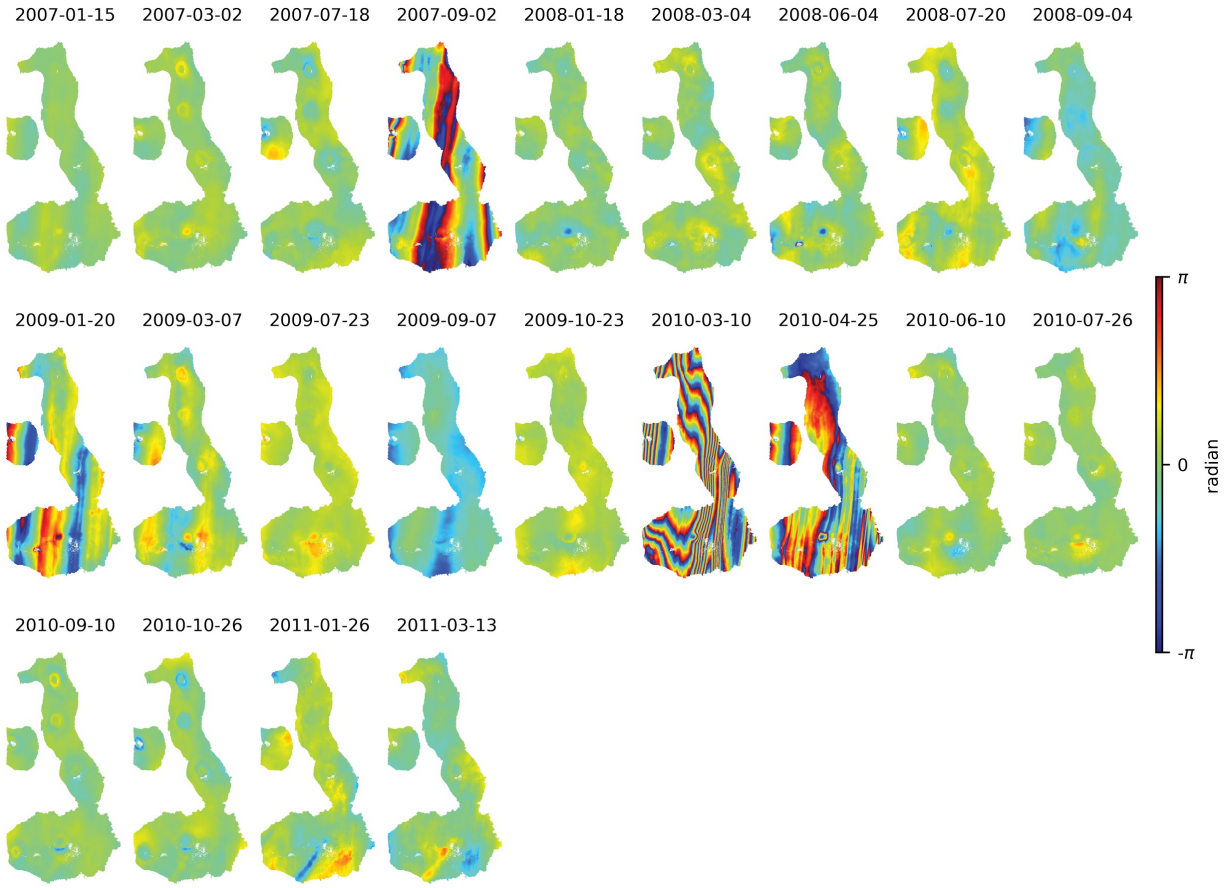
7

Figure S6. Coherence matrix of Sentinel-1 dataset for GPS stations within Sierra Negra caldera. Related to Fig. 8 in section 5.1. Both X and Y axis indicate the number of SAR acquisitions. Station GV10 is located in a densely vegetated area outside the caldera on the rim, resulting in fast decorrelation with low spatial coherence on interferograms with more than 2 lags.
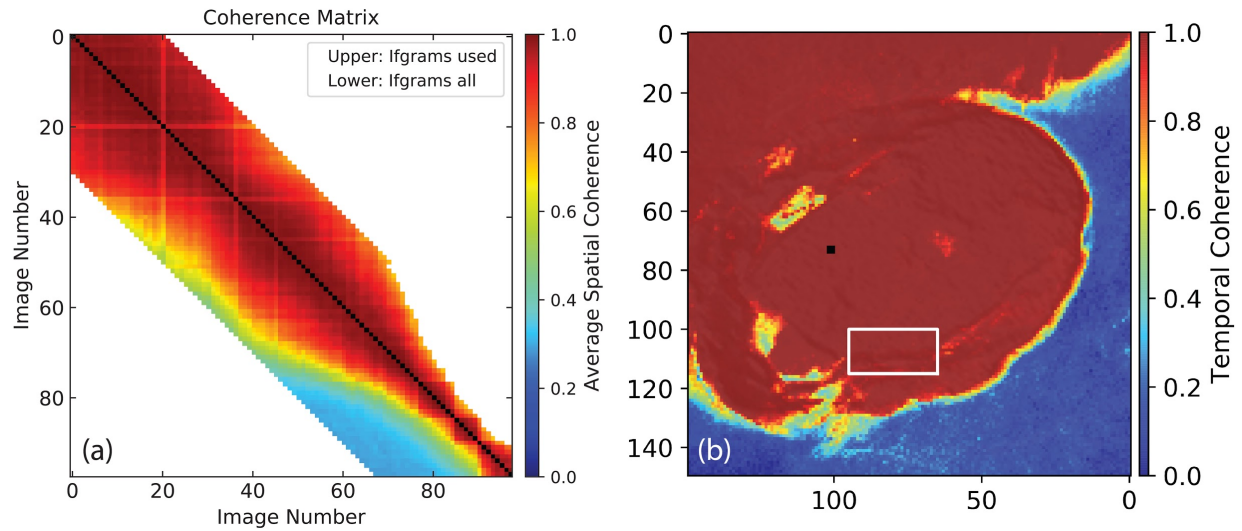
8

Figure S7. The estimated residual phase time-series $\hat{\phi}_{resid}$ of ALOS-1 dataset. Related to equation (13-14) in section 4.7 and Fig. 12 in section 5.4. A quadratic phase ramp has been estimated and removed from each acquisition. This is used in equation (14) to calculate the residual phase RMS value. Phases on 2 September 2007, 10 March 2010 and 25 April 2010 are severely contaminated by ionospheric streaks and are automatically identified as outliers. Phase on 20 January 2009 is contaminated by ionosphere also but is not identified as outlier due to its relatively small magnitude.
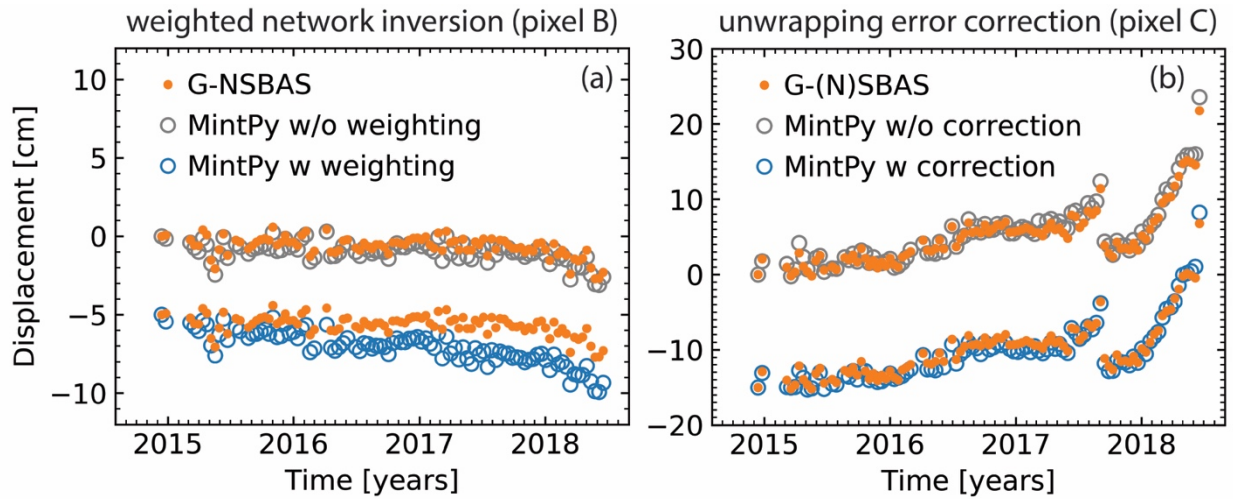
Figure S8. Coherence-based network modification for Sentinel-1 data used in section 6.3 in Sierra Negra. Related to Fig. 14 in section 6.3. (a) Coherence matrix of the customized area of interest along the trap door fault within Sierra Negra caldera (marked by the white rectangle in (b)). A network of interferograms with 30 sequential connections (2475 in total) are generated from 98 SAR acquisitions, as shown in the lower triangle. The upper triangle shows the interferogram kept after the network modification. A maximum of 20 connections are shown in Fig. 14 only. (b) Temporal coherence of the network inversion from the interferogram stack with a maximum of 20 connections.

Figure S9. Impact of (a) weighted network inversion and (b) unwrapping error correction on the displacement time-series. Related to Fig. 16 in section 6.5. The comparison within (a) shows that the difference on pixel B (Alcedo's flank) between MintPy and G-NSBAS is caused by the weighting during the network inversion. The comparison within (b) shows that the difference on pixel C (Fernandina's crater) between MintPy and G-(N)SBAS is caused by the unwrapping error correction.

118   **Table S1.** SAR dataset information with parameters used in InSAR stack processing

Satellite	ALOS-1	Sentinel-1A/B
Orbit direction	Ascending	Descending
Track number	133	128 (swath 1 & 2)
Start / end date (# of acquisitions)	2007-01-15 / 2011-03-13 (22)	2014-12-13 / 2018-06-19 (98)
Network selection criteria (# of Interferograms)	$B_{temp} \leq 1800$ days $B_{\perp} \leq 1800$ m (228)	Sequential with 5 connections (475)
# of looks in range / azimuth direction	8×16	15×5
Ground pixel size in range / azimuth direction (m)	60×51	62×70
InSAR Processor	ROI_PAC	ISCE
Phase Unwrapping	SNAPHU	SNAPHU

119

S2. Design matrices

This section shows examples to generate the design matrices used in the software. A demo set of $N = 8$ SAR images acquired at $[t_1,...,t_8]$ is used as the example. A stack of $M = 18$ interferograms is selected using the sequential method with 3 connections. An earthquake or volcanic eruption event occurred between t_6 and t_7 (red dashed line), which caused a permanent ground displacement offset.

Figure S10. *Network configuration of the demo dataset. Red dashed line marks the time of a displacement offset due to an earthquake or volcanic eruption.*

S2.1 Network inversion

To generate the design matrix A for network inversion used in equation (1) in section 2.1, we first generate an $M \times N$ matrix. For each row, it consists -1, 0 and 1 with -1 for the reference acquisition, 1 for the secondary acquisition and 0 for the rest. Due to the relative nature of InSAR measurement, the phase on the reference date (the first date by default) cannot be resolved, thus, we can only solve $[\phi^2,\ldots,\phi^N]$ instead of $[\phi^1,\ldots,\phi^N]$ and the corresponding column (the first column by default) is eliminated in the design matrix A, which results in size of $M \times (N-1)$.

137

$$A = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

138   (S1)

139

S2.2 Phase closure of interferograms triplets

141   Design matrix C describe the combination of interferograms to form the triplets used in equation

142   (10) in section 3.2 for the phase closure unwrapping error correction. An example of C is shown

143   below based on the demo network with number of triplets $T = 16$.

144   [1 -1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

145   [1 0 -1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0]

146   [0 1 -1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0]

147   [0 0 0 1 -1 0 1 0 0 0 0 0 0 0 0 0 0 0]

148   [0 0 0 1 0 -1 0 1 0 0 0 0 0 0 0 0 0 0]

149   [0 0 0 0 1 -1 0 0 0 1 0 0 0 0 0 0 0 0]

150   [0 0 0 0 0 0 1 -1 0 1 0 0 0 0 0 0 0 0]

151 $$\boldsymbol{C} = [0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ \text{-}1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0] \tag{S2}$$

152 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ \text{-}1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0]$$

153 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ \text{-}1\ 0\ 1\ 0\ 0\ 0\ 0\ 0]$$

154 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ \text{-}1\ 0\ 1\ 0\ 0\ 0\ 0]$$

155 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ \text{-}1\ 0\ 0\ 0\ 1\ 0\ 0]$$

156 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ \text{-}1\ 0\ 1\ 0\ 0]$$

157 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ \text{-}1\ 0\ 1\ 0]$$

158 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ \text{-}1\ 0\ 0\ 1]$$

159 $$[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ \text{-}1\ 1]$$

160 ## S2.3 Topographic residual correction

161 Design matrix $\boldsymbol{G}$ is used in equation (13) for topographic residual correction in section 4.8. It is

162 in size of $N \times (1 + N_{poly} + N_{step})$, where N_{poly} is the user-defined polynomial order N_{poly} (2 by

163 default), N_{step} is the number of Heaviside step functions (0 by default) describing offsets at

164 specific prior selected times. An example of $\boldsymbol{G}$ is shown below based on the demo network.

165

166
$$\boldsymbol{G} = \begin{bmatrix} \frac{4\pi}{\lambda}\frac{B_\perp^1}{rsin(\theta)} & 1 & (t_1 - t_1) & \frac{(t_1-t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda}\frac{B_\perp^2}{rsin(\theta)} & 1 & (t_2 - t_1) & \frac{(t_2-t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda}\frac{B_\perp^3}{rsin(\theta)} & 1 & (t_3 - t_1) & \frac{(t_3-t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda}\frac{B_\perp^4}{rsin(\theta)} & 1 & (t_4 - t_1) & \frac{(t_4-t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda}\frac{B_\perp^5}{rsin(\theta)} & 1 & (t_5 - t_1) & \frac{(t_5-t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda}\frac{B_\perp^6}{rsin(\theta)} & 1 & (t_6 - t_1) & \frac{(t_6-t_1)^2}{2} & 0 \\ \frac{4\pi}{\lambda}\frac{B_\perp^7}{rsin(\theta)} & 1 & (t_7 - t_1) & \frac{(t_7-t_1)^2}{2} & 1 \\ \frac{4\pi}{\lambda}\frac{B_\perp^8}{rsin(\theta)} & 1 & (t_8 - t_1) & \frac{(t_8-t_1)^2}{2} & 1 \end{bmatrix} \tag{S3}$$

167   Then equation (13) can be formed as a linear system with N equations as below:

169    $$\hat{\phi} - \hat{\phi}_{tropo} = \mathbf{G}X + \phi_{resid} \tag{S4}$$

171   where $X = [z_\varepsilon, c_0, c_1, c_2, s_7]^T$ is the vector of unknown parameters, $\hat{\phi}$, $\hat{\phi}_{tropo}$ and ϕ_{resid} are the

172    $N \times 1$ inverted raw phase time-series, estimated tropospheric delay time-series and residual

173   phase time-series, respectively. We apply the least squares estimation to obtain the solution as:

175    $$\hat{X} = (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T(\hat{\phi} - \hat{\phi}_{tropo}) \tag{S5}$$

176    $$\hat{\phi}_{resid} = \hat{\phi} - \hat{\phi}_{tropo} - \mathbf{G}\hat{X} \tag{S6}$$

178   The estimated residual phase $\hat{\phi}_{resid}$ is used to characterize the noise of phase time-series using

179   equation (14) in section 4.9. The noise-reduced displacement time-series is given as:

181    $$\phi_{dis}^i = \hat{\phi}^i - \hat{\phi}_{tropo}^i - \frac{-4\pi}{\lambda}\frac{B_\perp^i}{rsin(\theta)}\hat{z}_\varepsilon \tag{S7}$$

183   where $i = 1, \ldots, N$ and $\hat{z}_\varepsilon$ is the estimated DEM error in $\hat{X}$.

184   ## S2.4 Average velocity estimation

185   For each pixel, the average velocity is estimated as $d^i = vt_i + c$, where $d^i = -\frac{\lambda}{4\pi}\phi_{dis}^i$ is the

186   displacement at t_i in meters, v is the unknown velocity and c is the unknown offset. The solution

187   can be obtained using least squares approximation. An example of the design matrix $\mathbf{E}$ is shown

188   below based on the demo network.

189

$$E = \begin{bmatrix} t_1 - t_1 & 1 \\ t_2 - t_1 & 1 \\ t_3 - t_1 & 1 \\ t_4 - t_1 & 1 \\ t_5 - t_1 & 1 \\ t_6 - t_1 & 1 \\ t_7 - t_1 & 1 \\ t_8 - t_1 & 1 \end{bmatrix}$$ (S8)

190

191

192 For linear displacement, the uncertainty of the estimated velocity σ_v is given by equation (10) in

193 Fattahi and Amelung (2015) as:

194

195 $$\sigma_v = \sqrt{\frac{\sum_{i=1}^{N}(\phi_{dis}^i - \hat{\phi}_{dis}^i)^2}{(N-2)\sum_{i=1}^{N}(t_i - \bar{t})^2}}$$ (S9)

196

197 where $\hat{\phi}_{dis}^i$ is the predicted linear displacement at i_{th} acquisition $\bar{t}$ is the mean value of time in

198 years.

199

S3. Decorrelation noise simulation

S3.1 Coherence model

We simulate the coherence for a stack of interferograms on one pixel using a decorrelation model with exponential decay for temporal decorrelation. The spatial coherence γ^j of the j_{th} interferogram can be expressed as (Zebker and Villasenor, 1992; Hanssen, 2001; Parizzi et al., 2009):

$$\gamma = \gamma_{geom} \cdot \gamma_{DC} \cdot \gamma_{temporal} \tag{S10}$$

where γ_{geom} represents the geometric decorrelation, γ_{DC} represents the Doppler centroid decorrelation, $\gamma_{temporal}$ represents the temporal decorrelation, given by the equations below. Note that the thermal decorrelation $\gamma_{thermal}$ is served as the instantaneous decorrelation in temporal decorrelation $\gamma_{temporal}$ (Parizzi et al., 2009).

$$\gamma_{geom} = \begin{cases} 1 - \frac{|B_\perp|}{B_\perp^{crit}}, & |B_\perp| \leq B_\perp^{crit} \\ 0, & |B_\perp| > B_\perp^{crit} \end{cases} \tag{S11}$$

$$\gamma_{DC} = \begin{cases} 1 - \frac{|\Delta f_{DC}|}{B_{az}}, & |\Delta f_{DC}| \leq B_{az} \\ 0, & |\Delta f_{DC}| > B_{az} \end{cases} \tag{S12}$$
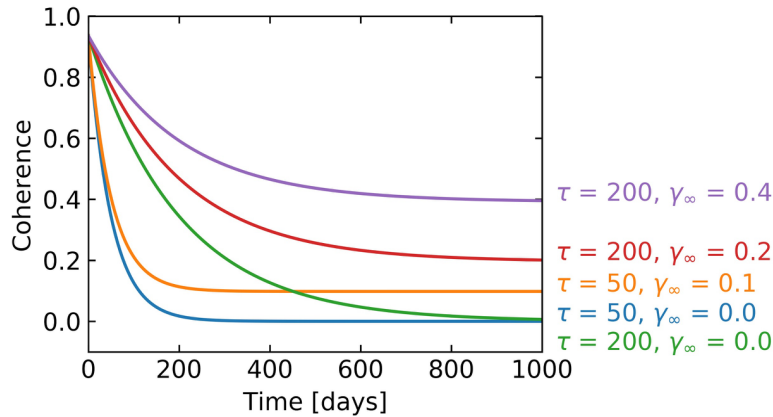
$$\gamma_{temporal}(t) = (\gamma_{thermal} - \gamma_\infty)e^{-t/\tau} + \gamma_\infty \tag{S13}$$

$$\gamma_{thermal} = \frac{1}{1 - SNR^{-1}} \tag{S14}$$

219 The critical perpendicular baseline $B_\perp^{crit} = \lambda \frac{B_{rg}}{c} R \cdot tan(\theta)$ is the baseline causing a spectral

220 shift equal to the radar bandwidth B_{rg} in range direction (Zebker and Villasenor, 1992; Hanssen,

221 2001), where λ is the radar wavelength, c is the speed of light, R is the distance between radar

222 antenna and ground target and θ is the incidence angle, *SNR* is the thermal signal-to-noise ratio

223 of radar receiver. τ is the time constant which depends on radar wavelength λ, it's the time for

224 coherence to drop down to $1/e$, i.e. 0.36, from its initial value (Parizzi et al., 2009; Rocca, 2007).

225 γ_∞ is the long-term coherence, or minimum attainable coherence value, which converged over

226 time, usually with high values in urban area and low values in vegetated area. Note that this

227 model does not consider the seasonal behavior of temporal decorrelation, volume decorrelation,

228 and processing-induced decorrelation. For a given set of SAR acquisitions, the geometric and

229 Doppler centroid decorrelation is almost constant among all pixels. All parameters are deployed

230 with typical parameters of Sentinel-1 SAR sensor.

231

232

233 **Figure S11.** *Simulated coherence as a function of temporal baseline, color coded by different τ*

234 *and γ_∞ settings used in Fig. S2.*

S3.2 Simulate decorrelation noise from coherence

For distributed scatterers (DS) in natural, vegetated terrain the interferometric phase exhibits highly unpredictable speckle characteristics. Its phase can be appropriately modeled by a random process, complex, stationary, circular Gaussian process in the case of SAR image. Applying the central limit theorem, the probability density function $pdf(\Delta\phi)$ of interferometric phase is obtained as (equation (66) from Tough et al., 1995; equation (4.2.23) from Hanssen, 2001):

$$pdf(\Delta\phi) = \frac{(1-|\gamma|^2)^L}{2\pi}\left\{\frac{\Gamma(2L-1)}{[\Gamma(L)]^2 2^{2(L-1)}} \times \left[\frac{(2L-1)\beta}{(1-\beta^2)^{L+\frac{1}{2}}}\left(\frac{\pi}{2}+arcsin\beta\right)+\frac{1}{(1-\beta^2)^L}\right]+D\right\} \qquad (S15)$$
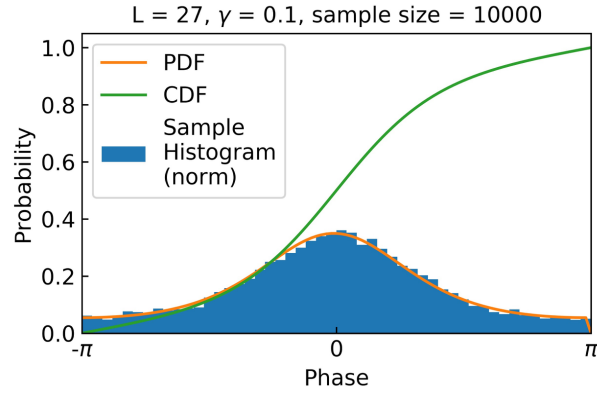
$$D = \frac{1}{2(L-1)}\sum_{r=0}^{L-2}\frac{\Gamma(L-\frac{1}{2})}{\Gamma(L-\frac{1}{2}-r)}\frac{\Gamma(L-1-r)}{\Gamma(L-1)}\frac{1+(2r+1)\beta^2}{(1-\beta^2)^{r+2}}$$

where $\beta = |\gamma|cos(\Delta\phi - \Delta\phi_0)$, expected interferometric phase $\Delta\phi_0 = E\{\Delta\phi\}$, gamma function $\Gamma(L) = \int_0^\infty t^{L-1}e^{-t}dt$, for $L \in R$ and D a finite summation term. Note that D vanishes for single-look datasets (L=1).

The 10,000 realizations/samples of decorrelation noise of each interferogram (used in section 2.4) is simulated by generating a distribution given by equation (S15) with corresponding coherence γ and number of looks L. One example with $\gamma = 0.1$ and $L = 9 \times 3$ is shown in Fig. S12.
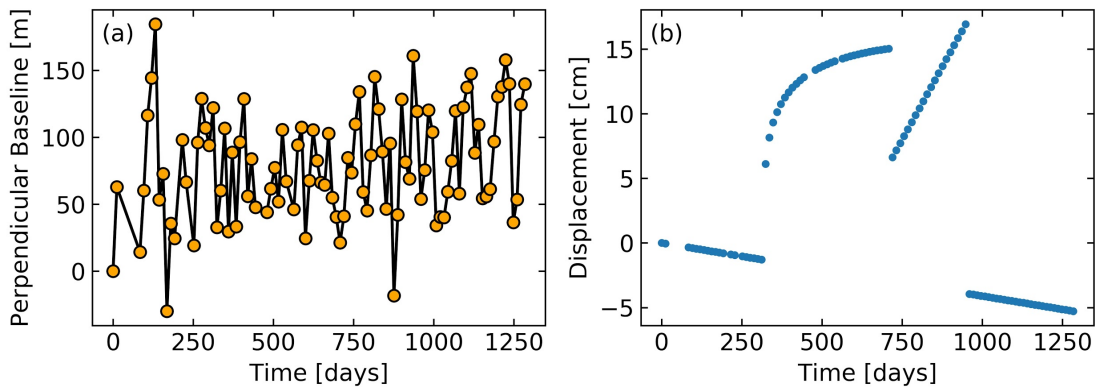
Figure S12. *Sampling the decorrelation noise based on phase PDF of distributed scatterers. Blue bars: normalized histogram of sampled decorrelation noises. Orange and green solid line: phase PDF and cumulative distribution function.*

Figure S13. *Time-series configuration for simulation. (a) Perpendicular baseline history from the 98 Sentinel-1 images of section 5. (b) Specified time-dependent displacement used in section 2.4 and 3.2.*
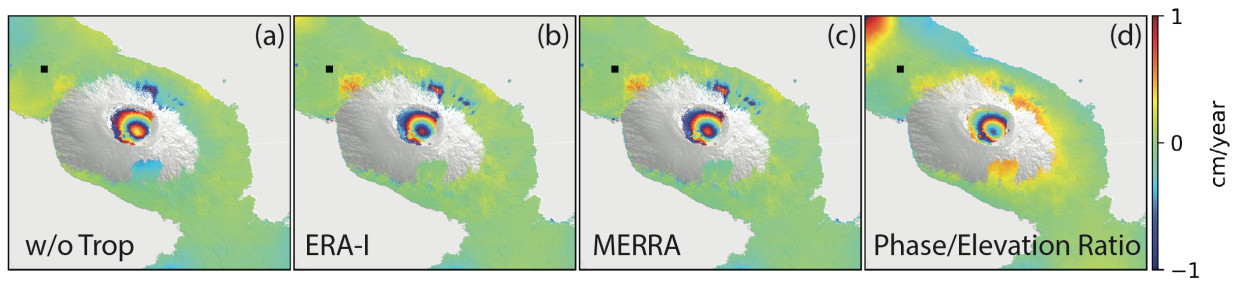
S4. Additional software features

S4.1 Customized workflow beyond `smallbaselineApp.py`

Most scripts in MintPy are stand-alone (summarized in Table S4). Users can apply any phase correction at any time to evaluate the impact. Fig. S14 shows an example, where we use individual scripts ([link on GitHub](#)) to compare velocities estimated from displacement time-series with different tropospheric delay correction methods on Alcedo volcano.

Figure S14. *Deformation velocity maps on Alcedo volcano from Sentinel-1 (a) without tropospheric correction, with tropospheric correction using (b) ERA-Interim, (c) MERRA-2 and (d) the empirical phase-elevation ratio method.*

Table S4. *Stand-alone scripts in MintPy*

`add.py`	Generate the sum of multiple input files
`asc_desc2horz_vert.py`	Project ascending and descending displacement in LOS direction to horizontal and vertical direction
`dem_error.py`	DEM error (topographic residual) correction
`diff.py`	Generate the difference of two input files

generate_mask.py	Generate mask file from input file
geocode.py	Resample radar-coded files into geo coordinates, or vice versa.
ifgram_inversion.py	Invert network of interferograms into time-series.
image_reconstruction.py	Reconstruct network of interferograms from time-series
image_math.py	Basic mathematic operation of input file(s)
info.py	Display metadata / structure of input file
load_data.py	Load a stack of interferograms into HDF5 files
load_gbis.py	Load the inversion result from GBIS software
load_hdf5.py	Load the binary file(s) into an HDF5 file
local_oscillator_drift.py	Correct local oscillator drift for Envisat data
mask.py	Mask input data file with input mask file by setting values on the unselected pixels into Nan or zero.
match.py	Merge two or more geocoded files which share common area into one file.
modify_network.py	Modify the network setting of an ifgramStack HDF5 file.
multilook.py	Multilook input file.
plot_coherence_matrix.py	Plot the coherence matrix of one pixel, interactively.
plot_network.py	Plot the network configuration of an ifgramStack file.
plot_transection.py	Plot the value of 2D matrix along a profile.
prep_aria.py	Prepare input data from ARIA GNUW products
prep_gamma.py	Prepare metadata file for GAMMA files.
prep_giant.py	Prepare metadata file for GIAnT files.

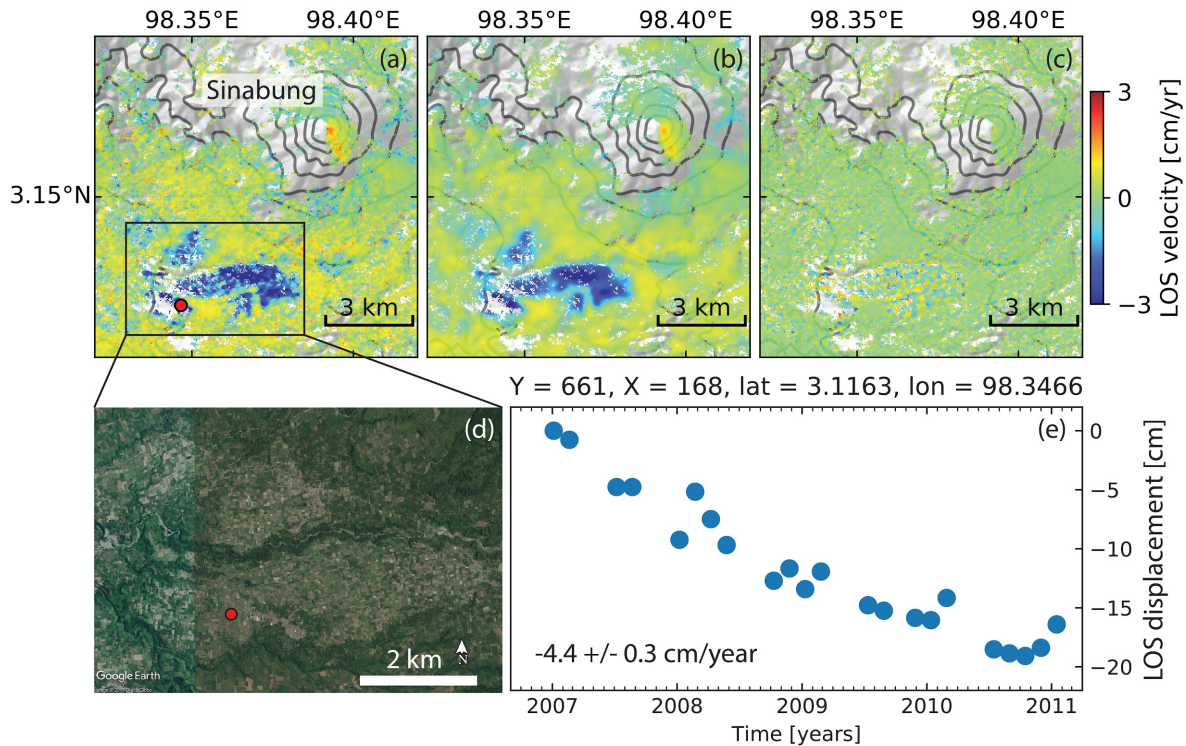|	
`prep_isce.py`	Prepare metadata file for ISCE files.
`prep_roipac.py`	Prepare metadata file for ROI_PAC files.
`prep_snap.py`	Prepare metadata file for SNAP geocoded products.
`reference_date.py`	Change the reference date of a time-series HDF5 file.
`reference_point.py`	Change the reference pixel of an input file.
`remove_ramp.h5`	Remove phase ramps for input file.
`save_gbis.py`	Save input files in GBIS *.mat file format.
`save_gmt.py`	Save input file in GMT *.grd file format.
`save_hdfeos5.py`	Save input time-series into HDF-EOS5 format.
`save_kmz.py`	Save input file into Google Earth raster image.
`save_kmz_timeseries.h5`	Save input file into Google Earth points, interactively.
`save_roipac.py`	Save input file into ROI_PAC style binary file format.
`select_network.py`	Select interferometric pairs from input baseline file.
`smallbaselineApp.py`	Routine time series analysis for small baseline InSAR stack.
`spatial_average.py`	Calculate average in space domain.
`spatial_filter.py`	Spatial filtering of input file.
`subset.py`	Generate a subset of (crop) input file.
`temporal_average.py`	Calculate average in time domain.
`temporal_derivative.py`	Calculate the temporal derivative of displacement time-series.
`temporal_filter.py`	Smooth time-series in time domain with a moving Gaussian window

`timeseries2velocity.py`	Invert time-series for the average velocity.
`timeseries_rms.py`	Calculate the root mean square for each acquisition of the input time-series file.
`tropo_phase_elevation.py`	Correct stratified tropospheric delay based on the empirical phase/elevation ratio method.
`tropo_pyaps.py`	Correct tropospheric delay estimated from global atmospheric model (GAM) using PyAPS software (Jolivet et al., 2011; 2014).
`tsview.py`	Interactive time-series viewer.
`unwrap_error_bridging.py`	Correct phase-unwrapping errors with bridging method.
`unwrap_error_ phase_closure.py`	Correct phase-unwrapping errors with the phase closure method.
`view.py`	2D matrix viewer.

276

S4.2 Filters tools in space and time domain

278 The software supports filters in space or time domain built on skimage (van der Walt et al.,

279 2014). Although filtering is not applied in the routine workflow, it is a useful tool to examine the

280 deformation signal because it allows removing undesired signals. Fig. S15 shows an example,

281 where we use spatial Gaussian filtering to confirm a patchy, rapid subsidence signal.
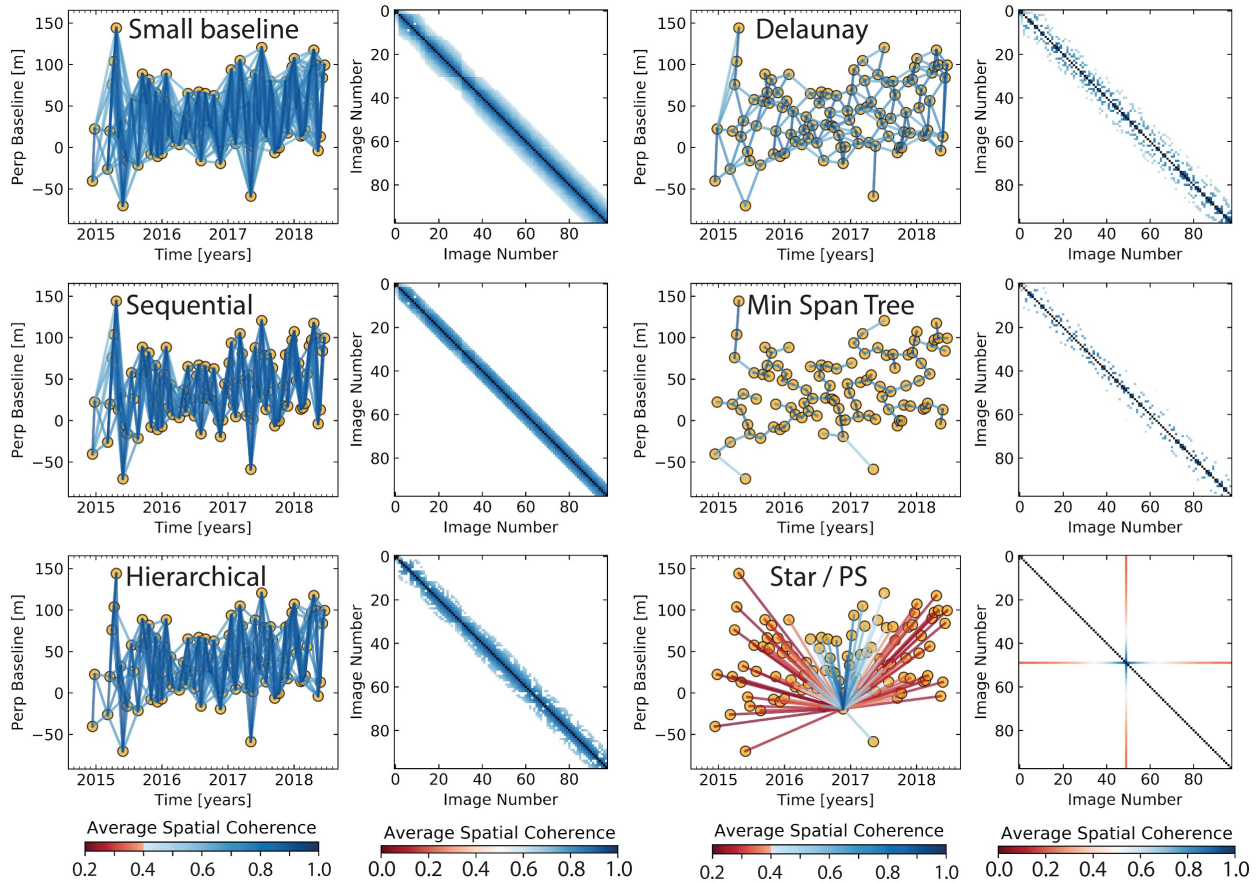
282

283

Figure S15. *Illustration of the spatial filtering. The LOS velocity from ALOS-1 ascending track*

495 acquired over Sinabung volcano, Indonesia during January 2007 to January 2011 is used.

(a) Original velocity in LOS direction, (b and c) velocities after lowpass and highpass Gaussian

filtering with the standard deviation of 3.0. (a) is the sum of (b) and (c). The lowpass filtering

eliminated the very short spatial wavelength features, thus, highlighted the relatively long spatial

wavelength deformation features, such as the volcanic deformation along the Sinabung's

southeast flank and an undocumented patchy, rapid subsidence area (up to -5.6 cm/year) is

found ~6 km to the southwest of the volcano. The spatial pattern of the subsidence signal

correlates well with the agricultural land use, suggesting that subsidence is caused by

groundwater extraction (Chaussard et al., 2013). Reference point is a pixel at [E98.4999°,

N3.1069°] outside of this figure. (d) Google Earth image for the marked rectangle area. (e) LOS

displacement time-series for pixel marked by red circle in (a) at [E98.3466°, N3.1163°].

S4.3 Interferometric pairs selection

The software supports several interferometric pairs selection methods to facilitate the pre-processing, such as small baseline, sequential, hierarchical, Delaunay triangulation, minimum spanning tree and star/PS-like methods, as shown in Fig. S16.

Figure S16. *Illustration of interferometric pairs selection. The temporal and perpendicular baselines are from Sentinel-1 dataset of section 5. For each method, network configuration on the left and the corresponding coherence matrix on the right. The spatial coherence calculation is described in section S3.1 with decorrelation rate of 200 days and long-term coherence of 0.2. The small baseline method selects interferograms with temporal and perpendicular baseline within the predefined thresholds (120 days and 200 m; Berardino et al., 2002). The sequential method selects for each acquisition with a predefined number (5) of its nearest neighbors back in*

308 *time (Reeves and Zhao, 1999). The hierarchical method specifies a predefined list of temporal*

309 *and perpendicular baselines as [6 days, 300 m; 12 days, 200 m; 48 days, 100 m; 96 days, 50 m],*

310 *each pair of temporal and perpendicular thresholds selects interferograms the same as small*

311 *baseline method (Zhao, 2017). The Delaunay triangulation method generates triangulations in*

312 *the temporal and perpendicular baseline domain and selects interferograms within the*

313 *predefined maximum temporal and perpendicular baseline (120 days and 200 m; Pepe and*

314 *Lanari, 2006). The minimum spanning tree method calculates a spatial coherence value based*

315 *on its simple relationship with the temporal and perpendicular baseline and selects N-1*

316 *interferograms that maximizes the total coherence (Perissin and Wang, 2012). The star-like*

317 *method selects network of N-1 interferograms with single common reference acquisition (usually*

318 *in the center of the time period; Ferretti et al., 2001).*

319

S4.4 Local oscillator drift correction for Envisat

321 Data from Envisat's Advanced Synthetic Aperture Radar instrument include a phase ramp in

322 range direction due to timing errors. We correct this local oscillator drift using the empirical

323 model given by Marinkovic and Larsen (2013).

324

$$\phi_{LOD}^i = \frac{-4\pi}{\lambda} 3.87 \times 10^{-7} r(t_i - t_1) \tag{S16}$$

326

327 where $(t_i - t_1)$ represents the time difference in years between SAR acquisition t_i and t_1 (see

328 also Fattahi and Amelung, 2014). Since this model is independent of the InSAR phase

329 measurement, this correction should be applied before any InSAR data-dependent phase

330 corrections.

Supplemental references

Chaussard, E., F. Amelung, H. Abidin, and S.-H. Hong (2013), Sinking cities in Indonesia: ALOS PALSAR detects rapid subsidence due to groundwater and gas extraction, *Remote Sensing of Environment*, 128(0), 150-161, doi:10.1016/j.rse.2012.10.015.

Du, Y., L. Zhang, G. Feng, Z. Lu, and Q. Sun (2017), On the Accuracy of Topographic Residuals Retrieved by MTInSAR, *IEEE Transactions on Geoscience and Remote Sensing,* 55(2), 1053-1065, doi:10.1109/TGRS.2016.2618942.

Marinkovic, P., and Y. Larsen (2013), Consequences of long-term ASAR local oscillator frequency decay - An empirical study of 10 years of data, paper presented at *Proceedings of the Living Planet Symposium (abstract)*, European Space Agency, Edinburgh, U. K.

Reeves, S. J., and Z. Zhao (1999), Sequential algorithms for observation selection, *IEEE Transactions on Signal Processing*, 47(1), 123-132, doi:10.1109/78.738245.

van der Walt, S., J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu (2014), scikit-image: image processing in Python, *PeerJ*, 2, e453, doi:10.7717/peerj.453.

Zhao, W. (2017), Small Deformation Detected from InSAR Time-Series and Their Applications in Geophysics, Dissertation thesis, 153 pp, University of Miami, Miami, FL.