Co-registration Toolbox for Sentinel-1 (CtSent)

CtSentv1.0 Users' Manual

CtSent v1.0 described here is a MATLAB-based software developed by a team of researchers, and here we are happy to provide a copy to non-commercial researchers. The code is well documented and has been tested in different regions with different coherence conditions. However, it has not been through the same quality control procedure that is expected for a commercial software. For using this software, some computing environments need to configured (such as GUROBI software for phase unwrapping), for which guidance is provided as below. If you have other concerns or are interested in collaborative researches, please contact us at the following E-mail: jspcmazhangfeng@hhu.edu.cn.

CtSent Designers:
Zhang-Feng Ma, Hohai University
Mi Jiang, Sun Yat-Sen University
Yi Zhao, Hohai University
Cheng Zhou, Hohai University
& ...

Required Environments:
A laptop with at least 8GB ram
Windows 10
Matlab version >=2018b
Gurobi software (https://www.gurobi.com/downloads/)
You need apply an academic license of Gurobi first.
(https://www.gurobi.com/downloads/end-user-license-agreement-academic
)

Tips: This is just a trial version of CtSent v1.0. It only supports geometrical co-registration, Enhanced Spectral Diversity, interferometry and deburst and phase unwrapping.
Some basic functions can be obtained from StaMPS software.
(https://homepages.see.leeds.ac.uk/~earahoo/stamps/).
If you need a full version which supports Near-Real-Time processing and time series analysis, please contact with Zhang-Feng Ma.
If you used CtSent software, please cite given literatures in your publications:
[1] Ma, Z.F et.al (2019), Minimum Spanning Tree Co-registration Approach for Time-Series Sentinel-1 TOPS Data, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(8), 3004-3013. doi: 10.1109/JSTARS.2019.2920717.
[2] Ma, Z.F et.al (2020), A Sequential Approach for Sentinel-1 TOPS Time-Series Co-Registration Over Low Coherence Scenarios, I*EEE*

Transactions on Geoscience and Remote Sensing, 1-9. doi: 10.1109/TGRS.2020.3009996.
[3] Ma, Z.F. et.al (2021), Time Series Phase Unwrapping Based on Graph Theory and Compressed Sensing, *IEEE Transactions on Geoscience and Remote Sensing*, 1-12. doi: 10.1109/TGRS.2021.3066784.

Detailed User Guidance:

Step1 Preparation

All steps of this software are run in MATLAB. Before you begin, you need to prepare some data. They are the unzipped SLC image, DEM (tiff format), precise orbits.

Tips: SLCs, DEM and orbits should be placed in their own file folders.

Step2 Initial parameters setting

Initial parameters are set in ctSent_parms_default.m. Usually, you only need to set the following parameters:

set a path of processing directory, used to store work files during data processing.

parms.workDir = '…\workDir';

set the rectangle of your area of interest [min lon, max lon, min lat, max lat]

parms.rectangle = [39.65 39.9 21.9 22.32];

set the path of SLCs, orbits and DEM.

parms.SLCPATH = '…\asc' ;

parms.ORBITPATH = '…\orbit' ;

parms.DEMPATH = '…\dem\dem.tif';

Then run >> ctSent_parms_default.m.

Tips: This step is just to initialize all parameters. Your defined area of interest is better to be small.

Step3 Data preprocessing

run >> ctSent_extractSLC.m

This step will extract annotation files, apply orbit file and stitch bursts in different swaths. Then the files of each swath and burst will generate in your set workDir.

Tips: This step is to extract bursts from SLCs.

Step4 Geometrical Co-registration

run >> ctSent_geocoregSLC.m

Tips: This step is to co-register all SLCs using Radar geometry and Zero-Doppler.

Step5 Enhanced Spectral Diversity

run >> ctSent_EnhancedSpectralDiversity.m

Tips: This step is to mitigate azimuth mis-registration.

Step6 Interferometry formation

```
run >> ctSent_Interferometry.m
```

Tips: This step is to geocode all SLCs and remove flat-earth phase from all processed SLCs.

Step7 Debrust

```
run >> ctSent_deburst.m
```

Tips: This step is to merge all needed bursts.

Step8 Reading processed results

```
run >> [slcstack,lat,lon,inc,hgt,bperp_mat,ac_date] = ctSent_readDeb;
```

slcstack: the processed SLC stack
lat,lon: latitude and longitude
inc: incidence angle
hgt: elevation
bperp_mat: perpendicular baseline stack
ac_date: acquired time of SLC stack

Step9 Generate interferogram and estimate its coherence, then multilook the generated interferogram

```
run >> int_phase = slcstack(:,:,1).*conj(slcstack(:,:,2));
```

You can compute the interferometric phase of any two images by set the third parm of slcstack. Yon can also compute Coherence Estimation by

```
run >> Coh = BoxCoherence(slcstack(:,:,1),slcstack(:,:,2));
```

Then multilooking by

```
run >> multilook_phase = cpxmultilook(int_phase,range looks,azimuth looks);
    >> fCoh = cpxmultilook(Coh,range looks,azimuth looks);
    >> flat = cpxmultilook(lat,range looks,azimuth looks);
    >> flon = cpxmultilook(lon,range looks,azimuth looks);
   %range looks=4, azimuth looks=1;
```

Step10 Filtering

```
run >> ph_out = goldstein_filt(multilook_phase,,n_win,alpha,n_pad);
```

Step11 Phase unwrapping

Usually, in order to reduce the unwrapping error and data redundancy, you can first select the high coherence pixels.

```
run >> ph = angle(ph_out((fCoh1>=0.3)));
   >> lonlat = [flon((fCoh>=0.3)),flat((fCoh>=0.3))];
   >> [idx,idy] = find(((fCoh>=0.3)));
   >> xy = [idx*3.5,idy*14];
```

Then phase unwrapping,

```
run >> CtSent_grid_wrapped('...\ workDir',ph,xy,pix_size,prefilt_win);
% pix_size=80, prefilt_win=40;
   >> CtSent_PhaseUnwrappingGrid('...\ workDir',flag);
   % flag=0;
   >> [ph_uw,msd] = CtSent_InterpPUgrid('...\ workDir',ph);
```

You can plot the unwrapped results by

```
run >> scatter(lonlat(:,1),lonlat(:,2),2,ph_uw(:,1));
```

Words in the end: If you have any issue when you used CtSent, please give us feedbacks.

It is indeed a difficult time for many people and countries. May the world be filled with health, peace and love.

Best.

--All designers.