HOW TO
INSTALL AND PROCESS GMTSAR
(or GMT5SAR)

OPEN SOURCE SOFTWARE TO PROCESS INTERFEROMETRY SYNTHETIC APERTURE RADAR IMAGES

Linux Operating System, Ubuntu 14.04

Noorlaila Hayati Isya

November, 2015
revised: Oct, 2017

Institut für Geodäsie und Photogrammetry
Technische Universität Braunschweig, Germany

A. Introduction

GMTSAR is an open source software developed by Scripps Institution of Oceanography, San Diego State University (David Sandwell, etc). ==Thank you for David Sandwell and his friends to create this software==. The code is written by C and will compile if the computer is already installed GMT (Generic Mapping Tools) and NETCDF. It is running on UNIX operating system such as Linux and Mac OS X but it also could run on Windows system using third supporting application. GMTSAR has 3 main processing:

a. A preprocess based on each satellite SAR data source (ALOS PALSAR1-2, ERS1-2, Envisat, Radarsat.2, TerraSAR X, Sentinel-1 and Cosmo Skymed). This step converting native format and orbital information to generic format. Please pay attention that there are many format published from different company, so user must know how to process the program (pre_process scripts) using the right format.
b. An InSAR process to focus and align stacks of images, map topography into phase and form the complex interferogram.
c. A postprocess that mostly done by GMT. It is include of process filter interferogram, phase and coherence product, converting to line of sight displacement and geocoding.

This tutorial will show you how to install and process SAR Images using GMT5SAR. GMT5SAR is a an update software from previous one, GMTSAR ver 4 using old GMT4. Installation on GMT5SAR is quite easier than GMTSAR ver 4. However, we should firstly install GMT5 on our computer.

Note:
if I give this symbol : ===> . It means all of command text on terminal is on one line or one row, type the command without push “enter” button.

B. How to Install GMT5

1. Open terminal in Ubuntu (terminal is a window with purple colour)
2. Install Ghostscript, Subversion, CMake (>=2.8.5), netCDF (>=4.0, *netCDF-4/HDF5 support mandatory*) and GDAL. Just type this command on terminal:

   ===>

```
sudo apt-get install subversion build-essential cmake libcurl4-
gnutls-dev libnetcdf-dev libgdal1-dev libfftw3-dev libpcre3-dev
liblapack-dev
```

3. Optionally, install also PCRE, GDAL, and single-precision FFTW (add Texlive and Sphinx if you wish to build the documentation, and GraphicsMagick if you wish to enable running the tests). I recommend to install GDAL and Graphicsmagick because postprocessing InSAR will often use these softwares.

   Type this command on terminal to install GraphicMagick

```
sudo apt-get install GraphicsMagick
```

   If you want to also install the documentation of GMT, you also have to install texlive. Type on terminal:

```
sudo apt-get install texlive texlive-latex-extra
sudo apt-get install python-sphinx
```

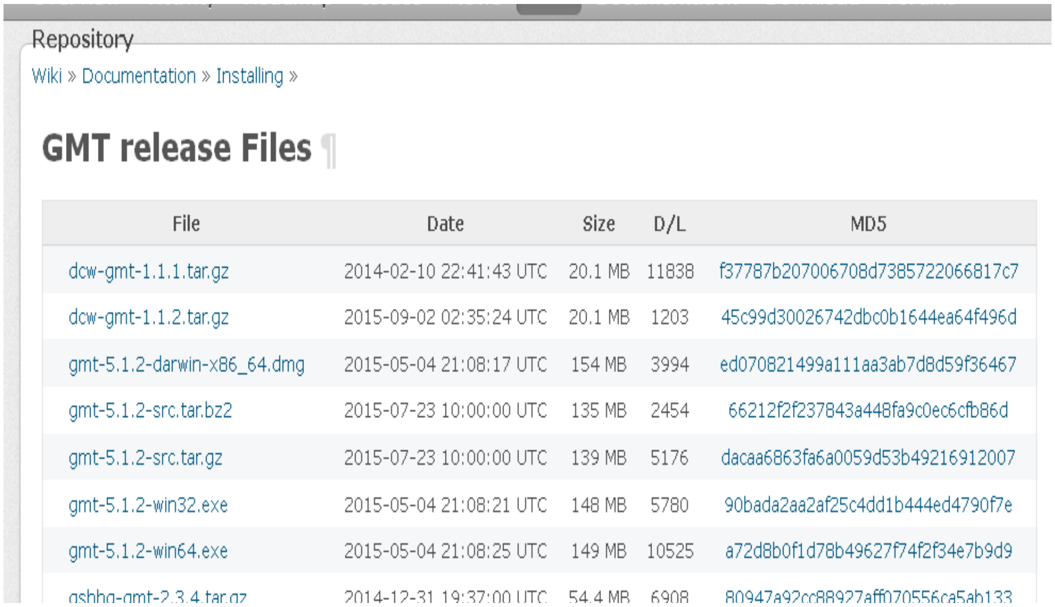
4. This the main step to install GMT. There are many ways to install either online or offline. The easy one is using online method. Firstly, get GMT from source:trunk

```
svn checkout svn://gmtserver.soest.hawaii.edu/gmt5/trunk gmt5-dev
```

   or new version

```
svn checkout svn://gmtserver.soest.hawaii.edu/gmt/trunk gmt-dev
```

5. Get GSHHG (coastlines, rivers, and political boundaries; filename: gshhg-gmt-x.x.x.tar.gz) from the download page or ftp://ftp.soest.hawaii.edu/gshhg and extract the files.
6. Get DCW (country polygons; filename: dcw-gmt-x.x.x.tar.gz) from the download page or ftp://ftp.soest.hawaii.edu/dcw and extract the files.


Repository

Wiki » Documentation » Installing »

GMT release Files ¶

File	Date	Size	D/L	MD5
dcw-gmt-1.1.1.tar.gz	2014-02-10 22:41:43 UTC	20.1 MB	11838	f37787b207006708d7385722066817c7
dcw-gmt-1.1.2.tar.gz	2015-09-02 02:35:24 UTC	20.1 MB	1203	45c99d30026742dbc0b1644ea64f496d
gmt-5.1.2-darwin-x86_64.dmg	2015-05-04 21:08:17 UTC	154 MB	3994	ed070821499a111aa3ab7d8d59f36467
gmt-5.1.2-src.tar.bz2	2015-07-23 10:00:00 UTC	135 MB	2454	66212f2f237843a448fa9c0ec6cfb86d
gmt-5.1.2-src.tar.gz	2015-07-23 10:00:00 UTC	139 MB	5176	dacaa6863fa6a0059d53b49216912007
gmt-5.1.2-win32.exe	2015-05-04 21:08:21 UTC	148 MB	5780	90bada2aa2af25c4dd1b444ed4790f7e
gmt-5.1.2-win64.exe	2015-05-04 21:08:25 UTC	149 MB	10525	a72d8b0f1d78b49627f74f2f34e7b9d9
gshhg-gmt-2.3.4.tar.gz	2014-12-31 19:37:00 UTC	54.4 MB	6908	80947a92cc88927aff070556ca5ab133

Picture 1. GMT download page

7. In the source tree copy cmake/ConfigUserTemplate.cmake to **cmake/ConfigUser.cmake** and edit the file according to your demands.

   I recommend that CMAKE_INSTALL_PREFIX is using default path, so when we install GMT5SAR, we don't have to be confused where is the location path of GMT. So leave the CMAKE_INSTALL_PREFIX without adding a new path. This is an example:

```
set (CMAKE_INSTALL_PREFIX <usr/local>)  #or let it default
set (GSHHG_ROOT <path to gshhg>)
set (DCW_ROOT <path to dcw>)
set (FLOCK TRUE)
```

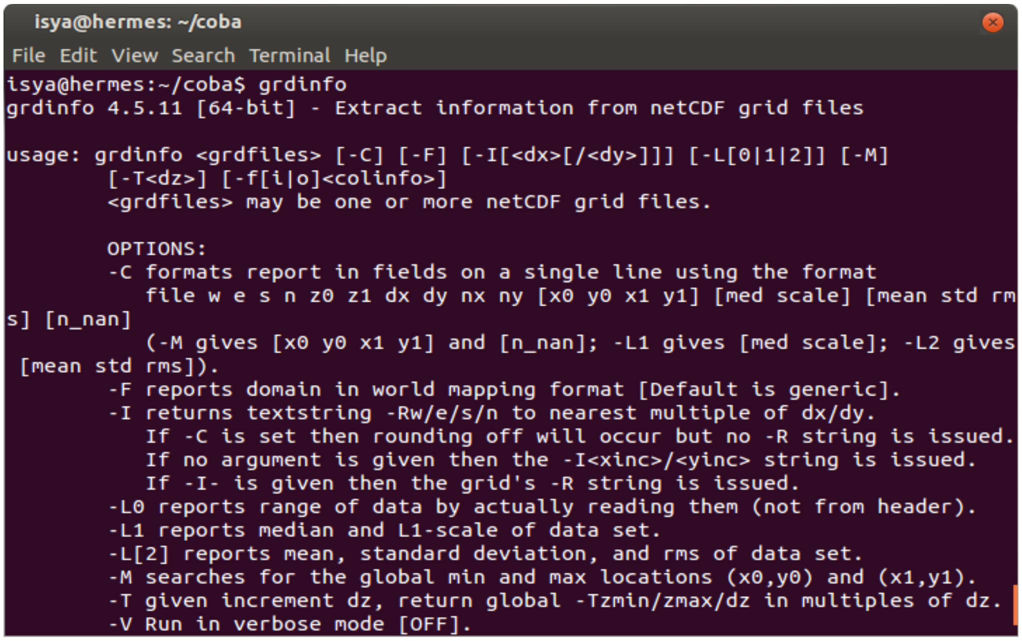
8. Build and install GMT:

```
cd <path to>/gmt-dev #or gmt5-dev
mkdir build
cd build
cmake ..
make
sudo make install  #use sudo if GMT is installed on root dir
```

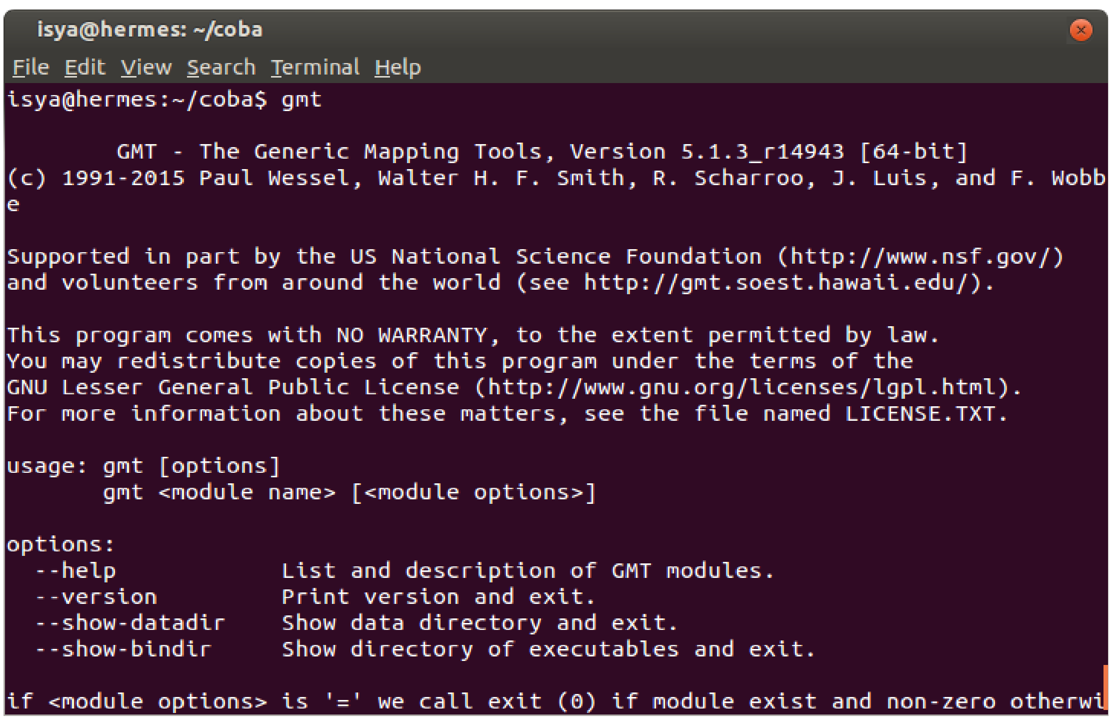
   if there is an error due to CURL library when you run “ cmake .. ”, please install additional CURL library, on terminal type

```
$ sudo apt-get install libcurl4-openssl-dev
```

9. Now, its' done! To test that you success to install GMT5. Type on your terminal **grdinfo** or **gmt.**



Picture 2. grdinfo on terminal



Picture 3.a. gmt version 5.1.3 on terminal

C. How to Install GMTSAR

1. When you already can use GMT properly, let we prepare GMTSAR installation, firstly we have to install the extra libraries and gfortran & g++ as compilers. Type on terminal this command:

===>

```
sudo apt-get install csh autoconf libtiff5-dev libhdf5-dev libboost-all-dev

sudo apt-get install liblapack-dev gfortran g++
```

2. Download (http://topex.ucsd.edu/gmtsar/tar/ORBITS.tar) and install orbit files. Remember to place in suitable directory. I recommend to use this path (e.g., /usr/local/orbits).

```
sudo -i

cd /usr/local

mkdir orbits

cd orbits

tar xvf ~/Downloads/ORBITS.tar #need full path to ORBITS.tar
```

3. Install GMTSAR trunk via subversion in a suitable directory :

"sudo -i" is a command to access the root directory. Becaue path /usr/local is in a root directory, we have to be an administartor. When type this command, usually, terminal will ask your password user.

```
sudo -i

cd /usr/local

svn checkout svn://gmtserver.soest.hawaii.edu/GMTSAR/branches/5.4 GMTSAR
```

If you are successful to download the GMTSAR, there is a new folder GMTSAR at your " */usr/local* " path.

4. Make and install GMTSAR. We have to change the orbit location because on the previous step we use /usr/local/orbits for the path.

```
cd GMTSAR

autoconf

./configure --with-orbits-dir=/usr/local/orbits

make

make install
```

Nt :

If your boost library is in an unusual place you may need to add the boost comment:

`--with-boost-include` and `--with-boost-lib`

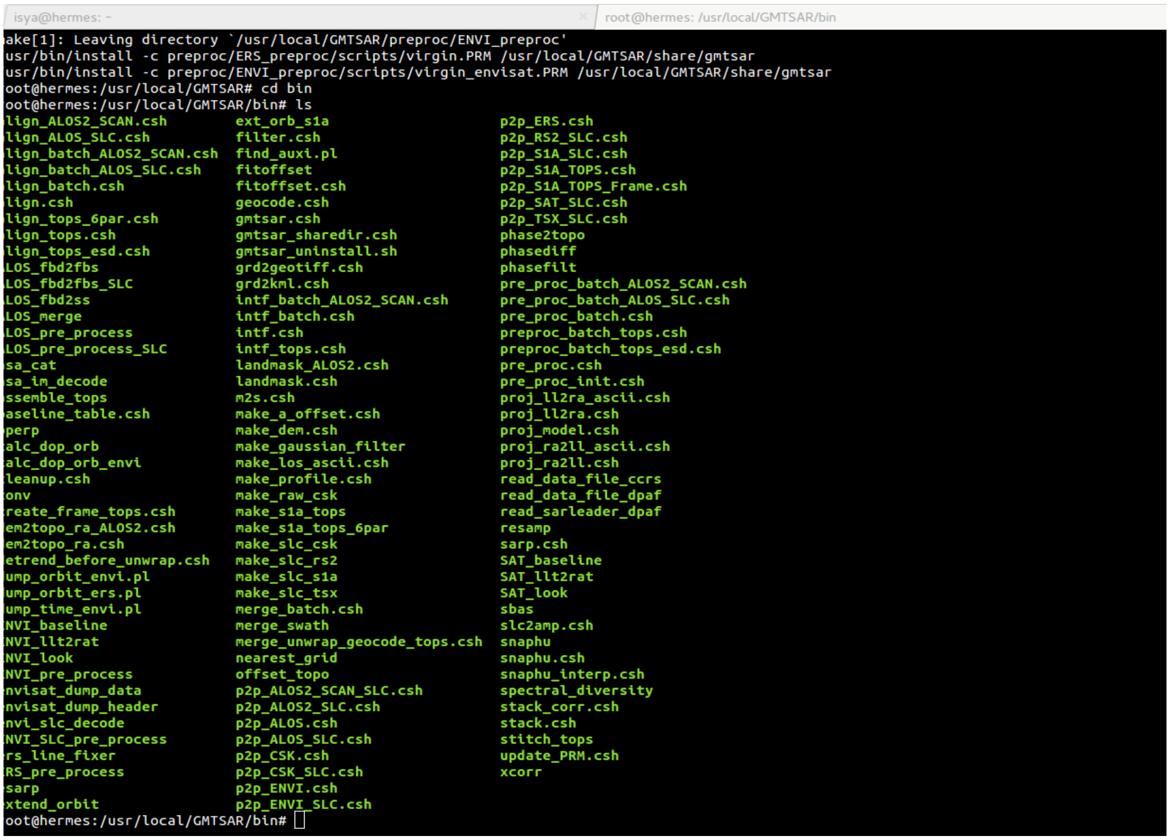
However, I hope you don't change any path on boost library

If the installation is successful, all of the compiler results are placed on **bin** directory. Please check on */usr/local/GMTSAR/bin/* directory whether all of programs (with extension .csh , .pl , etc) are already exist or not.

Type on terminal

```
/usr/local/GMTSAR# cd bin

/usr/local/GMTSAR/bin# ls
```

See picture 3.b. for the list GMTSAR (or GMT5SAR) programs.



Picture 3.b. all of successfully compiled programs on GMT5SAR

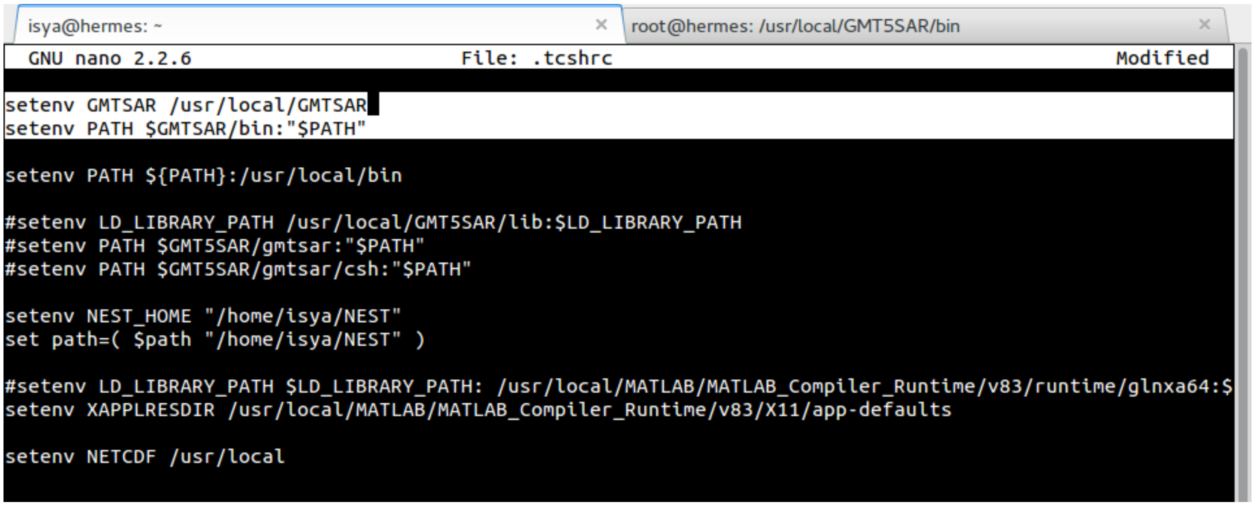
5. Lastly, the last step you have to edit is the csh or tcsh environment. Add the executables to your path (for csh or tcsh). Type on terminal

```
cd ~

pico .tcshrc
```

you will show tcsh environment, add the several following lines on the last text (or empty) of tcsh window

```
setenv GMTSAR /usr/local/GMTSAR

setenv PATH $GMTSAR/bin:"$PATH"
```

to save the setting, ctrl+x on keyboard and type “y” → enter.



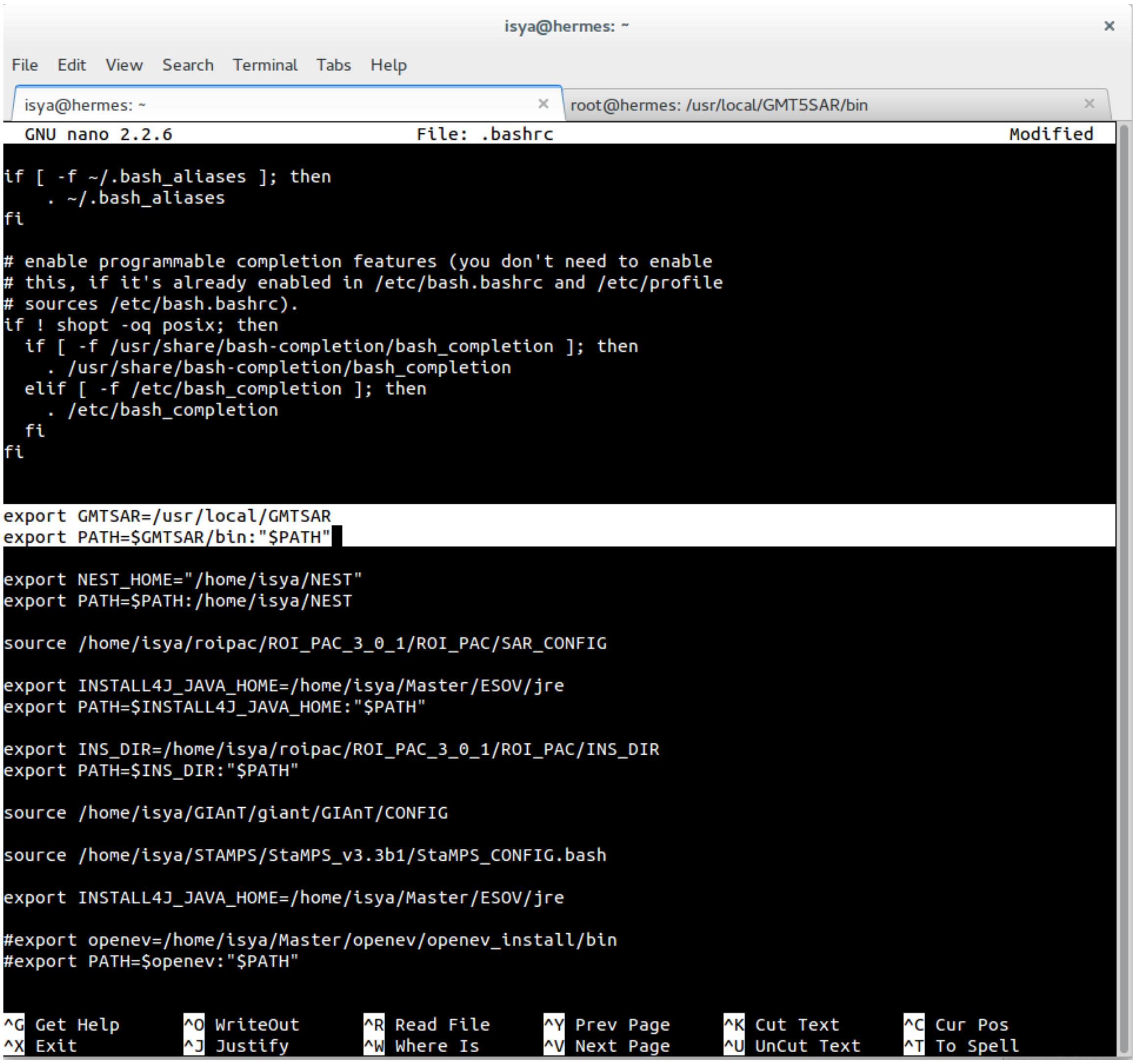
Picture 4. tcshrc environment, just set or type the light lines (on this environment there are so many lines because on my computer I set so many paths from other programs)

OR you could also edit the path on bash environment, type on terminal

```
cd ~

pico .bashrc
```

add some following lines on the last text of bashrc window

```
export GMTSAR=/usr/local/GMTSAR

export PATH=$GMTSAR/bin:"$PATH"
```



Picture 5. bashrc environment, just type the light lines

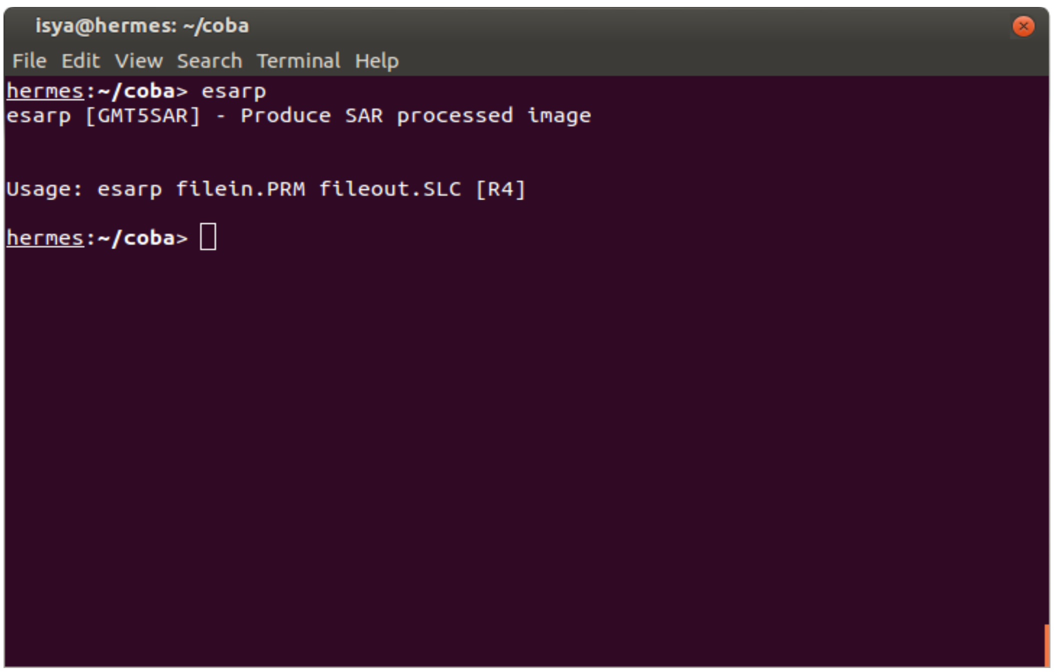
to save the setting, ctrl+x on keyboard and type “y” → enter.

6. Finally, you are already done! To check that GMTSAR success to be installed on your computer. Type on terminal “tcsh” , the symbol will change from ~$ to be ~>

```
tcsh

~>
```

type “esarp”, and terminal will show you some following comments

```
esarp [GMT5SAR] - Produce SAR processed image


Usage: esarp filein.PRM fileout.SLC [R4]
```

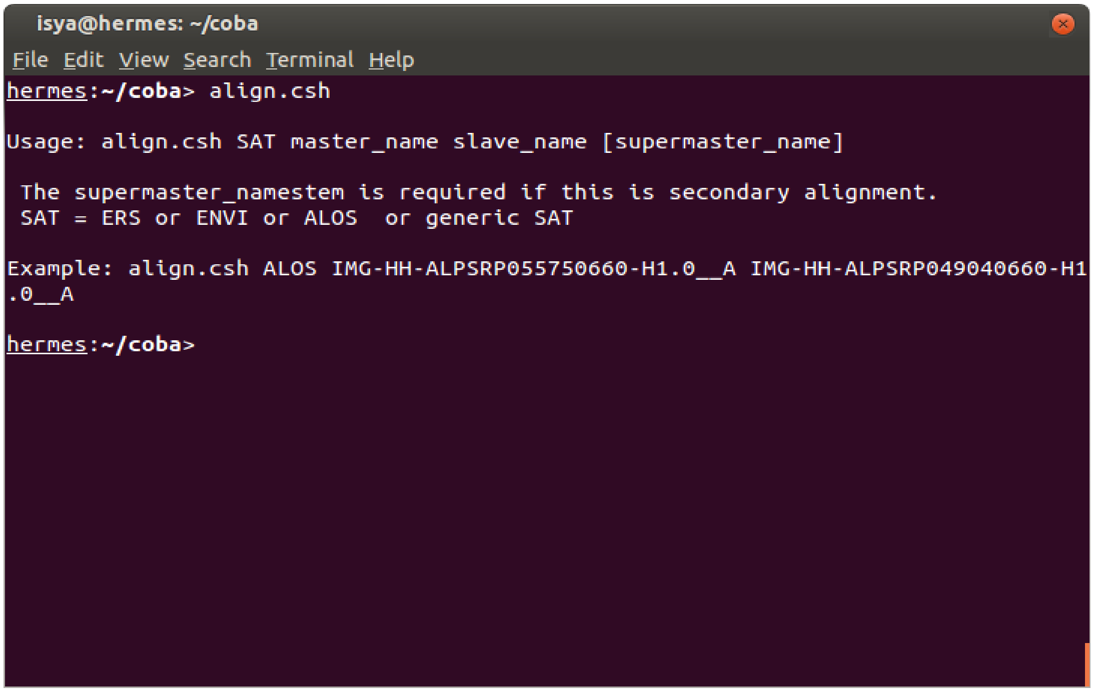


Picture 6. esarp on terminal

type “align.csh”, and terminal will show you some following comments

```
Usage: align.csh SAT master_name slave_name [supermaster_name]

 The supermaster_namestem is required if this is secondary
alignment.
 SAT = ERS or ENVI or ALOS  or generic SAT
```

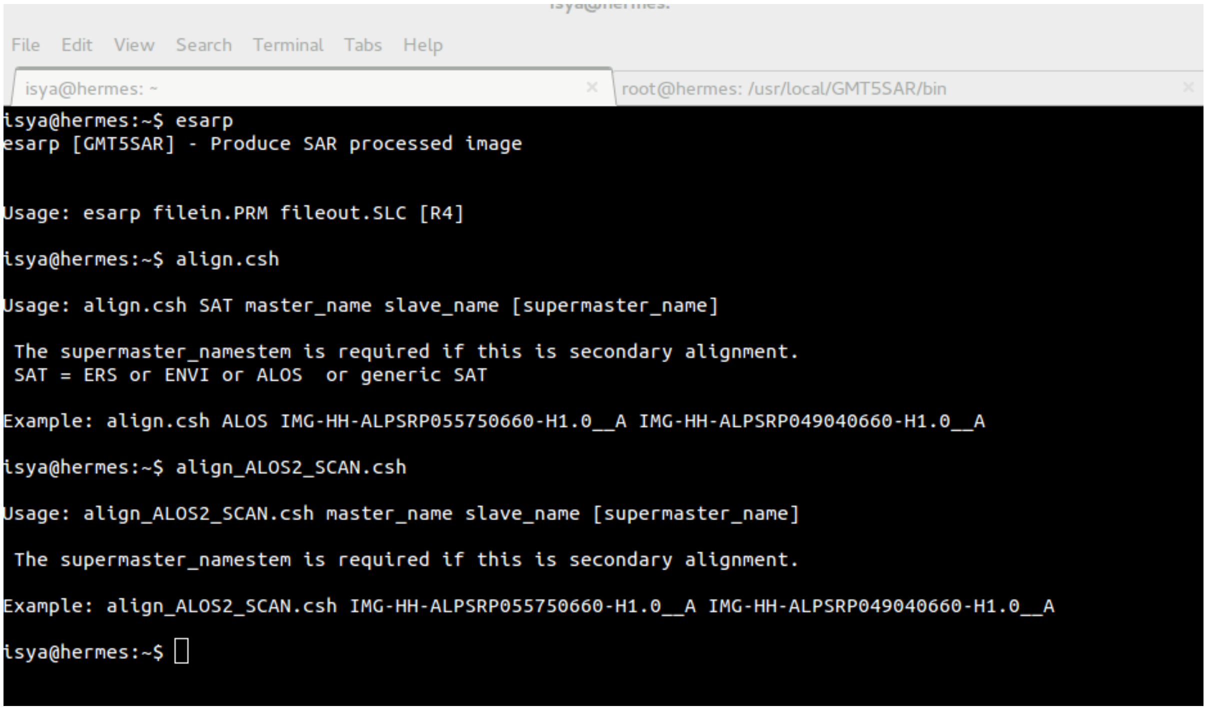
Example: align.csh ALOS IMG-HH-ALPSRP055750660-H1.0__A IMG-HH-ALPSRP049040660-H1.0__A



Picture 7. align.csh on terminal

OR if you use bash environment, exit tcsh environment and type on terminal

```
esarp

align.csh

align_ALOS2_SCAN.csh
```



Picture 8. *esarp*, *align.csh* and *align_ALOS2_SCAN.csh* on bash environment

Congratulations, you already installed GMTSAR :)

Let's work to InSAR processing using an example BAJA - ALOS1 data

D. HOW TO PROCESS InSAR from GMTSAR

You could process InSAR with **one-whole-steps** process or **step by step** process. GMT5SAR provide a script to do a whole process within just one line comment. For beginner, we could run the process using configure.txt (e.g config.alos.txt) as one-whole-steps process. As an example, we will use ALOS PALSAR 1 Data.

1. Download the data on GMTSAR website http://topex.ucsd.edu/gmtsar/downloads/ and choose Download ALOS-1 L1.0 (standard format CEOS)

2. Prepare a directory to make a processing and put the the downloaded data (ALOS_Baja_EQ.tar.gz) on the BAJA directory. For example, I create a new folder: BAJA

```
hermes:~> cd ~
hermes:~> mkdir BAJA
hermes:~> cd BAJA
hermes:~/BAJA>
```

3. Extract ALOS_Baja_EQ.tar.gz. You could extract the tar achive on terminal with

```
hermes:~/BAJA> tar xzf ALOS_Baja_EQ.tar.gz
hermes:~/BAJA> ls
ALOS_Baja_EQ.tar.gz  config.alos.txt  figures  raw  README.txt
           topo
```

There are 3 folders (figures, raw, and topo) and 2 txt files (config.alos.txt and README.txt)

4. All of the RAW that want to be processed must in raw folder. You could see what kind of format data you do need to process. Open the raw data and there are 4 files:

IMG-HH-ALPSRP207600640-H1.0__A
LED-ALPSRP207600640-H1.0__A
===========================>> this is the master images

IMG-HH-ALPSRP227730640-H1.0__A
LED-ALPSRP227730640-H1.0__A
===========================>> this is the slave images

5. We need also an extra DEM (SRTM or ASTER). From an example data, it's already downloaded and located on topo folder. If you want to generate DEM from different location, GMTSAR already make a tool to generate DEM.

Generate DEM→ http://topex.ucsd.edu/gmtsar/demgen/

6. Open config.alos.txt . There is an information what the main process steps that GMTSAR do to create InSAR result.

```
#####################
# processing stage  #
#####################
# 1 - start from preprocess
# 2 - start from focus and align SLC images
# 3 - start from make topo_ra
# 4 - start from make and filter interferograms
# 5 - start from unwrap phase
# 6 - start from geocode
proc_stage = 1
```

proc_stage = 1 means that you will process from 1-Step = preprocess

7. So to start the whole processing, just type a following line on terminal

```
p2p_ALOS.csh (master file) (slave file) config.alos.txt
```

===>

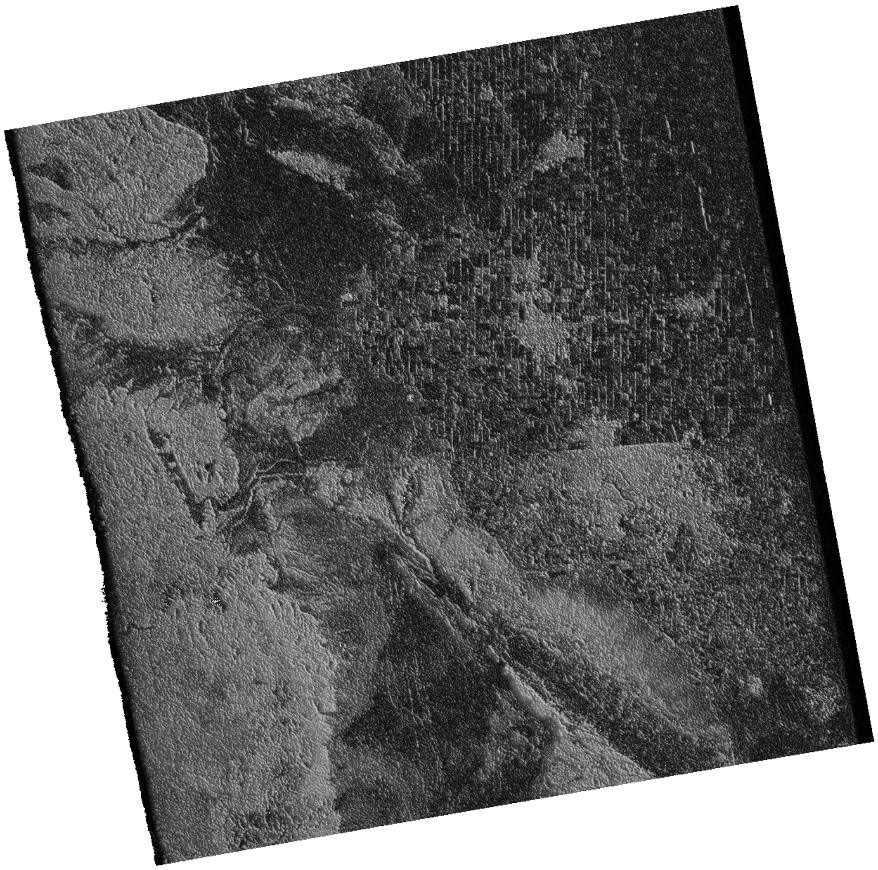
```
p2p_ALOS.csh IMG-HH-ALPSRP207600640-H1.0__A IMG-HH-ALPSRP227730640-H1.0__A config.alos.txt
```

8. The processing will take a long time depending on your computer performance. When the process is done, you will see there is new folders : SLC and intf

9. To see the result, open intf folder and a folder named based on master and slave date has been created. Open the folder. Generally , there are several files which are:

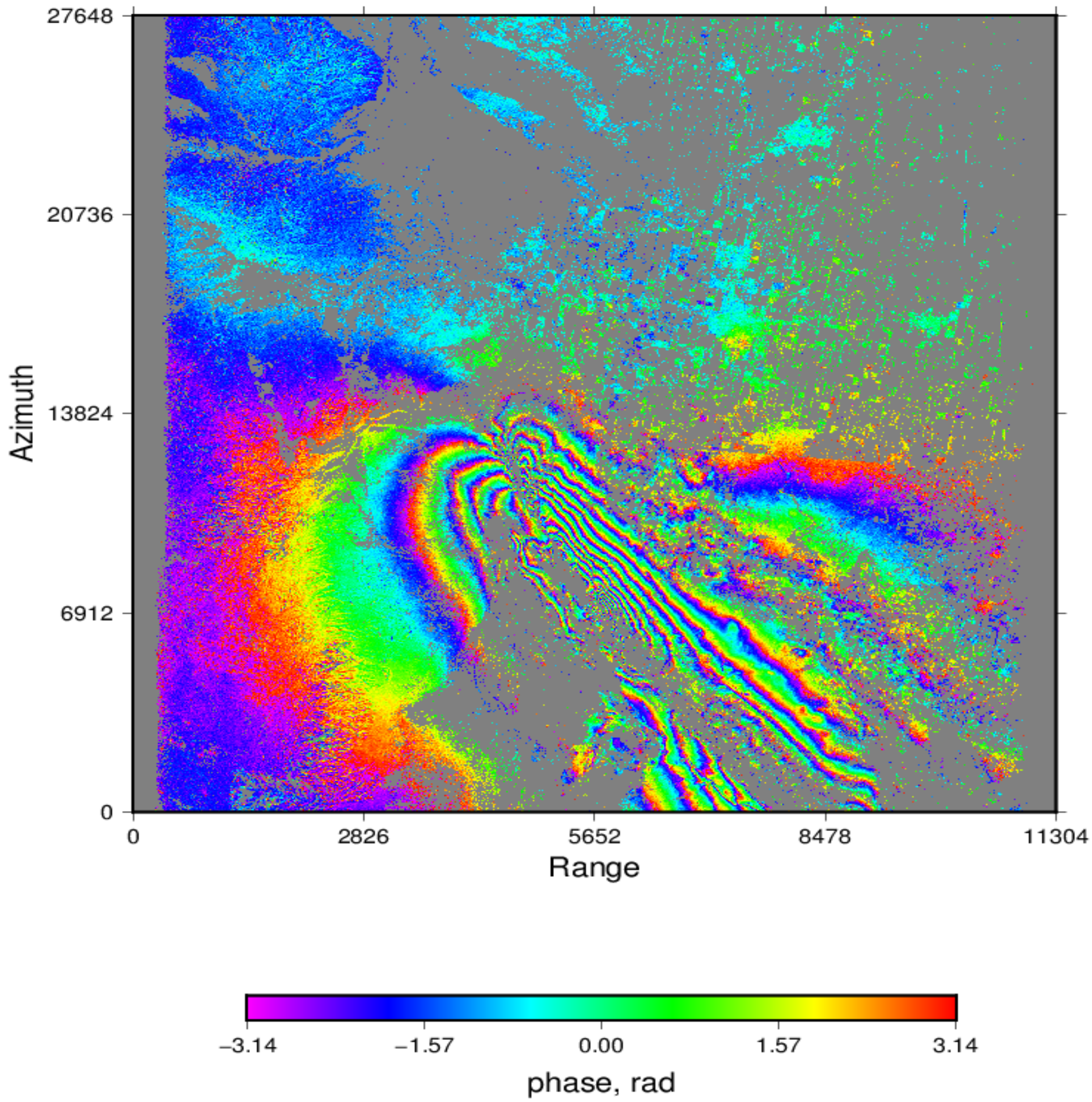
a. amp
b. corr
c. phase
d. phasefilt
e. unwrap
f. mask
g. phasefilt_mask
h. unwrap_mask
I. los

10. For analyzing purpose, you could start to open corr_ll.png or corr_ll.kml (to see on google earth), phase_mask_ll, unwrap_mask_ll and los_ll.



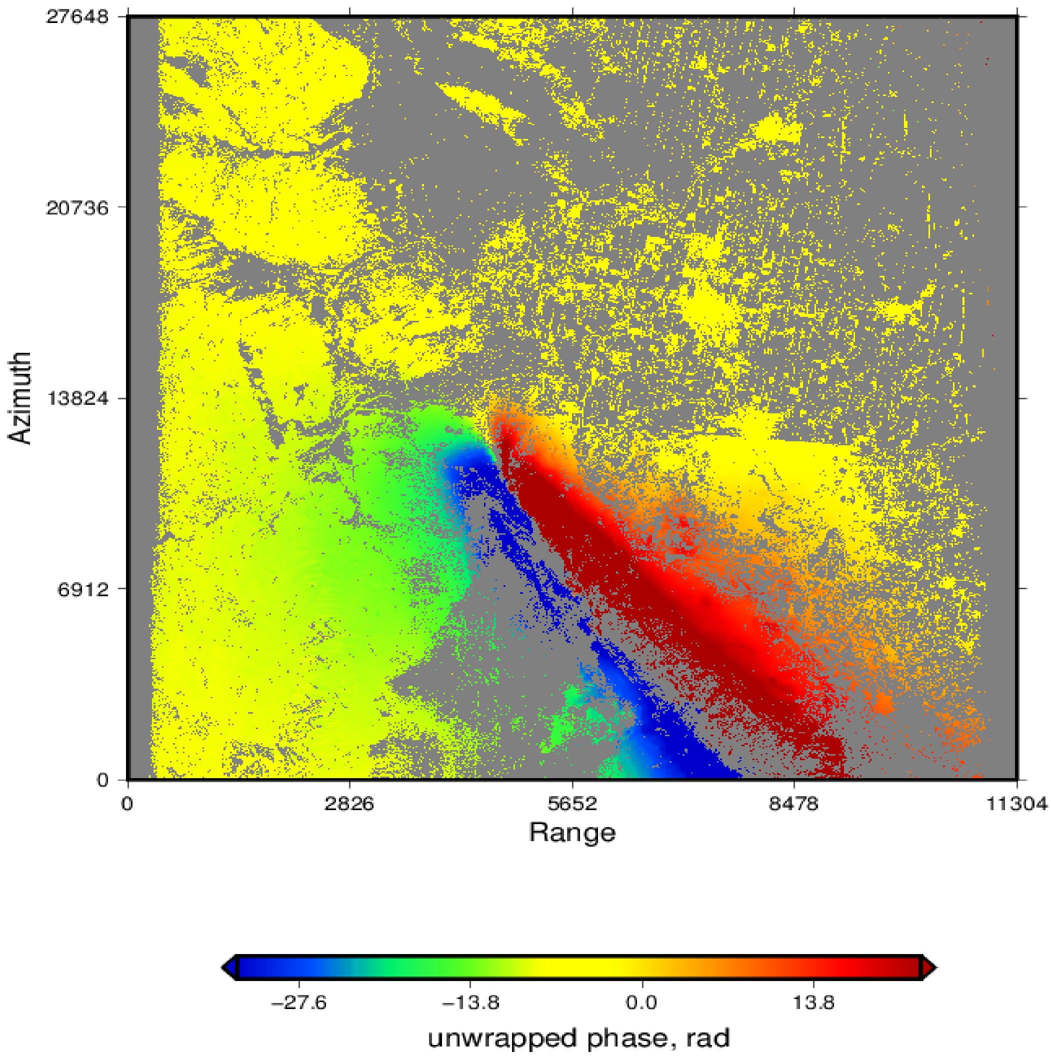
Picture 8. corr_ll.png in BAJA

*corr_ll = to see the correlation between master and slave, the range is 0-1 value. The better coherence between master and slave, the brighter the image (the value is close to 1).



Picture 9. interferometry result in BAJA

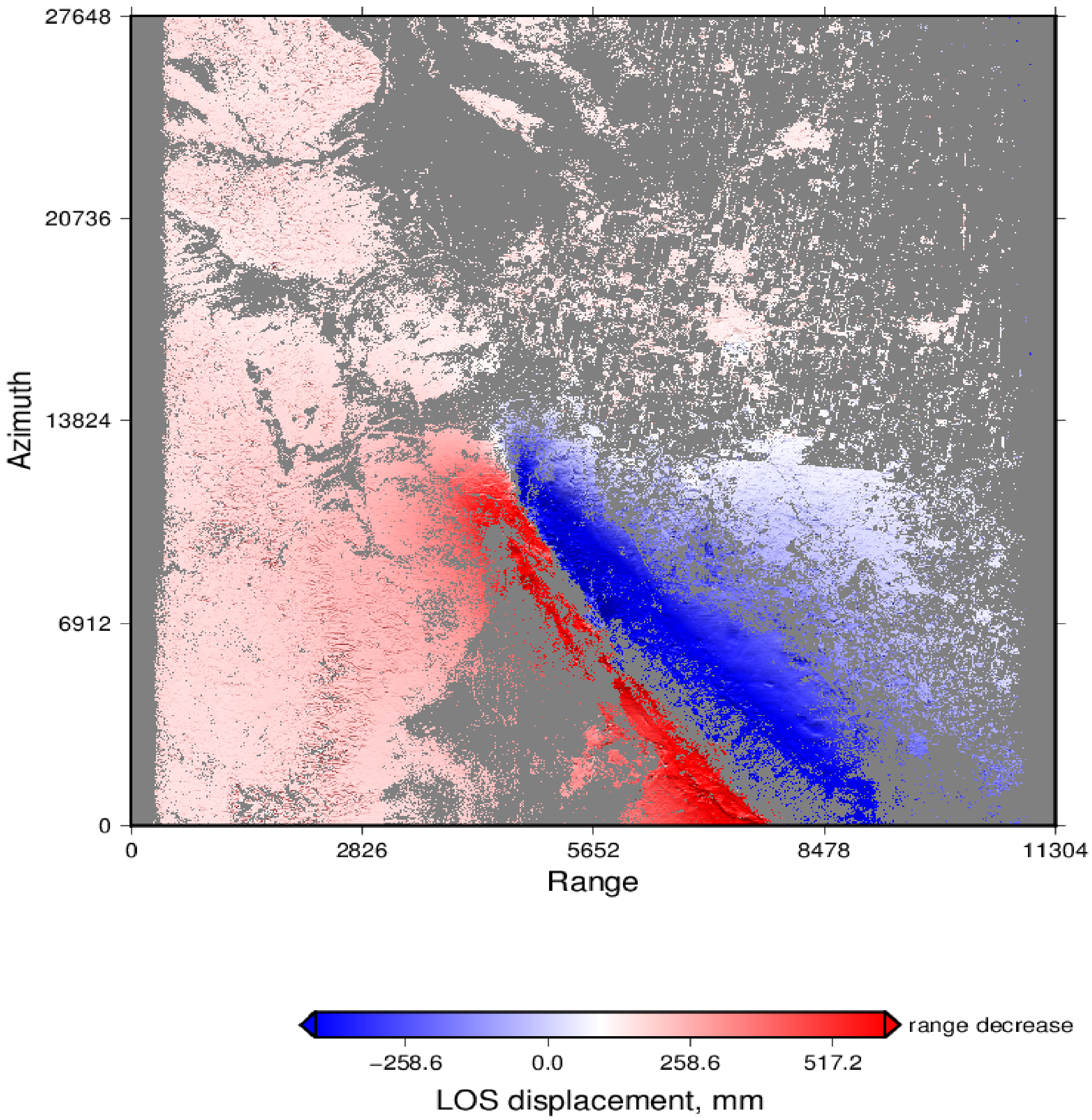
*phase_mask_ll = the result of differential interferometry, which showed the different condition surface between master and slave SAR images



Picture 10. unwrapped phase in BAJA

*unwrap_mask_ll = the result of interferometry which already unwrapped using SNAPHU. However the unit of measurement is still in radian unit.

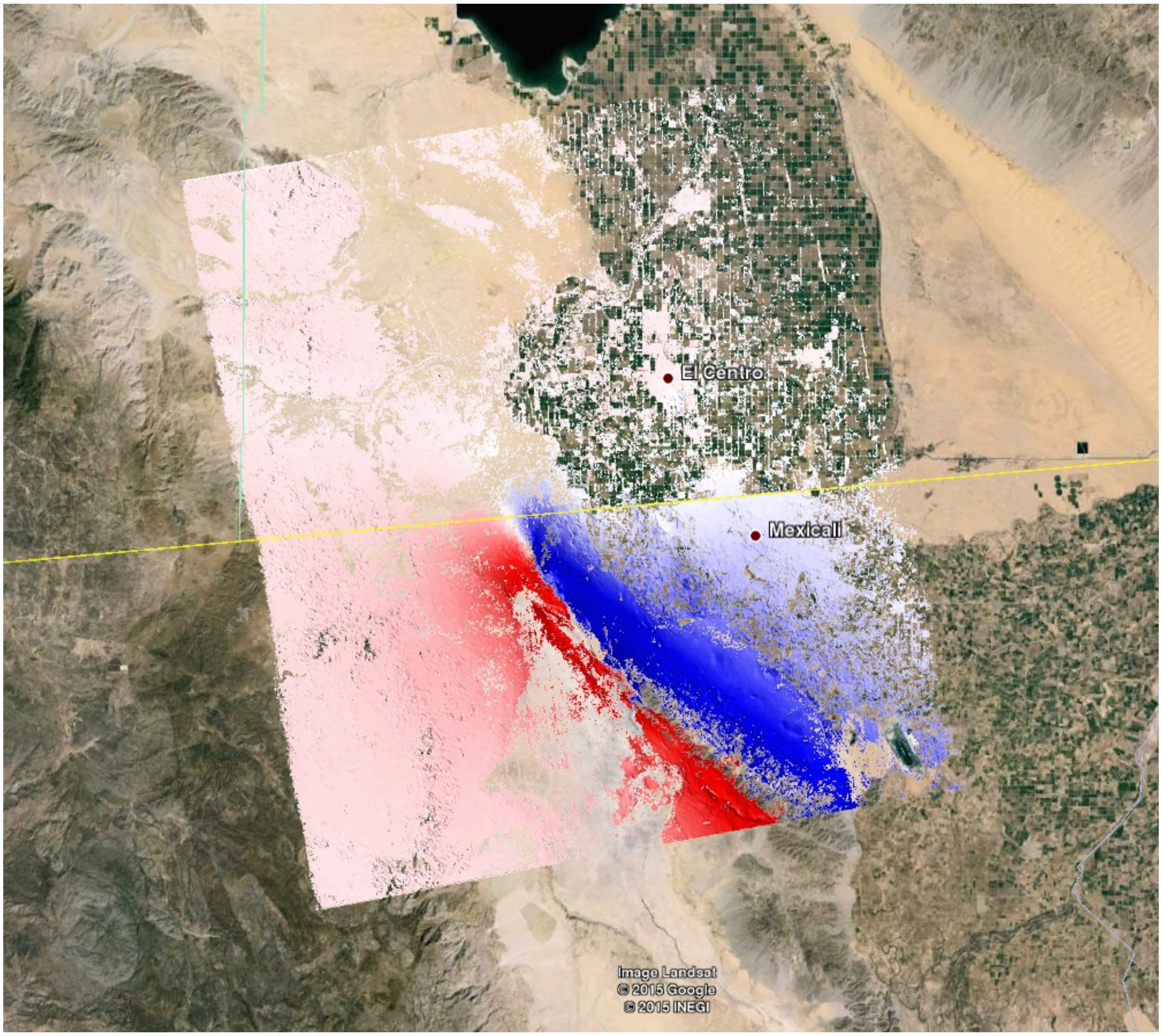
Unwrapped phase is a deformation measurement from satellite viewed. Therefore, the (-) negative value means **uplift** (toward to radar sensor or range decrease) and the (+) value means **down-lift** (away to radar sensor or range increase).



Picture 11. Line of Slight Displacement in BAJA California
(Date: 2009.12.17 – 2010.05.04)

*los = the DinSAR result based on line of sight dispalcement. The unit mesuarement is in mm (milimeter).

This LOS displacement is already converted to ground surface viewing so that the (-) negative value means **down-lift** and the (+) value means **uplift**.



Picture 12. DinSAR result viewed by Google Earth

The background of this InSAR result is A 7.2 magnitude earthquake was occured in Baja California (4 April 2010) called Sierra El Mayor earthquake. Based on DinSAR result we could know that radar remote sensing could detect the deformation phenomenon because of affection of earthquake.

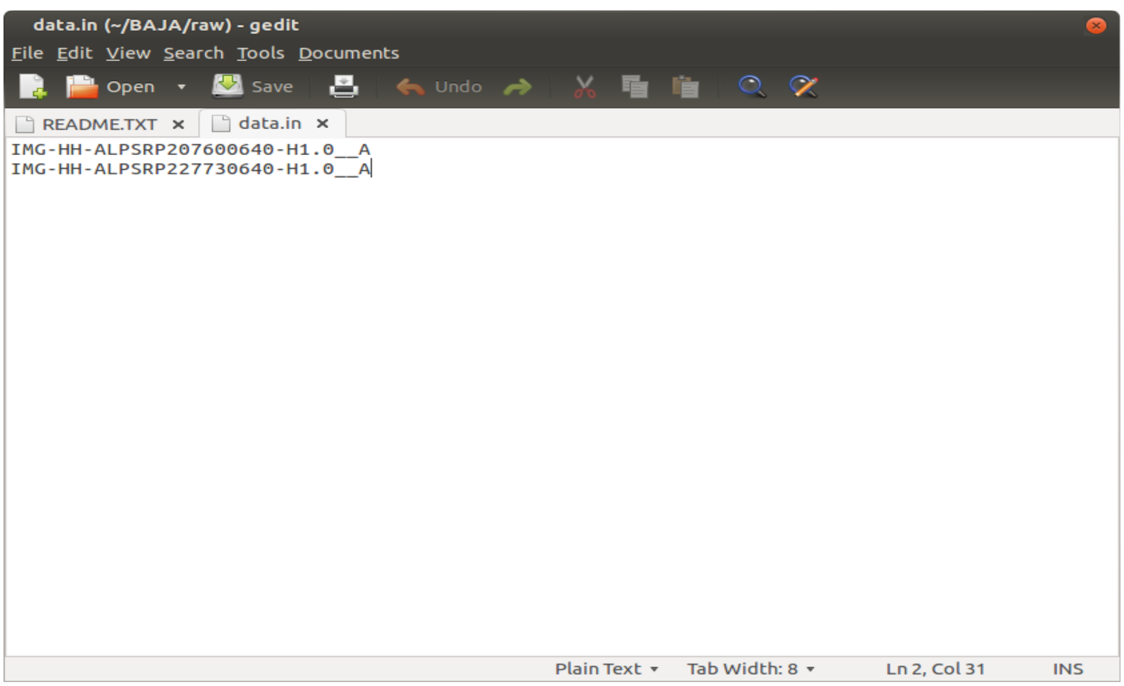
Another option is you could process InSAR with step by step process, which is:

1. Preprocess the RAW data

Type on terminal

```
cd ~
cd BAJA
cd raw

ls IMG* >> data.in

================ edit data.in =================

pre_proc_batch.csh ALOS data.in
```

ls IMG* >> data.in is a command to make a list all of the name filea with *IMG. To make GMTSAR know which one is master file, you must edit data.in file and put the master file in the first line. You can also set the -radius and -near_range on the first line to have this frame match other frames along the same track or you could leave the default setting.



Picture 13. data.in

If it is done, you will have some new files with extention .PRM and .raw

2. Align the SLC images. This is the focusing process to make raw file to be SLC (single look complex) file.

Type on terminal

```
cd ..
mkdir SLC
cd SLC
cp ../raw/*.PRM .
ln -s ../raw/*.raw .
ln -s ../raw/LED* .
```

===>

```
align.csh    ALOS    IMG-HH-ALPSRP207600640-H1.0__A    IMG-HH-ALPSRP227730640-H1.0__A
```

It will create SLC file and PRM file on each image.

3. Make the topo_ra.grd

```
cd topo
cp ../SLC/IMG-HH-ALPSRP207600640-H1.0__A.PRM master.PRM
ln -s ../raw/LED-ALPSRP207600640-H1.0__A .
```

This process need file dem.grd. The example data already has the dem, if you want to download another location, just generate the dem on http://topex.ucsd.edu/gmtsar/demgen

```
dem2topo_ra.csh master.PRM dem.grd
```

It will create topo_ra.grd file, to view the result just open the postscript image, topo_ra.ps. trans.dat file will be used for geocoding process which content information of lon, lat, topo to range and azimuth.

Picture 14. topo_ra.ps view

4. Interferogram

This is the main process to make an interferogram image, prepare the files with typing on terminal

```
cd ..
mkdir intf
cd intf
mkdir 091217_100504

cd 091217_100504
ln -s ../../raw/LED-ALPSRP207600640-H1.0__A .
ln -s ../../raw/LED-ALPSRP227730640-H1.0__A .
cp ../../SLC/IMG-HH-ALPSRP207600640-H1.0__A.PRM .
cp ../../SLC/IMG-HH-ALPSRP227730640-H1.0__A.PRM .
ln -s ../../SLC/IMG-HH-ALPSRP207600640-H1.0__A.SLC .
ln -s ../../SLC/IMG-HH-ALPSRP227730640-H1.0__A.SLC .
ln -s ../../topo/topo_ra.grd .
```

===>

```
intf.csh IMG-HH-ALPSRP207600640-H1.0__A.PRM IMG-HH-ALPSRP227730640-
H1.0__A.PRM  -topo topo_ra.grd
```

When it is finished, the results are :

display_amp.grd, display_amp.ps	- amplitude of interferogram
phase.grd, phase.ps	- phase of interferogram
corr.grd, corr.ps	- correlation of interferogram

5. Filter Interferogram

===>

```
filter.csh IMG-HH-ALPSRP207600640-H1.0__A.PRM IMG-HH-ALPSRP227730640-H1.0__A.PRM gauss_alos_200m 2
```

the number in the last means :

1 = if you want higher resolution images and will create bigger file size
2 = if you want smaller resolution images and will create smaller file size

For ALOS satellite, GMTSAR provide many filters:
- gauss_alos_100m
- gauss_alos_200m
- gauss_alos_300m
- gauss_alos_500m
- gauss_alos_700m

The result of this process is name with _filt in intf folder

6. Unwrap Phase

The unwrap phase will use SNAPHU program, type on terminal

```
cd intf/20760_22773
shaphu.csh .10
```

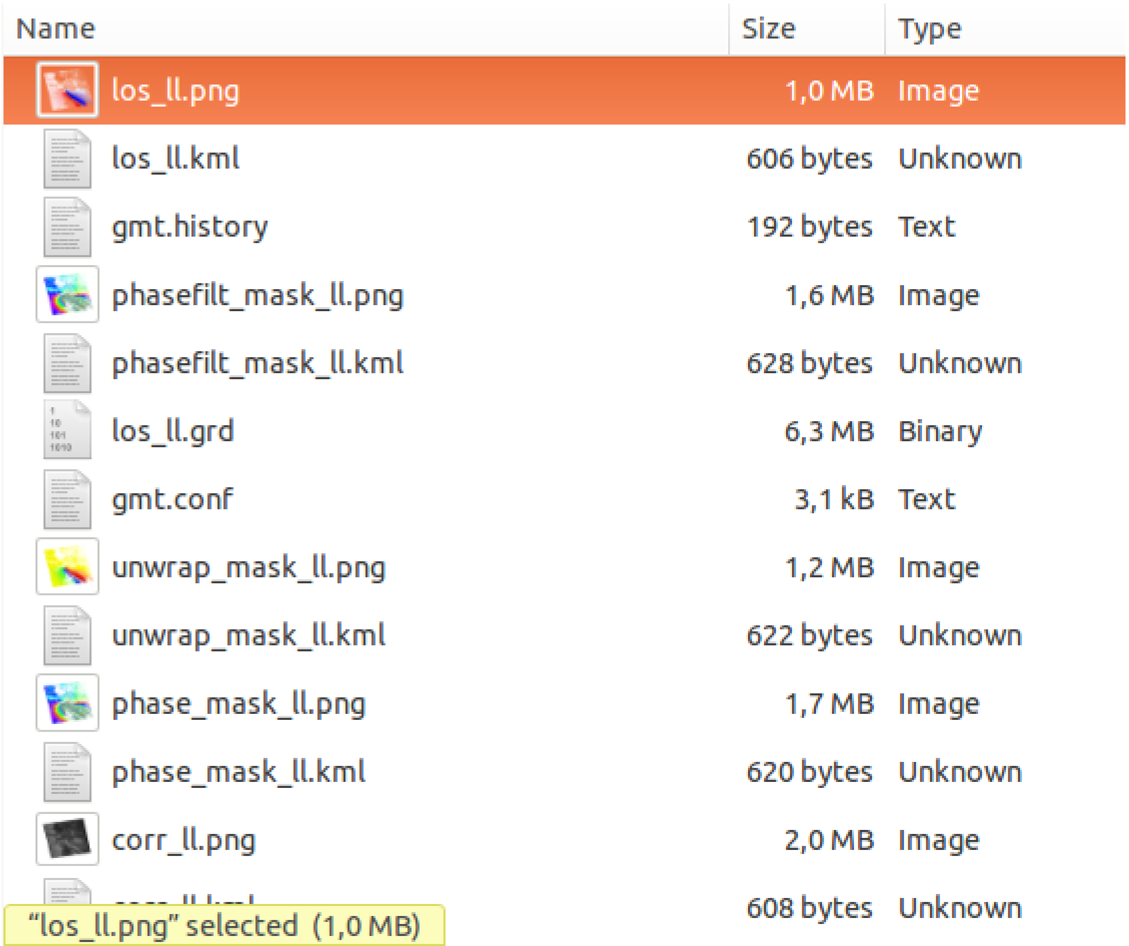
The result is name with unwarp_ file

7. Geocoding

```
cd intf/20760_22773
ln -s ../../topo/trans.dat .
geocode.csh .10
```

The argument 0.10 is used to mask the phase, if the coherence is less than <0.10, gmtsar will make it blank.

Finally, you are in the last but not least step, to see the result on google earth you will have the KML image of the los, unwrap, phase, correlation and display amplitude.



Picture 15. the result from GMTSAR InSAR procesing

Congratulation, you already processed InSAR !

This tutorial is an only overview of GMTSAR. I recommend you to also read the official GMTSAR Guide Book to see every step and theory of InSAR processing on GMTSAR. (Find the document on /usr/local/GMT5SAR/doc)

For any question and feedback to my tutorial, please send email to n.isya@tu-braunschweig.de or noorlaila@geodesy.its.ac.id

Reference:

Sandwell, David., Rob Mellors, Xiaopeng Tong, Matt Wei, and Paul Wessel. October 2010. GMTSAR: An InSAR Processing System Based on Generic Mapping Tools. Scripps Institution of Oceanography.

Sandwell, D. ., R. . Mellors, X. Tong, M. Wei, and P. Wessel (2011), Open radar interferometry software for mapping surface deformation, Eos Trans. AGU, 92(28), doi:10.1029/2011EO280002.

Sandwell, David, Mellors, Rob, Tong, Xiaopeng, Wei, Matt, & Wessel, Paul. (2011). GMTSAR: An InSAR Processing System Based on Generic Mapping Tools. UC San Diego: Scripps Institution of Oceanography. Retrieved from: http://escholarship.org/uc/item/8zq2c02m

http://topex.ucsd.edu/gmtsar/

https://en.wikipedia.org/wiki/2010_Baja_California_earthquake