A Python Based InSAR Processing Tool For ISRO SAR Missions

Rajvi Panchal[(1)]*, Sanid Chirakkal[(2)], Deepak Putrevu[(2)] and Arundhati Misra[(2)]
(1) Chhotubhai Gopalbhai Institute of Technology, Bardoli, Surat, India 394350, panchalrajvi@gmail.com
(2) Advanced Microwave and Hyperspectral Techniques Development Group, Space Applications Center,
Indian Space Research Organization, Ahmedabad, India 380015
{sanid, dputrevu, arundhati}@sac.isro.gov.in

Abstract

Interferometric Synthetic Aperture Radar (InSAR) is a remote sensing technique widely used to generate elevation maps, commonly known as interferograms, that depicts surface deformations and topographic trends. Changes in topography and deformations can be measured over span of days to years and are recorded in the form of fringes. Taking into consideration the upcoming NASA-ISRO Synthetic Aperture Radar (NISAR) mission, which is designed to support wide-swath interferometry, it is high time to develop a InSAR processing tool dedicated to ISRO missions. Due to its versatile features, popularity, flexibility and the huge library support, the tool development was chosen to be in Python3 programming language. In this paper the first results obtained from the in-house developed, python 3 based, software tool for InSAR processing are presented. The tool is slated to become part of the Microwave Data Analysis Software (MIDAS) of SAC, which is a generic SAR processing software suite. Currently, the tool accepts ERS 1/2, ENVISAT, RADARSAT-2 and ALOS-2 data. For verification purpose, the outputs generated by the tool are compared with those generated by the freely available Delft Object-oriented Radar Interferometric Software (DORIS) developed by the Delft Institute of Earth Observation and Space Systems (DEOS), Delft University of Technology.

1 Introduction

Synthetic Aperture Radar (SAR) is a microwave imaging system. It combines radar and signal processing to form high-resolution backscatter images. The key to SAR is coherent recording and processing of the received signal over multiple pulses. Due to side effects like forshortening and layover, a single SAR image cannot be used to produce elevation maps of terrains. To overcome this, Interferometric synthetic aperture radar (InSAR) was developed to record the complex amplitude and phase information digitally for each antenna. In this way, the relative phase of each image point could be reconstructed directly. The first demonstrations of such systems with an airborne platform were reported by Zebker and Goldstein [1], and with a spaceborne platform using SeaSAT data by Goldstein and colleagues [2, 3]. InSAR is the synthesis of conventional SAR techniques and interferometry techniques that have been developed over several decades in radio astronomy [4]. The phases of images with a difference of position (e.g., two antennae on one plane acquire images simultaneously) or with a difference of time (e.g., one antenna acquires images at two distinct times) can be compared after proper image registration [5]. SAR interferometry makes use of the phase information contained in SAR data. This relative phase changes with the surface being monitored. The phase difference obtained when one object is imaged from two locations can be used to determine the object's height. Similarly, if you image an object from the same location at two different times and the backscattered signals phase differs, then it can be infered that the object has moved. For these reasons, InSAR is often used to generate digital elevation maps or to detect surface changes. In interferometry, the complex phase information of one image is multiplied by the complex conjugate phase information of the second image to form an interferogram, effectively canceling the common backscatter phase in each resolution element, but leaving a phase term proportional to the differential path delay [6]. This difference in the phase of each pixel can be viewed as fringes in the interferogram generated. With InSAR, it is possible to detect and measure path length differences with centimeter or even millimeter accuracy which is independent of the distance between sensor and the scene being monitored. This makes InSAR highly relevant for both air-borne and space-borne remote sensing. After applying various filters, these outputs can be analysed by geophysical experts for studying seismic deformations, volcanic activities, ice drifts etc. The details of in-house InSAR processing tool developed in python3 programming language are mentioned in this paper.

2 Methodology

InSAR processing consists of a sqeuence of complex steps, but broadly speaking, these can be clustered into three steps. A high-level description of these steps is shown in figure 1. Block 1 represents the preprocessing of input SAR data. The tool takes SAR images as input and coverts them to a common format. Block 2 represents coregistration of the two input images. Block 3 represents interferogram generation along with in-house developed flat earth effect

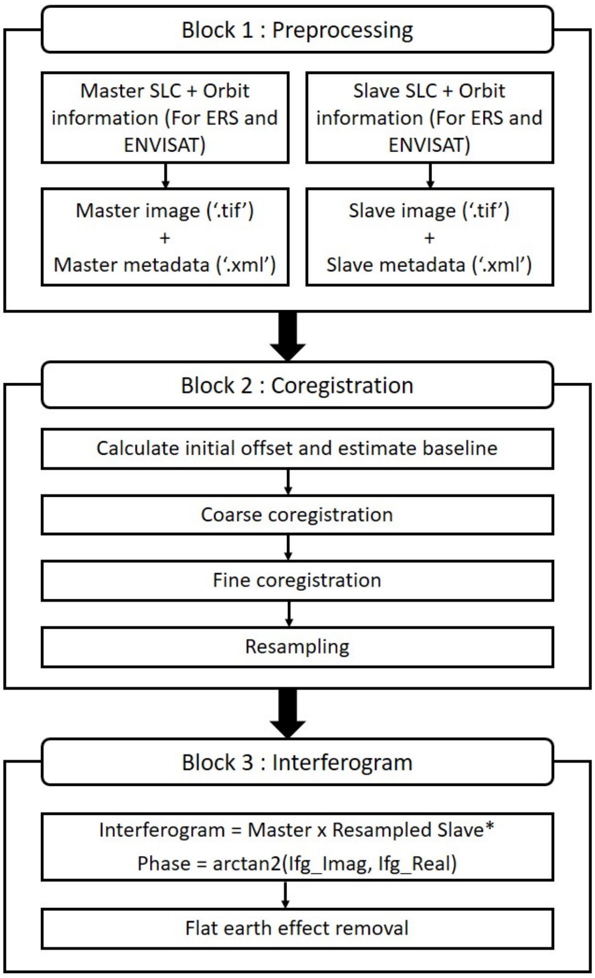

Block 1 : Preprocessing
Master SLC + Orbit information (For ERS and ENVISAT)
Slave SLC + Orbit information (For ERS and ENVISAT)
Master image ('.tif') + Master metadata ('.xml')
Slave image ('.tif') + Slave metadata ('.xml')
Block 2 : Coregistration
Calculate initial offset and estimate baseline
Coarse coregistration
Fine coregistration
Resampling
Block 3 : Interferogram
Interferogram = Master x Resampled Slave*
Phase = arctan2(Ifg_Imag, Ifg_Real)
Flat earth effect removal

Figure 1. InSAR processing steps.

removal. This is the end product of the tool. Detailed explanation of all these steps is given below.

2.1 Data preprocessing

Since, different satellite data are provided in different formats, it needs to be reassimilated in a common format. The tool generates a SLC '.tif'file and a '.xml'file containing metadata regarding the SAR image in a common format. The tool requires following files as input :

- For ERS 1/2 and ENVISAT : SLC and precise orbit files
- For RADARSAT-2 : SLC tif and metadata xml files
- For ALOS-2 : SLC and leader files

2.2 Coregistration

Even a small misalignment between the master and slave images would result in decorrelation noise and will generate inaccurate interferogram. Hence, the images need to be coregistered first. This is the most important and complex step in InSAR processing and it ensures that each ground target contributes to the same pixel in both images. Since all the information contained in the SLC must be preserved, coregistration accuracy must be very good i.e., in sub-pixel level. Typically, it is based on maximizing cross-correlation between the SAR images by applying local line and pixel shift to one of them [8].

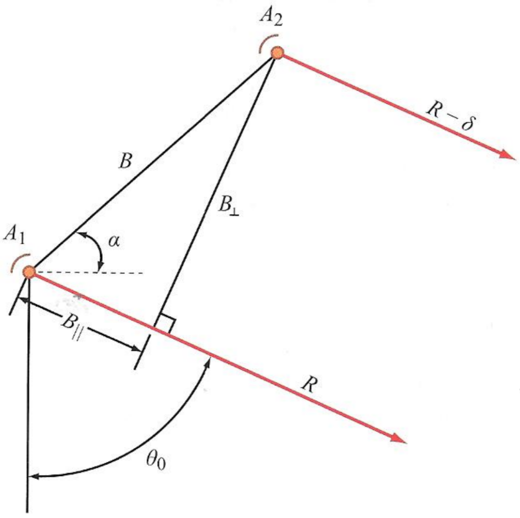


Figure 2. Parallel and perpendicular components of baseline B as defined by incidence angle θ. Range R is from A_1

Baseline plays an important role in computation of image offset. Interferometric baseline is the difference of platform position vectors when a given object is imaged. The total offset Δ between two images can be expressed as a function of the parallel ($B_{||}$) and perpendicular ($B_\perp$) baseline components as shown in figure 2.

$$\Delta = B_{||} + B_\perp \frac{dR}{R_o tan\theta_o} \tag{1}$$

Equation 1 suggests that Δ has a constant shift equal to parallel component of the baseline, and a linearly increasing shift/stretch that varies inversly with $R_o tan\theta_o$ [8]. Because of this shift and scale, an object observed at a pixel in one image would be at a different pixel in the other image. While forming interferogram, the phase difference relates the phase of a pixel in the first image to the corresponding phase of the same pixel (which may be at a different location) in second image. Offsets between them are computed and then slave is resampled in order to match with the master. Offset is defined in such a way that for a point in the master with coordinates $Master(line, pixel)$ and the same point in the slave image with (slave system) coordinates $Slave(line, pixel)$ it holds $Slave(l, p) = Master(l, p) + Offset(l, p)$. For implementing coregistration, we follow the algorithm mentioned below.

1. **Initial offset estimation from orbit information :**

   Initial offsets are estimated from satellite state vectors from precise orbit files. We calculate the initial offset, which will give an estimate for the coarse offsets which are refined later. For computing the initial offset, we implement the following algorithm [7]:

- Compute the position of point P on the ellipsoid corresponding to the center of master image. Precise orbits are necessary for this. The equations for point on the ellipsoid and satellite S in its orbit (where x denotes (x, y, x)) are [7] :

$$d\vec{x} = \vec{x} - \vec{x}_s \tag{2}$$

$$\dot{\vec{x}}_s . d\vec{x} = 0 \tag{3}$$

$$d\vec{x} . d\vec{x} - (c\, t_{range})^2 = 0 \tag{4}$$

$$\frac{x^2}{a^2} + \frac{y^2}{a^2} + \frac{z^2}{b^2} - 1 = 0 \tag{5}$$

  Here, doppler equation (3) ensures that point P at the surface lies perpendicular to orbit due to zero doppler processing. Range equation (4) states that geometrical distance to P on the surface is equal to the speed of light times the range time. Ellipsoid equation (5) ensures that the point lies on the ellipsoid (since Earth is assumed to be ellipsoid for InSAR processing). Now, compute the position and velocity of the satellite based on the line number and pulse repitition frequency (PRF). Next, compute the range time based on the pixel number and range sampling rate (RSR). Solve the above doppler, range and ellipsoid equations iteratively until convergence ($\delta(x) < 1e-8$) to solve for the point P on ellipsoid.
- Based on the Doppler equation, compute the position of satellite when the slave image was being aquired, corresponding to the point P on an ellipsoid, and then compute the (line,pixel) coordinates in the slave system.
- The difference (master-slave) between the (line,pixel) coordinates is defined as the offset.

2 **Coarse and fine coregistration :**

A single offset is not sufficient for coregistration. This can be overcome by estimating offsets for a large number of locations. Offsets are calculated by choosing a small region in master and cross-correlating it over a similar area in slave to find the best match. The position of correlation peak then gives the offset at that location [8]. 16-20 locations need to be identified and offset should be calculated at those locations while coarse coregistration. Cross-correlation is computed with equation 6.

$$K_c(i,k) = \frac{\sum_i \sum_k C_1(m,n)\, C_2^*(m_2+i, n_2+k)}{\sqrt{\overline{C_1}\, \overline{C_2}}} \tag{6}$$

Where,
$C_1(m_1,n_1)$ = Complex image of the selected region in master,
$C_2(m_2,n_2)$ = Complex region of the selected region in slave,
$K_c(i,k)$ = Cross-correlation function,
$\overline{C_1} = \sum\sum C_1(m_1,n_1)\, C_1^*\,(m_1,n_1)$ and
$\overline{C_2} = \sum\sum C_2(m_2+i,n_2+k)\, C_2^*\,(m_2+i,n_2+k)$

For fine coregistration, the offsets are computed in the space domain and the correlation is computed using magnitue in the spectral space. This offset will be calcuated with sub-pixel accuracy. We calculate the offsets for a large number of locations (typically, 400-500). We will use these offsets for the next step, i.e., resampling.

3 **Resample Slave :**

Resampling is the reconstruction of original signal at non-uniform fractional grid points using slave points on regular grid and azimuth and range offset matrices. For executing this, we will first require offsets at each pixel of the image. Offsets calculated in previous step are used to fit a 2d polynomial of degree 1 or 2. The function derived then produces the offset value for any pixel image. A 2d-polynomial is defined as in equation 7:

$$f(x,y) = \sum_{i=0}^{d} \sum_{j=0}^{i} \alpha_{i-j,j}\, x^{i-j}\, y^j \tag{7}$$

Here, $f(x,y)$ can be defined as the 2d function that gives the line offset and the pixel offset value at (line, pixel) value as (x, y) and d is the degree of the polynomial (1 or 2). The azimuth and range offset should be calculated independently. Once we get offset values for all the pixels of the image, we can resample/interpolate slave over the master.

2.3 Interferogram generation

The complex interferogram is calulated as :

$$I = M * R^* \tag{8}$$

Where, M is the complex master image and R^* is the conjugate of complex resampled slave. The phase image of interferogram is given by :

$$\phi = arctan_2(I_{imag}, I_{real}) \tag{9}$$

After generation of interferogram, periodic interference fringes can be seen. Flat-earth effect and can overshadow the phase variations of topographical changes, and moreover bring much burden for subsequent phase filtering and unwrapping. Hence, removal of flat earth effect is crucial in InSAR processing. There are various methods to remove flat earth effect. In our tool, we have used precise orbital information and fit a n-degree polynomial to generate the phase [9]. Path difference between the master and slave contains the flat earth phase information with the following equation :

$$\phi_{fe} = \frac{-4\pi}{\lambda}\,[d(M,p) - d(S,p)] \tag{10}$$

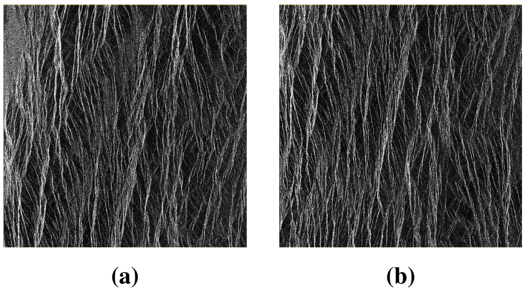


Figure 3. Subset of SAR intensity images for the ENVISAT pair of data over Bam, Iran, over which interferogram is to be generated. (a) Master (aquired on 2003-12-03) and (b) Slave (aquired on 2004-01-07).

Where, M is the position of master satellite on the orbit, S is the position of slave satellite on the orbit, p is the control point on ground and d is the euclidean distance. The coordinates of M and S can be estimated using the precise orbit information as done earlier. Now, the phase value of a set of scattered points over the image can be used to fit a n-degree polynomial (equation 7). The tool allows fitting of 3, 5 or 7 degree polynomial. For Bam data, we have fitted a 5-degree polynomial. The function then derived can be used to calculate the phase value at any location for the image. To eliminate the flat earth phase, the simulated flat earth phase is multiplied with the complex interference pattern.

$$Ifg_{flat} = Ifg * e^{-j\phi_{fe}} \tag{11}$$

This will subtract the simulated phase from the original phase of the image to obtain flattened interferogram. The phase obtained after the subtraction does not contain parallel fringes.

3 Results

ENVISAT data over the ancient city of Bam, in southeastern Iran is chosen as the study area. This area was chosen because a major earthquake hit that area and data for the same is available before and after the natural calamity occured. Thus, it is possible to identify the deformations caused by the earthquake by the means of InSAR processing. The area under supervision is shown as intensity images. The scene captured by the sensor on 2003-12-03 is taken as master and that taken on 2004-01-07 is taken as slave. A subset of master intensity image is shown in figure 3(a) and slave intensity image is shown in figure 3(b). We can see the shift between master and slave image clearly in the figure. Metadata regarding the SLC pair used for tesing is given in table 1.

First, the tool takes the SLC data and extracts metadata into '.xml'files and generates slc '.tif'images. Then, baseline parameters are calculated as mentioned in table 1. Initial offsets are estimated using the state vectors given in metadata. For ENVISAT pair of data over Bam, the azimuth and range offsets were found to be 1123 lines and 34 pixels.

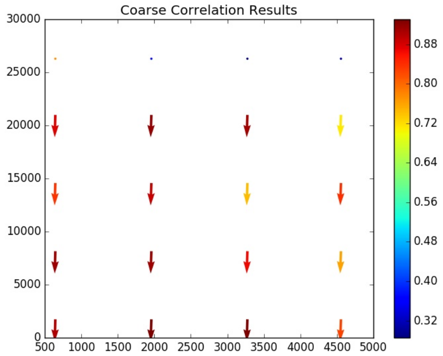


Figure 4. Output of coarse coregistration. Coarse offsets are plotted as vector field.

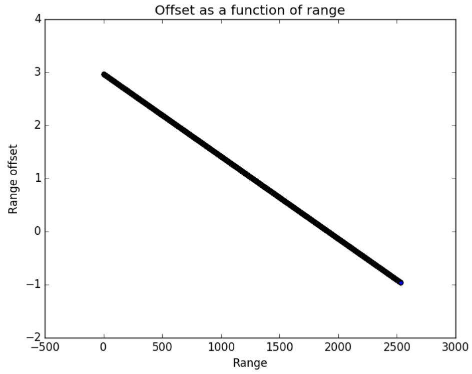
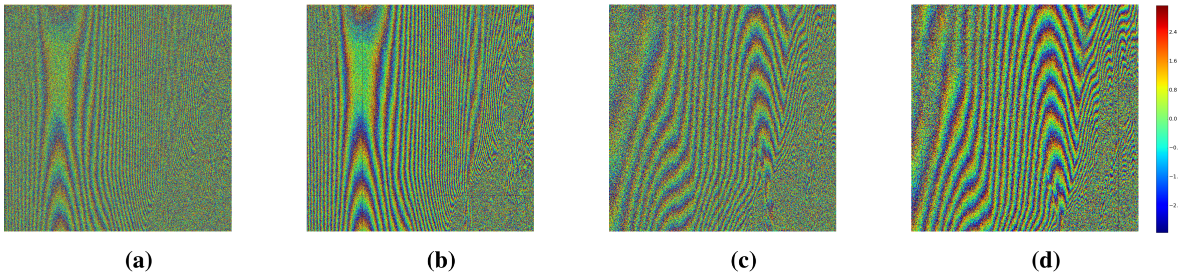


Figure 5. Plot of measured range offsets as a function of range location for a sample azimuth line.

Table 1. Metadata of the ENVISAT pair of data over Bam.

Parameter	**Master**	**Slave**
Date	2003-12-03	2004-01-07
Scene mid latitude	29.138499	29.105279
Scene mid longitude	58.521873	58.506576
Pulse Repetition Frequency (PRF in Hz)	1652.415649	
Range Sampling Rate (RSR in MHz)	19.207680	
Slant range time to first pixel (2-way in ms)	5.509109	
Radar Wavelength (in m)	0.0562356	
Radar Frequency (in MHz)	5331.004416	
Baseline (in m)	586.570228	
Baseline Parallel Component (in m)	270.259264	
Baseline Perpendicular Component (in m)	520.600194	
Horizontal Baseline (in m)	581.837303	
Vertical Baseline (in m)	-74.363867	



(a) **(b)** **(c)** **(d)**

Figure 6. Phase image of complex interferogram over the ENVISAT pair of data over Bam, Iran, generated by (a) in-house developed tool and (b) DORIS software. (c) and (d) show the phase images of interferogram after flattening produced by in-house developed tool and DORIS software respectively.

Next, the images are coregistered. Figure 4 shows the output of coarse coregistration applied pair of data. Here, the offsets have been plotted as a vector field. We can clearly see the trend of offset vectors being in a single direction. Fine coregistration is then implemented over the common area of the two images. The list of fine offsets with sub-pixel accuracy are then used to resample the slave. 1-degree polynomial fitting was done to generate the azimuth and range offsets for the whole image. Figure 5 shows the plot of range offset for a sample azimuth line as a function of range. After resampling the slave, raw interferogram is generated. The phase image of the complex interferogram generated by in-house developed tool is shown in figure 6(a) and the interferogram generated by using DORIS software is shown in figure 6(b). Here, we can clearly see flat earth effect in the form of parallel lines. By comparing both results visually, we can see that the graphs are almost indistinguishable demonstrating the reliability of our tool. In both the interferograms, we can see the parallel fringes due to flat earth effect. Figure 6(c) shows the phase image of the interferogram after flattening generated by in-house developed tool and figure 6(d) is the output of DORIS software. The flattened interferogram is the final product of the in-house developed tool.

4 Conclusion and future scope

This paper discusses the details of InSAR and the in-house developed tool for InSArR processing. The tool was developed in python programming language due to its versatile features, popularity, flexibility and the huge library support. It currently accepts ERS 1/2, ENVISAT, RADARSAT-2 and ALOS-2 data. The results generated by the tool have been compared to those obtained by DORIS software. DORIS is a freely available InSAR processing software. The reliability and robustness of the in-house developed tool has been demonstrated. Still, many refinements and performance enhancements are possible to the algorithms implemented in our tool. These will be taken up as future work to improve the tool. Steps like topography removal, phase unwrapping etc., will also be implemented for DInSAR applications as an expansion to the current work.

References

[1] H. A. Zebker and R. M. Goldstein, ‘Topographic mapping from interferometric SAR observations’, J. Geophys. Res., vol. 91, pp. 4993-4999, 1986.

[2] R. M. Goldstein, H. A. Zebker, and C. L. Werner, ‘Satellite radar interferometry: Two-dimensional phase unwrapping’, Radio Sci., vol. 23, no. 4, pp. 713-720, July/Aug. 1988.

[3] F. Li and R. M. Goldstein, ‘Studies of multibaseline spaceborne interferometric synthetic aperture radars’, IEEE Trans. Geosci. Remote Sensing, vol. 28, pp. 88-97, 1990.

[4] A. R. Thompson, J. M. Moran, and G. W. Swenson, Interferometry and Synthesis in Radio Astronomy. New York: Wiley Interscience, 1986.

[5] D. Massonnet and K. L. Feigl, ‘Radar interferometry and its application to changes in the earth’s surface’, Rev. Geophys., vol. 36, no. 4, pp. 441-500.

[6] Paul A. Rosen, Scott Hensley, Ian R. Joughin, Fuk K. Li, Soren N., Madsen, Ernesto Rodriguez and Richard M. Goldstein, ‘Synthetic Aperture Radar Interferometry’, Proceedings of the IEEE, Vol. 88, No. 3, Mar 2000

[7] Bert M Kampes, Ramon F Hanssen, and Zbigniew Perski. Radar interferometry with public domain tools. In Third International Workshop on ERS SAR Interferometry, ‘FRINGE03’, Frascati, Italy, 1-5 Dec 2003, page 6 pp., 2003.

[8] F. Ulaby, R.K.Moore, A.k.Fung, ‘Microwave Remote Sensing - Active and Passive’, Vol 1:Microwave Remote Sensing Fundamentals

[9] Prakruti Joshi, Keya Desai, Sanid Chirakkal, Deepak putrevu and Ranendu Ghosh, ‘Analysis of flat earth phase removal methods’, ISRPS Archives of the Photogrammetry, Remote Sensing and SIS (Accepted), 2018