INSAR PHASE UNWRAPPING BY MEANS OF A PARTICLE FILTER

Juan J. Martinez-Espla, Tomas Martinez-Marin, J. David Ballester-Berman and Juan M. Lopez-Sanchez
Signals, Systems and Telecommunication Group, EPS, University of Alicante, P.O. Box 99, E-03080 Alicante, Spain
Tel/Fax: +34 96590 9811/9750. E-mail: jjme1@alu.ua.es, tomas@dfists.ua.es, davidb@dfists.ua.es, juanma-lopez@ieee.org

ABSTRACT

This work presents a phase unwrapping (PU) algorithm for SAR interferometry based on a particle filter (PF). This PU algorithm performs simultaneously noise filtering and phase unwrapping. The formulation of this technique provides independence from noise statistics and is not constrained by the non linearity of the problem. Results show a significant improvement with respect to conventional PU algorithms in some situations.

Index Terms— Phase unwrapping, SAR interferometry, state space, grid filter, particle filter.

1. INTRODUCTION

Phase unwrapping (PU) is one of the key processing steps in all applications of SAR interferometry. The book by Ghiglia and Pritt [1] provides an excellent overview of PU algorithms. In general, there are two types of conventional PU methods. The algorithms of the first group, generally named *path following* or *region growing* algorithms, isolate and/or mask problematic zones containing residues and unwrap the interferogram by avoiding these zones. Instead, the techniques of the second group provide a *global solution* which minimizes a cost function over the whole interferogram. Consequences of these strategies are the following: phase information in noisy pixels is not recovered by the path following algorithms; and noisy pixels distort the solution in global approaches, thus affecting the noise-free areas. Some approaches incorporate a pre-filtering stage before staring the PU procedure [2], but this strategy entails the loss of the information contained in noisy pixels. Since we are interested in recovering as much information as possible, an ideal method would consider every pixel to be simultaneously unwrapped and filtered. There exists an algorithm sharing the same objective, which was published in [3]. That algorithm combines a slope estimator with an extended Kalman filter (EKF).

In previous works [4]-[5], we substituted the EKF by a grid-based filter (GbF), which is not subject to any linear or Gaussian constraints, and its better performance in some situations was shown. A detailed tutorial about different methods to broach nonlinear/non-Gaussian problems can be consulted in [6].

In the present work, we introduce a PU algorithm based on a particle filter. Specifically, this new phase unwrapping solution combines a particle filter with an improved version of the phase slope estimator presented in [3] and path following techniques to simultaneously unwrap and filter the interferometric phase.

2. PARTICLE FILTER PHASE UNWRAPPING

2.1. Introduction to the particle filter

The unwrapped phase at pixel k will be referred as the corresponding state x_k at pixel k. The particle filter solution [6] can be adopted since the continuous state space can be divided into N cells or states, $\{x^i_k;\ i=1,\ldots,\ N\}$, which correspond to all possible values of the phase.

Conceptually, the particle filter (PF) is simply a solution that represents the posterior probability function (*pdf*) $p(x_k|\ z_{1:k})$, by a distribution of N_s particles, defined below, extracted from the state space.

The particles are defined as the set of states whose weights w^i_k, introduced below, are the highest per pixel.

With the assumption of a first-order Markovian process, and after a few operations [6], the following expression for the weights is obtained

$$w_k^i \propto w_{k-1}^i \frac{p\left(z_k \mid x_k^i\right) p\left(x_k^i \mid x_{k-1}^i\right)}{q\left(x_k^i \mid x_{k-1}^i, z_k\right)} \tag{1}$$

where $p(x_k|\ x_{k-1})$ defines the evolution model, $p(z_k\ |\ x_k)$ defines the observation model, and $q(\cdot)$ is the so called *importance density* [6].

The posterior density of probability is given by:

$$p\left(x_k \mid z_{1:k}\right) \approx \sum_{i=1}^{N_s} w_k^i \delta\left(x_k - x_k^i\right) \tag{2}$$

where the weights are defined in (1) and $\delta()$ represents the Dirac Delta. It can be shown that the larger N_s, the more (2) approaches the true *pdf* $p(x_k\ |\ z_{1:k})$. Note also that the computational cost increases linearly with N_s.

If the importance density is defined as:

$$q\left(x_k \mid x_{k-1}^i, z_k\right) = p\left(x_k \mid x_{k-1}^i\right), \tag{3}$$

the calculus of the weights can be simplified with the following expression:

$$w_k^i \propto w_{k-1}^i\ p\left(z_k \mid x_k^i\right) \tag{4}$$

A sliding 2π wide window W^s_k is used to cover the complete state space [4].

978-1-4244-2808-3/08/$25.00 ©2008 IEEE

Degeneracy phenomenon is a critical problem affecting PF solutions. It means that after a few iterations, all but one particle will have insignificant weight. To avoid this effect, a resampling technique [6] will be used in this work due to its simplicity and excellent results. The basic idea is to drop particles that have small weights and to concentrate on those with large weights.

The pseudocode of both, a generic particle filter and the resampling strategy, can be consulted in detail in [6] and will not be repeated here due to space constraints.

2.2. Phase unwrapping algorithm (PFPU)

In many occasions, zones containing high density of residues or low coherence have to be processed and unwrapped. In these situations, row by row methods, like EKF [3] and GbF [4] and [5], i.e. methods with a predefined path for unwrapping, are not the best suited. The PF solution presented here combines a particle filter with an improved variant of the local phase estimator and a path following strategy.

The local phase estimator presented in [3] provides slope estimations in horizontal and vertical directions and, as a consequence, collection of slope information from up to four neighbors. That method has been improved in this work to provide slope estimations also in oblique directions. As a result, collection of slope information from up to the eight neighbors is possible, providing estimations with higher reliability. To do so, two types of 5x5 windows around the current pixel are used. As observed in Figure 1, one of the windows is rotated $\pi/4$ radians respect to the other.

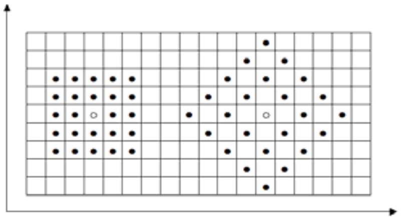


Figure 1: Windows used for computing the omnidirectional local phase estimator.

The procedure to obtain phase slope estimates is illustrated here with an example from the real interferogram introduced in Section 3. Phase slope estimates for pixel (7,76) are calculated and shown in the following.

To obtain phase slope information in horizontal and vertical directions, the left window in Figure 1 is used. The 2D Fast Fourier Transform (FFT2) is calculated for this window. By means of the FFT2, the power spectral density (PSD) can be obtained. As observed in Figure 2, the coordinates containing the maximum value of the PSD define the corresponding phase slope estimates in horizontal and vertical directions.

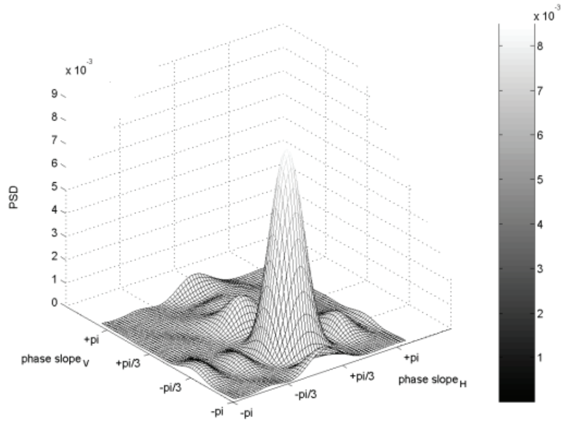


Figure 2: PSD_{max} and the corresponding values of phase slope information in vertical and horizontal directions (phase slope_{V} and phase slope_{H}, respectively).

Then, to obtain phase slope information in oblique directions, the right window in Figure 1 is used. The process is the same and the coordinates containing the maximum value of the PSD define the corresponding phase slope estimates in oblique directions, as observed in Figure 3.

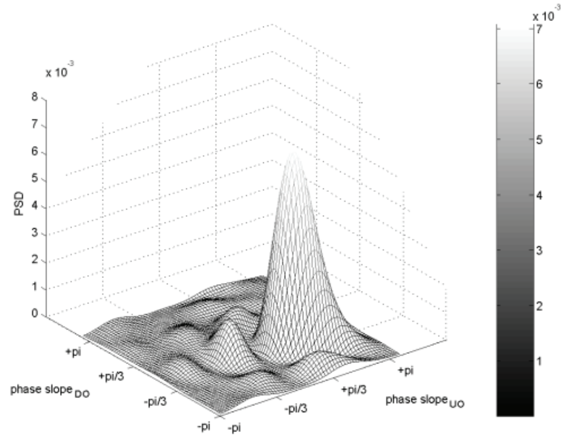


Figure 3: PSD_{max} and the corresponding values of phase slope information in oblique (Down-Oblique and Up-Oblique) directions (phase slope_{DO} and phase slope_{UO} , respectively).

This process is repeated for every pixel in the interferogram, thus obtaining an omnidirectional phase slope distribution map.

With respect to the unwrapping method itself, the evolution model $p(x_k | x_m)$ presents the following expression:

$$x_k^i \sim \frac{1}{N_{\text{n}}}\sum_{m=1}^{N_n} p_m\left(x_k^i \mid x_m^i\right), \qquad (6)$$

where N_n is the total number of unwrapped neighbors used for the prediction, and $p_j(x_k | x_m^i)$ is the evolution model to be applied in the direction from pixel m towards pixel k, given by:

$$x_k^i = x_m^i + \Delta\hat{\varphi}_m^j + n_m^j, \tag{7}$$

where x^i_m refers to the unwrapped phase at pixel m, $\Delta\hat{\varphi}_m^j$ refers to the slope estimation at pixel m in the direction j (the one towards pixel k), and n^j_m refers to the Gaussian noise sample at pixel m that models the uncertainty introduced by the slope estimation at pixel m in the direction j.

The observation model $p(z_k \mid x_k)$ can be expressed as:

$$z_k = x_k^i + v_k \tag{8}$$

where v_k is modeled as a residual Wishart discrete distribution.

To implement the path following search strategy, a map of flags will be generated for classifying the pixels, according to the following values:

- 0 for pixels not to be considered for unwrapping yet.
- 1 for those pixels being candidates to be unwrapped, i.e. pixels with at least two unwrapped neighbors.
- -1 for pixels already unwrapped and with neighbors still to unwrap.
- -5 for pixels unwrapped and without neighbors still pending of unwrapping.

In principle, the weight distribution for every unwrapped pixel must be saved to be recovered when necessary. However, this strategy would result in a waste of memory resources, especially for large interferograms. When an unwrapped pixel does not have any neighbor pending of unwrapping, its weight distribution is not necessary anymore. The distinction between flag -1 and flag -5 permits the removal of the weights distribution for those pixels whose flag value is -5 and, therefore, saving memory resources. Figure 4 represents the unwrapping process and maps of flags after the instants when 1000 and 5000 pixels have been unwrapped.

The path following strategy works as follows: at each repetition of the algorithm, from the group of pixels whose associated flag value is 1, the one with the highest associated coherence γ will be the one selected to be unwrapped.

For instance, at iteration 5001 the candidate to be unwrapped will be pixel (8,76). A zoom of the distribution of flags around is shown in Figure 5. As observed, five neighbors have already been unwrapped, and their slope information in the corresponding directions (from the unwrapped pixels towards to candidate to be unwrapped) will be available. At this stage, the pixel can be unwrapped by making use of the path following PFPU algorithm presented in Figure 6. The map of flags and the weight distribution list are updated accordingly.

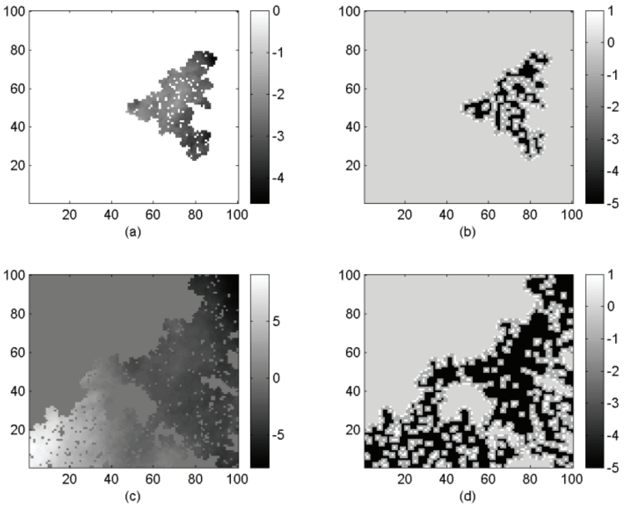


Figure 4: Path following strategy: (a) distribution of the first 1000 unwrapped pixels, (b) map of flags after the unwrapping of 1000 pixels, (c) distribution of the first 5000 unwrapped pixels, and (d) map of flags after the unwrapping of 5000 pixels

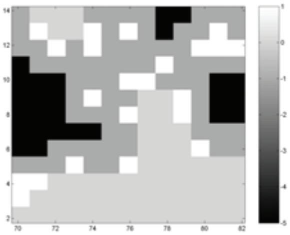
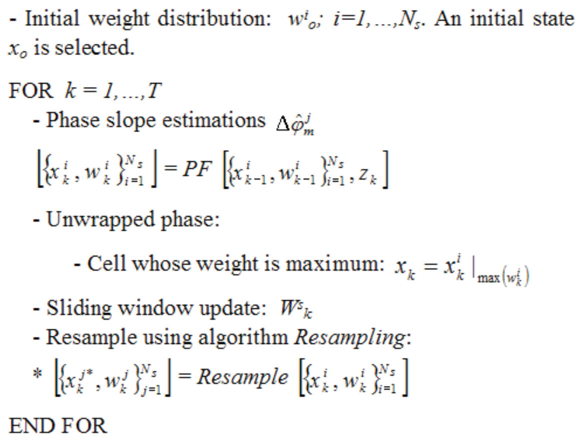


Figure 5: Map of flag values: 0 for pixels not unwrapped yet, 1 for pixels candidate to be unwrapped, -1 for pixels already unwrapped and with neighbors still to unwrap, and -5 for pixels unwrapped and without neighbors pending of unwrapping.



```
- Initial weight distribution: $w^i_o$; $i=1,\dots,N_s$. An initial state $x_o$ is selected.
FOR $k = 1,\dots,T$
   - Phase slope estimations $\Delta\hat{\varphi}_m^j$
   $\left[\{x_k^i, w_k^i\}_{i=1}^{N_s}\right] = PF\left[\{x_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, z_k\right]$
   - Unwrapped phase:
      - Cell whose weight is maximum: $x_k = x_k^i|_{\max(w_k^i)}$
   - Sliding window update: $W^s_k$
   - Resample using algorithm Resampling:
   * $\left[\{x_k^{j*}, w_k^j\}_{j=1}^{N_s}\right] = Resample\left[\{x_k^i, w_k^i\}_{i=1}^{N_s}\right]$
END FOR
```

Figure 6: Pseudocode of the path following PF PU algorithm.

3. RESULTS

The following results correspond to a small crop extracted from a real interferogram obtained with images acquired by the ERS-1 and ERS-2 satellites in tandem configuration. Figure 7 shows the coherence map and the input wrapped phase. The path following PFPU solution presented in Section 2.2 will be compared to an equivalent path following grid-based filter solution. The continuous space contained inside the 2π sliding window has been translated into a discrete state space composed of *N=100* cells. Note that for the PFPU solution a maximum of *Ns=50* particles is used.

As observed in Figure 8.a, the path following PFPU algorithm obtains the same results as the equivalent GbF solution (Figure 9.a) with the advantage of a 35% lower computational cost.

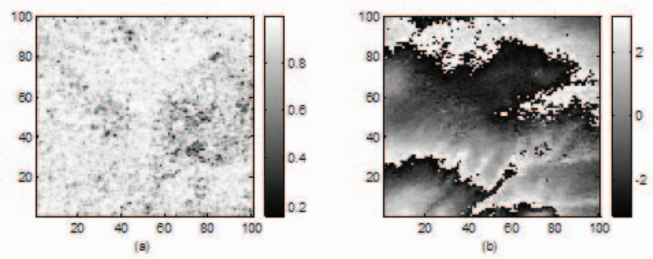


Figure 7: Small real interferogram with ERS images: (a) coherence, and (b) input wrapped phase.

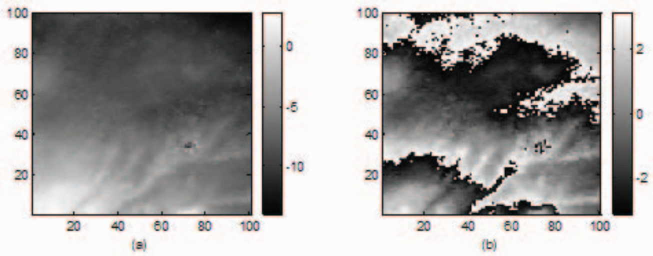


Figure 8: Path Following PF PU solution for the small ERS interferogram: (a) unwrapped phase, and (b) re-wrapped phase.

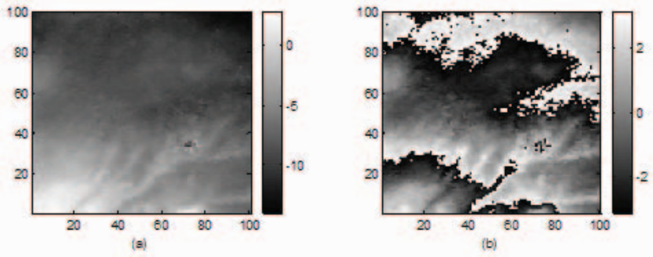


Figure 9: Path Following GbF PU solution for the small ERS interferogram: (a) unwrapped phase, and (b) re-wrapped phase.

4. CONCLUSION

A new solution for phase unwrapping that simultaneously filters and unwraps the phase contained in an interferogram is presented in this work. This solution is based on the combination of a PF, an enhanced version of the local phase estimator introduced in [3] and a path following strategy. The path following PFPU algorithm has been compared to an equivalent path following GbF solution. It has been shown that the PFPU obtains the same result as the GbF method with a lower computational cost.

Our research lines at present are focused on two topics. First, the introduction of better cost or quality functions involving not only the coherence of the pixel but some other parameters to provide more accurate information. Second, the introduction of different phase slope estimators into the PFPU approach. For instance, the matrix pencil algorithm proposed in [7] and the slope average-based method introduced in [8] are possible options.

5. ACKNOWLEDGMENTS

This work was supported by the Spanish Ministry of Education and Science, and EU FEDER, under Project TEC2005 06863-C02-02.

The SAR real images used in this work have been supplied by the European Spatial Agency (ESA) within the framework of the ESA EO project Cat.1-2494.

6. REFERENCES

[1] D. C. Ghiglia and M. D. Pritt, *Two-Dimensional Phase Unwrapping.Theory, Algorithms and Software*. New York: Wiley, 1998.

[2] D. Meng, V. Sethu, E. Ambikairajah and L. Ge. “A novel technique for noise reduction in InSAR images”, *IEEE Geoscience and Remote Sensing Letters.* Vol. 4, No. 2, pp. 226-230, April 2007.

[3] O. Loffeld, H. Nies, S. Knedlik and W. Yu, “Phase Unwrapping for SAR Interferometry – A Data Fusion Approach by Kalman Filtering”, *IEEE Transactions on Geoscience and Remote Sensing*. Vol. 46, No. 1, pp. 47-58, January 2008.

[4] J. J. Martinez-Espla, T. Martinez-Marin and J. M. Lopez-Sanchez, “A Grid-based Filter Approach to Solve InSAR Phase Unwrapping”, *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS’07)*, pp. 4497-4500, Barcelona, Spain, July 2007.

[5] J. J. Martinez-Espla, T. Martinez-Marin, and J. M. Lopez-Sanchez. “Using a Grid-based Filter to Solve InSAR Phase Unwrapping”. *IEEE Geoscience and Remote Sensing Letters*. Vol. 5, No. 2, pp. 147-151, April 2008.

[6] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. “A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking”. *IEEE Trans. Signal Processing.* Vol. 50, No. 2, pp. 174-188, Feb. 2002.

[7] G. Nico and J. Fortuny. “Using the Matrix Pencil Method to Solve Phase Unwrapping”, *IEEE Transactions on Signal Processing.* Vol. 51, No. 3, pp. 886-880, March 2003.

[8] W. Xu and I. Cumming. “A Region-Growing Algorithm for InSAR Phase Unwrapping”. *IEEE Transactions on Geoscience and Remote Sensing.* Vol. 37, No. 1, pp. 124-134, January 1999.