

A Cluster-Analysis and Convex Hull-Based Fast Large-Scale Phase Unwrapping Method for Single- and Multibaseline SAR Interferograms

Yang Lan, *Member, IEEE*, Hanwen Yu , *Senior Member, IEEE*, Mengdao Xing , *Fellow, IEEE*, and Antonio Plaza , *Fellow, IEEE*

Abstract—For synthetic aperture radar (SAR) interferometry (InSAR), phase unwrapping (PU) is an important and difficult step. Due to the high computational complexities of the classical and skilled PU methods, the size and number of interferograms to be processed together must be considered in InSAR applications. However, with the rapid development of InSAR technology, the scale of interferometric data has significantly increased, which has brought the following two difficulties to InSAR processing: excessive memory consumption and unacceptably long computing time. To solve these problems, a fast large-scale PU method based on cluster analysis and convex hull (CCFLS) is proposed in this article. It was developed for single-baseline InSAR and refined under the multibaseline two-stage programming approach InSAR framework. The main idea of CCFLS is to reduce the consumption of computing resources by discarding the PU processing of worthless low quality areas. The key to this work is to determine the discarded region at a low computational cost while ensuring the same PU accuracy for the remaining region as global processing. The theoretical analysis demonstrates the advantage of the CCFLS method for large-scale InSAR data processing, and experiments also verify that CCFLS can greatly reduce the consumption of computing resources while ensuring the PU accuracy of the solved region.

Index Terms—Large-scale (LS), multibaseline (MB), phase unwrapping (PU), single -baseline (SB), synthetic aperture radar (SAR) interferometry (InSAR).

I. INTRODUCTION

SYNTHETIC aperture radar (SAR) interferometry (InSAR) is a powerful and well-established remote sensing technique that can generate precise digital elevation models (DEMs) and measure terrain surface deformation at high spatial resolution [1]. With the rapid development of InSAR technology, the demand for large-scale data processing has increased. However, current technologies cannot satisfy practical requirements in processing accuracy, efficiency, and hardware resource consumption. Therefore, the task of developing computationally efficient InSAR data processing methods suitable for large-scale data is in great demand.

In practice, the measured data obtained from the InSAR system are interferograms, in which each pixel denotes the wrapped phase within the range of $(-\pi, \pi]$. The wrapped phase is the principal value of the absolute phase that contains elevation and/or deformation information. Phase unwrapping (PU) is the step of obtaining the unwrapped phase from the wrapped phase in the InSAR processing chain. To date, two main types of InSAR technologies have been developed: single-baseline (SB) and multibaseline (MB) InSAR, and both need to address the problem of high algorithm complexity and difficult-to-guarantee accuracy in PU processing. In this article, we use unified symbols to describe SB and MB PU. For pixel s , the mathematical relationship between the wrapped phase $\varphi_r(s)$ and the absolute phase $\psi_r(s)$ can be expressed as

$$\psi_r(s) = \varphi_r(s) + 2\pi \cdot k_r(s) \quad (1)$$

where the integer $k_r(s)$ is the ambiguity number. For an MB InSAR system with R different normal baselines (hereinafter referred to as baselines), there exist R corresponding interferograms. The subscript r denotes the serial number of the interferograms and baselines. For the SB case with only one baseline, r is 1. Fig. 1 illustrates a real InSAR dataset. It is obtained by the Advanced Land Observing Satellite (ALOS) Phased Array type L-band Synthetic Aperture Radar (PALSAR) InSAR system. The observed area is Crowley Lake and its surrounding region in the United States. For InSAR, the water surface can cause extensive noise in the InSAR interferometric dataset. The lake that spans nearly the entire image will pose challenges for InSAR processing. Fig. 1(a) displays the optical image of Crowley Lake (from Google Earth). Fig. 1(b) and (c) shows the interferograms

Manuscript received 25 March 2023; revised 30 April 2023; accepted 9 May 2023. Date of publication 24 May 2023; date of current version 21 June 2023. This work was supported in part by the Fundamental Research Funds for the Central Universities under Grant ZDRC2201, in part by the National Natural Science Foundation of China under Grant 42271456 and Grant 61931025, and in part by the National Science Fund for Distinguished Young Scholars under Grant 61825105. (*Corresponding author: Hanwen Yu.*)

Yang Lan is with the Academy of Advanced Interdisciplinary Research, Xidian University, Xian 710071, China (e-mail: ylan@xidian.edu.cn).

Hanwen Yu is with the School of Resources and Environment, University of Electronic Science and Technology of China (UESTC), Sichuan 611731, China (e-mail: yuhanwenxd@gmail.com).

Mengdao Xing is with the National Key Laboratory of Radar Signal Processing, Xidian University, Xian 710071, China, and also with the Academy of Advanced Interdisciplinary Research, Xidian University, Xian 710071, China (e-mail: xmd@xidian.edu.cn).

Antonio Plaza is with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, Escuela Politécnica, University of Extremadura, 10003 Cceres, Spain (e-mail: aplaza@unex.es).

Digital Object Identifier 10.1109/JSTARS.2023.3279432

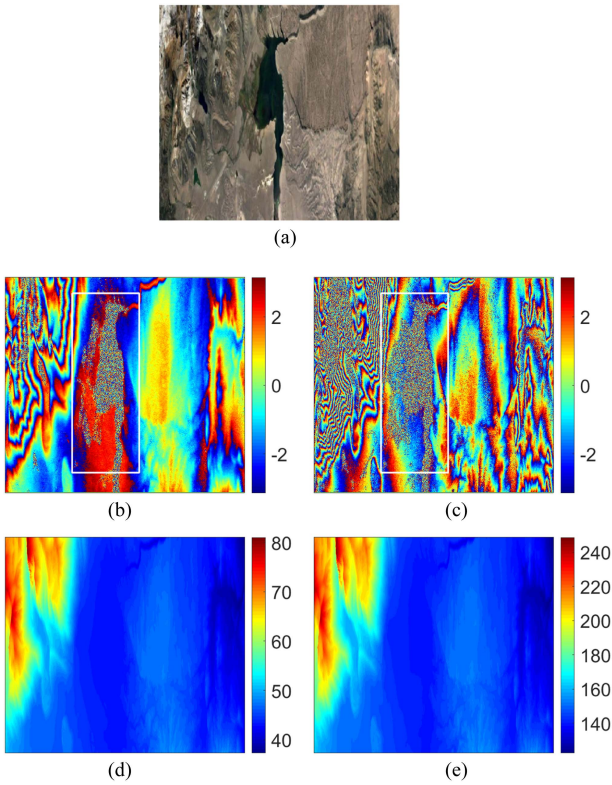


Fig. 1. Crowley Lake. (a) Google Earth image. (b) and (c) Interferograms with baseline length (b) 226.79 m and (c) 646.45 m. (d) and (e) Reference absolute phase of (b) and (c).

with baseline lengths of 226.79 and 646.45 m, respectively. Fig. 1(d) and (e) illustrates the reference absolute phase of Fig. 1(b) and (c). Fig. 1 illustrates the relationship between the actual topography, absolute phase, and interferograms.

Conventional SB PU rely on only a single interferogram to retrieve the absolute phase, which means it needs to solve the two unknowns, $\psi_r(s)$ and $k_r(s)$, from (1), resulting in an ill-posed inverse problem [2]. To determine a unique solution, most SB PU methods have to invoke the phase continuity assumption (also known as the Itoh condition [3], [4]), which posits good spatial continuity within the observed area. Many well-established SB PU methods have been widely and effectively applied to areas satisfying this assumption. In contrast, MB PU is not subject to the phase continuity assumption. MB PU resolves the absolute phase by combining multiple interferograms and exploiting the relationship between multiple baselines. MB PU is more suitable for application on discontinuous terrain, such as valleys and cliffs [5], [6].

With the rapid progress of InSAR technology, the scale of data that can be acquired by InSAR systems has increased dramatically. For example, the TerraSAR-X (an Earth observation synthetic aperture radar satellite that operates in the X-band) and TanDEM-X (TerraSAR-X add-on for digital elevation measurements) satellites launched by Germany in 2007 and 2010, respectively, can obtain high-precision interferograms with a resolution of up to 1 m [7], [8]. In some applications, significantly large and continuous spatial coverage is required, which

may even necessitate processing long-strip interferograms several times larger in size than standard interferograms [9]. This brings great challenges to PU processing. For traditional SB and MB PU methods, when the input data size exceeds a certain limit, the following two issues may arise: inadequate memory and significantly prolonged computation time. These two challenges are more serious in MB PU because MB PU needs to combine multiple interferograms for processing. For small- and medium-sized datasets, the traditional global PU methods have been extensively developed, especially the mathematical models and algorithms of SB PU, which are relatively mature [10], [11], [12]. However, the classical skilled global PU algorithms can hardly handle the computational load and memory consumption caused by excessive data size. This is because the classical skilled PU methods usually have high computational complexity, which requires excessive computing resources. For example, the minimum-cost flow (MCF) is a widely used PU method, but its computational complexity is highly nonlinear. As the data size increases, the memory required also increase significantly [9]. Even improving the performance of hardware, such as increasing the number and speed of CPUs or increasing the amount of memory, cannot meet the rapidly growing computing needs under existing technical conditions. For this reason, researchers soon turned to seek auxiliary strategies to reduce the computing resources required for PU of all large-scale data. The basic idea for the large-scale (LS) PU method is divide and conquer. This approach divides the large-scale interferogram into subinterferograms to solve separately, and then, stitches them together to obtain the overall PU result. In this direction, researchers have successively proposed several effective methods, which significantly reduce memory consumption and computing time. LS PU methods can be divided into two groups [4]: those based on regular subinterferograms and those based on irregular subinterferograms. The methods in the first group divide interferograms into regular subinterferograms [8], [9], [13], [14]. While the interferogram division tasks of these methods are easy to complete, difficulties arise in the stitching step, which usually requires considerable computation to correct inconsistencies between local and global PU results. Methods in the second group divide the input interferograms into subinterferograms according to residues or quality map of the input interferograms, resulting in irregular subinterferograms [7], [15], [16], [17]. Because these methods can strictly or approximately guarantee the consistency of local and global PU results, the stitching task can be easily completed and high-precision PU results obtained. Yu and Lan [6] proposed an MB PU algorithm based on two-stage programming (referred to as TSPA for short), which successfully extended the residue theory in SB PU to MB PU field, developing TSPA-InSAR. This approach effectively paved the way for subsequent development of large-scale MB PU methods. Yu et al. [18] extend the work of [7] and [16] to the MB PU field and achieves credible performance under the TSPA-InSAR framework.

While the prior research has advanced large-scale PU technology, there remains room for improvement in practical applications. In our study, we found that there exist some noise areas in the interferogram, such as Crowley Lake as shown

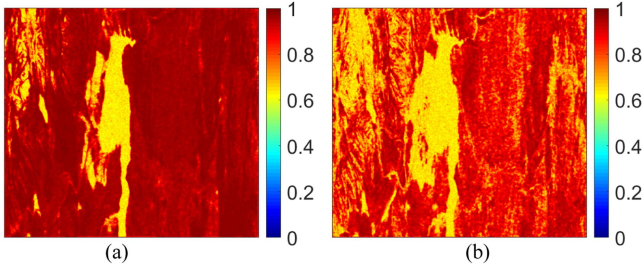


Fig. 2. (a) Pseudocorrelation coefficient of Fig. 1(b). (b) Pseudocorrelation coefficient of Fig. 1(c).

in the area enclosed by white boxes in Fig. 1(b) and (c). The pseudocorrelation coefficient of the lake area is very low [see Fig. 2(a) and (b)], indicating virtually no valuable topographic information in these regions of the interferograms. The salt and pepper noise in Fig. 1(c) is stronger than that in Fig. 1(b), resulting in a lower pseudocorrelation coefficient value in Fig. 2(b) than in Fig. 2(a). Since there is no meaningful information in the noise area, it is intuitive that if meaningless calculations can be discarded, the efficiency of PU processing could be improved. We have undertaken preliminary work based on this idea in [19] and [20]. The research in this article will synthesize and expand on our previous work. Traditional quality-guided PU methods typically utilize quality maps to generate masks, which cover low quality areas and are ignored during the PU process [12]. However, traditional mask generation methods based on quality maps are difficult to ensure the balance of residual polarity within the mask. Yu and Lee [19] point out that if the residues inside the mask are unbalanced, error propagation will occur, causing the PU results outside the mask to be inconsistent with the global optimal PU solutions. Therefore, it is important to locate the appropriate masks that can cover low quality areas to ensure that after removing these areas, the PU results of the remaining areas are consistent with the global optimal PU solution, while significantly reducing PU execution time. Based on the aforementioned analysis, the problem definition of this article is defined as follows.

Problem definition: For large-scale interferogram sets (that have been filtered and flattened), the goal is to locate low quality areas and generate related masks that should meet the following requirements:

- 1) the PU solution of the areas inside and outside the mask should be consistent with the global optimal PU solution;
- 2) the computing resources used to generate the mask should be significantly less than the reduced resource requirements enabled by the mask.

Here, we propose a fast large-scale PU method based on clustering analysis and convex hull (referred to as CCFLS). The method is suitable for both the SB and MB cases. The structure of this article is as follows. Section II provides a review of the PU background. Section III presents the proposed CCFLS method in detail. Section IV verifies the CCFLS method using both simulated and actual InSAR datasets. Finally, Section V concludes this article.

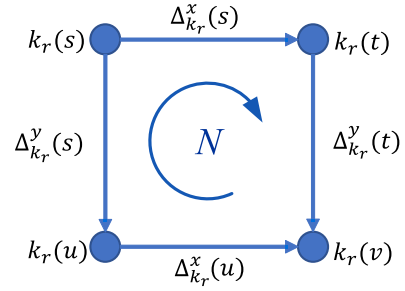


Fig. 3. Auxiliary example of residue.

II. REVIEW OF PU BACKGROUND

For both SB and MB PU, the residual theory is of great importance. In this section, SB and MB residues are reviewed.

A. SB Residue

It is known that SB PU cannot directly obtain the absolute phase through (1). To address this, researchers have proposed the phase continuity assumption, which establishes a relationship between adjacent pixels. Let symbols $s, t, u,$ and v represent four adjacent pixels arranged as shown in Fig. 3. According to the phase continuity assumption, if the gradient of the absolute phase between adjacent pixels does not exceed 2π , the ambiguity number gradients $\Delta_{k_r}^x(s)$ and $\Delta_{k_r}^y(s)$ can be estimated as follows:

$$\Delta_{k_r}^x(s) = \begin{cases} 0 & |\varphi_r(t) - \varphi_r(s)| \leq \pi \\ 1 & \varphi_r(t) - \varphi_r(s) < -\pi \\ -1 & \varphi_r(t) - \varphi_r(s) > \pi \end{cases} \quad (2)$$

$$\Delta_{k_r}^y(s) = \begin{cases} 0 & |\varphi_r(u) - \varphi_r(s)| \leq \pi \\ 1 & \varphi_r(u) - \varphi_r(s) \leq -\pi \\ -1 & \varphi_r(u) - \varphi_r(s) > \pi \end{cases} \quad (3)$$

where $r = 1$, the superscripts x and y denote the horizontal and vertical gradient orientations of adjacent pixels, respectively. Similar to (2) and (3), $\Delta_{k_r}^x(u)$ and $\Delta_{k_r}^y(t)$ can be calculated from $\varphi_r(t), \varphi_r(u),$ and $\varphi_r(v)$. If all gradient estimates are correct, the PU result of the entire interferogram can be obtained by simple integration according to (1). However, when the phase continuity assumption is not satisfied or noise exists, residues will be generated, making PU processing challenging. The residue can be calculated by

$$N = \Delta_{k_r}^x(s) + \Delta_{k_r}^y(t) - \Delta_{k_r}^x(u) - \Delta_{k_r}^y(s). \quad (4)$$

Because of the phase continuity assumption, N can take the value $+1, -1,$ or 0 . If $N = 1$, the residue is termed a positive residue. If $N = -1$, the residue is called a negative residue. As noted in [12], the residues can reflect the situation of phase continuity. Generally speaking, the greater the number of residues, the more difficult it is for PU. Fig. 4 shows the residues obtained from the interferograms in Fig. 1(b) and (c) by using (2)–(4). The total numbers of residues in Fig. 4(a) and (b) are 159 688 and 358 944, respectively. Comparing Figs. 1 and 4, it is apparent that the areas with pepper and salt noise in the interferogram

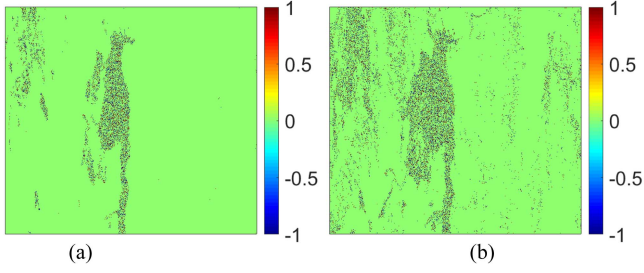


Fig. 4. (a) SB residue of Fig. 1(b). (b) SB residue of Fig. 1(c).

generate many densely distributed residues. In Section III, we will explain the difficulties arising from these densely distributed residues for the PU method.

In practice, there exist many terrains that do not satisfy the phase continuity assumption. Therefore, the SB PU method can hardly provide correct solutions in those cases. To address this issue, MB InSAR was developed. Next, concept of MB residue, proposed in TSPA-InSAR, is reviewed.

B. MB Residue

For the MB InSAR, multiple interferograms are available for PU processing. This article takes the dual-baseline (DB) case as an example to illustrate the relationship between multiple interferograms and the definition of MB residue. Let B_1 and B_2 denote the two baseline lengths of the DB InSAR system. The absolute phases of the interferograms with different baseline lengths have the following relationship [6]:

$$B_2(\varphi_1(s) + 2\pi \cdot k_1(s)) = B_1(\varphi_2(s) + 2\pi \cdot k_2(s)). \quad (5)$$

For the adjacent pixels t and u neighboring pixel s , (5) also applies. By subtracting (5) from the equations for pixels t and u , respectively, we can obtain the gradient relationship between different baselines, yielding (6) and (7)

$$\begin{aligned} & B_2(\Delta_{\varphi_1}^x(s) + 2\pi \cdot \Delta_{k_1}^x(s)) \\ &= B_1(\Delta_{\varphi_2}^x(s) + 2\pi \cdot \Delta_{k_2}^x(s)) \end{aligned} \quad (6)$$

$$\begin{aligned} & B_2(\Delta_{\varphi_1}^y(s) + 2\pi \cdot \Delta_{k_1}^y(s)) \\ &= B_1(\Delta_{\varphi_2}^y(s) + 2\pi \cdot \Delta_{k_2}^y(s)) \end{aligned} \quad (7)$$

where $\Delta_{\varphi_r}^x(s) = \varphi_r(t) - \varphi_r(s)$, $\Delta_{\varphi_r}^y(s) = \varphi_r(u) - \varphi_r(s)$, $\Delta_{k_r}^x(s) = k_r(t) - k_r(s)$, $\Delta_{k_r}^y(s) = k_r(u) - k_r(s)$, $r = 1, 2$. According to Chinese remainder theorem (CRT), (6) and (7) can be solved by the following optimization model:

$$\begin{cases} \arg \min_{\Delta_{k_1}^x(s), \Delta_{k_2}^x(s)} Z^x(s) \\ \text{s.t. } \Delta_{k_1}^x(s), \Delta_{k_2}^x(s) \in \text{integer} \end{cases} \quad (8)$$

$$\begin{cases} \arg \min_{\Delta_{k_1}^y(s), \Delta_{k_2}^y(s)} Z^y(s) \\ \text{s.t. } \Delta_{k_1}^y(s), \Delta_{k_2}^y(s) \in \text{integer} \end{cases} \quad (9)$$

where

$$Z^x(s) = \left| \frac{B_2(\Delta_{\varphi_1}^x(s) + 2\pi \cdot \Delta_{k_1}^x(s)) - B_1(\Delta_{\varphi_2}^x(s) + 2\pi \cdot \Delta_{k_2}^x(s))}{B_1(\Delta_{\varphi_2}^x(s) + 2\pi \cdot \Delta_{k_2}^x(s))} \right| \quad (10)$$

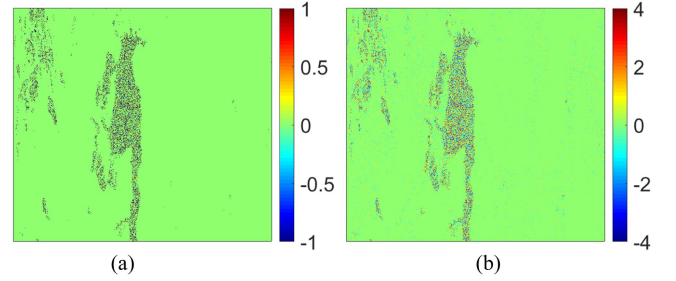


Fig. 5. (a) MB residue of Fig. 1(b). (b) MB residue of Fig. 1(c).

$$Z^y(s) = \left| \frac{B_2(\Delta_{\varphi_1}^y(s) + 2\pi \cdot \Delta_{k_1}^y(s)) - B_1(\Delta_{\varphi_2}^y(s) + 2\pi \cdot \Delta_{k_2}^y(s))}{B_1(\Delta_{\varphi_2}^y(s) + 2\pi \cdot \Delta_{k_2}^y(s))} \right|. \quad (11)$$

Because the decision variables $\Delta_{k_r}^x(s)$ and $\Delta_{k_r}^y(s)$ are integers, for some special combinations (B_1, B_2) , unique $\Delta_{k_r}^x(s)$ and $\Delta_{k_r}^y(s)$ can be found according to (10) and (11). After obtaining the gradient of the ambiguity number, the residue can also be obtained according to (4). It should be noted that, theoretically, the value N of the MB residue can be any integer. A value greater than zero indicates a positive MB residue, while a value less than zero indicates a negative MB residue. Take Fig. 1(b) and (c) as examples of DB interferograms, the MB residues of Fig. 1(b) and (c) are shown in Fig. 5(a) and (b). The number of residues in Fig. 5(a) and (b) are 177 614 and 608 571, and their polarity ranges are $[-1, 1]$ and $[-4, 4]$, respectively. Compared with SB residues, MB residues are more numerous and have a larger polarity range. Our previous research [6] has shown that it is precisely because MB PU allows the absolute phase gradient of adjacent pixels to exceed 2π that it can overcome the limitation of the phase continuity assumption. Yu and Lan [6] extend the residue theory from SB PU to MB PU, thus making the branch-cut theory in the SB PU also valid for MB PU. Next, we will analyze the impact of residues on PU accuracy and efficiency from the perspective of branch cut.

C. Branch Cut

The classic branch-cut PU method is to find the appropriate branch cuts to connect residues of different polarities, such that the sum of the polarities of residues on each branch cut is zero. As long as the path of integration does not cross any branch cut, the integration result (i.e., the PU result) will be independent of the integration path [12]. In other words, the PU result can be readily obtained once the correct branch cuts have been determined. However, seeking the optimal branch cut itself is a difficult task, so researchers have called branch cuts as ghost lines [12], [19]. Large-scale interferometric data will generate a large number of residues, requiring huge computational resources to generate the branch cuts.

In practical applications, a large number of densely distributed residues are generated in low quality areas (such as the lake in Fig. 1) of interferograms. If branch cuts are solved for all residues unselectively, a large amount of computing resources will be wasted. Therefore, it is a reasonable compromise by discarding low quality areas with densely distributed residues. Unlike the

previously mentioned PU method using quality map to generate masks, it is more appropriate to identify areas to discard from the perspective of residues and branch cuts. In the next section, we will design a fast large-scale PU method based on clustering analysis of residues.

III. CONVEX HULL AND CLUSTER-ANALYSIS BASED FAST LARGE-SCALE PU METHOD

Considering the preceding analysis, our goal is to develop a fast PU method for large-scale interferometric data that satisfies the following criteria:

- 1) achieves the same accuracy as traditional global PU methods;
- 2) significantly reduces memory consumption and computational time while maintaining the accuracy described in 1);
- 3) is applicable to both SB and MB interferometric data.

The primary idea of this article is to reduce computational time and memory consumption by masking out the noise areas. Therefore, it is first necessary to determine, which areas in the interferograms should be masked. In order to avoid solving for branch cuts in low quality regions, the masked regions should cover the branch cuts and their connected residues. The smaller the masked area, the better. To identify the area requiring masking, cluster analysis can be employed to categorize all residues according to their spatial distribution. Residues in close proximity should be classified into one group (referred to as a cluster), while residues sufficiently distanced from each other should belong to separate clusters. The residues within a cluster should be enclosed by a connected region (termed a mask). Next, we introduce the concepts of cluster analysis for residue classification and convex hull for mask generation. The method proposed in this article is a fast large-scale PU method based on cluster analysis and convex hull, hence, referred to as CCFLS. It should be noted that there is no fundamental difference between SB residue and MB residue in terms of cluster analysis and convex hull, as SB residue can be considered a special case of MB residue. In other words, SB residue can be viewed as a case that MB residue values are ± 1 and 0.

A. Cluster Analysis

From the perspective of branch cuts, it is necessary to ensure that the integral path cannot pass through branch cuts to guarantee the PU accuracy. This means that residues that should be connected by the same branch cuts cannot be divided into different clusters. In our previous work [18], we proposed the envelope sparse theorem, which ensures consistency between the local optimal solution and the global optimal solution. In this section, we apply it to the cluster analysis of residues.

Definition

Residue-set: The set of residues that should be connected by a branch cut in the global optimal solution.

Balanced-set: The union of multiple residue-sets.

To achieve the global optimal PU solution, identifying the residue-set, defined as the minimum set of residues that can be obtained by correctly dividing residues, is crucial. If multiple

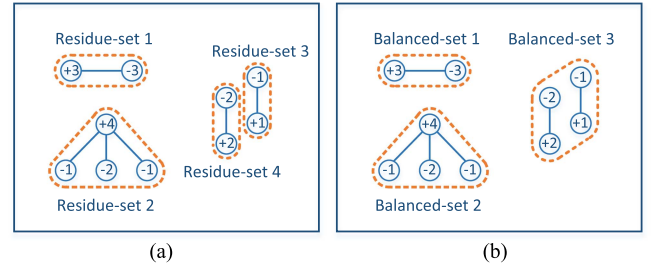


Fig. 6. Auxiliary example of the (a) residue-set, and (b) balance-set and branch-cut.

residues belonging to the same residue-set are divided into different sets, the global optimal solution cannot be achieved by PU. Identifying all residue-sets is equivalent to obtaining the global optimal solution. In general, identifying the residue-sets requires extensive computation (such as L^1 -norm PU method), or cannot be completed in polynomial time (such as L^0 -norm PU method). Identifying the balanced-sets instead of the residue-sets can reduce the computational difficulty.

Fig. 6 illustrates an example of the residue-set and balanced-set of MB residues. The sets in Fig. 6(a) are all residue-sets. In Fig. 6(b), the union of residue-set3 and residue-set4 forms balanced-set3. If we can identify all the balanced-sets and mask the areas occupied by them, we could guarantee that the integral path does not pass through these areas, thereby ensuring that it cannot cross any branch cut. This processing avoids a large amount of computational resource consumption caused by computing branch cuts and ensures that the integration result is independent of the integration path. The obtained PU result will be consistent with the global optimal solution except for the covered area. Next, we provide the residue clustering algorithm to obtain the balanced-set. The method for obtaining the mask is proposed in the next section.

For both SB and MB datasets, we adopt Algorithm I (as shown in Table I) to perform residue clustering. It is necessary to clarify the parameter α selected by the user. When $\alpha = 1$, in order to identify the residue-sets through the envelope sparse theorem, the residues generated from the input interferograms must be sufficiently sparse. However, in practice, it is often difficult to meet this requirement. The smaller the value of α is, the higher the sparsity requirement will be. When the distribution of residues is not sparse enough, too small an α will produce larger clusters, in worst case merging all residues into one cluster. An excessively large cluster will lead to too large an area being discarded, and thus, leading to the failure of PU processing. On the contrary, if α is increased, the envelope sparsity theorem cannot be satisfied, thereby obtaining an approximate solution to the optimal solution. Therefore, α can be regarded as a parameter that helps control the degree of approximation. For further discussion and experience of α , please refer to [18].

B. Convex Hull

After completing the cluster analysis of the residues, the mask for covering the branch cuts can be generated using the residues in each cluster. Owing to the randomness of residue distribution, the contours of residues in a balanced-set are irregular in

TABLE I
RESIDUE CLUSTERING ALGORITHM

Variable definition	
res_i	Clustering object.
β_{i0}	Initial value of clustering radius of res_i .
β_i	Clustering radius of res_i .
d_{ij}	Distance between res_i and res_j .
$cluster_n$	The n th cluster.
r_n	$Max(d_{ij})$ for all res_i and res_j in $cluster_n$, also be called envelope distance.
α	Approximation factor, $\alpha \geq 1$
κ	If $\kappa = 0$, the optimal criterion is L^0 -norm. If $\kappa = 1$, the optimal criterion is L^1 -norm.

Algorithm I	
Step 1	Set $\beta_i = \beta_{i0}$, and set each residue as a cluster. For all res_i and res_j , calculate d_{ij} .
Step 2	For any $res_i \in cluster_n, res_j \in cluster_m$ if $d_{ij} \leq \frac{2-\kappa}{\alpha} \cdot \max(\beta_i, \beta_j)$, merge $cluster_n$ and $cluster_m$.
Step 3	For any $cluster_n$ and $res_i \in cluster_n$, if the total polarity of the cluster is zero, set $\beta_i = r_n$. If the total polarity is not zero, increase β_i to a value greater than r_n .
Step 4	For any $res_i \in cluster_n, res_j \in cluster_m$, if $d_{ij} > \frac{2-\kappa}{\alpha} \cdot \max(\beta_i, \beta_j)$, the clustering ends; otherwise, go to step 2.

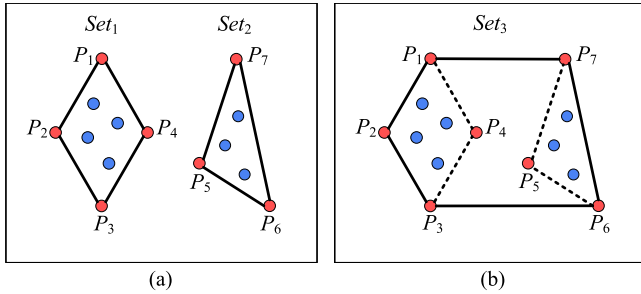


Fig. 7. Auxiliary example of the convex hull. (a) Convex hulls of residue-sets. (b) Convex hull of the union of residue-sets.

shape. In the field of computer science, addressing the irregular shape formed by discrete points poses a challenge. Fortunately, identifying the smallest convex polygon, i.e., convex hull, that contains a given set of points proves relatively straightforward. Efficient convex hull algorithms can be employed to produce masks in this study [21].

Convex hull has a useful property that makes the process of finding the convex hull can be effectively combined with cluster analysis. Next, we use Fig. 7 as an example to illustrate. In Fig. 7(a), there are two sets: set₁ and set₂. The red points in Fig. 7(a) are convex hull points of the sets. The convex hull of set₁ has four convex hull points: P_1, P_2, P_3 , and P_4 . The convex hull of set₂ has three convex hull points: P_5, P_6 , and P_7 . The blue points in Fig. 7(a) are the interior points of their convex hulls. As depicted in Fig. 7(b), when set₁ and set₂ are merged to set₃, the convex hull points of set₃ (i.e., P_1, P_2, P_3, P_6 , and P_7) are obtained from points P_1, P_2, \dots, P_7 , independent of the other interior points. In other words, the convex hull points of set₃ are the convex hull points of the union of the convex hull points of set₁ and set₂. This means that once a point becomes an interior point after participating in the convex hull operation, it

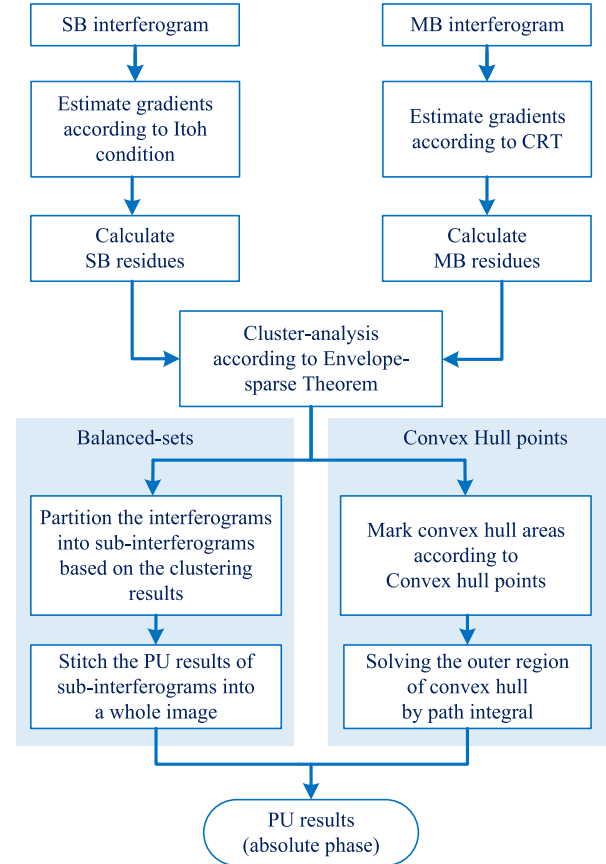


Fig. 8. Flow chart of the large-scale PU method based on the irregular tiling strategy and CCFLS method.

will not participate in subsequent convex hull operations during the merging process. In the clustering process, the clustering of residues always proceeds from near to far, with each clustering step involving the merging of points and clusters. This allows the calculation of convex hulls to be embedded in step 2 of the clustering process, consuming only a small amount of storage and computational resources.

The aforementioned presents the CCFLS method proposed in this article. To clearly illustrates the processing procedure of the CCFLS method, the flow chart of SB/MB PU methods based on an irregular tiling strategy and CCFLS is outlined in Fig. 8. As the flowchart shows, the CCFLS method can be applied to both SB and MB PUs. The only difference lies in the gradient estimation methods. The CCFLS method uses a convex hull to replace branch cuts, which is a reasonable choice in remote sensing applications. In application scenarios requiring real-time performance (such as disaster relief), the CCFLS method is of great importance of improving the speed of PU processing.

C. Analysis of Time and Space Complexities

In the CCFLS method, the primary calculations involve estimating ambiguity number gradients and performing cluster analysis of residues. Assuming there are two baselines, the number of pixels in a single input interferogram is M , and the number of residues in the interferogram r ($r = 1, 2$) is P_r .

The time complexity of the gradient estimation is approximately $O(K_1 \cdot K_2 \cdot M)$, where K_r ($r = 1, 2$) represent the search window ranges for $\Delta_{k_r}^x(i, j)$ and $\Delta_{k_r}^y(i, j)$. For residue clustering of single interferogram, assume the number of clustering iterations is C_r . By selecting an appropriate size parameter of search window in the clustering analysis process, the time complexity required to cluster each residue can be reduced, making the time complexity of the clustering process approximately linear [7]. Therefore, the time complexity of cluster analysis is $O(C_r \cdot P_r)$. To sum up, the time complexity of CCFLS is approximately $O(K_1 \cdot K_2 \cdot M) + O(C_r \cdot P_r)$. For interferograms containing practical and effective information, the number of residues is typically far less than the number of pixels. Consequently, the peak memory consumption of CCFLS is linear with respect to the number of residues [19]. In Section IV, the performance of CCFLS will be tested.

IV. PERFORMANCE ANALYSIS

In this section, we conduct four experiments to evaluate the performance of the CCFLS method. The first experiment uses simulated data to verify the computational performance of the CCFLS method by comparing the time and memory consumption between the global method and the CCFLS method. Given that the CCFLS method discards certain regions based on residues, the second experiment investigates the proportion of discarded regions under varying noise levels. The third and fourth experiments apply the CCFLS method to datasets measured by InSAR from two different types of landforms. Through these two experiments, we validate the effectiveness and practicality of the CCFLS method for measured datasets. In all experiments, the value of the parameter κ is set to 1. The underlying rationale for this selection can be found in our previous study [18].

A. Experiment 1

The goal of this experiment is to examine the reduction in time and memory consumption achieved by the CCFLS method compared to the global PU method for different data scales. To compare the global TSPA method [6] with the CCFLS method, ten simulated datasets ranging in size from 300×300 to 3000×3000 pixels (with increments of 300 rows and 300 columns) were utilized. In each MB InSAR dataset, the reference topography is constant, with a value of zero for simplicity. The approach for generating noise-corrupted interferograms (with an average coherence coefficient of 0.9) is adapted from [22]. For conciseness, the average coherence coefficient is referred to as the correlation coefficient hereafter. Each simulated MB InSAR dataset is comprised of five interferograms with different baseline lengths of 3, 5, 8, 13, and 21 m, respectively. Fig. 9(a) depicts a simulated interferogram of size 300×300 pixels and the baseline length of 3 m. The wrapped phase in Fig. 9(a) does not exhibit the typical stripe texture as shown in Fig. 1(b), due to the reference absolute phase being set to zero, thereby resulting in the ideal wrapped phase to be zero as well. Fig. 9(b) shows an enlarged view of the region inside the white box in Fig. 9(a). For clarity, the remaining figures in Fig. 9 display the results

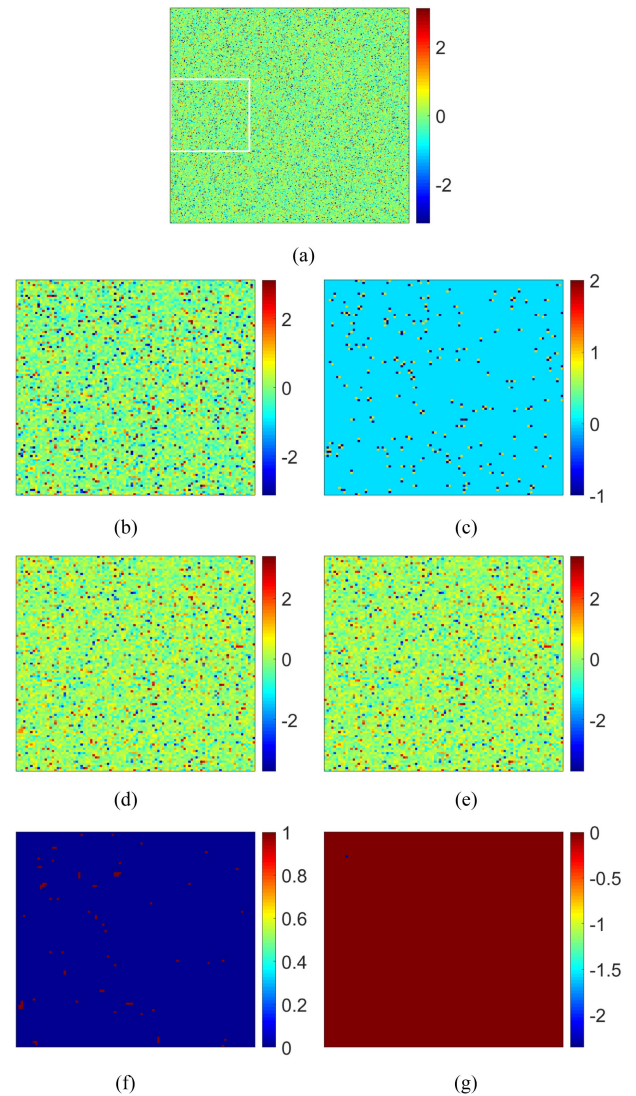


Fig. 9. Comparison of PU results between TSPA and CCFLS. (a) Wrapped phase with a baseline of 3 m and correlation coefficient of 0.9. (b) Wrapped phase within white box of (a). (c) Residues of (b) obtained by TSPA. (d) PU result of (a) within the white box, as determined by TSPA. (e) PU result of (a) within the white box, as determined by CCFLS. (f) Area enclosed by the convex hull. (g) Difference of PU results outside the area of convex hull between TSPA and CCFLS within white box.

within this white box. Fig. 9(c) shows the MB residue of 9(b). Fig. 9(d) and (e) presents the PU results obtained using TSPA and CCFLS, respectively. Fig. 9(f) illustrates the areas within the convex hull (with a value of 1), which are not solved and directly assigned zeros in PU result. The difference between the PU results of TSPA and CCFLS outside the convex hull is shown in Fig. 9(g). Almost all pixels in Fig. 9(g) are zero, with the exception of a few isolated pixels. This demonstrates that, under the clustering criterion of $\kappa = 1$, the convex hull encompasses most branch cuts, resulting in PU results outside the convex hull that are consistent with the global solution of TSPA.

The calculation of all statistical indicators, shown in Table II, is based on the PU result of the interferogram with a baseline length of 3 m. From Fig. 9 and Table II, the following phenomena can be observed.

TABLE II
STATISTICAL INDICATORS RELATED TO CONVEX HULLS

Interferogram size	Residue number	Residue ratio	Convex hull number	Mean residue number per convex hull	Convex hull ratio	Root mean square error	Var
300×300	3306	0.0370	1409	2.3463	0.76%	0.0258	6.6607e-04
600×600	13154	0.0367	5590	2.3531	0.78%	0.0226	5.1216e-04
900×900	29511	0.0365	12604	2.3414	0.77%	0.0231	5.3244e-04
1200×1200	52852	0.0368	22557	2.3430	0.78%	0.0228	5.1807e-04
1500×1500	82829	0.0369	35368	2.3419	0.79%	0.0231	5.3209e-04
1800×1800	119134	0.0368	50870	2.3419	0.79%	0.0234	5.4832e-04
2100×2100	162358	0.0369	69253	2.3444	0.78%	0.0235	5.5047e-04
2400×2400	212239	0.0369	90521	2.3446	0.79%	0.0237	5.3489e-04
2700×2700	268096	0.0368	114392	2.3437	0.78%	0.0239	5.2816e-04
3000×3000	331298	0.0368	141343	2.3439	0.78%	0.0241	5.4446e-04

TABLE III
COMPARISON OF EXECUTION TIME AND PEAK MEMORY CONSUMPTION

Interferogram Size	Times (min)			Peak memory consumption (Mb)		
	TSPA (t_1)	CCFLS (t_2)	$\frac{t_2}{t_1}$	TSPA (m_1)	CCFLS (m_2)	$\frac{m_2}{m_1}$
300×300	0.05	0.02	33.02%	43196	3532	8.18%
600×600	0.26	0.11	43.78%	207784	14116	6.79%
900×900	0.99	0.40	40.68%	527700	31740	6.01%
1200×1200	2.30	1.32	57.32%	988320	56416	5.71%
1500×1500	13.51	3.11	22.99%	1720460	88124	5.12%
1800×1800	27.66	6.35	22.96%	2551776	126888	4.97%
2100×2100	54.32	11.73	21.60%	3676276	172688	4.70%
2400×2400	108.9	20.05	15.22%	4715312	225536	4.78%
2700×2700	389.9	32.16	11.43%	30236376	285428	0.94%
3000×3000	386.3	49.05	4.39%	21295828	352368	0.17%

- 1) Since the correlation coefficient used for simulating noise in all interferograms is the same, the residue ratio (or density) produced by the CCFLS method is approximately the same.
- 2) Both the number of convex hulls and the number of residues within convex hulls are depend on clustering outcomes, which in turn rely on the distribution of residues and the chosen clustering parameters. In this study, the residues in the interferogram are relatively uniformly distributed, leading to a consistent mean residue number per convex hull and convex hull ratio across all interferograms of varying sizes. The convex hull ratio is defined as the proportion of the area enclosed by the convex hulls to the total area of the interferogram.
- 3) The last two columns in Table II display the root mean square (rms) and variance of the difference between the PU results obtained by TSPA and CCFLS in regions outside the convex hull. This demonstrates that the PU results derived from the CCFLS method are in agreement with those obtained from the TSPA method, in the region outside the convex hull.

Table III shows the execution time and peak memory consumption for both the global TSPA and CCFLS methods. The experiments were performed on a workstation equipped with a 12-core CPU running at 3.4 GHz (max turbo frequency up to 5.0 GHz) and 64 GB of memory. The implementation environment is MATLAB 2020a. The TSPA method is implemented using linear programming. As shown in Table III, the CCFLS

method has significantly improved performance over the TSPA method in terms of both execution time and peak memory consumption for all simulated data sizes. Notably, the advantages of the CCFLS method become substantially more pronounced as the data size increases. Fig. 10 illustrates this trend more intuitively. Fig. 10(a) and (b) depicts the curves of execution times and peak memory consumptions. As the interferogram size increases, the execution time and peak memory consumption of the CCFLS method increase slowly. However, the execution time and peak memory consumption of the TSPA method increase significantly with the increase in size. There was a significant fluctuation in the processing time of TSPA, which was caused by the different iterations of the L^1 -norm optimization model solver in processing different random data. Even so, it can be seen that the larger the data size, the more obvious the advantages of the CCFLS method over the TSPA method.

B. Experiment 2

In this experiment, we investigate the number of unsolvable regions generated by the CCFLS method under varying noise levels. Eight correlation coefficients, 0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90, and 0.95, are employed to simulate noise. Each correlation coefficient corresponds to a set of interferograms, containing ten interferograms with a size of 500×500 pixels, and the baseline lengths of 3, 5, 8, 13, 21, 31, 47, 79, 148, and 281 m, respectively. The method for generating the simulated interferograms is consistent with Experiment 1.

TABLE IV
STATISTICS OF THE PU RESULTS OF CCFLS METHOD FOR DATA WITH DIFFERENT NOISE LEVELS

	Correlation coefficient							
	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95
Number of baselines	3	3	3	3	3	3	3	3
Number of residues	65122	59453	53121	45280	36611	27873	17966	9018
Residue proportion (%)	26.1533	23.8766	21.3336	18.1847	14.7032	11.1939	7.2152	3.6217
Convex hull number	8819	10810	11978	12231	11386	9792	7061	3831
Number of residues per convex hull	7.4	5.5	4.4	3.7	3.2	2.8	2.5	2.4
convex hull ratio (%)	99.9688	99.9988	28.5608	17.1652	10.0892	5.3428	2.3228	0.7928
Execution time (min)	0.2371	0.1848	0.1661	0.1403	0.1149	0.0874	0.0643	0.0446
Peak Memory consumption (Mb)	5876	5876	5876	5876	5876	5876	5880	5880

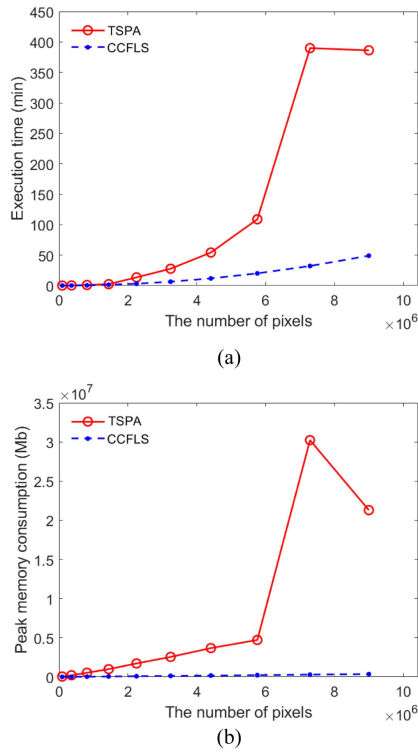


Fig. 10. (a) Execution time. (b) Peak memory consumption corresponding to varying interferogram sizes.

This experiment comprises two parts. In the first part, we use an interferograms set with three baselines (baseline lengths of 3, 5, and 8 m) as an example to explore the influence of different correlation coefficients on the convex hull generated by the CCFLS method. In the second part, we consider the impact of the number of baselines on the convex hull under the same correlation coefficient. The same working platform and environment as the first experiment are utilized. For each set of MB interferometric datasets, the interferograms with a 3-m baseline length are solved, and relevant statistical indicators are calculated.

1) *Influence of Correlation Coefficient on Convex Hull:* In this experiment, we utilized CCFLS to process interferometric datasets with three baselines at varying levels of noise, and the relevant information and statistical results are presented

in Table IV. The following patterns can be observed from the table.

- 1) As the correlation coefficient increases, the processing time of the CCFLS method monotonically decreases. This is because higher correlation coefficients correspond to fewer residues, and the computational time consumed by the clustering analysis in the CCFLS method depends on the number and distribution of residues.
- 2) Notably, the number of convex hulls does not exhibit a monotonic trend with increasing correlation coefficient. When the correlation coefficient is less than 0.75, a lower correlation coefficient corresponds to fewer convex hulls. However, when the correlation coefficient exceeds 0.75, the number of convex hulls decreases with increasing correlation coefficient. This is because lower correlation coefficients typically produce higher density residues, leading to the formation of larger residue clusters and decreasing the number of convex hulls. The most extreme case is that all residuals in the entire image are merged into one residual cluster. As the correlation coefficient increases beyond 0.75, the number of residues declines and their distribution becomes sparser. Both the number of convex hulls and the average number of residues per convex hull decrease with increasing correlation coefficient. Although the number of convex hulls does not monotonically decrease with correlation coefficient, the number of residues and average number of residues per convex hull do decrease monotonically.
- 3) As shown in Table IV, the convex hull ratio decreases as the correlation coefficient increases, indicating that the cost of improving the processing speed of CCFLS decreases as the number of residues decreases. When the correlation coefficient is 0.95, the enclosed area is less than 1% of the total area, having relatively little effect in practical application. When the correlation coefficient drops to 0.8, the enclosed area increases to about 10%. Although this loss cannot be ignored, it is still valuable for remote sensing product users to quickly obtain PU results outside the convex hull. When the correlation coefficient falls below 0.65, the enclosed area exceeds 99%, indicating that the CCFLS method has failed to process the data. In fact, areas with such low correlation coefficients typically do not provide reliable terrain information.

TABLE V
CONVEX HULL RATIO FOR DIFFERENT NOISE LEVELS AND NUMBERS OF BASELINES

Number of baselines	Correlation coefficient							
	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95
2	100.00%	100.00%	99.99%	100.00%	99.99%	99.99%	100.00%	11.13%
3	99.97%	100.00%	28.56%	17.17%	10.09%	5.34%	2.32%	0.79%
4	26.98%	17.08%	10.69%	6.08%	3.51%	1.83%	0.86%	0.41%
5	23.96%	15.34%	9.68%	5.54%	3.37%	1.75%	0.80%	0.41%
6	21.17%	13.68%	8.57%	4.97%	3.04%	1.62%	0.75%	0.39%
7	18.59%	12.16%	7.89%	4.54%	2.75%	1.52%	0.72%	0.41%
8	16.96%	11.38%	7.40%	4.32%	2.64%	1.48%	0.72%	0.40%
9	16.57%	11.26%	7.16%	4.21%	2.60%	1.49%	0.72%	0.41%
10	16.51%	11.12%	7.19%	4.28%	2.58%	1.47%	0.72%	0.40%

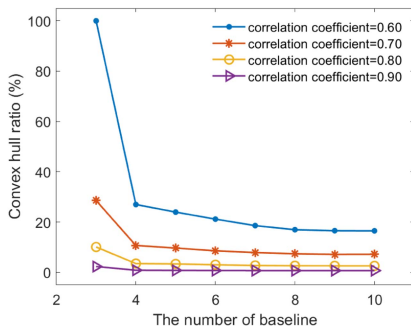


Fig. 11. Convex hull Ratio versus number of baselines.

4) From the residue proportion perspective, the CCFLS method can successfully process the data when the residue proportion is around 20% or less. Our previous study [5] found that increasing the number of baselines can reduce the number of residues. Therefore, the second part of this experiment investigated the effect of the number of baselines on the convex hull area.

2) *Effect of Baseline Number on Convex Hulls:* In this experiment, we utilized the CCFLS method to process interferograms with varying baseline numbers ranging from 2 to 10 and noise correlation coefficients from 0.60 to 0.95. The convex hull ratios are listed in Table V. The data in Table V are analyzed as follows.

- 1) For interferograms with only two baselines, when the correlation coefficient is lower than 0.90, the CCFLS method can hardly resolve them successfully. When processing actual interferogram datasets, the CCFLS method is more likely to determine the area with a correlation coefficient lower than 0.9 as the envelope area instead of resolving it.
- 2) For interferograms with more than three baselines, a significant improvement was observed compared to two baselines. The interferograms with three baselines can successfully resolve the areas with correlation coefficients greater than 0.70. Meanwhile, the interferograms with four or more baselines can successfully resolve the areas with correlation coefficients greater than 0.6.
- 3) By examining each column in Table V, it can be found that the reduction of the convex hull ratio is nonlinear with an increase in the number of baselines. The curves formed by each column's data in Table V are depicted in Fig. 11. For clarity and simplicity, four curves with correlation

TABLE VI
MAJOR INTERFEROMETRIC PARAMETERS OF EMPLOYED REAL DATASET OF TANDEM-X

Orbit altitude	Incidence angle	Wavelength
514.8 km	36.6°	0.032m

coefficients of 0.6, 0.7, 0.8, and 0.9 were chosen to be illustrated. As shown in Fig. 11, when the number of baselines exceeds four, the change of the convex hull ratios tends to be gentler. Moreover, when the number of baselines exceeds eight, further increasing the number of baselines can barely reduce the convex hull ratio.

Based on the aforementioned analysis, it is noteworthy that the statistical data presented in Table V can serve as a useful reference for users requiring the CCFLS method to resolve the PU problem. Table V provides a guidance on determining the number of baselines required to effectively solve the PU problem while balancing performance and computing resource consumption.

C. Experiment 3

This experiment used the single-pass TANDEM-X dual-baseline dataset to validate the accuracy and computational performance of the CCFLS method compared to the TSPA method. The interferograms size are 1001×1001 pixels. The observed area is located in Weinan of Shaanxi province, China, features a typical mountainous terrain with complex topography. Fig. 12(a) shows the reference unwrapped phase of the interferogram shown in Fig. 12(c). Fig. 12(b) and (c) depicts the interferograms with baseline lengths of 130.62 and 370.45 m, respectively. The major interferometric parameters of Fig. 12(b) and (c) are listed in Table VI. We can observe that, due to the complex terrain, the fringes in Fig. 12(c) are thin, dense, and irregular, which poses challenges for PU. Fig. 12(d) and (e) shows the corresponding pseudocorrelation coefficients for Fig. 12(b) and (c), respectively. The PU results and PU misfits are displayed in Fig. 13. Due to the regions enclosed by the convex hulls not being solved by CCFLS, in the PU results, these regions will be artificially assigned a constant value for observation.

Fig. 13(a) and (b) depicts the PU results obtained by the CCFLS method for Fig. 12(b) and (c). In Fig. 13(b), there are some small regions, such as the red region marked by the

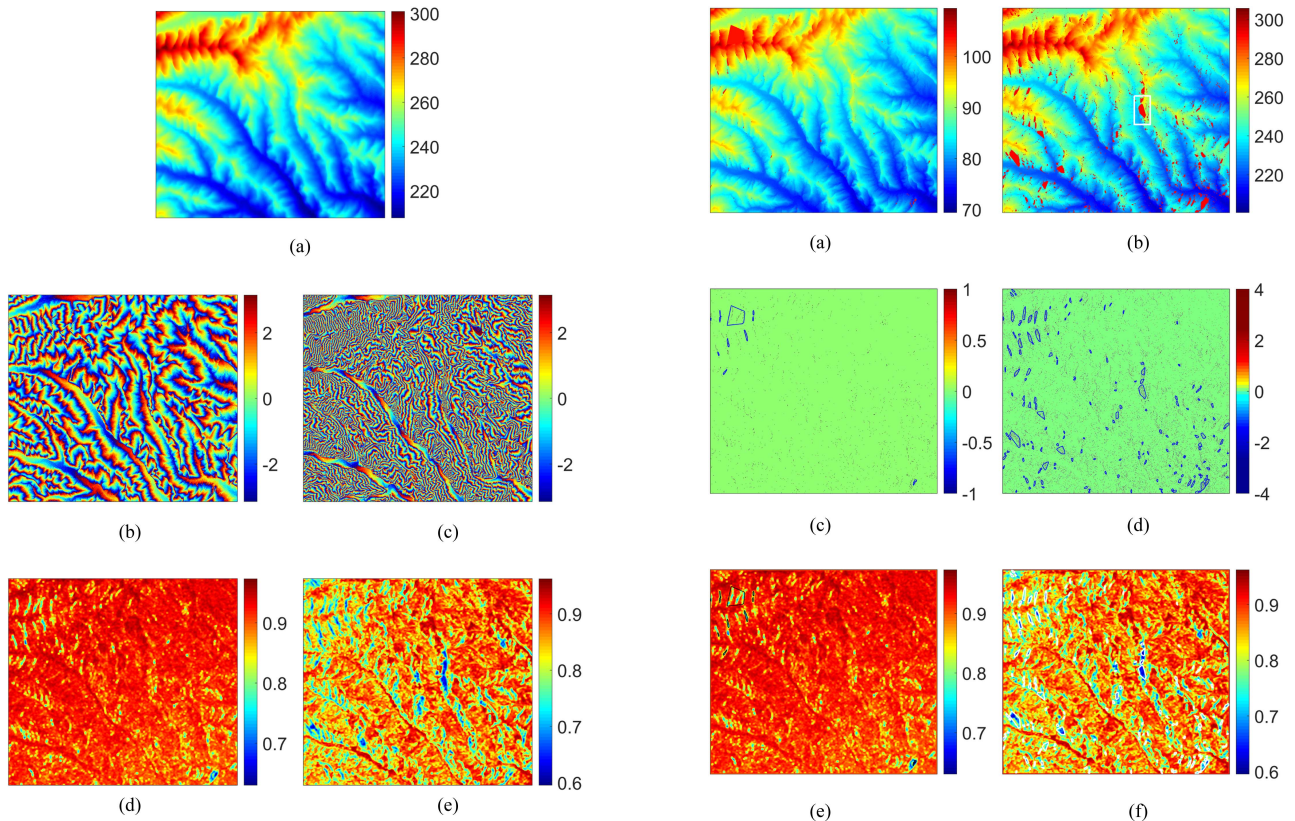


Fig. 12. (a) Reference absolute phase of (c) obtained from SRTM DEM. (b) TanDEM-X interferogram with short baseline. (c) TanDEM-X interferogram with long baseline. (d) Pseudocorrelation coefficient of (b). (e) Pseudocorrelation coefficient of (c).

white square, enclosed by the convex hull that have not been solved by the CCFLS method. In order to reveal the relationship between the convex hull and residue distribution, the boundary of the convex hull area is marked on the residue maps shown in Fig. 13(c) and (d). For clarity, only convex hulls containing more than 20 residues are marked. These convex hulls also are marked in the pseudocorrelation coefficients shown in Fig. 13(e) and (f). As can be clearly seen in the figure, the area where the pseudocorrelation coefficient is below 0.65 has almost been enclosed by the convex hulls constructed by CCFLS.

Fig. 13(g) and (h) shows the PU results obtained by the TSPA method for Fig. 12(b) and (c), respectively. Fig. 13(i) depicts the difference between Fig. 13(a) and (g), while Fig. 13(j) shows the difference between Fig. 13(b) and (h). The PU results of TSPA and CCFLS are almost identical, as observed from Fig. 13(i) and (j), except for the areas enclosed by convex hulls. Based on the results shown in Fig. 13, it can be concluded that the CCFLS method can accurately identify areas with poor image quality and enclose them with convex hulls. From the perspective of branch cuts, the convex hull generated by CCFLS can effectively avoid integration path passing through any branch cuts, enabling the PU result with the same accuracy as TSPA. The execution time and peak memory consumption of these two methods are listed in Table VII. The running time of TSPA method for PU is 1 136 s, while that of CCFLS method is only 95 s, which is only 8.36% of the TSPA method. More savings are achieved in terms of

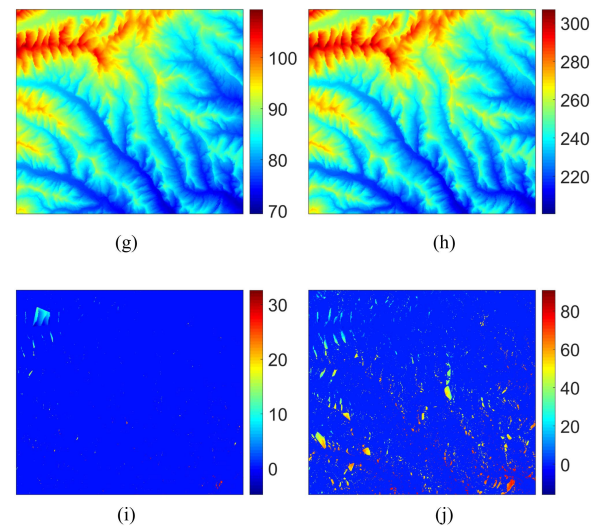


Fig. 13. Verification of the consistency of CCFLS and TSPA methods. (a) and (b) PU result of Fig. 12(b) and (c) obtained by CCFLS. (c) and (d) Residue maps of Figs. 12(b) and (c) marked by convex hull (the number of residues in convex hull more than 20). (e) and (f) Pseudocorrelation coefficient of Fig. 12(b) and (c) outlined by convex hull (the number of residues in convex hull more than 20). (g) and (h) PU result of Fig. 12(b) and (c) obtained by TSPA. (i) Difference between (a) and (g). (j) Difference between (b) and (h).

peak memory consumption. The peak memory consumption of the TSPA method is 2 006 Mb, whereas that of CCFLS method is 15 Mb, less than 1% of TSPA method. These statistics were consistent with the conclusion of Experiment 1. This experiment verifies that CCLFS can indeed effectively reduce the consumption of computing resources.

TABLE VII
COMPARISON OF EXECUTION TIME AND PEAK MEMORY CONSUMPTION

PU method	Execution time (sec)	Peak memory consumption (Mb)
CCFLS	95	15
TSPA	1136	2006
Reduced proportion	8.36%	0.75%

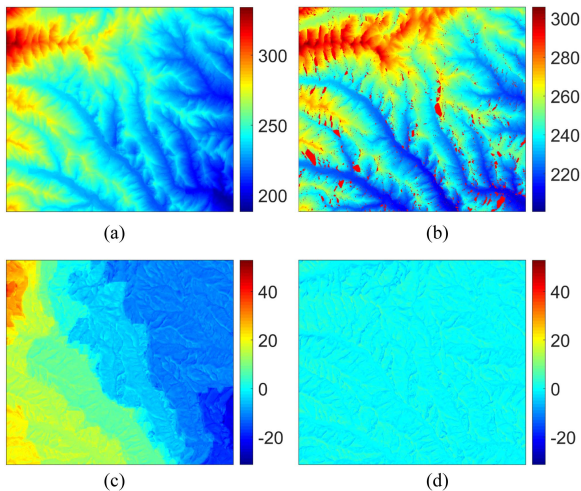


Fig. 14. Performance comparison between MCF SB PU method and CCFLS MB PU method. (a) PU result of Fig. 12(c) obtained by the MCF SB PU method. (b) PU result of Fig. 12(c) obtained by the CCFLS MB PU method. (c) Difference between (a) and Fig. 12(a). (d) Difference between (b) and Fig. 12(a).

Finally, we compared CCFLS with the classic MCF SB PU method to verify the PU accuracy of CCFLS. The interferogram used is shown in Fig. 12(c). Fig. 14(a) depicts the PU result of Fig. 12(c) obtained by the SB MCF method, and Fig. 14(b) shows the PU result of Fig. 12(c) obtained by CCFLS. Fig. 14(c) and (d) shows the differences between PU results [i.e., Fig. 14(a) and (b)] and the reference absolute phase [i.e., Fig. 12(a)], respectively. To facilitate comparison, the two error images use the same color bar range. Fig. 14(c) and (d) intuitively demonstrates that the PU result error of the SB MCF method is much greater than that of the CCFLS. The rms of Fig. 14(c) is 11.8249, while the rms of Fig. 14(d) is 1.5989, indicating that the CCFLS method proposed in this article has significant advantages in solving the complex terrain.

D. Experiment 4

The final experiment aims to investigate whether the CCFLS method can effectively process geomorphic data with a large area of noise. The observation area of the InSAR data employed in this experiment is the same as in Fig. 1, the Crowley Lake area, and its Google Earth image is shown in Fig. 1(a). Based on the trend of the convex hull ratio shown in Fig. 11, six interferograms were selected for this experiment. The interferograms are shown in Fig. 15(a)–(f), which have baseline lengths of 646.45, 957.72, 1862.30, 2018.68, 2671.79, and 3600.09 m, respectively. The major radar parameters are given in Table VIII.

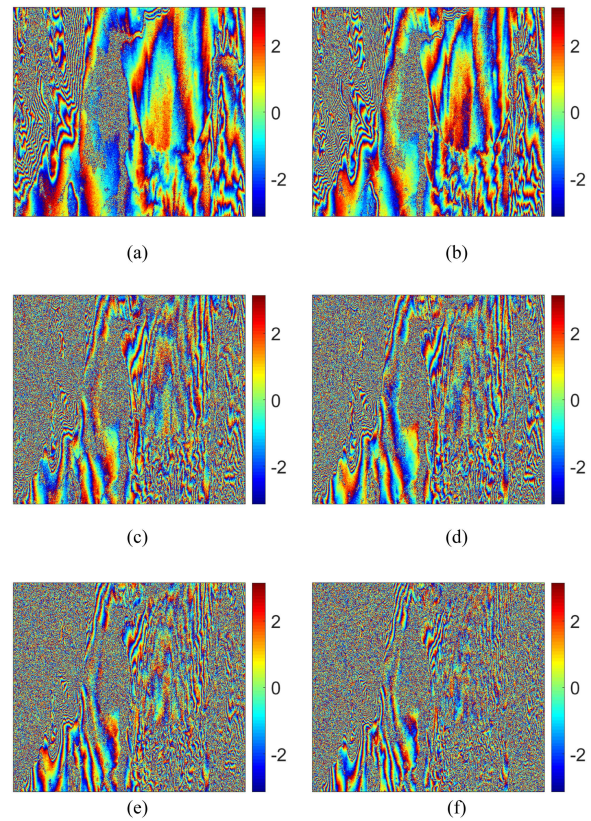


Fig. 15. Six interferograms with different baseline length (a) 646.45 m, (b) 957.72 m, (c) 1862.30 m, (d) 2018.68 m, (e) 2671.79 m, and (f) 3600.09 m.

TABLE VIII
MAJOR PARAMETERS OF EMPLOYED REAL DATASET

Orbit	titude	Incidence angle	Wavelength
7069.5km		39.39	0.236m
Latitude	Longitude	Size	
37.6135	-118.7485	2628 × 2480	

As shown in Fig. 1(a), the Crowley lake is located at the center of the observed area. The presence of the lake results in a large noise region in the interferogram. Such a large noise area poses substantial challenges for cluster analysis. In order to prevent all the residues in the interferogram from merging into one cluster, the value of α is set to 3000. In this experiment, we compared the accuracy of the CCFLS method with that of the SB MCF PU method, because the latter method has excellent noise robustness.

Fig. 16(a) and (b) shows the PU results of Fig. 15(a) obtained by the CCFLS and SB MCF PU methods, respectively. Fig. 16(c) depicts the MB residue map obtained by CCFLS. By comparing Fig. 16(a) and (c), it can be observed that the areas with dense residues in Fig. 16(c) correspond to the areas enclosed by the convex hulls in Fig. 16(a), especially the location of Crowley Lake. Although there is the high density and wide distribution of the residues on the left half of Fig. 16(c), due to the appropriate setting of the α value, the residues did not all merge into one

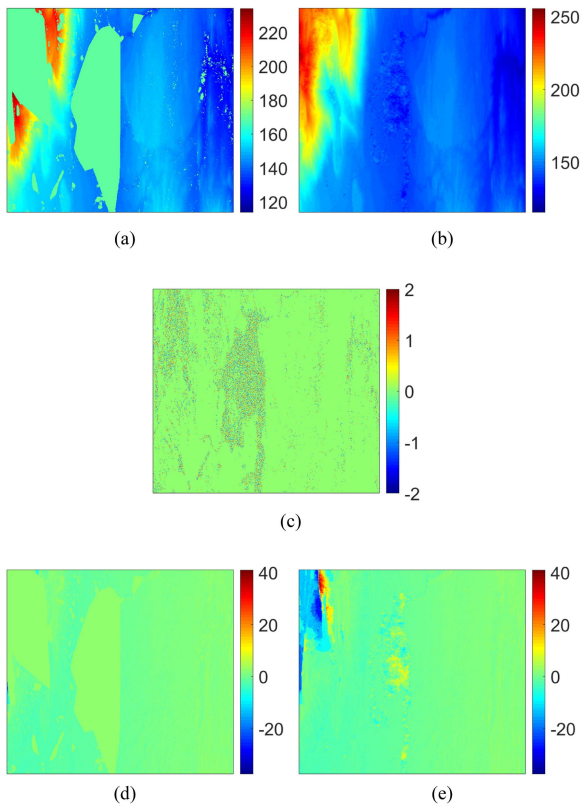


Fig. 16. Comparison between CCFLS and SB MCF PU methods for precision of the PU results. (a) PU result of Fig. 15(a) obtained by CCFLS. (b) PU result of Fig. 15(a) obtained by the SB MCF PU method. (c) MB residues of Fig. 15(a) obtained by CCFLS. (d) and (e) Difference between (a) and (b) and Fig. 1(e).

cluster. Therefore, the CCFLS successfully solved this dataset. Fig. 16(d) illustrates the error between the reference absolute phase in Figs. 1(e) and 16(a), while Fig. 16(e) shows the error between the Figs. 1(e) and 16(b). The rms of Fig. 16(d) and (e) are 1.3927 and 1.4582 (after removing the area in convex hulls), respectively. This shows that the PU accuracy of the CCFLS method is better than that of the SB MCF PU method.

V. CONCLUSION

In the field of remote sensing applications, especially in geological disaster rescue, there is an urgent need to accelerate the processing of large-scale InSAR data. The fast large-scale PU method based on cluster analysis and convex hull (CCFLS) proposed in this article can significantly reduce both processing time and peak memory consumption while ensuring the PU accuracy. For areas enclosed by convex hulls, PU methods with filtering functions, such as minimum infinity-norm-based PU method (MIN) [23] and CA-based MB PU and filtering algorithm (MBPUF) [24], could be used for further processing according to user needs to obtain more precise results. Emerging deep learning techniques have already been used in InSAR field and have demonstrated promising potential in improving accuracy, speed, and large-scale data processing [25], [26], [27], [28], [29], [30]. In future studies, deep learning could be considered for mask generation, which may be a potential approach.

In summary, the CCFLS method is a promising approach for large-scale InSAR data processing in geological disaster rescue and other remote sensing applications.

REFERENCES

- [1] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 1, pp. 6–43, Mar. 2013.
- [2] R. Bamler and P. Hartl, "Synthetic aperture radar interferometry," *Inverse Problems*, vol. 14, no. 4, pp. R1–R54, 1998.
- [3] K. Itoh, "Analysis of the phase unwrapping problem," *Appl. Opt.*, vol. 21, no. 14, pp. 2470–2470, 1982.
- [4] H. Yu, Y. Lan, Z. Yuan, J. Xu, and H. Lee, "Phase unwrapping in InSAR: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 7, no. 1, pp. 40–58, Mar. 2019.
- [5] Y. Lan, H. Yu, Z. Yuan, and M. Xing, "Comparative study of DEM reconstruction accuracy between single- and multibaseline InSAR phase unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5220411, doi: 10.1109/TGRS.2022.3140327.
- [6] H. Yu and Y. Lan, "Robust two-dimensional phase unwrapping for multibaseline SAR interferograms: A two-stage programming approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 9, pp. 5217–5225, Sep. 2016.
- [7] H. Yu, Y. Lan, J. Xu, D. An, and H. Lee, "Large-scale L^0 -norm and L^1 -norm two-dimensional phase unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 8, pp. 4712–4728, Aug. 2017.
- [8] Q. Huang, H. Zhou, S. Dong, and S. Xu, "Parallel branch-cut algorithm based on simulated annealing for large-scale phase unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 7, pp. 3833–3846, Jul. 2015.
- [9] K. Zhang et al., "Phase unwrapping for very large interferometric data sets," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 4048–4061, Oct. 2011.
- [10] D. C. Ghiglia and L. A. Romero, "Minimum L^p -norm two-dimensional phase unwrapping," *J. Opt. Soc. Amer. A*, vol. 13, no. 10, pp. 1999–2013, 1996.
- [11] C. W. Chen and H. A. Zebker, "Network approaches to two-dimensional phase unwrapping: Intractability and two new algorithms," *J. Opt. Soc. Amer. A*, vol. 17, no. 3, pp. 401–414, Mar. 2001.
- [12] D. C. Ghiglia and M. D. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software*. New York, NY, USA: Wiley, 1998.
- [13] C. W. Chen and H. A. Zebker, "Phase unwrapping for large SAR interferograms: Statistical segmentation and generalized network models," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 8, pp. 1709–1719, Aug. 2002.
- [14] GAMMA Remote Sensing AG, Interferometric SAR processing: GAMMA remote sensing, 2018. [Online]. Available: <https://www.gamma-rs.ch/>
- [15] H. Yu, Z. Li, and Z. Bao, "Residues cluster-based segmentation and outlier-detection method for large-scale phase unwrapping," *IEEE Trans. Image Process.*, vol. 20, no. 10, pp. 2865–2875, Oct. 2011.
- [16] H. Yu, M. Xing, and Z. Bao, "A fast phase unwrapping method for large-scale interferograms," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 7, pp. 4240–4248, Jul. 2013.
- [17] J. Xu, D. An, X. Huang, and G. Wang, "Phase unwrapping for large-scale P-band UWB SAR interferometry," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 10, pp. 2120–2124, Oct. 2015.
- [18] H. Yu, Y. Zhou, S. S. Ivey, and Y. Lan, "Large-scale multibaseline phase unwrapping: Interferogram segmentation based on multibaseline envelope-sparsity theorem," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 11, pp. 9308–9322, Nov. 2019, doi: 10.1109/TGRS.2019.2926096.
- [19] H. Yu and H. Lee, "A convex hull algorithm based fast large-scale two-dimensional phase unwrapping method," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Fort Worth, TX, USA, 2017, pp. 3824–3827, doi: 10.1109/IGARSS.2017.8127751.
- [20] Y. Lan, H. Yu, and M. Xing, "A convex hull and cluster-analysis based fast large-scale phase unwrapping method for multibaseline SAR interferograms," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Yokohama, Japan, 2019, pp. 1765–1768, doi: 10.1109/IGARSS.2019.8900492.
- [21] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.
- [22] J.-S. Lee, K. W. Hoppel, S. A. Mango, and A. R. Miller, "Intensity and phase statistics of multilook polarimetric and interferometric SAR imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 5, pp. 1017–1028, Sep. 1994.

- [23] H. Yu, Y. Lan, H. Lee, and N. Cao, "2-D phase unwrapping using minimum infinity-norm," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 12, pp. 1887–1891, Dec. 2018, doi: [10.1109/LGRS.2018.2865601](https://doi.org/10.1109/LGRS.2018.2865601).
- [24] Z. Yuan, Z. Lu, L. Chen, and X. Xing, "A closed-form robust cluster-analysis-based multibaseline InSAR phase unwrapping and filtering algorithm with optimal baseline combination analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4251–4262, Jun. 2020.
- [25] L. Zhou, H. Yu, and Y. Lan, "Deep convolutional neural network-based robust phase gradient estimation for two-dimensional phase unwrapping using SAR interferograms," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4653–4665, Jul. 2020, doi: [10.1109/TGRS.2020.2965918](https://doi.org/10.1109/TGRS.2020.2965918).
- [26] L. Zhou, H. Yu, Y. Lan, and M. Xing, "Artificial intelligence in interferometric synthetic aperture radar phase unwrapping: A review," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 2, pp. 10–28, Jun. 2021, doi: [10.1109/MGRS.2021.3065811](https://doi.org/10.1109/MGRS.2021.3065811).
- [27] L. Zhou, H. Yu, Y. Lan, S. Gong, and M. Xing, "CANet: An unsupervised deep convolutional neural network for efficient cluster-analysis-based multibaseline InSAR phase unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5212315, doi: [10.1109/TGRS.2021.3110518](https://doi.org/10.1109/TGRS.2021.3110518).
- [28] L. Zhou, H. Yu, Y. Lan, and M. Xing, "Deep learning-based branch-cut method for InSAR two-dimensional phase unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5209615, doi: [10.1109/TGRS.2021.3099997](https://doi.org/10.1109/TGRS.2021.3099997).
- [29] L. Zhou, H. Yu, Y. Wang, Y. Lan, and M. Xing, "PG-BCNet : A neural network combined with the PGNet and BCNet for 2-D InSAR phase unwrapping," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 2470–2473, doi: [10.1109/IGARSS46834.2022.9883237](https://doi.org/10.1109/IGARSS46834.2022.9883237).
- [30] K. Yang, Z. Yuan, X. Xing, and L. Chen, "Deep learning-based mask-cut method for InSAR phase-unwrapping," *IEEE J. Miniaturization Air Space Syst.*, vol. 4, no. 2, pp. 221–230, Jun. 2023, 2023, doi: [10.1109/JMASS.2023.3258379](https://doi.org/10.1109/JMASS.2023.3258379).



Yang Lan (Member, IEEE) received the B.S. degree in communication engineering, the M.S. degree in information and communication engineering, and the Ph.D. degree in signal and information processing from Xidian University, Xian, China, in 1997, 2005, and 2020, respectively.

She is currently a Lecturer with the Academy of Advanced Interdisciplinary Research, Xidian University. She has authored and co-authored more than ten research articles. Her research interests include phase unwrapping and interferometric synthetic aperture

radar signal processing.



Hanwen Yu (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electronic engineering from Xidian University, Xi'an, China, in 2007 and 2012, respectively.

He was a Postdoctoral Fellow with the Department of Civil and Environmental Engineering and the National Center for Airborne Laser Mapping, University of Houston. He is currently a Full Professor with the School of Resources and Environment, University of Electronic Science and Technology of China, Chengdu, China, and an Adjunct Full Professor with

the Academy of Advanced Interdisciplinary Research, Xidian University, and the Department of Engineering, University of Napoli, Parthenope, Italy. He has authored more than 60 scientific papers. His research interests include InSAR, and this work has led to new insights into the world-wide deformation monitoring and topographic mapping.

Dr. Yu has reviewed more than 280 manuscripts for more than 20 different journals. He has given scientific presentation about "Advanced techniques in InSAR Phase Unwrapping" invited by the IEEE Geoscience and Remote Sensing Society (IEEE GRSS) Webinar in 2021. He has been involved in IEEE in general and IEEE GRSS in particular. He is elected as 2019 IEEE TRANSACTIONS GEOSCIENCE AND REMOTE SENSING Best Reviewer. He is also elected as IEEE GRSS AdCom member beginning in 2023. He is the Editor-in-Chief for IEEE JOURNAL ON MINIATURIZATION FOR AIR AND SPACE SYSTEMS, the Topical Associate Editor for IEEE TRANSACTIONS GEOSCIENCE AND REMOTE SENSING, and the Associate Editor for IEEE GEOSCIENCE AND REMOTE SENSING MAGAZINE. He was the recipient of several awards and honors from IEEE GRSS, including the 2021 Transactions Prize Paper Award. He is the Technical Program Committee Member and Session Chair of 2022 International Geoscience and Remote Sensing Symposium and the PI of the IEEE GRSS 50%-funding project West China Initiative.



Mengdao Xing (Fellow, IEEE) received the B.S. and Ph.D. degrees in electrical engineering from Xidian University, Xian, China, in 1997 and 2002, respectively.

He is currently a Professor with the National Laboratory of Radar Signal Processing, Xidian University. He holds the appointment of the Associate Dean with the Academy of Advanced Interdisciplinary Research, Xidian University. He has authored or co-authored more than 200 refereed scientific journal articles and two books about synthetic aperture radar (SAR) signal processing. His research has been supported by various funding programs, such as National Science Fund for Distinguished Young Scholars. His current research interests include synthetic aperture radar (SAR), inversed SAR, sparse signal processing, and microwave remote sensing.

Dr. Xing serves as an Associate Editor for radar remote sensing of the IEEE TRANSACTIONS GEOSCIENCE AND REMOTE SENSING. The total citation times of his research are greater than 8000. He was rated as Most Cited Chinese Researchers by Elsevier. He has achieved more than 40 authorized China patents.



Antonio Plaza (Fellow, IEEE) received the M.Sc. and Ph.D. degrees in computer engineering from the Department of Technology of Computers and Communications, University of Extremadura, Cceres, Spain, in 1999 and 2002, respectively.

He is a Full Professor and the Head with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, University of Extremadura, Badajoz, Spain. He has authored more than 800 publications in this field, including 400 JCR journal, 25 book chapters, and

330 peer-reviewed conference proceeding papers. He has guest edited 17 special issues on hyperspectral remote sensing for different journals. His main research interests include hyperspectral data processing and parallel computing of remote sensing data.

Prof. Plaza was awarded the fellowship of the IEEE for contributions to hyperspectral data processing and parallel computing of Earth observation data and a Member of Academia Europaea, The Academy of Europe. He was the recipient of the recognition of Best Reviewers of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS (in 2009) and the recipient of the recognition of Best Reviewers of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING (in 2010), for which he served as an Associate Editor in 2007–2012. He is also an Associate Editor for IEEE ACCESS (receiving the recognition of Outstanding Associate Editor for the journal in 2017), and was also a member of the Editorial Board of the IEEE GEOSCIENCE AND REMOTE SENSING NEWSLETTER (2011–2012) and the IEEE GEOSCIENCE AND REMOTE SENSING MAGAZINE (2013). He was also a member of the steering committee of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (JSTARS). He was the recipient of the Best Column Award of the IEEE SIGNAL PROCESSING MAGAZINE in 2015, the 2013 Best Paper Award of the JSTARS journal, and the most highly cited paper (2005–2010) in the *Journal of Parallel and Distributed Computing*. He was the recipient of best paper awards at the IEEE Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, the IEEE International Conference on Space Technology, and the IEEE Symposium on Signal Processing and Information Technology. He served as the Director of Education Activities for the IEEE Geoscience and Remote Sensing Society (GRSS) in 2011–2012, and as the President of the Spanish Chapter of IEEE GRSS in 2012–2016. He is currently serving as the Chair of the Publications Awards Committee of IEEE GRSS and as a Vice Chair of the Fellow Evaluations Committee of the IEEE GRSS. He has reviewed more than 500 manuscripts for more than 50 different journals. He served as the Editor-in-Chief for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING journal for five years (2013–2017) and as the Editor-in-Chief of the IEEE JOURNAL ON MINIATURIZATION FOR AIR AND SPACE SYSTEMS (2019–2020). He has been included in the 2018, 2019, 2020, 2021, and 2022 Highly Cited Researchers List (Clarivate Analytics). Additional information: <http://www.umbc.edu/rssi/pl/people/aplaza>