

# Interferometric Phase Image Estimation via Sparse Coding in the Complex Domain

Hao Hongxing, José M. Bioucas-Dias, *Member, IEEE*, and Vladimir Katkovnik

**Abstract**—This paper addresses interferometric phase image estimation, i.e., the estimation of phase modulo- $2\pi$  images from sinusoidal  $2\pi$ -periodic and noisy observations. These degradation mechanisms make interferometric phase image estimation a quite challenging problem. We tackle this challenge by reformulating the true estimation problem as a sparse regression, often termed sparse coding, in the complex domain. Following the standard procedure in patch-based image restoration, the image is partitioned into small overlapping square patches, and the vector corresponding to each patch is modeled as a sparse linear combination of vectors, termed the atoms, taken from a set called dictionary. Aiming at optimal sparse representations, and thus at optimal noise removing capabilities, the dictionary is learned from the data that it represents via matrix factorization with sparsity constraints on the code (i.e., the regression coefficients) enforced by the  $\ell_1$  norm. The effectiveness of the new sparse-coding-based approach to interferometric phase estimation, termed the SpInPHASE, is illustrated in a series of experiments with simulated and real data where it outperforms the state-of-the-art.

**Index Terms**—Dictionary learning (DL), image similarity, interferometric phase estimation, online learning, phase estimation, phase unwrapping, sparse regression.

## I. INTRODUCTION

**I**N PHASE imaging applications, a physical quantity of interest is coded in an image of phase. Phase estimation plays, therefore, a central role in these imaging systems. For instance, in interferometric synthetic aperture radar (SAR)/sonar (InSAR/InSAS), the topography of a given surface is inferred from phase differences measured by multiple radar/sonar antennas; in magnetic resonance imaging, the phase is used, namely, to determine magnetic field deviation maps, which

are used to correct echo-planar image geometric distortions, to determine chemical-shift-based temperature, to identify veins in the tissues, and to segment water from fat; in optical interferometry, the phase is used to determine shape, deformation, and vibration of objects; in 3-D from structured light, a scene is illuminated with spatial sinusoidal light patterns, and its 3-D shape is inferred from the intensity of the scattered light; in X-ray phase-contrast imaging, the differential X-ray phase shift induced by the refractive index of an object is inferred from X-ray tomographic projections; and in phase retrieval, a noninterferometric technique, the phase is determined from intensity measurements.

The measurements in the phase imaging systems are noisy and usually  $2\pi$ -periodic functions of the true phase, as they are extracted from periodic signals or waves. The periodic nature of the measurement process yields very difficult inverse problems, regardless of the framework adopted to formulate and solve them. For example, under the regularization framework, and even using convex regularizers for the phase, the presence of a periodic data term in the objective function leads to unbearable nonconvex optimization problems (see, e.g., [1]–[9]).

### A. Related Work: Two-Step Approach

With the objective of formulating treatable phase imaging inverse problems, most approaches follow a two-step procedure: In the first step, an estimate of the true phase in the interval  $[-\pi, \pi)$ , the so-called *principal* phase values, or *wrapped* phase, or *interferometric* phase, is inferred from noisy wrapped observations, and in the second step, the true phase<sup>1</sup> is inferred from the interferometric phase estimate obtained in the first step. The latter procedure is known as *phase unwrapping* and corresponds to the addition of an integer number of  $2\pi$  multiples to the estimated interferometric phase [5].

In this paper, our primary objective is the estimation of the interferometric phase from noisy observations. This is a challenging inverse problem, owing to the nonlinear (sinusoidal) observation mechanism, as already aforementioned, in addition to the nonadditive nature of the phase noise and to the spatial undersampling<sup>2</sup> in areas of high phase rate or discontinuities. These degradation mechanisms can only be countered by the use of prior information about the true phase. One of the earliest and simpler approaches consists in assuming that the phase

Manuscript received May 29, 2013; revised October 20, 2013, May 5, 2014, and September 6, 2014; accepted September 15, 2014. This work was supported in part by the Portuguese Science and Technology Foundation through Projects PEst-OE/EEI/LA0008/2013 and PTDC/EEI-PRO/1470/2012, by the Academy of Finland through Project 138207, 2011–2014, and by China Scholarship Council (2011611073) through the Doctor Project of the National University of Defense Technology (B110503) and the National Natural Science Foundation of China (61103081).

H. Hongxing is with the College of Information Systems and Management, National University of Defense Technology, Changsha 410073, China, and also with the Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal (e-mail: hongxinghao87@gmail.com).

J. M. Bioucas-Dias is with the Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal (e-mail: bioucas@lx.it.pt).

V. Katkovnik is with the Department of Signal Processing, Tampere University of Technology, 33101 Tampere, Finland (e-mail: vladimir.katkovnik@tut.fi).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TGRS.2014.2361919

<sup>1</sup>In this paper, the term “phase” is used in the sense of absolute phase, i.e., not wrapped, as opposed to the interferometric phase.

<sup>2</sup>Herein, undersampling means that the magnitude of the phase difference between neighboring pixels is equal to or larger than  $\pi$ .

is constant in small windows and computing the maximum likelihood estimate of the interferometric phase [5] in these windows. The assumption of constant phase in small windows, termed the local polynomial approximation (LPA) of zero order in [10], is reasonable in areas of smooth variation. However, in areas of large variation or of discontinuities, it leads to oversmoothing.

A time–frequency analysis based on Fourier transform is a conventional tool in the analysis and filtering of the wrapped phase (see [11] and [12] and the references therein). These approaches exploit the fact that the windowed Fourier transform (WFT) of  $\exp(j\phi)$  (symbol  $j$  denotes the unit complex number) of phase surfaces  $\phi$  locally well approximated by first-order polynomials, i.e., by local planes, is localized in the Fourier plane. Presently, the WFT is considered to be a promising tool among the spatial techniques in use for phase measurement [7].

The size of the windows used under the LPA of zero order in [5] and of first order in the WFT-based methods [11], [12] plays a key role in the quality of the interferometric estimates. In fact, if the windows are too big, the assumption that the phase is smooth is violated, and the applied filtering is too strong, damaging the essential patterns of the phase coded in its interferometric observations. As a result, the reconstruction of true phase is compromised. On the other hand, if the windows are too small, the interferometric noise is little filtered out, and the unwrapping may be impossible. A conclusion is, therefore, that the size of the local windows should be selected very carefully.

One way to ensure effective denoising is to adapt the size of the local windows according to the phase smoothness and to the noise level. This filtering adaptiveness was the main motivation underlying the introduction of the Phase Estimation using Adaptive Regularization based on Local Smoothing (PEARLS) algorithm [13]. PEARLS implements a two-step strategy to phase estimation. In the first step, the interferometric phase is inferred based on a first-order LPA using an adaptive window size [10]. The absolute phase is determined in the second step by running the Phase Unwrapping MAX flow/min-cut (PUMA)[14] phase unwrapping algorithm on the estimated interferometric phase. PEARLS represents a step forward with respect to the WFT-based methodologies as it locally adapts the filtering strength according to the phase smoothness. Its major limitation is linked with the use of the first-order LPA for the phase surfaces, thus unable to represent, for example, a pure quadratic surface or surfaces containing discontinuities.

The nonlocal InSAR (NL-InSAR) method introduced in [15] is the state-of-the-art in InSAR. This method uses a nonlocal approach for the joint estimation of the reflectivity, the interferometric phase, and the coherence images from an interferometric pair of coregistered single-look complex (SLC) SAR images. The nonlocal paradigm exploits the self-similarity existing in most real-world images<sup>3</sup> [16] to find similar patches and then reduces their noise by applying some type of averaging. Well-known examples of this approach to image denoising are the nonlocal means filtering [17] and the state-of-the-art

block matching with 3-D filtering (BM3D) [18]. Adopting the nonlocal paradigm, NL-InSAR introduces a patch similarity criterion suitable to SLC images and a weighted maximum likelihood estimation of the SAR interferogram with weights derived in a data-driven way. Both similarity criterion and weighted maximum likelihood are grounded on the statistics of the SAR data.

## B. Related Work: Simultaneous Denoising and Unwrapping

In addition to the two-step-based algorithms to phase estimation, there are a few implementing interferometric phase estimations and the phase unwrapping simultaneously. The  $Z\pi M$  algorithm [19] is an iterative scheme for the computation of the maximum *a posteriori* probability (MAP) absolute phase estimate. Each iteration is composed of a discrete optimization step ( $Z$ -step), implemented by network programming techniques, and an iterative conditional mode step ( $\pi$ -step). The PhaseLa technique [20] applies LPA to the argument of  $\cos(\cdot)$  and  $\sin(\cdot)$  to estimate the absolute phase from noisy wrapped phase data and the intersection-of-confidence-interval algorithm to adaptively select pointwise varying window sizes. The major weakness of the  $Z\pi M$  and PhaseLa algorithms is that the phase discontinuities must be supplied or, otherwise, they are not preserved.

The matrix-pencil-based method [21] uses the *matrix-pencil* algorithm to estimate the local frequencies by fitting locally a complex exponential to the complex interferogram. The phase is then inferred by integrating the local instantaneous frequencies. Given that estimating a local frequency is equivalent to fitting locally a plane to the phase, this method is interpretable as a first-order LPA for the phase. Therefore, it works well in smooth surfaces but lacks capacity to model phase surfaces with high-order derivatives or with discontinuities.

The half-quadratic (HQ) regularization method [22] formulates the phase inference under the regularization framework. The phase is estimated by minimizing an objective function containing a quadratic data term, which measures the data misfit with respect to the wrapped local phase differences, and a regularizer that promotes phase smoothness. In order to model the phase discontinuities, a line field, which signals discontinuities between neighboring pixels, is introduced. The minimization alternates between the phase and the line process. The major weaknesses of the HQ approach are that the underlying objective function is nonconvex, thus yielding solutions highly dependent on the initialization, and that the data term does not capture the statistics of the observation mechanism.

The combinatorial absolute phase estimation (CAPE) algorithm [23] assumes a first-order Markov random field prior and a maximum MAP viewpoint. A combinatorial suboptimal algorithm that involves the computation of a multiprecision sequence is introduced to solve the MAP problem. The PUMA algorithm [14] is used in the coarser precision to unwrap the phase and to detect the discontinuities; each piecewise smooth unwrapped region is denoised in the subsequent iterations with increasing precision based on a graph min-cut algorithm. The success of CAPE relies on the detection of the phase discontinuities by PUMA in the first iteration of the algorithm. This

<sup>3</sup>That is, in a given image, there are many similar patches in different locations and/or with different scales.

detection is, however, compromised if the observed phase is too noisy.

The stochastic-filtering-based approaches estimate the phase sequentially and recursively according to some ordering of the image pixels or groups of pixels. Various forms of stochastic filtering have been exploited such as Kalman or extended Kalman filtering [24]–[26], grid-based filtering [27], particle filters [28], and combinations of ideas from particle filters, local slope estimation based on matrix pencil, and optimized region-growing techniques [29]. The stochastic filtering methods enable for an accurate modeling of the observation mechanism. This is not the case, however, regarding the prior information about the phase surfaces, such as piecewise smoothness, which is hard to encode within the stochastic filtering framework. In addition, stochastic filtering is a recursive technique which is a significant disadvantage compared with the batch processing, as, in the former case, the phase estimates are computed just from a subset of the available observations (i.e., past plus current observations), whereas the phase estimates in the batch methods depend on the complete image of observations.

### C. Proposed Approach

In this paper, we attack the estimation of interferometric phase by replacing the original problem by the estimation of  $\exp(j\phi)$ . While these problems are equivalent in a given statistical sense, the latter opens the door to the development of sparse coding [30], [31] techniques adapted to the complex field.

Following the standard procedure in patch-based image restoration, the image is partitioned into small overlapping square patches, and the vector corresponding to each patch is modeled as a sparse linear combination of vectors, termed atoms, taken from a set called dictionary. The process of finding the coefficients of the linear combination is termed *sparse coding* [30], [32]. Assuming that the sparse coding uses a quadratic data term, usually linked to the assumption of Gaussian noise, then the representation for a given data vector corresponds to an orthogonal projection on the subspace spanned by the active atoms (those with nonzero weights in the linear combination) of the dictionary. Therefore, the additive noise term present in the data vector is projected onto that subspace. Assuming zero-mean independent and identically distributed (iid) noise, the power of the projected noise is proportional to the dimension of the subspace, hence the relevance of sparse coding in the considered denoising problem as a way to identify low-dimensional subspaces.

Aiming at optimal sparse representations, and thus at optimal noise removing capabilities, the dictionary is learned from the data that it represents. The sparse decomposition of a signal on dictionaries learned from the data underlies a series of state-of-the-art results, for example, in image processing tasks such as denoising, inpainting, and demosaicking [30], [32] and in classification [33], [34].

We formulate the dictionary learning (DL) as a basis pursuit denoising (BPDN) [35] optimization in the complex domain. In this optimization, the objective function is the sum of the quadratic norm of the representation error plus a sparsity promoting term, the  $\ell_1$  norm of the linear regression coefficients.

The number of overlapping patches of an image of size  $N$  is on the order of  $N$ . This means that the learning of the dictionary yields a large-scale optimization, even considering small images, for example, of size  $N = 10\,000$ . To cope with this computational complexity, we adopt the online DL (ODL) algorithm introduced in [36] to our application. In addition, and also for faster computation, we implement the sparse coding step with an instance of the *split augmented Lagrangian shrinkage algorithm* (SALSA) introduced in [37].

As already referred to, this paper is mainly devoted to interferometric phase estimation. However, because the objective of the interferometric phase estimation is often to serve as input to a phase unwrapping algorithm, we will run unwrapping experiments illustrating the impact of interferometric filtering in the final estimate of the true phase. For phase unwrapping, we use PUMA [14], which is an exact integer optimization solver for convex pairwise potentials and is the state-of-the-art for discontinuity preserving ones.

### D. Contributions

This paper introduces the following contributions:

- 1) reformulation of the interferometric estimation problem as a sparse regression in the complex domain;
- 2) interferometric phase image estimation algorithm, termed the SpInPHASE, exploiting sparse coding in the complex domain.

In addition, and also for faster implementation of the ODL algorithm [36], we replace the *least angle regression* (LARS) [38] originally proposed to implement the BPDN problem sparse coding step with an instance of the SALSA introduced in [37], herein termed the *sparse regression by variable splitting and augmented Lagrangian* (SpaRSAL).

This paper is organized as follows. Section II introduces the interferometric phase problem and gives the rationale for sparse coding approach. Section III addresses the reconstruction of the interferometric image. Section IV is devoted to the DL algorithm. Section V presents a series of results with real and simulated data. Finally, Section VI finishes this paper with a few concluding remarks.

## II. PROBLEM FORMULATION

The details of the observation models relating the noisy interferometric phase with the true phase depend on the imaging system under consideration. For an account of observation models in different coherent imaging systems, see, e.g., [5], [15], [19], [20], and [39]. Let us assume, for now, that the observed data at a given image pixel are given by

$$z = a \exp(j\phi) + n \quad (1)$$

where  $a \geq 0$  and  $n = n_I + jn_Q$  is the complex-valued zero-mean Gaussian circular white noise of variance  $\sigma^2$  (i.e.,  $n_I$  and  $n_Q$  are zero-mean independent Gaussian random variables with variance  $\sigma^2/2$ ). The observation model (1) captures the essential features of the interferometric phase estimation and is a good approximation for magnetic resonance imaging, for



3-D surface shape from sinusoidal light patterns, and for all optical interferometric techniques operating in Poissonian noise regimes well approximated by Gaussian densities. It does not apply, however, to InSAR/InSAS and to low count regime in Poissonian noise. In order to deal with InSAR/SAR statistics, we will introduce, in Section III, a modification to SpInPHASE yielding state-of-the-art results.

Let us assume that we are dealing with images defined on a grid of size  $N := N_1 \times N_2$  and that  $\mathbf{z} := [z_i, i = 1, \dots, N]^T$  holds the observed image arranged into a column vector according to the lexicographical order of the set  $\{1, \dots, N_1\} \times \{1, \dots, N_2\}$ . By the same token, let  $\phi := [\phi_i, i = 1, \dots, N]^T$ ,  $\mathbf{a} := [a_i, i = 1, \dots, N]^T$ , and  $\mathbf{n} := [n_i, i = 1, \dots, N]^T$  hold the true phase image, the true amplitude image, and the noise image, respectively. We define the true complex image as  $\mathbf{x} := \mathbf{a} \odot e^{j\phi}$ , where  $\odot$  stands for component-wise multiplication and  $e^{j\phi}$  is to be understood component-wise.

Given the phase  $\phi$ , the corresponding interferometric phase is defined as

$$\phi_{2\pi} := \mathcal{W}(\phi)$$

where

$$\mathcal{W} : \mathbb{R} \longrightarrow [-\pi, \pi) \quad (2)$$

$$\phi \mapsto \text{mod}(\phi + \pi, 2\pi) - \pi \quad (3)$$

where  $\text{mod}(\cdot)$  is the modulus after division by  $2\pi$ . With these definitions in place, the interferometric phase estimation problem is to estimate the image  $\phi_{2\pi} := [\phi_{2\pi,i}, i = 1, \dots, N]^T$  from the observed image  $\mathbf{z}$  given by

$$\mathbf{z} = \mathbf{x} + \mathbf{n}.$$

Our approach consists in computing  $\hat{\mathbf{x}}$ , an estimate of  $\mathbf{x}$ , and then computing  $\hat{\phi}_{2\pi} := \arg(\hat{\mathbf{x}})$ , where  $\arg(\cdot)$  is to be understood component-wise and  $\arg(x) \in [-\pi, \pi)$  is the argument of the complex  $x$ . Later, we will elaborate more on this criterion.

#### A. Patch Decomposition and Composition of an Image

Consider a noisy image  $\mathbf{z} \in \mathbb{C}^N$  and the patch  $\mathbf{z}_i \in \mathbb{C}^m$  containing the pixels located inside a square window of size  $\sqrt{m} \times \sqrt{m}$  centered at the  $i$ th pixel. The total number of overlapping patches is  $N_p = (N_1 - \sqrt{m} + 1)(N_2 - \sqrt{m} + 1)$ .

Let  $\mathbf{x}_i \in \mathbb{C}^m$  and  $\mathbf{n}_i \in \mathbb{C}^m$  denote two vectors holding the elements of  $\mathbf{x}$  and of  $\mathbf{n}$  corresponding to the  $i$ th patch, respectively. We have then

$$\mathbf{z}_i = \mathbf{x}_i + \mathbf{n}_i, \quad i = 1, \dots, N_p. \quad (4)$$

Let  $\hat{\mathbf{x}}_i$  denote an estimate of  $\mathbf{x}_i$ , and write

$$\hat{\mathbf{x}}_i = \mathbf{x}_i + \varepsilon_i \quad (5)$$

where  $\varepsilon_i$  is the estimation error for the  $i$ th patch. Of course, our denoising scheme makes sense if the “size” of  $\varepsilon_i$  is smaller than the “size” of  $\mathbf{n}_i$ .

In order to produce an estimate of  $\mathbf{x}$  from the estimates  $\hat{\mathbf{x}}_i$ , for  $i = 1, \dots, N_p$ , we introduce the selection matrices  $\mathbf{M}_i$  such that  $\mathbf{x}_i = \mathbf{M}_i \mathbf{x}$  (i.e., each row of  $\mathbf{M}_i$  contains just one nonnull element of value of one) and the following matrix and vectors:

$$\begin{aligned} \mathbf{M} &:= [\mathbf{M}_1^T, \dots, \mathbf{M}_{N_p}^T]^T \\ \hat{\mathbf{x}}_P &:= [\hat{\mathbf{x}}_1^T, \dots, \hat{\mathbf{x}}_{N_p}^T]^T \\ \varepsilon_P &:= [\varepsilon_1^T, \dots, \varepsilon_{N_p}^T]^T. \end{aligned}$$

Using the aforementioned definitions, we have

$$\hat{\mathbf{x}}_P = \mathbf{M} \mathbf{x} + \varepsilon_P \quad (6)$$

and compute

$$\hat{\mathbf{x}} = \mathbf{M}^\# \hat{\mathbf{x}}_P \quad (7)$$

where  $\mathbf{M}^\# := (\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T$ . Given that  $\mathbf{M}^T \mathbf{x}_i$  places the patch number  $i$  at its position in the image and that  $(\mathbf{M}^T \mathbf{M})$  is a diagonal matrix whose  $i$ th diagonal element holds the number of times pixel  $i$  appears in any patch, then the estimate for the  $i$ th pixel is the average of all its estimates, one per patch containing thereof. This conclusion implies that matrix  $\mathbf{M}^T \mathbf{M}$  is invertible as far as any pixel belongs at least to one patch.

Assuming that  $\varepsilon_P$  is zero mean and iid, thus having a covariance matrix proportional to the identity matrix, i.e.,  $\mathbf{C}_\varepsilon \propto \mathbf{I}$ , then (7) is the *best linear unbiased estimator* (BLUE) for  $\mathbf{x}$  [40]. Due to the overlapping structure of the patches,  $\mathbf{C}_\varepsilon$  is not proportional to  $\mathbf{I}$ , and thus, (7) is not BLUE. However, because it is very difficult to accurately compute  $\mathbf{C}_\varepsilon$ , we adopt the sub-optimal estimate (7) that, nevertheless, yields state-of-the-art results, namely, regarding the interferometric phase estimation, as widely illustrated in the experiments shown in Section V. The connection between the interferometric phase estimate and the complex image estimate (7) is addressed in Section III-B.

### III. INTERFEROMETRIC PHASE ESTIMATION VIA SPARSE REGRESSION

The topic of sparse and redundant representations of real-world images has attracted tremendous interest from the research community in the last ten years [31], [41]. This interest stems from the fundamental role that low-dimensional models play in many signal and image areas such as compression, restoration, classification, and design of priors and regularizers, just to name a few. To add yet more interest to this topic, the dictionaries yielding sparse representations may be learned from the data that they represent [30], [32]. DL is currently one of the hottest research topics in this area.

A representation is said to be sparse if the images, or patches of them, are well approximated by linear combinations of a few atoms taken from a dictionary. It happens that the real-world images (and signals) admit sparse representations in suitable dictionaries. This characteristic is a consequence of the high level of self-similarity of real-world images (and signals), i.e., given an image patch, there is a high likelihood of finding similar patches at different locations and scales.

Herein, we explore the sparse and redundant representation framework to compute estimates of the complex image patches  $\mathbf{x}_i$ , for  $i = 1, \dots, N_P$ , from the noisy image  $\mathbf{z}$ . A key element for the success of this approach is that those patches admit a sparse representation on a given dictionary. Let  $\mathbf{a}_i$  and  $e^{j\phi_i}$  denote the amplitude and the complex phase patches corresponding to  $\mathbf{x}_i$ , respectively (i.e.,  $\mathbf{x}_i = \mathbf{a}_i \odot e^{j\phi_i}$ ). Assuming that  $\mathbf{a}_i$  and  $\phi_i$  admit sparse representations and using the fact that  $|e^{j\phi_i} - e^{j\phi_k}| \leq |\mathcal{W}(\phi_i - \phi_k)| \leq |\phi_i - \phi_k|$ , it can be qualitatively concluded that the  $\mathbf{x}_i$  also admits a sparse representation. A deep study of this issue is, however, out of the scope of this paper. Nevertheless, we will give strong experimental evidence that the patches of  $\mathbf{x}$  admit highly sparse representations when  $\mathbf{a}$  and  $\phi$  are real-world images.

Suppose that we are given a dictionary  $\mathbf{D} \equiv [\mathbf{d}_1, \dots, \mathbf{d}_k] \in \mathbb{C}^{m \times k}$  with respect to which the patches of  $\mathbf{x}$  admit a sparse representation. Following the standard formulation in synthesis-based approaches to sparse regression, the estimate of a given patch  $\mathbf{x}_i = \mathbf{M}_i \mathbf{x}$ , for a given  $i \in \{1, \dots, N_P\}$ , is given by  $\hat{\mathbf{x}}_i = \mathbf{D} \hat{\boldsymbol{\alpha}}$ , where  $\hat{\boldsymbol{\alpha}} \in \mathbb{C}^k$ , often termed the *code*, is the solution of the constrained optimization

$$\min_{\boldsymbol{\alpha}} \|\boldsymbol{\alpha}\|_0 \text{ subject to : } \|\mathbf{D}\boldsymbol{\alpha} - \mathbf{z}_i\|_2^2 \leq \delta \quad (8)$$

where  $\mathbf{z}_i = \mathbf{M}_i \mathbf{z}$  is the observed patch corresponding to the true patch  $\mathbf{x}_i$  and  $\|\boldsymbol{\alpha}\|_0$  is the number of nonzero elements of vector  $\boldsymbol{\alpha}$ , often termed the abusive  $\ell_0$  norm, and  $\delta \geq 0$  is a parameter controlling the reconstruction error.

Given that  $\mathbf{D} \hat{\boldsymbol{\alpha}} = \hat{\mathbf{x}}_i = \mathbf{x}_i + \boldsymbol{\varepsilon}_i$ , then if the norm of  $\boldsymbol{\varepsilon}_i$  is much smaller than the norm of the observation noise  $\mathbf{n}_i = \mathbf{z}_i - \mathbf{x}_i$ , it follows that  $\|\mathbf{D} \hat{\boldsymbol{\alpha}} - \mathbf{z}_i\|_2^2 \simeq \|\mathbf{n}_i\|_2^2$  and then that the parameter  $\delta$  should be closely related with  $\|\mathbf{n}_i\|_2^2$ . As in [32] and [42], we use the criterion  $P(\|\mathbf{n}_i\|_2^2 \leq \delta) = \gamma$ . Assuming that  $\mathbf{n}_i$  is zero mean Gaussian complex valued with covariance  $\sigma^2 \mathbf{I}$ , then  $\|\mathbf{n}_i\|_2^2 / (\sigma^2/2)$  follows a chi-squared distribution  $\chi^2(2m)$  with  $2m$  degrees of freedom. We have then  $\delta = (\sigma^2/2) F_{\chi^2(2m)}^{-1}(\gamma)$ , where  $F_{\chi^2(2m)}^{-1}(\cdot)$  is the inverse of the cumulative distribution function of  $\chi^2(2m)$ . In the experiments presented in Section V, we set  $\gamma = 0.96$ , which we have empirically observed to yield very good results.

#### A. Sparse Representation and Noise Reduction

The patch estimate error  $\boldsymbol{\varepsilon}_i = \hat{\mathbf{x}}_i - \mathbf{x}_i$ , with  $\hat{\mathbf{x}}_i = \mathbf{D} \hat{\boldsymbol{\alpha}}$  and  $\hat{\boldsymbol{\alpha}}$  given by (8), is strongly related with the level of sparsity of  $\hat{\boldsymbol{\alpha}}$  given by  $p := \|\hat{\boldsymbol{\alpha}}\|_0$ . To shed light into this issue, let us define  $S := \text{supp}(\hat{\boldsymbol{\alpha}})$  as the set of indexes of those nonzero elements of  $\hat{\boldsymbol{\alpha}}$ ,  $\mathbf{D}_S$  as a matrix holding the atoms of  $\mathbf{D}$  indexed by  $S$ , and  $\mathbf{P}$  as the projection matrix onto the range of  $\mathbf{D}_S$ . Assuming that  $\mathbf{x}_i$  is in the range of  $\mathbf{D}_S$  and that the feasible set of the optimization (8) is not empty, then we have

$$\delta \geq \min_{\boldsymbol{\beta} \in \mathbb{C}^p} \|\mathbf{D}_S \boldsymbol{\beta} - \mathbf{z}_i\|_2^2 = \|(\mathbf{I} - \mathbf{P}) \mathbf{n}_i\|_2^2$$

and the minimum residual is obtained for

$$\mathbf{D}_S \boldsymbol{\beta} = \mathbf{P} \mathbf{z}_i = \mathbf{P}(\mathbf{x}_i + \mathbf{n}_i) = \mathbf{x}_i + \mathbf{P} \mathbf{n}_i. \quad (9)$$

Since  $\hat{\mathbf{x}}_i := \mathbf{D}_S \boldsymbol{\beta}$ , we conclude therefore that the patch estimation error corresponding to the minimum residual is  $\boldsymbol{\varepsilon}_i = \mathbf{P} \mathbf{n}_i$ .

Assuming that the noise is zero mean and has covariance  $\sigma^2 \mathbf{I}$ , we have

$$\mathbb{E} [\|\boldsymbol{\varepsilon}_i\|_2^2] = \sigma^2 \text{trace}(\mathbf{P}) = p \sigma^2$$

yielding the relative attenuation of the noise

$$\frac{\mathbb{E} [\|\boldsymbol{\varepsilon}_i\|_2^2]}{\mathbb{E} [\|\mathbf{n}_i\|_2^2]} = \frac{p}{m}. \quad (10)$$

Expression (10) reflects an important aspect of the sparse representations: The estimation error is proportional to the sparsity level of the signal.

In practice, the ratio (10) is hardly achieved mainly owing to errors in  $\hat{\boldsymbol{\alpha}}$  with impact in  $S$ . These errors are linked to the intrinsic difficulty of the optimization problem (8), which is nondeterministic polynomial time (NP) hard [43], to the possible nonuniqueness of the solutions thereof, namely, when the atoms of the dictionary are strongly correlated, which is generally the case in dictionaries learned from real-world image, and to the noise present in the observation  $\mathbf{z}$ , which further degrades the quality of the solutions. In spite of these shortcomings, the sparse and redundant representations are in the heart of many state-of-the-art applications, namely, in signal and image processing, statistical inference, and machine learning fields.

#### B. Interferometric Phase Estimation

We recall that the main objective of this work is the estimation of the interferometric phase via sparse coding. Our approach consists in first computing the patch estimates  $\hat{\mathbf{x}}_i$ , for  $i = 1, \dots, N_P$ , and, from them, the interferometric phase image  $\hat{\phi}_{2\pi}$ . Consider a given pixel where the true complex data are  $x = a e^{j\phi}$ . Let  $i_1, \dots, i_q$  be the indexes of the patches that contain this pixel and  $\hat{x}_k$  be the estimate of  $x$  in the patch  $i_k$ . We may then write  $\hat{x}_k = x + \varepsilon_k$ , where  $\varepsilon_k$  is the respective estimation error. Assuming that the vector  $[\varepsilon_1, \dots, \varepsilon_q]^T$  is zero mean Gaussian with covariance matrix  $\mathbf{C}$ , it follows that the maximum likelihood estimate of  $a$  and  $\phi_{2\pi}$  is

$$\hat{\phi}_{2\pi} = \arg \left( \sum_{k=1}^q \hat{x}_k \gamma_k \right) \quad (11)$$

$$\hat{a} = \frac{|\sum_{k=1}^q \hat{x}_k \gamma_k|}{\sum_{k=1}^q \gamma_k} \quad (12)$$

where  $\gamma_k := \sum_{l=1}^q [\mathbf{C}^{-1}]_{kl}$ .

In practice, matrix  $\mathbf{C}$  is very hard to compute because it depends on the statistics of the noise and on the estimated code for the patches. So, we take the pragmatic option of setting  $\gamma_k$  constant in (11) and (12), originating the estimates  $\hat{\phi}_{2\pi} = \arg(\sum_{k=1}^q \hat{x}_k)$  and  $\hat{a} = (1/q) |\sum_{k=1}^q \hat{x}_k|$ . The estimated images are then given by

$$\hat{\phi}_{2\pi} = \arg(\hat{\mathbf{x}}) \quad (13)$$

$$\hat{\mathbf{a}} = \text{abs}(\hat{\mathbf{x}}) \quad (14)$$

where  $\arg(\cdot)$  and  $\text{abs}(\cdot)$  are to be understood component-wise and the complex image estimate  $\hat{\mathbf{x}}$  is given by (7). The setting of  $\gamma_k$  constant, compared with the use of the correct  $\gamma_k$ , increases slightly the variance of the corresponding estimates. However,

this disadvantage is largely compensated by the gains in the computational complexity.

As it may be concluded from the previous sections, and particularly from (13) and (14), SpInPHASE provides estimates of both amplitude  $a$  and interferometric phase  $\phi_{2\pi}$ . However, we remark that, in many applications, the noise variance is spatially variant, thus violating the hypothesis assumed for the observation model (1) that the noise variance is constant across the image pixels. A simple strategy to circumvent this model mismatch consists in dividing the observations  $z_i$ , for  $i = 1, \dots, N$ , by a local estimate of the noise standard deviation, for example,  $\hat{\sigma}_i$ . In this case, we obtain an estimate of the normalized amplitude  $a_i/\hat{\sigma}_i$ .

In conclusion, although SpInPHASE deals with amplitude information, the relation of this amplitude with the true one is not straightforward, deserving research which is, however, beyond the scope of this paper, as we are mainly focused on the interferometric phase estimation.

### C. InSAR Case

The observation model (1) does not apply directly to InSAR measurements. Notwithstanding, as shown in the following, we may still use that model, if a suitable spatially variant noise is considered. In InSAR, and for each pixel, we are given two complex-valued measurements, denoted by  $x_1$  and  $x_2$ , linked with the signals acquired by two SAR sensors flying along parallel and slightly displaced trajectories. Quite often,  $x_1$  and  $x_2$  are complex zero-mean circular Gaussian distributed random variables [39], thus completely characterized by covariance matrix with elements  $c_{11} := \mathbb{E}[|x_1|^2]$ ,  $c_{22} := \mathbb{E}[|x_2|^2]$ ,  $c_{12} := \mathbb{E}[x_1 x_2^*]$ , and  $c_{21} := c_{12}^*$ . The cross covariance factor  $c_{12}/\sqrt{c_{11}c_{22}}$  may be written as  $\gamma e^{j\phi}$ , where  $\phi$  is the propagation phase difference between the sensors and the pixel and  $0 \leq \gamma \leq 1$  is the so-called *coherence* [44].

A relevant objective of InSAR is the estimation of the phase  $\phi$  from the measurements  $x_1$  and  $x_2$  by first computing a denoised estimate from the noisy interferometric phase  $\psi = \arg(x_1 x_2^*)$  and then applying phase unwrapping. From the estimated phase  $\hat{\phi}$ , the elevation of the terrain is inferred based on simple trigonometric arguments.

The probability density function of the InSAR phase noise  $\varepsilon := \psi - \phi_{2\pi}$  in the interval  $[-\pi, \pi)$  is symmetric about the origin with variance given by (see, e.g., [39])

$$\sigma_\varepsilon^2(\gamma) = \frac{\pi^3}{3} - \pi \arcsin(\gamma) + \arcsin^2(\gamma) + \frac{\text{Li}_2(\gamma)}{2} \quad (15)$$

where  $\text{Li}_2$  is Euler's dilogarithm. If  $\gamma$  approaches zero, the density tends to a flat shape, and there is no information about  $\phi_{2\pi}$  in  $\psi$ . If  $\gamma = 1$ , the density is a Dirac at the origin and  $\phi_{2\pi} = \psi$ .

In a given pixel, from the interferometric noisy phase  $\psi = \arg(x_1 x_2^*)$ , define the complex-valued data

$$\begin{aligned} z &= e^{j\psi} \\ &= e^{j(\phi+\varepsilon)}. \end{aligned} \quad (16) \quad (17)$$

The perturbation  $\varepsilon$  in (17) is not additive. However, when  $|\varepsilon| \ll \pi$ , which, in a probabilistic sense, is equivalent to say

that the coherence is not too small, the observation model is approximately given by

$$z \simeq e^{j\phi} + j e^{j\phi} \varepsilon. \quad (18)$$

That is, the perturbation  $\varepsilon$  is additive, in addition to being zero mean, as in (1). However, given that the coherence  $\gamma$  varies across the image, the same is true for the variance, as the latter is a function of the former. In order to obtain constant variance across the image pixels, as assumed in (1), we compute  $z/\sigma(\gamma)$  and run SpInPHASE on the obtained data.

Although SpInPHASE was not conceived to operate with InSAR/InSAS data, we give in Section V strong experimental evidence that it yields state-of-the-art InSAR/InSAS interferometric phase estimates, provided that a light preprocessing step is applied to the observed data.

### D. Computing Sparse Codes

The only case in which the solution of the optimization (8) is easy to compute, assuming that there is one, is when  $\delta = 0$  and the dictionary has full column rank. This is not the scenario that we systematically have in sparse coding of noisy signals, where the dictionaries are often overcomplete and, due to noise,  $\delta > 0$ . In these circumstances, the optimization (8) is NP hard [43], meaning that there is little hope to solve it exactly in a straightforward way. Two different strategies have been followed to compute approximate solutions. One consists in replacing the  $\ell_0$  norm with a convex approximation, most often the  $\ell_1$  norm yielding the so-called least absolute shrinkage and selection operator [45], which is equivalent to BPDN [35]. The other strategy consists in attacking directly the original problem using greedy algorithms such as the orthogonal matching pursuit (OMP) [46], iterative hard thresholding, [47], hard thresholding pursuit [48], or approximate message passing [49].

The quality of the approximations provided by all of the aforementioned algorithms is linked with the degree of sparseness of the patches, with the observation noise, and with the coherence of the dictionary atoms, measured, e.g., in terms of the so-called mutual coherence [50] or the restricted isometric constants [51]. Qualitatively, the higher the mutual coherence, the higher the degree of sparseness ensuring perfect unmixing. In our application, the mutual coherence is often very high, thus imposing stringent limits on the minimum degree sparsity that a patch may have in order to be recovered. This drawback is somehow mitigated by the very high degree of sparseness of most image patches, implying that all algorithms referred to earlier produce useful results.

Herein, we adopt the OMP tailored to the complex field. The reason for this choice stems from the lower computational complexity of this algorithm compared with the others and also because we have experimentally observed a slight advantage, namely, in comparison with the BPDN algorithm. This finding is in line with that in [36].

Algorithm 1 shows the pseudocode for the OMP. We initialize  $e := \infty$  to ensure that  $S := \text{supp}(\alpha)$  contains at least one element. This constraint stems from the fact that, in interferometric phase applications, the amplitude is seldom zero, implying that  $|S| \geq 1$ . In line 5, the notation  $()^H$  stands for Hermitian.

In line 8,  $\mathbf{D}_S^\sharp$  denotes the pseudoinverse of  $\mathbf{D}_S$ , and in line 12,  $\alpha(S)$  denotes the components of  $\alpha$  with indexes in  $S$ .

---

**Algorithm 1:** Orthogonal Matching Pursuit (OMP)

---

**Input:**  $\mathbf{D} \in \mathbb{C}^{m \times k}$  (dictionary),  $\mathbf{z} \in \mathbb{C}^m$  (patch),  $\delta > 0$  (tolerance)  
**Output:**  $\alpha \in \mathbb{C}^k$  (code).

```

1 begin
2    $S := \emptyset, \alpha := \mathbf{0}, e := \infty$ 
3    $\mathbf{r} := \mathbf{z}$ 
4   while  $e > \delta$  do
5      $\mathbf{u} := \mathbf{r}^H \mathbf{D}$ 
6      $i := \arg \max_k |u_k|$ 
7      $S := S \cup \{i\}$ 
8      $\beta := \mathbf{D}_S^\sharp \mathbf{z}$ 
9      $\mathbf{r} := \mathbf{z} - \mathbf{D}_S \beta$ 
10     $e := \|\mathbf{r}\|^2$ 
11  end
12   $\alpha(S) := \beta$ 
13 end
```

---

#### IV. DL IN THE COMPLEX DOMAIN

So far, we have assumed that a dictionary  $\mathbf{D} \in \mathbb{C}^{m \times k}$ , regarding to which the image patches admit sparse representations, is available. Various types of predefined dictionaries have been used in the past adapted to different classes of signals (e.g., based on wavelets for real-world images and based on sinusoids for band-limited signals). The demand for “good” dictionaries, in the sense of sparse representation, has stimulated active research in DL, aiming at adapting the dictionaries to the data that they represent [34], [36], [52]–[55]. As a result of this research, DL is in the core of many state-of-the-art methodologies and algorithms, for example, in image processing tasks such as denoising, inpainting, and demosaicking [30], [32] and in classification [33], [34].

Given a set of patches from an image, the objective of DL is to find a dictionary that is able to accurately represent the patches with the smallest possible number of atoms. A formulation of this idea under the regularization framework is as follows:

$$\min_{\mathbf{D} \in \mathcal{C}, \alpha_1, \dots, \alpha_{N_p}} \sum_{i=1}^{N_p} \left( \frac{1}{2} \right) \|\mathbf{z}_i - \mathbf{D} \alpha_i\|_2^2 + \lambda \|\alpha_i\|_1 \quad (19)$$

where  $\mathcal{C} := \{\mathbf{D} \in \mathbb{C}^{m \times k} : |\mathbf{d}_l^H \mathbf{d}_l| \leq 1, l = 1, \dots, k\}$ . The quadratic terms account for the representation errors, and the  $\ell_1$  norm promotes sparse codes. The relative weight between the two terms is established by the regularization parameter  $\lambda > 0$ . The constraint  $\mathbf{D} \in \mathcal{C}$  prevents  $\mathbf{D}$  from being arbitrarily large.

In (19), the optimization with respect to  $\alpha_i$  is a BPDN problem. We could have used other sparsity-inducing regularizers based, for example, on the  $\ell_0$  norm as in (8), to compute the sparse code given the dictionary. We have experimentally observed, however, that the  $\ell_1$  regularizer yields slightly better dictionaries than the  $\ell_0$  one. This behavior may be linked with difficulties in obtaining exact solutions based on the  $\ell_0$  norm, which does not happen with the  $\ell_1$  norm.

Optimization problem (19) is an instance of the matrix factorization class (see, e.g., [36] and the references therein). Alternating optimization with respect to  $\mathbf{D}$  and  $\alpha_1, \dots, \alpha_{N_p}$  is the usual way of attacking these problems (19) [52], [54]–[57]. The optimization with respect to  $\mathbf{D}$  is a quadratic problem with convex constraints, and the optimization with respect to  $\alpha_1, \dots, \alpha_{N_p}$  is convex and decoupled. However, the optimization with respect to all variables is nonconvex, and therefore, there is no guarantee of finding the global minimum. However, in a large class of problems, the obtained stationary points have shown to produce state-of-the-art results.

In a typical image scenario, we have  $N_p = 100\,000$ ,  $m = 100$ , and  $k = 200$ . This means that the optimization with respect to  $\mathbf{D}$  is relatively light, but the optimization with respect to  $\alpha_1, \dots, \alpha_{N_p}$  is extremely heavy. To cope with this shortcoming, we adopt the online learning approach introduced in [36]. We generate a random sequence of patches (or groups of patches)  $\mathbf{z}^t$ , for  $t = 1, \dots$ , sampled from the set  $\mathbf{z}_1, \dots, \mathbf{z}_{N_p}$  and process them sequentially. For each new element in the sequence, we compute the respective code (or codes) via BPDN and then update the dictionary by decreasing the objective function

$$g_t(\mathbf{D}) = \frac{1}{t} \sum_{k=1}^t \left( \frac{1}{2} \right) \|\mathbf{z}^k - \mathbf{D} \alpha_k\|_2^2 + \lambda \|\alpha_k\|_1 \quad (20)$$

where the codes  $\alpha_i$  for  $i := 1, \dots, t-1$  have been computed in the previous iterations.

---

**Algorithm 2:** Online Dictionary Learning (ODL) [36].

---

**Input:**  $\mathbf{z}_i \in \mathbb{C}^m, i = 1, \dots, N_p$  (training set)  
 $T \in \mathbb{N}$  (iterations)  
 $\eta \in \mathbb{N}$  (patches per iteration)  
 $\lambda > 0$  (BPDN regularization parameter)  
 $\beta_t$  (damping sequence)  
 $\mathbf{D}_0 \in \mathbb{C}^{m \times k}$  (initial dictionary)  
**Output:**  $\mathbf{D} \in \mathbb{C}^{m \times k}$  (dictionary).

```

1 begin
2    $\mathbf{D} := [\mathbf{d}_1, \dots, \mathbf{d}_k] = \mathbf{D}_0$ 
3    $\mathbf{A} := [\mathbf{a}_1, \dots, \mathbf{a}_k] = \mathbf{0}, \mathbf{B} := [\mathbf{b}_1, \dots, \mathbf{b}_k] = \mathbf{0}$ 
4   for  $t \leq T$  do
5     Select randomly  $\mathbf{z}^t \equiv [\mathbf{z}_i^t, i = 1, \dots, \eta]$  from  $\mathbf{z}$ 
6     /* Sparse coding: BPDN problems */
7      $\alpha^t := \arg \min_{\alpha \in \mathbb{C}^{k \times \eta}} (1/2) \|\mathbf{z}^t - \mathbf{D} \alpha\|_F^2 + \lambda \|\alpha\|_1$ 
8      $\mathbf{A} := \beta_t \mathbf{A} + \sum_{i=1}^{\eta} \alpha_i^t (\alpha_i^t)^H$ 
9      $\mathbf{B} := \beta_t \mathbf{B} + \sum_{i=1}^{\eta} \mathbf{z}_i^t (\alpha_i^t)^H$ 
10    /* Optimize wrt  $\mathbf{D}$  ----- */
11    repeat
12      for  $l = 1$  to  $l = k$  do
13         $\mathbf{u}_l := \frac{1}{\mathbf{A}(l, l)} (\mathbf{b}_l - \mathbf{D} \mathbf{a}_l) + \mathbf{d}_l$ 
14         $\mathbf{d}_l := \mathbf{u}_l / \max\{\|\mathbf{u}_l\|_2, 1\}$ 
15      end
16    until convergence
17  end
18 end
```

---

For completeness, the pseudocode for the ODL method [36], adapted to the complex field, is shown in Algorithm 2. Line 6 computes the sparse code for the group of  $\eta$  patches  $\mathbf{z}^t$



via BPDN, and lines 9 to 13 decrease the objective function (20) with respect to  $\mathbf{D}$ . The LARS [38] was used in [36] to solve the BPDN problem. Herein, we introduce the SpaRSAL (see details hereinafter), which is much faster than LARS, at least for the problem in hand.

The optimization with respect to the dictionary  $\mathbf{D}$  implements a projected block-coordinate descent method to update the columns of the dictionary. As justified in [36], just one iteration per column is enough, owing to the high concentration of the elements of matrix  $\mathbf{A}$  along its diagonal motivated by the high level of sparsity of codes  $\alpha_i$ . The dictionary obtained in the previous iteration is used as a warm restart for computing the dictionary in the current iteration.

In lines 7 and 8, if  $0 < \beta_t < 1$ , the weight of the information accumulated in  $\mathbf{A}$  and  $\mathbf{B}$  with respect to a given time instant  $t_0$  decreases with  $t$ . This is a usual procedure in online learning, whose objective is to give more relevance to the newer information which is expected to be more accurate. In our implementation, we have used the schedule proposed in [36], given by

$$\beta_t = \left(1 - \frac{1}{t}\right)^\rho, \quad t = 1, 2, \dots \quad (21)$$

in which  $\rho > 0$ .

A proof of convergence of the sequence  $\mathbf{D}_t$  generated by Algorithm 2 to a stationary point of (19) is given in [36, Proposition 4] under the following hypotheses: 1) The observed data have a finite support; 2) the objective functions  $g_t(\mathbf{D})$ , for  $t = 1, \dots$ , in (20) are strictly convex with lower bounded Hessians; and 3) the solution of the BPDN optimization shown in line 6 of Algorithm 2 is unique. In our case, as in [36], hypothesis 3) cannot be guaranteed for matrices  $\mathbf{D} \in \mathcal{C}$ . However, this hypothesis can be enforced by including a very small quadratic term in the BPDN objective function which, from a practical point of view, does not modify the solutions.

#### A. Solving the BPDN Optimization With SpaRSAL

We now introduce the SpaRSAL algorithm to solve the BPDN optimization step in DL Algorithm 2. SparSAL is based on the SALSA developed in [37]. The core idea is to replace the BPDN optimization by the constrained version

$$\begin{aligned} \min_{\alpha \in \mathbb{C}^{k \times \eta}} \quad & \left(\frac{1}{2}\right) \|\mathbf{z}^t - \mathbf{D}\alpha\|_F^2 + \lambda \|\mathbf{u}\|_1 \\ \text{subject to : } \quad & \alpha = \mathbf{u} \end{aligned} \quad (22)$$

and then build the Lagrangian of (22) augmented by the quadratic term  $\|\alpha - \mathbf{u}\|_2^2$ . The algorithm implements alternating minimization with respect to the primal variables  $\alpha$  and  $\mathbf{u}$  and maximization with respect to the dual variables (a scaled version of the Lagrange multipliers). See, e.g., [37], for details. The SpaRSAL pseudocode is shown in Algorithm 3. We also remark that the SpaRSAL is closely related with work [57] and has the same structure of the SUnSAL algorithm [58], developed to solve sparse unmixing regression problems, where the nonnegative constraint was removed.

---

#### Algorithm 3: SpaRSAL algorithm

---

**Input:**  $\mathbf{D} \in \mathbb{C}^{m \times k}$ ,  $\mathbf{z}^t \in \mathbb{C}^{m \times \eta}$ ,  $\mu > 0$  ADMM parameter  $\lambda > 0$  (regularized parameter)  
**Output:**  $\alpha^t \in \mathbb{C}^{k \times \eta}$ .

```

1 begin
2    $\alpha^t := \mathbf{D}^H \mathbf{z}^t$ 
3    $\mathbf{u}^t := \alpha^t$ ,  $\mathbf{v}^t := \mathbf{0}$ 
4    $\mathbf{F} = (\mathbf{D}^H \mathbf{D} + \mu \mathbf{I})^{-1}$ 
5   while not converge do
6      $\mathbf{u}^t := \text{soft}(\alpha^t - \mathbf{v}^t, \lambda/\mu)$ 
7      $\alpha^t := \mathbf{F}(\mathbf{D}^H \mathbf{z}^t + \mu(\mathbf{u}^t + \mathbf{v}^t))$ 
8      $\mathbf{v}^t := \mathbf{v}^t - (\alpha^t - \mathbf{u}^t)$ 
9   end
10 end
```

---

The computational complexity of SpaRSAL to code a set of  $\eta$  patches using the same dictionary is  $O(km^2 + \eta km)$ . Hence, it is much lighter coding groups of patches using the same dictionary than coding the same number of patches with the different dictionaries; if  $\eta > k$ , the gain is about  $k$ . In [36], the LARS [38] is used to solve the BPDN optimization based on a similar rationale. Our preference for the SparSAL stems from the experimental observation that it is much faster than LARS.

The function  $\text{soft}(\mathbf{x}, \tau) := \max\{0, |\mathbf{x}| - \tau\}(\mathbf{x}/|\mathbf{x}|)$  (to be understood component-wise) is to the soft-threshold Moreau proximity operator for  $\ell_1$  norm. For large values of  $\eta$ , step 7 is the heaviest step with a complexity of  $O(k^2)$ . It can be computed with complexity  $O(km)$  based on the left singular vectors and on the singular values of  $\mathbf{D}^H$ . The algorithm converges regardless of the value of  $\mu > 0$ , although it affects the convergence speed. We have implemented the selection rule discussed in [59, Ch. 3.4] and therein formalized in expression (3.13).

## V. RESULTS

In this section, we present a series of experimental results using real and simulated data to illustrate the competitiveness and effectiveness of SpInPHASE. Based on the true phase  $\phi_{2\pi}$  and the estimated interferometric phase estimate  $\hat{\phi}_{2\pi}$ , we define the *peak signal-to-noise ratio* (PSNR) as

$$\text{PSNR} := 10 \log_{10} \frac{4N\pi^2}{\left\| \mathcal{W}(\hat{\phi}_{2\pi} - \phi_{2\pi}) \right\|_F^2} \text{ [dB]} \quad (23)$$

where  $\mathcal{W}$  is the wrapping operator introduced in (2).

As already mentioned, we also unwrap the estimated interferometric phase with the PUMA algorithm [14]. From the estimated phase  $\hat{\phi}$  of the true phase  $\phi$ , define the set of image pixels with an error not larger than  $\pi$ , i.e.,  $I := \{i : |\hat{\phi}_i - \phi_i| \leq \pi, i = 1, \dots, N\}$ , and based on this set, define the *number of errors larger than  $\pi$*  (NELP) and the *PSNR of the absolute phase* (PSNR<sub>a</sub>) as

$$\text{NELP} := N - |I| \quad (24)$$

$$\text{PSNR}_a := 10 \log_{10} \frac{4N\pi^2}{\left\| \hat{\phi}_I - \phi_I \right\|_F^2} \text{ [dB]} \quad (25)$$



where the notation  $\phi_I$  stands for the restriction of  $\phi$  to  $I$ . That is,  $\text{PSNR}_a$  is computed with respect to the set  $I$ . Because the unwrapping is defined apart from a constant  $2\pi$  multiple, when computing the set  $I$ , we identify the constant  $2\pi$  multiple that minimizes  $|I|$ . We remark that the performance indicators NLP and  $\text{PSNR}_a$ , in addition to the information they give about the quality of the estimated true phase, are also a characterization of the denoising algorithm, as the success of the unwrapping depends crucially on the quality of the interferometric phase.

SpInPHASE has two parts: the ODL Algorithm 2, to learn the dictionary, and the OMP Algorithm 1, to compute the sparse codes. The OMP algorithm can also be used with prelearned dictionaries learned, for example, from an image database. The setting of the ODL parameters is the following: regularization parameter  $\lambda = 0.11$  for all surfaces, the size of groups of patches processed simultaneously  $\eta = 0.0064 \times N$ , exponent of the damping sequence  $\rho = 2$ , and number of iterations  $T = 500$ . OMP has just one parameter, the tolerance  $\delta$ , which, as justified in Section III, is set to  $\delta = (\sigma^2/2)F_{\chi^2(2m)}^{-1}(0.96)$ . The stop criterion for SpARSAL is as follows: 1) a primal and a dual residual less than  $\sqrt{k\eta}$  tol, with  $\text{tol} = 10^{-3}$  or 2) the number of iterations reaches  $t = 100$ .

The patch size is  $10 \times 10$  ( $m = 100$ ) in all experiments. We consider two types of dictionaries: 1) learned from the noisy data with a number of atoms of  $k = 256$  and 2) prelearned from six clean synthetic images with a number of atoms of  $k = 512$ . The images are shown in the first row of Fig. 2. We stress that this setting was determined heuristically and therefore is not optimal in any sense. Nevertheless, it yields state-of-the-art results in a wide variety of scenarios regarding the phase surface shape and the amount and type of noise.

SpInPHASE is compared with WFT [11] and NL-InSAR [15], in the case of InSAR observations, using the codes publicly available.<sup>4</sup> WFT has two modes: 1) The *window Fourier filtering* (WFF), which computes the WFT of  $e^{j\phi}$ , hard thresholds the computed coefficients and applies the inverse WFT to the thresholded coefficients, and 2) the *window Fourier ridge* (WFR), which, for each window, computes the larger windowed Fourier coefficient and, based on it, estimates the wrapped phase of the pixel in the center of the window. Coincidentally, there is a strong connection between WFR and PEARLS [13]: For a fixed window, WFR and PEARLS are equal, although derived on rather different principles. However, PEARLS does more: For each pixel, it computes the phase estimates using a set of different window sizes and optimally selects the best estimate using the intersection confidence interval methodology [10]. Because of this adaptiveness, PEARLS yields better results than WFT in WFR mode. We have observed, however, that WFT in WFF mode yields in many cases better results than PEARLS, and consequently, for Gaussian noise, we do comparisons only with WFT in WFF mode.

WFT estimates were obtained with the following parameters: WFF mode, size of the windows  $\sigma_x = \sigma_y = 4$ , threshold for

the WFT  $\text{th} = 3\sigma$ , frequency interval  $[-\pi, \pi]$ , and a sampling interval of 0.1. This setting was determined experimentally, aiming at optimal performance for the set of experiments considered. We remark that this setting yields considerable better results than that recommended in [11] (i.e.,  $\sigma_x = \sigma_y = 10$  and  $[-\pi/2, \pi/2]$ ). All algorithms ran on a personal computer equipped with a Core i7-3770 CPU and 8.00-GB RAM. All results presented in this section can be reproduced by running the publicly available demos.<sup>5</sup>

The results presented in the next sections are organized into simulated data with iid Gaussian noise (Section V-A), simulated data with independent but spatially variant Gaussian noise (Section V-B), and InSAR data (Section V-C).

### A. Simulated Data With IID Gaussian Noise

We have simulated six data sets according to the observation model (1) with  $a = 1$  and the phase images shown in Fig. 2: first column—truncated Gaussian shaped, size  $100 \times 100$ ; second column—sinusoidal surface, size  $100 \times 100$ ; third column—discontinuous sinusoidal surface, size  $100 \times 100$ ; fourth column—mountains, size  $100 \times 100$ ; fifth column—shear planes, size  $100 \times 100$ ; and sixth column—Long’s Peak, size  $152 \times 458$ . In the latter phase image, corresponding to the digital terrain elevation model of mountainous terrain around Long’s Peak, CO, USA, distributed with book [5], the estimation is carried out only in a set of pixels signaled in the quality mask supplied with the data set.

In all of the experiments presented in this section, we ran SpInPHASE using the dictionary learned from the noisy data and the dictionary learned from the six clean images. Fig. 1, left-hand side, shows a false color image of the patches learned from the six images. The real and imaginary parts are mapped into red and green colors, respectively. Although a detailed characterization of the dictionary is beyond the scope of this paper, there are clearly patches with different spatial frequencies and orientations and patches adapted to discontinuities. The right-hand side of Fig. 1 plots the evolution of the batch and online objective functions (four Monte Carlo runs) given by (19) and (20), respectively, as a function of time.<sup>6</sup> The online version reaches values close to the minimum two orders of magnitude faster than the batch version.

Fig. 2 displays, from the top row to the bottom row, the true surface, the true interferometric phase, the noisy interferometric phase for  $\sigma = 0.7$ , the SpInPHASE estimate using the learned dictionary, the unwrapped phase from the SpInPHASE estimation, and the SpInPHASE interferometric estimation error  $\hat{\phi}_{2\pi} - \phi_{2\pi}$ .

Table I shows the performance indicators for the six surfaces and noise variances  $\sigma \in \{0.3, 0.5, 0.7, 0.9\}$  corresponding to increasing levels of difficulty from moderate to very hard. The three values shown for each row and indicator correspond, from left to right, to SpInPHASE with learned dictionary [Sp(ld)], SpInPHASE with prelearned dictionary [Sp(pd)], and WFT (W).

<sup>5</sup>SpInPHASE—<http://www.lx.it.pt/biucas/code/SpInPHASE.zip>.

<sup>4</sup>WFT—WFT—<http://www.mathworks.fr/matlabcentral/fileexchange/24892>. InSAR—InSAR—<http://www.math.u-bordeaux1.fr/cdeledal/nlsar.php>.

<sup>6</sup>The batch algorithm consists in applying alternating optimization to (19) with respect to  $\mathbf{D}$  and  $\mathbf{A}$ .

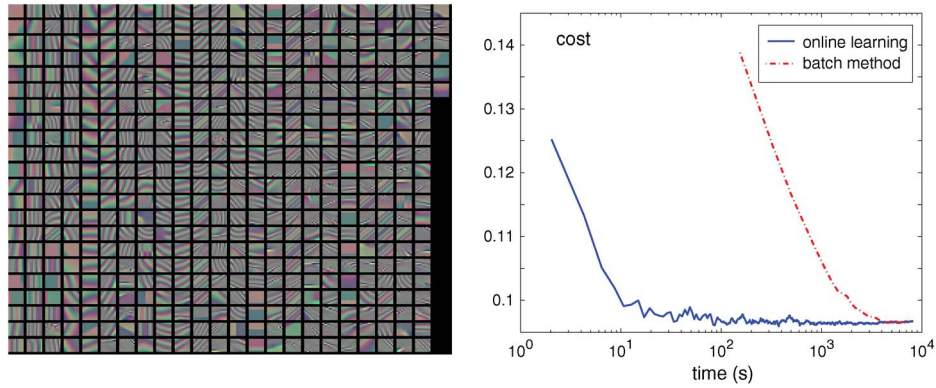


Fig. 1. (Left) Dictionary learned from the six simulated images without noise, shown in false color. (Right) Evolution of the batch and online objective functions given by (19) and (20), respectively.

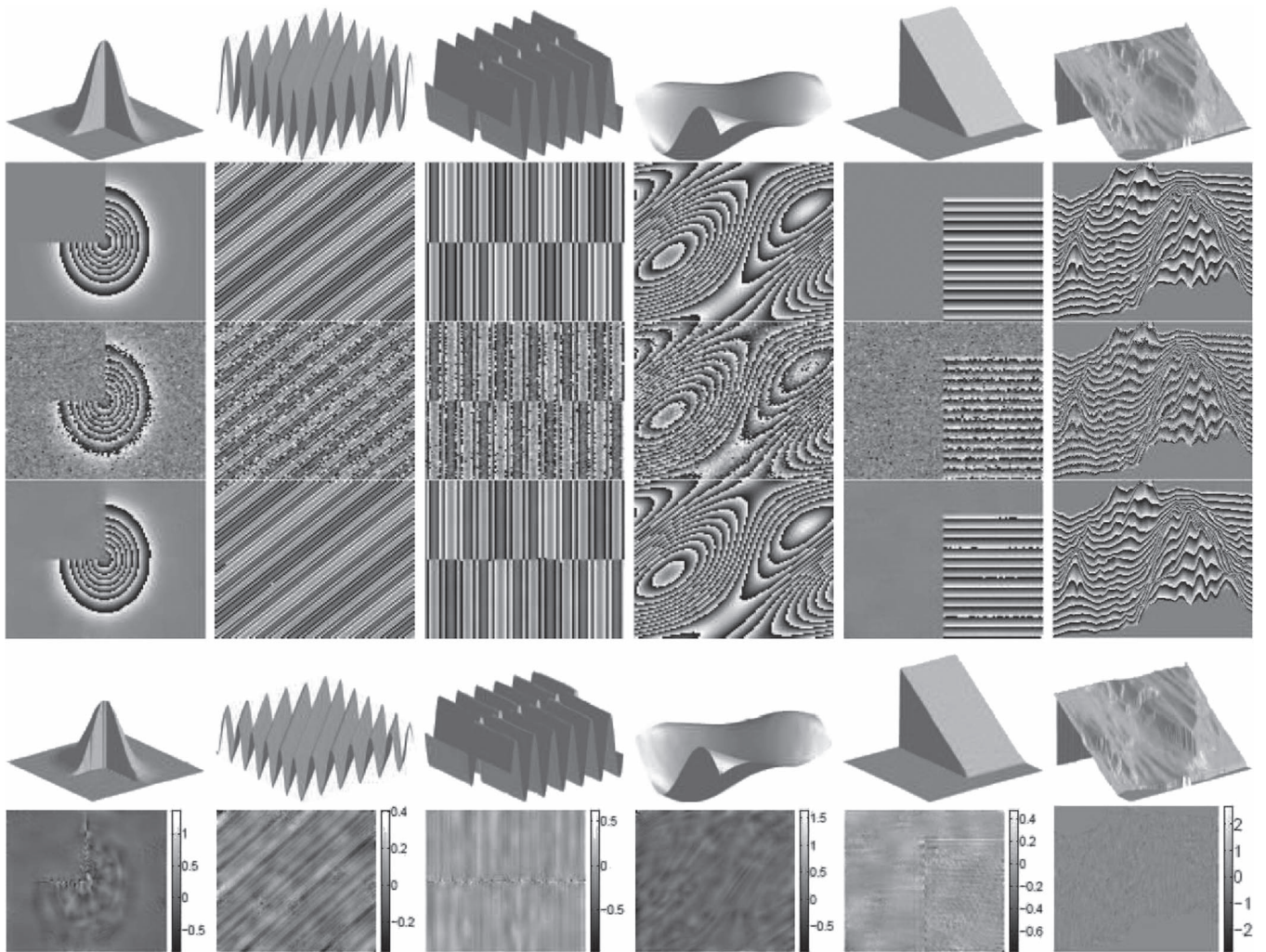


Fig. 2. Experimental results. (From top to bottom) True phase surface, the true interferometric phase, the noisy interferometric phase for  $\sigma = 0.7$ , the SpInPHASE estimate using the learned dictionary, the unwrapped phase from the SpInPHASE estimate, and the SpInPHASE interferometric estimation error. (From left to right) Truncated Gaussian, sinusoidal, discontinuous sinusoidal, mountains, shear planes, and Long's Peak.

SpInPHASE yields uniformly the best PSNR and PSNR<sub>a</sub> results. The advantage of SpInPHASE over WFT is substantial whenever the surfaces have discontinuities or nonnegligible  $n$ -order derivatives with  $n > 1$ , i.e., when the surfaces locally deviate from planes. We stress the very low values of NELP yielded by SpInPHASE and WFT, being exact zero in most

cases. Apart from Long's Peak and shear planes for  $\sigma = 0.3$ , SpInPHASE with prelearned dictionary displays the lowest computation time. Due to time spent in learning the dictionary, SpInPHASE with learned dictionary displays the highest computation time. Concerning the use of learned or prelearned dictionaries, there is no clear advantage of one strategy over



TABLE I  
PERFORMANCE INDICATORS FOR SURFACES SHOWN IN FIG. 2. SPINPHASE WITH LEARNED DICTIONARY [SP(LD)], SPINPHASE WITH PRELEARNED DICTIONARY [SP(PD)], AND WFT

| Surf.            | $\sigma$ | PSNR (dB)    |              |       | PSNR <sub>a</sub> (dB) |              |       | NELP      |           |           | TIME (s) |           |           |
|------------------|----------|--------------|--------------|-------|------------------------|--------------|-------|-----------|-----------|-----------|----------|-----------|-----------|
|                  |          | Sp(ld)       | Sp(pd)       | W     | Sp(ld)                 | Sp(pd)       | W     | Sp(ld)    | Sp(pd)    | W         | Sp(ld)   | Sp(pd)    | W         |
| Trunc.<br>Gauss. | 0.3      | 42.51        | <b>42.88</b> | 40.29 | 42.51                  | <b>42.88</b> | 40.29 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 69       | <b>6</b>  | 10        |
|                  | 0.5      | 39.63        | <b>39.95</b> | 36.71 | 39.63                  | <b>39.95</b> | 36.71 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 74       | <b>4</b>  | 10        |
|                  | 0.7      | 35.69        | <b>36.96</b> | 34.26 | 35.85                  | <b>36.98</b> | 34.37 | 8         | <b>3</b>  | 10        | 72       | <b>3</b>  | 10        |
|                  | 0.9      | 33.52        | <b>36.04</b> | 32.79 | 33.52                  | <b>36.23</b> | 32.79 | <b>0</b>  | 7         | <b>0</b>  | 72       | <b>3</b>  | 10        |
| Sinu.            | 0.3      | <b>48.94</b> | 47.77        | 35.76 | <b>48.94</b>           | 47.77        | 35.76 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 61       | <b>2</b>  | 10        |
|                  | 0.5      | 41.91        | <b>43.50</b> | 31.48 | 41.91                  | <b>43.50</b> | 31.48 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 65       | <b>2</b>  | 10        |
|                  | 0.7      | 38.44        | <b>41.20</b> | 28.90 | 38.44                  | <b>41.20</b> | 28.90 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 65       | <b>2</b>  | 10        |
|                  | 0.9      | 36.42        | <b>39.30</b> | 26.36 | 36.42                  | <b>39.30</b> | 26.36 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 63       | <b>2</b>  | 10        |
| Sinu.<br>discon. | 0.3      | <b>44.45</b> | 42.29        | 35.91 | <b>44.45</b>           | 42.29        | 35.91 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 63       | <b>6</b>  | 10        |
|                  | 0.5      | <b>39.41</b> | 38.61        | 31.86 | <b>39.41</b>           | 38.61        | 31.86 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 72       | <b>3</b>  | 10        |
|                  | 0.7      | <b>37.09</b> | 35.95        | 29.86 | <b>37.09</b>           | 35.95        | 29.95 | <b>0</b>  | <b>0</b>  | 1         | 71       | <b>2</b>  | 10        |
|                  | 0.9      | <b>34.17</b> | 34.00        | 27.64 | <b>34.17</b>           | 34.00        | 27.71 | <b>0</b>  | <b>0</b>  | 6         | 66       | <b>2</b>  | 10        |
| Mount.           | 0.3      | <b>40.66</b> | 38.90        | 40.00 | <b>40.66</b>           | 38.90        | 40.00 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 57       | <b>10</b> | <b>10</b> |
|                  | 0.5      | <b>37.20</b> | 35.66        | 36.55 | <b>37.20</b>           | 35.66        | 36.55 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 60       | <b>6</b>  | 10        |
|                  | 0.7      | <b>34.35</b> | 33.29        | 34.17 | <b>34.35</b>           | 33.29        | 34.17 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 62       | <b>5</b>  | 10        |
|                  | 0.9      | <b>32.55</b> | 31.66        | 32.31 | <b>32.70</b>           | 31.79        | 32.31 | 1         | 1         | <b>0</b>  | 60       | <b>4</b>  | 10        |
| Shear<br>plane   | 0.3      | <b>49.36</b> | 47.01        | 40.67 | <b>49.36</b>           | 47.01        | 40.67 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 57       | 23        | <b>10</b> |
|                  | 0.5      | <b>42.95</b> | 44.05        | 37.07 | <b>42.95</b>           | 44.05        | 37.07 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 63       | <b>2</b>  | 10        |
|                  | 0.7      | 38.39        | <b>39.58</b> | 34.13 | 38.39                  | <b>39.58</b> | 34.13 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 68       | <b>2</b>  | 10        |
|                  | 0.9      | 33.53        | <b>38.72</b> | 33.24 | 33.53                  | <b>38.72</b> | 33.24 | <b>0</b>  | <b>0</b>  | <b>0</b>  | 72       | <b>2</b>  | 10        |
| Long's<br>Peak   | 0.3      | 35.49        | <b>35.68</b> | 35.40 | 35.51                  | <b>35.69</b> | 35.41 | <b>28</b> | <b>28</b> | <b>28</b> | 515      | 179       | <b>31</b> |
|                  | 0.5      | 33.05        | <b>33.19</b> | 32.89 | 33.08                  | <b>33.24</b> | 32.93 | 32        | 33        | <b>31</b> | 357      | 77        | <b>30</b> |
|                  | 0.7      | 31.32        | <b>31.46</b> | 31.19 | 31.46                  | <b>31.53</b> | 31.28 | <b>26</b> | 48        | 32        | 326      | 42        | <b>30</b> |
|                  | 0.9      | 29.97        | <b>30.17</b> | 29.90 | 30.09                  | <b>30.26</b> | 29.99 | 34        | <b>32</b> | 35        | 308      | <b>27</b> | 30        |

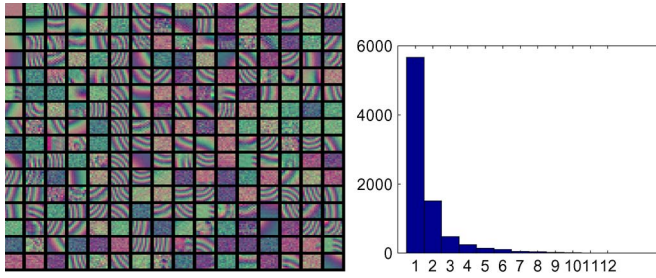


Fig. 3. Gaussian-shaped surface. (Left) Complex-valued dictionary. (Right) Histogram of the number of nonzero elements in the code for a patch.

the other, being the results given by both comparable for most cases.

Fig. 3 shows, in the left-hand side, the complex-valued learned dictionary for the truncated Gaussian surface with  $\sigma = 0.7$  and, in the right-hand side, plots the histogram of the number nonzero coefficients in the code vectors  $\hat{\alpha}_i$ , for  $i = 1, \dots, N_p$ . From the histogram, we conclude, as expected, that the patches are represented by the linear combination of a very small number of atoms, most of them just by one atom. It gives evidence that the patch approximations (recall that the patch size is  $m = 100$ ) are indeed very sparse.

### B. Simulated Data With Spatially Variant Gaussian Noise

SpInPHASE was conceived to work with iid noise. Therefore, when the noise variable is spatially variant, SpInPHASE cannot be applied. It is, however, trivial converting spatially variant into identically distributed noise, provided that the variance is known: It amounts to divide the observations  $z_i$ , for  $i = 1, \dots, N$ , by the variance  $\sigma_i$  producing equivalent

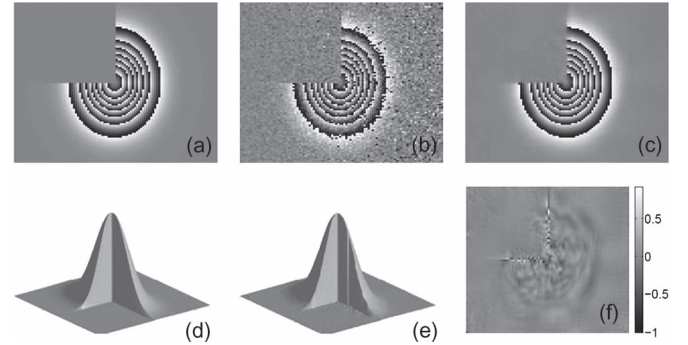


Fig. 4. Truncated Gaussian surface observed under independent but spatial variant Gaussian noise: The noise standard deviation varies linearly along the left-right direction from 0.3 to 0.9. (Top and from left to right) True, noisy, and SpInPHASE estimates using the learned dictionary. (Bottom and from left to right) True phase, unwrapped phase, and the SpInPHASE interferometric error  $\hat{\phi}_{2\pi} - \phi_{2\pi}$ .

observations with unit variance. Fig. 4 illustrates the result of such procedure. The true image is the truncated Gaussian, and the noise standard deviation varies linearly along the left-right direction from 0.3 to 0.9, as it can be perceived from the noisy interferometric phase shown in Fig. 4(b). The estimated interferometric phase shown in Fig. 4(c) is qualitatively of very good quality in terms of denoising and detail preservation, yielding PSNR = 37.15 dB, PSNR<sub>a</sub> = 37.15 dB, NELP = 0, and TIME = 76 s, which are between values shown in Table I for  $\sigma \simeq 0.5$  and  $\sigma \simeq 0.7$ . Moreover, although the noise increases from the left to right of the image, the perceived quality of the interferometric estimate is almost constant along the left-right direction. We have applied WFT to the same noisy data having obtained PSNR = 36.10 dB, PSNR<sub>a</sub> = 36.40 dB, NELP = 10, and TIME = 10 s.



TABLE II  
PERFORMANCE INDICATORS FOR THE TRUNCATED  
GAUSSIAN SURFACE AND INSAR NOISE

| Indicator              | Algorithm | coherence    |              |              |              |
|------------------------|-----------|--------------|--------------|--------------|--------------|
|                        |           | 0.95         | 0.9          | 0.85         | 0.8          |
| PSNR (dB)              | NL-InSAR  | 31.70        | 31.69        | 31.68        | 28.97        |
|                        | SpInPHASE | <b>38.00</b> | <b>35.57</b> | <b>33.48</b> | <b>31.74</b> |
| PSNR <sub>a</sub> (dB) | NL-InSAR  | 32.09        | 32.49        | 32.82        | 31.52        |
|                        | SpInPHASE | <b>38.00</b> | <b>36.05</b> | <b>33.67</b> | <b>32.99</b> |
| NELP                   | NL-InSAR  | 23           | 45           | 24           | 202          |
|                        | SpInPHASE | <b>0</b>     | <b>24</b>    | <b>12</b>    | <b>95</b>    |
| TIME (s)               | NL-InSAR  | <b>34.07</b> | <b>33.43</b> | <b>32.60</b> | <b>32.07</b> |
|                        | SpInPHASE | 368.63       | 362.56       | 380.97       | 365.11       |

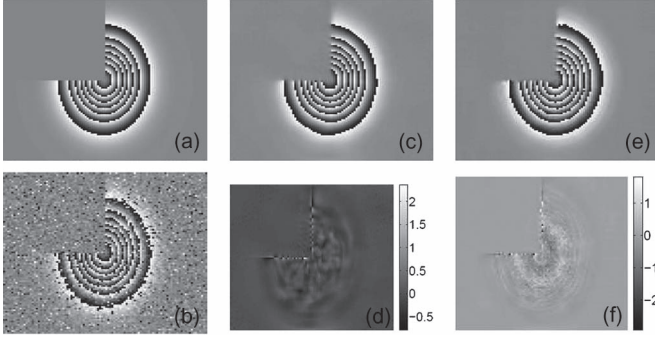


Fig. 5. Estimation results for the truncated Gaussian elevation with InSAR noise generated by a constant coherence of 0.9. (a) Original wrapped phase image. (b) Noisy phase image. (c) SpInPHASE estimate. (d) SpInPHASE estimation error. (e) NL-InSAR estimate. (f) NL-InSAR estimation error.

### C. InSAR Data

Table II shows the performance indicators for SpInPHASE and NL-InSAR [15] algorithms, for different values of  $\gamma$ , in the estimation interferometric phase associated with the truncated Gaussian-shaped surface shown in Fig. 5. The noise observations were generated according to the InSAR observation model described previously with  $c_{11} = c_{22} = 1$  and  $c_{12} = \gamma e^{j\phi}$ . SpInPHASE parameters are as in Section V-A apart from the window size which we set to  $12 \times 12$ . NL-InSAR parameters are as described in [15]: search window size =  $21 \times 21$ , patch size =  $7 \times 7$ , minimum equivalent number of looks  $L = 10$ ,  $h = 12$ ,  $T = 0.2 \times (7 \times 7)$ , and the number of iterations = 10. Since the publicly available NL-InSAR code<sup>7</sup> is for 32-b machines with a Windows system, all results presented in this section are based on a machine with Core 2 Duo CPU E6550 and 2 GB. SpInPHASE shows a clear advantage over NL-InSAR with respect to PSNR, PSNR<sub>a</sub>, and NELP. Regarding time, SpInPHASE is about ten times slower than NL-InSAR. Fig. 5 shows the denoised result when coherence is 0.9. Although both estimates are similar at the naked eye, a detailed analysis of the estimation errors reveals a higher NL-InSAR error with a high-frequency component not present in the SpInPHASE estimation error and a slightly better discontinuity preservation in the SpInPHASE estimate, underlying lower values of the NELP performance indicator.

The next two experiments are based on the InSAR data distributed with the book [5]. The data sets were generated based on a real digital elevation model of mountainous terrain

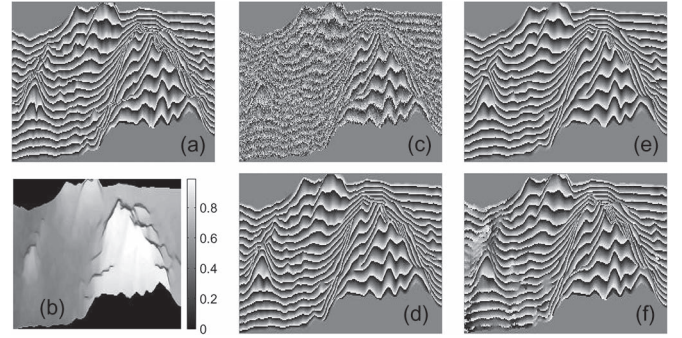


Fig. 6. Estimation results for the real digital elevation model data around Long's Peak with InSAR noise. (a) Original wrapped phase from the digital elevation model. (b) Coherence. (c) Noisy phase. (d) SpInPHASE estimate. (e) WFF estimate. (f) NL-InSAR estimate.

around Long's Peak and Isolation Peak, CO, USA, using a high-fidelity InSAR simulator that models the SAR point spread function, the InSAR geometry, the speckle noise, and the lay-over and shadow phenomena. For a detailed description of the simulator, see [5, Ch. 3] and the references therein. In order to make the estimation problem more challenging, we have regenerated the data according to the InSAR statistics presented in the beginning of this section using the supplied phase and coherence images and setting the power of the two complex images to  $c_1 = c_2 = 1 + \gamma$ . In this way, we force the image discontinuities to be colocated in the amplitude, coherence, and interferometric phase, rendering a challenging inference problem.

To apply SpInPHASE to InSAR data, we need an estimate of the coherence. We have observed that the estimate need not be very accurate in order to get useful interferometric phase results. Herein, we have adopted the estimator proposed in [5] given by

$$\hat{\gamma}_i = \frac{\sqrt{\left(\sum_{j \in w_i} \cos \phi_j\right)^2 + \left(\sum_{j \in w_i} \sin \phi_j\right)^2}}{K^2} \quad (26)$$

where  $w_i$  is a  $K \times K$  window centered at pixel  $i$ . To preserve the coherence details, we use  $K = 3$ , which is a small window, thus enforcing little low-pass filtering. The downside of using such a small window is a relatively large variance. Aiming at a cleaner estimate, we denoise it with the state-of-the-art BM3D algorithm [18], which, in real-world images, preserves the details and is extremely fast.

The first experiment regards the Long's Peak data set. The noisy phase and the denoised results with true coherence are shown in Fig. 6. The obtained performance indexes for SpInPHASE, WFT, and NL-InSAR are, by the same order, PSNR = (25.99, 24.71, 21.99) dB, PSNR<sub>a</sub> = (26.29, 25.51, 23.44) dB, and NELP = (142, 3082, 3414). With estimated coherence, SpInPHASE yields PSNR = 26.11 dB, PSNR<sub>a</sub> = 26.41 dB, and NELP = 164, which are close to those obtained with known coherence. SpInPHASE yields the best results with respect to the three indicators. Fig. 6 shows (a) the original wrapped phase, (b) the coherence estimate provided in [5], (c) the noisy phase, (d) the SpInPHASE estimate, (e) the WFF estimate, (f) the NL-InSAR estimate. Fig. 7 shows the

<sup>7</sup><http://www.math.u-bordeaux1.fr/cdeledal/nlsar>.

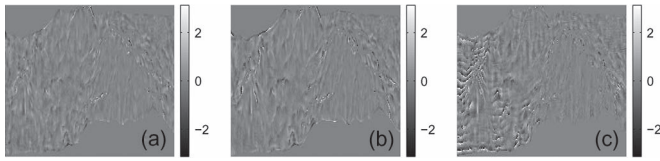


Fig. 7. Estimation errors for the real digital elevation model data around Long's Peak with InSAR noise. (a) SpInPHASE. (b) WFT. (c) NL-InSAR.

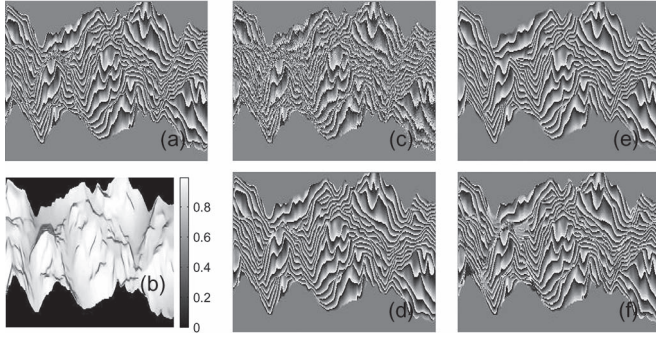


Fig. 8. Estimation results for the real digital elevation model data around Isolation Peak with InSAR noise. (a) Original wrapped phase from the digital elevation model. (b) Coherence. (c) Noisy phase. (d) SpInPHASE estimate. (e) WFF estimate. (f) NL-InSAR estimate.

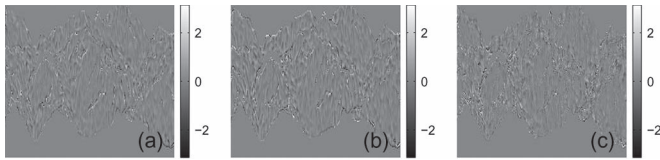


Fig. 9. Estimation error for the real digital elevation model data around Isolation Peak with InSAR noise. (a) SpInPHASE. (b) WFT. (c) NL-InSAR.

estimation errors for the three algorithms. The SpInPHASE advantage over NL-InSAR is evident. Compared with WFF, SpInHASE is performing better as well, although the advantage is more subtle. A careful analysis of Fig. 7 leads to the conclusion that the SpInPHASE advantage is mainly located in areas of discontinuities, which explains its lower NELP values compared with the competitors.

We now do as above for the Isolation Peak data set. The estimation results and estimation errors with true coherence are shown in Figs. 8 and 9, respectively. The obtained performance indexes for SpInPHASE, WFT, and NL-InSAR are, by the same order, PSNR = (24.00, 22.83, 23.14) dB, PSNR<sub>a</sub> = (24.73, 23.12, 24.03) dB, and NELP = (1990, 19 427, 7597). With estimated coherence, SpInPHASE yields PSNR = 23.69 dB, PSNR<sub>a</sub> = 24.44 dB, and NELP = 1755. The conclusions are in line with those of the Long's Peak data set.

In the aforementioned two experiments, we highlight the small differences in PSNR obtained with the true coherence and the estimated one. The proximity of the two estimates provides experimental support for the proposed InSAR variance normalization. In addition, we highlight the SpInPHASE ability to preserve the interferometric information coded in discontinuities and areas of high phase rate, which is an essential requirement for the success of phase unwrapping.

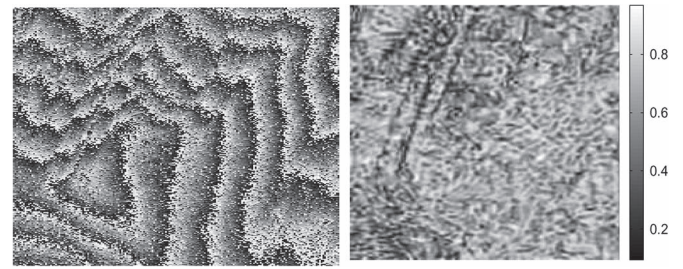


Fig. 10. Real interferometric phase near Evaggelistris, Greece. (Left) Noisy interferogram. (Right) Estimated coherence.

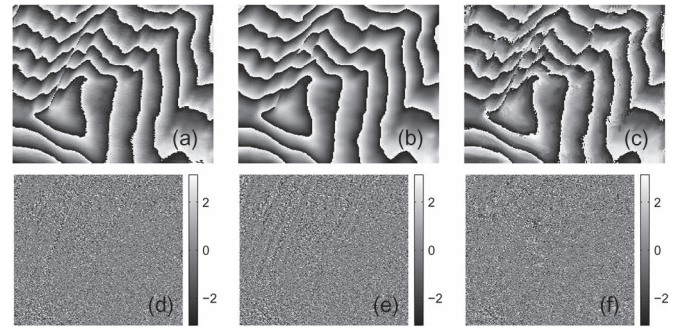


Fig. 11. Denoised results and errors between the estimate and the noisy phase of real interferometric phase near the city of Evaggelistris, Greece. (a) SpInPHASE. (b) WFT. (c) NL-InSAR. (d)–(f) Corresponding errors to the noisy phase.

Our final experiments are based on two real European Remote Sensing Satellite (ERS)-1/ERS-2 InSAR SLC image pairs distributed<sup>8</sup> by the European Space Agency (ESA). The first data set corresponds to an area from near Evaggelistris, Greece (from 38°18'30"N to 38°19'32"N and from 23°01'15"E to 23°04'24"E). The master and slave images were acquired by ERS-1 on April 29, 1996 (orbit 25 045, track 7) and ERS-2 on April 30, 1996 (orbit 5372, track 7), respectively. The size of the processed subset is 250 (azimuth) × 200 (range) pixels, and the spatial resolution is about 4 m in azimuth and 8 m in range.

The coherence is estimated as described earlier. This data set is quite challenging as it can be perceived from the highly noisy interferometric phase shown in the left-hand side of Fig. 10 and from the low values of the estimated coherence shown in the right-hand side of Fig. 10.

The estimates produced by SpInPHASE, WFT, and NL-InSAR and the corresponding errors between the estimate and the noisy phase are shown in Fig. 11. Given that we do not have the true surface, our analysis is mostly qualitative. WFT produces the smoother results but washes out the discontinuities present along the direction connecting the top of the mountain toward the top of the image. On the contrary, NL-InSAR preserves the discontinuities but retains a considerable amount of noise. SpInPHASE yields the best balance between discontinuity preservation and noise reduction. To further assess the quality of the estimates, we display the images of  $\hat{\phi}_{2\pi} - \psi = (\hat{\phi}_{2\pi} - \phi_{2\pi}) + \epsilon$ . If a given estimator oversmooths the discontinuities, then its estimation error  $(\hat{\phi}_{2\pi} - \phi_{2\pi})$  displays

<sup>8</sup><http://eo-virtual-archive4.esa.int>.



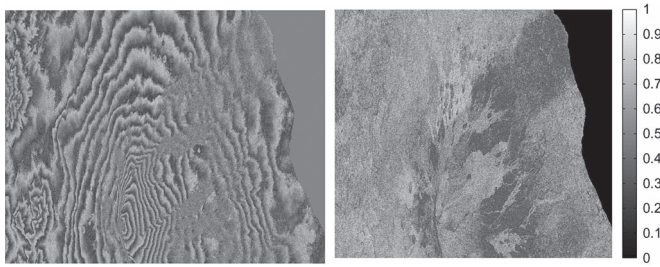


Fig. 12. Real interferometric phase from ETNA volcano, Sicily, Italy. (Left) Noisy interferogram. (Right) Estimated coherence.

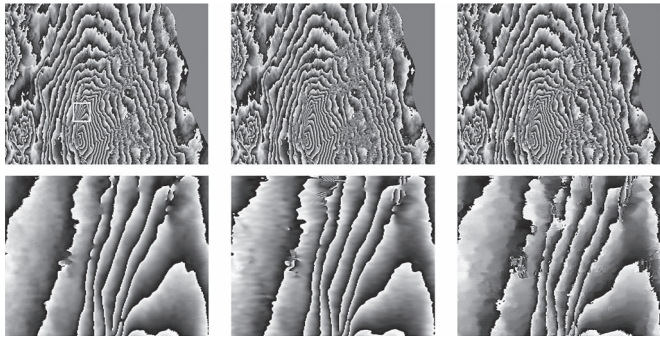


Fig. 13. Denoised results for the real interferometric phase from ETNA volcano, Sicily, Italy, shown in Fig. 12. (Top and from left to right) SpInPHASE, WFT, and NL-InSAR. (Bottom and from left to right) SpInPHASE, WFT, and NL-InSAR estimates of the interferometric phase in the area delimited by the white square shown in the top left image.

larger values in the areas of discontinuities, which, in turn, are perceived in the images of  $(\hat{\phi}_{2\pi} - \phi_{2\pi}) + \epsilon$ , as  $\epsilon$  is uncorrelated. Comparing the images (d), (e), and (f) in Fig. 11, we conclude that SpInPHASE yields the less structured error and then the smallest bias.

The second data set corresponds to an area centered at Etan volcano, Sicily, Italy (from  $37^{\circ}33'32''\text{N}$  to  $37^{\circ}40'41''\text{N}$  and from  $14^{\circ}43'55''\text{E}$  to  $15^{\circ}14'31''\text{E}$ ). The master and slave images were acquired by ERS-1 on August 2, 1995 (orbit 21 159, track 129) and ERS-2 on August 2, 1995 (orbit 1486, track 129), respectively. The size of the processed subset is 4800 (azimuth)  $\times$  2400 (range) pixels, and the spatial resolution is about 4 m in azimuth and 8 m in range.

Fig. 12, left-hand side, shows the interferogram of the Etna data set. The mask in gray on the right-hand side of the interferogram corresponds to water. Fig. 12, right-hand side, shows the estimated coherence. From these two images, we conclude that this data set is even more challenging than the previous one as a significant percentage of the image has coherence less than 0.4 and a corresponding interferometric noise variance higher than 1.4.

Fig. 13 shows in the first row, from left to right, the SpInPHASE, WFT, and NLInSAR estimates. The second row shows the estimation results of the area delimited by the white square shown in the top left image. The order of the estimates is as aforementioned. The conclusions that we may take from these results are as before: SpInPHASE yields the best balance between discontinuity/high-phase-rate preservation and noise reduction.

## VI. CONCLUDING REMARKS

This paper has introduced SpInPHASE, an effective algorithm for interferometric phase image estimation, i.e., the estimation of phase modulo- $2\pi$  images from sinusoidal  $2\pi$ -periodic and noisy observations. The true problem was recast as the estimation of the true complex-valued image via sparse representation of the complex image patches on learned synthesis dictionaries. The sparse representation also termed sparse coding, is computed by the OML algorithm in the complex domain. The synthesis dictionaries are learned via matrix factorization with sparsity constraints promoted by the  $\ell_1$  norm regularizer. With the objective of reducing the DL time, a recently introduced ODL version of the matrix factorization was adapted to our problem. The  $\ell_2 - \ell_1$  sparse coding optimization is implemented with an instance of the SALSA. The obtained ODL algorithm allows one to learn dictionaries on the order of  $150 \times 250$  in the complex domain from images of  $512 \times 512$  in times on the order of 4 min, which is orders of magnitude faster than the batch version and much faster than the version without SALSA.

In a series of experiences with simulated data, SpInPHASE produced systematically better estimates than the state-of-the-art, in many cases with a significant advantage. The first results obtained with real InSAR data are equally encouraging. We highlight the SpInPHASE ability to preserve the interferometric information coded in discontinuities and areas of high phase rate, which is an essential requirement for the success of phase unwrapping. Nevertheless, a full evaluation of the relative SpInPHASE merit requires more experiments using, namely, high-resolution InSAR images. Another point requiring further investigation is the impact of discarding the amplitude in the modified observation model.

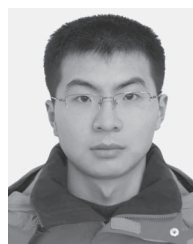
As future work, we will develop research efforts to decrease the estimation time. We are convinced that a suitable coding of the ODL and OMP steps, for example, in C++, will reduce the time that the algorithm takes by an order of magnitude. Another area where we will develop efforts is DL under more complex observation mechanisms in which we do not have direct access to the noisy interferometric phase as it is the case, for example, in various optical interferometric applications.

## REFERENCES

- [1] L. Graham, "Synthetic interferometer radar for topographic mapping," *Proc. IEEE*, vol. 62, no. 6, pp. 763–768, Jun. 1974.
- [2] H. Zebker and R. Goldstein, "Topographic mapping from interferometric synthetic aperture radar observations," *J. Geophys. Res.*, vol. 91, no. B5, pp. 4993–4999, Apr. 1986.
- [3] P. Lauterbur, "Image formation by induced local interactions: Examples employing nuclear magnetic resonance," *Nature*, vol. 242, no. 5394, pp. 190–191, Mar. 1973.
- [4] M. Hedley and D. Rosenfeld, "A new two-dimensional phase unwrapping algorithm for MRI images," *Magn. Reson. Med.*, vol. 24, no. 1, pp. 177–181, Mar. 1992.
- [5] D. Ghiglia and M. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software*. Hoboken, NJ, USA: Wiley, 1998.
- [6] T. Kreis, *Handbook of Holographic Interferometry: Optical and Digital Methods*. Weinheim, Germany: Wiley-VCH, 2005.
- [7] A. Patil and P. Rastogi, "Moving ahead with phase," *Opt. Lasers Eng.*, vol. 45, no. 2, pp. 253–257, Feb. 2007.
- [8] N. Karpinsky and S. Zhang, "High-resolution, real-time 3D imaging with fringe analysis," *J. Real-Time Image Process.*, vol. 7, no. 1, pp. 55–66, Mar. 2012.



- [9] C. David, B. Nohammer, H. Solak, and E. Ziegler, "Differential X-ray phase contrast imaging using a shearing interferometer," *Appl. Phys. Lett.*, vol. 81, no. 17, pp. 3287–3289, Oct. 2002.
- [10] V. Katkovnik, K. Egiazarian, and J. Astola, *Local Approximation Techniques in Signal and Image Processing*. Bellingham, WA, USA: SPIE, 2006, ser. SPIE field guides.
- [11] Q. Kemao, "Two-dimensional windowed Fourier transform for fringe pattern analysis: Principles, applications and implementations," *Opt. Lasers Eng.*, vol. 45, no. 2, pp. 304–317, Feb. 2007.
- [12] Q. Kemao, L. Nam, L. Feng, and S. Soon, "Comparative analysis on some filters for wrapped phase maps," *Appl. Opt.*, vol. 46, no. 30, pp. 7412–7418, Oct. 2007.
- [13] J. Bioucas-Dias, V. Katkovnik, J. Astola, and K. Egiazarian, "Absolute phase estimation: Adaptive local denoising and global unwrapping," *Appl. Opt.*, vol. 47, no. 29, pp. 5358–5369, Oct. 2008.
- [14] J. Bioucas-Dias and G. Valadão, "Phase unwrapping via graph cuts," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 698–709, Mar. 2007.
- [15] C.-A. Deledalle, L. Denis, and F. Tupin, "NL-InSAR: Non-local interferogram estimation," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 4, pp. 1441–1452, Apr. 2011.
- [16] A. Efros and T. Leung, "Texture synthesis by non-parametric sampling," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, 1999, vol. 2, pp. 1033–1038.
- [17] A. Buades, B. Coll, and J. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE CVPR*, 2005, vol. 2, pp. 60–65.
- [18] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
- [19] J. Bioucas-Dias and J. Leitão, "The  $Z\pi M$  algorithm: A method for interferometric image reconstruction in SAR/SAS," *IEEE Trans. Image Process.*, vol. 11, no. 4, pp. 408–422, Apr. 2002.
- [20] V. Katkovnik, J. Astola, and K. Egiazarian, "Phase local approximation (PhaseLa) technique for phase unwrap from noisy data," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 833–846, Jun. 2008.
- [21] G. Nico and J. Fortuny-Guasch, "Using the matrix pencil method to solve phase unwrapping," *IEEE Trans. Signal Process.*, vol. 51, no. 3, pp. 886–888, Mar. 2003.
- [22] M. Rivera and J. Marroquin, "Half-quadratic cost functions for phase unwrapping," *Opt. Lett.*, vol. 29, no. 5, pp. 504–506, Mar. 2004.
- [23] G. Valadão and J. Bioucas-Dias, "CAPE: Combinatorial absolute phase estimation," *J. Opt. Soc. Amer. A, Opt. Image Sci.*, vol. 26, no. 9, pp. 2093–2106, Sep. 2009.
- [24] J. Leitão and M. Figueiredo, "Absolute phase image reconstruction: A stochastic nonlinear filtering approach," *IEEE Trans. Image Process.*, vol. 7, no. 6, pp. 868–882, Jun. 1998.
- [25] O. Loffeld, H. Nies, S. Knedlik, and Y. Wang, "Phase unwrapping for SAR interferometry—A data fusion approach by Kalman filtering," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 1, pp. 47–58, Jan. 2008.
- [26] B. Osmanoglu, T. Dixon, and S. Wdowski, "Three-dimensional phase unwrapping for satellite radar interferometry, I: DEM generation," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 2, pp. 1059–1075, Feb. 2014.
- [27] J. Martinez-Espla, T. Martinez-Marin, and J. Lopez-Sanchez, "Using a grid-based filter to solve InSAR phase unwrapping," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 147–151, Apr. 2008.
- [28] J. Martinez-Espla, T. Martinez-Marin, and J. Lopez-Sanchez, "A particle filter approach for InSAR phase filtering and unwrapping," *IEEE Trans. Geosci. Remote Sens.*, vol. 47, no. 4, pp. 1197–1211, Apr. 2009.
- [29] J. Martinez-Espla, T. Martinez-Marin, and J. Lopez-Sanchez, "An optimized algorithm for InSAR phase unwrapping based on particle filtering, matrix pencil, and region-growing techniques," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 6, no. 4, pp. 835–839, Oct. 2009.
- [30] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.
- [31] M. Elad, *Sparse and Redundant Representations*. Berlin, Germany: Springer-Verlag, 2010.
- [32] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53–69, Jan. 2008.
- [33] R. Raina, A. Battle, H. Lee, B. Packer, and A. Ng, "Self-taught learning: Transfer learning from unlabeled data," in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 759–766.
- [34] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 791–804, Apr. 2012.
- [35] S. Chen, D. Donoho, M. Saunders, and A. Michael, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.
- [36] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. 26th Annu. ICML*, 2009, pp. 689–696.
- [37] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2345–2356, Sep. 2010.
- [38] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Stat.*, vol. 32, no. 2, pp. 407–499, 2004.
- [39] R. Bamler and P. Hartl, "Synthetic aperture radar interferometry," *Inverse Probl.*, vol. 14, no. 4, pp. R1–R54, Aug. 1998.
- [40] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, vol. 1. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.
- [41] M. Elad, "Sparse and redundant representation modeling—What next?" *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 922–928, Dec. 2012.
- [42] J. Mairal *et al.*, "Non-local sparse models for image restoration," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 2272–2279.
- [43] K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.
- [44] D. Just and R. Bamler, "Phase statistics of interferograms with applications to synthetic aperture radar," *Appl. Opt.*, vol. 33, no. 20, pp. 4361–4368, Jul. 1994.
- [45] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. R. Stat. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.
- [46] Y. Pati, R. Rezaifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput.*, 1993, vol. 1, pp. 40–44.
- [47] T. Blumensath and M. Davies, "Iterative hard thresholding for compressed sensing," *Appl. Comput. Harmon. Anal.*, vol. 27, no. 3, pp. 265–274, Nov. 2009.
- [48] S. Foucart, "Hard thresholding pursuit: An algorithm for compressive sensing," *SIAM J. Numer. Anal.*, vol. 49, no. 6, pp. 2543–2563, Nov. 2011.
- [49] J. Vila and P. Schniter, "Expectation-maximization Bernoulli-Gaussian approximate message passing," in *Conf. Rec. 45th ASIOMAR Conf. Signals, Syst. Comput.*, 2011, pp. 799–803.
- [50] D. Donoho and M. Elad, "Optimal sparse representation in general (non-orthogonal) dictionaries via  $\ell_1$  minimization," *Proc. Nat. Acad. Sci. USA*, vol. 100, no. 5, pp. 2197–2202, Mar. 2003.
- [51] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, Aug. 2006.
- [52] A. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1," *Vis. Res.*, vol. 37, no. 23, pp. 3311–3326, Dec. 1997.
- [53] M. Lewicki and T. Sejnowski, "Learning overcomplete representations," *Neural Comput.*, vol. 12, no. 2, pp. 337–365, Feb. 2000.
- [54] M. Aharon, M. Elad, and A. Bruckstein, "The K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
- [55] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient sparse coding algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, vol. 19, pp. 801–808.
- [56] A. Charles, B. Olshausen, and C. Rozell, "Learning sparse codes for hyperspectral imagery," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 5, pp. 963–978, Sep. 2011.
- [57] A. Rakotomamonjy, "Applying alternating direction method of multipliers for constrained dictionary learning," *Neurocomputing*, vol. 106, pp. 126–136, Apr. 2012.
- [58] J. Bioucas-Dias and M. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *Proc. 2nd WHISPERS*, 2010, pp. 1–4.
- [59] S. Boyd, N. Parikh, E. Chu, B. Peleato, and E. Jonathan, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.



**Hao Hongxing** received the B.S. degree in information systems from the National University of Defense Technology, Changsha, China, where he is currently developing research activities toward the Ph.D. degree. His Ph.D. research has been developed in the Instituto de Telecomunicações, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal, under the supervision of Prof. Jose Bioucas-Dias.

His research interests include image processing, phase imaging, phase estimation, and sparse representation in the complex domain.



**José M. Bioucas-Dias** (S'87–M'95) received the E.E., M.Sc., Ph.D., and “Agregado” degrees in electrical and computer engineering from the Instituto Superior Técnico (IST), the engineering school of the Technical University of Lisbon (currently Universidade de Lisboa), Lisboa, Portugal, in 1985, 1991, 1995, and 2007, respectively.

Since 1995, he has been with the Department of Electrical and Computer Engineering, IST, Universidade de Lisboa, where he was an Assistant Professor from 1995 to 2007 and has been an Associate Professor since 2007.

Since 1993, he has also been a Senior Researcher with the Pattern and Image Analysis Group, Instituto de Telecomunicações IST, Lisbon, Portugal, which is a private nonprofit research institution. His research interests include inverse problems, signal and image processing, pattern recognition, optimization, and remote sensing. He authored or coauthored more than 250 scientific publications, including more than 70 journal papers (48 of which were published in IEEE journals) and 180 peer-reviewed international conference papers and book chapters.

Dr. Bioucas-Dias was an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS (1997–2000), and he is an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING and the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING. He was a Guest Editor of the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING for the Special Issue on Spectral Unmixing of Remotely Sensed Data, the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING for the Issue on Hyperspectral Image and Signal Processing, the IEEE SIGNAL PROCESSING MAGAZINE for the Special Issue on Signal and Image Processing in Hyperspectral Remote Sensing, and the IEEE JOURNAL OF SELECTED TOPICS IN SIGNAL PROCESSING for the Issue on Advances in Hyperspectral Data Processing and Analysis. He was the General Cochair of the 3rd IEEE Geoscience and Remote Sensing Society Workshop on Hyperspectral Image and Signal Processing, Evolution in Remote Sensing (2011) and has been a member of program/technical committees of several international conferences.



**Vladimir Katkovnik** received Ph.D. and D.Sc. degrees in technical cybernetics from Leningrad Polytechnic Institute (LPI), Leningrad, Russia, in 1964 and 1974, respectively.

From 1964 to 1991, he was an Associate Professor and then Professor with the Department of Mechanics and Control Processes, LPI. From 1991 to 1999, he was a Professor with the Department of Statistics, University of South Africa, Pretoria, South Africa. From 2001 to 2003, he was a Professor with the Kwangju Institute of Science and Technology,

Gwangju, Korea. Since 2003, he has been with the Department of Signal Processing, Tampere University of Technology, Tampere, Finland. He has published six books and over 300 papers. His research interests include stochastic signal processing, linear and nonlinear filtering, nonparametric estimation, computational imaging, nonstationary systems, and time–frequency analysis.