

# 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 – that is, 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 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 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 SpInPHASE, is illustrated in a series of experiments with simulated and real data where it outperforms the state-of-the-art.

**Index Terms**—Phase estimation, interferometric phase estimation, phase unwrapping, image similarity, dictionary learning, sparse regression, online learning.

## I. INTRODUCTION

In 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/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 3D from structured light, a scene is illuminated with spatial sinusoidal light patterns and its 3D

shape is inferred from the intensity of the scattered light; in X-ray phase-contrast imaging, the differential X-rays phase shift induced by the refractive index of an object is inferred from X-ray tomographic projections; and in phase retrieval, a non-interferometric 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 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 non-convex 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; 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 the non-additive 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 is constant in small windows and compute the maximum likelihood estimate of the interferometric phase [5] in these windows. The assumption of constant phase in small windows, termed 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.

<sup>1</sup>In this work 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 or larger than  $\pi$ .

Hao Hongxing is with the College of Information Systems and Management, National University of Defense Technology, Changsha, Hunan, China, and with the Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, 1049-1, Portugal. E-mail: hongxinghao87@gmail.com.

J. M. Bioucas-Dias is with the Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, Universidade de Lisboa, 1049-1, Portugal. E-mail: bioucas@lx.it.pt

Vladimir Katkovnik is with the Signal Processing Department, Technology University of Tampere, PO Box 527, FI-33101 Tampere, Finland. E-mail: vladimir.katkovnik@tut.fi

This work was supported by the Portuguese Science and Technology Foundation, Projects PEst-OE/EEI/LA0008/2013 and PTDC/EEI-PRO/1470/2012, by the Academy of Finland, project no. 138207, 2011-2014, and by China Scholarship Council (2011611073), Doctor project of NUDT (B110503) and National Natural Science Foundation of China (61103081)



Time-frequency analysis based on Fourier transform is a conventional tool in the analysis and filtering of the wrapped phase (see [11], [12] and 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 PEARLS algorithm [13]. PEARLS implements a two-step strategy to phase estimation. In the first step, the interferometric phase is inferred based on first order LPA using adaptive window size [10]. The absolute phase is determined in the second step by running 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 first order LPA approximation for the phase surfaces, thus unable to represent, for example, a pure quadratic surface or surfaces containing discontinuities.

The NL-InSAR method introduced in [15] is the state-of-the-art in InSAR. This method uses a non-local approach for the joint estimation of the reflectivity, the interferometric phase, and the coherence images from an interferometric pair of co-registered single-look complex (SLC) SAR images. The non-local paradigm exploits the self-similarity existing in most real world images<sup>3</sup> [16] to find similar patches and then reducing their noise by applying some type of averaging. Well known examples of this approach to image denoising are the non-local means filtering [17] and the state-of-the-art block matching with 3D filtering (BM3D) [18]. Adopting the non-local 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, the similarity criterion and the 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 estimation 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 modes (ICM) 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 (ICI) 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 is that the underlying objective function is non-convex, 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 a posteriori probability (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-cuts algorithm. The success of CAPE relies on the detection of the phase discontinuities by PUMA in the first iteration the algorithm. This 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],

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



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 non-zero weights in the linear combination) of the dictionary. Therefore, the additive noise term present in 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 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 of 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  $N = 10000$ . To cope with this computational complexity, we adopt the online dictionary

learning algorithm (ODL) introduced in [36] adapted 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 state-of-the-art for discontinuity preserving ones.

### D. Contributions

The paper introduces the following contributions:

- 1) A reformulation of the interferometric estimation problem as a sparse regression in the complex domain.
- 2) An interferometric phase image estimation algorithm, termed 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 *split augmented Lagrangian shrinkage algorithm* (SALSA) introduced in [37], herein termed *sparse regression by variable splitting and augmented Lagrangian* (SpARSAL).

The 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 dictionary learning algorithm. Section V presents a series of results with real and simulated data. Finally, Section VI finishes the 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], [39]. Let us assume, for now, that the observed data at a given image pixel is given by

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

where  $a \geq 0$ ,  $n = n_I + jn_Q$  is 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 3D 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 V, 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, respectively, the true phase image, the true amplitude image, and the noise image. We define the true complex image as  $\mathbf{x} := \mathbf{a} \odot e^{j\phi}$ , where  $\odot$  stands for componentwise multiplication and  $e^{j\phi}$  is to be understood componentwise.

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 componentwise 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, respectively, the elements of  $\mathbf{x}$  and of  $\mathbf{n}$  corresponding to the  $i$ -th patch. 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

$$\hat{\mathbf{x}}_i = \mathbf{x}_i + \boldsymbol{\varepsilon}_i, \quad (5)$$

where  $\boldsymbol{\varepsilon}_i$  is the estimation error for the  $i$ -th patch. Of course, our denoising scheme makes sense if the “size” of  $\boldsymbol{\varepsilon}_i$  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 non-null element of value 1), and the following matrix and vectors:

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

Using the above definitions, we have

$$\hat{\mathbf{x}}_P = \mathbf{M} \mathbf{x} + \boldsymbol{\varepsilon}_P, \quad (6)$$

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 belong at least to one patch.

Assuming that  $\boldsymbol{\varepsilon}_P$  is zero-mean and iid, thus having 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 suboptimal 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 [41], [31]. 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 they represent [30], [32]. Dictionary learning 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, respectively, the amplitude and the complex phase patches corresponding to  $\mathbf{x}_i$  (i.e.,  $\mathbf{x}_i = \mathbf{a}_i \odot e^{j\phi_i}$ ). Assuming that  $\mathbf{a}_i$  and  $e^{j\phi_i}$  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 admit a sparse representation. A deep study of this issue is, however, out



of the scope of the 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 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 *code*, is the solution of the constrained optimization

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

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 abusively  $\ell_0$  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], [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 non-zero 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 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 NP-hard [43], to the possible non-uniqueness 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 is  $x = ae^{j\phi}$ . Let  $i_1, \dots, i_q$  be the indexes of the patches that contain this pixel and  $\hat{x}_k$  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

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

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

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 componentwise 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 the expressions (13) and (14), SpInPHASE provides estimates of both the amplitude,  $\mathbf{a}$ , and of the interferometric phase,  $\phi_{2\pi}$ . We make, however, the following remarks:

- 1) In many applications, the noise variance is spatially variant, thus violating the hypothesis taken in the observation model (1). 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, say by  $\hat{\sigma}_i$ . In this case, we obtain an estimate of the normalized amplitude  $a_i/\hat{\sigma}_i$ .

- 2) Often, only the noisy interferometric phase  $e^{j\psi} := e^{j(\phi+\varepsilon)}$ , ( $\varepsilon$  is a phase perturbation) is available or, although we may have more information, it does not fit the model (1), as it is the case of InSAR. In this case, and assuming that  $|\varepsilon| \ll \pi$ , we may consider the pseudo observation model

$$z \simeq e^{j\phi} + \underbrace{j e^{j\phi} \varepsilon}_n, \quad (15)$$

which is formally equivalent to the observation model (1) and where  $n$  plays the role of the additive perturbation. In Section V, we sketch a justification for this procedure and, in the case of InSAR data, give experimental evidence that it yields result competitive with the state-of-the-art.

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

### C. 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 is full column range. This is not the scenario 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 (LASSO) [44], which is equivalent to BPDN [35]. The other strategy consists in attacking directly the original problem using greedy algorithms such as the orthogonal basis pursuit (OMP) [45], iterative hard thresholding (IHT), [46], Hard Thresholding Pursuit [47], or approximate message passing (MP) [48].

The quality of the approximations provided by all the above 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 [49] or of the restricted isometric constants [50]. Qualitatively, the higher the mutual coherence, the lower 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 low degree of sparseness of most image patches, implying that all the algorithms referred to above 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 of [36].

Algorithm 1 shows the pseudo code 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 8,  $\mathbf{D}_S^\#$  denotes the pseudo inverse of  $\mathbf{D}_S$  and, in line 11,  $\alpha(S)$  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),  $z \in \mathbb{C}^m$  (patch),  $\delta > 0$  (tolerance)

**Output:**  $\alpha \in \mathbb{C}^k$  (code).

```

1 begin
2    $S := \emptyset, \alpha := \mathbf{0}, e := \infty$ 
3    $\mathbf{r} := z$ 
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      $\beta := \mathbf{D}_S^\# z$ 
9      $\mathbf{r} := z - \mathbf{D}_S \beta$ 
10     $e := \|\mathbf{r}\|^2$ 
11  end
12   $\alpha(S) := \beta$ 
13 end
```

---

## IV. DICTIONARY LEARNING 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 bandlimited signals). The demand for “good” dictionaries, in the sense of sparse representation, has stimulated active research in dictionary learning aiming at adapting the dictionaries to the data they represent [34], [36], [51]–[54]. 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 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:

$$\min_{\mathbf{D} \in \mathcal{C}, \alpha_1, \dots, \alpha_{N_p}} \sum_{i=1}^{N_p} (1/2) \|\mathbf{z}_i - \mathbf{D} \alpha_i\|_2^2 + \lambda \|\alpha_i\|_1, \quad (16)$$

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 (16), the optimization with respect to  $\alpha_i$  is a BPDN problem. We could have used other sparsity inducing regularizers based, for example, in 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 (16) is an instance of the matrix factorization class (see, e.g., [36] and references therein). Alternating optimization with respect to  $\mathbf{D}$  and  $\alpha_1, \dots, \alpha_{N_p}$  is the usual way of attacking these problems (16) [51], [53]–[56]. 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 non-convex, 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  $N_p = 100000$ ,  $m = 100$ , 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

$$g_t(\mathbf{D}) = \frac{1}{t} \sum_{k=1}^t (1/2) \|\mathbf{z}^k - \mathbf{D}\alpha_k\|_2^2 + \lambda \|\alpha_k\|_1, \quad (17)$$

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

For completeness, the pseudo code for the online dictionary (ODL) learning 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 (17) with respect to  $\mathbf{D}$ . The Least Angle Regression (LARS) [38] was used in [36] to solve the BPDN problem. Herein, we introduce the *sparse regression by variable splitting and augmented Lagrangian* (SpaRSAL) (see details below), 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

---

**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  $\mathbf{z}^t \equiv [\mathbf{z}_i^t \mid i = 1, \dots, \eta]$  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

```

---

$t_0$  decreases with  $t$ . This is an 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

$$\beta_t = \left(1 - \frac{1}{t}\right)^\rho, \quad t = 1, 2, \dots \quad (18)$$

in which  $\rho > 0$ .

A proof of convergence of the sequence  $\mathbf{D}_t$  generated by the Algorithm 2 to the a stationary point of (16) is given in the Proposition 4 of [36] under the hypothesis that **i**) the observed data has a finite support, **ii**) the objective functions  $g_t(\mathbf{D})$ , for  $t = 1, \dots$ , in (17) are strictly convex with lower-bounded Hessians, and **iii**) the solution of the BPDN optimization shown in line 6 of Algorithm (2) is unique. In our case, as in [36], hypothesis **iii**) 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 *split augmented Lagrangian shrinkage algorithm* (SALSA) developed in [37]. The core idea is to replace the



BPDN optimization by the constrained version

$$\begin{aligned} \min_{\alpha \in \mathbb{C}^{k \times \eta}} \quad & (1/2) \|\mathbf{z}^t - \mathbf{D}\alpha\|_F^2 + \lambda \|\mathbf{u}\|_1 \quad (19) \\ \text{subject to: } & \alpha = \mathbf{u} \end{aligned}$$

and then build the Lagrangian of (19) 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 pseudo-code is shown in Algorithm 3. We also remark the SpaRSAL is closely related [56] and has the same structure of the SUnSAL algorithm [57], developed to solve sparse unmixing regression problems, where the non-negative constraint was removed.

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 least angle regression (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.

---

**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 function  $\text{soft}(\mathbf{x}, \tau) := \max\{0, |\mathbf{x}| - \tau\} (\mathbf{x}/|\mathbf{x}|)$  (to be understood componentwise) 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 [58, 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 on the estimated interferometric phase estimate  $\hat{\phi}_{2\pi}$ , we

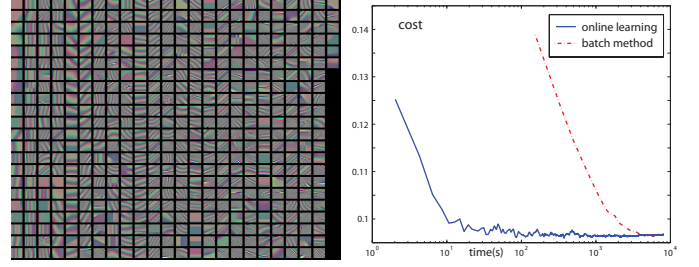


Fig. 1. Left: dictionary learned from the six simulated images without noise, shown in false color. Right: evolution of the batch and online objective functions given, respectively, by the expressions (16) and (17).

define the *peak signal-to-noise ratio* (PSNR) as

$$\text{PSNR} := 10 \log_{10} \frac{4N\pi^2}{\|\mathcal{W}(\hat{\phi}_{2\pi} - \phi_{2\pi})\|_F^2} \quad [\text{dB}], \quad (20)$$

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 error no 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 *peak signal-to-noise ratio* (PSNR<sub>a</sub>) as

$$\text{NELP} := N - |I| \quad (21)$$

$$\text{PSNR}_a := 10 \log_{10} \frac{4N\pi^2}{\|\hat{\phi}_I - \phi_I\|_F^2} \quad [\text{dB}], \quad (22)$$

where the notation  $\phi_I$  stands for the restriction of  $\phi$  to  $I$ . That is, PSNR<sub>a</sub> 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 NELP and PSNR<sub>a</sub>, 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 pre-learned dictionaries learned, for example, from an image data base. The setting of the ODL parameters is the following: regularization parameter  $\lambda = 0.11$  for all the surfaces; size of groups of patches processed simultaneously  $\eta = 0.0064 \times N$ ; exponent of the damping sequence  $\rho = 2$ ; 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 criteria for SpaRSAL is **i**) a primal and a dual residual less than  $\sqrt{k\eta}$  tol, with tol =  $10^{-3}$ , or **ii**) 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: **i**) learned from the noisy data with a number of atoms of  $k = 256$  and **ii**) pre-learned 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 with NL-InSAR [15], in the case of InSAR observations, using the codes publicly available<sup>4</sup>. WFT has two modes: **i)** the *window Fourier Filtering* (WFF), which computes the windowed Fourier transform of  $e^{j\phi}$ , hard thresholds the computed coefficients, and applies the inverse windowed Fourier transform to the thresholded coefficients; and **ii)** the *window Fourier Ridges* (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 these 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 windowed Fourier transform  $th = 3\sigma$ ; frequency interval  $[-\pi, \pi]$ ; and sampling interval 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 the algorithms ran on a personal computer equipped with a Core i7-3770 CPU and 8.00GB RAM. All the 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), real MRI data (Section V-C), and InSAR data (Section V-D).

#### 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 latter phase image, corresponding to the digital terrain elevation model of mountainous terrain around Long’s Peak, Colorado distributed with book [5], the

estimation is carried out only in set the pixels signaled in the quality mask supplied with the data set.

In all 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 the 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 (4 Monte Carlo runs) given by (16) and (17), 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 pre-learned dictionary (Sp (pd)), and WFT (W). 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 have non-negligible  $n$ -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 Peaks and the Shear Planes for  $\sigma = 0.3$ , SpInPHASE with pre-learned 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 or learned or pre-learned dictionaries, there is no clear advantage of one strategy over the other, being the results given by both comparable for most cases.

Fig. 3 shows, in left hand side, the complex valued learned dictionary for the truncated Gaussian surface with  $\sigma = 0.7$ , and, in right hand side, plots the histogram of the number non-zero 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

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

<sup>5</sup>SpInPHASE - <http://www.lx.it.pt/~bioucas/code/SpInPHASE.zip>

<sup>6</sup>The batch algorithm consists in applying alternating optimization to (16) with respect to  $\mathbf{D}$  and  $\mathbf{A}$ .



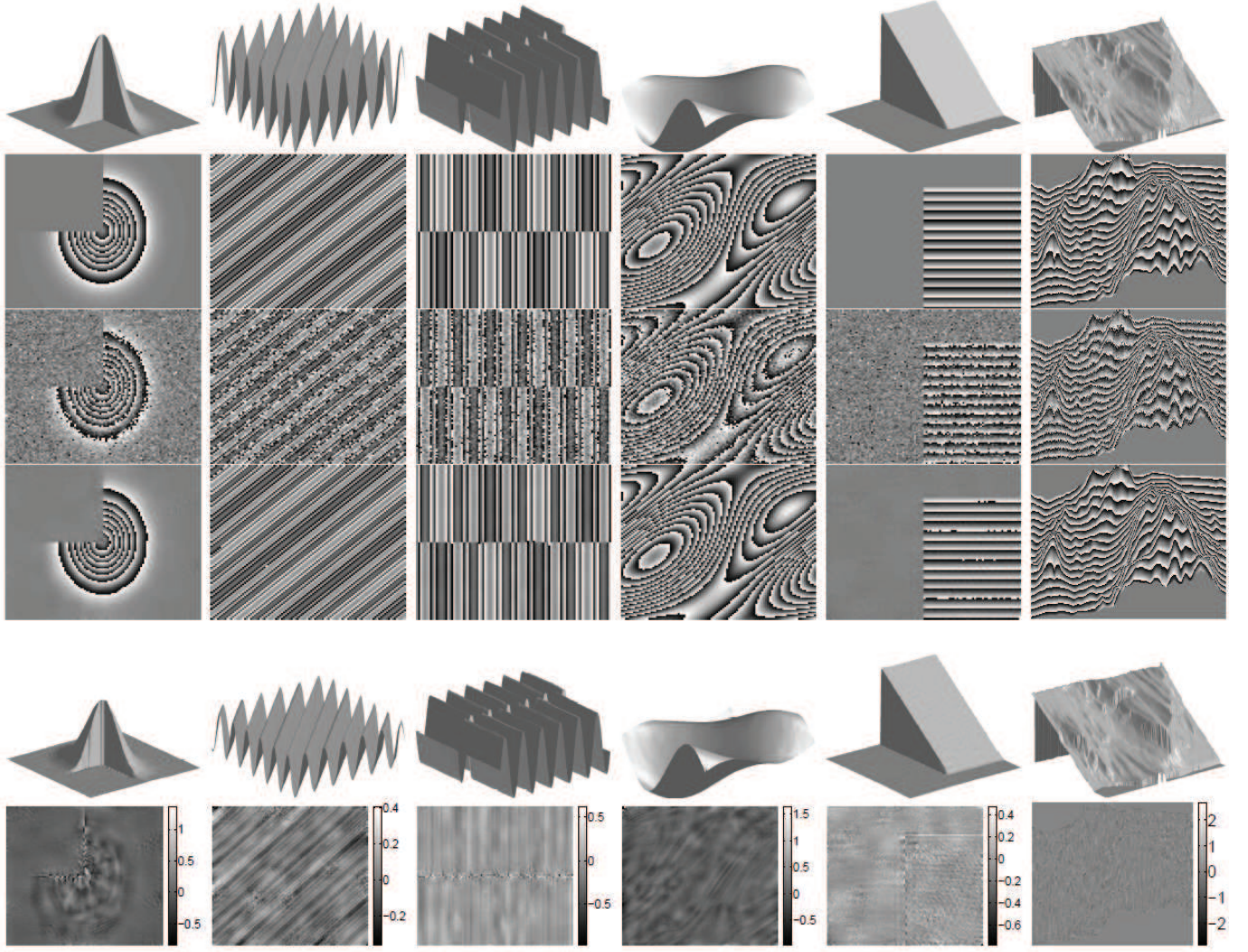


Fig. 2. Experimental results. From top to bottom: the true phase surface, the true interferometric phase, the noisy interferometric phase for  $\sigma = 0.7$ , the SpInPHASE estimate using the learned dictionary, the unwrapped phase from the SpInPHASE estimate, and the SpInPHASE interferometric estimation error. From the left to the right: truncated Gaussian, Sinusoidal, discontinuous Sinusoidal, Mountains, Shear Planes, and Long's Peak.

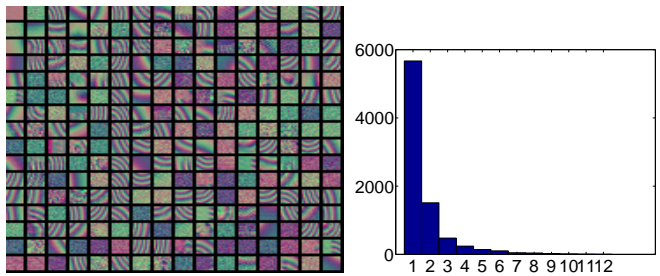


Fig. 3. Gaussian shaped surface. Left: complex valued dictionary. Right: histogram of the number of nonzero elements in the code for a patch.

can not 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 observations with unit variance. Fig. 4 illustrates the result of such procedure. The true image is the truncated Gaussian

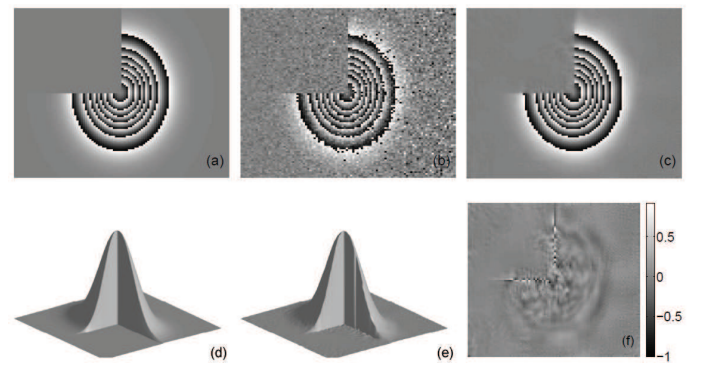


Fig. 4. Truncated Gaussian surface observed under independent but spatial variant Gaussian noise: the noise standard deviation varies linearly along the left-right directions from 0.3 to 0.9. Top and from left to right: true, noisy, and SpInPHASE estimate using the learned dictionary. Bottom and from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



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 part (b). The estimated interferometric phase shown in part (c) is qualitatively of very good quality in terms of denoising and detail preservation yielding  $\text{PSNR} = 37.15$  dB,  $\text{PSNR}_a = 37.15$  dB,  $\text{NELP} = 0$ , and  $\text{TIME} = 76$  sec, 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  $\text{PSNR} = 36.10$  dB,  $\text{PSNR}_a = 36.40$  dB,  $\text{NELP} = 10$ , and  $\text{TIME} = 10$  sec.

### C. Real MRI data

Fig. 5 shows in the top three MRI interferometric phase images of a bottle filled with liquid. The exterior of the bottle is meaningless and was set to zero. The MRI images, of size  $512 \times 228$ , were acquired using decreasing scan times from the left to right. Given that the noise level increases as the scan times decreases, the noise increases from the left to right.

The observation mechanism in MRI is well described by model (1) (see, *e.g.*, [5]). The noise was estimated from the first order horizontal and vertical differences, under the assumption that the complex valued signal varies smoothly. The obtained noise variances were, from the left to the right 0.156, 0.382, and 0.728. The SpInPHASE estimates corresponding to the top images are in the bottom. In the three cases shown in Fig. 5 the noise was removed without

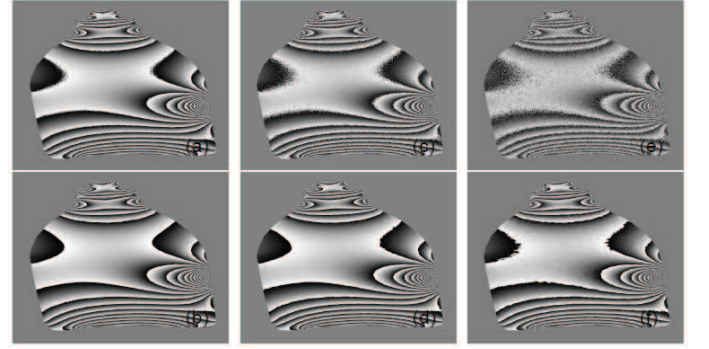


Fig. 5. Top: MRI interferometric phase images of the same bottle with increasing noise level from left to right. Bottom: Corresponding SpInPHASE estimates.

damaging the tinny fringes (image regions corresponding to a phase excursion of  $2\pi$ ) in the right hand side, bottom, and top of the bottle.

Since the left hand side image in Fig. 5 has very low level of noise compared with the right hand side one, we use it as a true image to compute the performance indicators corresponding latter. We obtained  $\text{PSNR} = 32.53$  dB. We have also computed these performance indicators for WFT having obtained  $\text{PSNR} = 31.58$  dB.

The unwrapped phase corresponding to the three noise levels of Fig. 5 are shown in Fig. 6. The phase surface is well unwrapped in the three cases even in the most difficult areas corresponding to the higher phase variations in the right hand side, bottom and top of the bottle. Underlying the quality of the

TABLE I  
PERFORMANCE INDICATORS FOR SURFACES SHOWN IN FIG. 2. SPINPHASE WITH LEARNED DICTIONARY (SP (LD)), SPINPHASE WITH PRE-LEARNED 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. Gauss.	0.3	42.51	<b>42.88</b>	40.29	42.51	<b>42.88</b>	40.29	0	0	0	69	<b>6</b>	10
	0.5	39.63	<b>39.95</b>	36.71	39.63	<b>39.95</b>	36.71	0	0	0	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	0	7	0	72	<b>3</b>	10
Sinu.	0.3	<b>48.94</b>	47.77	35.76	<b>48.94</b>	47.77	35.76	0	0	0	61	<b>2</b>	10
	0.5	<b>41.91</b>	<b>43.50</b>	31.48	41.91	<b>43.50</b>	31.48	0	0	0	65	<b>2</b>	10
	0.7	38.44	<b>41.20</b>	28.90	38.44	<b>41.20</b>	28.90	0	0	0	65	<b>2</b>	10
	0.9	36.42	<b>39.30</b>	26.36	36.42	<b>39.30</b>	26.36	0	0	0	63	<b>2</b>	10
Sinu. discon.	0.3	<b>44.45</b>	42.29	35.91	<b>44.45</b>	42.29	35.91	0	0	0	63	<b>6</b>	10
	0.5	<b>39.41</b>	38.61	31.86	<b>39.41</b>	38.61	31.86	0	0	0	72	<b>3</b>	10
	0.7	<b>37.09</b>	35.95	29.86	<b>37.09</b>	35.95	29.95	0	0	1	71	<b>2</b>	10
	0.9	<b>34.17</b>	34.00	27.64	<b>34.17</b>	34.00	27.71	0	0	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	0	0	0	57	<b>10</b>	10
	0.5	<b>37.20</b>	35.66	36.55	<b>37.20</b>	35.66	36.55	0	0	0	60	<b>6</b>	10
	0.7	<b>34.35</b>	33.29	34.17	<b>34.35</b>	33.29	34.17	0	0	0	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	0	60	<b>4</b>	10
Shear plane	0.3	<b>49.36</b>	47.01	40.67	<b>49.36</b>	47.01	40.67	0	0	0	57	23	<b>10</b>
	0.5	<b>42.95</b>	44.05	37.07	<b>42.95</b>	44.05	37.07	0	0	0	63	<b>2</b>	10
	0.7	38.39	<b>39.58</b>	34.13	38.39	<b>39.58</b>	34.13	0	0	0	68	<b>2</b>	10
	0.9	33.53	<b>38.72</b>	33.24	33.53	<b>38.72</b>	33.24	0	0	0	72	<b>2</b>	10
Long's 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



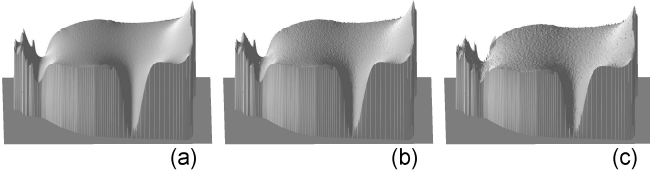


Fig. 6. Unwrapping result of denoised real MRI image. (a), (b), and (c) were obtained from the estimates shown in Fig. 5, parts (b), (d), and (f), respectively.

unwrapping is the ability of SpInPHASE to remove the noise without damaging the tinny fringes present in the referred to areas.

#### D. Interferometric SAR data (InSAR)

In InSAR, for each pixel, we are given two complex valued measurements, say  $x_1$  and  $x_2$ , of the electromagnetic field 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  $c_{11} := \mathbb{E}[|x_1|^2]$ ,  $c_{22} := \mathbb{E}[|x_2|^2]$ ,  $c_{12} := \mathbb{E}[x_1 x_2^*]$ ,  $c_{21} := c_{12}^*$ . 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* [59].

The 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])

$$\sigma_\varepsilon^2(\gamma) = \frac{\pi^3}{3} - \pi \arcsin(\gamma) + \arcsin^2(\gamma) + \frac{\text{Li}_2(\gamma)}{2}, \quad (23)$$

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  $\phi_{2\pi} = \psi$ .

Although SpInPHASE was not conceived to operate with InSAR/InSAS data, we now give 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.

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

$$z = e^{j\psi} \quad (24)$$

$$= e^{j(\phi + \varepsilon)}. \quad (25)$$

The perturbation  $\varepsilon$  in (25) is not additive. However, when  $|\varepsilon| \ll \pi$ , which in a probabilistic sense is equivalent to say

TABLE II  
PERFORMANCE INDICATORS FOR THE TRUNCATED GAUSSIAN SURFACE AND IN SAR 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

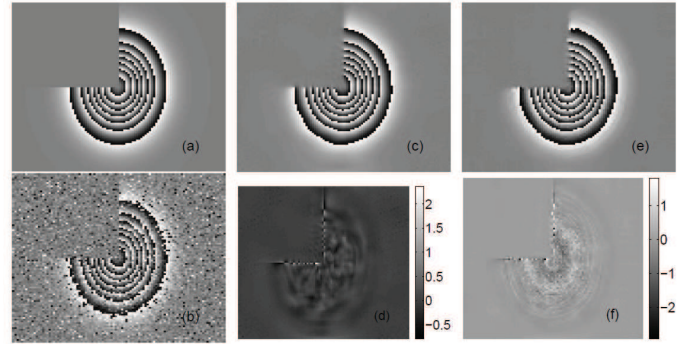


Fig. 7. Estimation results for the truncated Gaussian elevation with InSAR noise generated by a constant coherence of 0.9. (a) Original wrapped phase image, (b) Noisy phase image, (c) SpInPHASE estimate, (d) SpInPHASE estimation error, (e) NL-InSAR estimate, (f) NL-InSAR estimation error.

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

$$z \simeq e^{j\phi} + j e^{j\phi} \varepsilon. \quad (26)$$

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, so does 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. Given that SpInPHASE was designed to operate with additive and identically distributed independent noise across the image pixels, the very good results obtained with this strategy are certainly linked with the fact that the noise term  $j e^{j\phi} \varepsilon / \sigma(\gamma)$ , although non-Gaussian, is approximately additive, zero-mean, and of constant variance.

Table. II shows the performance indicators for SpInPHASE and for NL-InSAR [15] algorithms, for different values of  $\gamma$ , in the estimation interferometric phase associated with the truncated Gaussian shaped surface shown in Fig. 7. The noise observations were generated according to the InSAR observation model described above 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 paper [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>

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



is for 32 bit machines with Windows system, all the 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,  $\text{PSNR}_a$ , and NELP. Regarding time, SpInPHASE is about 10 times slower than NL-InSAR. Fig. 7 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 around Long's Peak and Isolation Peak, Colorado, using a high-fidelity InSAR simulator that models the SAR point spread function, the InSAR geometry, the speckle noise, and the layover 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 co-located in the amplitude, coherence, and in the 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

$$\hat{\gamma}_i = \frac{\sqrt{(\sum_{j \in w_i} \cos \phi_j)^2 + (\sum_{j \in w_i} \sin \phi_j)^2}}{K^2}, \quad (27)$$

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 lowpass 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. 8. The obtained performance indexes for SpInPHASE, WFT and NL-InSAR are, by the same order,  $\text{PSNR} = (25.99, 24.71, 21.99)$  dB,  $\text{PSNR}_a = (26.29, 25.51, 23.44)$  dB, and  $\text{NELP} = (142, 3082, 3414)$ . With estimated coherence, SpInPHASE yields  $\text{PSNR}=26.11$ dB,  $\text{PSNR}_a=26.41$ dB, and  $\text{NELP}=164$ , which are close to those obtained with known coherence. SpInPHASE yields the best results with respect to the three indicators. Fig. 8 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. 9 shows the estimation errors for the three algorithms. The SpInPHASE advantage over NL-InSAR is evident. Compared with WFF, SpInPHASE is

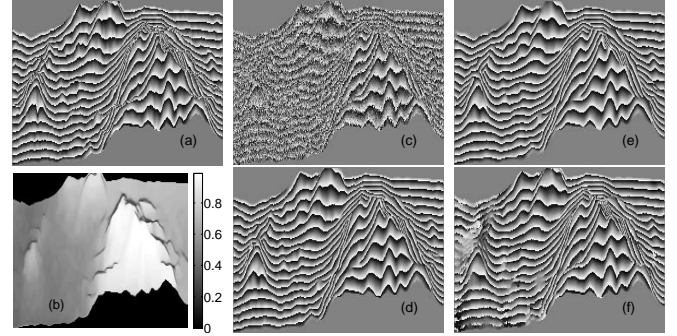


Fig. 8. Estimation results for the real digital elevation model data around Long's Peak with InSAR noise. (a) Original wrapped phase from the digital elevation model, (b) coherence, (c) noisy phase, (d) SpInPHASE estimate, (e) WFF estimate, (f) NL-InSAR estimate.

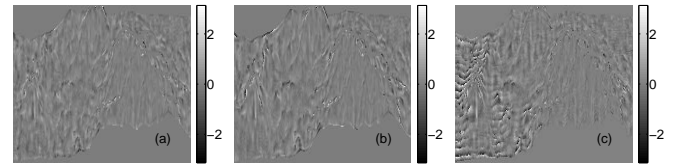


Fig. 9. Estimation errors for the real digital elevation model data around Long's Peak with InSAR noise: (a) SpInPHASE, (b) WFT, (c) NL-InSAR.

performing better as well, although the advantage is more subtle. A careful analysis of Fig. 9 leads to the conclusion that 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 Fig. 10 and Fig. 11, respectively. The obtained performance indexes for SpInPHASE, WFT and NL-InSAR are, by the same order,  $\text{PSNR} = (24.00, 22.83, 23.14)$  dB,  $\text{PSNR}_a = (24.73, 23.12, 24.03)$  dB, and  $\text{NELP} = (1990, 19427, 7597)$ . With estimated coherence, SpInPHASE yields  $\text{PSNR} = 23.69$  dB,  $\text{PSNR}_a = 24.44$  dB and  $\text{NELP} = 1755$ . The conclusions are inline with those of the Long's Peak data set.

In the above two experiments, we highlight the small PSNR differences obtained with the true coherence and with the estimated one:  $-0.12$  dB and  $+0.31$  dB for the, respectively, Long's Peak and the Isolation Peak data sets. This proximity of the two estimates provides experimental support for the proposed InSAR variance normalization. The fact that we got a negative number in the first dataset is probably due to the  $K \times K$  window present in (27), which yields an estimated of the coherence image smoother than the original one. Nevertheless, 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.

Our final experiment is based on a subset of a real ERS-1/ERS-2 InSAR pair distributed<sup>8</sup> by the European Space Agency (ESA). The area is from near Evaggelistris, Greece

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



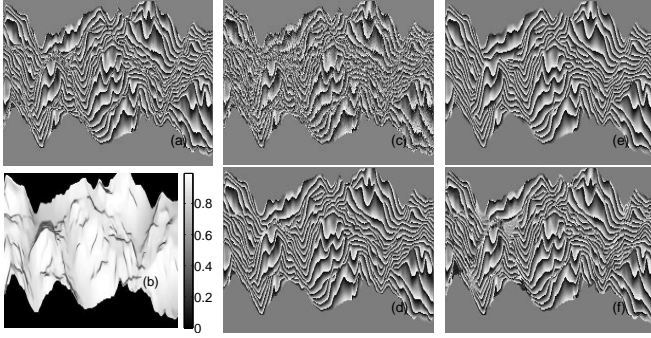


Fig. 10. Estimation results for the real digital elevation model data around Isolation Peak with InSAR noise: (a) Original wrapped phase from the digital elevation model, (b) coherence, (c) noisy phase, (d) SpInPHASE estimate, (e) WFF estimate, (f) NL-InSAR estimate.

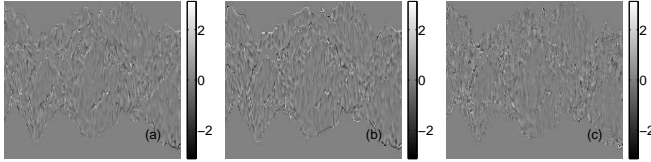


Fig. 11. Estimation error for the real digital elevation model data around Isolation Peak with InSAR noise: (a) SpInPHASE, (b) WFT, (c) NL-InSAR.

(from  $38^{\circ}18'30''N$  to  $38^{\circ}19'32''N$  and from  $23^{\circ}01'15''E$  to  $23^{\circ}04'24''E$ ). The master and slave images were acquired, respectively, by ESR-1 on 29th April 1996 (orbit 25045, track 7) and by ERS-2 on 30th April 1996 (orbit 5372, track 7). The size of the processed subset is  $250$  (along track)  $\times$   $200$  (across track) and the resolution is about  $25$  m in both dimensions.

The coherence is estimated as described above. This data set is quite challenging as it can be perceived from the highly noisy interferometric phase shown in the left hand of Fig. 12 and from the low values of the estimated coherence shown in the right hand of Fig. 12.

The estimates produced by SpInPHASE, WFT, and NL-InSAR and the corresponding errors between the estimate and the noisy phase are shown in Fig. 13. 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 towards the top of the image. On the contrary, NL-InSAR preserves the discontinuities but retains a considerable

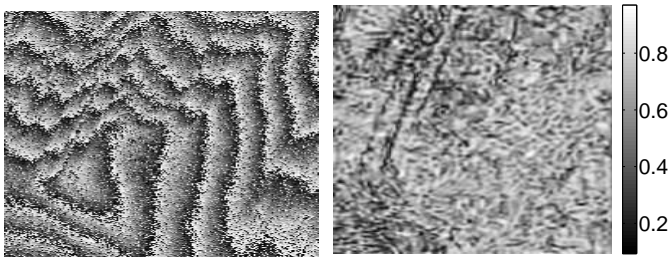


Fig. 12. Real Interferometric phase near Evaggelistria, Greece. Left: noisy phase. Right: estimated coherence.

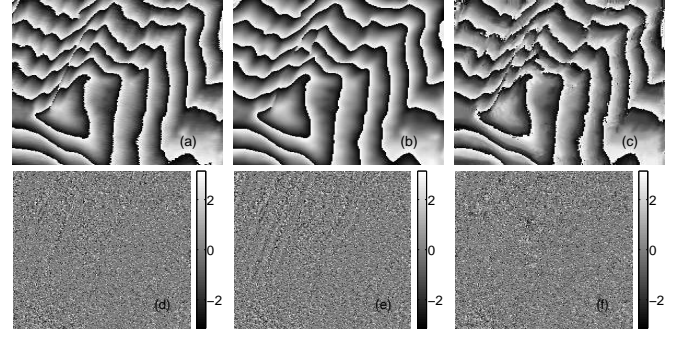


Fig. 13. Denoised results and errors between the estimate and the noisy phase of real Interferometric phase near city Evaggelistria, Greece: (a) SpInPHASE, (b) WFT (c) NL-InSAR, and (d),(e),(f) are the corresponding errors to the noisy phase

amount of noise. SpInPHASE yields the best balance between discontinuity preservation and noise reduction. To further assess the quality 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 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. 13, we conclude that SpInPHASE yields the less structured error and then the smallest bias.

In order to further illustrate SpInPHASE performance in real InSAR data, we have applied it to an InSAR much larger data set corresponding to an area centered at Etan volcano, Sicily, Italy (from  $37^{\circ}33'32''N$  to  $37^{\circ}40'41''N$  and from  $14^{\circ}43'55''E$  to  $15^{\circ}14'31''E$ ). The master and slave images were acquired, respectively, by ESR-1 on 2/8/1995 (orbit 21159, track 129) and by ERS-2 on 2/8/1995 (orbit 1486, track 129). 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.

Given that that paper is already too big, we present the results in a complementary report<sup>9</sup>. Although this data set is even more challenging than the previous one, the conclusions we may take from results displayed in the complementary report are as before; SpInPHASE yields the best balance between discontinuity/high phase rate preservation and noise reduction.

## VI. CONCLUDING REMARKS

This paper introduced SpInPHASE, an effective algorithm for interferometric phase image estimation, that is, 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 representations, also termed sparse coding, are computed by the orthogonal matching pursuit (OML) algorithm in the complex domain. The synthesis dictionaries are learned via matrix factorization with

<sup>9</sup>[http://www.lx.it.pt/~bioucas/files/ieee\\_tgrs\\_SpInPHASE\\_extended\\_2014.pdf](http://www.lx.it.pt/~bioucas/files/ieee_tgrs_SpInPHASE_extended_2014.pdf)



sparsity constraints promoted by the  $\ell_1$  norm regularizer. With the objective of reducing the dictionary learning time, a recently introduced online dictionary learning (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 *split augmented Lagrangian shrinkage algorithm* (SALSA). The obtained online dictionary learning algorithm allows to learn dictionaries of the order of  $150 \times 250$  in the complex domain from images of  $512 \times 512$  in times of the order of 4 minutes, 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 most cases with a significant advantage. The first results obtained with real MRI and InSAR data are equally encouraging. We highlight 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.

As future work, we will develop research efforts to decrease the estimation time. We are convinced that a suitable coding of the ODL and of the OMP steps, for example in C++, will reduce the time the algorithm takes by an order of magnitude. Another area where we will develop efforts is dictionary learning under more complex observations 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," *Proceedings of the IEEE*, vol. 62, no. 6, pp. 763–768, 1974.
- [2] H. Zebker and R. Goldstein, "Topographic mapping from interferometric synthetic aperture radar observations," *Journal of Geophysical Research*, vol. 91, pp. 4993–4999, 1986.
- [3] P. Lauterbur, "Image formation by induced local interactions: Examples employing nuclear magnetic resonance," *Nature*, vol. 242, no. 5394, pp. 190–191, 1973.
- [4] M. Hedley and D. Rosenfeld, "A new two-dimensional phase unwrapping algorithm for MRI images," *Magnetic Resonance in Medicine*, vol. 24, no. 1, pp. 177–181, 1992.
- [5] D. Ghiglia and M. Pritt, *Two-Dimensional Phase Unwrapping: Theory, Algorithms, and Software*. Wiley, 1998.
- [6] T. Kreis, *Handbook of Holographic Interferometry: Optical and Digital Methods*. Wiley-VCH, 2005.
- [7] A. Patil and P. Rastogi, "Moving ahead with phase," *Optics and Lasers in Engineering*, vol. 45, no. 2, pp. 253–257, 2007.
- [8] N. Karpinsky and S. Zhang, "High-resolution, real-time 3D imaging with fringe analysis," *Journal of Real-Time Image Processing*, vol. 7, no. 1, pp. 55–66, 2012.
- [9] C. David, B. Nohammer, H. Solak, and E. Ziegler, "Differential X-ray phase contrast imaging using a shearing interferometer," *Applied Physics Letters*, vol. 81, no. 17, pp. 3287–3289, 2002.
- [10] V. Katkovnik, K. Egiazarian, and J. Astola, *Local Approximation Techniques in Signal and Image Processing*, ser. SPIE field guides. Bellingham WA: SPIE Press, 2006.
- [11] Q. Kemao, "Two-dimensional windowed Fourier transform for fringe pattern analysis: principles, applications and implementations," *Optics and Lasers in Engineering*, vol. 45, no. 2, pp. 304–317, 2007.
- [12] Q. Kemao, L. Nam, L. Feng, and S. Soon, "Comparative analysis on some filters for wrapped phase maps," *Applied Optics*, vol. 46, no. 30, pp. 7412–7418, 2007.
- [13] J. Bioucas-Dias, V. Katkovnik, J. Astola, and K. Egiazarian, "Absolute phase estimation: adaptive local denoising and global unwrapping," *Applied Optics*, vol. 47, no. 29, pp. 5358–5369, 2008.
- [14] J. Bioucas-Dias and G. Valadão, "Phase unwrapping via graph cuts," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 698–709, 2007.
- [15] C.-A. Deledalle, L. Denis, and F. Tupin, "NL-InSAR: Non-local interferogram estimation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, pp. 1441–1452, 2011.
- [16] A. Efros and T. Leung, "Texture synthesis by non-parametric sampling," in *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, 1999, pp. 1033–1038.
- [17] A. Buades, B. Coll, and J. Morel, "A non-local algorithm for image denoising," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 2, 2005, pp. 60–65.
- [18] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.
- [19] J. Bioucas-Dias and J. Leitão, "The  $Z\pi M$  algorithm: a method for interferometric image reconstruction in SAR/SAS," *IEEE Transactions on Image Processing*, vol. 11, no. 4, pp. 408–422, 2002.
- [20] V. Katkovnik, J. Astola, and K. Egiazarian, "Phase local approximation (PhaseLa) technique for phase unwrap from noisy data," *IEEE Transactions on Image Processing*, vol. 17, no. 6, pp. 833–846, 2008.
- [21] G. Nico and J. Fortuny-Guasch, "Using the matrix pencil method to solve phase unwrapping," *IEEE Transactions on Signal Processing*, vol. 51, no. 3, pp. 886–888, 2003.
- [22] M. Rivera and J. Marroquin, "Half-quadratic cost functions for phase unwrapping," *Optics Letters*, vol. 29, no. 5, pp. 504–506, 2004.
- [23] G. Valadão and J. Bioucas-Dias, "CAPE: combinatorial absolute phase estimation," *Journal of the Optical Society of America A*, vol. 26, no. 9, pp. 2093–2106, 2009.
- [24] J. Leitão and M. Figueiredo, "Absolute phase image reconstruction: a stochastic nonlinear filtering approach," *IEEE Transactions on Image Processing*, vol. 7, no. 6, pp. 868–882, 1998.
- [25] O. Löffel, H. Nies, S. Knedlik, and Y. Wang, "Phase unwrapping for SAR interferometry - a data fusion approach by kalman filtering," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 46, no. 1, pp. 47–58, 2008.
- [26] B. Osmanoglu, T. Dixon, and S. Wdowski, "Three-dimensional phase unwrapping for satellite radar interferometry, I: DEM generation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. PP, no. 99, 2013.
- [27] J. Martinez-Espla, T. Martinez-Marin, and J. Lopez-Sanchez, "Using a grid-based filter to solve InSAR phase unwrapping," *IEEE Transactions on Geoscience and Remote Sensing Letters*, vol. 5, no. 2, pp. 147–151, 2008.
- [28] —, "A particle filter approach for InSAR phase filtering and unwrapping," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 4, pp. 1197–1211, 2009.
- [29] —, "An optimized algorithm for InSAR phase unwrapping based on particle filtering, matrix pencil, and region-growing techniques," *IEEE Transactions on Geoscience and Remote Sensing Letters*, vol. 6, no. 4, pp. 835–839, 2009.
- [30] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.
- [31] M. Elad, *Sparse and Redundant Representations*. Springer, 2010.
- [32] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.
- [33] R. Raina, A. Battle, H. Lee, B. Packer, and A. Ng, "Self-taught learning: transfer learning from unlabeled data," in *Proceedings of the 24th international conference on Machine learning*. New York, NY, USA: ACM, 2007, pp. 759–766.
- [34] J. Mairal, F. Bach, and J. Ponce, "Task-driven dictionary learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 791–804, 2012.
- [35] S. Chen, D. Donoho, M. Saunders, and A. Michael, "Atomic decomposition by basis pursuit," *SIAM Review*, 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 *Proceedings of the 26th Annual International Conference on Machine Learning*, ser. ICML '09. New York, USA: ACM, 2009, pp. 689–696.
- [37] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2345–2356, 2010.
- [38] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.



- [39] R. Bamler and P. Hartl, "Synthetic aperture radar interferometry," *Inverse Problems*, vol. 14, no. 4, pp. R1–R54, 1998.
- [40] S. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall, 1993, vol. I.
- [41] M. Elad, "Sparse and redundant representation modeling – what next?" *IEEE Signal Processing Letters*, vol. 19, no. 12, pp. 922–928, 2012.
- [42] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local sparse models for image restoration," in *IEEE 12th International Conference on Computer Vision*, 2009, pp. 2272–2279.
- [43] K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM Journal on Computing*, vol. 24, no. 2, pp. 227–234, 1995.
- [44] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society (Series B)*, vol. 58, pp. 267–288, 1996.
- [45] Y. Pati, R. Rezaifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, 1993, pp. 40–44.
- [46] T. Blumensath and M. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, 2009.
- [47] S. Foucart, "Hard thresholding pursuit: an algorithm for compressive sensing," *SIAM Journal on Numerical Analysis*, vol. 49, no. 6, pp. 2543–2563, 2011.
- [48] J. Vila and P. Schniter, "Expectation-maximization Bernoulli-Gaussian approximate message passing," in *Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, 2011, pp. 799–803.
- [49] D. Donoho and M. Elad, "Optimal sparse representation in general (non-orthogonal) dictionaries via  $\ell_1$  minimization," *Proceedings of the National Academy of Sciences*, vol. 100, pp. 2197–2202, 2003.
- [50] E. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, 2006.
- [51] A. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1," *Vision Research*, vol. 37, no. 23, pp. 3311–3326, 1997.
- [52] M. Lewicki and T. Sejnowski, "Learning overcomplete representations," *Neural Computation*, vol. 12, no. 2, pp. 337–365, 2000.
- [53] M. Aharon, M. Elad, and A. Bruckstein, "The K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.
- [54] H. Lee, A. Battle, R. Raina, and A. Ng, "Efficient sparse coding algorithms," *Advances in Neural Information Processing Systems*, vol. 19, pp. 801–808, 2007.
- [55] A. Charles, B. Olshausen, and C. Rozell, "Learning sparse codes for hyperspectral imagery," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 963–978, 2011.
- [56] A. Rakotomamonjy, "Applying alternating direction method of multipliers for constrained dictionary learning," *Neurocomputing*, vol. 106, no. 15, pp. 126–136, 2012.
- [57] J. Bioucas-Dias and M. Figueiredo, "Alternating direction algorithms for constrained sparse regression: Application to hyperspectral unmixing," in *2nd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*. IEEE, 2010, pp. 1–4.
- [58] S. Boyd, N. Parikh, E. Chu, B. Peleato, and E. Jonathan, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [59] D. Just and R. Bamler, "Phase statistics of interferograms with applications to synthetic aperture radar," *Applied Optics*, vol. 33, no. 20, pp. 4361–4368, 1994.