CAPE: combinatorial absolute phase estimation

Gonçalo Valadão* and José Bioucas-Dias

*Instituto de Telecomunicações and Instituto Superior Técnico, Avenida Rovisco Pais, Torre Norte,
Piso 10, 1049-001 Lisboa, Portugal
Corresponding author: gvaladao@lx.it.pt

An absolute phase estimation algorithm for interferometric applications is introduced. The approach is Bayesian. Besides coping with the 2π-periodic sinusoidal nonlinearity in the observations, the proposed methodology assumes a first-order Markov random field prior and a maximum *a posteriori* probability (MAP) viewpoint. For computing the MAP solution, we provide a combinatorial suboptimal algorithm that involves a multiprecision sequence. In the coarser precision, it unwraps the phase by using, essentially, the previously introduced PUMA algorithm [IEEE Trans. Image Proc. **16**, 698 (2007)], which blindly detects discontinuities and yields a piecewise smooth unwrapped phase. In the subsequent increasing precision iterations, the proposed algorithm denoises each piecewise smooth region, thanks to the previously detected location of the discontinuities. For each precision, we map the problem into a sequence of binary optimizations, which we tackle by computing min-cuts on appropriate graphs. This unified rationale for both phase unwrapping and denoising inherits the fast performance of the graph min-cuts algorithms. In a set of experimental results, we illustrate the effectiveness of the proposed approach. © 2009 Optical Society of America

OCIS codes: 100.5088, 100.3020, 100.3175.

1. INTRODUCTION

There are nowadays many applications based on phase images, e.g., interferometric synthetic aperture radar (InSAR) [1], magnetic resonance imaging [2], adaptive optics [3,4], vibration and deformation measurements [5], and diffraction tomography [6]. InSAR is being successfully applied, e.g., to the generation of digital elevation models and in the monitoring of land subsidence; among the plethora of magnetic resonance imaging applications, we emphasize venography (and angiography as well) [7] and tissue elastography [8]; concerning adaptive optics, we point to applications in medicine and industry [9]. Interferometry-based vibration and deformation measurements are widespread among metrology techniques, and diffraction tomography finds application in, e.g., geophysical subsurface prospection and 3D microscopic imaging.

In all of these imaging systems, the acquisition sensors read only the sine and the cosine of the absolute phase; that is, we have access only to the phase modulo 2π, the so-called interferogram. Besides this sinusoidal nonlinearity, the observed data are corrupted by some type of noise. Because of these degradation mechanisms, absolute phase estimation is known to be a very difficult problem. In particular, if the magnitude of absolute phase variation between neighboring pixels is larger than π, i.e., if the so-called Itoh condition [10] is violated, then the inference of the absolute phase is an ill-posed problem [11]. These violations may be due to spatial undersampling, original phase discontinuities, or noise. The goal in this paper is to present an algorithm to solve this nonlinear inverse problem: the so-called absolute phase estimation problem.

The structure of the observation models relating the noisy wrapped phase with the true phase depends on the

system under consideration (see, e.g., [12–14] for an account of observation models in different coherent imaging systems). The essence of most of these observation mechanisms is, however, captured by the relation

$$z = Ae^{j\phi} + n, \quad A > 0,$$
$$= |z|e^{j(\phi + \phi_n)}, \tag{1}$$

where ϕ is the true phase value (the so-called absolute phase value), $n = n_I + jn_Q$ is complex-valued zero-mean circular white additive noise with variance σ^2 (i.e., n_I and n_Q are zero-mean independent Gaussian random variables with variance $\sigma^2/2$), and ϕ_n is the phase due to n.

Let us define $\mathcal{W}$, the *wrapping operator*, such that

$$\psi = \text{angle}(z) = \mathcal{W}(\phi + \phi_n), \qquad \psi \in [-\pi, \pi). \tag{2}$$

Operator $\mathcal{W}$ maps the noisy phase $\phi + \phi_n$ into the principal phase interval $[-\pi, \pi)$. For $\phi_n = 0$, there is an obvious link between the wrapped phase ψ and the nonwrapped absolute phase ϕ, $\phi = \psi + 2\pi k$, $\psi \in [-\pi, \pi)$, where k is an integer. The basic unwrapping problem is to reconstruct $\phi(x,y)$, $(x,y) \in X \subset \mathbb{Z}^2$, from the observations $\psi(x,y)$. There is, of course, no one-to-one relation between the wrapped and the absolute phases. It is, however, straightforward to show that, in the absence of noise, if the phase difference $\Delta\phi$ between any two neighboring pixels is smaller than π in magnitude (i.e., the Itoh condition is verified), then the correspondent wrapped phase difference is the same, i.e.,

$$\Delta\phi = \mathcal{W}(\Delta\psi). \tag{3}$$

We are, then, able to compute (unwrap) the absolute phase for every other pixel, up to a constant, by integrating the wrapped differences $\mathcal{W}(\Delta\psi)$. This is the basis of

the path-following phase unwrapping algorithms [15]. It happens, quite often, that phase images have discontinuities, areas of high phase rate, and noise. In this case, the Itoh condition may fail, and therefore different integration paths may lead to different unwrapped phase values.

To cope with these difficulties, there are in the literature two main classes of algorithms: the basic phase unwrapping type, which solves only the 2π-multiples problem [15–17,19], and the estimation type (or regularization type), in which the absolute phase is estimated [13,20–23], and thus the estimation type has a broader scope.

A. Contributions
The main contribution of this paper is twofold:

1. A new approach to absolute phase estimation. Contrary to the majority of absolute phase methods, we first apply unwrapping and only then denoising. By using a discontinuity-preserving prior, the unwrapping algorithm not only infers the 2π multiples of the absolute phase, but also implicitly detects the discontinuity locations. This is crucial information for the effectiveness of the phase denoising that would not be available if this step were applied prior to unwrapping.

2. A unified rationale for both phase unwrapping and denoising. This is achieved through the proposal of a (state-of-the-art competitive) multiprecision combinatorial optimization algorithm based on graph cuts. The multiprecision technique improves the performance by decreasing the algorithm's complexity.

B. Related Work
In the field of phase unwrapping, we refer to [12] and references therein. References [17–19,24] introduce phase unwrapping algorithms with discontinuity-preserving capabilities. Furthermore, PUMA [25] is a state-of-the-art phase unwrapping algorithm, while $\mathbb{Z}\pi M$ [13] and PEARLS [26] are state-of-the-art absolute phase estimation algorithms. Regarding $\mathbb{Z}\pi M$, the present work extends it by allowing a wider family of Markov random fields (MRFs), namely, nonconvex pairwise potentials, which brings the discontinuity-preservation ability. The PEARLS algorithm [26] employs PUMA for phase unwrapping and differs from ours essentially in the denoising technique: specifically, it applies denoising before unwrapping. This is a sensitive issue, since the effectiveness of the denoising step depends on the knowledge of the location of the discontinuities. PEARLS implements a filtering technique based on local polynomial approximation with a varying adaptive neighborhood used in reconstruction (see [26] for details). The adaptive mechanism trades bias with variance in such a way that the window size stretches in areas where the underlying true phase is smooth and shrinks otherwise, that is, in the presence of discontinuities.

In this paper, we circumvent the need of filtering the interferogram (noisy wrapped phase image) by basically first using PUMA to unwrap the interferogram; only then is denoising applied after PUMA has implicitly located the discontinuities. Compared with PUMA [25], the present work also goes further by both allowing denoising and improving the performance of each binary step in the optimization sequence.

C. Integer Optimization
With respect to the optimization referred to above, we deal with a labeling problem on first-order Markovian random field made of unary (one-variable dependence) and pairwise on difference (two-variable difference dependence) terms. If the unary and pairwise terms are convex, this problem is known as the dual of the convex cost network flow problem which is extensively discussed in the literature (see, e.g., [27–30]). References [13,25,31] attacked particular instances of the dual of the convex cost network flow problem by solving a sequence of descent binary optimization problems, each of which may be solved by computing the max-flow/min-cuts on appropriate graphs. This is a relevant practical advantage, as there are max-flow/min-cut algorithms specifically tuned to image processing and computer vision problems [32].

References [29,33–35] introduced and characterized generalizations of the descent algorithms [13,25,31] to dual of the convex cost network flow problems containing pairwise and unary terms. An even more general class of functions, the so-called L-convex and $L^\natural$-convex functions, was considered in [36]. References [33,34], in addition to introducing new results, give an excellent account of the connections between the dual of the convex cost network flow problems and of L-convex and $L^\natural$-convex based problems. Concerning Markovian first-order objective functions (having unary and pairwise terms), we still should refer to [37] (pairwise term depending on the difference of the two values involved), which, although being theoretically very appealing, seems not very practical for most of the computer vision problems; we should also refer to [38], which deals with general first-order submodular objective functions. In practical terms, we can think about a submodular function as a discrete analogue of a convex function. A binary pairwise interaction function $E(\cdot,\cdot)$ is submodular iff $E(0,0)+E(1,1)\leq E(1,0)+E(0,1)$. Concerning this subject we refer to [36] and references therein.

Discontinuity-preserving pairwise interaction terms are, quite often, nonconvex. In this case, the underlying optimization problem is NP-hard [39], and none of the cited algorithms applies. The source of difficulties is twofold: (a) a sequence of descent optimizations leads to a local minimum, and (b) each subproblem can no longer be mapped onto a max-flow/min-cut. In the vein of [25] we insist, however, in the descent approach, which, in the case of phase unwrapping, leads systematically to high-quality results. Concerning each binary subproblem, we replace the objective function with a convex majorizer, thus adopting the majorization–minimization framework [40].

D. Paper Organization
In the next section, we present the core ideas and concepts of the proposed approach. Subsection 2.A characterizes the posterior density of the proposed Bayesian framework; then Subsection 2.B concerns the roles of unwrapping and denoising in the optimization process to compute the maximum *a posteriori* (MAP); and subse-

quently Subsection 2.C establishes and discusses the optimization technique in some detail. Finally, Section 3 presents a series of experimental results, and then Section 4 provides some concluding remarks.

2. PROPOSED APPROACH

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph associated with a first-order MRF where the set of nodes $\mathcal{V}$ represents image pixels and the set of edges $\mathcal{E}$ represents pairs of neighboring pixels. We assume that if $(i,j) \in \mathcal{E}$, then $(j,i) \notin \mathcal{E}$.

In this paper, we consider first-order MRFs, and therefore the set of edges $\mathcal{E}$ represents the set of horizontal and vertical neighbors. Nevertheless, all the concepts and results presented are valid for any set of pairwise interactions.

A. Posterior Function

We follow the Bayesian framework. Accordingly, we need to build the posterior density function $p(\phi|z)$ of the phase image $\phi \in \mathbb{R}^{|\mathcal{V}|}$ given the observed complex image $z \in \mathbb{C}^{|\mathcal{V}|}$ ($\mathbb{C}$ denotes the complex field). Invoking the Bayes law we have $p(\phi|z) \propto p(z|\phi) p(\phi)$, where $p(z|\phi)$ is the likelihood function measuring the data fit and $p(\phi)$ is the prior density encoding *a priori* knowledge about the phase image ϕ.

Let us assume conditional independence in the observation mechanism, i.e., $p(z|\phi) = \Pi_{i \in \mathcal{V}} p(z_i|\phi_i)$. Furthermore, let us consider priors such that $\log p(\phi) = -\mu \Sigma_{(i,j)} V_{i,j}(\phi_i - \phi_j) + c$, where c is an irrelevant constant, $\mu > 0$ is a scale parameter often termed the regularization parameter, and $V_{i,j}(\cdot)$ is the so-called potential associated with edge (i,j). In these circumstances, computing the MAP estimate is equivalent to minimize the negative logarithm of the posterior density $E : \mathbb{R}^{|\mathcal{V}|} \to \mathbb{R} \cup \{+\infty\}$ given by

$$E(\phi) \equiv \underbrace{\sum_{i \in \mathcal{V}} D_i(\phi_i)}_{\text{Data fidelity term}} + \mu \underbrace{\sum_{(i,j) \in \mathcal{E}} V_{i,j}(\phi_i - \phi_j)}_{\text{Prior term}}, \qquad (4)$$

where $D_i(\phi_i) \equiv -\log p(z_i|\phi_i)$.

MAP Estimation. Given the observation mechanism introduced in Eq. (2), we have (see, e.g., [13])

$$p(z_i|\phi_i) = \frac{1}{\pi\sigma^2} e^{(-|z_i - Ae^{j\phi_i}|^2)/\sigma^2},$$

and, thus, by dropping some irrelevant constants we get

$$D_i(z_i) = -\lambda_i \cos(\phi_i - \psi_i) \qquad \text{for } i \in \mathcal{V},$$

with $\lambda_i \equiv 2A|z_i|/(\sigma^2)$ and $\psi_i \equiv \text{angle}(z_i)$; i.e., the log likelihood function is proportional to a shifted cosine. The MAP absolute phase estimate is then obtained by minimizing the negative of the log posterior function given by

$$E(\phi) \equiv \underbrace{\sum_{i \in \mathcal{V}} -\lambda_i \cos(\phi_i - \psi_i)}_{\text{Data fidelity term}} + \mu \underbrace{\sum_{(i,j) \in \mathcal{E}} V_{i,j}(\phi_i - \phi_j)}_{\text{Prior term}}. \qquad (5)$$

Notice that μ, the so-called regularization parameter, sets the relative weight between the data fidelity term and the prior term.

We emphasize that the MAP estimation gives place to minimizing the log posterior function (5). For a given image phase ϕ candidate to a MAP solution, this function comprises a data fidelity term, which measures the misfit between the observed data and ϕ, and a prior term, which measures the lack of plausibility of ϕ, induced by the potentials $V_{i,j}(\cdot)$; roughly, enforcing smooth surfaces implies convex potentials, whereas enforcing piecewise smoothness, and thus preserving surfaces discontinuities, implies nonconvex potentials.

Assume that the noise approaches zero. Then, $\lambda_i \equiv 2A|z_i|/(\sigma^2) \to +\infty$, and any MAP solution satisfies $\cos(\phi_i - \psi_i) = 1$, implying the constraints

$$\phi_i = \psi_i + 2\pi k_i \quad \text{for } i \in \mathcal{V} \text{ and } k_i \in \mathbb{Z}. \qquad (6)$$

Therefore, computing the MAP solution reduces to minimizing the prior term in Eq. (5) under constraints (6). This the so-called phase unwrapping problem, which is an integer optimization problem well known to be a difficult task to perform [41]. The phase unwrapping is even more tricky because, usually, the phase images are very large (e.g., 10^6 variables for a 1000×1000 image). Such a problem is addressed by the PUMA algorithm [25], which yields, in polynomial time, exact or approximate solutions as long as the potential V in Eq. (5) is, respectively, convex or nonconvex. In fact, in the latter case, we cannot hope to achieve more than approximate solutions, as the problem is NP-hard [32,42].

Aiming at simultaneous noise filtering and discontinuity preservation, we use in this paper half-quadratic-type [43,44] potentials, all with the same shape $V_{i,j} = V$ given by

$$V(x) = \begin{cases} |x|^2 & \text{if } |x| \le \pi \\ \pi^2 - \pi^p + |x|^p & \text{if } |x| > \pi \end{cases}, \qquad (7)$$

with $0 < p < 1$ and thus nonconvex. Figure 1 plots this function for $p = 0.4$. These potentials are quadratic near the origin, thus modeling Gaussian noise, and with a flatness trend for phase magnitudes larger than π, in order to preserve discontinuities [45]. When we do not have discontinuities to deal with, we simply employ quadratic (or near quadratic) potentials. In any case, the concepts and ideas introduced next are valid for a wide class of potential functions.
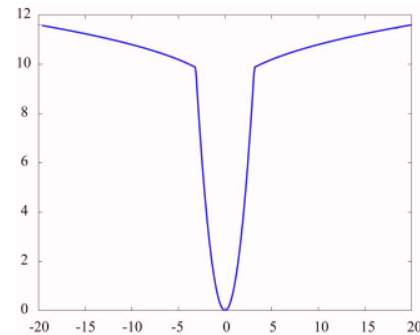

Fig. 1. (Color online) Graph of a half-quadratic potential employed in this paper ($p = 0.4$).

B. Optimization

Our goal is to compute

$$\phi^* = \arg \min_{\phi}[E(\phi)], \qquad (8)$$

where $E(\phi)$ is given by Eq. (5). This is a hard problem because both data fidelity and prior terms are nonconvex.

First Unwrap, then Denoise. Very often, in interferometric applications, we are just interested in computing the 2π multiple in the representation $\phi_i = \psi_i + 2\pi k_i$, the so-called phase unwrapping problem. Since the data fidelity term, $D_i(\phi_i = \psi_i + 2\pi k_i) = -\lambda_i$, does not depend on k_i, then the unwrapping optimization problem consists in minimizing the prior term of Eq. (5) with respect to $\phi \in \Pi_{i \in \mathcal{V}}\{\psi_i + 2\pi k_i : k_i \in \mathbb{Z}\}$. In our previous works [25,31], we introduced a descent method that, depending on the potentials, yields exact (in the case of convex potentials) or approximated (in the case of nonconvex potentials) solutions. Each step of this method solves a binary problem by computing the min-cut of an appropriate graph.

We note that the basic phase unwrapping step can be viewed as a discretization of the original domain, using a sampling interval of 2π. A consequence of this discretization is that the resulting objective function is easier to deal with, since, for $\phi \in \Pi_{i \in \mathcal{V}}\{\psi_i + 2\pi k_i : k_i \in \mathbb{Z}\}$, it does not depend on the nonconvex data terms $-\lambda_i \cos(\phi_i - \psi_i)$.

After the unwrapping step, we get absolute phase estimates given by $\hat{\phi}_i = \psi_i + 2\pi \hat{k}_i$. Even if the integer image estimate is exact, we still have error in $\hat{\phi}_i$ due to the noise present in ψ_i [please see expression (2)]. In order to filter out the noise, we compute a sequence of binary descent optimizations, using a multiprecision schedule. The precision $q \in \{0, 1, \ldots, N\}$ corresponds to a sampling interval of $\Delta = 2\pi/2^q$. Thus, the coarser precision implements phase unwrapping and the following denoising.

We highlight the following qualitative characteristics of the approach just described:

1. The PUMA algorithm [25], used for phase unwrapping, is able to deal with discontinuities and implicitly locate them.

2. The PUMA solution yields an error much smaller than π in magnitude in most of the pixels.

3. Given that for precisions $q > 0$ we have, for most $i \in \mathcal{V}, |\phi_i - \psi_i| \ll \pi$, then most of the unary terms $-\lambda_i \cos(\phi_i - \psi_i)$ behave as convex functions, rendering a much easier optimization problem.

One expectable advantage of coarse-to-fine multiprecision schedule is computation time. However, in our scenario this is not, perhaps, the most important feature: owing to the nonconvexity of the problem, the algorithm very likely would have become stuck in a local minimum if we had begun with the highest precision.

C. Algorithm

Algorithm 1 shows the pseudocode for the optimization, where we use the following sets:

$$M^U(\phi', \Delta) \equiv \{\phi \in \mathbb{R}^{|\mathcal{V}|} : \phi_i = \phi'_i + \delta_i \Delta\},$$

$$M^D(\phi', \Delta) \equiv \{\phi \in \mathbb{R}^{|\mathcal{V}|} : \phi_i = \phi'_i - \delta_i \Delta\},$$

where $\delta_i \in \{0, 1\}$, $\Delta \in \mathbb{R}$.

Our Algorithm 1:

Algorithm 1. Combinatorial absolute phase estimation.
Initialization: $\phi = \psi$ {Interferogram}, successup=false, successdown=false

1: **for** $\Delta = 2\pi \times \{2^0, 2^{-1}, \ldots, 2^{-N}\}$ **do**
2:   **while** (successup=false OR successdown=false) **do**
3:     **if** successup=false **then**
4:       $\hat{\phi} = \arg\min_{\hat{\phi} \in M^U(\phi, \Delta)} \tilde{E}(\hat{\phi})$
5:       **if** $E(\hat{\phi}) < E(\phi)$ **then**
6:         $\phi = \hat{\phi}$
7:       **else**
8:         successup=true
9:       **end if**
10:     **end if**
11:     **if** successdown=false **then**
12:       $\hat{\phi} = \arg\min_{\hat{\phi} \in M^D(\phi, \Delta)} \tilde{E}(\hat{\phi})$
13:       **if** $E(\hat{\phi}) < E(\phi)$ **then**
14:         $\phi = \hat{\phi}$
15:       **else**
16:         successdown=true
17:       **end if**
18:     **end if**
19:   **end while**
20: **end for**

---

engages in a greedy succession of up and down binary optimizations. The precision interval Δ starts with the value 2π and ends with the value $2\pi/(2^N)$, where N is a depth of precision. In order to characterize the algorithm, we start by assuming $E(\phi) = \tilde{E}(\phi)$ and the terms D_i and $V_{i,j}$ in expression (4) to be convex. Then, for each precision interval Δ, Algorithm 1 finds, in finite time, a minimizer of E in a grid of size Δ. More precisely, we have the following.

Theorem 1. If the unary and pairwise terms of E, defined in Eq. (4), respectively, $D_i(\cdot)$ for $i \in \mathcal{V}$, and $V_{ij}(\cdot)$ for $(i,j) \in \mathcal{E}$, are convex, then, at a given precision interval $\Delta = 2\pi/2^q$, for $q \in \{0, \ldots, N\}$, the following holds:

1. The output of the **while** loop is a minimizer of $E(\phi)$ in the set $\{\phi \in \mathbb{R}^{|\mathcal{V}|} : \phi_i = \psi_i + z_i \Delta, z_i \in \mathbb{Z}\}$.

2. The number of **while** iterations at each precision $q \in \{0, \ldots, N\}$ is bounded by $K_q + 1$, where $K_q = K_0 2^q$ is the range of the 2π multiples variables at precision q.

Proof: For the precision interval Δ, the pseudocode embraced by the **while** loop (lines between 2 and 19) finds a minimizer of $E(\phi)$ in $\{\phi \in \mathbb{R}^{|\mathcal{V}|} : \phi_i = \phi' + z_i \Delta, z_i \in \mathbb{Z}\}$, where ϕ' is the minimizer obtained in the previous precision. Proofs are given in [25], in the absence of unary terms, and in [33,35], for the general case, i.e., when E contains unary and pairwise terms.

Since the successive precisions are powers of $(1/2)$ we can write $\phi'_i = \psi_i + l_i \Delta$, where ϕ'_i and ψ_i are the ith components of ϕ' and ψ, respectively, and l_i is an integer. Therefore, it follows that $\{\phi \in \mathbb{R}^{|\mathcal{V}|} : \phi_i = \phi' + z_i \Delta, z_i \in \mathbb{Z}\} = \{\phi \in \mathbb{R}^{|\mathcal{V}|} : \phi_i = \psi_i + z_i \Delta, z_i \in \mathbb{Z}\}$. $\qquad\square$

As already pointed out, we solve the binary minimizations displayed in lines 5 and 14 of Algorithm 1 by com-

puting max-flows/min-cuts on appropriate graphs. Denoting by $T(n,m)$ the time complexity of the max-flow/min-cut algorithm used, where n is the number of graph nodes and m the number of edges, then Algorithm 1 takes the pseudopolynomial time $O(K_q T(n,m))$ to find a minimum of E at the precision q.

The rationale underlying the multiprecision minimization is that of a minimum length search for a minimizer of E. Still considering the convex scenario, given a minimizer at a precision q, say ϕ^q, there exists a minimizer at the precision $q+1$ such that $\|\phi^q - \phi^{q+1}\|_\infty < n$ [36, Theorem 7.18], where $\|x\|_\infty \equiv \max_i x_i$ is the l^∞ norm of x. Therefore, the algorithm takes at most n iterations to find a minimizer at resolution $q+1$ [25]. Consequently, the number of iterations to find a minimizer of E is bounded by $O(n \log K_N)$. In practice, we have observed systematically $\|\phi^q - \phi^{q+1}\|_\infty \ll n$, and very often we have $\|\phi^q - \phi^{q+1}\|_\infty < 2$, making the algorithm highly efficient from the time complexity point of view. This behavior is illustrated in Fig. 4(c) below, where we show the evolution of Algorithm 1 in a convex scenario, both with and without multiprecision. Notice that in the former case the number of up and down iterations (lines 6 and 14 of Algorithm 1) to find a minimum, in a given precision, is $2 \times (K_q + 1)$.

To solve the binary optimizations shown in lines 4 and 12, we use the graph cuts techniques. The main idea, which was introduced into computer vision in [46], and popularized in [37,39,41,47], is a one-to-one mapping of a submodular binary function onto the set of cuts of a certain source-terminal graph. As referred to at the very beginning of this section, to each pixel of the image to be inferred there corresponds a node. Each node is in general linked, through edges, with each neighbor (either vertical or horizontal in our case) and, furthermore, to the source (s) and/or to the sink (t) (the edges always have either positive or null weights, the latter corresponding to nonexistence of the edge). An $s-t$ cut $C=(S,T)$ is a partition of the set of vertices $\mathcal{V}$ into two disjoint sets S and T, such that $s \in S$ and $t \in T$. The cost of the cut is the sum of costs of all edges between S and T. A min-cut is a cut of minimum cost. After computing an $s-t$ min-cut, each node v ends having the binary value 1 if $v \in S$ or binary value 0 otherwise.

Figure 2 illustrates the building of such a graph. As long as the binary function is submodular, the exact global minimum of the function is attained [39]. According to the Ford–Fulkerson theorem [48], the min-cut is equal to the max-flow. As there are plenty of efficient low-order polynomial complexity algorithms to compute min-cuts/max-flows, it turns out that each binary problem is efficiently solved. We have followed the mapping proposed in [49], which gives an improvement relative to [39].

In our case, neither the unary nor the pairwise energy [Eq. (4)] interaction terms, $D_i(\phi_i)$ and $V_{i,j}(\phi_i - \phi_j)$, respectively, are convex. First, this means that we cannot map the binary optimizations discussed above into min-cuts on suited graphs, because the resulting binary energy is nonsubmodular [39]; moreover, we do not meet the conditions in which Theorem 1 applies. We address this problem by approximating energy $E(\phi)$ with $\tilde{E}(\phi)$. This approximation consists in replacing the nonsubmodular pairwise terms $V_{i,j}$ with submodular majorizers. Then we apply the
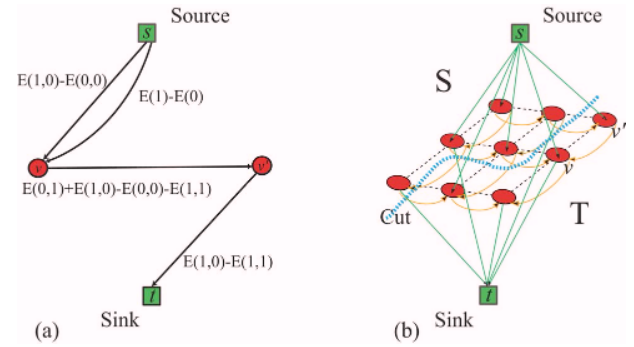

Fig. 2. (Color online) Elementary graph representing a unary and a pairwise interaction energy term. Nodes s and t represent source and sink, respectively, and v and v' represent the two pixels involved in the interaction; pixel v also has a unary term attached to it. In this case $E(1,0)-E(0,0)>0$, $E(1,0)-E(0,0)>0$, and $E(1,0)-E(1,1)>0$. We note that the two edges linking pixels s and t are equivalent to a single edge with weight $E(1)-E(0)+E(1,0)-E(0,0)$. (b) Illustration of the graph obtained at the end: it results from adding elementary graphs.

majorize–minimize principle [40]. Figure 3 illustrates the process of finding the majorizer.

We note that although we employ the approximation $\tilde{E}(\phi)$ to compute the optimizations, the evaluation of descending energy in lines 5 and 13 of Algorithm 1 is made through $E(\phi)$, the true energy.

We now emphasize two facts:

• For the first precision ($\Delta = 2\pi$) there is no unary term in the energy $E(\phi)$ [Eq. (4)]. This means that with $\tilde{E}$ we are in a convex scenario.

• For the subsequent precisions the nonconvex sinusoidal unary terms in $\tilde{E}$ are not null; however, for these small precisions, we are likely to be in a convex attraction basin of $\tilde{E}(\phi)$ because, as already referred to in Subsection 2.B, $|\phi_i - \psi_i| \ll \pi$ for most of the unary terms.

Figure 4 illustrates the virtues of employing a multiprecision approach. In the first example [Figs. 4(a)–4(c)] we intend to illustrate the pure effect of the algorithm's speed enhancing. For that, we consider a convex energy Eq. (4), for which we set $D_i(z_i) = |\phi_i - \psi_i|^2$, instead of the
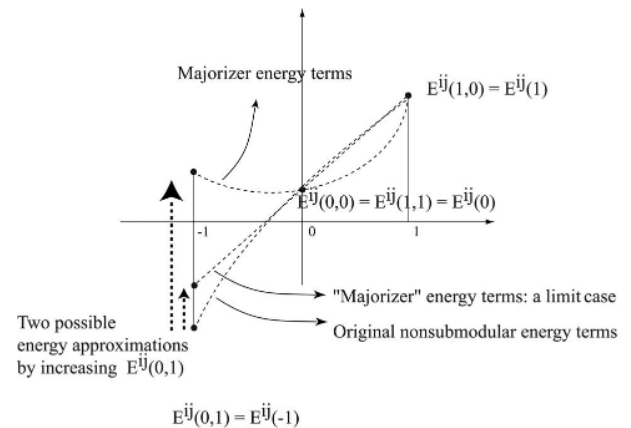

Fig. 3. Replacing nonsubmodular energy terms by submodular ones; we end up with an approximate energy. One of the possible approximations is to increase $E^{ij}(0,1)$.
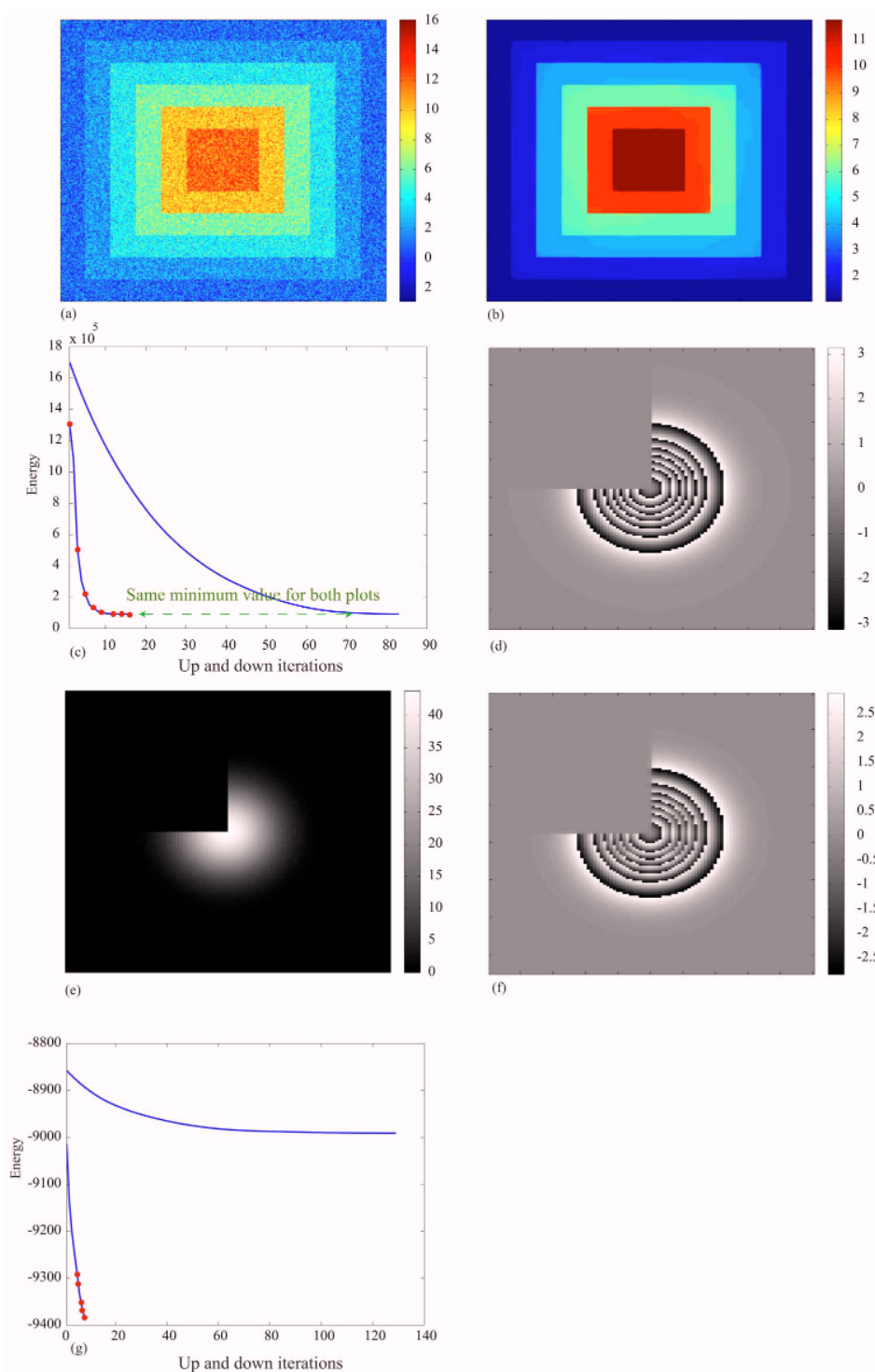
Fig. 4.   (Color online) (a) Discretized pyramid with additive Gaussian noise ($\sigma=1$). (b) Image in (a) denoised by Algorithm 1 (convex energy). (c) Energy decrease versus iterations for (b): dots indicate an increase of precision; the other curve corresponds to using the finest precision only. (d) Wrapping of a clipped Gaussian with 14π rad height. (e) Perfect unwrapping obtained by Algorithm 1 using the multiprecision approach. (f) Completely failed unwrapping, running the algorithm with finest precision only. (g) Energy evolution either (e) with or (f) without multiprecision; dots indicate increase of precision; the other curve corresponds to using the finest precision only.

sinusoidal nonconvex observation data model, and $V(x)$ $=|x|$ for the pairwise interaction, discontinuity-preserving, potential. In this example, ψ plays the role of a noisy observation of ϕ in a Gaussian additive model. Figure 4(a) shows an image that corresponds to a discretized pyramid

with additive Gaussian noise ($\sigma=1$). Figure 4(b) shows the image in Fig. 4(a) denoised by applying algorithm 1; the result is very good. Figure 4(c) illustrates the energy decreasing versus iterations. The curve with dots represents the energy evolution obtained by using multipreci-

sion; each dot corresponds to a change of precision. The other curve corresponds to a performance of the algorithm in the finest precision from the beginning. It can be seen that both ways we end up with the same energy (and in fact the same denoised image), as expected by using a convex energy function; however, multiprecision makes the algorithm much faster. In the second example [Figs. 4(d)–4(g)] we illustrate the absolute phase estimation using the energy [Eq. (4)] as almost elsewhere in the paper. Figure 4(d) shows the wrapped phase corresponding to a true phase given by a clipped Gaussian with 14π rad height. Figure 4(e) shows a perfect unwrapping obtained by algorithm 1 with the multiprecision approach. Figure 4(f) displays the completely failed unwrapping that one obtains by running the algorithm with the finest precision from the very beginning. Finally, Fig. 4(g) displays the energy evolutions both with and without multiprecision; the curve with dots, corresponds to the multiprecision run (each dot represents a precision change). These plots illustrate that multiprecision avoids the poor-energy local minima that are obtained without multiprecision. Again multiprecision enhances the speed of the algorithm.

In the next section, we present a series of experiments confirming the effectiveness of the proposed approach in absolute phase estimation.

3. EXPERIMENTAL RESULTS

In this section, we present four experiments illustrating the performance of the proposed algorithm: the first three concern synthetic data generated according to the model shown in Eq. (1) and with $A=1$; the fourth deals with In-SAR data distributed with [12], which is a commonly used benchmark to score absolute phase estimation algorithms. In all the experiments, we have employed a depth of precision (see Algorithm 1) $N=8$, which gives a minimum precision interval of $2\pi/2^8$ rad $\simeq 2.5 \times 10^{-3}$ rad. The regularization parameter μ [see Eq. (5)] and the exponent p [see Eq. (7)] of the half-quadratic discontinuity-preserving potential were hand-tuned for the best performance. We note, however, that in the four experiments the regularization parameter takes values in the set [0.2,0.4] and that the exponent p takes values close to 0.5 when the original absolute phase has discontinuities, and equal to 2 when there are no discontinuities. These values conform to the rationale given in Subsection 2.A: to preserve discontinuities, the exponent p of the half-quadratic potential must satisfy $p<1$, and thus it is nonconvex. On the other hand, if there are no discontinuities to preserve, convex potentials are preferable (e.g., $p=2$), as they impose smoothness on the estimated surfaces and lead to easier optimization problems.

In this section we use the following error measures:

1. RMSE $\equiv \mathrm{std}(\hat{\phi} - \phi)$,
2. ISNR $\equiv 10 \log 10 |e^{j\phi} - e^{j\psi}|^2 / |e^{j\hat{\phi}} - e^{j\psi}|^2$,

where ϕ is the true absolute phase, ψ the noisy wrapped phase, $\hat{\phi}$ the estimated absolute phase, and std$(\cdot)$ denotes the sample standard deviation. RMSE (root mean squared error) is a measure of the total error, whereas

ISNR (improvement in signal-to-noise ratio), introduced in [26], is a measure of the noise reduction independent of the phase unwrapping.

We note that in all the experiments, owing to discontinuities (original phase differences with magnitude greater than π), noise, or aliasing (due, e.g., to subsampling), there is a huge number of neighboring pixels having phase differences greater than π in magnitude, making the absolute phase estimation a very difficult problem.

A. Synthetic Data
In this section, the standard deviation of the Gaussian noise is set to $\sigma=0.5$ which corresponds to a signal-to-noise ratio SNR$\equiv 1/\sigma^2$ [see generation model (1)] of 6.02dB. We should remark that in this scenario, which poses a very hard task given the amount of noise, we have run for each experiment a Monte Carlo simulation with ten iterations.

1. Sheared Ramp
For this experiment, we set the prior parameter $\mu=0.4$ and the potential with exponent $p=0.4$. Figure 5(a) displays an image (100×150 pixels) corresponding to an absolute phase surface formed by two equal sized planes with slopes, respectively, of 1 and 0 rad/pixel (maximum height difference is 99 rad); Fig. 5(b) displays the image shown in Fig. 5(a) wrapped and noisy; Fig. 5(c) shows the image in Fig. 5(b) unwrapped and denoised; and Fig. 5(d) displays a wrapped version of estimated phase shown in Fig. 5(c).

The original absolute phase image displays a vertical discontinuity between the two planes. Thus, from the absolute phase estimation point of view, the two planes correspond to two disconnected images, rendering a very hard estimation problem, if no external discontinuity information is used. Assuming that the phase estimation algorithm is able to blindly detect discontinuities, the most we can hope for is to obtain two planes correctly estimated up to an unknown 2π multiple constant phase difference between them. The proposed algorithm accomplishes that almost perfectly, without any additional information such as quality maps [12, Chap. 3]. The total root mean square error RMSE$=[(n_1 \mathrm{var}_1 + n_2 \mathrm{var}_2)/(n_1 + n_2)]^{1/2}$, where n_i and var$_i$ are, respectively, the number of pixels and the sample variance of the estimated absolute phase surface and $i=1,2$, is RMSE$=0.14$ rad. The improvement in signal-to-noise ratio, as defined above, is ISNR$=8.99$ dB. The number of wrong wrap counts (number of wrong multiples of 2π throughout the image) is 0 (Monte Carlo simulation).

Figure 5(h) displays the discontinuities that the unwrapping step is able to blindly detect, as mentioned in Subsection 1.A. Figure 5(d) shows the rewrapped image shown in Fig. 5(c). As already noted, the shear discontinuity plus the noise poses a very hard task; the good unwrapping and denoising are noticeable. Figure 5(e) shows a 3D rendering of the image shown in Fig. 5(c); Fig. 5(f) shows a corresponding 3D rendering after unwrapping and before denoising. Figure 5(g) shows the descending objective function along the iterations of the minimization algorithm.
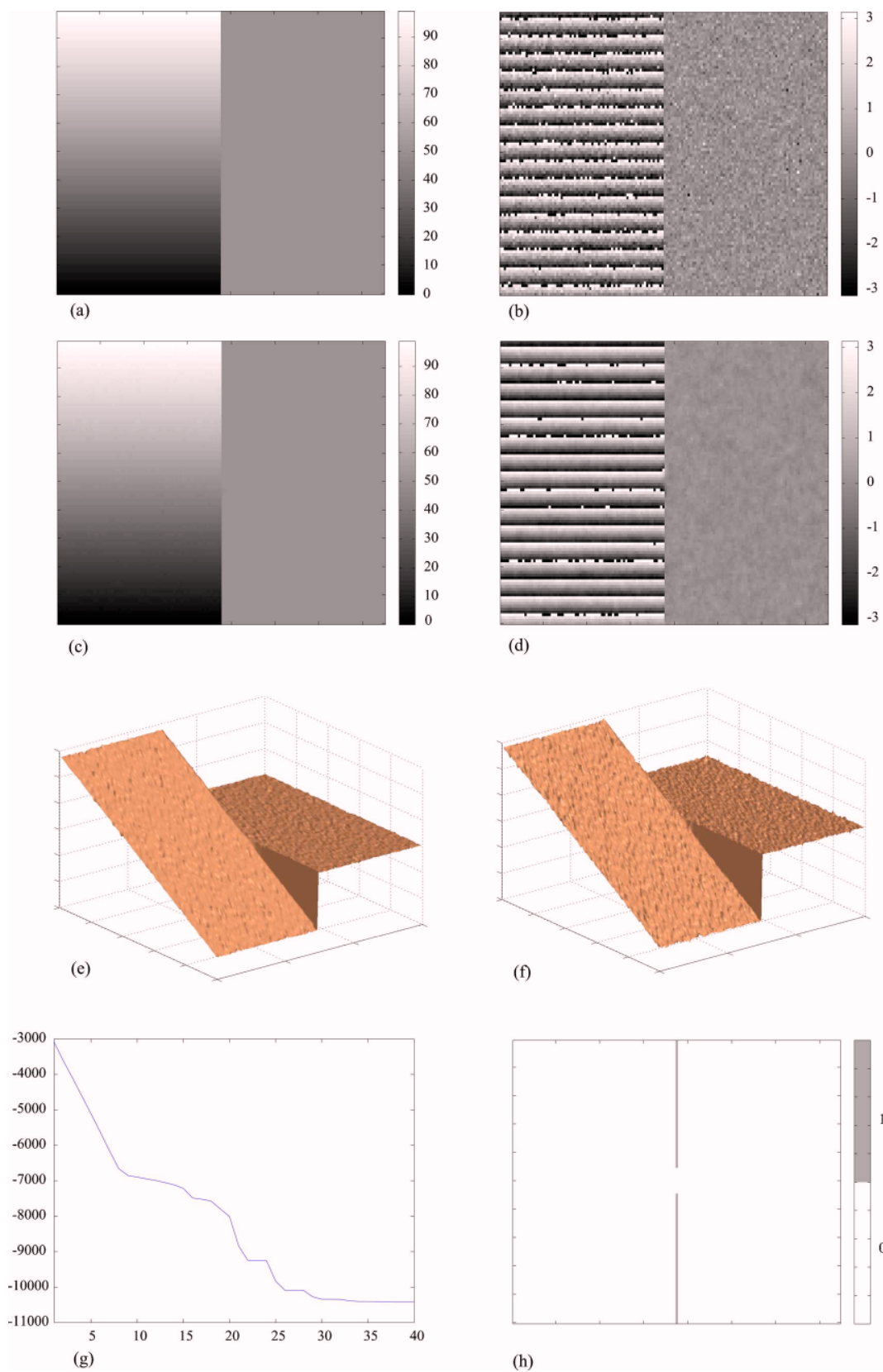
Fig. 5.   (Color online) (a) Sheared ramp image (99 rad height) (b) Wrapped and noisy image shown in (a). (c) Image in (b) unwrapped and denoised by our algorithm (RMSE=0.14 rad). (d) Image in (c) rewrapped. (e) 3D rendering of image in (c). (f) 3D rendering of the image in (c) before denoising. (g) Objective function evolution along the iterations of the algorithm. (h) Discontinuities blindly detected by the algorithm.

2. Clipped Gaussian

For this experiment, we used prior parameter $\mu = 0.4$ and the potential with exponent $p = 0.4$. Figure 6 is analogous to Fig. 5, but now the original absolute phase surface is a 100×100 pixel sized Gaussian elevation with a height of 14π rad and standard deviations of $d = 15$ pixels (vertically) and $d = 10$ pixels (horizontally); additionally, in a quarter of the plane the Gaussian has zero height, introducing surface discontinuities. We stress that this is a hard absolute phase estimation problem given these discontinuities plus the noise. The absolute phase estimate shown in Fig. 6(c) has RMSE = 0.7 rad and ISNR = 7.85 dB. The number of wrong wrap counts in the image is 20.4 (Monte Carlo simulation). The obtained reconstruction is almost perfect.

In contrast to the shear ramp, there is just one connected component in this example, and then, up to a 2π multiple constant phase, the solution is unique. The reason is that, for a given data mismatch, the adopted potential minimizes the number of discontinuities, and any other solution having the same data mismatch would have more discontinuities.

We emphasize that the quarter of the Gaussian having zero height, plus the noise, introduces a lot of discontinuities, which the unwrapping algorithm is able to detect (the algorithm also marks other pixels as discontinuities as a result of noise plus high phase rate effects) as shown in Fig. 6(h). The denoising effect is quite evident in Fig. 6(d). The 3D rendering of Figs. 6(e) and 6(f) illustrates the unwrapping plus denoising and unwrapping effects, respectively; in Fig. 6(g) is shown the evolution of the objective function along with the iterations.

We again mention that binary optimizations such as those employed in lines 4 and 12 of Algorithm 1 are demanding not only because they handle objective functions with a huge (typical of usual phase unwrapping problems) number of discrete variables, but also because those functions have, for nonconvex potentials, nonsubmodular terms (see Subsection 1.C).

3. Gaussian

For this experiment we used prior parameter $\mu = 0.4$ and the potential with exponent $p = 2$. We found better results with such a relatively high exponent p (compared with the p used in the last presented results); this can be explained by the fact that the original surface does not have discontinuities, although it has high phase rates that may create problems when noise is added. Figure 7 is similar to Fig. 6, but now the original absolute phase surface is a 100×100 pixel sized Gaussian elevation with a height of 14π rad and standard deviations of $d = 15$ pixels (vertically) and $d = 10$ pixels (horizontally). Figure 6(c) exhibits an almost perfect unwrapping and RMSE = 0.15 rad. This denoising corresponds to ISNR = 5.74 dB and is quite evident in Fig. 7(d) where we show the rewrapped denoised image. The 3D rendering of Figs. 7(e) and 7(f) illustrates the unwrapping [Fig. 7(f)] and unwrapping plus denoising effects [Fig. 7(e)]. In Fig. 7(g) we show the evolution of the objective function along with the iterations. The wrong wrap count is 0 (Monte Carlo simulation).

B. Real Data

Finally, we illustrate the performance of the algorithm on a (152×458 pixel) InSAR image. We have employed a prior parameter $\mu = 0.2$ and a quadratic potential $p = 2$. Figure 8(a) displays an image corresponding to an absolute phase surface generated by a (simulated) InSAR acquisition for a real steep-relief area (Long's Peak, Colorado, USA, data distributed with book [12]), thus inducing many discontinuities and posing a very hard absolute phase estimation problem. Figure 8(b) displays a corresponding wrapped and noisy image. In some areas the characteristic fringes are destroyed because of typical phenomena such as shadowing and layover (see, e.g., [12]). Figure 8(f) shows a quality map that is an input to the algorithm: white color corresponds to pixels whose phase value is meaningless, and gray corresponds to the rest of the pixels. This information accounts for jamming phenomena such as the above mentioned layover and shadowing. Our algorithm is able to screen those white pixels from the absolute phase estimation process (therefore those pixels do not contaminate the results for the rest of the image), and this is the reason why we employ a non-discontinuity-preserving quadratic potential. Obviously the error values here presented refer only to gray image areas on the quality map (in fact we also do not count pixels at the border of the gray image, as well as those pixels from the first and the last columns of the image; this is so because such values are outliers driven by data acquisition artifacts). Figure 8(c) shows the unwrapped resulting image, with RMSE = 0.18 rad corresponding to ISNR = 3.8 dB. Figure 8(d) displays the image in Fig. 8(c) rewrapped. Comparing with the image shown in Fig. 8(b), the denoising effect is apparent; this is made quite clear by comparing the enlarged patches in Figs. 8(b) and 8(d). Figure 8(e) displays a 3D rendering of the image in Fig. 8(c). We further add that the performance took 35 iterations (within Algorithm 1), and we got one wrap count error.

C. Benchmarking

In this section, we benchmark the proposed algorithm CAPE (combinatorial absolute phase estimation) against the state-of-the-art competitors PEARLS (phase estimation using adaptive regularization based on local smoothing), introduced in [26], and LPN0, presented in [12], Chap. 5. The benchmarks are evaluated on the experiments addressed in the previous sections, with the exception that we run the experiments for a set of noise values $\sigma = \{0.1, 0.3, 0.5, 0\}$ (see Subsection 2.A for the definition of σ). We emphasize that we do not benchmark against $\mathbb{Z}\pi$M [13] (an algorithm that we have already mentioned above) because both PEARLS and CAPE can be considered as extensions of it.

Table 1 summarizes the quantitative results obtained for the benchmarks. The bullets indicate that, for our purposes, that experiment is not relevant. In fact LPN0 aims at achieving phase unwrapping (not denoising) and thus it is not fair to apply it to noisy phase images.

We further add that for noiseless cases ($\sigma = 0$) both CAPE and PEARLS run in the phase unwrapping mode,
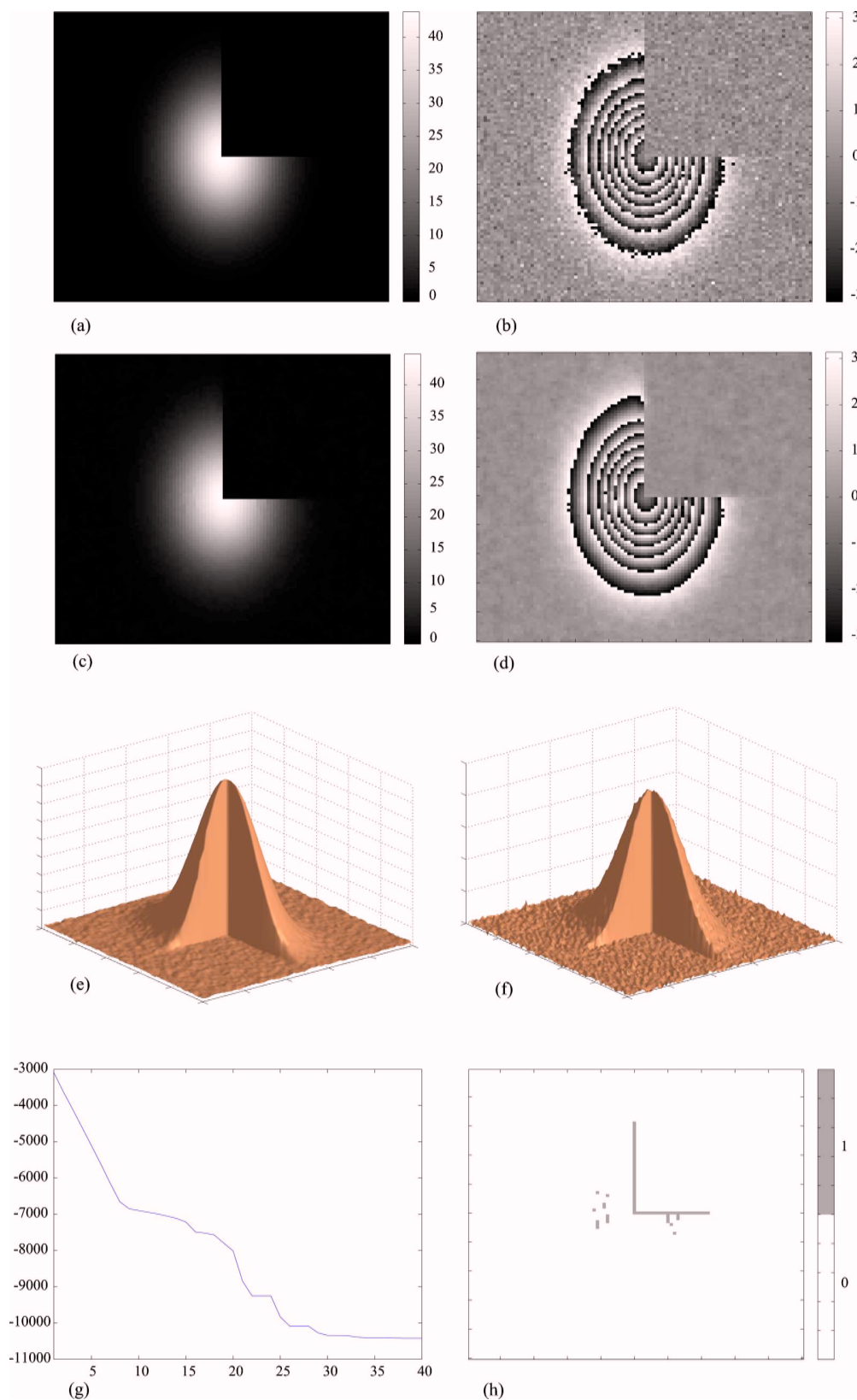
Fig. 6. (Color online) (a) Gaussian image with 14π maximum height and with a quarter set to zero. (b) Wrapped and noisy image shown in (a). (c) Image in (b) unwrapped and denoised by our algorithm (RMSE=0.7 rad). (d) Image in (c) rewrapped. (e) 3D rendering of image in (c). (f) 3D rendering of the image in (c) before denoising. (g) Objective function evolution along the iterations of the algorithm. (h) Discontinuities blindly detected by the algorithm.
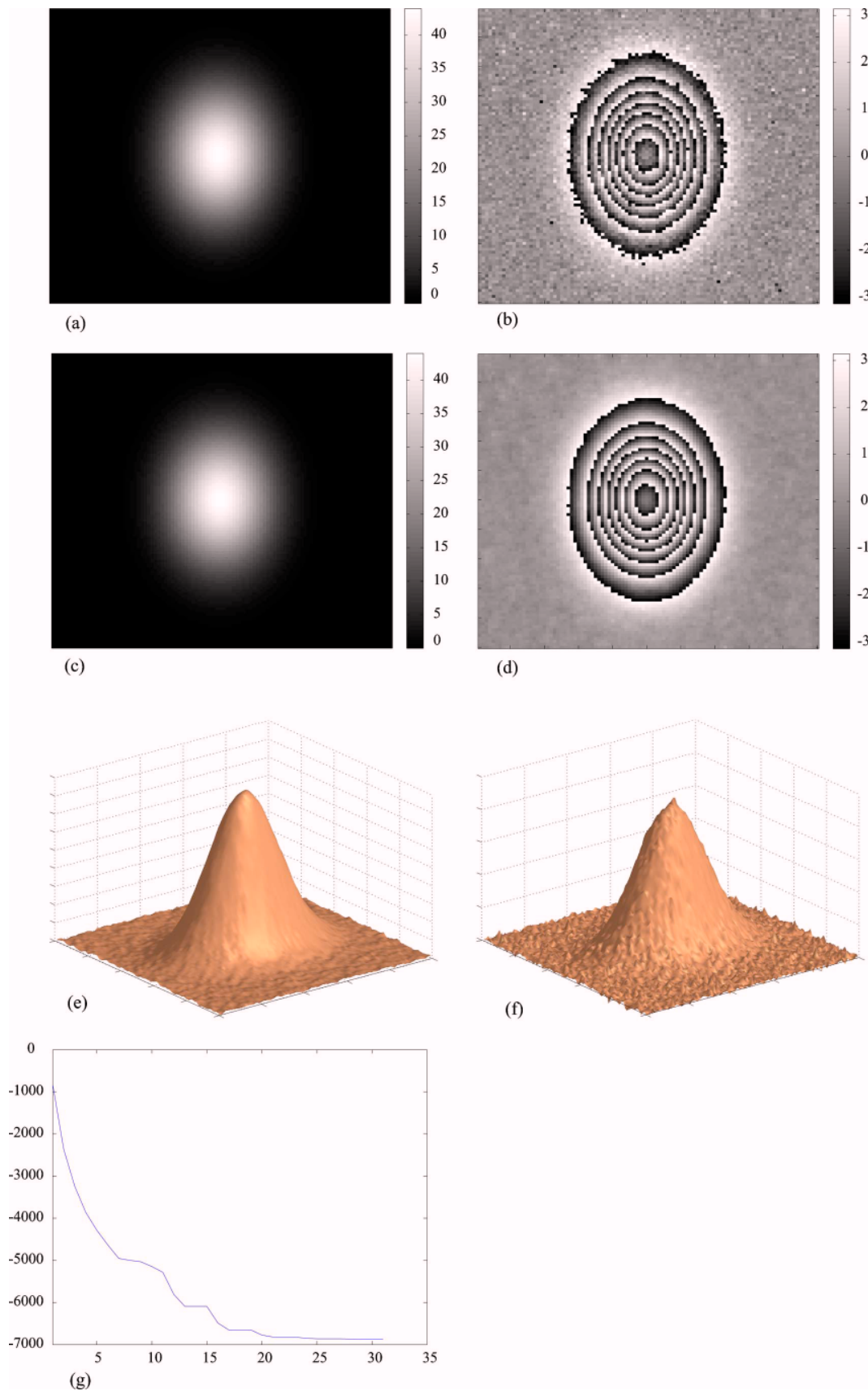
Fig. 7.   (Color online) (a) Gaussian image with 14π maximum height. (b) Wrapped and noisy image shown in (a). (c) Image in (b) unwrapped and denoised by our algorithm (RMSE=0.15 rad). (d) Image in (c) rewrapped. (e) 3D rendering of image in (c). (f) 3D rendering of the image in (c) before denoising. (g) Objective function evolution along the iterations of the algorithm.
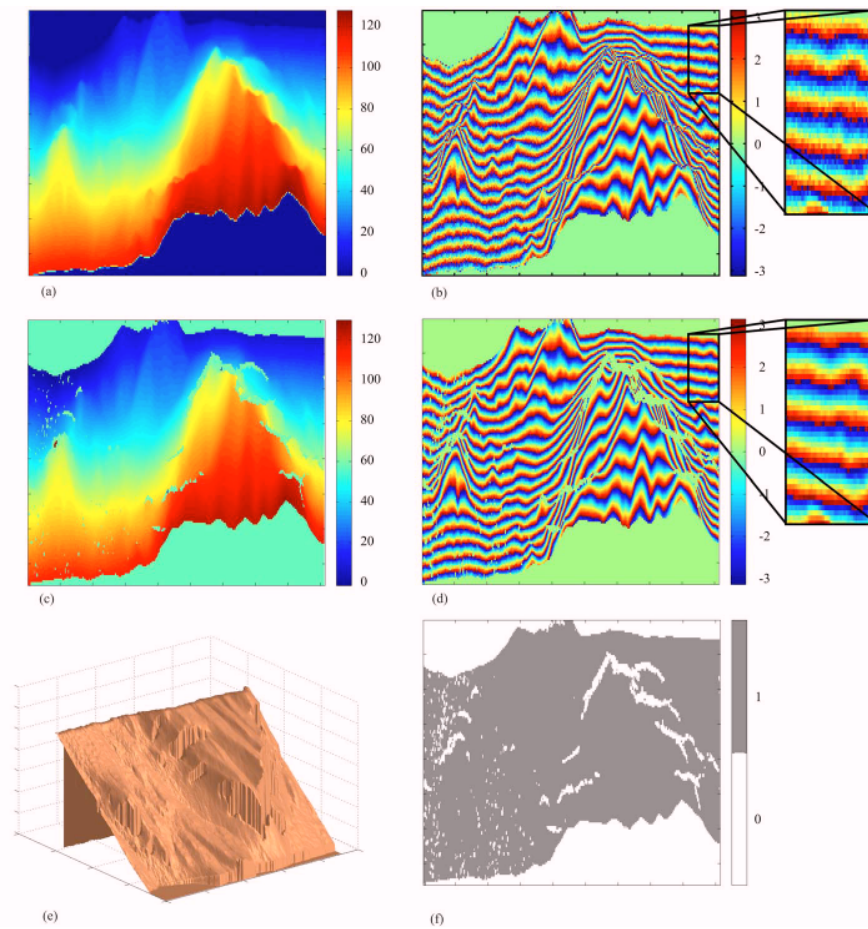
Fig. 8. (Color online) (a) Absolute phase gray-level image generated by a (simulated) InSAR acquisition for a real steep-relief area. (b) Wrapped and noisy image shown in (a). (c) Image in (b) unwrapped and denoised by our algorithm (RMSE = 0.18 rad). (d) Image in (c) rewrapped. (e) 3D rendering of the image in (c). (f) Quality map that is an input to the algorithm: white corresponds to pixels whose phase value is meaningless, and gray corresponds to the rest of the pixels.

Table 1. Root Mean Square Error (rad)

Experiments	Algorithms		
	CAPE	PEARLS	LPN0
Gaussian			
$\sigma=0$	0	0	0
$\sigma=0.1$	0.05	0.05	•
$\sigma=0.3$	0.11	0.08	•
$\sigma=0.5$	0.15	0.11	•
Shear ramp			
$\sigma=0$	0	0	1.21
$\sigma=0.1$	0.06	0.07	•
$\sigma=0.3$	0.10	0.09	•
$\sigma=0.5$	0.14	0.11	•
Clipped Gaussian			
$\sigma=0$	0	0	5.48
$\sigma=0.1$	0.13	0.85	•
$\sigma=0.3$	0.4	0.90	•
$\sigma=0.5$	0.7	0.98	•
Long's Peak	0.18	0.20	•

i.e., CAPE runs with $N=0$ (see Algorithm 1), which corresponds to running as PUMA [25], and PEARLS also runs as PUMA in this case [26].

We note the following:

1. Both CAPE and PEARLS tend to outperform LPN0 in phase unwrapping ($\sigma=0$ case) when discontinuities exist to deal with blindly.

2. CAPE tends to outperform PEARLS when there are discontinuities. As we have previously stated, PEARLS implements a local polynomial approximation (LPA) with varying adaptive neighborhood. This adaptiveness trades bias with variance in such a way that the neighborhood's window stretches in areas where the underlying true phase is smooth and shrinks otherwise, namely, in the presence of discontinuities. It happens that, sometimes, this shrinkage is not enough, implying a few unwrapping errors near these discontinuities. This phenomenon is illustrated in the results for the clipped Gaussian.

3. PEARLS tends to outperform CAPE when there are no discontinuities and the SNR is low. The adaptive window selection technique jointly with LPA results are very powerful. This is illustrated in the Gaussian experiment. In the Long's Peak experiment PEARLS and CAPE tend to behave very similarly (the slight difference in favor of

CAPE is meaningless because, as we have already noted, the error is evaluated in a subset of the image, which may have slight differences in the experiments using both PEARLS and CAPE).

To our knowledge, PEARLS is the state-of-the-art absolute phase estimation algorithm. From all of the experiments, CAPE can also be considered state-of-the-art. Finally, we remark that code has been run in a 2.2 GHz Intel dual core processor, in a maximum of a few dozens of seconds; CAPE has been consistently observed to be faster than PEARLS (we should note that, even so, the code is a mix of MATLAB and C++, and therefore is not optimized).

4. CONCLUDING REMARKS

We have introduced a (discontinuity-preserving) denoising Bayesian approach to absolute phase estimation in interferometric applications. In the scientific community there is still a lively debate of whether denoising should be done after phase unwrapping, before phase unwrapping, or any other solution in between. In this paper we have chosen the first option in order to avoid having the denoising step corrupt the phase unwrapping process. The graph-cuts-based proposed algorithm, CAPE, first performs phase unwrapping, with discontinuity-preserving capabilities; then, basically using the same rationale, but with a multiprecision scheduling technique, it achieves denoising. In the experimental results the simplicity of the algorithm translates into fast execution and, furthermore, high accuracy. We emphasize that a short running time can be essential for some applications (e.g., some medical applications in magnetic resonance imaging). A further step to speed up the algorithm may be obtained in the future by using dynamic graph cuts [50].

The obtained experimental results for absolute phase estimation have, to our knowledge, state-of-the-art accuracy; the running time is typically a few seconds in a 2.2 GHz Intel dual core for images of size 150×100, and was experimentally found to grow linearly with image size. Theoretically, the worst-case complexity is $O(kn^2m)$ (see [25]), where $n = |\mathcal{V}|$ and $m = |\mathcal{E}|$, respectively, are the number of nodes and edges of the graph that our algorithm deals with, and k is the excursion of the 2π labels of the nodes in the absolute phase. We emphasize that our algorithm, in a set of representative experiments, turns out to be faster than PEARLS [26].

In the future, we intend to explore diversity techniques (e.g., Chinese remainder theorem) applied to absolute phase estimation, in order to extend the set of the phase images that we can unwrap (namely, images for which the absolute phase exhibits high rates); we should note that dealing with noise in such a scenario turns out to be quite more challenging. We also intend to employ higher-order Markov random fields; this is an effort to avoid the metrication artifacts well known to be associated with graph-cut-based techniques [51]. In addition we intend to develop learning schemes for the selection of the best prior parameter μ, given some absolute phase estimation problem.

REFERENCES

1. P. Rosen, S. Hensley, I. Joughin, F. LI, S. Madsen, E. Rodriguez, and R. Goldstein, "Synthetic aperture radar interferometry," Proc. IEEE **88**, 333–382 (2000).
2. P. Lauterbur, "Image formation by induced local interactions: examples employing nuclear magnetic resonance," Nature **242**, 190–191 (1973).
3. D. L. Fried, "Adaptive optics wave function reconstruction and phase unwrapping when branch points are present," Opt. Commun. **200**, 43–72 (2001).
4. T. Venema and J. Schmidt, "Optical phase unwrapping in the presence of branch points," Opt. Express **16**, 6985–6998 (2008).
5. S. Pandit, N. Jordache, and G. Joshi, "Data-dependent systems methodology for noise-insensitive phase unwrapping in laser interferometric surface characterization," J. Opt. Soc. Am. A **11**, 2584–2592 (1994).
6. A. Devaney, "Diffraction tomographic reconstruction from intensity data," IEEE Trans. Image Process. **1**, 221–228 (1992).
7. A. Rauscher, M. Barth, J. Reichenbach, R. Stollberger, and E. Moser, "Automated unwrapping of MR phase images applied to BOLD MR-venography at 3 Tesla," J. Magn. Reson. Imaging **18**, 175–180 (2003).
8. A. Manduca, T. Oliphant, M. Dresner, J. Mahowald, S. Kruse, E. Amromin, J. Felmlee, J. Greenleaf, and R. Ehman, "Magnetic resonance elastography: non-invasive mapping of tissue elasticity," Med. Image Anal. **5**, 237–254 (2001).
9. M. C. Roggemann and B. M. Welsh, *Imaging Through Turbulence* (CRC, 1996).
10. K. Itoh, "Analysis of the phase unwrapping problem," Appl. Opt. **21**, 2470 (1982).
11. J. Hadamard, "Sur les problèmes aux dérivées partielles et leur signification physique," Princeton University Bulletin (1902), pp. 49–52.
12. D. Ghiglia and M. Pritt, *Two-Dimensional Phase Unwrapping. Theory, Algorithms, and Software* (Wiley, 1998).
13. J. Dias and J. Leitão, "The $\mathbb{Z}\pi$M algorithm for interferometric image reconstruction in SAR/SAS," IEEE Trans. Image Process. **11**, 408–422 (2002).
14. V. Katkovnik, J. Astola, and K. Egiazarian, "Phase local approximation (phasela) technique for phase unwrap from noisy data," IEEE Trans. Image Process. **17**, 833–846 (2008).
15. R. Goldstein, H. Zebker, and C. Werner, "Satellite radar interferometry: two-dimensional phase unwrapping," in *Symposium on the Ionospheric Effects on Communication and Related Systems* (Radio Science, 1988), Vol. 23, pp. 713–720.
16. J. Huntley, "Noise-immune phase unwrapping algorithm," Appl. Opt. **28**, 3268–3270 (1989).
17. T. Flynn, "Two-dimensional phase unwrapping with minimum weighted discontinuity," J. Opt. Soc. Am. A **14**, 2692–2701 (1997).
18. D. Ghiglia and L. Romero, "Minimum L^p-norm two-dimensional phase unwrapping," J. Opt. Soc. Am. A **14**, 1–15 (1996).
19. M. Costantini, "A novel phase unwrapping method based on network programing," IEEE Trans. Geosci. Remote Sens. **36**, 813–821 (1998).
20. J. Marroquin and M. Rivera, "Quadratic regularization functionals for phase unwrapping," J. Opt. Soc. Am. A **12**, 2393–2400 (1995).
21. J. Leitão and M. Figueiredo, "Absolute phase image reconstruction: a stochastic nonlinear filtering approach," IEEE Trans. Image Process. **7**, 868–882 (1997).

22. M. Datcu and G. Palubinskas, "Multiscale Bayesian height estimation from insar using a fractal prior," Proc. SPIE **3497**, 155–163 (1998).

23. G. Nico, G. Palubinskas, and M. Datcu, "Bayesian approach to phase unwrapping: theoretical study," IEEE Trans. Signal Process. **48**, 2545–2556 (2000).

24. M. Rivera and J. Marroquin, "Half-quadratic cost functions for phase unwrapping," Opt. Lett. **29**, 504–506 (2004).

25. J. Bioucas-Dias and G. Valadão, "Phase unwrapping via graph cuts," IEEE Trans. Image Process. **16**, 698–709 (2007).

26. J. Bioucas-Dias, V. Katkovnik, J. Astola, and K. Egiazarian, "Absolute phase estimation: adaptive local denoising and global unwrapping," Appl. Opt. **47**, 5358–5369 (2008).

27. R. Ahuja, D. Hochbaum, and J. Orlin, "Solving the convex cost integer dual network flow problem," Manage. Sci. **49**, 950–964 (2003).

28. M. Iri, *Network Flow, Transportation and Scheduling— Theory and Algorithms* (Academic, 1969).

29. K. Murota, "On steepest descent algorithms for discrete convex functions," SIAM J. Optim. **14**, 699–707 (2003).

30. R. Rockafellar, *Network Flows and Monotropic Optimization* (Wiley, 1984).

31. J. Bioucas-Dias and G. Valadão, "Discontinuity preserving phase unwrapping using graph cuts," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, A. Rangarajan, B. Vemuri, and A. Yuille, eds., Vol. 3757 of Lecture Notes in Computer Science (Springer, 2005), pp. 268–284.

32. Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," IEEE Trans. Pattern Anal. Mach. Intell. **26**, 1124–1137 (2004).

33. V. Kolmogorov, "Primal-dual algorithm for convex Markov random fields," Tech. Rep. MSR-TR-2005-117 (Microsoft Research, 2005).

34. V. Kolmogorov and A. Shioura, "New algorithms for the dual of the convex cost network flow problem with application to computer vision." Tech. Rep. (University College London, 2007).

35. J. Darbon, "Composants logiciels et algorithmes de minimisation exacte d'énergies dédiés au traitement des images," Ph.D. thesis (Ecole Nationale Supérieure des Télécommunications, 2005).

36. K. Murota, *Discrete Convex Analysis* (Society for Industrial and Applied Mathematics, 2003).

37. H. Ishikawa, "Exact optimization for Markov random fields with convex priors," IEEE Trans. Pattern Anal. Mach. Intell. **25**, 1333–1336 (2003).

38. J. Darbon, "Global optimization for first order Markov random fields with submodular priors," Discrete Appl. Math. (to be published).

39. V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" IEEE Trans. Pattern Anal. Mach. Intell. **26**, 147–159 (2004).

40. K. Lange, *Optimization* (Springer Verlag, 2004).

41. Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," IEEE Trans. Pattern Anal. Mach. Intell. **23**, 1222–1239 (2001).

42. C. Chen, "Statistical-cost network-flow approaches to two-dimensional phase unwrapping for radar interferometry," Ph.D. thesis (Stanford University, 2001).

43. A. Blake and A. Zisserman, *Visual Reconstruction* (MIT Press, 1987).

44. D. Geman and G. Reynolds, "Constrained restoration and the recovery of discontinuities," IEEE Trans. Pattern Anal. Mach. Intell. **14**, 367–383 (1992).

45. S. Z. Li, *Markov Random Field Modeling in Computer Vision*, Vol. 9 of Computer Science Workbench (Springer-Verlag, 1995).

46. D. Greig, B. Porteous, and A. Seheult, "Exact maximum a posteriory estimation for binary images," J. R. Stat. Soc. Ser. B (Methodol.) **51**, 271–279 (1989).

47. R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, "A comparative study of energy minimization methods for Markov random fields with smoothness-based priors," IEEE Trans. Pattern Anal. Mach. Intell. **30**, 1068–1080 (2008).

48. L. Ford and D. Fulkerson, *Flows in Networks* (Princeton Univ. Press, 1962).

49. C. Rother, V. Kolmogorov, V. Lempitsky, and M. Szummer, "Optimizing binary MRFs via extended roof duality," in *IEEE Conference on Computer Vision and Pattern Recognition 2007. CVPR, '07* (IEEE, 2007), pp. 1–8.

50. P. Kohli and P. Torr, "Dynamic graph cuts for efficient inference in Markov random fields," IEEE Trans. Pattern Anal. Mach. Intell. **29**, 2079–2088 (2007).

51. Y. Boykov and V. Kolmogorov, "Computing geodesics and minimal surfaces via graph cuts," in *Proceedings of Ninth IEEE International Conference on Computer Vision 2003* (IEEE, 2003), Vol. 1, pp. 26–33.