



A Self-Referencing Level-Set Method for Image Reconstruction from Sparse Fourier Samples*

JONG CHUL YE

Philips Research - USA, 345 Scarborough Road, Briarcliff Manor, New York 10510, USA

jong.ye@philips.com

YORAM BRESLER AND PIERRE MOULIN

Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, 1308 W. Main Street, Urbana, IL 61801, USA

Received December 6, 2001; Revised July 23, 2002; Accepted July 23, 2002

Abstract. We address an ill-posed inverse problem of image estimation from sparse samples of its Fourier transform. The problem is formulated as joint estimation of the supports of unknown sparse objects in the image, and pixel values on these supports. The domain and the pixel values are alternately estimated using the level-set method and the conjugate gradient method, respectively. Our level-set evolution shows a unique switching behavior, which stabilizes the level-set evolution. Furthermore, the trade-off between the stability and the speed of evolution can be easily controlled by the number of the conjugate gradient steps, thus avoiding the re-initialization steps in conventional level set approaches.

Keywords: inverse problems, Fourier imaging, medical imaging, partial-data, level-set, geometry-driven diffusion

1. Introduction

Reconstruction of an image from samples of its Fourier transform is a classical problem in image processing. Examples include synthetic aperture radar (SAR), radio astronomy, and magnetic resonance imaging (MRI) (Blahut, 2001). If enough Fourier samples are available, the reconstruction can be easily obtained using the inverse Fourier transform. However, in many practical situations, owing to physical or economic limitations, only a small number of Fourier samples is available.

With sparse Fourier data, the reconstruction problem becomes ill-posed, because it does not admit a unique solution. However, if the image can be assumed to consist of objects supported on a small unknown set, or *do-*

main D on a known background (Fig. 1(a)), a unique solution exists, provided the sampling pattern satisfies appropriate so-called universality conditions (Bresler and Feng, 1996; Venkataramani and Bresler, 1998; Bresler et al., 1999). These conditions are typically satisfied by random sampling patterns. As will be explained later in Section 2, the computation of a reconstruction requires, however, the solution of a difficult non-convex nonlinear optimization problem, where the functional should be simultaneously minimized with respect to the unknown image values (grey levels) and domain D .

A related (but somewhat different, as elaborated below) optimization problem has appeared in Mumford and Shah's edge detection and segmentation algorithm (Mumford and Shah, 1989):

$$\min_{v, \Gamma} \left(\int_{\Omega} |y - v|^2 d\mathbf{x} + \lambda \int_{\Omega - \Gamma} |\nabla v|^2 d\mathbf{x} + v \int_{\Gamma} d\Gamma \right), \quad (1)$$

*This work was supported by a grant from DARPA under Contract F49620-98-1-0498, administered by AFOSR, and by NSF Infrastructure Grant CDA 96-24396.

where y is the input image, v is the output image, Ω denotes the image domain, Γ labels the discontinuities, and λ , ν denote relative weighting parameters for smoothness, and the domain boundary length terms, respectively. As observed in (1), this optimization problem also requires a minimization with respect to the unknown set Γ , and is nonconvex. One of the first practical numerical implementations of the Mumford-Shah model was developed by Richardson (1989). Instead of using the original functional (1), their implementation is based upon an elliptical approximation of the functional, considered by Ambrosio and Tortorelli (1990). In this approximation, the boundaries between modeled homogeneous regions were “smeared” into a set with nonzero Lebesgue measure. The elliptical approximation leads to a coupled system of two equations, to which standard PDE methods can be applied (for extensive lists of such approaches, see ter Haar Romeny (1994) and the references therein). Instead of resorting to elliptical approximations, recently, Tsai et al. presented a curve evolution method for minimizing the *original* Mumford-Shah functional (1), where simultaneous image segmentation and smoothing are achieved in a coupled manner (Tsai et al., 2001). Chan and Vese independently presented a level set approach for minimizing the Mumford-Shah functional using multi-phase level set functions (Chan and Vese, 1999, 2000). For general segmentation problems, numerous recent papers use region-based active contour models and level set algorithms (Chakraborty et al., 1996; Jchan-Besson et al., 2001; Paragios and Deriche, 1999; Zhu and Yuille, 1996).

While the optimization problem of Mumford-Shah (1), and to a lesser degree, other segmentation problems, have many features in common with the sparse Fourier imaging problem, there exists however a fundamental difference which hampers the direct application of existing minimization algorithms for the Mumford-Shah functional, or of other segmentation techniques, to our problem. In segmentation problems, the image is observed directly, while in the sparse Fourier imaging problem the image is observed through a many-to-one mapping, and the locality of observations is lost: each data sample depends on all the pixels of the observed image.

For such inverse problems, one can consider deterministic edge-preserving regularization (EPR). EPR has been extensively employed for many ill-posed inverse problems such as computed tomography (CT), positron emission tomography (PET), and Fourier

imaging (Geman and Yang, 1995). In particular, the so-called ARTUR algorithm (Charbonnier et al., 1997) is, in principle, similar to the elliptical approximation for Mumford-Shah functional: the analog edge map of the image is iteratively updated, reducing smoothing across the edges. Delaney and Bresler (1998) provided a convergence proof for a generalized version of ARTUR, and showed excellent image reconstruction results in limited-angle tomography problems. For binary image estimation, Santosa (1996) presented a level-set algorithm that estimates the image domain using the level set evolution in a steepest descent direction, assuming that the binary image magnitude is known. Santosa’s approach is unique in the sense that he does not resort to elliptical approximation of the functional and estimates the geometric information from the original cost functional; however, his algorithm can be only applied to binary images. In tomographic reconstruction problems, Yu and Fessler (2001) incorporated the image discontinuity information in the form of a regularization term and updated the term using a level-set algorithm. Their approach is more general than Santosa’s because it can deal with piecewise smooth images. However, they incorporated the domain discontinuity information only in the form of the regularization term (rather than the data terms in the Santosa’s approach), so the boundary deformation is not controlled by the data mismatch. Hence, the level set evolution is not guaranteed to stop even when there is no data mismatch. In addition, instead of obtaining the exact formula for the functional derivative, Yu and Fessler resorted to a numerical method.

The main contribution of this paper is a new level set algorithm, which addresses a more general problem where the image is observed through a many-to-one mapping, and both the domain of support and pixel values are unknown, and are estimated jointly, without resorting to the elliptical approximation of the functional. In addition to reconstructing non-constant objects, our approach allows to reconstruct objects on a known non-homogeneous background, neither of which is possible using Santosa’s (1996) or Yu and Fessler’s approaches (2001). The latter feature is important in tasks of “change detection” in a reference image acquired at an earlier time. Unlike Yu and Fessler’s method (Yu and Fessler, 2001), we incorporate the discontinuity information both in the data mismatch and regularization terms, hence the level-set evolution and termination is controlled by the data mismatch. Furthermore, in our level-set approach the embedding level function

is initialized using the direct Fourier inversion estimate of pixel value, rather than the conventional signed distance function. This formulation has several unique advantages. First, the initialization of the level-set and the necessary extension of the evolution velocity are simplified. Second, the resultant level-set evolution exhibits a switching behavior with significant stabilizing effects, hence the level function is never re-initialized. Finally, by combining with conjugate gradient (CG) steps of pixel value updates, the stability and the speed of the level-set evolution can be easily controlled by the number of the CG steps.

This paper is organized as follows. After the problem formulation in Section 2, we briefly explain our alternating minimization strategy in Section 3, and derive the so-called the shape derivative in Section 4. Using the shape differentials, Section 5 explains our new optimization technique—which we call the self-referencing level set method. Numerical implementation and computational cost are analyzed in Section 6, and numerical examples are provided in Section 7.

2. Problem Formulation

Let $\Omega \subset \mathbb{R}^2$ denote the image domain and the open set D a subset (which may be disconnected) of Ω . Suppose the unknown image $u(\mathbf{x})$ ($\mathbf{x} \in \Omega$) has the following form:

$$u(\mathbf{x}) = \begin{cases} v(\mathbf{x}), & \mathbf{x} \in D \\ 0 & \mathbf{x} \in \Omega \setminus D, \end{cases} \quad (2)$$

where $v(\mathbf{x})$ is an unknown pixel value at $\mathbf{x} \in \Omega$. Thus, the objects may have arbitrary values over their support. If the image $u(\mathbf{x})$ had a known nonzero background (over $\Omega \setminus D$), the model (2) would be applied to the difference between the original image and this known background.

The so-called noiseless measurement in the Fourier imaging problem is given by:

$$F^D v(\mathbf{f}) = \int_{\Omega} \exp(-j\mathbf{f} \cdot \mathbf{x}') \chi_D(\mathbf{x}') v(\mathbf{x}') d\mathbf{x}', \quad \mathbf{f} \in \Phi \subset \mathbb{R}^2, \quad (3)$$

where $\mathbf{f} \in \mathbb{R}^2$ is the frequency, Φ denotes the set of 2-D frequency sample locations, and $d\mathbf{x}'$ denotes the differential surface element. The localized Fourier transform operator $F^D : L^2(\Omega) \rightarrow L^2(\Phi)$ operates on the space of finite-energy functions defined on Ω , producing a

finite-energy transform defined on Φ . Note, for future reference, the following facts.

1. The spaces $L^2(\Omega)$ and $L^2(\Phi)$ are Hilbert spaces, with the standard L^2 inner products and norms, and F^D has the adjoint operator $F^{D*} : L^2(\Phi) \rightarrow L^2(\Omega)$ given by

$$(F^D)^* y(\mathbf{x}) = \chi_D(\mathbf{x}) \int_{\Phi} \exp(j\mathbf{f}' \cdot \mathbf{x}) y(\mathbf{f}') d\mathbf{f}'. \quad (4)$$

2. Owing to the indicator function in (4), all functions in the range space of $(F^D)^*$ vanish on $\Omega \setminus D$.

Suppose we have noisy samples $y(\mathbf{f})$, $\mathbf{f} \in \Phi$ of $F^D v$ where the sampling pattern satisfies so-called universality conditions (Bresler and Feng, 1996; Venkataramani and Bresler, 1998; Bresler et al., 1999). These conditions are typically satisfied by random sampling patterns. Then, even if the sampling rate in the Fourier domain is under the Nyquist sampling rate, the reconstruction problem admits a unique solution if the non-zero pixel domain in the image is sparse enough. Therefore, our goal is to fit the Fourier transform of the reconstructed image to the noisy samples, while imposing the sparseness of the reconstructed image. The resultant penalized least-squares estimator of D and v can be computed by

$$\min_{D,v} \left(\frac{1}{2} \|y - F^D v\|_{\Phi}^2 + \alpha \int_{\Gamma} d\Gamma \right), \quad (5)$$

where $\|z\|_{\Phi}^2 = \int_{\Phi} |z(\mathbf{f})|^2 d\mathbf{f}$, the second integral is a line-integral along the boundary $\Gamma = \partial D$ (hence the length of the domain boundary), and α is a regularization constant. Because the length of the boundary is used as a regularization functional, the size of the domain is indirectly penalized. This form of penalized likelihood can be motivated in various other ways, including one based on Risannen's minimum description length (MDL) criterion (Leclerc, 1989).

3. Alternating Minimization

Since the optimization problem (5) is nonlinear and nonconvex, the direct optimization problem is very difficult. In order to solve it, we therefore use an alternating minimization of the cost function with respect to the pixel values v and the domain D . Such an alternating optimization technique has been used for minimizing the Mumford-Shah functional (Tsai et al., 2001;

Chan and Vese, 2000). For fixed D , the optimization problem is simple, admitting the linear least-squares solution

$$\hat{v} = (F^D)^\dagger y, \tag{6}$$

where A^\dagger denotes the pseudo inverse of A . Note that the range space of $(F^D)^\dagger$ is identical to that of $(F^D)^*$, and therefore the estimate $\hat{v} : \Omega \rightarrow \mathbb{C}$ vanishes on $\Omega \setminus D$. Because of the large dimensions of the matrix involved in the discrete form of (6), the estimator \hat{v} of (6) is preferably obtained using the conjugate gradient algorithm, without ever having to form the matrix itself.

4. Shape Derivatives

4.1. Shape Deformation

The main technical difficulty of the optimization problem (5) is the estimation of the shape D . Such problems are often solved using iterative gradient-based nonlinear optimization (Luenberger, 1989). Calculation of a gradient requires the sensitivity analysis of functionals with respect to the shape deformation. Such a shape deformation is described by constructing a family of shapes D_t , $D_0 = D$ that are perturbations of D for $0 \leq t < \epsilon$, t being a fictitious time parameter. These admissible domains D_t are assumed to be subsets of a larger fixed domain Ω and their boundary is assumed to retain the same regularity C^k as that of the original domain.

The next question is how to define this family of perturbations. One could use ideas developed in continuum mechanics where a family of transformations $T_t : \bar{\Omega} \rightarrow \mathbb{R}^2$ for $t \in [0, \epsilon)$ is constructed such that (Sokolowski and Zolesio, 1991)

$$D_t = T_t(D), \tag{7}$$

where $T_t(D)$ is defined by

$$T_t(D) = \{\mathbf{x}_t : \mathbf{x}_t = T_t(\mathbf{x}), \mathbf{x} \in D\}. \tag{8}$$

The perturbation of the identity

$$T_t(\mathbf{x}) = \mathbf{x} + t\eta(\mathbf{x}), \tag{9}$$

where η is of class C^k , is a possible form for these transformations. One could also use a more general method known as the speed method (Sokolowski and

Zolesio, 1991). Instead of being defined by its transformation T_t , the deformation is defined by its speed field $V(t, \mathbf{x}) \in C([0, \epsilon), V^K(\Omega))$ where $C([0, \epsilon), V^K(\Omega))$ denotes the space of \mathbb{R}^2 -valued functions continuously differentiable with respect to $t \in [0, \epsilon)$ and K -continuously differentiable with respect to \mathbf{x} with compact support included in Ω . In order that the admissible domain D_t remain a subset of Ω , the speed field should satisfy the boundary conditions:

$$\langle V(t, \mathbf{x}), n(\mathbf{x}) \rangle = 0, \quad \text{for almost everywhere } \mathbf{x} \in \partial\Omega, \tag{10}$$

$$V(t, \mathbf{x}) = 0, \quad \text{for all singular points } \mathbf{x}, \tag{11}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product between two \mathbb{R}^2 -valued functions, $n(\mathbf{x}) \in \mathbb{R}^2$ is the outer-normal unit vector on the boundary $\partial\Omega$, and a singular point is a point where $n(\mathbf{x})$ is not defined. In fact, these two definitions of the perturbation are equivalent. Indeed, under appropriate regularity conditions, it is possible to associate a unique velocity field $V(t, \mathbf{x})$ to a given transformation, and vice versa (Sokolowski and Zolesio, 1991).

Now, let $\mathbf{x}(t)$ denote the solution to the system of ordinary differential equations

$$\frac{\partial \mathbf{x}(t)}{\partial t} = V(t, \mathbf{x}(t)), \quad \mathbf{x}(0) \in D. \tag{12}$$

Then, the following result from shape analysis allows us to use a gradient-based approach for the domain update:¹

Proposition 1. (Sokolowski and Zolesio, 1991) *Let the domain functions $Q(D)$ and $L(D)$ be given by*

$$Q(D) = \int_D Y \, d\mathbf{x}, \quad L(D) = \int_\Gamma Y \, d\Gamma, \tag{13}$$

where Γ is of class C^1 and Y is a function in $W_{loc}^{1,1}(\mathbb{R}^2)$. Suppose D_t is deformed by (12) where $V(t, \mathbf{x}) \in C([0, \epsilon), V^1(\Omega))$. Then, the domain functions $Q(D)$ and $L(D)$ are shape differentiable,

$$Q'(D, V) = \left(\frac{\partial}{\partial t} \int_{D_t} Y \, d\mathbf{x} \right)_{t=0} = \int_\Gamma Y F \, d\Gamma \tag{14}$$

$$L'(D, V) = \int_\Gamma \kappa Y F \, d\Gamma, \tag{15}$$

where $F : \Gamma \rightarrow \mathbb{R}$ is given by $F = \langle V(0), n \rangle$ and denotes the component of the velocity normal to the boundary

Γ at $t = 0$; $n(\mathbf{x})$, $\mathbf{x} \in \Gamma$ denotes outer normal vector on Γ , and $\kappa(\mathbf{x})$, $\mathbf{x} \in \Gamma$ denotes the curvature of Γ .

4.2. Shape Derivatives of the Cost Functional

We wish to obtain a closed form for the derivatives of the cost functional (5) with respect to a perturbation of the geometry. Using Proposition 1, we can show that the first term of (5),

$$C(D) = \frac{1}{2} \|y - F^D v\|_{\Phi}^2, \quad (16)$$

has the shape derivative

$$\operatorname{Re} \left[\int_{\Gamma} \left(\int_{\Phi} (F^D v(\mathbf{f}) - y(\mathbf{f}))^* e^{-j\mathbf{f}\cdot\mathbf{x}} d\mathbf{f} \right) v(\mathbf{x}) F(\mathbf{x}) d\Gamma \right], \quad (17)$$

where $\mathbf{x} \in \Gamma$ and the superscript $*$ denotes the complex conjugate and $\operatorname{Re}[\cdot]$ denotes the real part of its argument. Similarly, the shape derivative of the second term of (5) is given by

$$\int_{\Gamma} \kappa F d\Gamma. \quad (18)$$

Therefore, the Cauchy-Schwartz inequality tells us that the steepest descent direction of the domain update for the minimization problem (5) is given by a normal boundary speed function $F : \Gamma \rightarrow \mathbb{R}$ (boundary flow) of the form

$$F(\mathbf{x}) = -(E(\mathbf{x}) + \alpha\kappa(\mathbf{x})), \quad \mathbf{x} \in \Gamma \quad (19)$$

$$E = \operatorname{Re}[v^*(F^D)^*(F^D v - y)] \quad (20)$$

where $E : \Omega \rightarrow \mathbb{R}$ vanishes on $\Omega \setminus D$ owing to the indicator function χ_D in (4). Note that (similarly to the curvature κ) $F = F(\mathbf{x})$ is defined only on the boundary Γ .

5. Self-Referencing Level-Set Method

5.1. Shape Evolution by Level-Sets

The problem of minimizing the cost function (5) with respect to the domain D has now been reduced to deforming or evolving the boundary Γ by the boundary flow F . This will provide a steepest descent procedure for the minimization of (5). However, rather

than evolve the boundary directly, we adopt the powerful level-set approach pioneered by Osher and Sethian (1988). They described the evolution of boundary Γ_t in (12) and the domain D_t in terms of the zero-level set of a three-dimensional embedding function $\phi(t, \mathbf{x}) : \mathbb{R}^+ \times \mathbb{R}^2 \rightarrow \mathbb{R}$:

$$\begin{aligned} \Gamma_t &= \{\mathbf{x} \in \mathbb{R}^2 \mid \phi(t, \mathbf{x}) = 0\} \\ D_t &= \{\mathbf{x} \in \mathbb{R}^2 \mid \phi(t, \mathbf{x}) < 0\}. \end{aligned} \quad (21)$$

For a boundary flow F , the evolution of the domain at the zero level set is then given by

$$\frac{\partial \phi}{\partial t} = -F |\nabla \phi|. \quad (22)$$

Equation (22) is a Hamilton-Jacobi-type equation in continuum mechanics (Sethian, 1996). If the boundary flow F is extended over all the domain Ω , topological changes of the domain evolution can be handled easily solving (22) on Ω . Furthermore, the level-set algorithm can be implemented in a fixed Cartesian grid throughout the iterative process.

5.2. The Velocity Extension Problem

If one is able to specify $\phi(t, \cdot)$ for all t , then the problem of computing Γ is solved because Γ is just the zero level set of ϕ . A somewhat hidden aspect of the level-set method (22) is that F should be defined on all Ω , not just on Γ . Thus, not only is the interface embedded in a higher dimensional function, but the normal speed of the interface is itself embedded in a higher dimensional function:

$$\frac{\partial \phi}{\partial t} = -F_e |\nabla \phi|, \quad (23)$$

where $F_e : \Omega \rightarrow \mathbb{R}$ is some normal boundary speed field which, in the zero level-set, equals the given speed $F = \langle V(0), n \rangle$. This new speed field F_e is known as the ‘‘extension velocity,’’ and must be chosen judiciously.

There are various approaches to constructing the extension velocity. Osher and Sethian’s original level-set calculations (Osher and Sethian, 1988) were concerned with interface problems with geometric propagation speeds such as curvature flow, leading to a natural construction of an extension velocity by using the geometry of the given level-set. In these flows, the evolution Eq. (22) converges to a correct boundary for any

choice of the embedding function that satisfies certain smoothness conditions. Based on this, the signed distance function is usually used as a level function owing to its simplicity. For the cases where such a natural velocity extension does not exist, Adalsteinsson and Sethian (1999) proposed a fast extension method of the velocity which preserves the signed distance in narrow band around the zero level curve. For the Mumford-Shah problem (Tsai et al., 2001), let the velocities of the non-zero level sets take on their values from the closest point on the zero level set, while Chan and Vese (2000) solved an additional PDE to extend the velocity.

However, neither the signed distance, nor these choices of velocity extension are effective for our problem, for the following reasons. If the signed distance is used as a level function, the level function must be re-initialized after several iterations to preserve the signed distance. Usually this is done by solving another PDE of the level function (Sethian, 1996). This procedure is time-consuming, but it is essential to stabilize the level-set evolution. However, the more frequent the re-initialization iteration, the more the zero level set may stray from the expected position (Sethian, 1996). The advantage of our approach over such re-initialization is the “switching” behavior of our level-set evolution, which keeps the level function stable avoiding the problematic re-initialization step. Furthermore, unlike the velocity extension methods in Tsai et al. (2001) and Chan and Vese (2000) which require additional computational steps, the extended velocity in our algorithm can be easily deduced from the value of the level-function without additional computation.

We are aware that in the narrow-band level set method (Malladi et al., 1995), the velocity extension method provides a way to update an interface such that the signed distance function is maintained, and the front is never re-initialized. However, in comparison to narrow-band extension methods, our velocity extension is easily obtained from the level function evolution, reducing the computation and simplifying the algorithm. In the following, we explain the details of our new level-set formulation.

5.3. Velocity Extension

Recall the normal velocity field (19). The extension of κ in (19) to the whole domain Ω can be easily derived from its explicit form. If the boundary curve Γ is represented as the zero level set of a level function ϕ , it is known (Sethian, 1996) that the curvature $\kappa(\mathbf{x})$, $\mathbf{x} \in \Gamma$

of the boundary curve Γ can be represented using the level function:

$$\begin{aligned}\kappa &= \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) \\ &= \frac{\phi_{xx}\phi_y^2 - 2\phi_y\phi_x\phi_{xy} + \phi_{yy}\phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}}.\end{aligned}\quad (24)$$

Since ϕ is defined on whole domain Ω , we can therefore simply extend (24) to the whole domain Ω by extending the domain on which κ is defined. We will denote this extension of κ by $\kappa_e : \Omega \rightarrow \mathbb{R}$.

Since the first term in (19) and κ_e are now defined on whole domain Ω , one could try to simply extend the velocity field as follows:

$$\begin{aligned}F_e &= -(\text{Re}[v^*(F^D)^*(F^D v - y)] + \alpha\kappa_e), \\ F_e : \Omega &\rightarrow \mathbb{R}.\end{aligned}\quad (25)$$

However, because of the indicator function χ_D in (4), the first term in (25) vanishes on $\Omega \setminus D$. Therefore, (25) is not a good extension method.

Santosa considered a binary image reconstruction problem (Santosa, 1996) in which the derivation of the velocity extension is also based on the steepest descent direction boundary flow at Γ . In this problem, the image is assumed to take known *constant* values v_{in} and v_{out} on D and $\Omega \setminus D$, respectively. Santosa showed that in this binary problem, the required velocity field $F' : \Gamma \rightarrow \mathbb{R}$ at the boundary is given by

$$F' = -(v_{in} - v_{out})\text{Re}[(F^D)^*(F^D v_{in} - y)] \quad (26)$$

where A^* denotes the adjoint of an operator A . In order to extend the velocity, Santosa replaced the operator $(F^D)^*$ by $(F^\Omega)^* = (F^D)^* + (F^{\Omega \setminus D})^*$ and used the extension velocity $F'_e : \Omega \rightarrow \mathbb{R}$ defined by

$$\begin{aligned}F'_e &= -(v_{in} - v_{out})(\text{Re}[(F^D)^*(F^D v_{in} - y)] \\ &\quad + \text{Re}[(F^{\Omega \setminus D})^*(F^D v_{in} - y)])\end{aligned}\quad (27)$$

where the first and the second term vanishes on $\Omega \setminus D$ and on D , respectively, because of the indicator function in (4). Analogously to (27), we define

$$\begin{aligned}F_e &= -(E_e + \alpha\kappa_e) \\ E_e &= \text{Re}[v^*(F^D)^*(F^D v - y)] \\ &\quad + \text{Re}[w^*(F^{\Omega \setminus D})^*(F^D v - y)],\end{aligned}\quad (28)$$

where $w : \Omega \rightarrow \mathbb{C}$ is a complex-valued function defined on Ω that vanishes on D . Using (28), the level set evolution is now given by

$$\frac{\partial \phi}{\partial t} = E_e |\nabla \phi| + \alpha \kappa_e |\nabla \phi|. \quad (29)$$

The function $w(\cdot)$ can be chosen to balance the magnitude of the velocity E_e on D and on $\Omega \setminus D$. The following section provides one such candidate.

5.4. A Self-Referencing Level-Set Method

In many Fourier imaging problems, the inverse Fourier transform image $\hat{u}_0 : \Omega \rightarrow \mathbb{C}$ defined by

$$\hat{u}_0 = (F^\Omega)^{-1} y = (F^\Omega)^* y, \quad (30)$$

is used as a reconstructed image or at least as an initial guess for image enhancement algorithms. However, in the level set algorithm for binary image reconstruction problems (Santosa, 1996), such information has not been utilized explicitly.

How can we inject the information (30) into the level set algorithm? Recall that we have freedom in the choice of a level-set function. One idea is to use (30) to initialize the level function. Indeed in an inverse scattering problem of permittivity inversion, Dorn et al. (2000) already used the approximate solution from the nonlinear inverse scattering solver as a level function initialization.

Such a formulation of the level function has several advantages over the signed distance function. First, the new level function conveys information about the image intensity. The level function has multiple valleys at the desired locations, which allows fast evolution of the level set. Second, since the term $|\nabla \phi|$ of (23) works as an edge map of the level function, we can expect more rapid evolution of the level function in the edge area of the approximate images.

Based on these observations, we define an image magnitude function $m(t, \mathbf{x})$, and initialize it to the magnitude of the inverse Fourier reconstruction of (30), $m(0, \mathbf{x}) = |\hat{u}_0(\mathbf{x})|$. We then pick the level-set function to be

$$\phi(t, x) = g[m(t, \mathbf{x})], \quad \forall t \geq 0, \quad (31)$$

where $g : \mathbb{R}^+ \rightarrow \mathbb{R}$ is a decreasing, invertible, differentiable function. For example, an adequate choice might

be $g(z) = T_H - z$, where T_H is an appropriate threshold value, which is less than the maximum value of $m(0, \mathbf{x}) = |w(0, \mathbf{x})|$. This choice then initializes the level set to

$$\phi(0, \mathbf{x}) = g[m(0, \mathbf{x})], \quad m(0, \mathbf{x}) = |\hat{u}_0(\mathbf{x})|. \quad (32)$$

Importantly, Eq. (31) can now be used to provide a new evolution equation for $w(t, \mathbf{x})$, which can be used for our velocity extension (28).

Proposition 2. *Suppose the level set $\phi(t, \mathbf{x})$ is defined as in (31), and ϕ is a solution to the evolution Eq. (29) with initial conditions (32). Then the magnitude function $m = g^{-1}(\phi)$ is a solution to the following evolution equation:*

$$\frac{\partial m}{\partial t} = -E_e |\nabla m| + \alpha \kappa_m |\nabla m|, \quad (33)$$

$$m(0, \mathbf{x}) = |\hat{u}_0(\mathbf{x})| \quad (34)$$

where κ_m denotes the curvature of $m(t, \mathbf{x})$.

Proof: See Appendix. \square

Note that the flow (33) and (34) is an anisotropic diffusion of the magnitude image $|\hat{u}_0(\mathbf{x})|$ produced by direct Fourier inversion. Related anisotropic diffusions have been used for noise removal. For example, if $E_e = 0$, then the flow (33) and (34) becomes a curvature flow (Sethian, 1996), where it diffuses m in the direction orthogonal to the gradient ∇m , and does not diffuse in the direction of ∇m . This means that the image is being smoothed on both sides of the edge, with minimal smoothing across the edge itself.² Based on this observation, we propose to use $m(t, \mathbf{x})$ as an estimate of the image magnitude $|v(t, \mathbf{x})|$ on $\Omega \setminus D$. Now, by assuming that the phase of the image on $\Omega \setminus D$ does not change significantly during the evolution, we obtain the following evolution for $w(t, \mathbf{x})$:

$$w(t, \mathbf{x}) = m(t; \mathbf{x}) e^{j \angle \hat{u}_0(\mathbf{x})}, \quad (35)$$

where \angle denotes the argument of a complex quantity. Note that with this definition of $w(t, \mathbf{x})$, we have $w(0, \mathbf{x}) = \hat{u}_0(\mathbf{x})$, so that $w(0, \mathbf{x})$ agrees with (30). If the phase of the image does change significantly during the evolution, (35) will no longer be a good approximation of w . However, numerical experiments suggest that the result is much more sensitive to the magnitude

of $w(t, \mathbf{x})$ than to its phase. Hence, in this paper we use (35) without further elaboration.

Because the level function is initialized as a function of the image estimate (rather than e.g., a signed distance function), we will call this level-set formulation a *self-referencing level set* method.

5.5. Stability of Level Function

Recall that for the choice of the signed distance function as a level function, periodic re-initialization is performed to maintain the signed distance, and hence stabilize the evolution (Sethian, 1996). Likewise, in order to stabilize their evolution, Dorn et al. (2000) periodically solve the heat equation for $\phi(t, \mathbf{x})$, smoothing out irregular behavior of the level function during the evolution. (Such an approach tends to drive the zero level curve away from the expected position.) In contrast, the proposed self-referencing level set method has no explicit formula for stabilizing the level-function evolution. This raises some questions about its stability.

The main advantage of our algorithm is that such re-initialization steps are not necessary. Note that by substituting (6) into (28), the extension velocity is simplified as:

$$\begin{aligned} E_e &= \operatorname{Re}[v^*(F^D)^* P_D^\perp y] + \operatorname{Re}[w^*(F^{\Omega \setminus D})^* P_D^\perp y] \\ &= \operatorname{Re}[w^*(F^{\Omega \setminus D})^* P_D^\perp y], \end{aligned} \quad (36)$$

where P_D^\perp is the orthogonal projection onto the orthogonal complement of the range space of F^D , or, equivalently, onto the null space of $(F^D)^*$, and the second equality follows because $(F^D)^* P_D^\perp = 0$.

Note that E_e in (36) is nonzero only in $\Omega \setminus D$. Therefore, the level-set evolution is given by

$$\frac{\partial \phi}{\partial t}(t, \mathbf{x}) = \begin{cases} \alpha \kappa |\nabla \phi(t, \mathbf{x})|, & \mathbf{x} \in D \\ (E_e + \alpha \kappa) |\nabla \phi(t, \mathbf{x})|, & \mathbf{x} \in \Omega \setminus D. \end{cases} \quad (37)$$

Equation (37) tells us that if $\mathbf{x} \in \Omega \setminus D$, the level-set evolution is driven by the data-driven force E_e and the roughness of the level-set $|\nabla \phi|$. If the estimate of D is incorrect, there will be data mismatch (i.e., $P_D^\perp y \neq 0$), resulting in a nonzero data-driven force E_e and allowing the level-set to keep evolving. However, for $\mathbf{x} \in D$ the curvature flow will stabilize the level-set evolution. Thus, the evolution exhibits a switching behavior between the interior and exterior of the domain D . Whatever the estimated domain D , within D the level function ϕ only undergoes smoothing by non-

isotropic diffusion, and outside D it evolves under data influence.³

This “switching” controls the minimum value of the level function. More specifically, consider the adequate choice $g(z) = T_H - z$ with T_H constant throughout the evolution iteration. If $\phi(t, \mathbf{x}) < 0$, then \mathbf{x} becomes an element of D , and the value of $\phi(t, \mathbf{x})$ is now suppressed to zero by the curvature flow. Similarly, if $\phi(t, \mathbf{x}) \geq 0$, then $\mathbf{x} \in \Omega \setminus D$ hence it is evolving by the data driven flow. Owing to this feedback cycle, we can show that asymptotically the minimum value of the level function is zero, hence

$$\max_{\mathbf{x} \in \Omega} \lim_{t \rightarrow \infty} |w(t, \mathbf{x})| = T_H, \quad (38)$$

which implies that as the evolution continues, the maximum of the $|w(t, \mathbf{x})|$ approaches T_H .

Without such “switching” behavior, the contribution of E_e on D may not be negligible and the evolution of ϕ may not converge even for the correct domain D if the measurement y is noisy. For example, in the velocity extension method (27) for binary image reconstruction problem (Santosa, 1996), if the measurement y is noisy, $F \neq 0$ even for the correct domain D , hence there is no explicit way to stop the evolution.

Another advantage of our algorithm is that the “switching” effect can be easily controlled by the number of CG steps. For examples, if a small number of CG steps are used, $\hat{v} \neq (F^D)^\dagger y$ and the contribution of E_e on D may be significant. This tends to allow rapid evolution of the level function. On the contrary, for large number of CG steps, the contribution of E_e becomes negligible, hence the level function is stabilized. Therefore, the number of CG steps will be used to control the evolution speed and the stability of evolution. In practice, we will use small number of CG iterations in the early phase of evolution in order to allow rapid evolutions, and increase the number of CG steps according to the evolution in order to keep the level set stable.

6. Numerical Implementation and Computational Cost

6.1. Level Set Evolution

Once the velocity is chosen, all coefficients that are involved in the Hamilton-Jacobi equation are defined. Since $\phi(t, \mathbf{x}) = g[m(t, \mathbf{x})]$ and $g: \mathbb{R}^+ \rightarrow \mathbb{R}$ is a deterministic function, the level set evolution can be implemented by the evolution of either $\phi(t, \mathbf{x})$ or $m(t, \mathbf{x})$.

Because $w(t, \mathbf{x})$ is defined by $m(t, \mathbf{x})$ as shown in (35) and used in the evolution, $m(t, \mathbf{x})$ have to be computed at each t . Therefore, the level set is updated indirectly by solving (33) and (34). Now we describe the numerical algorithm for Eq. (33).

We assume that the image domain Ω is rectangular and discretized into $N \times N$ pixels of size Δx and Δy . Define

$$m_{ij}^t = m(t, i\Delta x, j\Delta y). \quad (39)$$

The Hamilton-Jacobi equation is replaced by

$$\frac{1}{\Delta t}(m_{ij}^{t+1} - m_{ij}^t) = -E_{ij}^t H(m_{ij}^t) + \alpha \kappa_{ij}^t H(m_{ij}^t), \quad (40)$$

where f_{ij}^t denotes the value taken by a function $f : \Omega \rightarrow \mathbb{R}$ at the point $(i\Delta x, j\Delta y) \in \Omega$ at time t and where H corresponds to the numerical Hamiltonian, which is in fact an approximation of the gradient $|\nabla m|$, and κ_{ij}^t denotes an approximation of the curvature κ_m .

Three issues need to be addressed in the implementation of (40). The first is how to define the numerical Hamiltonian and curvature. In order to ensure the stability of the numerical evolution, we follow Sethian's proposal for the *upwinding* finite difference scheme (Sethian, 1996) for H , which is determined by the sign of the velocity E_{ij} in (40). The choice ensures that the domain expands when the velocity is positive:

– If $E_{ij} \geq 0$, the numerical Hamiltonian is given by

$$H(m_{ij}^t)^2 = \max(D_-^x m_{ij}^t, 0)^2 + \min(D_+^x m_{ij}^t, 0)^2 + \max(D_-^y m_{ij}^t, 0)^2 + \min(D_+^y m_{ij}^t, 0)^2. \quad (41)$$

– If $E_{ij} < 0$, then

$$H(m_{ij}^t)^2 = \min(D_-^x m_{ij}^t, 0)^2 + \max(D_+^x m_{ij}^t, 0)^2 + \min(D_-^y m_{ij}^t, 0)^2 + \max(D_+^y m_{ij}^t, 0)^2, \quad (42)$$

where $D_+^x m_{ij}^t = (m_{i+1,j}^t - m_{i,j}^t)/\Delta x$ and $D_-^y m_{ij}^t = (m_{i,j}^t - m_{i,j-1}^t)/\Delta y$. However, the numerical Hamiltonian in the last term of (40) is implemented using the following central difference approximation:

$$H(m_{ij}^t)^2 = (D^x m_{ij}^t)^2 + (D^y m_{ij}^t)^2 \quad (43)$$

where

$$D^x m_{ij}^t = \frac{m_{i+1,j}^t - m_{i-1,j}^t}{2\Delta x} \quad (44)$$

$$D^y m_{ij}^t = \frac{m_{i,j+1}^t - m_{i,j-1}^t}{2\Delta y}. \quad (45)$$

Using (24), the numerical approximation of the extended curvature κ_e is implemented as

$$\begin{aligned} \kappa_{ij}^t = & [D^{xx} m_{ij}^t (D^y m_{ij}^t)^2 \\ & - 2(D^x m_{ij}^t)(D^y m_{ij}^t)(D^{xy} m_{ij}^t) \\ & + D^{yy} m_{ij}^t (D^x m_{ij}^t)^2] / H(m_{ij}^t)^{3/2} \end{aligned} \quad (46)$$

where $D^x m_{ij}^t$ and $D^y m_{ij}^t$ are given in (44) and (45) and

$$D^{xx} m_{ij}^t = \frac{m_{i+1,j}^t - 2m_{i,j}^t + m_{i-1,j}^t}{\Delta x^2} \quad (47)$$

$$D^{yy} m_{ij}^t = \frac{m_{i,j+1}^t - 2m_{i,j}^t + m_{i,j-1}^t}{\Delta y^2} \quad (48)$$

$$\begin{aligned} D^{xy} m_{ij}^t \\ = & \frac{m_{i+1,j+1}^t + m_{i-1,j-1}^t - m_{i+1,j-1}^t - m_{i-1,j+1}^t}{4\Delta x \Delta y}. \end{aligned} \quad (49)$$

The second issue concerns the boundary conditions to impose on the image domain Ω . Since the boundary of the image domain should remain fixed, the level-set function satisfies a Neumann boundary condition:

$$\langle \nabla m(t, \mathbf{x}), \mathbf{n}(\mathbf{x}) \rangle = 0, \quad \mathbf{x} \in \partial\Omega. \quad (50)$$

Following the implementation in Sethian (1996), we have chosen a reflecting boundary condition to approximate (50). More specifically, we create an extra layer of ghost cells around the image domain Ω with values that are simply direct copies of the values of the level set function m along the boundary $\partial\Omega$ of Ω . For a rectangular image domain Ω , this is equivalent to setting $D^x m_{ij}^t = 0$ on the vertical boundaries and $D^y m_{ij}^t = 0$ on the horizontal boundaries, hence satisfying (to within the finite differencing approximation) the condition (50). Furthermore, we set the value of κ_m at the image domain boundary $\partial\Omega$ to zero so that the curve-minimizing flow will not shrink the image domain.

The third implementation issue concerns the time step Δt . There are inherent time step requirements in

the above discrete implementation. We have a Courant-Friedrich-Lewy condition (Sethian, 1996) which requires the front to cross no more than one grid cell each time step. Thus, we require that

$$\max_{\Omega} |F_e| \Delta t \leq \min\{\Delta x, \Delta y\} \quad (51)$$

where the maximum is taken over values for F_e for all possible grid points. We use the following time step size:

$$\Delta t = \frac{0.99 \min\{\Delta x, \Delta y\}}{\max_{i,j} |-E_{ij}^t + \alpha \kappa_{ij}^t|}. \quad (52)$$

In practice, the numerical implementation of κ_{ij}^t in (46) is *numerically* unstable especially when the Hamiltonian $H(m_{ij}^t)$ approaches to zero,⁴ so Δt of (52) can become very small, slowing the progress of the iterations. In order to avoid this situation, we compute Δt in (52) assuming $\alpha = 0$.

6.2. CG Iteration and Velocity Extension

We use the Polak-Ribiere version of the conjugate gradient algorithm (Luenberger, 1989).

Initialize:

$$q = (F^D)^* y \quad (53)$$

$$\hat{v}^0 = 0 \quad (54)$$

$$d^0 = q. \quad (55)$$

For $k = 0, 1, \dots$,

$$\hat{v}^{k+1} = \hat{v}^k + \alpha_k d^k \quad (56)$$

$$r^{k+1} = (F^D)^* F^D \hat{v}^{k+1} - q \quad (57)$$

$$d^{k+1} = -r^{k+1} + \beta_k d^k \quad (58)$$

where

$$\beta_k = \frac{\langle r^{k+1} - r^k, r^{k+1} \rangle}{\langle r^k, r^k \rangle} \quad (59)$$

$$\alpha_k = -\frac{\langle r_k, d_k \rangle}{d_k^* (F^D)^* F^D d_k} \quad (60)$$

For a regular grid, the computation of q in (53) can be implemented using the fast Fourier transform (FFT) after zero padding y on $\Omega \setminus \Phi$. In (57), the computation of $(F^D)^* F^D \hat{v}^{k+1}$ can be implemented by computing the FFT of \hat{v}^{k+1} , zero-padding on $\Omega \setminus \Phi$, and finally

performing the inverse FFT (IFFT). Computation of α_k can be done similarly using one FFT of d_k . In total, for $k \geq 1$ one iteration of CG algorithm requires two FFT and one IFFT operations on a $N \times N$ image.

To compute the extension velocity E_e of (28), we need to compute

$$v^* (F^D)^* (y - F^D v), \quad (61)$$

which can be again implemented using one FFT and additional IFFT.

The computational cost of the algorithm is dominated by the CG steps for the pixel-value update and the computation of the extension velocity. Therefore, overall complexity of the algorithm per evolution iteration is given by $O(N^2 \log N)$ for a $N \times N$ images.

7. Numerical Results

We provide four examples to evaluate the performance of our algorithm. In all cases, the regularization parameter α in (29) is hand-picked for best performance. To describe the experiments, we introduce the following definitions. For an $N \times N$ -pixel image we measure region sizes, $|D| |\Phi|$ and $|\Omega| = N^2$ by the number of pixels they contain. The fraction $\eta = |D|/N^2$ of the image occupied by the supports of the objects is called the *occupancy rate*. Likewise, because the total number of Fourier samples is $|\Phi|$, we call $\rho = |\Phi|/N^2$ the *sampling rate*. To obtain reliable reconstruction we need $\rho > \eta$ (often by some factor) (Venkataramani and Bresler, 1998; Bresler et al., 1999; Gastpar and Bresler, 2000b).

We use the function $g(z) = T_H - z$ for all our simulations. Parameter T_H is a threshold value, which is set separately for each problem using a rough a priori estimate $\bar{\eta}$ of the true occupancy rate η . Note that T_H determines the size of the initial domain estimate $\hat{\eta}_0$ for the iteration. If T_H is too small, then the initial occupancy rate $\eta_0 = |\hat{D}_0|/N^2$ can exceed the sampling rate, $\rho < \eta_0$, and the non-uniform sampling theorem (Venkataramani and Bresler, 1998; Bresler et al., 1999) tells that unique reconstruction is not possible. On the contrary, if T_H is too big, then \hat{D}_0 is too small, and the level set algorithm needs many iterations to evolve \hat{D} to its correct size. It would be useful to develop an analytical or numerical method to choose an optimal value for T_H . However, in this paper we have chosen T_H empirically by repeating the

calculation with several different values, and selecting the one that yields $\eta_0 \approx 0.3\bar{\eta}$, i.e., an initial domain estimate with total area equal to about 30% of the assumed total area of the actual object domain D .

Example 1. The original piecewise constant 128×128 image of Fig. 1(a) consists of two rectangles and a circle with different pixel values (2.353, 1.568, and 0.784). The occupancy rate is $\eta = 25\%$. In this

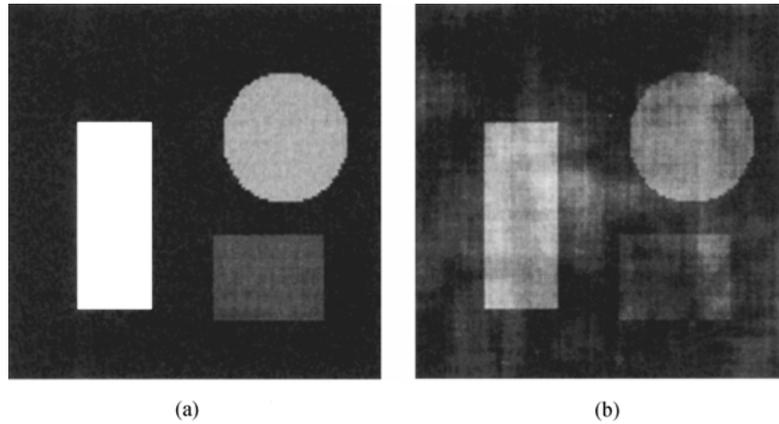


Figure 1. (a) Magnitude image of the original ground truth, and (b) Fourier inversion result with zero-padding.

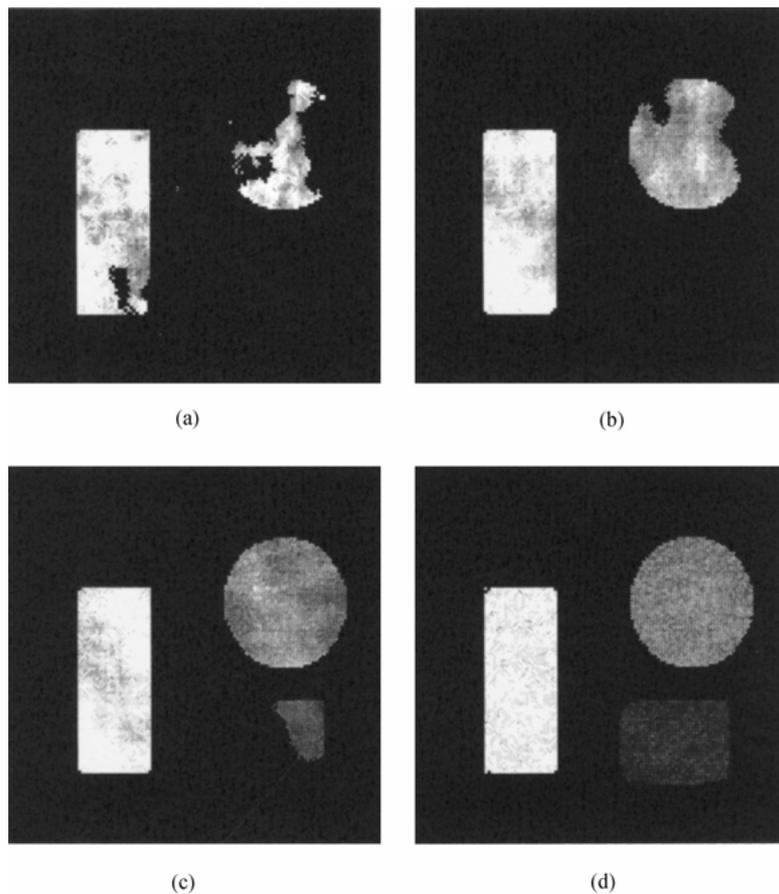


Figure 2. Reconstruction results after (a) 1, (b) 10, (c) 20, and (d) 200 evolution steps.

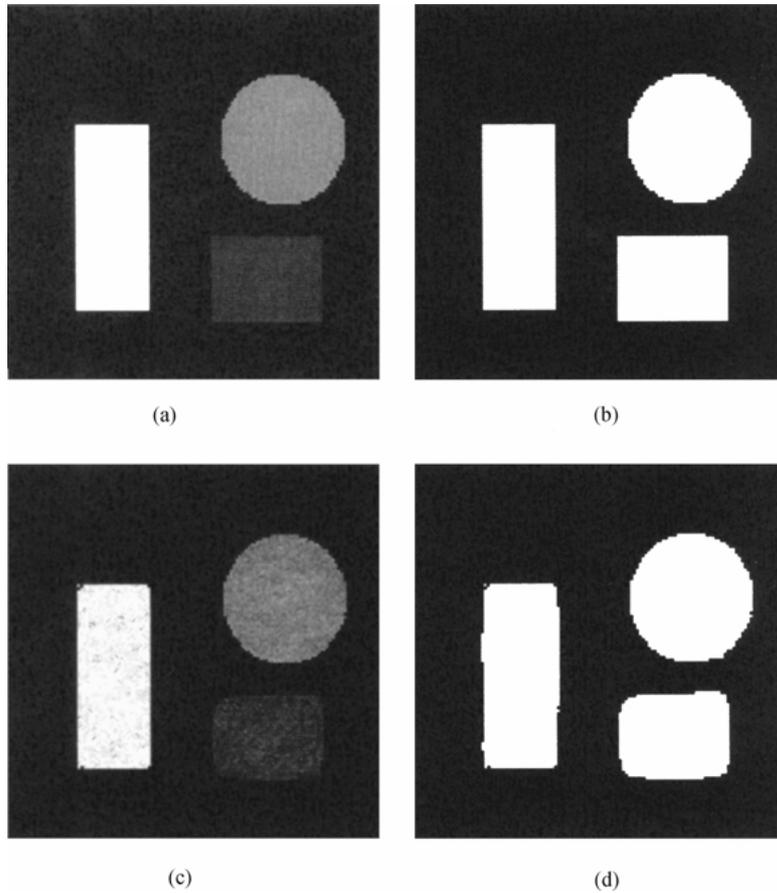


Figure 3. (a) Magnitude image of the ground truth, (b) true domain, (c) magnitude of the estimated image, and (d) estimated domain.

experiment, we reconstruct the pixel values and the domain from randomly chosen samples in the 2-D Fourier domain, with sampling rate $\rho = 50\% = 2\eta$. Additive complex Gaussian noise is added to the samples at an rms level of 5% of the signal energy. The direct Fourier inversion result $\hat{u}_0(\mathbf{x})$ after zero-padding (Fig. 1(b)) is quite blurred.

In order to control the evolution speed and stability, we use the following empirical rule for the number of the CG steps in the subsequent simulations:

$$N_{CG} = 5 + \left\lfloor \frac{N_{EV}}{20} \right\rfloor \quad (62)$$

where N_{CG} and N_{EV} denote the number of CG steps and the evolution number, respectively, and $\lfloor x \rfloor$ denotes the largest integer less than or equal to x . Asymptotically, the choice in (62) guarantees that the level set becomes stable the number of N_{CG} increases and the solution of the CG-iteration becomes more accurate.

The magnitude of the pixel values and the domain estimates are compared to ground truth in Fig. 3, with intermediate reconstruction results given in Fig. 2. Accurate pixel value estimates and nearly perfect domain estimate are obtained. The resulting level-function (Fig. 4(a), $T_H - \phi(t, \mathbf{x})$ shown) appears to have a well-behaved form, free of spurious excursions. Its value is clipped at $\phi = T_H$, as predicted by the discussion in Section 5.5. The rms plot (Fig. 4(b)) suggests nearly monotone convergence of the algorithm. Note, however, the unique oscillatory behavior of the convergence plot. This is because a small perturbation of the domain can allow large rms value changes. Here, the threshold T_H and the regularization parameter α are set to 60% of the maximum value of $m(0, \mathbf{x})$ and 0.3, respectively.

Example 2. The test image is generated by cutting the Lena image as shown in Fig. 5. The pixel values are normalized to have maximum value of 2, and T_H

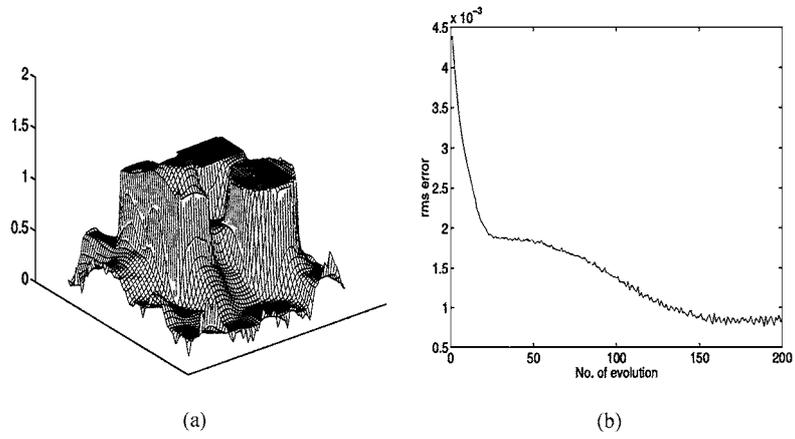


Figure 4. Example 1: (a) level function and (b) rms error plot.

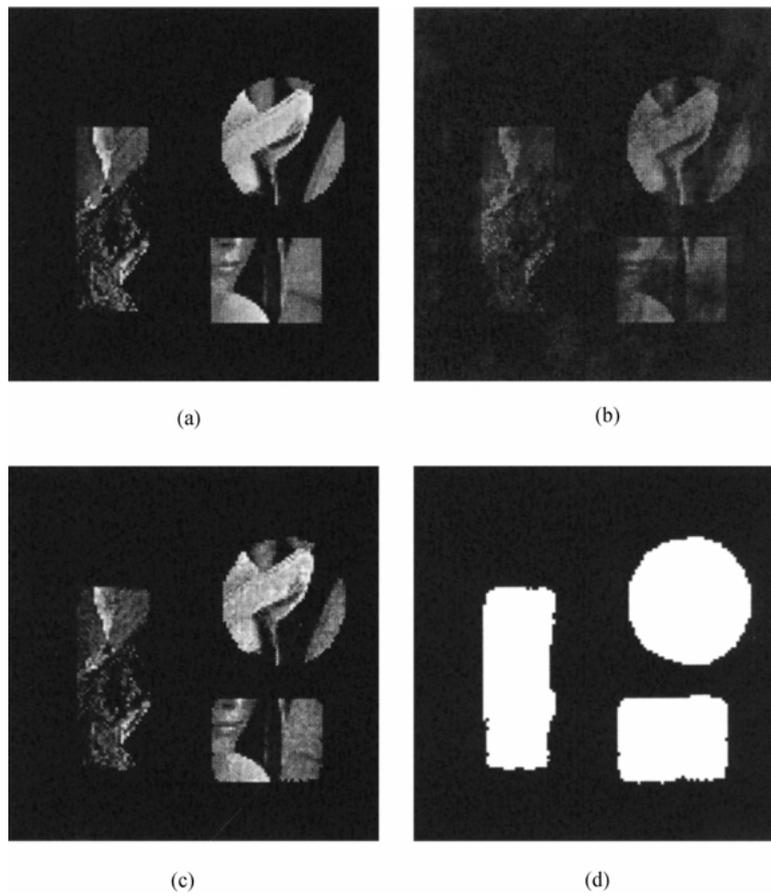


Figure 5. Example 2: (a) partial Lena image, (b) direct Fourier inversion, (c) reconstruction by our algorithm, and (d) estimated domain.

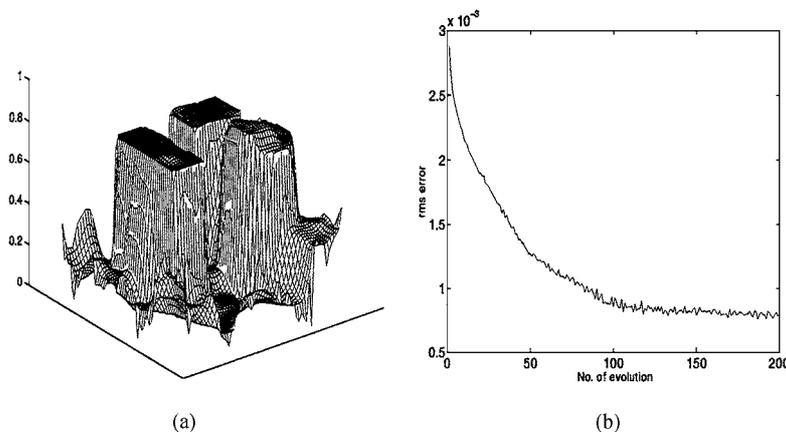


Figure 6. Example 2: (a) level function and (b) rms error plot.

and α are set to 60% of the maximum value of $m(0, \mathbf{x})$ and 0.1, respectively. The domain D , occupancy and sampling rates, and the noise level are all the same as in Example 1. Again, while the direct Fourier inversion (Fig. 5(b)) is blurry, the reconstruction by our algorithm after 200 iteration (Fig. 5(c)) is accurate. The resultant level function and the rms error plot (Fig. 6) are similar to those in Example 1.

Next, we reduce the sampling rate down to $\rho = 37.5\% = 1.5\eta$. While the direct Fourier inversion (Fig. 7(a)) is noticeably degraded, our reconstruction (Fig. 7(b)) is still reasonable.

Example 3. This example is drawn from a passive radar imaging problem (Lanterman, 1999; Ye et al., 2001b), where the sampling pattern in Fourier space is determined by the locations of the radar and commer-

cial TV/FM stations, and the trajectory of the airborne targets. The pixel values are normalized to have the maximum magnitude of 2, and additive complex Gaussian noise is added to the samples at an rms level of 5%. The occupancy rate is now $\eta = 37\%$, but the sampling rate is only $\rho = 0.33\eta$. This is a severely undersampled case, which according to theory (Gastpar and Bresler, 2000a, 2000b) does not suffice for reconstruction of even one third of the object support. A true VFY aircraft reflectance image and the direct Fourier inversion result are given in Figs. 8(a) and (b). The threshold T_H and the regularization parameter α are set to 30% of the maximum value of $m(0, \mathbf{x})$ and 0.001, respectively. The reconstruction by our algorithm after 100 iterations (Fig. 8(c)) clearly shows the structure of the VFY aircraft. The rms error plot in Fig. 8(d) again reveals fairly rapid convergence.

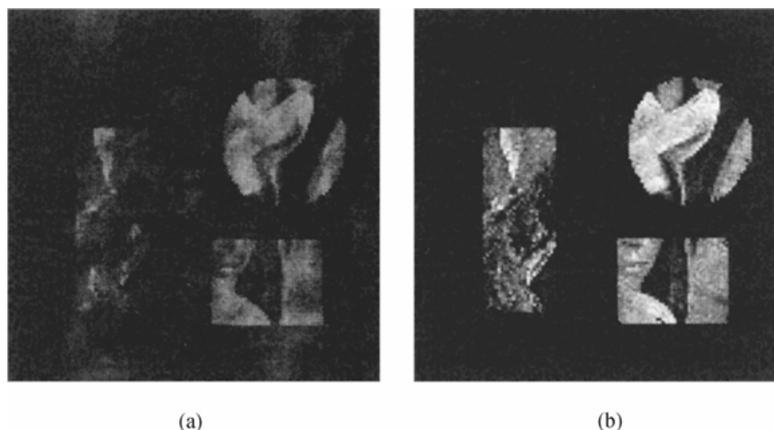


Figure 7. Example 2: (a) direct Fourier inversion and (b) reconstruction by our algorithm when the sampling rate is reduced to 150% of the pixel occupancy.

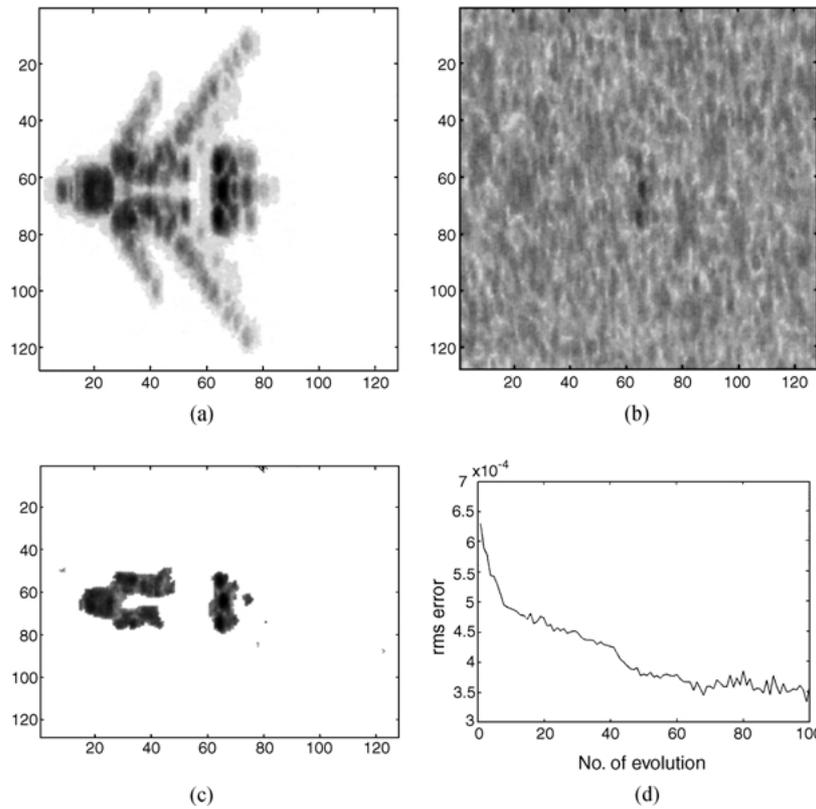


Figure 8. Example 3: (a) original VFY aircraft image, (b) direct Fourier reconstruction, (c) reconstruction by our algorithm, and (d) rms error plot.

Example 4. This example is drawn from a magnetic resonance imaging problem. The goal is “change detection” in a complex background, which is however known, from previous reference scans. Consider the original “Shepp-Logan” phantom given in Fig. 9(a). The maximum value of the phantom is normalized to

the value of 5. Suppose we add three “tumors” as shown in Fig. 9(b), which have *unknown* additional absorption values of 0.5, 0.6, and 1.0. The supports of the three tumors are also unknown. Assuming that we know the original healthy phantom of Fig. 9(a), our goal is now to estimate the image in Fig. 9(b) from its noisy Fourier

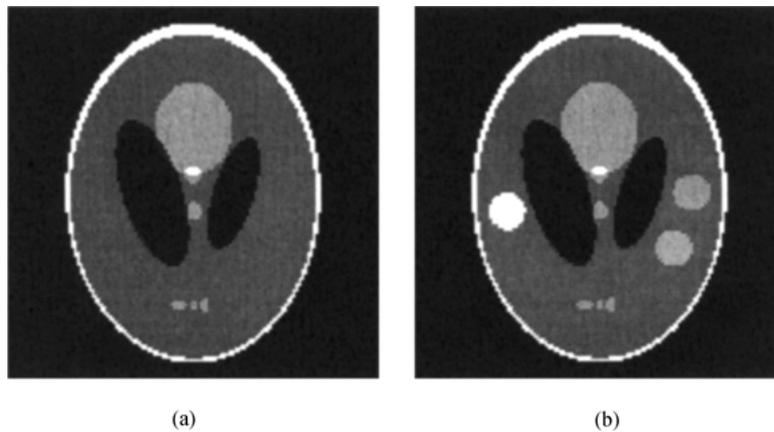


Figure 9. Example 4: (a) original Shepp-Logan phantom, (b) modified phantom with three additional tumors.

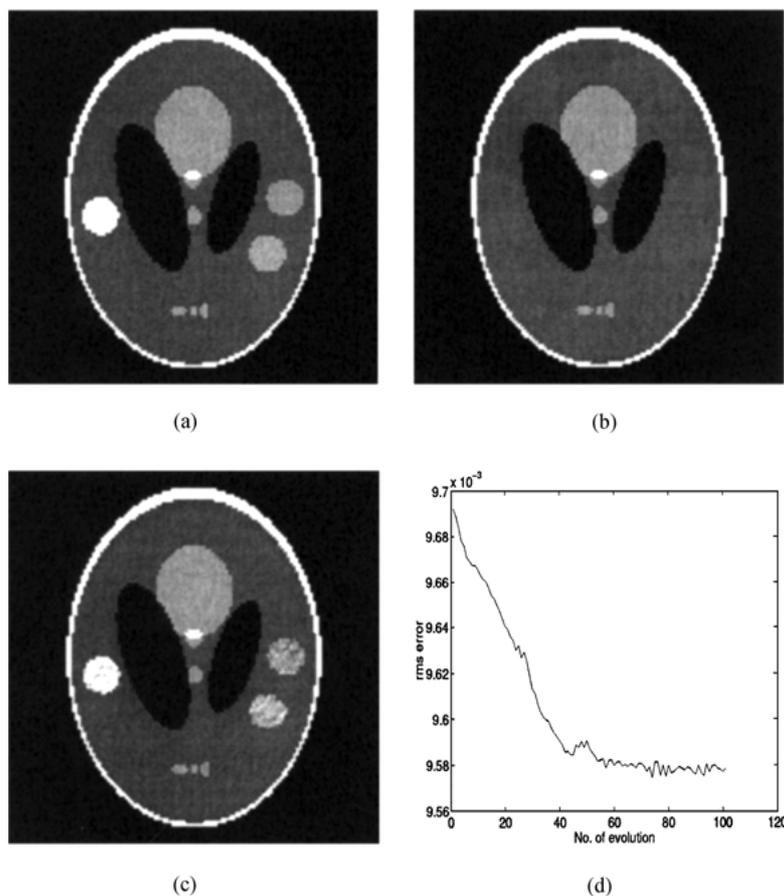


Figure 10. Example 4: (a) original phantom image with tumors, (b) direct Fourier reconstruction, (c) reconstruction by our algorithm, and (d) rms error plot.

samples. Again, additive complex Gaussian noise is added to the samples at an rms level of 5%. The occupancy rate of the difference image—which corresponds to three tumors—is now $\eta = 2.3\%$, and the sampling rate is only $\rho = 2\eta = 4.6\%$. A Fourier reconstruction in Fig. 10(b) is very poor and we cannot locate any tumors, because of the poor point spread function of the system. However, the reconstruction after 100 iterations using the new method (Fig. 10(c)) is still good. The rms error plot in Fig. 10(d) again reveals fairly rapid convergence. Here, the threshold T_H and the regularization parameter α are set to 70% of the maximum value of $m(0, \mathbf{x})$ and 0.0005, respectively.

8. Conclusion

We have introduced a new algorithm for image estimation from sparse Fourier samples. The problem was formulated as joint estimation of the supports of un-

known sparse objects in the image, and pixel values on these support. The domain and the pixel values are alternately estimated using the new self-referencing level-set method and the conjugate gradient method, respectively.

While the conventional level set formulation using the signed distance could be used to solve the problem, our self-referencing level-set method has many unique properties, particularly attractive for the sparse Fourier imaging problem. First, our level function was initialized as a function of the magnitude of the direct Fourier-inverted image, hence the new level function has multiple valleys at the desired locations, which allows fast evolution of the level set. This choice also provides an evolution equation for the magnitude image allowing simple velocity extension without additional computations. Furthermore, the number of conjugate gradient steps controls the speed and the stability of evolution, which totally removes the re-initialization

step in conventional level set methods. Simulation results suggest that our algorithm provides accurate image and domain estimates from sparse Fourier samples (close to the theoretical minimum) and is quite robust to noise.

9. Appendix

Note that $\phi(t, \mathbf{x}) = g(m(t, \mathbf{x}))$, where $g: \mathbb{R}^+ \rightarrow \mathbb{R}$ is a decreasing invertible continuously differentiable function. Using the chain rule, the various terms in the evolution Eq. (29) can be written as:

$$\frac{\partial \phi(t, \mathbf{x})}{\partial t} = g'(m(t, \mathbf{x})) \frac{\partial m(t, \mathbf{x})}{\partial t} \quad (63)$$

$$|\nabla \phi(t, \mathbf{x})| = |\nabla m(t, \mathbf{x})| g'(m(t, \mathbf{x})) \quad (64)$$

$$\begin{aligned} \kappa &= \nabla \cdot \left(\frac{\nabla \phi(t, \mathbf{x})}{|\nabla \phi(t, \mathbf{x})|} \right) \\ &= \nabla \cdot \left(\frac{g'(m(t, \mathbf{x})) \nabla m(t, \mathbf{x})}{|g'(m(t, \mathbf{x}))| |\nabla m(t, \mathbf{x})|} \right) \\ &= \text{sgn}(g'(m(t, \mathbf{x}))) \nabla \cdot \left(\frac{\nabla m(t, \mathbf{x})}{|\nabla m(t, \mathbf{x})|} \right) \\ &= \text{sgn}(g'(m(t, \mathbf{x}))) \kappa_m, \end{aligned} \quad (65)$$

where κ and κ_m are the curvatures of the functions $\phi(t, \mathbf{x})$ and $m(t, \mathbf{x})$, respectively. Hence,

$$g' \frac{\partial m}{\partial t} = E_e |\nabla m| |g'| + \alpha g' \kappa_m |\nabla m|. \quad (66)$$

Because $g'(m(t, \mathbf{x})) < 0$ by assumption, dividing both sides of (66) by g' yields (33). This concludes the proof.

Notes

1. Proposition 1 has been used in other image estimation problems, such as the computation of fundamental performance limits in parametric shape estimation for general inverse problems (Ye et al., 2000, 2001a).
2. This can be shown by noting that $\kappa_m |\nabla m| = \nabla \cdot \left(\frac{\nabla m}{|\nabla m|} \right) |\nabla m| = m_{\eta\eta}$ where η is the direction normal to ∇m .
3. As revealed by the derivation leading to (37), this feature is obtained owing to the problem formulation and the choice of steepest descent—and has little to do with the particular velocity extension. The latter only affects the evolution on $\Omega \setminus D$.
4. However, $\kappa_{ij}^i H(m_{ij}^i)$ is well defined.

References

Adalsteinsson, D. and Sethian, J.A. 1999. The fast construction of extension velocities in level set methods. *J. Computational Physics* 148(1):2–22.

- Ambrosio, L. and Tortorelli, V. 1990. Approximation of functionals depending on jumps by elliptic functionals via Γ -convergence. *Comm. Pure Appl. Math.* (8):707–711.
- Blahut, R.E. 2001. *Theory of Remote Image Formation*. Cambridge University Press: Cambridge, UK. (preprint).
- Bresler, Y. and Feng, P. 1996. Spectrum-blind minimum-rate sampling and reconstruction of 2-D multiband signals. In *Proc. 3rd IEEE Int. Conf. on Image Processing, ICIP'96*, vol. I., Lausanne, Switzerland, pp. 701–704.
- Bresler, Y., Gastpar, M., and Venkataramani, R. 1999. Image Compression On-the-fly by universal sampling in Fourier imaging systems. In *Proc. 1999 IEEE Information Theory Workshop on Detection, Estimation, Classification, and Imaging*. Santa-Fe, NM, pp. 48–48.
- Chakraborty, A., Staib, L., and Duncan, J. 1996. Deformable boundary finding in medical images by integrating gradient and region information. *IEEE Trans. on Medical Imaging*, 15:859–870.
- Chan, T. and Vese, L. 1999. An active contour model without edges. In *Scale-Space Theories in Computer Vision*. Corfu, Greece.
- Chan, T. F. and Vese, L.A. 2000. A level set algorithm for minimizing the Mumford-Shah functional in image processing. Technical Report, University of California, Los Angeles.
- Charbonnier, P., Blanc-Férand, L., Aubert, G., and Barland, M. 1997. Deterministic edge-preserving regularization in computed imaging. *IEEE Trans. on Image Processing*, 6(2):298–311.
- Delaney, A.H. and Bresler, Y. 1998. Globally convergent edge-preserving regularized reconstruction: An application to limited-angle tomography. *IEEE Trans. on Image Processing* 7(2):204–221.
- Dorn, O., Miller, E., and Rappaport, C. 2000. A shape reconstruction method for electromagnetic tomography using adjoint fields and level sets. *Inverse Problems* 16(5):1119–1156.
- Gastpar, M. and Bresler, Y. 2000a. On the necessary density for spectrum-blind nonuniform sampling subject to quantization. In *Proc. of IEEE Int'l Conf. on Acoust., Speech and Sig. Proc. Istanbul*.
- Gastpar, M. and Bresler, Y. 2000b. On the optimum density for nonuniform sampling. In *Proc. ISIT, International Symposium on Information Theory*. Sorrento, Italy.
- Geman, D. and Yang, C. 1995. Nonlinear image recovery with half-quadratic regularization. *IEEE Trans. on Image Processing*, 4(7):932–946.
- Jchan-Besson, S., Barland, M., and Aubert, G. 2001. Video object segmentation using Eulerian region-based active contours. In *Int. Conf. on Computer Vision*. Vancouver.
- Lanterman, A.D. 1999. Tracking and recognition of airborne targets via commercial television and FM radio signals. In *Acquisition, Tracking and Pointing XIII*, vol. 3692. Orlando, FL.
- Leclerc, Y.G. 1989. Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision* 3(1):73–102.
- Luenberger, D.G. 1989. *Linear and Nonlinear Programming*, 2nd edn. Addison-Wesley: Reading, MA.
- Malladi, R., Sethian, J., and Vemuri, B. 1995. Shape modeling with front propagation: A level set approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 17(2):158–175.
- Mumford, D. and Shah, J. 1989. Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.* (42):577–684.

- Osher, S.J. and Sethian, J.A. 1988. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulation. *J. of Comp. Phys.*, 79:12–49.
- Paragios, N. and Deriche, R. 1999. Geodesic active regions for motion estimation and tracking. In *Int. Conf. on. Computer Vision*. Corfu, Greece.
- Richardson, T.J. 1989. Scale independent piecewise smooth segmentation of images via variational methods. Ph.D. Dissertation, Mass. Inst. Technol.
- Santosa, F. 1996. A level-set approach for inverse problems involving obstacles. *ESAIM: Control, Optimisation and Calculus of Variations*, 1:17–33.
- Sethian, J.A. 1996. *Level Set Methods and Fast Marching Methods*. Cambridge University Press: United Kingdom.
- Sokolowski, J. and Zolesio, J. 1991. *Introduction to Shape Optimization: Shape Sensitivity Analysis*. Springer-Verlag: New York.
- ter Haar Romeny, B.M. 1994. *Geometry-Driven Diffusion in Computer Vision*. Doordrecht, Kluwer Academic Publishers: The Netherlands.
- Tsai, A., Yezzi, A., and Willsky, A. 2001. Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification. *IEEE Trans. on Image Processing* 10(8):1169–1186.
- Venkataramani, R. and Bresler, Y. 1998. Further results on spectrum blind sampling of 2D signals. In *Proc. IEEE Int. Conf. Image Proc., ICIP*, vol. 2. Chicago, pp. 752–756.
- Ye, J.C., Bresler, Y., and Moulin, P. 2000. Asymptotic global confidence regions in parametric shape estimation problems. *IEEE Trans. on Information Theory* 46(5):1881–1895.
- Ye, J.C., Bresler, Y., and Moulin, P. 2001a. Cramér-Rao bound of shape estimation in inverse problems. *IEEE Trans. on Image Processing*, to appear.
- Ye, J.C., Bresler, Y., and Moulin, P. 2001b. Cramér-Rao bounds for 2-D target shape estimation in nonlinear inverse scattering problems with application to passive radar. *IEEE Trans. on Antennas and Propagat.* 89(5):771–783.
- Yu, D.F. and Fessler, J. 2001. Edge-preserving tomographic reconstruction with nonlocal regularization. *IEEE Trans. on Image Processing*, submitted.
- Zhu, S.C. and Yuille, A. 1996. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 18(9):884–900.