Simulation of a Hybrid Model for Image Denoising

Ricolindo L. Cariño[1], Ioana Banicescu[2,1], Hyeona Lim[3], Neil Williams[3] and Seongjai Kim[4]
[1]Center for Computational Sciences – ERC, Mississippi State University
Mississippi State, MS 39762-9627
[2]Department of Computer Science and Engineering, Mississippi State University
Mississippi State, MS 39762-9637
[3]Department of Mathematics and Statistics, Mississippi State University
Mississippi State, MS 39762-5921
[4]Department of Mathematics, University of Kentucky
Lexington, Kentucky 40506-0027
(rlc@erc,ioana@cse,hlim@math).msstate.edu, tnw7@msstate.edu, skim@ms.uky.edu

Abstract

We propose a new model for image denoising which is a hybrid of the total variation model and the Laplacian mean-curvature model. An efficient numerical procedure to compute the hybrid model is also presented. The hybrid model and its computational procedure introduce a number of parameters. As a preliminary step to the synthesis of a method for selecting optimal parameters, the hybrid model was simulated on a number of known images with synthetically added noise. The simulation was implemented as a parallel application on a general-purpose Linux cluster. The estimated parallel efficiency of the simulation is in excess of 96%.

1 Introduction

Denoising is an important image processing (IP) step for various image-related applications and often necessary as a pre-processing for other imaging tasks such as segmentation and compression. Thus image denoising methods have occupied a peculiar position in IP, computer graphics, and their applications [12, 20, 21, 22, 27]. Recently, as the field of IP requires higher levels of reliability and efficiency, various powerful tools of partial differential equations (PDEs) and functional analysis have been successfully applied to image restoration [1, 4, 8, 14, 19, 23, 25, 26, 32] and color processing [2, 9, 15, 17, 28]. In particular, a considerable amount of research has been carried out for the theoretical and computational understanding of the total variation (TV) model [26] and its variants [1, 4, 5, 8, 14, 16, 19, 20, 29].

However, most of those denoising models may lose fine structures of the image due to a certain nonphysical dissipation. As remedies, the employment of the Besov norm [20] and iterative refinement [24] have been studied. But these new methods are either difficult to minimize utilizing the Euler-Lagrange equation approach or have the tendency to keep an observable amount of noise. Recently, in order to overcome the drawbacks, one of the authors suggested the method of non-flat time evolution (MONTE) [7] and the equalized net diffusion (END) approach [6]. The MONTE and END techniques are applicable to various (conventional) denoising models as either a time-stepping procedure or a variant of mathematical modeling.

As another remedy to the nonphysical dissipation, fourth-order PDE models have been emerged [13, 18, 31]. In particular, the Laplacian mean-curvature (LMC) model has been paid a particular attention due to its potential capability to preserve edges of linear curvatures. However, it has been numerically verified [30] that the LMC model can easily introduce granule-shaped spots to restored images. To overcome the granularity, this article proposes a hybrid model which combines a TV-based model and the LMC.

In this paper, we investigate the properties of a proposed hybrid TV–LMC model for image denoising. The hybrid model introduces a number of parameters, highlighting the need for a procedure for selecting optimal parameters. Preliminary to the development of such a selection procedure, we have undertaken a parametric study of the hybrid model in order to discover the solitary and interactive effects of the parameters on model accuracy. Such a parametric study is necessarily time-consuming due to the huge number of combinations of the parameter values to be tested. In addition, the study has to be performed on a number of different images, thereby increasing the overall investigation time. Thus, the parametric study was implemented as a parallel computing application. This paper focuses on the performance of the parallel implementation on a general-purpose Linux cluster.

The rest of this paper is organized as follows. It discusses the hybrid model for image denoising in Section 2, and then describes the parallel implementation of the parametric study of the model in Section 3. Section 4 presents sample results of performance tests of the parallel implementation. Section 5 gives a summary and describes future work.

2 A Hybrid Model for Image Denoising

We begin with the Laplacian mean-curvature (LMC) model:

$$\frac{\partial u}{\partial t} + \Delta\kappa(u) = \beta(f-u), \tag{1}$$

where $\beta \geq 0$, a constraint coefficient, and $\kappa(u)$ denotes the mean-curvature defined as

$$\kappa(u) = \nabla \cdot \Big(\frac{\nabla u}{|\nabla u|}\Big).$$

In equation (1), f is a contaminated image and the solution u represents a denoised image. The LMC model have a major drawback: granularity. The restored image can easily incorporate granule-shaped spots. The LMC model also shows staircasing, a phenomenon that tends to make the restored image locally constant. However, it is relatively easy to cure.

2.1 The model

As a remedy for the granule-shaped spots introduced by the LMC model, consider the following hybrid model

$$\frac{\partial u}{\partial t} - \sigma\,\widetilde{\kappa}(u) + \Delta\widetilde{\kappa}(u) = \beta(f-u), \tag{2}$$

where $\sigma \geq 0$ is a regularization parameter and

$$\widetilde{\kappa}(u) = |\nabla u|\,\kappa(u) = |\nabla u|\,\nabla \cdot \Big(\frac{\nabla u}{|\nabla u|}\Big).$$

Here the gradient magnitude $|\nabla u|$ has been incorporated into $\widetilde{\kappa}(u)$, as a scaling factor, in order to reduce staircasing [19]. The second-order term is introduced for (2) to hold a certain degree of maximum principle, with which the model in turn can eliminate the granularity [30]. In this article, we will call (2) the *generalized LMC* (GLMC) model. In the following subsection, we present an efficient numerical procedure for the GLMC model.

2.2 A numerical procedure

Let Δt be a timestep size and $t^n = n\Delta t$. Set $u^n = u(\cdot, t^n)$, $n = 0, 1, \cdots$, with $u^0 = f$. Let $(D_{x_1}, D_{x_2})^T$ and $(D_{2x_1}, D_{2x_2})^T$ be the half-step (regular) central difference and wider central difference operators for the gradient ∇, respectively. Assume that the iterates have been computed up to the $(n-1)$-th time level. For the computation of the solution in the nth time level, define matrices: for $\ell = 1, 2$ and $m = n-1, n$,

$$\begin{aligned}
\mathcal{K}_\ell u^m &= -|\nabla_E u^{n-1}| D_{x_\ell}\left(\frac{D_{x_\ell} u^m}{|\nabla_h u^{n-1}|}\right), \\
\mathcal{K}_\ell^2 u^m &= -|\nabla_E u^{n-1}| D_{2x_\ell}\left(\frac{D_{2x_\ell} u^m}{|\nabla_h u^{n-1}|}\right), \\
\mathcal{K}_\ell^\alpha u^m &= (1-\alpha)\mathcal{K}_\ell\, u^m + \alpha \mathcal{K}_\ell^2\, u^m, \quad \alpha \in [0,1], \\
\mathcal{L}_\ell u^m &= -D_{x_\ell} D_{x_\ell} u^m,
\end{aligned} \tag{3}$$

where $|\nabla_E u^{n-1}|$ and $|\nabla_h u^{n-1}|$ denote appropriate finite difference approximations of $|\nabla u^{n-1}|$. (The above matrices depend on u^{n-1}; however, we did not put the dependence on the matrices for a simpler presentation.) Let

$$\mathcal{K} = \mathcal{K}_1 + \mathcal{K}_2, \quad \mathcal{K}^\alpha = \mathcal{K}_1^\alpha + \mathcal{K}_2^\alpha, \quad \mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2.$$

and

$$\mathcal{D} = \beta + \sigma \mathcal{K}^\alpha + \mathcal{L}\,\mathcal{K}. \tag{4}$$

Then, a linearized Crank-Nicolson difference equation for (2) reads

$$\frac{u^n - u^{n-1}}{\Delta t} + \mathcal{D}\,\frac{u^n + u^{n-1}}{2} = \beta f. \tag{5}$$

Now, let

$$\mathcal{A}_\ell = \frac{\beta}{2} + \sigma \mathcal{K}_\ell^\alpha + \mathcal{L}_\ell\, \mathcal{K}_\ell, \quad \ell = 1, 2.$$

Since

$$\mathcal{D} = (\mathcal{A}_1 + \mathcal{A}_2) + (\mathcal{L}_1\mathcal{K}_2 + \mathcal{L}_2\mathcal{K}_1),$$

(5) can be rewritten as

$$\begin{aligned} &\frac{u^n - u^{n-1}}{\Delta t} + (\mathcal{A}_1 + \mathcal{A}_2)\frac{u^n + u^{n-1}}{2} \\ &\qquad = \beta f - \frac{1}{2}(\mathcal{L}_1\mathcal{K}_2 + \mathcal{L}_2\mathcal{K}_1)(u^n + u^{n-1}). \end{aligned} \tag{6}$$

Thus, replacing the last term in (6) by known values with the error not larger than the truncation error, an alternating direction implicit (ADI) method for (5) can be formulated [10, 11] as

$$\begin{aligned} \left(1 + \frac{\Delta t}{2}\mathcal{A}_1\right) u^* &= \left(1 - \frac{\Delta t}{2}\mathcal{A}_1 - \Delta t\mathcal{A}_2\right) u^{n-1} + \Delta t\beta f \\ &\quad - \frac{\Delta t}{2}(\mathcal{L}_1\mathcal{K}_2 + \mathcal{L}_2\mathcal{K}_1)\,(3u^{n-1} - u^{n-2}), \\ \left(1 + \frac{\Delta t}{2}\mathcal{A}_2\right) u^n &= u^* + \frac{\Delta t}{2}\mathcal{A}_2 u^{n-1}. \end{aligned} \tag{7}$$

The quantity u^* is an intermediate solution. In this article, we will call (7) the *Crank-Nicolson ADI* (CN-ADI) method for (2). Each half step of CN-ADI requires inverting a series of quint-diagonal matrices, which is computationally inexpensive.

Note that the CN-ADI algorithm (7) is a three-step procedure, defined well for $n \geq 2$. For $n = 1$, one may conveniently assume $u^{-1} = f = u^0$.

2.3 Algorithm parameters

In addition to having the three model parameters (β, σ and α), the CN-ADI iteration involves two extra algorithm parameters: Δt and n. The denoising computation must stop after an appropriate number of iterations. However, it is hard to find an analytic stopping time. Therefore, finding an appropriate iteration count, n, is an important problem. The timestep size Δt is also important for the effectiveness of the algorithm, due to the fact that the timestep is strongly related to the frequencies of image content that will be eliminated by the algorithm [14].

Thus, for a given contaminated image, values have to be selected for the algorithm parameters $\beta, \sigma, \alpha, \Delta t$ and n which result in the best restored image. However, when the original uncontaminated image is not known, assessing the quality of the restored image is difficult, if not impossible. In order to gain insight on the influence of these parameters on the quality of the restored image, we simulate the model on known images with synthetically-added noise. This simulation is the discussed in the next section.

3 Parametric Study

Preliminary to the development of a procedure for selecting parameters for the hybrid model described in the previous section, we simulated the model by applying it to restore known images with artificial Gaussian noise. As a measure of the quality of the restored images, we adopted the peak signal-to-noise ratio (PSNR) defined as

$$\text{PSNR} \equiv 10\log_{10}\left(\frac{\sum_{ij} 255^2}{\sum_{ij}(g_{ij}-u_{ij})^2}\right)\text{dB},$$

where g denotes the original uncontaminated image and u is the restored image.

In order to gain insight on the influence of the parameters $\beta, \sigma, \alpha, \Delta t$ and n on PSNR, we conducted a parametric study, following the pseudocode in Figure 1.

Establish uncontaminated image g
Add Gaussian noise to g to produce contaminated image f
Establish parameter counts $\text{N}_\beta, \text{N}_\sigma, \text{N}_\alpha, \text{N}_{\Delta t}, \text{N}_n$
Establish parameter values $\beta[1], \ldots, \beta[\text{N}_\beta]; \quad \sigma[1], \ldots, \sigma[\text{N}_\sigma];$
    $\alpha[1], \ldots, \alpha[\text{N}_\alpha]; \quad \Delta t[1], \ldots, \Delta t[\text{N}_{\Delta t}]; \quad n[1], \ldots, n[\text{N}_n]$
For each combination of $\beta, \sigma, \alpha, \Delta t, n$ values
  Apply denoising procedure on f to produce restored image u
  Calculate PSNR; output $\beta, \sigma, \alpha, \Delta t, n$ and PSNR
End for

Figure 1: High-level outline of parametric study

Various plots from the outputs of the study could be produced, including animations of PSNR as a function of $\beta, \sigma, \alpha,$ with either Δt or n fixed and using the other as the variable for the animation.

The number of combinations of parameter values is simply $\text{N}_\beta \times \text{N}_\sigma \times \text{N}_\alpha \times \text{N}_{\Delta t} \times \text{N}_n,$ which could be huge even for small to moderate values of the parameter counts. Fortunately, the denoising procedure can be computed simultaneously for several combinations of the parameters, on a parallel machine. We used a version of a dynamic load balancing tool based on MPI that was developed by the first author [3] to parallelize the serial code for Figure 1. The performance of the resulting parallel code on a general-purpose Linux cluster is described in the next section.

4 Parallel Performance

We conducted preliminary parametric studies with $N_\beta = 9$, $N_\sigma = 9$, $N_\alpha = 9$, $N_{\Delta t} = 9$ and $N_n = 15$, for a total of 98,415 parameter combinations, for a number of images. The studies were executed on the heterogeneous general-purpose EMPIRE cluster of the Mississippi State University. The cluster can be abstracted as in Figure 2 and has a total of 1038 processors. A rack contains 32 nodes of

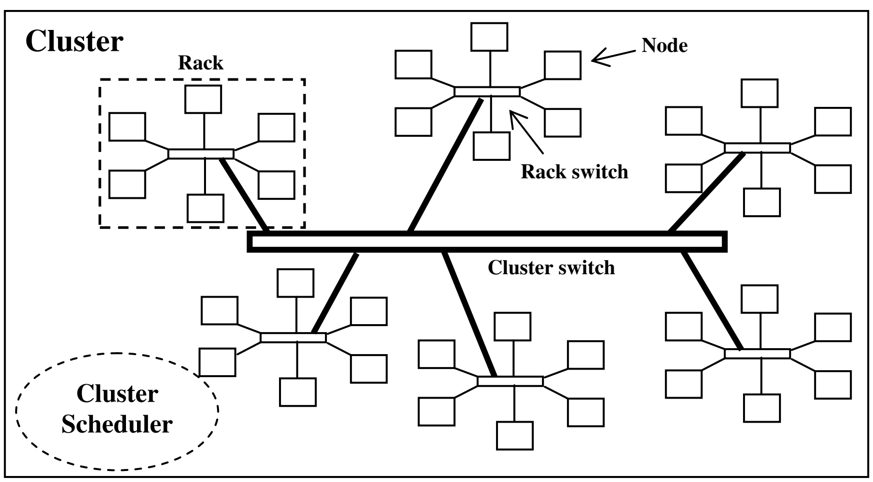

Figure 2: A popular interconnection network for clusters

dual 1.0GHz or 1.266GHz Pentium III processors and 1.25GB RAM. Each node is connected to a 100Mb/s Ethernet rack switch. The rack switches are connected by a gigabit Ethernet cluster switch. Installed software includes RedHat Linux and PBS. The general submission queue allows 64-processor, 48-hour jobs; a special queue allows 128-processor, 96-hour jobs from a restricted set of users. According to the Top 500 Supercomputer Sites list published in June 2002, EMPIRE then was the 126th fastest computer in the world and the 10th among educational institutions in the U.S.

Figure 3 gives a summary of the performance of the parallel code for the parametric study on the image *LenaGray256*. This study was submitted as a 32-processor job on the EMPIRE cluster; the cluster scheduler assigned homogeneous processors to the job. The left axis (for the bars) denotes the number of iterations of the loop in Figure 1 executed by a processor, while the right axis (for the diamonds) denotes the time in seconds taken by the processor to execute the iterations. The large differences in the number of iterations done by the processors indicate load imbalance. The job time measured by the cluster scheduler was 8.453 hours. An estimate of the sequential cost of the study is 260.4547 hours (∼10.9 days), which is the sum of the work times of the 32 processors. Thus, an estimate of the efficiency is: (estimated sequential cost)/(parallel cost) = (260.4547)/(32*8.453) = 0.963. The high efficiency indicates that the dynamic load balancing tool successfully addressed the load imbalance.

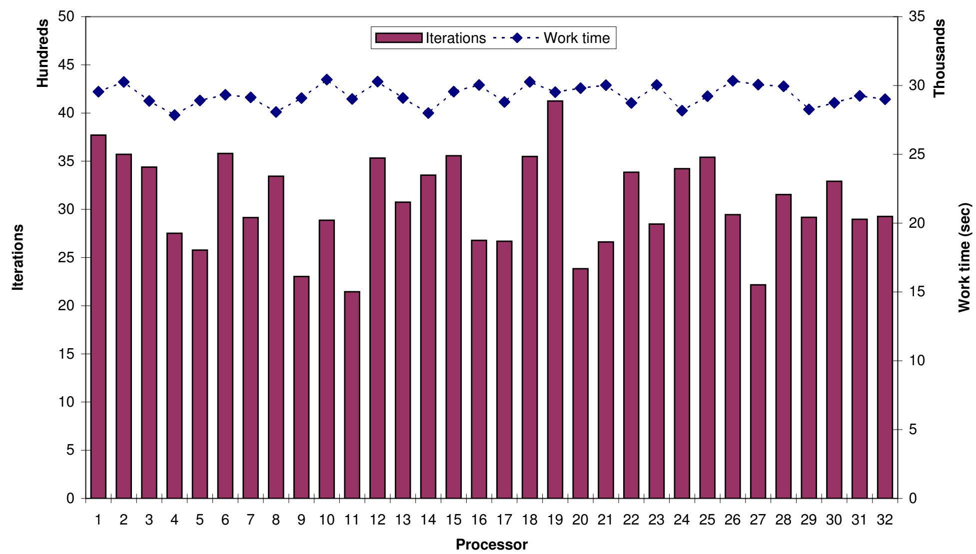

Figure 3: Distribution of iterations and work times for the parametric study on the image *LenaGray256*

Figure 4 gives the summary for the parametric study on the image *BlackCircle*.

The cluster scheduler again assigned homogeneous processors to the study, and the job time was 39.546 hours. The differences in iteration counts are significant, indicating the presence of load imbalance. An estimate of the sequential cost is 1,223.279 hours (∼51 days), which is the sum of the work times of the 32 processors. Thus, an estimate of the efficiency is: (estimated sequential cost)/(parallel cost) = (1223.279)/(32*39.546) = 0.967.

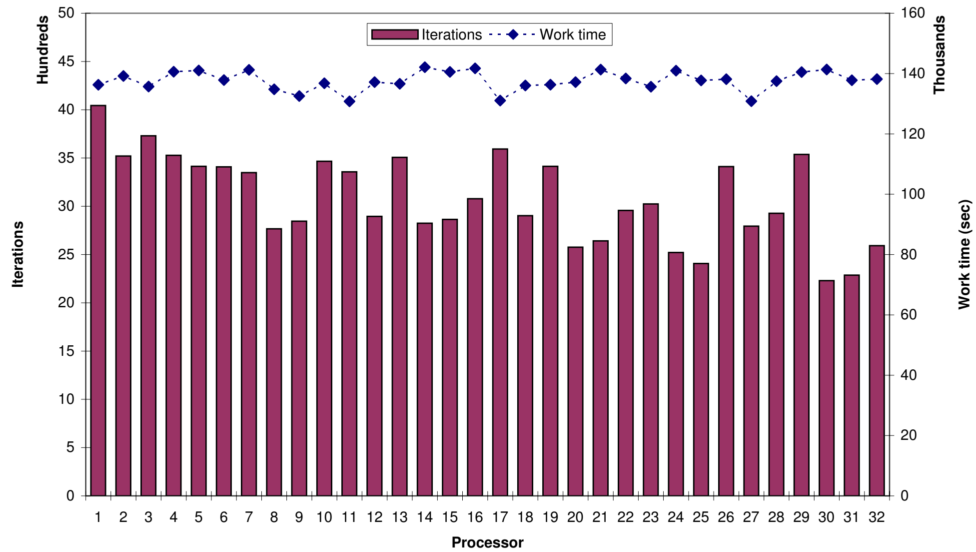

Figure 4: Distribution of iterations and work times for the parametric study on the image *BlackCircle*

5 Summary

We proposed a hybrid model for restoring noisy images. The hybrid model, which is based on the total variation model and the Laplacian mean-curvature model, contains a number of parameters that need to be fine-tuned in order to produce the best restored images. To guide the development of a procedure for selecting the parameters, we simulated the hybrid model by applying it to restore known images that were contaminated with Gaussian noise. The simulations were conducted as parametric studies on a general-purpose heterogeneous Linux cluster. To address the load imbalance that potentially arises from algorithmic and systemic sources, we used a dynamic load balancing tool in the simulation code. The simulations achieved very high estimated efficiencies.

We are currently analyzing the outputs of the parametric studies to develop methods for selecting optimal parameters of the hybrid model. The results of this analysis will be reported in the future.

Acknowledgments

This work was partially supported by the following National Science Foundation grants: EPS 0132618 (Cariño, Lim, Williams); CAREER 9984465 (Banicescu); and DMS 0312223 (Kim).

References

[1] L. Alvarez, P. Lions, and M. Morel, "Image selective smoothing and edge detection by nonlinear diffusion. II," *SIAM J. Numer. Anal.*, vol. 29, pp. 845–866, 1992.

[2] P. Blomgren and T. Chan, "Color TV: Total variation methods for restoration of vector valued images," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 304–309, 1998.

[3] R. L. Cariño and I. Banicescu. "A load balancing tool for distributed parallel loops." *Cluster Computing: The Journal of Networks, Software Tools and Applications*, (to appear).

[4] F. Catte, P. Lions, M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion." *SIAM J. Numer. Anal.*, vol. 29, pp. 182–193, 1992.

[5] Y. Cha and S. Kim, "Edge-forming methods for color image zooming," (accepted to IEEE Trans. Image Process.).

[6] ——, "Equalized net diffusion (END) for the preservation of fine structures in PDE-based image restoration," (submitted to IEEE Trans. Image Process.).

[7] ——, "MONTE: The method of nonflat time evolution in PDE-based image restoration," (submitted to IEEE Trans. Image Process.).

[8] T. Chan, S. Osher, and J. Shen, “The digital TV filter and nonlinear denoising,” Department of Mathematics, University of California, Los Angeles, CA 90095-1555, Technical Report #99-34, October 1999.

[9] T. Chan and J. Shen, “Variational restoration of non-flat image features: Models and algorithms,” *SIAM J. Appl. Math.*, vol. 61, no. 4, pp. 1338–1361, 2000.

[10] J. Douglas, Jr. and J. Gunn, “A general formulation of alternating direction methods Part I. Parabolic and hyperbolic problems,” *Numer. Math.*, vol. 6, pp. 428–453, 1964.

[11] J. Douglas, Jr. and S. Kim, “Improved accuracy for locally one-dimensional methods for parabolic equations,” *Mathematical Models and Methods in Applied Sciences*, vol. 11, pp. 1563–1579, 2001.

[12] R. Gonzalez and R. Woods, *Digital Image Processing, 2nd Ed.* Upper Saddle River, New Jersey: Prentice-Hall, Inc., 2002.

[13] J. Greer and A. Bertozzi, “Traveling wave solutions of fourth order pdes for image processing,” *SIAM J. Numer. Anal.*, vol. 36, no. 1, pp. 38–68, 2004.

[14] S. Kim, “Edge-preserving noise removal: Motion by mean curvature,” (submitted to SIAM J. Sci. Comput.).

[15] ——, “PDE-based image restoration: A hybrid model and color image denoising,” (accepted to IEEE Trans. Image Process.).

[16] S. Kim and H. Lim, “A non-convex diffusion model for simultaneous image denoising and edge enhancement,” (submitted to Electron. J. Differ. Equ.).

[17] R. Kimmel and N. Sochen, “Orientation diffusion or how to comb a porcupine ?” *Special issue on PDEs in Image Processing, Computer Vision, and Computer Graphics, J. Visual Comm. Image Representation*, vol. 13, pp. 238–248, 2002.

[18] M. Lysaker, A. Lundervold, and X. Tai, “Noise removal using fourth-order partial differential equations with applications to medical magnetic resonance images in space and time,” *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1579–1590, 2003.

[19] A. Marquina and S. Osher, “Explicit algorithms for a new time dependent model based on level set motion for nonlinear deblurring and noise removal,” *SIAM J. Sci. Comput.*, vol. 22, pp. 387–405, 2000.

[20] Y. Meyer, *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations*, ser. University Lecture Series. Providence, Rhode Island: American Mathematical Society, 2001, vol. 22.

[21] S. Mitra and G. Sicuranza, *Nonlinear Image Processing*. San Diego, San Francisco, New York, Boston, London, Sydney, ToKyo: Academic Press, 2001.

[22] J.-M. Morel and S. Solimini, *Variational Methods in Image Segmentation*, ser. Progress in Nonlinear Differential Equations and Their Applications. Birkhäuser, Boston, 1995, vol. 14.

[23] M. Nitzberg and T. Shiota, “Nonlinear image filtering with edge and corner enhancement,” *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 14, pp. 826–833, 1992.

[24] S. Osher, M. Burger, D. Goldfarb, J. Xu, and W. Yin, “Using geometry and iterated refinement for inverse problems (1): Total variation based image restoration,” Department of Mathematics, UCLA, LA, CA 90095, CAM Report #04-13, 2004.

[25] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol. 12, pp. 629–639, 1990.

[26] L. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D*, vol. 60, pp. 259–268, 1992.

[27] G. Sapiro, *Geometric partial differential equations and image analysis*. Cambridge: Cambridge University Press, 2001.

[28] N. Sochen, R. Kimmel, and R. Malladi, “A general framework for low level vision,” *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 310–318, 1998.

[29] L. Vese and S. Osher, “Numerical methods for p-harmonic flows and applications to image processing,” Department of Mathematics, University of California, Los Angeles, CA 90095, USA, Technical Report #01-22, August 2001, (To appear in SIAM Numer. Anal.).

[30] N. Williams, H. Lim, and S. Kim, "Numerical schemes for the Laplacian mean-curvature flow and their applications to image denoising," (In preparation).

[31] Y. You and M. Kaveh, "Fourth-order partial differential equations for noise removal," *IEEE Trans. Image Process.*, vol. 9, no. 10, pp. 1723–1730, 2000.

[32] Y. You, W. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image processing," *IEEE Trans. Image Process.*, vol. 5, pp. 1539–1553, 1996.