

An Effective Termination Criterion for the Rudin-Osher-Fatemi Algorithm for Image Restoration

Haoran Zhang and Wu-Sheng Lu
Department of Electrical and Computer Engineering
University of Victoria
Victoria, BC, Canada
{haoran, wslu}@ece.uvic.ca

Abstract—The Rudin-Osher-Fatemi (ROF) algorithm for de-noising and de-blurring of still digital images is well known to offer excellent image restoration performance. However, application of the ROF algorithm in the past has indicated that one can get the best results from the algorithm only when one terminates the algorithm at the “right time” which turns out to be way before the corresponding parabolic partial differential equation converges to its steady-state solution, and the determination of that right time for terminating the algorithm becomes a key issue for the best utilization of the algorithm. In this paper, a simple yet effective termination criterion for the ROF algorithm is proposed. Performance results are given to demonstrate the effectiveness of the proposed criterion when it is incorporated with the ROF algorithm.

I. INTRODUCTION

Image degradations are always undesirable. The most common example is digital imaging in which degradations impair the observer’s visual experience. In advanced image and video applications such as object detection and recognition, degradations also impose certain difficulties to image analysis tasks. Therefore, it is of critical importance that digital image processing techniques be investigated and applied in order to remove or reduce such degradations [1]-[3]. Image restoration is a signal processing procedure that attempts to reconstruct or recover a digital image degraded by blurring and noise contamination, and has been a subject of studies over the past four decades [1]-[7]. Among other things, the Rudin-Osher-Fatemi (ROF) algorithm for de-noising [4] and de-blurring [5] has been recognized as one of the most effective methods for the restoration of degraded still digital images. However, application of the ROF algorithm in the past has indicated that one can get the best result from the algorithm only when one terminates the algorithm at the “right time” which turns out to be way before the corresponding parabolic partial differential equation (PDE) converges to its steady-state solution, and the determination of that “right time” for the termination of the algorithm becomes a key issue for the best utilization of the

algorithm. In this paper, a simple yet effective termination criterion for the ROF algorithm is proposed. Performance results are given to demonstrate the effectiveness of the proposed criterion when it is incorporated with the original ROF algorithm.

The rest of the paper is organized as follows. In Section 2, the image model to be adopted and the ROF algorithm are briefly reviewed. In Section 3, algorithmic behavior of the ROF algorithm is analyzed by examining simulation results of the algorithm applied to a test image. The simulations serve to illustrate the performance difference between the best possible result of the algorithm and the result corresponding to the steady state of the evolutionary PDE. Based on the analysis, a termination criterion for the ROF algorithm is proposed. For simplicity and clarity, throughout we shall focus our attention on the de-noising problem. We should stress however that the proposed termination criterion has been found to work equally well when incorporated with the de-blurring algorithm proposed in [5]. Simulation studies of the ROF algorithm with the proposed termination criterion incorporated are presented in Section 4 where performance of the ROF algorithm is evaluated and compared with the well-known wavelet-based de-noising algorithm proposed by Donoho et al [6].

II. IMAGE MODEL AND THE ROF ALGORITHM

A. Image Model

The degradation model in the spatial domain is given by

$$u_0(x, y) = u(x, y) + v(x, y) \quad (1)$$

where $u_0(x, y)$ is the observed image, $u(x, y)$ is the desired image, and $v(x, y)$ denotes the noise that contaminates the pure image $u(x, y)$. It is assumed that the noise $v(x, y)$ is a realization of a Gaussian random process with zero mean and standard deviation σ . The de-noising problem at hand is to derive a kind of optimal estimation of the original pure image $u(x, y)$ based on the contaminated observation $u_0(x, y)$.

B. The ROF Algorithm

Let Ω be the region where the image function $u(x, y)$ in (1) is defined. The *total variation* (TV) of the image is defined as

$$J_{TV}[u] = \iint_{\Omega} \sqrt{u_x^2 + u_y^2} dx dy \quad (2)$$

where $u = u(x, y)$, $u_x = \partial u / \partial x$, and $u_y = \partial u / \partial y$. In [4], it is proposed that the de-noising problem be addressed by solving the constrained variational optimization problem

$$\text{Minimize } \iint_{\Omega} \sqrt{u_x^2 + u_y^2} dx dy \quad (3a)$$

$$\text{Subject to: } \iint_{\Omega} u(x, y) dx dy = \iint_{\Omega} u_0(x, y) dx dy \quad (3b)$$

$$\frac{1}{2} \iint_{\Omega} [u(x, y) - u_0(x, y)]^2 dx dy = \sigma^2 \quad (3c)$$

Since the presence of noise in an image leads to a higher value of the TV of the image but, on the other hand, the TV of the image is also contributed by the edges and oscillatory features (such as textures) of the image itself, it is intuitively justified that the de-noising problem can be addressed by minimizing the TV of the image subject to the reasonable constraints (3b) and (3c) which essentially state that the noise contained in the image as a 2-D random field has zero mean and standard deviation σ .

By defining the Lagrangian of (3) as the functional

$$L(u, \lambda_1, \lambda_2) = \iint_{\Omega} \left[\sqrt{u_x^2 + u_y^2} + \lambda_1(u - u_0) + \lambda_2 \left[\frac{1}{2}(u - u_0)^2 - \sigma^2 \right] \right] dx dy \quad (4)$$

where λ_1 and λ_2 are Lagrange multipliers and applying the Euler-Lagrange equation to it, one concludes that the solution of problem (3) satisfies the second-order PDE

$$\frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}} \right) + \frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right) - \lambda_1 - \lambda_2(u - u_0) = 0 \quad (5a)$$

subject to the boundary condition

$$\frac{\partial u(x, y)}{\partial n} = 0 \text{ on the boundary } \Gamma = \partial\Omega \quad (5b)$$

where n denotes the normal along the boundary of region Ω . In the ROF method [4], the solution of (5) is obtained by introducing a time-dependent image function $u(t, x, y)$ and solving the nonlinear parabolic PDE for $u(t, x, y)$ with time t as an evolution parameter:

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}} \right) + \frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right) - \lambda(u - u_0) \text{ for } t > 0, (x, y) \in \Omega \quad (6a)$$

$$u(x, y, 0) \text{ given} \quad (6b)$$

$$\frac{\partial u(x, y, t)}{\partial n} = 0 \text{ on the boundary } \Gamma = \partial\Omega \text{ for } t > 0 \quad (6c)$$

where the initial value $u(x, y, 0)$ in (6b) is typically taken to be the observed noisy image data $u_0(x, y)$. It was argued that as time t moves forward, the solution of (6) converges to a steady-state and that steady-state solution $u(x, y) = \lim_{t \rightarrow \infty} u(x, y, t)$ as t approaches infinity, which is obviously independent of time t , therefore satisfies Eq. (5) [4]. The Lagrange multiplier in (6a) is now time-dependent and is found to be given by

$$\lambda(t) = -\frac{1}{2\sigma^2} \iint_{\Omega} \left[\sqrt{u_x^2 + u_y^2} - \left(\frac{(u_0)_x u_x}{\sqrt{u_x^2 + u_y^2}} + \frac{(u_0)_y u_y}{\sqrt{u_x^2 + u_y^2}} \right) \right] dx dy \quad (7)$$

III. ANALYSIS OF THE ROF ALGORITHM AND AN EFFECTIVE TERMINATION CRITERION

A. Analysis of the ROF Algorithm

The theoretical justification for the approach taken in [4], as outlined above, “comes from the fact that it is merely the gradient-projection method of Rosen.” After comparing the method with a work by Geman and Reynolds based on simulated annealing, it is realized in [4] that the steady-state solution of (5) offers only a *local* solution of the original variational optimization problem (3). This sub-optimality is also confirmed by our application experiences with the ROF algorithm as they have indicated that improved de-noising performance can be achieved by terminating the algorithm at a right time before the steady-state is reached. To illustrate our point, we apply the ROF algorithm to an 8-bit test image $u(x, y)$ of size 512×512 known as “boat”. In the simulation, the observed noisy image $u_0(x, y)$ is generated by adding a Gaussian noise with zero-mean and $\sigma = 30$ to the original noise-free image $u(x, y)$. As usual, the performance of the algorithm is evaluated in terms of peak signal-to-noise ratio which is defined as

$$\text{PSNR} = 10 \log_{10} \left[\frac{255^2}{\text{MSE}} \right] \text{ (dB)} \quad \text{where MSE is given by}$$

$$\text{MSE} = \frac{1}{MN} \sum_{i=1}^N \sum_{j=1}^M [u(i, j) - \hat{u}(i, j)]^2$$

with $u(i, j)$ and $\hat{u}(i, j)$ representing the original and de-noised images, respectively. The numerical method described in [6] for solving (6) was applied to the noisy “boat”, i.e. $u_0(x, y)$, and the results obtained are shown in Figs. 1 and 2. The two curves shown Fig. 1 are the progressive PSNR over the first 500 iterations (see Fig. 1a on the left) and the Lagrange function $\lambda(t)$ in (7) over the first 500 iterations (see Fig. 1b on the right). Since the iteration number is proportional to time t , these curves can also be viewed as the PSNR and Lagrange multiplier values versus time t . From the figures it is observed that the Lagrange function converges after 500 or so iterations and the PSNR tends to converge after 500 iterations. It is also observed that the peak of the progressive PSNR occurs after 120 iterations, way before the algorithm converges. The PSNR

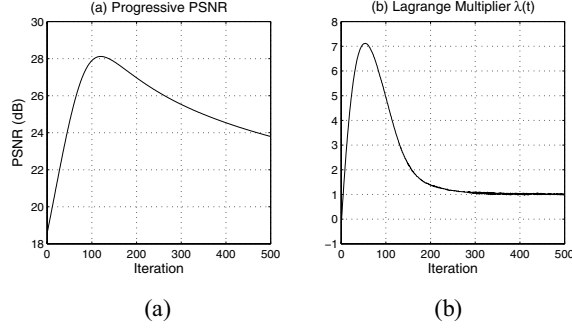


Figure 1: (a) Progressive PSNR for image *boat* and (b) Lagrange function $\lambda(t)$ over first 500 iterations for image *boat*.

of the noisy data, i.e. $u_0(x, y)$, was found to be 18.59 dB, after 120 iterations the PSNR reached its peak at 28.12 dB, and after 500 iterations the PSNR was 23.80 dB.

Visual examination of the de-noising performance is provided by Fig. 2 where the de-noising results after 120 iterations and 500 iterations are shown and compared with the original and initial noisy data. We see that the de-noised image after 120 iterations (in Fig. 2c on the lower left) possesses superior visual quality where the noise is largely removed but the image edges and other details are well preserved, while the de-noised image after 500 iterations (in Fig. 2d on the lower right) is obviously over smoothed where the noise is completely wiped out but a great deal of image details are also lost. It should be stressed that the observations made here are not only valid for the particular image used in the simulation, but also for a large variety of digital images.

The above analysis serves a motivation to develop a termination criterion that can be used to identify the right time to bring the algorithm to a halt so as to achieve near-optimal performance the ROF algorithm has to offer.

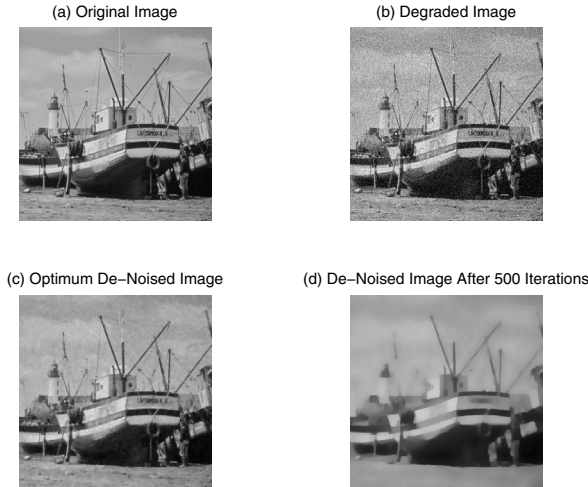


Figure 2: (a) original image “boat”, (b) noisy “boat”, (c) de-noised “boat” after 120 iterations, and (d) de-noised “boat” after 500 iterations.

B. A Termination Criterion for the ROF Algorithm

As is seen from Fig. 1a, the progressive PSNR exhibits a monotonic increasing period before reaching its peak value.

Mathematically, assuming the progressive PSNR is a differentiable function of time, this means that the PSNR curve over a time duration from $t = 0$ to certain moment, say $t = t_c$, has positive derivative which then gradually tend to zero. As such, detecting the appropriate time instant to terminate the algorithm amounts to detecting the *first time instant when the rate of PSNR improvement becomes very small in magnitude*. The idea just described can be implemented in terms of the available image data as follows. Because the original noise-free image is not available in a practical application, we define a quantity, $d(n)$, as

$$d(n) = \|u_n - u_0\|_F \quad (8)$$

where $\|\cdot\|_F$ is the Frobenius norm, u_0 is the noisy image, and u_n is the de-noised image after n iterations. Using (1), $d(n)$ can be expressed as

$$d(n) = \|u - u_n + v\|_F \quad (9)$$

From (9) we see that as iterate u_n gets closer to the original image u , $d(n)$ tends to be a constant value $\|v\|_F$. In words, as the iteration starts, we shall see a fairly dynamic $d(n)$ as the part $u - u_n$ changes considerably, but as the iteration continues, $d(n)$ becomes less and less dynamic and the “right time” to terminate the iteration is the moment when the dynamics of $d(n)$ is considered insignificant. The nice thing about using $d(n)$ to catch the right moment is that, as (8) indicates, $d(n)$ is readily available while the PSNR is not because the latter involves the original image u . Since the significance of the rate of $d(n)$ can be measured by

$$\delta_n = |d(n) - d(n-1)| \quad (10)$$

we propose that the ROF algorithm be terminated when the value of δ_n falls below a prescribed threshold ε , i.e.,

$$|d(n) - d(n-1)| < \varepsilon \quad (11)$$

In a wide range of simulations, it was found that a value of threshold ε in the range $[0.01, 0.05]$ usually gives good results. The additional computational burden required by this criterion is fairly light that is the evaluation of the Frobenius norm of $u_n - u_0$ and verification of condition (11) in each iteration. As an example, the plots in Fig. 3 demonstrate how this termination criterion works. Again the image *boat* was used with the same noise-corrupted version as in Fig. 2a and b. Fig. 3a shows a profile of PSNR during the first 200 iterations while Fig. 3b shows how the Frobenius norm $\|u - u_n\|_F$ changes as the iteration continues where u represents the noise-free image. From this figure it is quite clearly that the de-noised image u_n is closest to the ideal image u after 120 iterations. The problem with this approach is however that this profile is usually *not* available because it would require the precise knowledge of image u . Fig. 3c and d depict the profiles of $d(n)$ and δ_n defined by (8) and (10), respectively. If a threshold $\varepsilon = 0.05$ is used, then the number of iterations required to perform so that δ_n begins to fall below threshold

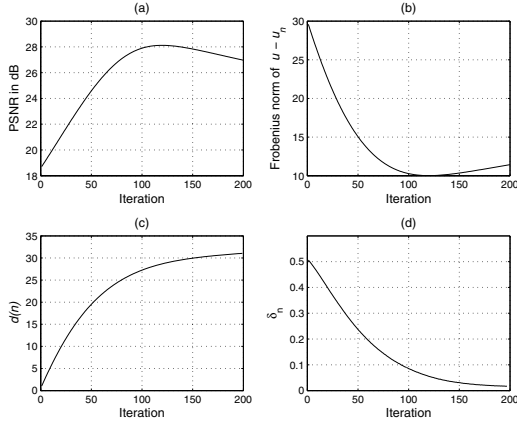


Figure 3: (a) Progressive PSNR for image *boat*, (b) $\|u - u_n\|_F$ versus iteration, (c) $d(n)$ versus iteration, and (d) δ_n versus iteration.

ε is found to be 125 (see also Fig. 3d), which appears to be a good estimate of the optimum iteration number. Here we stress the point that both $d(n)$ and δ_n are readily available in practice.

IV. PERFORMANCE EVALUATION

Two test images, namely *boat* and *lena* (both are 8 gray-level images of size 512×512), were used in the simulations. To each test image, Gaussian noise of zero-mean and $\sigma = 30$ was added in order to generate a noisy image u_0 , and a threshold $\varepsilon = 0.05$ was used for the ROF algorithm for both images. For comparison purposes, the wavelet-based algorithm proposed by Donaho et al [6] for image de-noising was also applied to the same test images. The wavelet used was the Daubechies wavelet of length 8 (D8) which was found to produce the best de-nosing results within the Daubechies wavelet family for the images used here. The algorithms' performance is evaluated in terms of PSNR and visual quality of the de-noised images. In addition, the CPU times required by the algorithms were also recorded.

For image *boat*, the PSNR obtained by the ROF and wavelet algorithms were found to be 28.09 dB and 25.29 dB, respectively. The ROF algorithm however required 59.61 seconds of CPU time versus 9.34 seconds for the wavelet algorithm on a PC with a 3.1 GHz Pentium 4 processor. For image *lena*, the PSNR achieved by the ROF and wavelet algorithms were found to be 29.54 dB and 26.70 dB, respectively. The ROF algorithm required 56.91 seconds of CPU time versus 9.34 seconds for the wavelet algorithm. We see that in both cases the ROF algorithm offers a 2.8 dB gain relative to the wavelet algorithm. Visual examination of the de-noising results (see Figs. 4 and 5) also confirms the superiority of the ROF algorithm as it well preserves edges and other image details. It is also noted that the performance gains of the ROF algorithm were achieved at the cost of increased computational complexity. Our simulations have indicated that the ROF algorithm shall be an excellent candidate method for image de-noising tasks where performance is a primary concern relative to complexity.

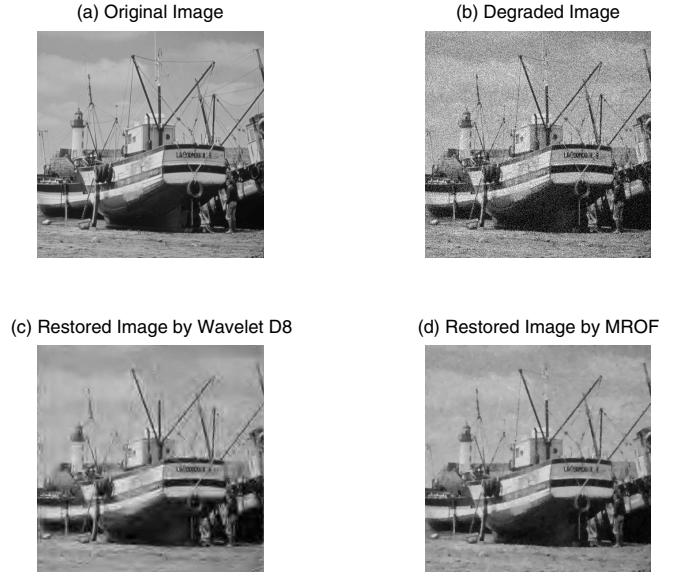


Figure 4: (a) Original *boat*, (b) noisy *boat*, (c) de-noised *boat* by wavelet D8, and (d) de-noised *boat* by the ROF algorithm.



Figure 5: (a) Original *lena*, (b) noisy *lena*, (c) de-noised *lena* by wavelet D8, and (d) de-noised *lena* by the ROF algorithm.

REFERENCES

- [1] A. Rosenfeld, *Picture Processing by Computer*, Academic Press, 1969.
- [2] A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, 1989.
- [3] R. C. Gonzalez, and R. E. Woods, *Digital Image Processing*, 2nd edition, Prentice-Hall, 2002.
- [4] L. I. Rudin, S. Osher and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D.*, vol. 60, pp. 259-268, 1992.
- [5] L. I. Rudin and S. Osher, "Total variation based image restoration with free local constraints," *Proc. ICIP*, pp. 31-35, Austin, TX., 1994.
- [6] D. Donoho, I. Johnstone, G. Kerkyacharian, and D. Picard, "Wavelet shrinkage: Asymptopia?" *J. Roy. Stat. Soc.*, vol. 57, pp. 301-369, 1995.
- [7] A. Chambolle, "An algorithm for total variation minimization and application" *J. Math. Imaging and Vision*, vol. 20, pp. 89-97, 2004.