

# Iterative Projective Wavelet Methods for Denoising

Alyson K. Fletcher, Vivek K Goyal and Kannan Ramchandran

alyson@eecs.berkeley.edu, v.goyal@ieee.org, kannanr@eecs.berkeley.edu

Berkeley Audio-Visual Signal Processing and Communication Systems group  
Department of Electrical Engineering and Computer Sciences  
University of California, Berkeley  
Berkeley, CA 94720 USA

## ABSTRACT

Wavelet thresholding is a powerful tool for denoising images and other signals with sharp discontinuities. Using different wavelet bases gives different results, and since the wavelet transform is not time-invariant, thresholding various shifts of the signal is one way to use different wavelet bases. This paper describes several denoising methods that apply wavelet thresholding or variations on wavelet thresholding recursively. (We previously termed one of these methods “recursive cycle spinning.”) These methods are compared experimentally for denoising piecewise polynomial signals. Though similar, the methods differ in computational complexity, convergence speed, and sensitivity to threshold selection.

**Keywords:** adaptive projection, cycle spinning, estimation, frames, shift invariance, thresholding

## 1. INTRODUCTION

Removing additive noise is a signal processing problem that arises in many contexts. The standard setting has a wide-sense stationary (WSS), Gaussian signal corrupted by additive, WSS Gaussian noise that is independent of the signal. The ideal estimate is a linear function of the observation, and the appropriate filter to compute this estimate can be determined from the power spectra of the signal and noise. Denoising becomes conceptually more difficult when one strays from the textbook model because there is no reason to assume that linear estimators will perform well.

An effective technique for denoising signals that are approximately piecewise smooth, including natural images, is to apply nonlinear threshold estimators in the wavelet domain. (The *hard threshold* estimator is described in the following section.) By identifying and preserving significant coefficients, wavelet thresholding preserves important high-pass features of the signal such as discontinuities.

Wavelet thresholding has been empirically justified for quite some time<sup>1</sup> and its present popularity has followed largely from the optimality properties shown by Donoho and Johnstone.<sup>2,3</sup> Since wavelet transforms are not shift invariant, denoising by thresholding of wavelet coefficients is also not shift invariant. Coifman and Donoho<sup>4</sup> suggested using a shift-invariant estimate obtained by averaging over all of the shifts that give distinct estimates; they call this *cycle spinning*.

Extending our earlier work,<sup>5</sup> this paper introduces a number of iterative algorithms that attempt to better combine the information from the various shifts of the signal. Simulations on denoising piecewise polynomials show that recursive algorithms offer significant improvements over conventional wavelet denoising methods. Trade-offs between robustness and speed of convergence are observed. In addition, we consider the issue of threshold selection in greater detail. The ultimate aim is to understand how to best utilize sparseness of signal representations in various bases and frames for denoising; more immediately, we are attempting to exploit sparse representation in wavelet bases and periodic shift invariance of wavelet threshold operators. The sparseness of natural images in wavelet bases has led to the dominance of wavelet-based techniques in image compression.<sup>6,7</sup>

---

V. K Goyal is a Visiting Scholar at the University of California, Berkeley. Effective January 2004 he will be with the Department of Electrical Engineering and Computer Science and the Research Laboratory of Electronics at the Massachusetts Institute of Technology.

## 2. EXISTING NON-ITERATIVE TECHNIQUES

### 2.1. Basic Wavelet Thresholding

Consider estimating an unknown discrete-time signal  $x[n]$ ,  $n \in \{0, 1, \dots, N-1\}$ , from a noise-corrupted signal,

$$y[n] = x[n] + d[n], \quad (1)$$

where  $d[n]$  is additive noise. Let  $X[n] = \mathcal{W}(x[n])$  denote a discrete wavelet transform of  $x[n]$ . Only orthogonal wavelet transforms are considered here.

Basic wavelet denoising is performed by taking the wavelet transform of the noise-corrupted  $y[n]$  and then zeroing out a subset of the coefficients that fall below a certain threshold. An inverse wavelet transform is applied to the thresholded signal to yield an estimate for the true signal, as below:

$$\hat{x}[n] = D(y[n]) = \mathcal{W}^{-1}(\Lambda(\mathcal{W}(y[n])))$$

where  $\Lambda$  is the diagonal *hard thresholding* operator

$$(\Lambda Y)[k] = \begin{cases} 0, & \text{if } |Y[k]| \leq T(k); \\ Y[k], & \text{otherwise} \end{cases} \quad (2)$$

where  $T(k)$  are certain threshold levels. The idea is that the signal is sparse in the wavelet domain; i.e., most of its energy is contained in a few large components, usually corresponding to discontinuities or other local high-pass features. Thresholding removes all the noise in the other components while preserving the large signal components.

### 2.2. Cycle Spinning

Wavelet transforms are not time-invariant. Consequently, if the noisy signal is shifted in time, denoised, and then shifted back, the result will, in general, be different from the estimate obtained from denoising without shifting. Specifically, for each shift  $i$ , we can obtain an estimate

$$\hat{x}_i[n] = D_i(y[n]) = S_{-i} D S_i(y[n]) \quad (3)$$

where  $D$  is a wavelet denoising operator described above and  $S_i$  is a left shift by  $i$  samples. It can be shown that if a wavelet transform has  $J$  stages, one can obtain up to  $M = 2^J$  different estimates,  $\hat{x}_i[n]$ ,  $i = 0, \dots, M-1$ . The cycle spinning estimate is simply the linear average of these  $M$  estimates. The idea is that the errors in the estimates are different and will be reduced by the averaging.

### 2.3. Wavelet Footprints

Wavelet footprint bases were introduced by Dragotti and Vetterli.<sup>8,9</sup> In brief, given a point for a possible discontinuity, using a wavelet footprint basis accounts for the deterministic relationships between detail coefficients within the cone of influence of the discontinuity. This allows wavelet footprint bases to represent piecewise polynomial signals very sparsely. Denoising by thresholding in a wavelet footprint basis has been shown to be more effective than basic wavelet thresholding. However, this technique is not shift-invariant and can thus be improved by cycle spinning.<sup>9</sup>

## 3. ITERATIVE TECHNIQUES

### 3.1. Basic Algorithm and Motivation

In certain applications, cycle spinning can be improved upon by a recursive algorithm that we call *recursive wavelet thresholding* (RWT). RWT attempts to better utilize the presumed sparseness of the signal in the  $M$  bases. To motivate the algorithm, we reconsider wavelet thresholding and cycle spinning in the context of projections.

Wavelet thresholding, in essence, identifies a subspace in which the unknown signal is likely to have most of its energy and projects the noisy signal onto that subspace. Cycle spinning thereby identifies  $M$  subspaces, performs the corresponding denoising projections, and averages the  $M$  projections. Instead of averaging the projections, it can be argued that it is better to project once onto the *intersection* of the  $M$  subspaces: If the unknown signal has most of its energy in each of the subspaces, it will have most of its energy in their intersection. Since the intersection of subspaces typically has a much smaller dimension than the subspaces themselves, projecting to the intersection produces an estimate much closer to the true signal. Unfortunately, there is no simple way to directly project to this intersection; RWT can be seen as an approximate iterative solution. The algorithm is initialized to the original noisy signal,  $\hat{x}_0[n] = y[n]$ , and recursively updates with the rule

$$\hat{x}_{\ell+1} = D_i(\hat{x}_\ell), \quad i \equiv \ell \pmod{M}, \quad (4)$$

where  $D_i(\cdot)$  is the denoising operator that uses a shift of  $i$ . The algorithm repeatedly cycles through the  $M$  shifts, using the denoised output from one iteration as the input to the next. Any fixed point,  $\hat{x}_\infty$ , will satisfy  $\hat{x}_\infty = D_i(\hat{x}_\infty)$  for all  $i$ , and will therefore be in the range space of denoising projections for all  $i$ , as desired. A previous conference paper shows that this recursion is guaranteed to converge under certain mild assumptions on the thresholding.<sup>5</sup>

### 3.2. Windowed Thresholding

In RWT, it is critical that not too many wavelet coefficients are set to zero in the thresholding step. For example, suppose the true signal has a discontinuity at some point. The discontinuity will result in a number of non-zero detail coefficients depending on the filter length. However, even for a large discontinuity, some of these coefficients may be small and set to zero in the thresholding step. RWT will force the final signal estimate to also be zero on these coefficients. If too few coefficients are allowed to be non-zero, there may be insufficient degrees of freedom in the estimate and the RWT algorithm will incorrectly enforce a linear dependence between the portion of the signal before and after the discontinuity.

To avoid this problem, RWT can be implemented with “windowed” thresholding. In windowed thresholding, a wavelet coefficient is set to zero only if all the coefficients in some interval around the coefficient do not exceed the threshold level. That is, the thresholding operator in (2) is replaced by the rule that  $(\Lambda Y)(k) = 0$  when

$$|Y(j)| \leq T(j) \text{ for all } j \in [k - W(k), k + W(k)] \quad (5)$$

where  $W(k)$  is a window size that depends on the filter length and length of the filter.

### 3.3. Soft vs. Hard Thresholding

In wavelet denoising, it is common practice to use a soft thresholding function, where the coefficient is slightly reduced even when it is not set to zero. Under certain statistical assumptions, soft thresholding can result in slightly greater noise reduction. In particular, a soft threshold function gives a MAP estimate of a Laplacian signal variable in Gaussian noise.

RWT must be implemented with hard thresholding; if soft thresholding were used repeatedly, the estimate would decrease in norm in each iteration, eventually converging to zero.

### 3.4. Threshold Selection

A large fraction of the engineering literature on wavelet-based methods for denoising focuses on the selection of threshold values for denoising images. Initially many researchers noted that the universal and asymptotically optimal thresholds of Donoho<sup>3</sup> could be improved by ad hoc means. Some proposed means for computing a better single threshold to apply to all the wavelet coefficients while others suggested using different thresholds for different subbands. The best thresholding in a wavelet basis is achieved when the threshold is adapted spatially and with scale.<sup>10</sup> A similar adaptation is motivated by Rissanen’s minimum description length

(MDL) principle in work by Hansen and Yu.<sup>11</sup> Further, the concepts of denoising by compression articulated by Natarajan<sup>12</sup> and MDL inspired the threshold selections in Ref. 13.

Here we use a scale-adapted threshold that is proportional to an estimate of the noise standard deviation. The proportionality factor is explored experimentally. The threshold level in each subband is

$$T = \beta \cdot \frac{\hat{\mu}}{\sqrt{2}\text{erf}^{-1}(\frac{1}{2})}, \quad (6)$$

where  $\beta$  is a constant and  $\hat{\mu}$  is the median absolute value of the coefficients in the subband.\* Our simulations show that the RWT algorithm is relatively insensitive to the choice of  $\beta$  for a number of test signals.

## 4. ALGORITHM VARIATIONS

### 4.1. Fixed Projections

In RWT, the coefficients to be zeroed out in the thresholding phase of the denoising are reselected in each iteration. The reselection is what makes it different from projection onto convex sets (POCS) with  $M$  convex sets. The intuitive advantage of reselecting the coefficients to be zeroed is that, as the denoising progresses, a better estimate can be made of the subspace that the true signal belongs to. For comparison, we consider a variant where the coefficients to be zeroed out are *not* reselected in each iteration. We will call this *RWT with fixed projections* (RWT-F).

### 4.2. Recursive Cycle Spinning

Another simple variant of the RWT algorithm is to replace the RWT update (4) with

$$\hat{x}_{\ell+1} = \frac{1}{M} \sum_{i=0}^{M-1} D_i(\hat{x}_{\ell}).$$

This is identical to infinitely repeating the standard cycle spinning denoising. We will call this *recursive cycle spinning* (RCS).

## 5. SIMULATION RESULTS

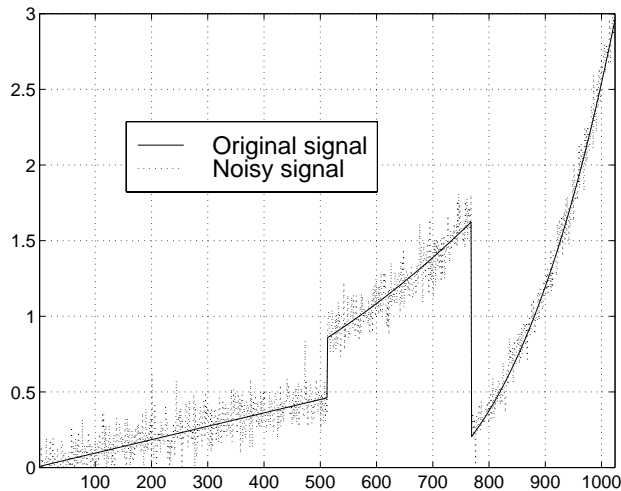
The methods presented above were tested for denoising the signal shown in Fig. 1. (A typical noisy version with an SNR of 20 dB is also shown.) The signal is a length-1024 piecewise fourth-order polynomial with five discontinuities. Experiments with several other piecewise polynomials gave similar results.

Fig. 2 compares methods with Daubechies  $D_2$  and  $D_3$  filter pairs and 1 to 4 transform stages. Noise reduction was averaged over 50 independent noise samples for each wavelet type and method. The three recursive methods were performed for 100 iterations and used the windowed thresholding in Section 3.2. All methods used the median-based thresholding rule in Section 3.4 with  $\beta$  selected for optimal noise reduction. The RWT method shows up to a 5 dB increase in noise reduction over standard, single-basis, wavelet thresholding, and RCS offers up to a 4 dB improvement over standard cycle spinning.

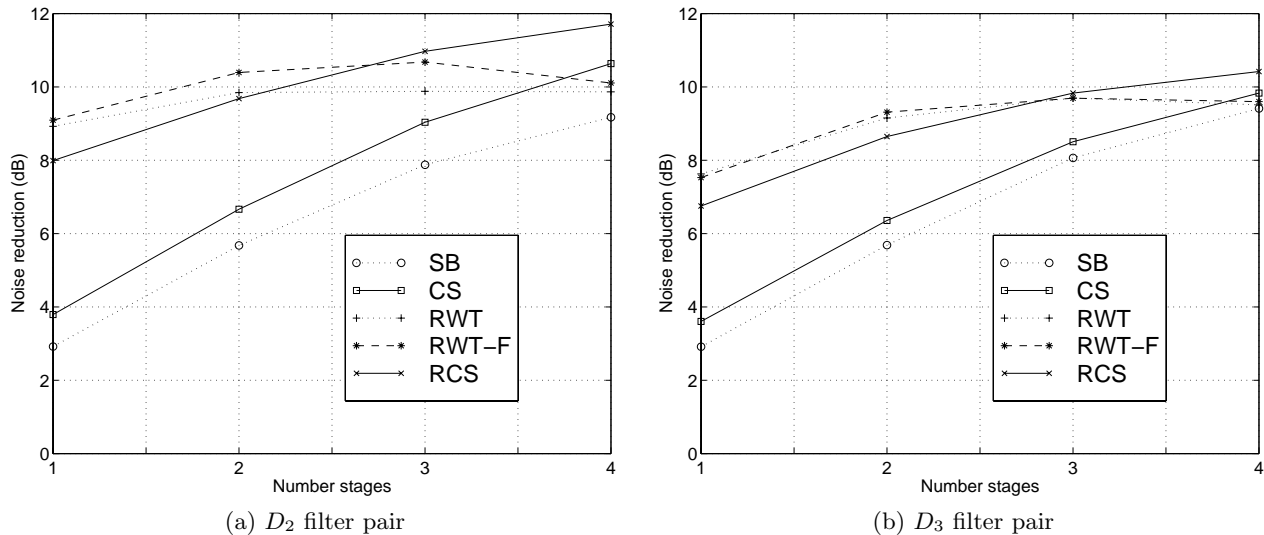
Fig. 3 shows an exploration of the importance of the threshold factor  $\beta$ . At a low input SNR of 10 dB, the noise reduction is roughly constant for  $\beta \geq 3$ . At the higher input SNR of 20 dB, the RWT and RCS method degrades only by less than 2 dB from  $\beta = 3$  to 7. With the median-based filtering, the RWT and RCS methods are not overly sensitive to the threshold level selection, particularly at lower SNRs. However, at the higher SNR value of 20 dB, the performance of the RWT method with fixed projections does degrade significantly when the threshold level is significantly larger than the optimal value of  $\beta = 3$ . This demonstrates that although fixed projections reduces the computational complexity, the RWT-F algorithm is significantly more sensitive to the threshold selection.

---

\*The second factor in (6) is a robust estimate of the standard deviation of the noise.<sup>14</sup>



**Figure 1.** Test piecewise polynomial: True signal and typical noisy signal with SNR=20 dB.



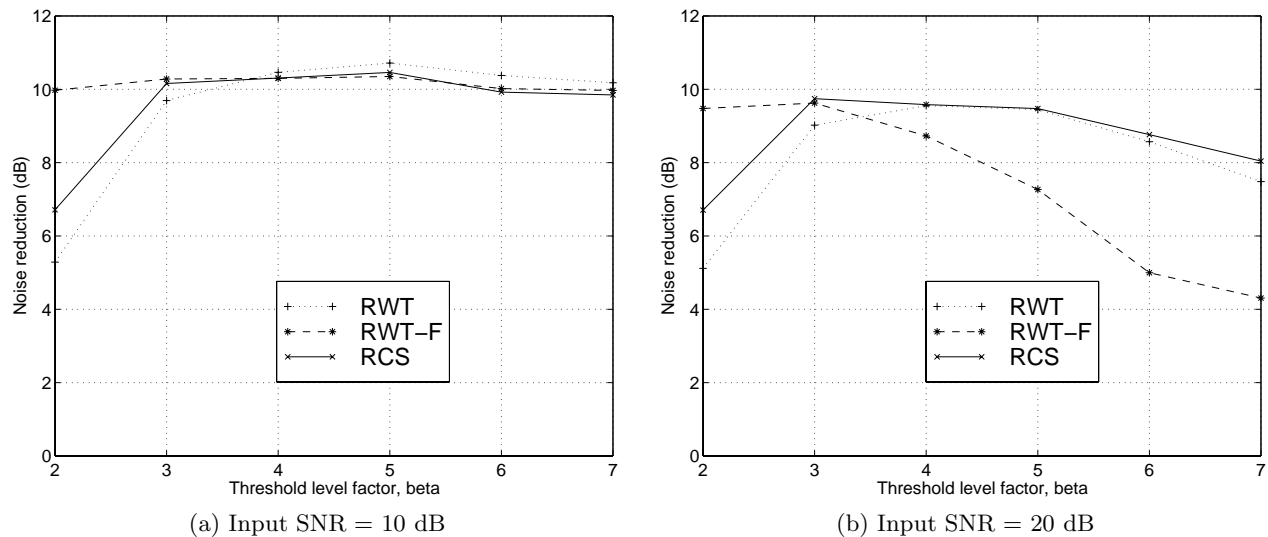
**Figure 2.** Noise reduction of a piecewise polynomial signal with various wavelet methods. Original SNR=20 dB. Methods are SB: basic (single-basis) wavelet thresholding, CS: standard cycle spinning, RWT: recursive wavelet thresholding, RWT-F: RWT with fixed projections and RCS: recursive cycle spinning.

## 6. COMMENTS AND FURTHER EXPERIMENTATION

This work is motivated by the following line of reasoning:

Denosing by wavelet thresholding works because of the high accuracy of sparse approximations in wavelet bases.<sup>†</sup> Averaging signals that are sparse in various bases generally gives a result that is sparse in none of the bases. Thus cycle spinning is somehow inconsistent with the optimal exploitation of sparseness in the various bases.

<sup>†</sup>Another way to say this is that wavelet thresholding works because of the power of nonlinear approximation in wavelet bases.<sup>15</sup>



**Figure 3.** Noise reduction of recursive methods using various threshold levels  $\beta$ .

The iterative procedures described in this paper illustrate various ways to promote more sparseness than is obtained with cycle spinning. When wavelets designed with vanishing moment properties are used, enforcing sparseness in a variety of bases amounts to the use of a piecewise polynomial signal model or a close relative to such a model. This leads to a counterargument of which we are completely aware: If the point is to optimally fit a piecewise polynomial signal model in an MSE sense, there are computationally more efficient means than to iteratively use any form of wavelet thresholding. In some of our other work, we explore the use of Kalman filtering to simultaneously choose the breakpoints in the signal model and fit polynomial pieces.

Signals of practical interest are not piecewise polynomial but rather are approximately piecewise polynomial. Thus, fitting a piecewise polynomial generally results in an estimate that is oversmoothed. For such signals, the lesson of this work is that while repeated application of denoising operators until convergence may result in oversmoothing, there is no reason to believe that *one* is the optimal number of applications.

In experiments with the *Lena* image,<sup>16</sup> we have found that two applications of cycle spinning can be better than one—albeit by a very small margin. For example, in experiments with a noise standard deviation of  $\sigma = 15$  we found that the optimal threshold value for cycle spinning with hard thresholding is approximately 44.3 and results in 7.988 dB noise reduction. However, using first a threshold of 18.1 and then a threshold of 38.9 gives 8.021 dB noise reduction. The results are similar with  $\sigma = 50$ : the best noise reduction in one cycle spinning step is by 12.74 dB with threshold 166, while two cycle spinning steps with thresholds 77 and 129 gives 12.81 dB noise reduction. Also similar are the results obtained with soft thresholding. With soft thresholding and  $\sigma = 50$ , the best one-step noise reduction of 9.74 dB is improved to 10.26 dB for two steps. Furthermore, 67 steps with a small threshold of 1.0 improved the noise reduction to 10.75 dB. The interest in these results is that different (and slightly better) results are obtained without any signal modeling effort.

## ACKNOWLEDGMENTS

The authors thank Dr. Sundeep Rangan for extensive feedback at various stages of this research.

## REFERENCES

1. Y. Xu, J. B. Weaver, D. M. Healy, Jr., and J. Lu, "Wavelet transform domain filters: A spatially selective noise filtration technique," *IEEE Trans. Image Proc.* **3**, pp. 747–758, Nov. 1994.
2. D. L. Donoho and I. M. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika* **81**, pp. 425–455, 1994.
3. D. L. Donoho, "De-noising by soft-thresholding," *IEEE Trans. Inform. Th.* **41**, pp. 613–627, May 1995.
4. R. R. Coifman and D. L. Donoho, "Translation-invariant de-noising," in *Wavelets and Statistics*, A. Antoniadis and G. Oppenheim, eds., *Springer Lecture Notes in Statistics* **103**, pp. 125–150, Springer-Verlag, (New York), 1995.
5. A. K. Fletcher, K. Ramchandran, and V. K. Goyal, "Wavelet denoising by recursive cycle spinning," in *Proc. IEEE Int. Conf. Image Proc.*, **2**, pp. 873–876, (Rochester, NY), Sept. 2002.
6. B. E. Usevitch, "A tutorial on modern lossy wavelet image compression: Foundations of JPEG 2000," *IEEE Sig. Proc. Mag.* **18**, pp. 22–35, Sept. 2001.
7. M. Vetterli, "Wavelets, approximation, and compression," *IEEE Sig. Proc. Mag.* **18**, pp. 59–73, Sept. 2001.
8. P. L. Dragotti and M. Vetterli, "Wavelet transform footprints: Catching singularities for compression and denoising," in *Proc. IEEE Int. Conf. Image Proc.*, **2**, pp. 363–366, (Vancouver), Sept. 2000.
9. P. L. Dragotti and M. Vetterli, "Wavelet footprints: Theory, algorithms, and applications," *IEEE Trans. Signal Proc.* **51**, pp. 1306–1323, May 2003.
10. S. G. Chang, B. Yu, and M. Vetterli, "Spatially adaptive wavelet thresholding with context modeling for image denoising," *IEEE Trans. Image Proc.* **9**, pp. 1522–1531, Sept. 2000.
11. M. Hansen and B. Yu, "Wavelet thresholding via MDL for natural images," *IEEE Trans. Inform. Th.* **46**, pp. 1778–1788, Aug. 2000.
12. B. K. Natarajan, "Filtering random noise from deterministic signals via data compression," *IEEE Trans. Signal Proc.* **43**, pp. 2595–2605, Nov. 1995.
13. S. G. Chang, B. Yu, and M. Vetterli, "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. Image Proc.* **9**, pp. 1532–1546, Sept. 2000.
14. S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, second ed., 1999.
15. D. L. Donoho, M. Vetterli, R. A. DeVore, and I. Daubechies, "Data compression and harmonic analysis," *IEEE Trans. Inform. Th.* **44**, pp. 2435–2476, Oct. 1998.
16. "Lena." Photograph of Lena Sjöblom from Playboy Magazine, November 1972, scanned and modestly cropped at the University of Southern California.<sup>17</sup> This is the most common data set in image processing. An eight-bit,  $512 \times 512$  grayscale version is used.
17. D. C. Munson, Jr., "A note on Lena," *IEEE Trans. Image Proc.* **5**, p. 3, Jan. 1996.