

# WAVELET-BASED DENOISING BY CUSTOMIZED THRESHOLDING

Byung-Jun Yoon and P. P. Vaidyanathan

Dept. of Electrical Engineering  
California Institute of Technology, Pasadena, CA 91125, USA  
E-mail: bjyoon@caltech.edu, ppvnath@systems.caltech.edu

## ABSTRACT

The problem of estimating a signal that is corrupted by additive noise has been of interest to many researchers for practical as well as theoretical reasons. Many of the traditional denoising methods have been using linear methods such as the Wiener filtering. Recently, nonlinear methods, especially those based on wavelets have become increasingly popular, due to a number of advantages over the linear methods. It has been shown that wavelet-thresholding has near-optimal properties in the minimax sense, and guarantees better rate of convergence, despite its simplicity. Even though much work has been done in the field of wavelet-thresholding, most of it was focused on statistical modeling of the wavelet coefficients and the optimal choice of the thresholds. In this paper, we propose a custom thresholding function which can improve the denoised results significantly. Simulation results are given to demonstrate the advantage of the new thresholding function.

## 1. INTRODUCTION

Estimating a signal that is corrupted by additive noise has been of interest to many researchers for practical as well as theoretical reasons. The problem is to recover the original signal from the noisy data. We want the recovered signal to be as close as possible to the original signal, retaining most of its important properties (e.g. smoothness). Traditional denoising schemes are based on linear methods, where the most common choice is the Wiener filtering. Recently, nonlinear methods, especially those based on wavelets have become increasingly popular.

One of the earliest papers in the field of wavelet-based denoising may be that of Weaver, et. al. [1]. In this pioneering work, they proposed a new method for filtering noise from MR (Magnetic Resonance) images based on the so-called hard-thresholding scheme. They showed that by using wavelet-thresholding, the noise could be significantly reduced without reducing the edge sharpness [1]. While Weaver, et al. demonstrated the advantages of the wavelet denoising scheme mainly based on experimental results, Donoho and Johnstone proved several important theoretical results on wavelet thresholding, or wavelet shrinkage [2, 3]. They showed that wavelet shrinkage has many excellent properties, such as near optimality in minimax sense, and a better rate of convergence [2, 3]. DeVore and Lucier have also arrived at the wavelet thresholding concept, starting from their independent work on variational problems [4]. In particular, they were interested in finding an approximation  $\tilde{f}$  to a given function  $f$  on a finite domain  $I$  that will balance the smoothness of  $\tilde{f}$  and the closeness to the original function  $f$ . In order to find such  $\tilde{f}$  one tries to

minimize

$$\|f - g\|_{L_2(I)}^2 + \lambda \|g\|_Y \quad (1)$$

over all  $g$  where  $Y$  is a space that measures the smoothness of the approximations  $g$  [4]. The function that minimizes (1) is taken to be  $\tilde{f}$ . The positive parameter  $\lambda$  balances the smoothness of  $\tilde{f}$  and the closeness to  $f$ . If  $\lambda$  is small, the approximation  $\tilde{f}$  will become closer to  $f$ , while if it is large, the resulting  $\tilde{f}$  will be smoother. Interestingly enough, DeVore and Lucier showed that when we consider the Besov space  $Y = B_\tau^\beta(L_\tau(I))$ , an approximate minimizer of (1) can be found by hard-thresholding the wavelet coefficients<sup>1</sup> [4]. In their later work, it is also shown that if we take  $Y = B_1^1(L_1(I))$ , then the exact minimizer of (1) can be found using the wavelet-shrinkage [5].

Besides wavelet-thresholding, many other approaches have been suggested as well. For example, wavelet-based denoising using Hidden Markov Trees [6], which was initially proposed by Crouse, et. al. has been quite successful, and it gave rise to a number of other HMT-based schemes. They tried to model the dependencies among adjacent wavelet coefficients using the HMT, and used the minimum mean-squared error (MMSE)-like estimators for suppressing the noise.

Even though much work has been done in the field of wavelet-thresholding, most of it was focused on the statistical modeling of wavelet coefficients for a certain class of signals (e.g. natural images), and the optimal choice of the threshold values. In this paper, we propose a new thresholding function that can take the place of the traditional thresholding functions, such as soft-thresholding and hard-thresholding. We will demonstrate that the custom thresholding function outperforms the traditional ones, improving the denoised results significantly. Simulation results are given where appropriate, which show the advantage of the proposed scheme.

## 2. WAVELET THRESHOLDING

Let us consider a signal  $x_i$ , which is corrupted by additive i.i.d. Gaussian random noise  $z_i \sim N(0, \sigma^2)$  as follows.

$$y_i = x_i + z_i \quad (i = 0, 1, \dots, N-1) \quad (2)$$

From this noisy signal  $y_i$ , we want to find an approximation  $\tilde{x}_i$  to the original  $x_i$ , that minimizes the mean squared error

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|^2 \triangleq \frac{1}{N} \sum_{i=0}^{N-1} |x_i - \tilde{x}_i|^2 \quad (3)$$

<sup>1</sup>If we consider the Sobolev space  $Y = W^\beta(L_2(I))$ , the approximate minimizer of (1) is obtained from a linear algorithm [4]

Work supported in part by the ONR grant N00014-99-1-1002, USA.

where  $\mathbf{x} = [x_0 \cdots x_{N-1}]^T$  and  $\tilde{\mathbf{x}} = [\tilde{x}_0 \cdots \tilde{x}_{N-1}]^T$ . Let  $\mathcal{W}$  be an orthogonal wavelet transformation. Then (2) can be written as

$$d_j = c_j + \epsilon_j \quad (4)$$

with  $\mathbf{d} = \mathcal{W}\mathbf{y}$ ,  $\mathbf{c} = \mathcal{W}\mathbf{x}$  and  $\epsilon = \mathcal{W}\mathbf{z}$ . Since  $\mathcal{W}$  is an orthogonal transform,  $\epsilon_j$  are also i.i.d. Gaussian random variables with  $\epsilon_j \sim N(0, \sigma^2)$ . Now, let  $T(\cdot)$  be a wavelet thresholding function. Then the wavelet thresholding based denoising schemes can be expressed as follows.

$$\tilde{\mathbf{x}} = \mathcal{W}^{-1}(T(\mathcal{W}\mathbf{y})) \quad (5)$$

We first take the wavelet transformation of the noisy signal and pass it through the thresholding function  $T(\cdot)$ . The output is then inverse wavelet transformed to obtain the estimate  $\tilde{\mathbf{x}}$ .

The most common choices for  $T(\cdot)$  are the hard-thresholding function and the soft-thresholding function (which is also known as the wavelet shrinkage function). The hard-thresholding function chooses all wavelet coefficients that are greater than the given threshold  $\lambda$  and sets the others to zero.

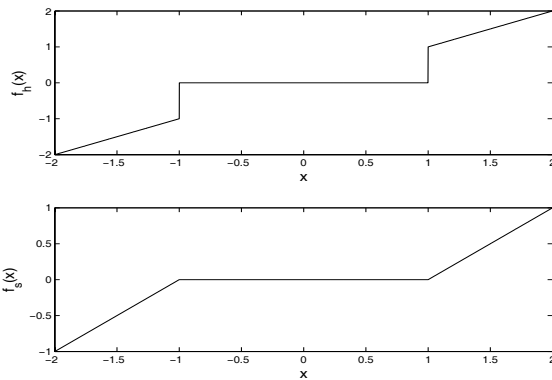
$$f_h(x) = \begin{cases} x & \text{if } |x| \geq \lambda \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

The threshold  $\lambda$  is chosen according to the signal energy and the noise variance  $\sigma^2$ . If a wavelet coefficient is greater than  $\lambda$ , we assume that it is significant and attribute it to the original signal. Otherwise, we consider it to be due to the additive noise and discard the value. The soft-thresholding function has a somewhat different rule from the hard-thresholding function. It shrinks the wavelet coefficients by  $\lambda$  towards zero, which is the reason why it is also called the wavelet shrinkage function.

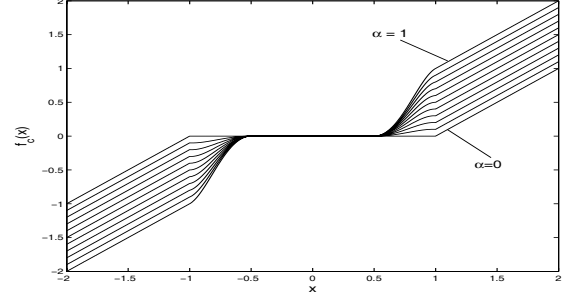
$$f_s(x) = \begin{cases} x - \lambda & \text{if } x \geq \lambda \\ 0 & \text{if } |x| < \lambda \\ x + \lambda & \text{if } x \leq -\lambda \end{cases} \quad (7)$$

These functions are illustrated in Fig. 1 for  $\lambda = 1$ .

Note that the hard-thresholding function is discontinuous at  $|x| = \lambda$ . Due to this discontinuity at the threshold, the hard-thresholding function is known to yield abrupt artifacts in the denoised signal, especially when the noise level is significant [7]. Moreover, Chang, et. al. [7] conclude from their simulation results, that the optimal soft-thresholding estimator yields a smaller



**Fig. 1.** Top: Hard-thresholding function. Bottom: Soft-thresholding function



**Fig. 2.** Custom thresholding function for various values of  $\alpha$ .

estimation error than the optimal hard-thresholding estimator. For this reason, soft-thresholding is generally preferred to hard-thresholding. However, for some class of signals, we could see that hard-thresholding results in superior estimates to that of soft-thresholding (which will be shown later), despite some of its disadvantages.

### 3. THE CUSTOM THRESHOLDING FUNCTION

This motivates us to introduce a new thresholding function that is continuous around the threshold, and which can be adapted to the characteristics of the input signal. Based on extensive experiments, we could see that soft-thresholding outperforms hard-thresholding in general. However, there were also cases where hard-thresholding yielded a much superior result, and in those cases the quality of the estimate could be improved by using a custom thresholding function which is similar to the hard-thresholding function but with a smooth transition around the threshold  $\lambda$ . Based on these observations, we defined a new custom thresholding function as follows

$$f_c(x) = \begin{cases} x - \text{sgn}(x)(1 - \alpha)\lambda & \text{if } |x| \geq \lambda \\ 0 & \text{if } |x| \leq \gamma \\ \alpha \left( \frac{|x| - \gamma}{\lambda - \gamma} \right)^2 \{ (\alpha - 3) \left( \frac{|x| - \gamma}{\lambda - \gamma} \right) + 4 - \alpha \} & \text{otherwise} \end{cases} \quad (8)$$

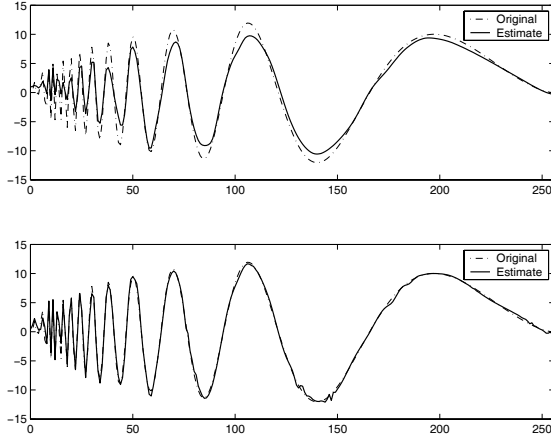
where  $0 < \gamma < \lambda$  and  $0 \leq \alpha \leq 1$ . This idea is similar to that of the *semisoft* or *firm shrinkage* proposed by Gao and Bruce [8], and the *non-negative garrote* thresholding function suggested by Gao [9], in the sense that they are continuous at  $\lambda$  and can be adapted to the signal characteristics. In our definition of  $f_c(x)$ ,  $\gamma$  is the cut-off value, below which the wavelet coefficients are set zero, and  $\alpha$  is the parameter that decides the shape of the thresholding function  $f_c(x)$ . Fig. 2 depicts  $f_c(x)$  for various  $\alpha$ 's, when  $\lambda = 1$  and  $\gamma = \lambda/2$ . This function can be viewed as the linear combination of the hard-thresholding function and the soft-thresholding function  $\alpha \cdot f_h(x) + (1 - \alpha) \cdot f_s(x)$  that is made continuous around the threshold  $\lambda$ . Note that,

$$\lim_{\alpha \rightarrow 0} f_c(x) = f_s(x) \quad \text{and} \quad \lim_{\alpha \rightarrow 1, \gamma \rightarrow \lambda} f_c(x) = f_h(x)$$

which shows that the custom thresholding function can be adapted to both the soft- and hard-thresholding function.

### 4. ADVANTAGES OF CUSTOM THRESHOLDING

In this section, we demonstrate the advantages of the custom thresholding function. In section 4.1, we show that the performance of VisuShrink [2] can be significantly improved by replacing the soft-thresholding function by the proposed function. Then, we compare



**Fig. 3.** Top: Estimate obtained using *VisuShrink*. Bottom: Estimate when using the custom thresholding function.

the optimal bounds when using a universal threshold, for different thresholding functions in section 4.2. Finally, in section 4.3, we compare the optimal bounds for each thresholding schemes, when the thresholds are chosen optimally at each scale.

#### 4.1. Improvement on VisuShrink

In their landmark paper [2], Donoho and Johnstone proposed a simple yet powerful wavelet-based denoising scheme called *VisuShrink*. This method uses a single universal threshold for all scales, and it is shown that *VisuShrink* yields estimates which are near-optimal in minimax sense [2]. The resulting estimate is very smooth with a pleasant visual appearance. However, it is known that *VisuShrink* tends to oversmooth the signal, thereby losing some details (e.g. sharp edges) of the original signal which can result in an increased estimation error. This is shown in Fig. 3 (Top). The noisy signal is decomposed into 8 levels using the Daubechies asymmetric wavelet with 8 vanishing moments [10]. The wavelet coefficients at the coarsest scale are left intact, while the coefficients at all the other scales are thresholded via soft-thresholding with the universal threshold  $\lambda = \sigma\sqrt{2\log N}$ , where  $\sigma^2$  is the noise variance and  $N$  is the length of the signal. Although the resulting estimate is smooth, we can see the discrepancy between the estimate and the original signal.

This situation can be significantly improved by replacing the soft-thresholding function by the custom thresholding function, as shown in Fig. 3 (Bottom). We used the same threshold as before with  $\alpha = 1$  and  $\gamma = \lambda/2$ . Note that the estimate follows the original signal very closely, although it is not as smooth as when using soft-thresholding. In order to evaluate the performance of different thresholding functions, we repeated the experiment above, for different input signals and various random seeds. The estimation errors were computed using (3) and averaged over 10 runs. The simulation results are summarized in Table. 1. It can be seen that custom thresholding yields better estimates with considerably smaller estimation error than the traditional thresholding schemes.

#### 4.2. Optimal Universal Thresholding

Although we have seen in section 4.1 that custom thresholding outperformed the others, the result might have been due to the

Signal	Mean Squared Error		
	Soft	Hard	Custom
Blocks	4.0387	1.2375	0.8523
Bumps	3.4158	1.1655	0.8021
HeavySine	1.0483	0.3911	0.3538
Doppler	2.3867	0.4776	0.3836

**Table 1.** Estimation error for different thresholding functions (averaged over 10 runs, SNR=7,  $\sigma = 1$ ).

specific choice of the threshold, i.e.  $\lambda = \sigma\sqrt{2\log N}$ . Therefore, it would be interesting to ask how the optimal performance of custom thresholding would compare to that of soft- and hard-thresholding, when using a single universal threshold. In order to answer this question, we tried to find the optimal universal threshold for each thresholding function, with the assumption that the original signal was known. The optimal threshold was found using exhaustive search. For the custom thresholding function, we also searched for the optimal  $\alpha$  that resulted in the minimum estimation error. The simulation was repeated for 5 times with the signal-to-noise-ratio set to SNR=7,  $\sigma = 1$  and  $\gamma = \lambda/2$ . The resulting estimation errors have been averaged, which are summarized in Table 2.

Signal	Mean Squared Error			
	Soft	Hard	Custom	(opt. $\alpha$ )
Blocks	0.8536	0.9906	0.8516	(0.31)
Bumps	0.8348	0.8651	0.79610	(0.94)
HeavySine	0.3474	0.3278	0.2620	(0.94)
Doppler	0.5151	0.4371	0.3477	(0.97)

**Table 2.** Optimal performance of each thresholding function when using a universal threshold (averaged over 5 runs, SNR=7,  $\sigma = 1$ ).

From Table 2, we can see that if the universal threshold is chosen optimally, the quality of the estimate can be significantly improved. When we compare this result with that in Table 1, it can be seen that *VisuShrink* yields a much larger estimation error, which is due to its tendency to oversmooth the given signal. This result also shows that the optimal performance of soft-thresholding is superior to that of hard-thresholding for many kinds of signals. However, when we compare the estimation errors for the “Doppler” signal, we can see that hard-thresholding is much better than soft-thresholding, which motivates us to use a new thresholding scheme that can be adapted to the signal characteristics. As expected, custom thresholding yielded the best estimates for all signals, with  $\alpha$  chosen appropriately for each signal.

#### 4.3. Optimal Scale Adaptive Thresholding

The denoised results can be further improved, if the thresholds at each scale are chosen optimally. State-of-the-art wavelet-based denoising algorithms such as *SureShrink* [11] and *BayesShrink* [7] are based on this approach, where the threshold levels are chosen in a subband-adaptive manner. How much would the quality of the estimates be improved if we use custom thresholding instead of the traditional thresholding methods? To compare the optimal performance of each thresholding function, we searched for the optimal threshold at each subband level, based on the assumption that we know the original signal. For hard-thresholding and soft-thresholding, they are respectively what Chang, et. al.

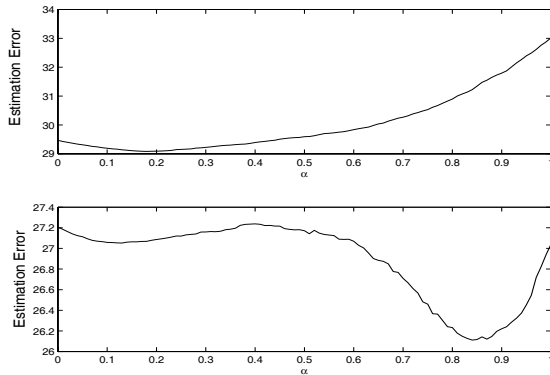
Signal	Mean Squared Error			
	Soft	Hard	Custom	(opt. $\alpha$ )
Blocks	0.7707	0.9141	0.7668	(0.24)
Bumps	0.7167	0.8070	0.6966	(0.78)
HeavySine	0.1402	0.1594	0.1376	(0.60)
Doppler	0.3941	0.3205	0.2893	(0.94)

**Table 3.** Optimal performance of each thresholding function when the threshold at each subband level is chosen adaptively (averaged over 5 runs, SNR=7,  $\sigma = 1$ ).

called the *OracleThresh* and *OracleShrink*, in [7]. The signal-to-noise ratio was again set to SNR=7 where  $\sigma = 1$ , and  $\gamma$  was set to  $\gamma = \lambda/2$ . The experiment was repeated 5 times with different random seeds. The resulting estimation errors are shown in Table 3. Clearly, the average estimation error could be further minimized by choosing the threshold at each scale optimally. Notice that the custom thresholding function resulted in the best estimates also in this case.

## 5. ADAPTING TO SIGNAL CHARACTERISTICS

The simulation results in the previous section show that by using custom thresholding, we can improve the quality of the estimates considerably. However, in order to obtain the best results possible, the parameter  $\alpha$  has to be chosen appropriately. For example, for piecewise constant signals such as *Blocks* that have many discontinuities, small value of  $\alpha$  in the range of 0.1~0.3 yielded the smallest estimation error. On the other hand, for continuous signals with a large frequency variation such as *Doppler*, relatively large values of  $\alpha$  (0.85~0.95) resulted in the best performance. This shows that for different class of signals, the value of  $\alpha$  has to be chosen differently. One way to make a good choice of  $\alpha$  is to optimize it for a class of signals, which are in some sense similar to each other. For example, when we are interested in denoising noisy images, we may optimize  $\alpha$  for cartoons (which mainly consist of edges), or natural images of trees and leaves, etc. The  $\alpha$  optimized for a set of images that belong to the same class, can be used for denoising another image that can be categorized in the same class. In order to demonstrate the idea, we optimized alpha for two different class of signals. The first set consisted of signals close to *Blocks*, and the second set consisted of signals which were



**Fig. 4.** Average estimation error corresponding to different values of  $\alpha$  for two classes of signals. Top: *Blocks*. Bottom: *Doppler*.

similar to *Doppler*. There were four signals in each set, and for every signal we ran the simulation for 15 different random seeds. For any signal in a given set, the estimation error corresponding to the value of  $\alpha$  showed similar tendency to those of the others in the same set. The estimation errors have been averaged for each class of signals which can be seen in Fig. 4. Based on this result, we can see that  $\alpha = 0.18$  and  $\alpha = 0.84$  will result in the best estimate for denoising *Blocks* and *Doppler*-like signals, respectively.

## 6. CONCLUDING REMARKS

In this paper, we proposed a new thresholding scheme that can considerably improve the performance of the well-known wavelet-based denoising algorithms. The custom thresholding function can be adapted to the characteristics of the given signal, resulting in a smaller estimation error. It was shown that the custom thresholding proposed in this paper outperforms the traditional soft and hard-thresholding schemes. Topics for future research include algorithms for selecting optimal thresholds and  $\alpha$  based on the characteristics of the observed signal, and application of custom-thresholding to many of the existing algorithms.

## 7. REFERENCES

- [1] Weaver, J. B., Yansun, X., Healy, D. M. Jr., and Cromwell, L. D., "Filtering noise from images with wavelet transforms", *Magnetic Resonance in Medicine*, 24 (1991), 288-295.
- [2] Donoho, D. L. and Johnstone, I. M., "Ideal spatial adaptation via wavelet shrinkage", *Biometrika*, vol. 81, pp. 425-455, 1994.
- [3] Donoho, D. L., "De-noising by soft-thresholding", *IEEE Trans. Info. Theory*, vol. 41, pp. 613-627, May 1995.
- [4] DeVore, R. A. and Lucier, B. J., "Fast wavelet techniques for near-optimal image processing", *IEEE Military Communications Conference Record*, San Diego, Oct. 11-14, 1992, vol. 3, pp. 1129-1135.
- [5] Chambolle, A., DeVore, R. A., Lee, N.-Y., and Lucier, B. J., "Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage", *IEEE Trans. Image Proc.*, vol. 7, pp. 319-335, Mar. 1998.
- [6] Crouse, M. S., Nowak, R. D., Baraniuk, R. G., "Wavelet-based signal processing using hidden Markov models", *IEEE Trans. Signal Proc.*, vol. 46, pp. 886-902, Apr. 1998.
- [7] Chang, S. G., Yu, B., and Vetterli, M., "Adaptive wavelet thresholding for image denoising and compression", *IEEE Trans. Image Proc.*, vol. 9, pp. 1532-1546, Sep. 2000.
- [8] Gao, H.-Y., and Bruce, A. G., "WaveShrink with firm shrinkage", *Technical Report 39*, StatSci Division of MathSoft, Inc. 1996.
- [9] Gao, H.-Y., "Wavelet shrinkage denoising using the non-negative garrote", *J. Comput. Graph. Statist.*, 7 (4), pp. 469-488, 1998.
- [10] Daubechies, I., *Ten Lectures on Wavelets*, Proc. CMBS-NSF Regional Conference Series in Applied Mathematics, vol. 61, Philadelphia, PA:SIAM, 1992.
- [11] Donoho, D. L. and Johnstone, I. M., "Adapting to unknown smoothness via wavelet shrinkage", *Journal of American Stat. Assoc.*, vol. 90, no. 432, pp. 1200-1224, Dec. 1995.