# Vector Wavelet Thresholding for Vector Field Denoising


Michel A. Westenberg* Thomas Ertl†

Institute for Visualization and Interactive Systems, University of Stuttgart
Universitätsstr. 38, 70569 Stuttgart, Germany



## ABSTRACT

Noise reduction is an important preprocessing step for many visualization techniques that make use of feature extraction. We propose a method for denoising 2-D vector fields that are corrupted by additive noise. The method is based on the vector wavelet transform and wavelet coefficient thresholding. We compare our wavelet-based denoising method with Gaussian filtering, and test the effect of these methods on the signal-to-noise ratio (SNR) of the vector fields before and after denoising. We also study the effect on relevant details for visualization, such as vortex measures. The results show that for low SNR, Gaussian filtering with large kernels has a somewhat higher performance than the wavelet-based method in terms of SNR. For larger SNR, the wavelet-based method outperforms Gaussian filtering. This is mostly due to the fact that Gaussian filtering tends to remove small details, which are preserved by the wavelet-based method.



**CR Categories:** I.4.3 [Image Processing and Computer Vision]: Enhancement—Filtering; G.1.2 [Numerical Analysis]: Approximation—Wavelets and fractals

**Keywords:** denoising, flow visualization, wavelets


## 1 INTRODUCTION

Data acquired by physical measurements are often corrupted by noise. The goal of denoising is to suppress the noise while retaining the relevant details. A commonly used denoising method is to smooth the data by Gaussian filtering. However, this does not only affect the noise, but also may destroy detailed features in the data.

About one decade ago, Donoho introduced a nonlinear signal denoising technique based on wavelet thresholding [3]. Since then, much work has been done in this area, and many wavelet-based denoising methods have been proposed for scalar data. The purpose of this paper is to report on work in progress on denoising 2-D vector data by thresholding wavelet coefficients that are obtained by a so-called vector wavelet transform [7]. This is an extension of the scalar wavelet transform that deals with vector data. It is important to note that the vector wavelet transform is *not* just a component-wise scalar wavelet transform.

## 2 VECTOR WAVELETS

Vector wavelet transforms are based on so-called multiwavelets, which consist of multiple scaling functions and wavelet functions rather than a single pair [7]. In principle, multiwavelets can be used directly to construct a vector wavelet transform, however, it turns out that the performance for signal processing applications is poor [4]. The source of the problem is the fact that constant input signals (all vectors point in the same direction) are not preserved when performing a reconstruction from wavelet approximation coefficients only. Intuitively, one would expect a constant signal, however, most multiwavelets introduce an oscillatory distortion. This is rather disturbing, as most denoising and compression schemes tend to preserve the approximation coefficients and discard detail coefficients. One possible solution to this problem was proposed, and it involves appropriate multiwavelet design criteria for vector data [4].

We refer the readers to the papers [6] and [7] for details of the vector wavelet transform. It is relatively straightforward to extend the pyramid algorithm of Mallat to compute the vector wavelet transform. The extension to 2-D is done in the standard way by applying the 1-D transform to the rows and columns. The wavelet transform for $M$ levels then results in approximation coefficients $c^M$ and three sets of detail coefficients $d^{j,1}$, $d^{j,2}$, and $d^{j,3}$, $j = 1,\ldots,M$. Note that these coefficients are now vectors and not scalars.

## 3 WAVELET-BASED DENOISING

We assume that the noise is *additive*, and has a normal distribution with zero mean and variance $\sigma_n^2$. Wavelet-based denoising methods then work in three steps. (1) Compute an $M$-level wavelet transform. (2) Modify the detail coefficients $d^{j,1}$, $d^{j,2}$, and $d^{j,3}$, $j = 1,\ldots,M$, by hard or soft thresholding. Both methods set the coefficients below the threshold $T$ to zero. Soft thresholding additionally reduces the amplitude of the other coefficients by $T$, a procedure also called shrinkage. The approximation coefficients $c^M$ are not modified. (3) Compute the inverse wavelet transform.

Many methods have been proposed to select a good threshold $T$, a number of which are contained in the WaveLab software [1]. In this paper, we use a method called BayesShrink [2], which computes a data-driven estimate of $T$ for each set of detail coefficients independently. As the original method deals only with scalar data, we made adaptations such that it can deal with vector data. In our method, the threshold selection is based on the vector magnitude, and our soft thresholding extension shrinks the vector magnitudes. This means that thresholding does not affect the direction of the vectors, but only their lengths.

## 4 RESULTS

We conducted a series of experiments in which noise of known standard deviation was added to a slice ($490 \times 490$) of a hurricane data set. The resulting noisy vector fields had signal-to-noise ratios (SNR) of $\{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$. The SNR is expressed in dB and computed from the standard deviations $\sigma$ (data) and $\sigma_n$ (noise) as

$$\mathrm{SNR} = 20 \log_{10} \frac{\sigma}{\sigma_n}.$$

We applied our wavelet-based denoising method to the noisy vector fields, using the biorthogonal OBSA 7-5 multiwavelets [4], with filter lengths 7 and 5 for the low-pass and high-pass filter, respectively. The depth of the wavelet decomposition was set to three. We also performed filtering with Gaussian kernels of various widths. The width of the Gaussian kernel is described by its width


*e-mail: Michel.Westenberg@informatik.uni-stuttgart.de

†e-mail:Thomas.Ertl@informatik.uni-stuttgart.de

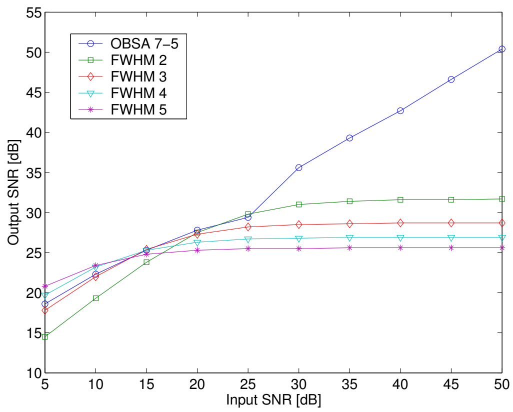


Figure 1: Performance of wavelet-based denoising (OBSA) and Gaussian filtering (FWHM).

in pixels at half of the maximum of the height of the Gaussian, a measure called Full Width at Half Maximum (FWHM). For example, for FWHM $= 5$, the total kernel width from $-3\sigma$ to $3\sigma$ is 13 pixels.

Figure 1 shows the output SNR plotted against the input SNR. The plot shows that Gaussian filtering with large kernels performs slightly better than the wavelet-based method for very low SNRs. For an SNR between 15 and 20 dB, both methods show similar performance. For larger SNRs, the Gaussian filtering method smooths to strongly, and for SNRs above 30 dB, the output SNR is actually lower than the input SNR. The wavelet-based method does not have this problem, and the output SNR is in the worst case equal to the input SNR. We also performed the experiment (results not included) with the OBSA 5-3 wavelet, and its performance is similar to the performance of the OBSA 7-5 wavelet. However, the performance for low SNR is worse, which can be explained by the fact that the OBSA 5-3 wavelet is not as smooth as the OBSA 7-5 wavelet.

Figure 2 shows color-encoded (blue to red) $\lambda_2$ values [5] in the range $[0.05, 1.0]$ for some of the generated noisy vector field input data sets, the best results obtained by Gaussian filtering, and the results of wavelet-based denoising. These images confirm that for low SNR, Gaussian filtering produces a somewhat better result. For the high SNR input (almost noise free), Gaussian filtering misses details, especially in the areas with fine detail. An example of loss of detail is shown in Fig. 3, in which a small vertical structure is visible in the original data (Fig. 3(a)), which is lost by Gaussian filtering (Fig. 3(b)), but retained by our wavelet-based method (Fig. 3(c)).

## 5 Discussion

We have proposed a denoising method for 2-D vector fields that are corrupted by additive noise. The method is an extension of scalar wavelet-based denoising techniques to vector data, and makes use of a vector wavelet transform. Currently, we are working on an extension to vectors with three components. This is challenging, since most research has focussed on multiwavelet design for vectors of only two components. This extension would open up the possibility of denoising 3-D vector fields, and could result in a promising denoising method for diffusion-tensor MRI volumetric data.

## Acknowledgement

This research was funded by the project SFB 382 of the German Research Foundation (DFG).

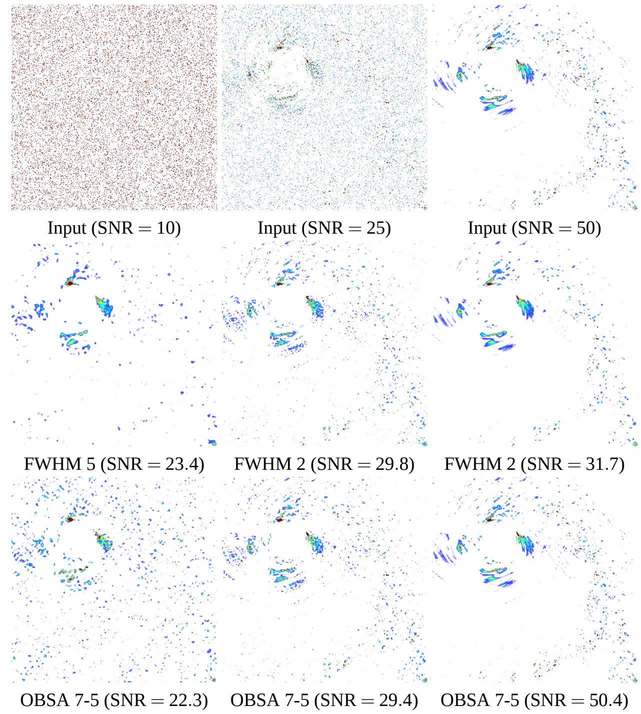


Figure 2: All images show color-encoded $\lambda_2$ values in a selected range. Top row: input data with increasing SNR from left to right. The second and third row show denoised versions of these input data sets. Second row: Gaussian filtering with the filter with the best performance. Third row: wavelet-based denoising.

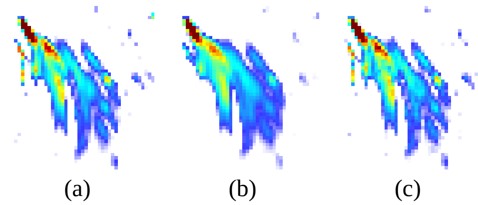


Figure 3: Detail images of a larger coherent feature in the data, selected from the larger structures in the upper left quadrants of the images in the third column of Fig. 2. (a) Original data. (b) Gaussian filtering. (c) Wavelet-based denoising. Note how the small vertical structure on the left disappears with Gaussian filtering.

## References


[1] J. B. Buckheit and D. L. Donoho. WaveLab and reproducible research. Technical Report 474, Dept. of Statistics, Stanford University, 1995.

[2] S. G. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Trans. Image Processing*, 9(9):1532–1546, 2000.

[3] D. L. Donoho. De-noising by soft thresholding. *IEEE Trans. Information Theory*, 41:613–627, May 1995.

[4] J. E. Fowler and L. Hua. Wavelet transforms for vector fields using omnidirectionally balanced multiwavelets. *IEEE Trans. Signal Processing*, 50:3018–3027, 2002.

[5] J. Jeong and F. Hussain. On the identification of a vortex. *J. Fluid Mechanics*, 285:69–94, 1995.

[6] H. H. Tan, L.-X. Shen, and J. Y. Tham. New biorthogonal multiwavelets for image compression. *Signal Processing*, 79(1):45–65, 1999.

[7] X.-G. Xia and B. W. Suter. Vector-valued wavelets and vector filter banks. *IEEE Trans. Signal Processing*, 44(3):508–518, 1996.