Denoising 2-D Vector Fields by Vector Wavelet Thresholding

Michel A. Westenberg and Thomas Ertl
Institute for Visualization and Interactive Systems, University of Stuttgart
Universitätsstr. 38
70569 Stuttgart, Germany
{westenberg, ertl}@vis.uni-stuttgart.de

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, which transforms a vector input signal to wavelet coefficients that are also vectors. We introduce modifications to scalar wavelet coefficient thresholding for dealing with vector-valued coefficients. 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 compare our method with component-wise scalar wavelet thresholding. Furthermore, we use a vortex measure to study the performances of the methods for retaining relevant details for visualization. The results show that for very low SNR, Gaussian filtering with large kernels has a slightly better performance than the wavelet-based method in terms of SNR. For larger SNR, the wavelet-based method outperforms Gaussian filtering, because Gaussian filtering removes small details that are preserved by the wavelet-based method. Component-wise denoising has a lower performance than our method.

Keywords

Denoising, flow visualization, multiwavelets, wavelets.

1. INTRODUCTION

Data acquired by physical measurements are often corrupted by noise. In fluid mechanics, such data may be obtained by, for instance, particle image velocimetry (PIV). This is a technique that provides global velocity measurements by recording the position over time of small tracer particles inserted into the flow [Pra00]. Noise in the recorded images is a source of errors in PIV measurements, and it can result in spurious vectors or global noise in the reconstructed vector field. The spurious vectors can be repaired by averaging or median filtering, however, the global noise requires a different removal method.

The Journal of WSCG, Vol.13, ISSN 1213-6964
WSCG 2005, January 31-February 4, 2005
Plzen, Czech Republic.

The process of removing noise is called denoising, and its goal is to suppress the noise while retaining the relevant details. A commonly used denoising method is smoothing by Gaussian filtering. However, this does not only affect the noise, but also may destroy small features in the data.

Better performance is usually obtained by a smoothing technique that is edge-preserving, such as anisotropic diffusion [Per90]. This technique has been extended for smoothing orientation fields [Per98], but it has not been tested in a practical application, and has not been evaluated on directional fields. Another successful iterative method for image denoising is based on minimizing the total variation of the image subject to constraints that involve the noise statistics [Rud92]. This approach has been extended to vector-valued functions, and has been used for denoising color images [Blo98]. In a recent paper, this method was used for the reconstruction of flow velocity images acquired by magnetic resonance velocity imaging [Ng03]. Such images are used in the study of cardiovascular function by analyzing the blood flow patterns and their interaction with cardiovascular structure. Noise has detri-

mental effects on this analysis, and it is very important that features in the data are retained by the denoising method.

Another class of denoising methods is based on thresholding of wavelet coefficients, an idea introduced about one decade ago by Donoho [Don95]. Since then, much work has been done in this area, and many wavelet-based denoising methods have been proposed for scalar signals [Str01], natural images [Cha00, Sim96], and medical images [Pižˇ03, Win04], to name a few.

The purpose of this paper is to report on work in progress on denoising 2-D vector data that are corrupted by additive noise. Our method performs thresholding on wavelet coefficients that are obtained by a so-called vector wavelet transform [Xia96]. This is an extension of the scalar wavelet transform that deals with vector data, and it maps vector data to wavelet coefficients that are also vectors. It is important to note that the vector wavelet transform is different from a component-wise scalar wavelet transform, and that the mathematical foundation is based on multiwavelets. We introduce extensions to the scalar wavelet-based denoising technique, in order to be able to deal with the vector-valued coefficients.

The organization of this paper is as follows. Section 2 discusses the mathematical background of vector wavelets, and describes the algorithm to compute the vector wavelet transform efficiently. In Section 3 we briefly describe wavelet-based denoising of scalar data, and we introduce our modifications for dealing with vector data. Section 4 compares the results of vector wavelet-based denoising and Gaussian smoothing, and we perform an experiment with component-based scalar wavelet denoising. Finally, we draw conclusions in Section 5 and discuss future work.

2. VECTOR WAVELETS

The concept of a vector wavelet transform has existed for about a decade, and the theory follows scalar wavelet theory closely [Xia96]. Vector wavelet transforms are based on so-called multiwavelets, which expand a scalar function by several scaling functions and wavelet functions rather than by a single pair. In the following, we briefly describe multiwavelets, and we refer the readers to the papers [Tan99] and [Xia96] for full details.

2.1 Multiwavelets

A biorthogonal multiwavelet basis consists of a multiscaling function vector $\Phi(t) := [\phi_1(t), \ldots, \phi_r(t)]^{\mathrm{T}}$ and its dual $\widetilde{\Phi}(t) := [\widetilde{\phi}_1(t), \ldots, \widetilde{\phi}_r(t)]^{\mathrm{T}}$, with r an integer, and x^{T} denoting the transpose of x. Typically, $r = 2$ or $r = 3$ in practical applications with 2-D and 3-D vector fields, respectively. These multiscaling functions satisfy the two-scale dilation equations

$$\begin{aligned} \Phi(t) &= \sqrt{2}\sum_n H_n \Phi(2t-n), \\ \widetilde{\Phi}(t) &= \sqrt{2}\sum_n \widetilde{H}_n \widetilde{\Phi}(2t-n), \end{aligned} \tag{1}$$

in which H_n and $\widetilde{H}_n$ are real-valued $r \times r$ matrix sequences. The multiwavelet functions $\Psi(t)$ and $\widetilde{\Psi}(t)$ are associated with the multiscaling functions by the two-scale wavelet equations

$$\begin{aligned} \Psi(t) &= \sqrt{2}\sum_n G_n \Phi(2t-n), \\ \widetilde{\Psi}(t) &= \sqrt{2}\sum_n \widetilde{G}_n \widetilde{\Phi}(2t-n), \end{aligned} \tag{2}$$

in which G_n and $\widetilde{G}_n$ are also real-valued $r \times r$ matrix sequences.

The expansion of an input vector signal $f^{\mathrm{T}}(t)$ on a biorthogonal multiwavelet basis is given by

$$\begin{aligned} f^{\mathrm{T}}(t) &= \sum_k (c_k^M)^{\mathrm{T}} \Phi_{M,k}(t) + \sum_{j=1}^{M} \sum_k (d_k^j)^{\mathrm{T}} \Psi_{j,k}(t), \\ \Phi_{j,k}(t) &= 2^{-j/2} \Phi(2^{-j}t - k), \\ \Psi_{j,k}(t) &= 2^{-j/2} \Psi(2^{-j}t - k), \end{aligned} \tag{3}$$

where M denotes the depth of the decomposition. The coefficients c_k^M and d_k^j are called approximation coefficients and detail coefficients, respectively, as in the scalar case. Note that these coefficients are now $r \times 1$ column vectors.

2.2 Fast vector wavelet transform

Given coefficient sequences H_n, G_n, $\widetilde{H}_n$, and $\widetilde{G}_n$ that are $r \times r$ matrices, and which satisfy the perfect reconstruction conditions, we can compute the 1-D discrete vector wavelet transform of the input sequence c^0 by the pyramid algorithm of Mallat. The main difference with the scalar algorithm is that scalar multiplications are replaced by matrix-vector multiplications. The M-level wavelet decomposition computes the coefficients c_k^j and d_k^j as

$$c_k^j = \sum_n \widetilde{H}_{n-2k} c_n^{j-1} \qquad d_k^j = \sum_n \widetilde{G}_{n-2k} c_n^{j-1}. \tag{4}$$

Reconstruction is computed as

$$c_k^{j-1} = \sum_n H_{k-2n}^{\mathrm{T}} c_n^j + \sum_n G_{k-2n}^{\mathrm{T}} d_n^j. \tag{5}$$

The extension to a 2-D transform is done in the standard way by applying the 1-D transform to the rows

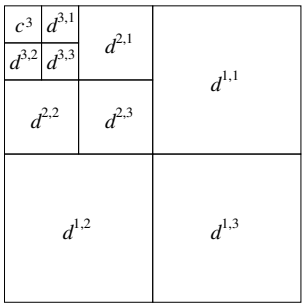


Figure 1. Coefficients of a three level 2-D (vector) wavelet transform.

and columns. The wavelet transform for M levels then results in approximation coefficients $c_{k,l}^M$ and three sets of detail coefficients $d_{k,l}^{j,\tau}$, $j = 1,\ldots,M$, $\tau = \{1,2,3\}$. The coefficients are ordered as shown in Fig. 1.

2.3 Filter coefficients

In principle, the filter coefficients of the multiwavelets available from the literature could be used for computing the vector wavelet transform. However, it turns out that the performance for vector signal processing applications is poor [Fow02]. The source of the problem lies in the fact that constant input signals are not preserved when performing a reconstruction from wavelet approximation coefficients only. Constant in the context of vector fields means that all vectors point in the same direction. Intuitively, one would expect a constant signal, however, most multiwavelets result in an oscillatory distortion. This means that the coefficients $c_{k,l}^M$ do not consist of a low resolution approximation of the original data. This is rather disturbing, as most denoising and compression schemes preserve the approximation coefficients and discard detail coefficients.

Fowler and Hua [Fow02] have proposed a scheme to design filter coefficients that define a multiwavelet basis that does not suffer from the problem mentioned above. The resulting wavelets are known by the name omnidirectionally balanced symmetric-antisymmetric (OBSA); part of this name refers to the constraints formulated for the construction process. In the remainder of this paper, we will use the OBSA 5-3 and OBSA 7-5 filters. The numbers denote the lengths of the coefficient sequences H_n and $\widetilde{H}_n$, respectively.

3. WAVELET-BASED DENOISING

We assume that the noise is *additive*, and has a normal distribution with zero mean and variance σ_n^2, denoted as $N(0,\sigma_n^2)$. Wavelet-based denoising methods in the 1-D scalar case then work in three steps. (1) Compute an M-level wavelet transform. (2) Modify the detail coefficients d_k^j, $j = 1,\ldots,M$, by a threshold function. The approximation coefficients c_k^M are not modified. (3) Compute the inverse wavelet transform. The extension to higher dimensions is straightforward.

There are two popular threshold functions in use: hard and soft thresholding. Both set the coefficients below the threshold T to zero. Hard thresholding retains the coefficients above the threshold unaltered. Soft thresholding, also called shrinkage, reduces the amplitude of the coefficients above T as follows

$$\eta_T(x) = \mathrm{sgn}(x)\cdot\max(|x|-T,0). \tag{6}$$

For image denoising, soft thresholding generally yields more visually pleasing results than hard thresholding, and it is therefore the preferred choice.

Many methods have been proposed to select a good threshold T, a number of which are contained in the WaveLab software [Buc95]. In this paper, we use a method called BayesShrink [Cha00], which computes a data-driven estimate of T for each set of detail coefficients $d_{k,l}^{j,\tau}$, $\tau = \{1,2,3\}$ independently. This method was proposed for image denoising, and it is based on the observation that the detail coefficients in a subband of a natural image can be characterized by a generalized Gaussian distribution (GGD) [Mal89, Sim96]. The probability density function is given by

$$p(x) = \left[\frac{\nu\eta(\nu,\sigma)}{2\Gamma(1/\nu)}\right] e^{-[\eta(\nu,\sigma)|x|]^\nu}, \tag{7}$$

with

$$\eta(\nu,\sigma) = \frac{1}{\sigma}\sqrt{\frac{\Gamma(3/\nu)}{\Gamma(1/\nu)}}, \tag{8}$$

where $\Gamma(x)$ denotes the gamma function. The shape parameter ν controls the exponential rate of decay. A Gaussian distribution is obtained by $\nu = 2$. The parameter σ is the standard deviation.

We have observed that the individual components of the vector detail coefficients also follow a GGD. Figure 2 shows parts of the histograms of the second-level vector detail coefficients $d_{k,l}^{2,1}$, $d_{k,l}^{2,2}$, and $d_{k,l}^{2,3}$ of a slice of a hurricane data set as an example. The top row shows the histograms of the first components of the vectors, and the bottom row shows the histograms of the second components. All these histograms can be qualitatively described by a GGD. It is therefore valid to use the BayesShrink method.

We can now describe our modifications to the scalar wavelet-based denoising scheme for dealing with vector data. Calculations that involve the absolute value of a scalar coefficient now use the vector magnitude of

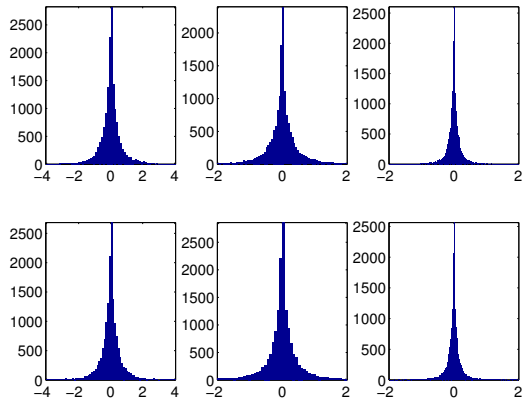


Figure 2. Histograms of vector wavelet detail coefficients at level 2 of an example data set. Histograms of the first vector components are on the top row and of the second components on the bottom row. From left to right are $d_{k,l}^{2,1}$, $d_{k,l}^{2,2}$, and $d_{k,l}^{2,3}$, respectively. All histograms can be described qualitatively by a generalized Gaussian distribution.

that coefficient. Furthermore, we define the variance σ^2 of an $N \times N$ vector field $v_{k,l}$ as

$$\sigma^2 = \frac{1}{rN^2} \sum_{k=1}^{N} \sum_{l=1}^{N} ||v_{k,l} - \bar{v}||^2, \tag{9}$$

where the average $\bar{v}$ is a vector that contains the component-wise averages of $v_{k,l}$, and $||\cdot||$ denotes the Euclidian norm. This definition includes a division by r, the number of components of the vectors, for the following reason. If a vector field contains only noise, i.e., each component contains noise distributed as normal $N(0, \sigma^2)$, the equation above will yield precisely σ^2.

The threshold is dependent on the variance $\hat{\sigma}_d^2$ of the coefficients $d_{k,l}^{j,\tau}$ under consideration and the global noise variance σ^2. If the noise characteristics of the data acquisition process are known, it may be possible to determine the global noise variance from that information. Alternatively, the global noise variance can be estimated from the detail coefficients $d_{k,l}^{1,3}$ by the robust median estimator [Cha00]:

$$\hat{\sigma} = \frac{\text{median}(|d_{k,l}^{1,3}|)}{0.6745}. \tag{10}$$

Finally, the threshold T is computed as

$$T = \frac{\hat{\sigma}^2}{\sqrt{\max(\hat{\sigma}_d^2 - \hat{\sigma}^2, 0)}}. \tag{11}$$

If the denominator in this equation becomes equal to zero, the threshold T becomes ∞, and all coefficients are assigned the zero vector.

For our method, we adapted the soft thresholding method such that it shrinks the vector magnitudes. We define the modified soft thresholding $\vec{\eta}_T(x)$ for a vector x as

$$\vec{\eta}_T(x) = x \cdot \frac{\max(|x| - T, 0)}{|x|}. \tag{12}$$

When $|x| = 0$, we set $\vec{\eta}_T(x) = [0,0]^{\mathrm{T}}$.

4. RESULTS

We conducted a series of experiments in which noise of known standard deviation was added to a slice (490×490) of a hurricane data set, consisting of 2-component velocity vectors, see Fig. 3(a). The resulting noisy vector fields had signal-to-noise ratios (SNR) of $\{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$. An example rendering of the vector magnitudes of a noisy vector field with SNR $= 10$ is shown in Fig. 3(b). The SNR is expressed in dB and computed from the standard deviations σ (data) and σ_n (noise) as

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

To provide some intuition, an SNR around 40 dB is considered acceptable in image processing.

We applied our wavelet-based denoising method to the resulting noisy vector fields, using the biorthogonal OBSA 5-3 and OBSA 7-5 multiwavelets. The depth of the wavelet decomposition was fixed to three. We also performed filtering with Gaussian kernels of various widths. The width of the Gaussian kernel is described by its width in pixels at half of the maximum of the height of the Gaussian, a measure called Full Width at Half Maximum (FWHM). For example, a Gaussian filter with FWHM $= 5$ contains 13 pixels when sampled between -3σ and 3σ. Filter values beyond 3σ are negligibly small, and are therefore not used.

Example renderings of the vector magnitudes of the results of both Gaussian filtering and our method are shown in Fig. 3(c) and Fig. 3(d), respectively. The noisy input vector data had SNR $= 10$ (Fig. 3(b)), a high noise level at which the standard deviation of the noise is about one-third the standard deviation of the data. Qualitatively, both output images look similar, although the Gaussian filtered data appears to be more smooth, due to the large filter kernel used. The performance of the methods is comparable, as they both yield similar output signal-to-noise ratios.

Figure 4 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

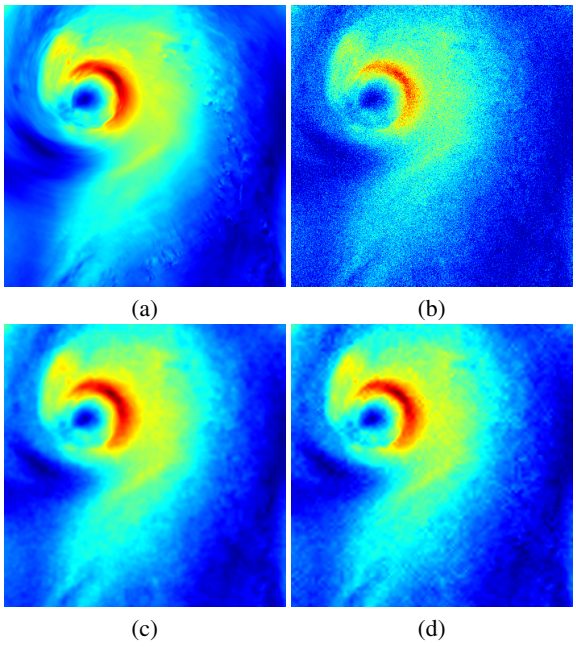


Figure 3. The images show color-encoded vector magnitudes; red corresponds to high velocities and dark blue to low velocities. (a) Noise-free test data. (b) Noisy test data with SNR $= 10$. (c) Result after denoising by Gaussian filtering with FWHM 5. The filtered data has SNR $= 23.4$. (d) Result after denoising by wavelet coefficient thresholding. The resulting data has SNR $= 22.3$.

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.

For comparison, we implemented component-based scalar wavelet-based denoising, i.e. we treated each component of the vector field as a scalar data set, and applied scalar denoising. We used a fourth-order B-spline wavelet [Chu92] as a basic wavelet. It is clear that component-wise denoising has a consistently lower performance than a vector-based approach for the wavelets we tested, see Fig. 4. We presume this is due to possibility of changing the orientation of a vector when its components are thresholded independently. A more extensive investigation is necessary to see if this is indeed the cause of the lower performance.

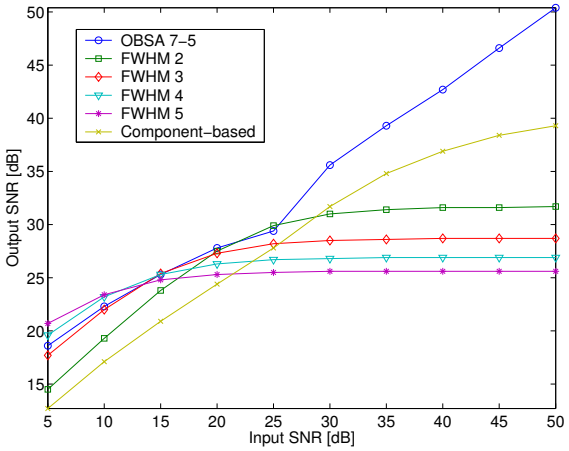


Figure 4. The output SNR plotted against the input SNR of wavelet-based denoising (OBSA 7-5) and Gaussian filtering (FWHM) with filters of increasing width. Also plotted is the performance of scalar wavelet-based denoising of the individual vector components independently of each other.

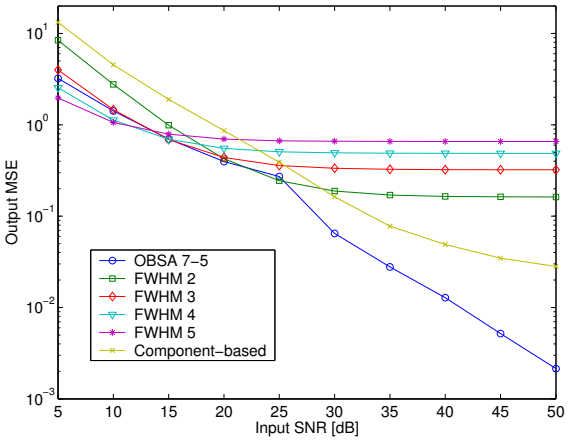
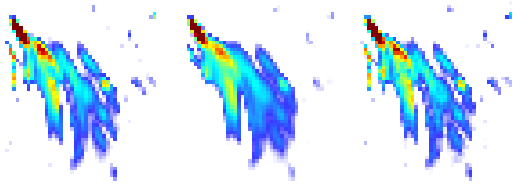


Figure 5. Output MSE plotted against the input SNR of wavelet-based denoising (OBSA 7-5) and Gaussian filtering (FWHM) with filters of increasing width. Also plotted is the performance of scalar wavelet-based denoising of the individual vector components independently of each other.

We also computed the mean square errors (MSE) between the original data and the denoised data, and the results are shown in Fig. 5. The vertical axis is on a logarithmic scale. The plot confirms that Gaussian filtering smooths too much when the noise level is low, which results in an MSE that is almost two orders of magnitude larger in comparison with our method.

Although the SNR is a good measure for the overall performance, it is not suitable to measure how well local features are retained. A problem, however, is that



(a) (b) (c)

Figure 7. 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 row of Fig. 6. (a) Noise-free data. (b) Gaussian filtering. (c) Wavelet-based denoising. Note how the small vertical structure on the left disappears with Gaussian filtering.

we do not actually have suitable quantitative measures, therefore, we render an image of the feature of interest, and make a visual assessment of the performance. Our feature of interest is a measure of vorticity, commonly referred to as the λ_2-definition [Jeo95]. The method computes the eigenvalues λ_1, λ_2, and $\lambda_3, \lambda_1 \geq \lambda_2 \geq \lambda_3$, of the matrix

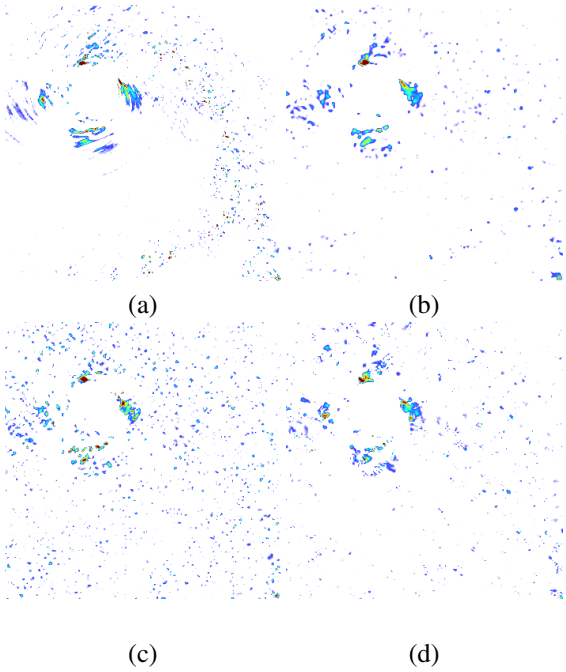
$$M = \left[\frac{J+J^{\mathrm{T}}}{2}\right]^2 + \left[\frac{J-J^{\mathrm{T}}}{2}\right]^2. \qquad (13)$$

Here, J is the velocity gradient tensor. Vortex cores are defined as the points where λ_2 is negative.

Figure 6 shows color-encoded (blue to red) λ_2 values in a selected range for some of the generated noisy vector fields (left column), and the results of denoising these data sets by Gaussian filtering and our method. The middle column shows the best results obtained by Gaussian filtering, and the right column shows the results of our method using the OBSA 7-5 multiwavelets. The SNR is displayed below each image, as well as the filter size of the Gaussian kernel, and the percentage of wavelet coefficients that remain after thresholding. These percentages are indicative of the power of wavelets to capture relevant features with only a small number of coefficients.

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. 7, in which a small vertical structure is visible in the original data (Fig. 7(a)), which is lost by Gaussian filtering (Fig. 7(b)), but retained by our wavelet-based method (Fig. 7(c)).

We have seen that for high noise levels, Gaussian filtering performs better, because stronger low-pass filtering is needed. However, this is also possible to perform with our method. It appears that the threshold selection process underestimates the noise level,



(a) (b) (c) (d)

Figure 8. Denoising of the noisy vector data with SNR $= 10$. All images show color-encoded λ_2 values in a selected range. (a) Rendering of the noise-free data. (b) Result of Gaussian filtering with FWHM 5. (c) Wavelet-based denoising with automatic threshold selection. The threshold is such that 5% of the largest detail coefficients remain after thresholding. (d) Wavelet-based denoising with the threshold lowered to a value such that only 2% of the largest detail coefficients are retained.

and that a lower threshold value is necessary. We performed a simple experiment with the noisy vector data with SNR $= 10$ to see if it is possible to improve the output of our method, and the results are shown in Fig. 8. The λ_2 values of the noise-free data are shown in Fig. 8(a). We repeat the results of Gaussian filtering and our method in Fig. 8(b) and Fig. 8(c), respectively. Our method retains about 5% of the largest detail coefficients. When we lower the threshold such that only 2% of the largest coefficients are retained, we obtain the image shown in Fig. 8(d). The SNR improves only slightly to SNR $= 22.5$, but the visual appearance of the features is much improved, and we also see a reduction of artifacts, i.e., features introduced that were not in the original noise-free data. Although this shows that it is possible to obtain a more 'smooth' result with our method, the problem is that this approach introduces a parameter (the number of coefficients to retain) in the method.

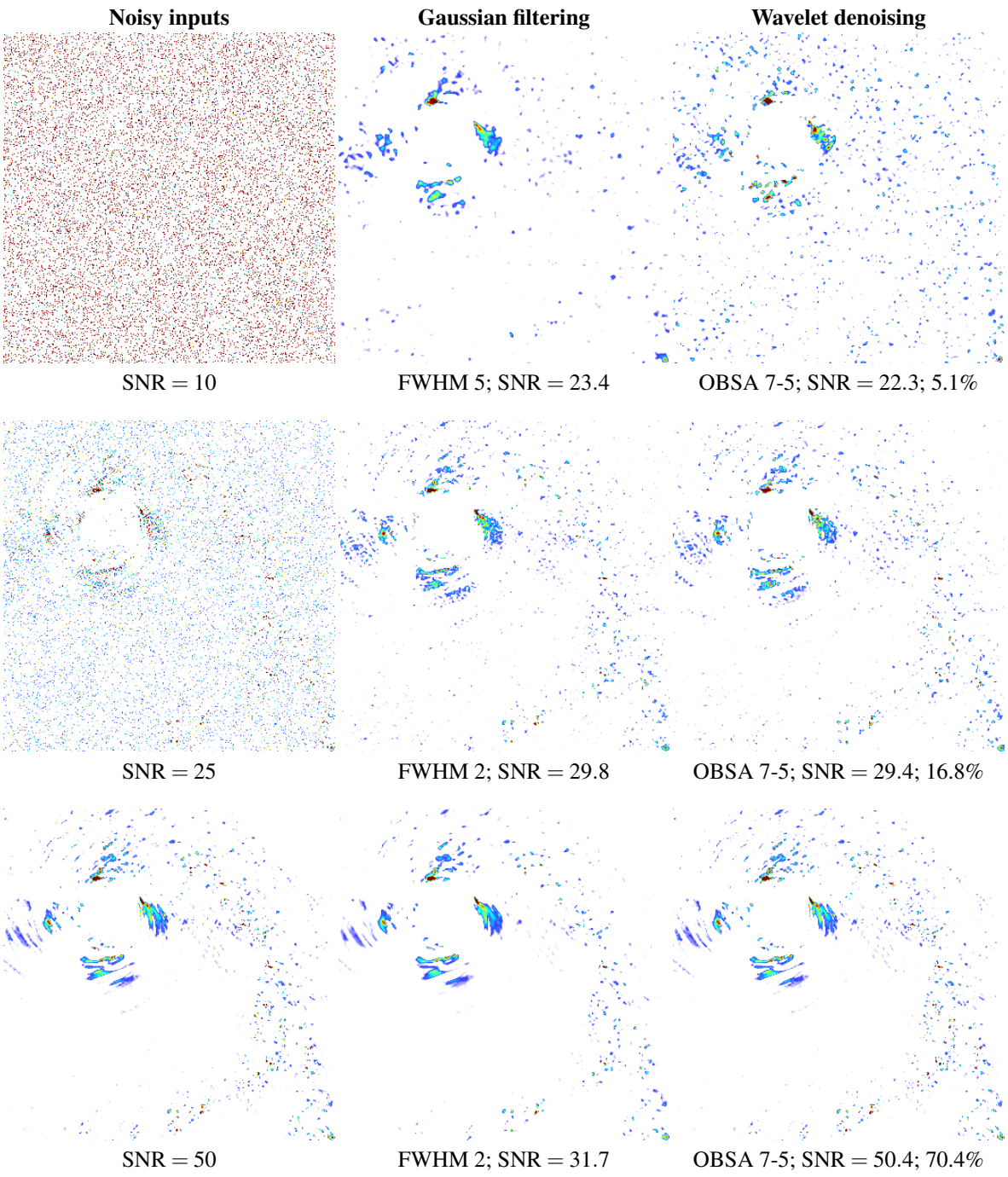


Figure 6. Results of denoising using Gaussian filtering and wavelet-based denoising. All images show color-encoded λ_2 values in a selected range. Left column: noisy input data of various signal-to-noise ratios. Middle column: results of Gaussian filtering using the filter with the best performance. Right column: wavelet-based denoising with the OBSA 7-5 multiwavelets. The depth of the wavelet decomposition was fixed at three levels. The resulting SNR after denoising is shown below the images. Additionally, the right column shows the percentage of remaining wavelet detail coefficients.

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.

We have shown that the proposed method outperforms Gaussian smoothing for low to moderate noise levels. For very high noise levels, the wavelet threshold selection appears to underestimate the noise level, and in such case, Gaussian filtering performs better. However, by adapting the threshold, we have demonstrated that the result can be improved. This should be investigated in a more systematic way, and it would be interesting to see if other wavelet coefficient threshold selection schemes produce better results.

We have also performed a simple experiment in which we used scalar denoising applied to the vector components independently. The result of this experiment shows that it is necessary to treat the vector components in a coupled way. It would be possible to use a component-wise scalar wavelet transform combined with our proposed vector coefficient thresholding. We expect, however, that the performance will still be lower, since the vector wavelet transform already considers the coupling of the vector components during the decomposition phase.

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 also result in a promising denoising method for diffusion-tensor MRI volumetric data. It may also be useful for the study of cardiovascular function, and a comparison with the method proposed by Ng [Ng03], should be made. Finally, it is necessary to evaluate the method on PVI data sets, which is ongoing work.

6. ACKNOWLEDGEMENTS

This research was funded by the project SFB 382 of the German Research Foundation (DFG), and by the Alexander von Humboldt Foundation with a Humboldt Research Fellowship for the first author.

7. REFERENCES

[Blo98] Blomgren, P. and Chan, T. F. Color TV: total variation methods for restoration of vector-valued images. *IEEE Trans. Image Processing*, 7(3):304–309, 1998.

[Buc95] Buckheit, J. B. and Donoho, D. L. WaveLab and reproducible research. Technical Report 474, Dept. of Statistics, Stanford University, 1995.

[Cha00] Chang, S. G., Yu, B., and Vetterli, M. Adaptive wavelet thresholding for image denoising and compression. *IEEE Trans. Image Processing*, 9(9):1532–1546, 2000.

[Chu92] Chui, C. K. *An Introduction to Wavelets*. Academic Press, 1992.

[Don95] Donoho, D. L. De-noising by soft thresholding. *IEEE Trans. Information Theory*, 41:613–627, May 1995.

[Fow02] Fowler, J. E. and Hua, L. Wavelet transforms for vector fields using omnidirectionally balanced multiwavelets. *IEEE Trans. Signal Processing*, 50:3018–3027, 2002.

[Jeo95] Jeong, J. and Hussain, F. On the identification of a vortex. *J. Fluid Mechanics*, 285:69–94, 1995.

[Mal89] Mallat, S. G. A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 11(7):674–693, 1989.

[Ng03] Ng, Y.-H. P. and Yang, G.-Z. Vector-valued image restoration with applications to magnetic resonance velocity imaging. *J. WSCG*, 11(2):338–345, 2003.

[Per90] Perona, P. and Malik, J. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[Per98] Perona, P. Orientation diffusions. *IEEE Trans. Image Processing*, 7(3):457–467, 1998.

[Pižˇ03] Pižurica, A., Philips, W., Lemahieu, I., and Acheroy, M. A versatile wavelet domain noise filtration technique for medical imaging. *IEEE Trans. Medical Imaging*, 22(3):323–331, 2003.

[Pra00] Prasad, A. K. Particle image velocimetry. *Current Science*, 79(1):51–60, 2000.

[Rud92] Rudin, L. I., Osher, S., and Fatemi, E. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[Sim96] Simoncelli, E. P. and Adelson, E. H. Noise removal via Bayesian wavelet coring. In *Proc. IEEE Int. Conf. Image Processing*, volume 1, pages 379–382, Lausanne, Switzerland, September 16–19 1996.

[Str01] Strela, V. and Walden, A. T. Signal and image denoising via wavelet thresholding: Orthogonal and biorthogonal, scalar and multiple wavelet transforms. In Fitzgerald, W. F., Smith, R. L., Walden, A. T., and Young, P. C., editors, *Nonlinear and Nonstationary Signal Processing*. Cambridge University Press, 2001.

[Tan99] Tan, H. H., Shen, L.-X., and Tham, J. Y. New biorthogonal multiwavelets for image compression. *Signal Processing*, 79(1):45–65, 1999.

[Win04] Wink, A. M. and Roerdink, J. B. T. M. Denoising functional MR images: a comparison of wavelet denoising and Gaussian smoothing. *IEEE Trans. Medical Imaging*, 23(3):374–387, 2004.

[Xia96] Xia, X.-G. and Suter, B. W. Vector-valued wavelets and vector filter banks. *IEEE Trans. Signal Processing*, 44(3):508–518, 1996.