

Median Filtering of Tensor-Valued Images

Martin Welk, Christian Feddern, Bernhard Burgeth, and Joachim Weickert

Mathematical Image Analysis Group
Faculty of Mathematics and Computer Science, Bldg. 27
Saarland University, 66041 Saarbrücken, Germany
{welk, feddern, burgeth, weickert}@mia.uni-saarland.de
<http://www.mia.uni-saarland.de>

Abstract. Novel matrix-valued imaging techniques such as diffusion tensor magnetic resonance imaging require the development of edge-preserving nonlinear filters. In this paper we introduce a median filter for such tensor-valued data. We show that it inherits a number of favourable properties from scalar-valued median filtering, and we present experiments on synthetic as well as on real-world images that illustrate its performance.

1 Introduction

Diffusion tensor magnetic resonance imaging (DT-MRI) is a recent medical image acquisition technique that measures the diffusion characteristics of water molecules in tissue. The resulting diffusion tensor field is a positive semidefinite matrix field that provides valuable information for brain connectivity studies as well as for multiple sclerosis or stroke diagnosis [15]. These matrix-valued data are often polluted with noise, hence it is necessary to develop filters to remove this noise without losing too much valuable information. Similar problems also occur in other situations where matrix-valued data are to be smoothed: Tensor fields have shown their use as a common description tool in image analysis, segmentation and grouping [9]. This also includes widespread applications of the so-called structure tensor (Förstner interest operator, second moment matrix, scatter matrix) [8] in fields ranging from motion analysis to texture segmentation. Moreover, a number of scientific applications require to process tensor fields: The tensor concept is a common physical description of anisotropic behaviour in solid mechanics and civil engineering, where stress-strain relationships, inertia tensors, diffusion tensors, and permittivity tensors are used.

For scalar-valued images, the median filter is one of the most frequently used structure-preserving smoothing methods, since it is simple, robust against outliers, and preserves discontinuities. The goal of this paper is to introduce a median filter for matrix-valued images where the matrices are positive (semi-)definite. To this end we will start with a review of the properties of the scalar-valued median in Section 2. In Section 3, we will introduce a median for tensor fields as a solution of a minimisation problem originating from a basic property of the median for scalar-valued data. Algorithmic aspects will be sketched in Section 4.

The fifth section shows experiments on synthetic and real-world images. In the final sixth section we present concluding remarks.

Related work. The search for good smoothing techniques for DT-MRI data and related tensor fields is a very recent research area. Several authors have addressed this problem by smoothing derived expressions such as the eigenvalues and eigenvectors of the diffusion tensor [16, 6, 17] or its fractional anisotropy [14]. Some methods that work directly on the tensor components use linear [20] or nonlinear [10] techniques that filter all channels *independently*, thus performing scalar-valued filtering again. Nonlinear regularisation methods for matrix-valued filtering with channel coupling have been proposed in [17, 19]. Related nonlinear diffusion methods for tensor-valued data have led to the notion of a nonlinear structure tensor [19] that has been used for optic flow estimation [4].

There are several proposals on how to generalise the median filter to vector-valued data; see e.g. [3, 13] and the references therein. To our knowledge, however, no attempts have been made so far to design median filters for tensor fields.

2 Properties of Scalar-Valued Median Filters

One of the basic tasks of statistics is the description of some arbitrary sample data $x = \{x_1, x_2, \dots, x_n\}$ by a single number that is representative of the data. Such a number is commonly called an average. The median \tilde{x} is a prominent example of a position average, in contrast to the arithmetic mean \bar{x} as a computed average. The median is found by locating the place of a value in a sample series. As a measure of central tendency the median \tilde{x} is the value of the middle item in a sample series when the items are ordered according to their magnitude.

It can be formally defined as that value which divides a sample series in such a way that at least 50 percent of the items are equal to or less than it and at least 50 percent of the items are equal to or greater than it. This alludes to the origin of the median as a so-called 50 percent quantile. It is clear that the median depends heavily on the existence of a total order for the sample items. If the number of items in a sample is odd, the median is the value of the middle term. If the number of items in a sample is even, it is usually chosen as the arithmetic mean of the two middle items (though any other average would be formally acceptable). Thus, for an ordered sample with $x_1 \leq x_2 \leq \dots \leq x_n$, the median is defined as

$$\tilde{x} := \text{med}(x_1, \dots, x_n) := \begin{cases} x_k & \text{for } n = 2k - 1, \\ \frac{1}{2}(x_k + x_{k+1}) & \text{for } n = 2k. \end{cases} \quad (1)$$

Typical for a position average, the median is highly robust with respect to outliers of the sample. This makes median filtering the method of choice when impulse noise such as salt-and-pepper noise is present, but it is equally popular for other types of noise.

Median filtering in signal processing goes back to Tukey [18]. In image processing, median filtering is usually based on considering a neighbourhood of size $(2k + 1) \times (2k + 1)$ of some pixel. Median filtering may be iterated. In this case one usually observes that after a relatively small number of iterations, the result becomes stationary (so-called root signal). It is easy to see that median filters preserve straight edges, while they round off corners. For more properties of median filters and their numerous modifications we refer to monographs on statistics [5, 11] and nonlinear image processing [7, 12].

The median has a very interesting minimisation property: The sum of absolute deviations from the median is smaller than the sum of the absolute deviations from any other point:

$$\sum_{i=1}^n |x_i - x| \geq \sum_{i=1}^n |x_i - \tilde{x}| = \min. \quad (2)$$

This property has been used in [2, 1] to generalise median filters to vector-valued data. It will also be essential for our design of matrix-valued median filters.

3 A Median for Matrix-Valued Images

The definition of a median for matrix-valued functions should inherit as many properties of the standard median described above as possible. We restrict our attention to real 2×2 -matrices $A \in \mathbb{R}^{2 \times 2}$ but the extension to larger matrices is straight forward.

We recall the definition of the Frobenius norm $\|A\|$ of a matrix $A \in \mathbb{R}^{2 \times 2}$:

$$\|A\| = \left\| \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix} \right\| := \sqrt{\sum_{i,j=1}^2 a_{ij}^2}. \quad (3)$$

We use this norm to define a median of an odd number of sample items.

Definition: *The median of the set of matrices $\{A_i : i = 1, \dots, n\}$ is the matrix \tilde{A} which solves the minimisation problem*

$$\tilde{A} := \operatorname{argmin}_X \sum_{i=1}^n \|A_i - X\|. \quad (4)$$

The solution of this minimisation problem is an element of the convex hull of the matrices $\{A_i : i = 1, \dots, n\}$. If these matrices are positive (semi-)definite, then the median is again a positive (semi-)definite matrix since the set of all such matrices is convex.

There is a new property for the median of a sample of matrices: the median should be rotationally invariant. The matrix

$$R(\varphi) := \begin{pmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{pmatrix} \quad (5)$$

describes a rotation with angle $\varphi \in [0, \pi]$ and the requirement of rotational invariance amounts to the equality

$$\text{med}(R(\varphi)A_1R^\top(\varphi), \dots, R(\varphi)A_nR^\top(\varphi)) = R(\varphi) \text{med}(A_1, \dots, A_n) R^\top(\varphi) \quad (6)$$

for any $\varphi \in [0, \pi]$ and any choice of matrices A_1, \dots, A_n . This property is clearly desirable from the practical point of view, although it has no counterpart in case of scalar-valued data. The median induced by the minimisation problem inherits the rotational invariance of the Frobenius norm:

$$\sum_{i=1}^n \|R(\varphi)AR^\top(\varphi) - R(\varphi)XR^\top(\varphi)\| = \sum_{i=1}^n \|A - X\| \quad (7)$$

holds for all X and also for the minimising \tilde{A} . Hence, $\tilde{A} = \text{med}(A_1, \dots, A_n)$ is independent of $R(\varphi)$.

4 Algorithmic Aspects

When computing the median of a set of matrices as defined here, one problem has to be solved. Each of the functions $\|A_i - X\|$ in the definition is differentiable except in A_i itself. Thus their sum is also differentiable except in the matrices of the given set. It is therefore an obvious idea to use a gradient descent method. Unfortunately the gradient vector $\nabla\|A_i - X\|$ has the same length everywhere. Although $-\nabla\|A_i - X\|$ always points in the direction of A_i , it contains no information about the distance to A_i . In the same way the gradient of the sum lacks information on the distance to the minimum. We overcome this problem by implementing a step size control based on the over- and undershoots encountered in the subsequent iteration steps.

The algorithm therefore works as follows. First we find the A_j for which $\sum_{i=1}^n \|A_i - A_j\|$ takes its minimal value within the given set. For that A_j , we compute $\nabla \sum_{i \neq j} \|A_i - A_j\|$. If this gradient is of length 1 or smaller, then $X = A_j$ minimises

$$S(X) = \sum_{i=1}^n \|A_i - X\| \quad (8)$$

and is therefore the median. Otherwise we proceed by gradient descent using the gradient of $S(X)$. After each step in which we change X to

$$X' = X - s \nabla S(X) \quad (s > 0), \quad (9)$$

we compare $\nabla S(X)$ with the projection of $\nabla S(X')$ onto $\nabla S(X)$. This provides an estimate for over- or undershoots which allows us to adapt s for the subsequent step and, in case of extreme overshoots, even to roll back the last step.

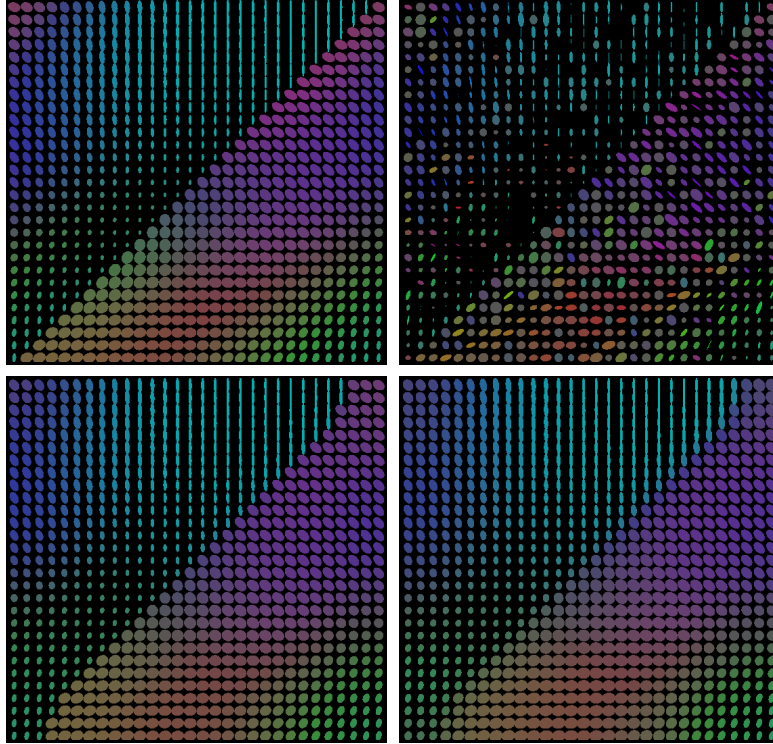


Fig. 1. Edge preservation and noise robustness of matrix-valued median filtering. **(a) Top Left:** Tensor field with a discontinuity. The matrices are visualised by ellipses. Colour indicates orientation and brightness encodes eccentricity. **(b) Top Right:** Degraded version of (a) where the eigenvalues are perturbed by Gaussian noise. **(c) Bottom Left:** Median filtering of (a) shows the discontinuity-preserving qualities (5×5 median, 5 iterations). **(d) Bottom Right:** Median filtering of (b) illustrates the denoising capabilities (5×5 median, 5 iterations).

5 Experiments

Symmetric positive definite matrices $A \in \mathbb{R}^{2 \times 2}$ can be visualised as ellipses

$$\{x \in \mathbb{R}^2 : x^\top A^{-2} x = 1\}. \quad (10)$$

We prefer this representation using the matrix A^{-2} rather than A , since then the larger (smaller) eigenvalue corresponds directly to the semi-major (-minor) axis of the displayed ellipse.

In Figure 1 we illustrate the discontinuity-preserving properties of matrix-valued median filtering by applying it to a synthetic data set that contains a discontinuity. We observe that five iterations of 5×5 median filtering hardly affects this discontinuity. Almost the same results can be obtained when noise

is present which perturbs the eigenvalues of the matrix. This illustrates that the median filter inherits its high robustness against outliers from its scalar-valued counterpart. We observe that at the image boundary, the location of the discontinuity is shifted. This effect has been caused by imposing reflecting boundary conditions. This symmetry constraint encourages structures that are perpendicular to the boundary, since deviations from the perpendicular behaviour create corner-like structures.

The behaviour of matrix-valued median filtering on real-world images is studied in Figure 2. In this case we have extracted a 2D frame from a 3D DT-MRI data set of a human head and restricted ourselves to four channels in this plane. 30 % of all data have been replaced by noise matrices. Their angle of the eigen-system was uniformly distributed in $[0, \pi]$, and their eigenvalues are uniformly distributed in the range $[0, 127]$. We observe that the noise robustness and discontinuity preservation that has already been observed in Figure 1 is also present in this case. Moreover, Figure 2(f) suggests that root signals also exist in the matrix-valued case. As can be expected, increasing the stencil size leads to a more pronounced filtering.

6 Conclusions

In this paper we have extended the notion of median filtering to the case of matrix-valued data sets. This has been achieved by seeking the matrix that minimises the sum of the distances to the other sample items in the Frobenius norm. Experiments on synthetic and real-world tensor fields show that the resulting median filter inherits important properties from its scalar-valued counterpart: It is robust against outliers and it preserves discontinuities. In our future work we plan to generalise other nonlinear filters in order to make them applicable to tensor field filtering.

Acknowledgements

We are grateful to Anna Vilanova i Bartrolí (Biomedical Imaging Group, TU Eindhoven) and Carola van Pul (Maxima Medical Center, Eindhoven) for providing us with the DT-MRI data set and for discussing questions concerning data conversion. Susanne Biehl has written our conversion tool and Rico Philipp has developed our tensor visualisation software.

References

1. J. Astola, P. Haavisto, and Y. Neuvo. Vector median filters. *Proceedings of the IEEE*, 78(4):678–689, 1990.
2. T. L. Austin, Jr. An approximation to the point of minimum aggregate distance. *Metron*, 19:10–21, 1959.
3. V. Barnett. The ordering of multivariate data. *Journal of the Royal Statistical Society A*, 139(3):318–355, 1976.

4. T. Brox and J. Weickert. Nonlinear matrix diffusion for optic flow estimation. In L. Van Gool, editor, *Pattern Recognition*, volume 2449 of *Lecture Notes in Computer Science*, pages 446–453. Springer, Berlin, 2002.
5. Y. Chou. *Statistical Analysis*. Holt, Reinehart and Winston, London, 1969.
6. O. Coulon, D. C. Alexander, and S. A. Arridge. A regularization scheme for diffusion tensor magnetic resonance images. In M. F. Insana and R. M. Leahy, editors, *Information Processing in Medical Imaging – IPMI 2001*, volume 2082 of *Lecture Notes in Computer Science*, pages 92–105. Springer, Berlin, 2001.
7. E. R. Dougherty and J. Astola, editors. *Nonlinear Filters for Image Processing*. SPIE Press, Bellingham, 1999.
8. W. Förstner and E. Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *Proc. ISPRS Intercommission Conference on Fast Processing of Photogrammetric Data*, pages 281–305, Interlaken, Switzerland, June 1987.
9. G. H. Granlund and H. Knutsson. *Signal Processing for Computer Vision*. Kluwer, Dordrecht, 1995.
10. K. Hahn, S. Pigarin, and B. Pütz. Edge preserving regularization and tracking for diffusion tensor imaging. In W. J. Niessen and M. A. Viergever, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2001*, volume 2208 of *Lecture Notes in Computer Science*, pages 195–203. Springer, Berlin, 2001.
11. J. Hartung. *Statistik*. R. Oldenbourg Verlag, München, 4 edition, 1985.
12. R. Klette and P. Zamperoni. *Handbook of Image Processing Operators*. Wiley, New York, 1996.
13. A. Koschan and M. Abidi. A comparison of median filter techniques for noise removal in color images. In *Proc. Seventh German Workshop on Color Image Processing*, pages 69–79, Erlangen, Germany, Oct. 2001.
14. G. J. M. Parker, J. A. Schnabel, M. R. Symms, D. J. Werring, and G. J. Barker. Nonlinear smoothing for reduction of systematic and random errors in diffusion tensor imaging. *Journal of Magnetic Resonance Imaging*, 11:702–710, 2000.
15. C. Pierpaoli, P. Jezzard, P. J. Basser, A. Barnett, and G. Di Chiro. Diffusion tensor MR imaging of the human brain. *Radiology*, 201(3):637–648, Dec. 1996.
16. C. Poupon, J. Mangin, V. Frouin, J. Régis, F. Poupon, M. Pachot-Clouard, D. Le Bihan, and I. Bloch. Regularization of MR diffusion tensor maps for tracking brain white matter bundles. In W. M. Wells, A. Colchester, and S. Delp, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 1998*, volume 1496 of *Lecture Notes in Computer Science*, pages 489–498. Springer, Berlin, 1998.
17. D. Tschumperlé and R. Deriche. Diffusion tensor regularization with constraints preservation. In *Proc. 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, pages 948–953, Kauai, HI, Dec. 2001. IEEE Computer Society Press.
18. J. W. Tukey. *Exploratory Data Analysis*. Addison–Wesley, Menlo Park, 1971.
19. J. Weickert and T. Brox. Diffusion and regularization of vector- and matrix-valued images. In M. Z. Nashed and O. Scherzer, editors, *Inverse Problems, Image Analysis, and Medical Imaging*, volume 313 of *Contemporary Mathematics*, pages 251–268. AMS, Providence, 2002.
20. C. Westin, S. E. Maier, B. Khidhir, P. Everett, F. A. Jolesz, and R. Kikinis. Image processing for diffusion tensor magnetic resonance imaging. In C. Taylor and A. Colchester, editors, *Medical Image Computing and Computer-Assisted Intervention – MICCAI 1999*, volume 1679 of *Lecture Notes in Computer Science*, pages 441–452. Springer, Berlin, 1999.

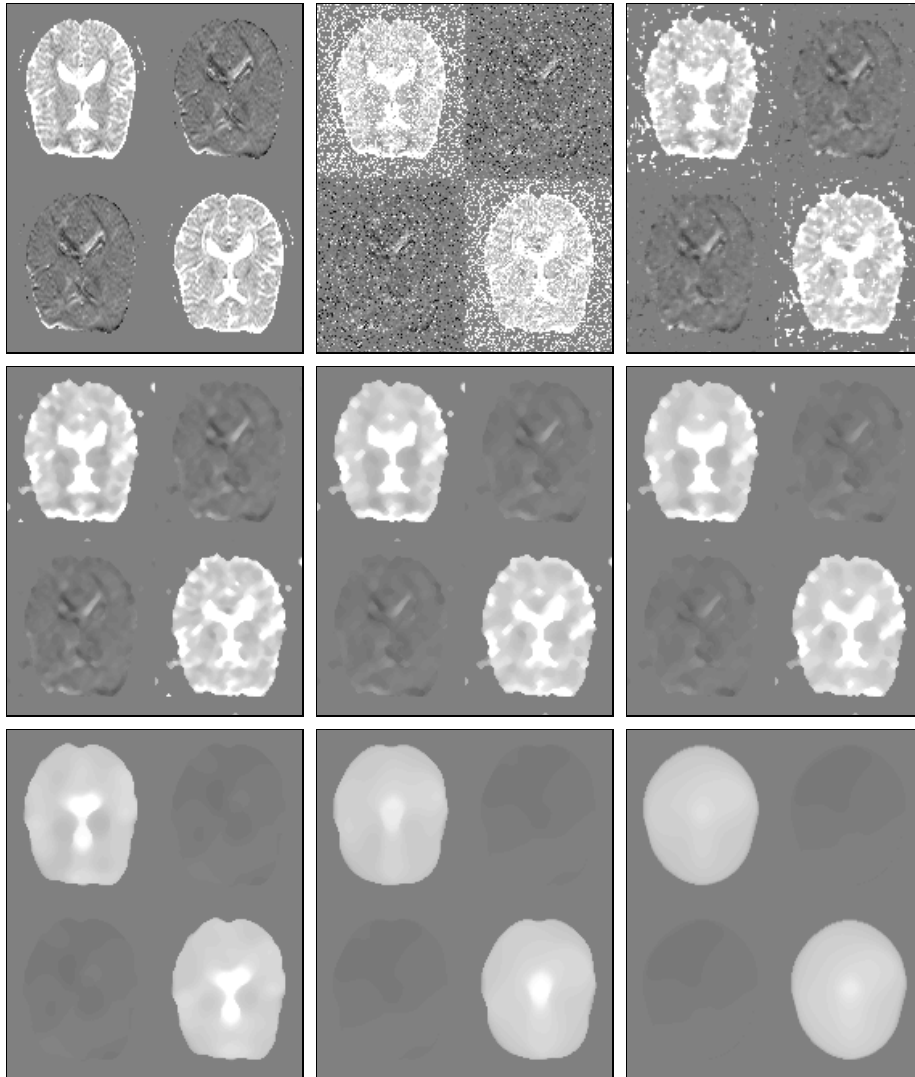


Fig. 2. Matrix-valued median filtering applied to a 2D DT-MRI frame. **(a) Top Left:** The four channels (x, x) , (x, y) , (y, x) , and (y, y) create four subimages of size 92×108 . **(b) Top Middle:** Degraded version of (a) where uniform noise is used. **(c) Top Right:** Median filtering of (b) with 1 iteration of a 3×3 median. **(d) Middle Left:** Ditto, 5 iterations. **(e) Middle Middle:** 25 iterations. **(f) Middle Right:** 125 iterations. Note the similarity to (e). **(g) Bottom Left:** 25 iterations with a 5×5 median. **(h) Bottom Middle:** Ditto, 7×7 median. **(i) Bottom Right:** 9×9 median.