Matrix-Valued Filters as Convex Programs

Martin Welk[1], Florian Becker[2], Christoph Schnörr[2], and Joachim Weickert[1]

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

[2] Computer Vision, Graphics, and Pattern Recognition Group
Faculty of Mathematics and Computer Science
University of Mannheim, 68131 Mannheim, Germany
flbecker@rumms.uni-mannheim.de, schnoerr@uni-mannheim.de
http://www.cvgpr.uni-mannheim.de

Abstract. Matrix-valued images gain increasing importance both as the output of new imaging techniques and as the result of image processing operations, bearing the need for robust and efficient filters for such images. Recently, a median filter for matrix-valued images has been introduced. We propose a new approach for the numerical computation of matrix-valued median filters, and closely related mid-range filters, based on sound convex programming techniques. Matrix-valued medians are uniquely computed as global optima with interior point solvers. The robust performance is validated with experimental results for matrix-valued data including texture analysis and denoising.

1 Introduction

In this paper, we are concerned with the processing of images where the value attached to each pixel or voxel is a symmetric matrix. Image data of this kind appear in a variety of different contexts in modern image acquisition and processing. For example, diffusion tensor magnetic resonance imaging (DT-MRI) is an upcoming medical image acquisition technique which measures the diffusion characteristics of water molecules in tissue, yielding valuable insights into the structure and function of tissues, particularly fibre connectivity in the brain [13]. Moreover, structure tensors arise as derived quantities in motion detection, texture analysis and segmentation and other fields of image processing [8]. Tensor data also occur in solid and fluid mechanics. The latter can have eigenvalues of either sign while diffusion tensors and structure tensors are positive semidefinite.

All of these data, be they directly measured or computed, are often degraded by noise. One of the basic tasks in matrix-valued image processing as in other fields of image processing is therefore denoising. A simple but effective denoising filter is the matrix-valued median filter introduced in [18]. Based on generalising the minimisation property of the scalar-valued median, it inherits from its scalar counterpart the robustness and capability to preserve discontinuities. To compute matrix-valued medians, in [18] a gradient descent algorithm was proposed.

In this paper, we introduce a new and efficient algorithm for the computation of matrix-valued medians according to the (slightly generalised) definition from [18]. The new approach is based on convex conic programming methods and can easily be adapted to closely related problems like the computation of matrix-valued mid-range filters. We apply the new algorithm to DT-MRI data to demonstrate its use. Furthermore, we use matrix-valued medians to smooth structure tensor data from textured grey-value images as a preprocessing step for texture segmentation.

We proceed as follows. In Section 2 we describe the local matrix-valued image filters that we are concerned with. Sections 3 shows how these filters can be rewritten as convex optimisation problems which are then solved in Section 4. Experiments on DT-MRI data and local orientation estimation of grey-value images are presented and discussed in Section 5. Conclusive remarks are given in Section 6.

Related Work. Median filtering of matrix-valued data is closely related to that of vector valued data. Indeed, the definition from Welk et al. [18] has an obvious vector-valued analog. For earlier approaches to vector-valued median filtering in the image processing literature we refer to Astola et al. [1] and Caselles et al. [7]. While Caselles et al. [7] require that the median has always to be one of the given data vectors, Astola et al. [1] relax this condition somewhat while the definition given in [18] does not make such a restriction at all. Barni et al. [3] define a vector median using the Euclidean distance sum minimisation similarly as [18], but again restricted to the given data vectors. Interestingly, the exact analog to the definition from [18] for 2-D vectors has already been proposed in 1959 by Austin [2] along with a graphical algorithm which is closely related to the gradient descent procedure from [18]. The problems of this procedure and improvements have been discussed in Seymour's 1970 reply [15]. Vector-valued medians as well as vector-valued mid-range values (often called 1-centres) have also been studied in the context of facility location problems, see e.g. Megiddo [12], Fekete et al. [9] and the references therein.

The concept of the structure tensor goes back to Förstner and Gülch [8]. It is common in image analysis to smooth the rank one matrices which arise directly from the gradient vectors in single points by Gaussian convolution which leads in general to rank two matrices which integrate directional information from a neighbourhood and suffer less from noise sensitivity. The observation that the Gaussian convolution used in this process is essentially a linear diffusion of the directional information, thus introducing a blurring that is unwished at times, led to the definition of a nonlinear structure tensor by Weickert and Brox [17], [6] in which Gaussian convolution is replaced by nonlinear diffusion. For its better preservation of discontinuities, the nonlinear structure tensor is well-suited for texture segmentation [4], [14] and optical flow analysis [6]. Smoothing structure tensors with medians is also related to the robust structure tensor introduced by van den Boomgaard and van der Weijer in [16] and which for a particular choice of the penaliser function ϱ also amounts to a minimisation similar to that in the matrix-valued median.

Regarding convex programming, all concepts we use can be found in corresponding textbooks (e.g., Boyd and Vandenberghe [5]). Recently, these optimisation methods have been also successfully applied to various other image processing problems by Keuchel et al. [11].

Notation and Preliminaries. Throughout the paper, e denotes the vector $(1, \ldots, 1)^\top \in \mathbb{R}^n$. By I_d we denote the $d \times d$ unit matrix. Further, $\mathcal{L}^d$ is the convex cone of vectors $\left\{x \in \mathbb{R}^d \mid x_d \geq \sqrt{x_1^2 + \cdots + x_{d-1}^2}\right\}$ while $\mathcal{S}^d$ is the linear space of symmetric $d \times d$ real matrices. The i^{th} eigenvalue of $X \in \mathcal{S}$ in the order $\lambda_1(X) \geq \cdots \geq \lambda_d(X)$ will be denoted by $\lambda_i(X)$. Finally, by $\mathcal{S}^d_+$ we mean the convex cone of positive semidefinite symmetric matrices $\{X \in \mathcal{S}^d \mid \lambda_d(X) \geq 0\}$.

2 Problem Statement: Local Matrix Filters

Given n real numbers $a_1, a_2, \ldots, a_n$, their median is defined as the middle value in the sequence that contains all the numbers ordered by size. The median concept gives rise to a class of image filters, called *median filters*, which are known for their outstanding capability for edge-preserving denoising of images. Median filtering of a discrete grey-value image requires the specification of a pixel mask, the so-called structure element, which is used to select a neighbourhood for each pixel. The new grey-value of each pixel is taken to be the median of the old grey-values of all pixels within its neighbourhood. Median filtering can be iterated, thereby performing a progressive edge-preserving smoothing. This can be compared to the approximation of the (non-edge-preserving) Gaussian smoothing by iterated box averaging.

The matrix-valued generalisation of median filtering introduced in [18] is based on an interesting energy minimisation property of the scalar-valued median: The median of $a_1, a_2, \ldots, a_n$ is exactly the real number x for which $\sum_{i=1}^n |x - a_i|$ is minimal. The median of n matrices $A_1, \ldots, A_n \in \mathcal{S}^d$ is then defined as

$$\operatorname{med}(A_1, \ldots, A_n) := \operatorname{argmin}_{X \in \mathcal{S}^d} \sum_{i=1}^n d(X, A_i)$$

where d is a suitable, rotationally invariant metric on $\mathcal{S}^d$. In [18], the Frobenius norm was used,

$$d(X, A_i) = \|X - A_i\|_2 = \sqrt{\operatorname{tr}\left[(X - A_i)(X - A_i)\right]} \; ;$$

another possible choice is the spectral norm,

$$d(X, A_i) = |X - A_i| = \max_{i=1,\ldots,d} \left|\lambda_i(X - A_i)\right| \; .$$

Interestingly, the so-called mid-range value of real numbers $a_1, a_2, \ldots, a_n$ which is defined as the arithmetic mean of their maximum and minimum, can be described by an extremality property very similar to that of the median – instead of the sum of the distances $|x - a_i|$, their maximum is minimised. The transfer to matrices is therefore straightforward. We define

$$\operatorname{midr}(A_1, \ldots, A_n) := \operatorname{argmin}_{X \in \mathcal{S}^d} \max \left\{ d(X, A_1), \ldots, d(X, A_n) \right\}$$

with the same requirements for d as in the case of the median. Midrange filtering is less attractive by itself but stands in close relation to other matrix filters.

3 Convex Optimisation

In this section, we show that each filter introduced in the previous section is defined as global optimum of a *convex* optimisation problem.

3.1 Median Filter: Frobenius Norm

We consider the optimisation problem:

$$\operatorname{med}_F(A_1, \ldots, A_n) := \operatorname{argmin}_{X \in \mathcal{S}^d} \sum_{i=1}^{n} \|X - A_i\|_2 \tag{1}$$

and identify the unknown matrix $X \in \mathcal{S}^d$ with a vector $X \in \mathbb{R}^{d^2}$. Introducing n additional variables $t = (t_1, \ldots, t_n)^\top$, we rewrite (1):

$$\inf_{X \in \mathcal{S}^d, t \in \mathbb{R}^n} \langle e, t \rangle \ , \quad \|X - A_i\|_2 \leq t_i \ , \quad i = 1, \ldots, n \tag{2}$$

Each constraint is convex, because $(X^\top, t_i)^\top$ varies in the convex cone $\mathcal{L}_i^{d^2+1}$ translated by $(A_i^\top, 0)^\top$. Denoting the corresponding convex constraint sets with C_i, $i = 1, \ldots, n$, problem (2) reads:

$$\inf_{X \in \mathcal{S}^d, t \in \mathbb{R}^n} \langle e, t \rangle \ , \quad \begin{pmatrix} X \\ t \end{pmatrix} \in \bigcap_{i=1}^{n} C_i \tag{3}$$

This optimisation problem is convex, since the objective function is linear, and since the intersection of convex sets is convex, too.

3.2 Median Filter: Spectral Norm

We consider the optimisation problem:

$$\operatorname{med}_S(A_1, \ldots, A_n) := \operatorname{argmin}_{X \in \mathcal{S}^d} \sum_{i=1}^{n} |X - A_i| \tag{4}$$

Similary to section 3.1, we introduce auxiliary variables $t \in \mathbb{R}^n$ and corresponding constraints:

$$|X - A_i| \leq t_i \ , \quad i = 1, \ldots, n$$

These constraints are satisfied if

$$t_i I_d - (X - A_i) \in \mathcal{S}^d_+ \qquad \text{and} \qquad t_i I_d + (X - A_i) \in \mathcal{S}^d_+ \,, \qquad i = 1, \dots, n$$

Again, the variables (X, t_i) are constrained to convex sets, defined by the intersection of affine sets (left hand sides) with the convex cone $\mathcal{S}^d_+$. Denoting the constraint sets with $C_{i,+}$, $C_{i,-}$, $i = 1, \dots, n$, we can rewrite problem (4):

$$\min_{X \in \mathcal{S}^d, t \in \mathbb{R}^n} \langle e, t \rangle \,, \quad \begin{pmatrix} X \\ t \end{pmatrix} \in \bigcap_{i=1}^{n} \big(C_{i,+} \cap C_{i,-} \big) \tag{5}$$

This optimisation problem is convex, since the objective function is linear, and since the intersection of convex sets is convex, too.

We remark that for positive semidefinite data $A_i \in \mathcal{S}^d_+$, $i = 1, \dots, n$, the constraints represented by the sets $C_{i,-}$ are redundant and can be dropped.

3.3 Midrange Filters

For midrange filters defined by

$$\mathrm{midr}(A_1, \dots, A_n) := \mathrm{argmin}_{X \in \mathcal{S}^d} \max \big\{ d(X, A_1), \dots, d(X, A_n) \big\} \,, \tag{6}$$

we introduce the *scalar* auxiliary variable $t := \max \big\{ d(X, A_1), \dots, d(X, A_n) \big\}$. Similar to the derivation of (3) and (5), problem (6) results in two convex optimisation problems, depending on which norm we choose. We focus on the median filters in the remainder of this paper.

4 Convex Programming and Duality

We represent the optimisation problems defined in the previous section as convex programs. This allows to implement matrix-valued median filters using corresponding numerical interior-point algorithms. The corresponding dual programs reveal that solutions automatically satisfy plausible conditions whose direct computation (without convex programming) would be more involved.

4.1 Convex Conic Programs

Conic programs generalize linear programs by replacing the standard cone $\mathbb{R}^n_+$ with more general convex cones $\mathcal{K}$:

$$\inf_x \langle c, x \rangle \,, \quad Fx - g \in \mathcal{K} \tag{7}$$

The corresponding dual conic program reads:[3]

$$\sup_y \langle g, y \rangle \,, \quad F^\top y = c \,, \quad y \in \mathcal{K} \tag{8}$$

[3] In general, conic duals are defined w.r.t. dual cones $\mathcal{K}_*$. In this paper, however, we consider only self-dual cones $\mathcal{K}_* = \mathcal{K}$.

If at least one of these problems is bounded and strictly feasible, then $\{x, y\}$ is a pair of optimal solutions if and only if the duality gap is zero:

$$\langle c, x \rangle = \langle g, y \rangle \tag{9}$$

4.2 Medians as Conic Programs

We consider problem (1) and identify again matrices $X, A_i \in \mathcal{S}^d$ with vectors $X, A_i \in \mathbb{R}^{d^2}$. (2) and (3) corresponds to (7):

$$\inf_{X \in R^{d^2}, t \in \mathbb{R}^n} \langle e, t \rangle \; , \quad F \begin{pmatrix} X \\ t \end{pmatrix} - g \in \mathcal{K} \; , \tag{10}$$

where F and g are obtained by stacking the matrices resp. vectors

$$\begin{pmatrix} I_{d^2} & 0_{d^2 \times n} \\ 0^\top & e_i^\top \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} A_i \\ 0 \end{pmatrix} \; , \quad i = 1, \dots, n$$

together, e_i is the i-th unit vector, and $\mathcal{K} = \mathcal{L}^{d^2+1} \times \cdots \times \mathcal{L}^{d^2+1}$.

Below, $X \in \mathcal{S}^d$ is again regarded as a matrix. Problem (4) or (5), respectively, directly lead to (7), formulated as semidefinite program:

$$\inf_{X \in \mathcal{S}^d, t \in \mathbb{R}^n} \langle e, t \rangle \; , \qquad \text{subject to } \mathcal{F} \begin{pmatrix} X \\ t \end{pmatrix} - G \in \mathcal{S}_+^{n \times d^2} \; , \tag{11}$$

with the linear mapping:

$$\mathcal{F}(X, t) = \mathrm{Diag}\big\{ \dots, t_i I_d - X, \dots, t_i I_d + X, \dots \big\} \tag{12}$$

and:

$$G = \mathrm{Diag}\big\{ \dots, -A_i, \dots, +A_i, \dots \big\} \tag{13}$$

4.3 Dual Programs and Optimality Conditions

Evaluating (8), the conic dual program to (10) reads:

$$\sup_{Y_i \in R^{d^2}} \sum_{i=1}^n \langle Y_i, A_i \rangle \; , \quad \sum_{i=1}^n Y_i = 0 \; , \quad \|Y_i\|_2 \leq 1 \; , \quad \forall i \tag{14}$$

Since $\langle \sum_{i=1}^n Y_i, X \rangle = 0$, we can rewrite the objective function as $\sum_{i=1}^n \langle Y_i, A_i - X \rangle$. Using (9), we obtain:

$$\sum_{i=1}^n \|X - A_i\|_2 = \sum_{i=1}^n \langle Y_i, A_i - X \rangle$$

The constraints $\|Y_i\|_2 \leq 1$ suggest as solution to (14):

$$Y_i = \frac{A_i - X}{\|A_i - X\|_2} \; , \quad i = 1, \dots, n$$

Inserting this into the constraint $\sum_{i=1}^{n} Y_i = 0$ yields the stationarity conditions of the original problem (1):

$$\sum_{i=1}^{n} \frac{X - A_i}{\|X - A_i\|_2} = 0 \tag{15}$$

Using this condition for the computation of X, however, leads to a non-trivial numerical optimisation problem, the need of choosing suitable damping parameters to achieve convergence, and differentiability problems in cases where the median X coincides with some data point A_i (in this case, the corresponding term in (15) is ill-defined, whereas Y_i in (14) is not). In constrast, all these problems can be avoided by the convex programming formulation presented above.

In order to compute the dual program to (11), we first have to clarify the meaning of $F^\top$ in (8) for the mapping $\mathcal{F}$ in (12). According to (12), the mapping $\mathcal{F}z = \sum_i z_i F_i$ defines elementary matrices F_i for each single variable $z_i = X_{j,k}$ or $z_i = t_j$. $F^\top$ in (8) is then given by the adjoint mapping[4] $\mathcal{F}^* Y = (\dots, \langle F_i, Y\rangle, \dots)^\top$. Computing the dual program to (11) then results – analogously to (12) and (13) – in a block-diagonal matrix of the dual variables:

$$Y = \mathrm{Diag}\{Y_1^-, \dots, Y_n^-, Y_1^+, \dots, Y_n^+\} ,$$

and, using the definition

$$Y_i := Y_i^+ - Y_i^- , \quad \forall i ,$$

to the optimisation problem:

$$\sup_{Y_i \in \mathcal{S}^d} \sum_{i=1}^{n} \langle Y_i, A_i\rangle , \quad \sum_{i=1}^{n} Y_i = 0 , \quad \mathrm{tr}[Y_i] = 1 , \quad Y_i \in S_+^d , \quad \forall i \tag{16}$$

Note the similarity of (16) and (14). Using the same reasoning as after (14), we obtain:

$$\sum_{i=1}^{n} |X - A_i| = \sum_{i=1}^{n} \langle Y_i, A_i - X\rangle$$

Again, the dual matrices Y_i seem to play the role of normalized gradients of the original objective function (4). Because the spectral norm $|\cdot|$ is non-smooth, it is not obvious how to make this more explicit. More important, however, are the computational advantages of the convex programming formulation presented above, as compared to directly optimizing (4).

5 Experiments and Discussion

In our first experiment (Fig. 1) we demonstrate the capability of matrix-valued median filtering to remove outliers from structure tensor data. The photograph

[4] $\langle F_i, Y\rangle$ denotes the matrix inner product $\mathrm{tr}[F_i^\top Y]$.

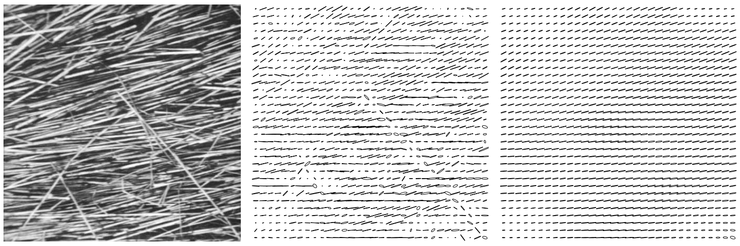

Fig. 1. Left to right: (a) Image containing oriented texture with inhomogeneities. **(b)** Structure tensors computed by smoothing the outer products $\nabla u \nabla u^\top$ with 15×15 Gaussian. The gradients themselves have been calculated by 3×3 derivative-of-Gaussian filtering. The final matrix field has been subsampled for visualisation. **(c)** Result of median filtering of (b) with 7×7 structure element and Frobenius norm (subsampled).

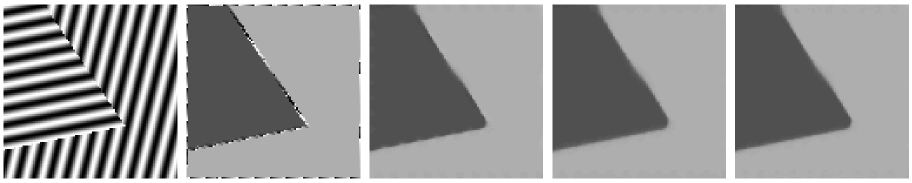

Fig. 2. Left to right: (a) Synthetic image with oriented textures. **(b)** Local orientations computed via DoGs and visualised as grey-values. **(c)** Orientations after median filtering of the orientation matrices with Frobenius norm and a disk-shaped structure element of diameter 7. **(d)** Same with structure element of diameter 9. **(e)** Spectral norm median filtering, diameter 9.

(a) shows a texture with randomly interspersed inhomogeneities. The outer products $\nabla u \nabla u^\top$ have been computed by 3×3 derivative-of-Gaussian (DoG) filters and smoothed with a 15×15 Gaussian mask. In (b) a subsampling of the resulting matrix field is shown. Outliers are removed from this matrix field by applying a 7×7 median filter (with Frobenius norm) as can be seen in (c).

In the following experiments we show the application of matrix-valued median filtering in the context of texture analysis. The synthetic test image in Fig. 2 (a) contains two oriented texture regions separated by a sharp edge. We compute the gradient ∇u at each pixel using a 3×3 DoG filter and the outer product matrix $\nabla u \nabla u^\top$ (of rank one) which estimates the local orientation. We visualise the orientations of the principal eigenvectors by mapping angles directly into grey-values (b). The direct transitions between black and white at image boundaries and along the texture edge are caused by the fact that black and white in fact represent orientations which are very close to each other because of the cyclic nature of angles. Median filtering of the outer product matrices yields new matrix fields. We visualise their orientation in the same way as before (c–e). Juxtaposing orientation fields obtained with Frobenius norm (d) and spectral norm (e) shows

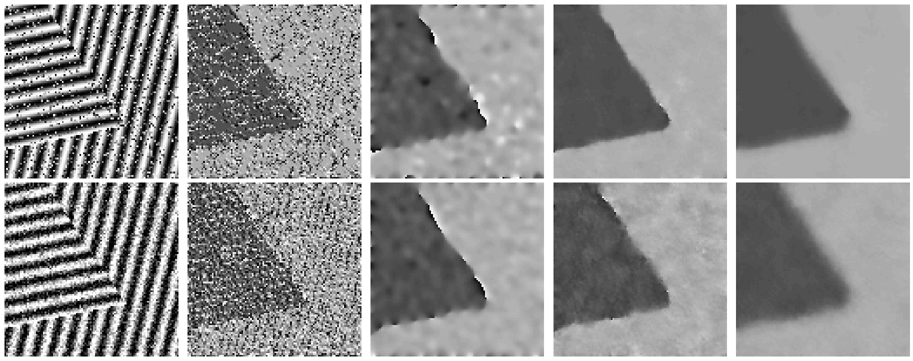

Fig. 3. Top, left to right: (a) Test image with 20 % impulse noise. **(b)** Orientation field of (a). **(c)** Structure tensor orientation obtained by Gaussian smoothing of the outer product matrices with standard deviation 19. **(d)** Same after median filtering with Frobenius norm and disk-shaped structure element of diameter 9. **(e)** Median filtering of (a) with Frobenius norm and disk-shaped structure element of diameter 19. **Bottom, left to right: (f)** Test image perturbed by Gaussian noise of standard deviation 0.2 (where grey-values vary between 0 and 1). **(g)** Orientation field of (f). **(h)** Structure tensor orientation as in (c). **(i)** Median filtering as in (d). **(k)** Median filtering as in (e).

that the two distance measures yield no significantly different results. In the following we therefore restrict ourselves to the Frobenius norm.

Let us turn now to investigate orientation estimation in noisy images. Fig. 3 shows two noisy versions of the test image (Fig. 2 (a)) together with their local orientation estimates. Each orientation matrix field is then smoothed by matrix-valued median filtering. For comparison, we show also the orientation of the standard structure tensor obtained by Gaussian smoothing of the orientation matrices. While in (a–d) impulse noise is shown where the grey values at 20 % of all pixels have been replaced with random values from $[0, 1]$, images (e–h) show perturbation by Gaussian noise. While for impulse noise the median filter yields denoises orientation better and also better preserves the discontinuity, the removal for Gaussian noise is still less satisfactory. Increasing the size of the structure element reduces noise at the cost of blurring also the discontinuity and rounding corners, see Fig. 4 (a, e). We are therefore led to propose two modifications which improve the quality of the orientation estimation by median filtering in the case of noisy images.

The first modification is to normalise the gradients before computing the outer products and applying the median filter. This leads to a sharper representation of the discontinuity in the case of impulse noise as shown in Fig. 4 (a, b). With Gaussian noise, however, only a marginal improvement is achieved (f, g).

Our second modification is to iterate median filtering. Although it can be combined with the normalisation procedure, we demonstrate its application to the non-normalised outer product matrices. While the improvement achieved for

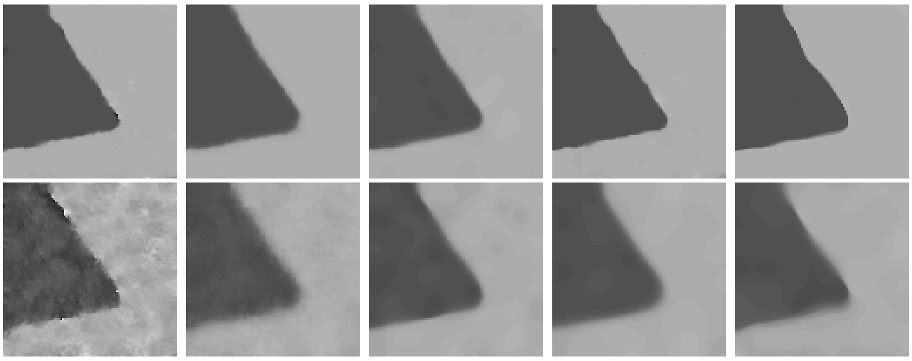

Fig. 4. Top, left to right: (a) Median filtering of local orientation derived from normalised gradients of Fig. 3 (a) with Frobenius norm and disk-shaped structure element of diameter 9. **(b)** As (a) but with structure element of diameter 19. **(c)** Median filtering without normalisation of gradients as in Fig. 3 (d), iterated four times. **(d)** Median filtering with normalisation, structure element of diameter 5, iterated five times. **(e)** Orientation estimate from Boomgaard–Weijer's robust structure tensor, parameters (see [16]) $m = 0.05$, $s = 5$. **Bottom, left to right:** Filtering of Fig. 3 (e). **(f)** Median filtering (Frobenius norm) with normalisation, structure element of diameter 9. **(g)** Same with diameter 19. **(h)** Filtering as in Fig. 3 (h), iterated five times. **(i)** Median filtering with normalisation, structure element of diameter 15, iterated four times. **(k)** Boomgaard–Weijer's robust structure tensor, $m = 0.05$, $s = 9$.

the impulse-noise image is comparable to that of the normalisation procedure, see Fig. 4 (d), it outperforms it in the case of Gaussian noise as shown in (h). Compared to a single median filtering step with the same structure element, corners are rounded slightly more but less than with a single step with larger structure element. The sharpness of the discontinuity is not reduced considerably compared to a single iteration while noise is removed more effectively.

The smoothing of outer product matrices by iterated median filtering can be interpreted as computation of a robust structure tensor. When computing classical structure tensors as in Fig. 3 (b, g), the outer product matrices are smoothed by Gaussian filtering, thus by linear diffusion. Nonlinear structure tensors as established by Weickert and Brox [17] use instead nonlinear diffusion to achieve a better representation of orientation discontinuities. The robust structure tensor introduced by van den Boomgaard and van der Weijer [16] smoothes the outer product matrices by minimising an energy in which a function ϱ is applied to matrix distances. In the case $\varrho(s) = s$, their robust structure tensor is similar to a single step of median filtering, with the difference that not a sharp structure element but Gaussian weights are used. In our filtering procedure, *iterated* matrix-valued median filtering takes the role of the smoothing process. This is primarily a change in theoretic perspective since it means that linear filtering is replaced by robust filtering more consequently. Orientation estimates obtained by Boomgaard and Weijer's method are shown in Fig. 4 (e, k). In Table 1, we

Table 1. Average angular errors (AAE) for orientation estimation. Values in brackets are method-specific parameters: for median filtering, diameter of structure element and number of iterations; for Boomgaard–Weijer method, m and s (see [16]).

Method	AAE undisturbed	AAE impulsive noise	AAE Gaussian noise
gradient direction	3.387°	20.612°	31.429°
Frobenius median	1.591° (7, 1)	1.914° (9, 4)	3.207° (9, 4)
Frobenius median, norm.	1.312° (7, 1)	1.655° (5, 5)	3.434° (15, 4)
Boomgaard–Weijer	1.634° (0.1, 3)	1.489° (0.05, 5)	3.657° (0.05, 9)

compare the different orientation estimation methods by their average angular errors. As the experiments show, both types of robust structure tensors yield comparable results.

6 Conclusion and Further Work

In this paper, we have introduced a novel numerical algorithm for the computation of matrix-valued median filters which in their basic form have been introduced in [18], and for closely related mid-range filters. This algorithm is based on convex programming ideas. It uses interior-point techniques to compute the filtered matrices as global optima. Further, we have demonstrated the application of matrix-valued median filtering as a discontinuity-preserving denoising technique for orientation data obtained from grey-value images with oriented textures. It has become evident that median filtering of local orientation matrices is an attractive alternative to Gaussian–smoothed structure tensors. It also leads in a natural way to a concept of robust structure tensor in which matrix-valued median filtering takes the role of the smoothing process.

Future work will include the embedding of matrix-valued median filtering into texture segmentation procedures. Moreover, it will address a better understanding of the properties of the so defined type of robust structure tensor and its comparison to the already existing concepts of nonlinear and robust structure tensors.

References

1. J. Astola, P. Haavisto, and Y. Neuvo. Vector median filters. Proc. IEEE, vol. 78 no. 4, 678–689, 1990
2. T. L. Austin. An approximation to the point of minimum aggregate distance. Metron, 19, 10–21, 1959
3. M. Barni, F. Buit, F. Bartolini, and V. Cappellini. A quasi-Euclidean norm to speed up vector median filtering. IEEE Trans. Image Processing, vol. 9 no. 10, 1704–1709, 2000
4. T. Brox, M. Rousson, R. Deriche, J. Weickert. Unsupervised segmentation incorporating colour, texture, and motion. In N. Petkov, M. A. Westenberg, editors,

Computer Analysis of Images and Patterns, vol. 2756 of *Lecture Notes in Computer Science,* pages 353–360, Berlin, 2003, Springer
5. S. Boyd, and L. Vandenberghe. *Convex Optimization.* 2004, Cambridge University Press
6. T. Brox, J. Weickert. Nonlinear matrix diffusion for optic flow estimation. In L. Van Gool, editor, *Pattern Recognition,* vol. 2449 of *Lecture Notes in Computer Science,* pages 446–453, Berlin, 2002, Springer
7. V. Caselles, G. Sapiro, and D. H. Chung. Vector median filters, inf-sup operations, and coupled PDE's: Theoretical Connections. J. Mathematical Imaging and Vision 8, 109–119, 2000
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. S. P. Fekete, J. S. B. Mitchell, K. Beurer. On the continuous Fermat–Weber problem. To appear in Operations Research. Preprint: http://arxiv.org/abs/cs/0310027 (last visited: Sept. 27, 2004)
10. G. H. Granlund and H. Knutsson. *Signal Processing for Computer Vision.* Kluwer, Dordrecht, 1995
11. J. Keuchel, C. Schnörr, C. Schellewald, and D. Cremers. Binary partitioning, perceptual grouping, and restoration with semidefinite programming. IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 25, no. 11, pages 1364–1379, 2003
12. N. Megiddo. The weighted Euclidean 1-center problem. Mathematics of Operations Research, vol. 8, no. 4, pages 498–504, 1983
13. C. Pierpaoli, P. Jezzard, P. J. Basser, A. Barnett, and G. Di Chiro. Diffusion tensor MR imaging of the human brain. *Radiology,* 201(3) pages 637–648, Dec. 1996
14. M. Rousson, T. Brox, R. Deriche. Active unsupervised texture segmentation on a diffusion based feature space. Technical report no. 4695, Odyssée, INRIA Sophia-Antipolis, France, 2003
15. D. R. Seymour. Note on Austin's "An approximation to the point of minimum aggregate distance". Metron, 28, 412–421, 1970
16. R. van den Boomgaard, and J. van de Weijer. Least squares and robust estimation of local image structure. In L. D. Griffin and M. Lillholm, editors, *Scale Space Methods in Computer Vision,* volume 2695 of *Lecture Notes in Computer Science,*, pages 237–254, Berlin, 2003, Springer
17. J. Weickert, 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,* vol. 313 of *Contemporary Mathematics,* pages 251–268, Providence, 2002, AMS
18. M. Welk, C. Feddern, B. Burgeth, and J. Weickert. Median filtering of tensor-valued images. In B. Michaelisi and G. Krell, editors, *Pattern Recognition,* volume 2781 of *Lecture Notes in Computer Science,*, pages 17–24, Berlin, 2003, Springer