

A Higher-Order Structure  
Tensor

Thomas Schultz, Joachim Weickert,  
and Hans-Peter Seidel

MPI-I-2007-4-005

July 2007

## Authors' Addresses

Thomas Schultz  
Max-Planck-Institut für Informatik  
Stuhlsatzenhausweg 85  
66123 Saarbrücken  
Germany  
`schultz@mpi-inf.mpg.de`

Joachim Weickert  
Mathematical Image Analysis Group  
Faculty of Mathematics and Computer Science  
Saarland University, Building E 2 4  
66041 Saarbrücken  
Germany  
`weickert@mia.uni-saarland.de`

Hans-Peter Seidel  
Max-Planck-Institut für Informatik  
Stuhlsatzenhausweg 85  
66123 Saarbrücken  
Germany  
`hpseidel@mpi-inf.mpg.de`

## Acknowledgements

We would like to thank Holger Theisel, who is with Bielefeld University, for discussions at all stages of this project. Discussions with Torsten Langer, who is with the MPI Informatik, helped in developing parts of the “mathematical toolbox” in Chapter 4.

Our implementation uses the CImg library by David Tschumperlé, available from <http://cimg.sf.net/>.

This research has partially been funded by the Max Planck Center for Visual Computing and Communication (MPC-VCC).

## **Abstract**

Structure tensors are a common tool for orientation estimation in image processing and computer vision. We present a generalization of the traditional second-order model to a higher-order structure tensor (HOST), which is able to model more than one significant orientation, as found in corners, junctions, and multi-channel images. We provide a theoretical analysis and a number of mathematical tools that facilitate practical use of the HOST, visualize it using a novel glyph for higher-order tensors, and demonstrate how it can be applied in an improved integrated edge, corner, and junction detector.

## **Keywords**

Structure Tensor, Higher-Order Tensors, Corner Detection, Multivalued Images

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>A Higher-Order Structure Tensor</b>	<b>4</b>
<b>3</b>	<b>Glyphs for Higher-Order Tensors</b>	<b>6</b>
3.1	Generalized Ellipses as Higher-Order Tensor Glyphs . . . . .	6
3.2	Experimental Results . . . . .	7
<b>4</b>	<b>A Mathematical Toolbox</b>	<b>10</b>
4.1	Efficient Representation . . . . .	10
4.2	Relation to Truncated Fourier Series . . . . .	10
4.3	Generalized Tensor Trace . . . . .	12
4.4	Generalized Eigenvector Decomposition . . . . .	13
4.5	Extrema of the Contrast Function . . . . .	14
4.5.1	Accelerating the Brute Force Method . . . . .	15
4.5.2	A Faster Method . . . . .	16
4.5.3	Implementation of the Faster Method . . . . .	17
<b>5</b>	<b>Integrated Edge and Junction Detection</b>	<b>18</b>
<b>6</b>	<b>Conclusions and Future Work</b>	<b>21</b>

# 1 Introduction

The second-order structure tensor, formed as the outer product of the image gradient with itself, is a common tool for local orientation estimation. Since it was first introduced for edge and corner detection [9], it has been applied to a wide variety of problems in image processing and computer vision, including optic flow estimation [2], image diffusion [23], texture segmentation [21], image inpainting [22], and image compression [10].

Two popular extensions are its generalization to vector- and tensor-valued images, which goes back to an idea of Di Zenzo [7], and the introduction of nonlinear local averaging [24], which lead to nonlinear structure tensors [4].

It is a known limitation of the traditional structure tensor that it can only represent a single dominant orientation. Recently, there have been attempts to overcome this: Arseneau and Cooperstock [1] have placed second-order structure tensors in discrete directional bins and derived parameters of multimodal *directional distribution functions* from them. Their work concentrates on lifting the constraint of antipodal symmetry (i.e., they treat direction  $\mathbf{v}$  differently than direction  $-\mathbf{v}$ ), a property which our approach preserves. Moreover, they use the structure tensors only as an intermediate representation, finally reducing them to two scalar parameters for each direction.

Herberthson et al. [12] have used outer products to handle pairs of orientations. However, their approach is specific to the case of two orientations: It neither generalizes to more than two directions, nor does it indicate cases in which representing a single orientation is sufficient.

In our present work, we present a generalization of the second-order structure tensor to a higher-order tensor model, which is able to capture the orientations of more complex neighborhoods, for example corners, junctions, and multivalued images. The tensor order allows to specify the maximum complexity the structure tensor can represent and can be chosen based on the requirements of a given application.

This report is structured as follows: Chapter 2 introduces our new higher-

order structure tensor (HOST). In Chapter 3, we present a novel glyph for higher-order tensors and use it to visualize first experimental results. A theoretical analysis and a number of mathematical tools that help to use the HOST in practice are presented in Chapter 4. They include an efficient representation of the structure tensor, an alternative representation as a truncated Fourier Series, a generalization of the matrix trace and the eigenvector decomposition, and an algorithm to extract contrast extrema from a higher-order tensor representation. Chapter 5 shows a proof-of-concept application, in which the HOST is used for an improved integrated edge and junction detection. Finally, Chapter 6 concludes this chapter and points out directions of future research.

## 2 A Higher-Order Structure Tensor

The standard second-order structure tensor  $\mathbf{J}$  is given by the outer product of the image gradient  $\nabla f$  with itself [9]:

$$\mathbf{J} := \nabla f \nabla f^T \quad (2.1)$$

It is typically averaged over a neighborhood to obtain a descriptor of local image structure. Our generalization to a higher-order tensor  $\mathcal{J}$  simply repeats the outer product. For a vector  $\mathbf{v}$ , taking the outer product with itself  $l$  times will be written  $\mathbf{v}^{\otimes l}$ . It yields an order- $l$  tensor, indexed by  $\{i_1, i_2, \dots, i_l\}$ :

$$(\mathbf{v}^{\otimes l})_{i_1 i_2 \dots i_l} := v_{i_1} \cdot v_{i_2} \cdots v_{i_l} \quad (2.2)$$

To ensure antipodal symmetry of the resulting tensor, we choose  $l$  to be even.

Interpretation of the higher-order structure tensor requires a *contrast function*  $J$ , which specifies the local contrast in a given direction. When the direction is represented by a unit-length vector  $\mathbf{u}$ ,  $J(\mathbf{u})$  is defined by repeating the inner tensor-vector product of  $\mathcal{J}$  and  $\mathbf{u}$  until a scalar is left, i.e.,  $l$  times. In  $n$  dimensions, this can be written as

$$J(\mathbf{u}) := \sum_{i_1=1}^n \sum_{i_2=1}^n \cdots \sum_{i_l=1}^n (\mathcal{J})_{i_1 i_2 \dots i_l} u_{i_1} u_{i_2} \cdots u_{i_l} \quad (2.3)$$

This definition is inspired by a work of Özarslan and Mareci [18], who have derived a diffusivity function  $D(\mathbf{u})$  in the same manner from higher-order diffusion tensors  $\mathcal{D}$  in the context of generalized diffusion tensor magnetic resonance imaging (DT-MRI). For a second-order structure tensor,  $J$  is unimodal, which reflects the fact that it is suitable to model only one dominant orientation. For higher orders,  $J$  can become multimodal, which allows a more accurate representation of corners, junctions, and multi-images.

We consider it a sensible requirement that the values of the contrast function should remain comparable, independent of the tensor order that we use. When evaluated in direction of the gradient, the contrast function yields the squared gradient magnitude in the second-order case. However, taking the outer product  $l$  times would raise the gradient magnitude to the  $l$ th power. We compensate this by scaling the gradient vector beforehand. An order- $l$  structure tensor  $\mathcal{J}$  that reduces to the well-known second-order tensor  $\mathbf{J}$  for  $l = 2$  is then given by

$$\mathcal{J} := \left( \frac{\nabla f}{|\nabla f|^{\frac{l-2}{l}}} \right)^{\otimes l} \quad (2.4)$$

In some applications, it is beneficial to have a contrast function that gives the non-squared gradient magnitude [3]. This can be achieved by replacing the exponent  $\frac{l-2}{l}$  by  $\frac{l-1}{l}$  in Equation (2.4).

# 3 Glyphs for Higher-Order Tensors

In the literature on higher-order diffusion tensors, *generalized Reynolds glyphs* constitute the only glyph-based visualization technique [18, 15]. Let  $S$  be the unit sphere and  $J$  the contrast function as defined above. Then, these glyphs are formed by the set of points  $\{J(\mathbf{u})\mathbf{u} \mid \mathbf{u} \in S\}$ , which directly depicts the contrast profile of the tensor. However, these glyphs have a round shape around their maxima, which makes their exact orientation difficult to see. To compensate this problem, Hlawitschka and Scheuermann [15] suggest to add arrows that point to the maxima.

## 3.1 Generalized Ellipses as Higher-Order Tensor Glyphs

While the diffusion ellipsoid is accepted as the standard glyph for second-order diffusion tensors, the Reynolds glyph does *not* reduce to it for  $l = 2$ . Since the tensor ellipsoid can be constructed by transforming the unit sphere under the linear mapping induced by the tensor, it is natural to generalize it by taking the inner tensor-vector product  $l - 1$  times, until a vector is left. We denote the inner product  $\mathcal{J} \bullet \mathbf{u}$ , where

$$(\mathcal{J} \bullet \mathbf{u})_{i_1 i_2 \dots i_{l-1}} := \sum_{i_l=1}^n (\mathcal{J})_{i_1 i_2 \dots i_l} u_{i_l} \quad (3.1)$$

and use the shortcut notation  $\mathcal{J} \overset{l-1}{\bullet} \mathbf{u}$  to indicate that we repeat it  $l-1$  times. Then, the surface of our glyph is given by the points  $\{\mathcal{J} \overset{l-1}{\bullet} \mathbf{u} \mid \mathbf{u} \in S\}$ .

As the 2D examples in Figure 3.1 illustrate, the extrema of the generalized ellipses coincide with the extrema of the Reynolds glyphs. However, they

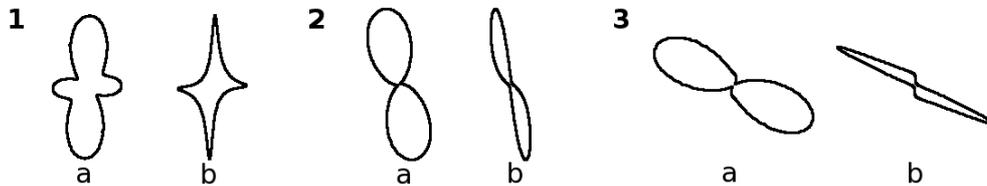


Figure 3.1: Three tensors of order six, visualized with Reynolds glyphs (a) and our generalized ellipses (b). In (b), maxima of the contrast function appear more localized.

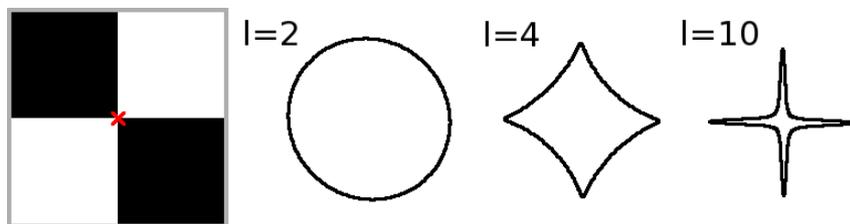
develop sharp features around the maxima, at the cost of a smoother shape around the minima. In examples two and three, the generalized ellipses immediately make clear that the respective tensors are not axially symmetric, a fact which the Reynolds glyph may not reveal at first glance. Since we are generally more interested in the maxima than in the minima of the contrast function, we will use the new glyphs in the remainder of this chapter.

## 3.2 Experimental Results

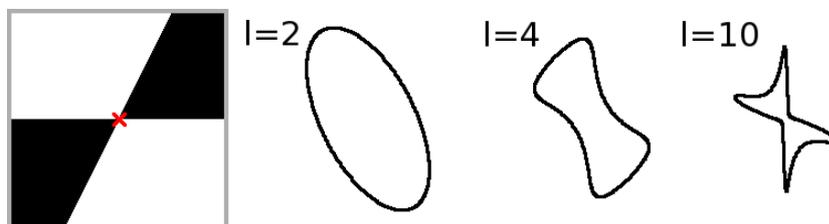
We will now present some experiments to confirm that higher-order structure tensors indeed give a more accurate representation of junctions and multivalued images. Our first experiment uses simple junctions in synthetic grayscale images. Derivatives are calculated by convolution with a derivative-of-Gaussian filter ( $\sigma = 0.7$ ). After HOSTs of different order  $l$  have been computed, their information is propagated to a local neighborhood by convolution with a Gaussian kernel ( $\rho = 1.4$ ).

Figure 3.2 shows the test images, with the position of the displayed structure tensor marked by a cross. The results show that a HOST of order  $l = 4$  is sufficient to represent two edges that cross orthogonally, while the traditional structure tensor ( $l = 2$ ) does not distinguish any particular direction. In the non-orthogonal case, the traditional model indicates a principal direction which does not correspond to any gradient found in the image. While the generalized ellipse of order four gives an impression of the involved directions, a clear separation of the maxima in the contrast profile now requires higher orders. However, Chapter 4.4 will show that the generalized eigenvectors of the HOST give a good approximation of the gradient directions already with  $l = 4$ .

The second experiment is based on a natural color image. Derivatives are now calculated channel-wise and according to the conventional generalization to multi-channel images, the HOSTs of the red, green, and blue color channels



(a) Orthogonal edges are clearly distinguished with order  $l = 4$ .



(b) For non-orthogonal edges, higher orders give more accurate representations.

Figure 3.2: Two junctions in grayscale images and the corresponding structure tensors. For orders  $l > 2$ , the directions of the meeting edges can be represented.

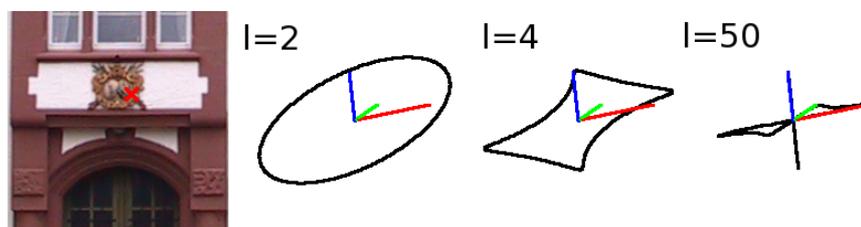


Figure 3.3: In a color image, the channel-wise gradients may point into different directions. Higher-order structure tensors can be used to model this situation accurately.

are added. In this case, we do not propagate the structure information ( $\rho = 0$ ).

For comparison, Figure 3.3 also shows the gradients of the individual color channels. Again, the structure tensor of order four already gives a much better impression of the dominant directions than the traditional model. To demonstrate the feasibility of going to very high tensor orders, we also present the representation with  $l = 50$ .

# 4 A Mathematical Toolbox

## 4.1 Efficient Representation

An order- $l$  tensor in  $n$  dimensions has  $n^l$  tensor channels, which becomes impractical already for moderate  $l$ . However, higher-order structure tensors are totally symmetric, i.e., invariant under permutation of their indices. This reduces the number of independent channels to  $N = \binom{n+l-1}{l}$ , which means merely linear growth for  $n = 2$  ( $N = l + 1$ ) and quadratic growth for  $n = 3$ .

With some additional notation, it is possible to evaluate  $J(\mathbf{u})$  directly from this non-redundant representation: Call the  $i$ th non-redundant element  $[\mathcal{J}]_i$ , stored in a zero-based linear array  $[\mathcal{J}]$ . Let  $\nu_{i,k} \in \{0, 1, \dots, l\}$  denote the number of times  $k \in \{1, 2, \dots, n\}$  appears as an index of the  $i$ -th element. The multiplicity of element  $i$ , denoted  $\mu_i$ , is the number of times it appears as a channel of the original tensor. For  $n = 2$ ,  $\mu_i = \binom{l}{\nu_{i,1}}$ , for  $n = 3$ ,  $\mu_i = \binom{l}{\nu_{i,1}} \binom{l-\nu_{i,1}}{\nu_{i,2}}$ . Then, Equation (2.3) can be rewritten as

$$J(\mathbf{u}) = \sum_{i=0}^{N-1} \mu_i [\mathcal{J}]_i u_1^{\nu_{i,1}} u_2^{\nu_{i,2}} \dots u_n^{\nu_{i,n}} \quad (4.1)$$

For  $n = 2$ , we chose indices such that  $\nu_{i,1} = l - i$  (e.g.,  $[\mathcal{J}_{1111}, \mathcal{J}_{1112}, \dots]$ ).

## 4.2 Relation to Truncated Fourier Series

From generalized DT-MRI, it is known that using a higher-order tensor model in 3D is equivalent to approximating the diffusivity profile with a truncated Laplace series [18]. We will now show that the corresponding result in 2D is a relation of higher-order tensors to truncated Fourier Series. This fact will serve as the basis of the methods in Chapters 4.3 and 4.5.

Consider a Fourier Series, truncated after order  $l$ :

$$f(\phi) = \frac{1}{2}a_0 + \sum_{k=1}^l a_k \cos(k\phi) + \sum_{k=1}^l b_k \sin(k\phi) \quad (4.2)$$

Setting  $a_k := b_k := 0$  for odd  $k$  leaves a  $l + 1$  dimensional vector space of functions. For  $n = 2$ , Equation (4.1) can be rewritten in polar coordinates:

$$J(\phi) = \sum_{i=0}^l [\mathcal{J}]_i \binom{l}{i} \cos^{l-i} \phi \sin^i \phi \quad (4.3)$$

Let us regard  $[\mathcal{J}]_i$  as coefficients and  $\binom{l}{i} \cos^{l-i} \phi \sin^i \phi$  as basis functions. We will now show that these basis functions span the same space as the truncated Fourier Series.

**Proof** by induction on order  $l$ . Let  $\{\mathbf{f}_k\}$  denote the basis functions of a truncated Fourier Series in which only even multiples of  $\phi$  are allowed:

$$\mathbf{f}_k := \begin{cases} 0.5 & \text{if } k = 0 \\ \cos((k+1)\phi) & \text{if } k \text{ odd} \\ \sin(k\phi) & \text{if } k \text{ even } (k \neq 0) \end{cases}$$

Likewise,  $\mathbf{t}_k^l$  is the  $k$ -th basis function of an order- $l$  tensor:

$$\mathbf{t}_k^l := \binom{l}{k} \cos^{l-k} \phi \sin^k \phi$$

For  $l = 0$ , both the Fourier Series and the tensor basis represent constant functions and  $\mathbf{f}_0 = 0.5\mathbf{t}_0^0$ . Assume that the functions that can be represented using  $\{\mathbf{f}_k\}$  with  $k \leq l$  are equivalent to the functions represented by  $\{\mathbf{t}_k^l\}$ . Further, assume that we know how to express the Fourier basis in terms of the tensor basis. Then, we can show that the same assumption also holds for  $l + 2$ : Observe that

$$\begin{aligned} \cos^{l-i} \phi \sin^i \phi &= (\cos^2 \phi + \sin^2 \phi) \cos^{l-i} \phi \sin^i \phi \\ &= \cos^{l+2-i} \phi \sin^i \phi + \cos^{l-i} \phi \sin^{i+2} \phi \end{aligned}$$

and that the latter functions are proportional to functions in  $\{\mathbf{t}_k^{l+2}\}$ . Thus, we can express the first  $l + 1$  Fourier basis functions in terms of  $\{\mathbf{t}_k^{l+2}\}$  by replacing each occurrence of  $\mathbf{t}_k^l$  in their known representation by

$$\mathbf{t}_k^l = \frac{(l+2-k)(l+1-k)}{(l+2)(l+1)} \mathbf{t}_k^{l+2} + \frac{(k+2)(k+1)}{(l+2)(l+1)} \mathbf{t}_{k+2}^{l+2}$$

$l = 2$	$a_0 = [\mathcal{J}]_0 + [\mathcal{J}]_2$	
	$a_2 = \frac{1}{2}[\mathcal{J}]_0 - \frac{1}{2}[\mathcal{J}]_2$	$b_2 = [\mathcal{J}]_1$
$l = 4$	$a_0 = \frac{3}{4}[\mathcal{J}]_0 + \frac{3}{2}[\mathcal{J}]_2 + \frac{3}{4}[\mathcal{J}]_4$	
	$a_2 = \frac{1}{2}[\mathcal{J}]_0 - \frac{1}{2}[\mathcal{J}]_4$	$b_2 = [\mathcal{J}]_1 + [\mathcal{J}]_3$
	$a_4 = \frac{1}{8}[\mathcal{J}]_0 - \frac{3}{4}[\mathcal{J}]_2 + \frac{1}{8}[\mathcal{J}]_4$	$b_4 = \frac{1}{2}[\mathcal{J}]_1 - \frac{1}{2}[\mathcal{J}]_3$

Table 4.1: Relation of Fourier coefficients and tensor components for orders  $l = 2$  and  $l = 4$ . A method to compute these relations for general  $l$  is given in the text.

It remains to be shown how to express  $\mathbf{f}_{l+1}$  and  $\mathbf{f}_{l+2}$  in terms of  $\{\mathbf{t}_k^{l+2}\}$ . For this, we use trigonometric identities for multiple angles:

$$\begin{aligned}
\mathbf{f}_{l+1} &= \cos((l+2)\phi) \\
&= \sum_{i=0}^{l/2+1} (-1)^i \binom{l+2}{2i} \cos^{l+2-2i} \phi \sin^{2i} \phi = \sum_{i=0}^{l/2+1} (-1)^i \mathbf{t}_{2i}^{l+2} \\
\mathbf{f}_{l+2} &= \sin((l+2)\phi) \\
&= \sum_{i=0}^{l/2} (-1)^i \binom{l+2}{2i+1} \cos^{l+1-2i} \phi \sin^{2i+1} \phi = \sum_{i=0}^{l/2} (-1)^i \mathbf{t}_{2i+1}^{l+2}
\end{aligned}$$

■

Our proof is constructive in the sense that it implies a recursive method to construct a change-of-basis matrix. For reference, Table 4.1 presents the relations for  $l = 2$  and  $l = 4$ .

### 4.3 Generalized Tensor Trace

The second-order tensor trace has been used as a substitute of the squared gradient magnitude [8]. For the higher-order case, Özarslan et al. [19] have proposed a generalized trace operation “gentr” in 3D, which is based on integrating  $J$  over the unit hemisphere  $\Omega$  and reduces to the standard matrix trace for  $l = 2$ :

$$\text{gentr}(\mathcal{J}) := \frac{3}{2\pi} \int_{\Omega} J(\mathbf{u}) d\mathbf{u} \quad (4.4)$$

In the 2D case,  $\Omega$  is one half of the unit circle and the normalization factor  $\frac{3}{2\pi}$  is to be replaced with  $\frac{2}{\pi}$ . Since the generalized trace of an order- $l$  2D tensor

equals its Fourier coefficient  $a_0^l$ , we can use the results from Chapter 4.2 to verify that

$$\text{gentr}(\mathcal{J}) = a_0^l = 2 \sum_{i=0}^{l/2} [\mathcal{J}]_{2i} \frac{(l-1)!!}{(l-2i)!! \cdot (2i)!!} \quad (4.5)$$

where  $!!$  is the *double factorial*, i.e., the product of integers in steps of two.

In the definition of  $\mathcal{J}$ , we scaled the gradient magnitude such that the maximum value of  $J$  is invariant to the tensor order. However, maxima become narrower with increasing  $l$ , so the generalized trace decreases. It follows from Equation (4.5) that the generalized trace of an order- $l$  structure tensor equals

$$\text{gentr}(\mathcal{J}) = 2 \frac{(l-1)!!}{l!!} |\nabla f|^2 \quad (4.6)$$

## 4.4 Generalized Eigenvector Decomposition

Many applications of the second-order structure tensor depend on its spectral decomposition into eigenvectors and eigenvalues (e.g., [23, 16, 22, 8, 10]). In this section, we introduce the CAND (Canonical Decomposition), which can be regarded as a generalized eigendecomposition for higher-order tensors and has first been studied by Hitchcock [13, 14]. A review in the context of higher-order statistics and some new results are given by Comon et al. [6, 5].

We will concentrate on the symmetric CAND (sCAND), which decomposes a symmetric order- $l$  tensor  $\mathcal{J}$  into a sum of  $r$  outer powers  $\mathbf{v}_i^{\otimes l}$  of unit vectors  $\mathbf{v}_i$ ,  $i \in \{1, 2, \dots, r\}$ , scaled with  $\lambda_i$ :

$$\mathcal{J} = \sum_{i=1}^r \lambda_i \mathbf{v}_i^{\otimes l} \quad (4.7)$$

For  $l = 2$ , Equation (4.7) reduces to the spectral decomposition, where  $\lambda_i$  are the eigenvalues and  $\mathbf{v}_i$  the eigenvectors. It can be shown that any symmetric higher-order tensor  $\mathcal{J}$  has a sCAND [5]. In analogy to the matrix rank, the symmetric rank  $R_S$  of  $\mathcal{J}$  is defined as the smallest number  $r$  for which a sCAND exists. In dimension  $n = 2$ , it holds that  $R_S \leq l$  [6].

The CAND is a current research topic and while some theoretical results have been obtained, practical algorithms for efficient computation are rare. Fortunately, Comon et al. [6] present an algorithm that works for  $n = 2$  and thus can be applied to our HOSTs. It is outside the scope of this chapter to review the full theory required to derive the algorithm. For our experiments,

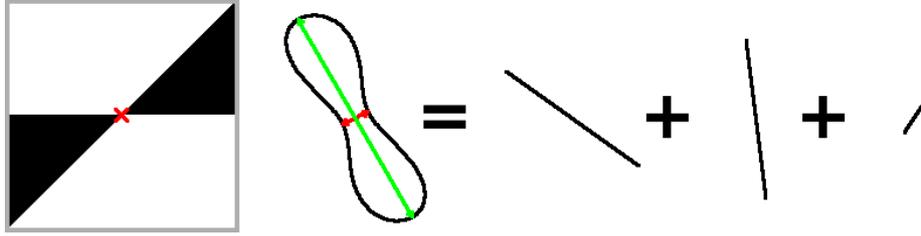


Figure 4.1: Generalized eigenvectors can be used to recover individual directions from a higher-order structure tensor.

we have simply re-implemented the MATLAB code given in [6] in C, using routines from LAPACK<sup>1</sup> and the Numerical Recipes [20].

The algorithm returns pairs of  $\lambda_i$  and  $\mathbf{v}_i$ , where the  $\mathbf{v}_i$  are not normalized. While it appears trivial to convert this result to the canonical form, the algorithm proves numerically unstable for vectors  $\mathbf{v}_i$  which are nearly aligned with the y-axis: In such cases,  $\lambda_i$  tends to zero, while the magnitude of  $\mathbf{v}_i$  tends to infinity. We work around this problem by reconstructing a tensor  $\mathcal{J}'$  only from those  $\mathbf{v}_i$  which have a reasonable magnitude. Then, the residual  $\tilde{\mathcal{J}} := \mathcal{J} - \mathcal{J}'$  can be rotated by  $90^\circ$  to obtain the remaining  $\mathbf{v}_i$ . Note that the tensor rotation only requires a simple permutation of its elements and some sign changes: In the array representation from Section 4.1, it is sufficient to reverse the array and to multiply entries  $[\mathcal{J}]_i$  with an odd index  $i$  by  $-1$ .

Figure 4.1 visually represents Equation (4.7) for a particular HOST. Even though the gradient directions in the neighborhood of the considered pixel are too close to be resolved in the contrast profile of an order-four tensor, they are well approximated by the two largest generalized eigenvectors.

Our prototype implementation found the SCAND of 160 000 order six structure tensors from a natural color image in around 2.5s on a 2GHz Athlon 64.

## 4.5 Extrema of the Contrast Function

A frequent problem when dealing with higher-order structure tensors will be to find the angles at which the contrast function  $J$  attains an extremum. In the second-order case, maxima and minima are given by the directions of the major and minor eigenvectors, respectively. However, the generalized eigenvectors do not in general coincide with maxima in the contrast function.

To find the extrema of the contrast function, we use the Fourier Series

<sup>1</sup><http://www.netlib.org/lapack/>

representation from Section 4.2. This makes it easy to take derivatives of  $J$ , since they are again Fourier Series of the same order, whose coefficients are straightforward to compute.

### 4.5.1 Accelerating the Brute Force Method

Extracting extrema requires to find angles  $\phi$  at which  $J'(\phi) = 0$ . The obvious method to find such points for an arbitrary differentiable function  $J$  is to sample its derivative with some resolution  $r$  to identify intervals in which it changes sign and to refine the result to a desired accuracy  $a$  by a binary search for the sign change on these intervals.

This method may miss pairs of extrema whose distance is less than the sampling resolution  $r$ . Fortunately, such pairs are usually only minor local variations in the contrast function, which are of no practical interest (cf. Figure 4.2), so we found  $r = 2^\circ$  sufficient for structure tensors of order six. For higher orders, denser sampling will be necessary, as peaks in the contrast function become sharper.

The computational cost of this method is dominated by the cost of evaluating the derivative  $J'$ . Evaluating Equation (4.2) directly involves  $l$  sines and  $l$  cosines. A recursive formulation exists which is known alternatively as Clenshaw's algorithm or as the Goertzel-Watt algorithm and requires only a single sine and cosine. It is given by the recursion rule

$$u_{l+1} = u_{l+2} = 0 \quad (4.8a)$$

$$u_r = f_r + 2u_{r+1} \cos \phi - u_{r+2} \quad \text{with } r = l, l-1, \dots, 1 \quad (4.8b)$$

where  $f_r := a_r$  if the sum of cosines is to be computed,  $f_r := b_r$  for the sum of sines. From  $u_1$  and  $u_2$ , the final result is determined as

$$\sum_{k=1}^l a_k \cos(k\phi) = u_1 \cos \phi - u_2 \quad \text{and} \quad \sum_{k=1}^l b_k \sin(k\phi) = u_1 \sin \phi \quad (4.9)$$

Gentleman [11] has shown that this method magnifies roundoff errors when evaluated near  $\phi = k\pi$  ( $k \in \mathbb{Z}$ ) and can produce unusable results for large orders  $l$ . Newbery [17] suggests to adaptively perform a phase shift of  $\pi/2$  to avoid such cases. However, an experimental comparison of direct evaluation, the original and the modified version of Clenshaw's algorithm indicates that the error is tolerable for the moderate values of  $l$  that occur in our context: For 160 000 structure tensors of order six from a natural color image, all methods produced identical results (with  $r = 2^\circ$  and  $a = (2^{-7})^\circ$ , at single precision). Even with  $l = 50$ , all methods gave the same extrema, now

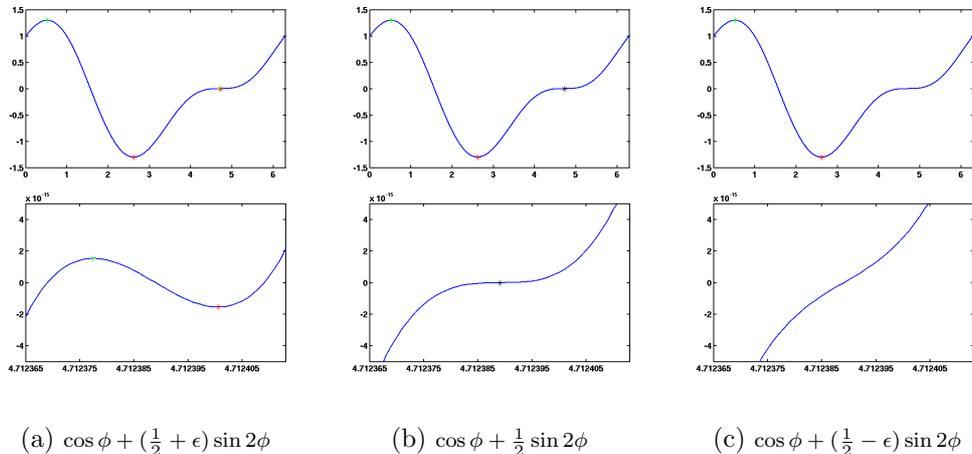


Figure 4.2: In a Fourier Series, a small  $\epsilon$  (here,  $\epsilon = 10^{-10}$ ) can make the difference between a pair of extrema, a saddle, and no stationary point. The top row shows one full period, while the bottom row gives a close-up of the affected extrema.

with a maximum angular deviation of  $(2^{-7})^\circ$ , in less than 1% of the cases. Thus, we chose the unmodified Clenshaw algorithm, which gave a speedup factor of 2.2 for  $l = 6$ .

## 4.5.2 A Faster Method

We will now present a more efficient algorithm which exploits the fact that even higher derivatives of  $J$  are easy to evaluate. The basic idea of the method is to expand  $J'$  into a Taylor series  $\tilde{J}'$ , which is terminated after degree three to obtain a polynomial that can be solved analytically in a numerically stable manner [20]. Then, the error bounds of the expansion define a corridor around the x-axis. For intervals in which  $\tilde{J}'(\phi)$  is outside this corridor, we can be sure that  $J'(\phi) \neq 0$ , i.e., they do not contain an extremum.

It is possible to recurse on the (now shorter) intervals in which  $\tilde{J}'(\phi)$  lies within the corridor; this allows to identify extrema which are so close that finding them with the brute force approach would be computationally infeasible. For example, finding the pair of extrema which is shown in Figure 4.2(a) would require sampling at  $r \approx (2^{-10})^\circ$ . In contrast, our algorithm needs only 10 Taylor expansions to identify all four extrema.

While it is nice to be able to find such small extrema, they are not im-

portant in our context, so we would prefer to find the major ones more efficiently. As intervals get down to a certain width, the computational effort of reducing them further via the Taylor expansion exceeds the cost of the brute force method. A simple way to combine both methods is to use the Taylor expansion once to cut down search space, and to run the brute force method on the remaining intervals. In the above experiment with order six tensors ( $r = 2^\circ$ ,  $a = (2^{-7})^\circ$ ), this gave a speedup of factor two, reducing the total time for processing all 160 000 tensors on a 2 GHz Athlon 64 processor to 2.2 seconds, which is about as fast as a SCAND on the same data. For smaller  $r$ , the speedup was much higher (factor 7.5 for  $r = 0.1^\circ$ ); for  $r = 5^\circ$ , it vanished.

### 4.5.3 Implementation of the Faster Method

Since we cannot expect a polynomial of degree three to reasonably approximate a sine of frequency  $l$  in an interval larger than  $\pi/l$ , we initially partition  $[0, 2\pi)$  into  $2l$  equal intervals for a Fourier Series of order  $l$ . For each of these intervals, the Taylor expansion is performed by evaluating the higher derivatives at its center  $\phi_0$ . The required Fourier coefficients are pre-computed once. For a third-order approximation, the error bound  $\Delta$  is

$$\Delta = \frac{J^{(5)}(\phi)}{4!}(\phi - \phi_0)^4 \quad (4.10)$$

for some  $\phi$  within the interval. Taking the order-five derivative in this expression is appropriate, since we approximate  $J'$ . To obtain a simple, safe estimate of  $\Delta$ , the Fourier coefficients  $a_k$  and  $b_k$  of  $J^{(5)}(\phi)$  are used to state that

$$J^{(5)}(\phi) \leq \sum_{k=1}^l \sqrt{a_k^2 + b_k^2} \quad (4.11)$$

and half the interval length is taken for  $(\phi - \phi_0)$ .

Now, the roots of  $\tilde{J}' - \Delta$  and  $\tilde{J}' + \Delta$  give the intersections with the upper and lower error bound, respectively. Starting from the value of  $\tilde{J}'$  at the left interval boundary, we can go through the sorted error bound intersections to determine the intervals in which  $\tilde{J}'$  is within the error corridor. Within some intervals,  $\tilde{J}'$  may lie fully inside or outside the corridor.

## 5 Integrated Edge and Junction Detection

Given a grayscale image, edges can be seen as lines across which local contrast is high. Then, corners are points in which edges bend sharply, and junctions are points in which two or more edges meet. Corner detection has been one of the first applications of the structure tensor [9]. More recently, Köthe [16] has presented an algorithm that uses the structure tensor for “integrated” edge and junction detection.

While this method is a considerable improvement over previous approaches, it finally decomposes the structure tensor field into an “edge” and a “junction” part, which are processed by separate algorithms to produce the respective edge and junction maps. Because of this, edges break down near junctions, it is not always clear which edges are connected by a junction, and isolated or duplicate junctions can occur (cf. Figure 5.1).

Unlike the second-order structure tensor, the HOST at a junction holds enough information to find the adjacent edges. This allows to extract edges, corners, and junctions in a single, fully integrated process. The fundamental idea is similar to tracking lines in higher-order tensor fields [15], except that we assume that edges are *orthogonal* to contrast maxima.

Figure 5.2 illustrates the process of tracking edges in a HOST field: Starting from a seed with locally maximal generalized trace, we integrate edges orthogonal to the major contrast indicated by the tensor (red arrow). When a secondary peak in the contrast function attains a local maximum, we insert a junction and start new edges from it (yellow arrows).

While standard hyperstreamline integration techniques can be applied by interpolating the higher-order tensors channel-wise, we are typically not interested in edge maps that have a greater resolution than the structure tensor grid (note, however, that it is advisable to sample the structure tensors at twice the image resolution [16]).

Thus, we implemented a simple tracking algorithm that works on the

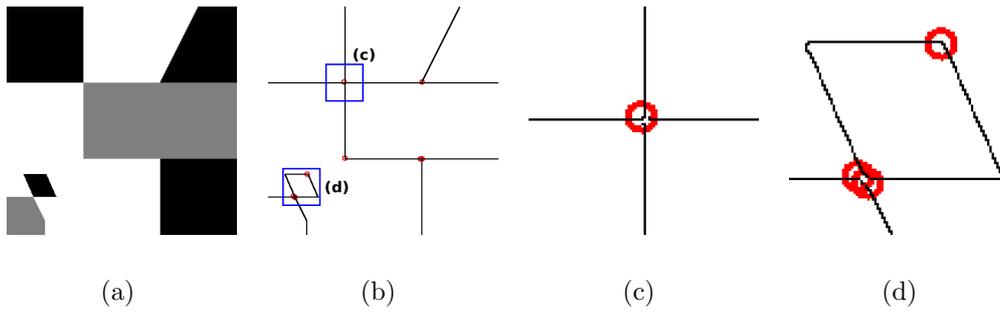


Figure 5.1: Subfigure (b) shows edges and junctions extracted from (a) using second-order structure tensors. (c) and (d) illustrate that edges break down as the tensor becomes isotropic near junctions (marked with circles).

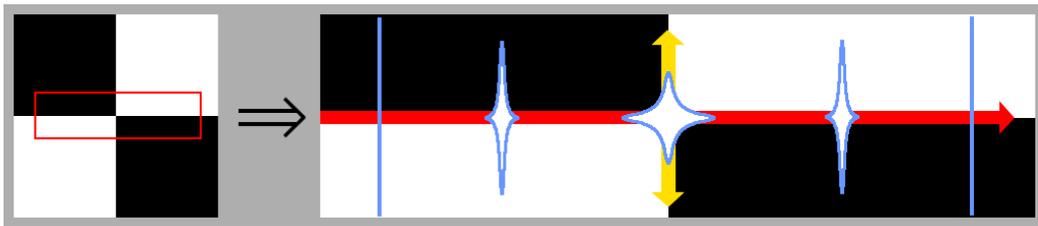


Figure 5.2: Scheme of the tracking process which extracts edges and junctions simultaneously.

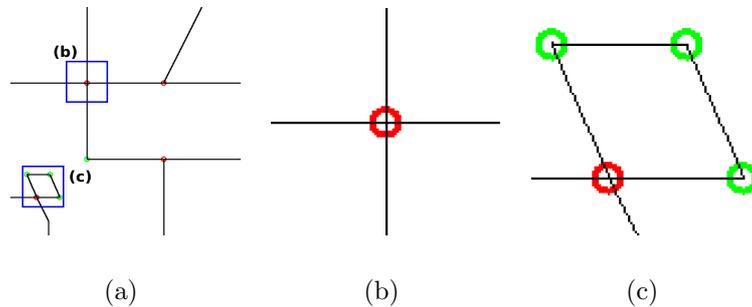


Figure 5.3: With higher-order tensors, edges can be traced through and assigned to junctions, which allows easy classification of corners (green) and junctions (red).

given grid. For each edge, it outputs a list of sub-pixels that belong to it, and each junction holds references to the edges it connects. This information allows an easy classification of corners and junctions, which are shown in different colors in Figure 5.3, based on the number of incident edges.

Integration over a Gaussian neighborhood propagates the contrast from edges beyond corner points, which leads to short “phantom” edges that make corners appear as junctions. To avoid such artifacts, we reject edges along which the HOST indicates multiple directions in all sub-pixels. Also, edges may reach existing junctions through a cycle (like in Figure 5.3(c)), and we need to connect them explicitly to such junctions.

Figure 5.3 presents a result of our method with HOSTs of order  $l = 6$ . The present algorithm gives superior results on synthetic images and demonstrates the applicability of the methods introduced in Chapter 4. However, a version which is robust enough for natural images has to be left for future research. In particular, we plan to explore the potential of using generalized eigenvectors instead of contrast maxima to steer the tracking.

Our implementation of Köthe’s algorithm took 0.3s for the shown example image (including his anisotropic averaging), while our method needed 0.9s, again including computation and integration of the tensors. The original image size was  $300 \times 300$  pixels, giving a  $599 \times 599$  sub-pixel structure tensor field.

## 6 Conclusions and Future Work

In the present work, we have shown how higher-order tensors can be used to represent the average of orientations in greater detail than it is possible using traditional second-order structure tensors. We have introduced the notions, definitions and mathematical tools required to work with such higher-order structure tensors efficiently and to visualize them appropriately.

While Chapter 5 demonstrates the advantages of the HOST for integrated edge and junction detection, it is intended as a proof of concept for the introduced methods, not as the ultimate goal of our research. Consequently, our next step will be to explore the potential of the HOST in several image processing and computer vision applications. We have already conducted some promising experiments on using HOSTs for texture segmentation and for steering image diffusion.

All results in this chapter are in 2D. Some of the theory (Chapter 2) and practical methods (Chapters 3 and 4.1) easily carry over to three dimensions, or equivalents can be taken from the literature (Chapters 4.2 and 4.3). However, efficient methods for the SCAND with  $n = 3$  and for finding the maxima of 3D contrast functions are still missing and probably require substantial research.

# Bibliography

- [1] S. Arseneau and J. R. Cooperstock. An improved representation of junctions through asymmetric tensor diffusion. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. V. Nefian, M. Gopi, V. Pascucci, J. Zara, J. Molineros, H. Theisel, and T. Malzbender, editors, *Advances in Visual Computing*, volume 4291 of *Lecture Notes in Computer Science*, pages 363–372. Springer, 2006.
- [2] J. Bigün, G. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, 1991.
- [3] T. Brox and J. Weickert. A TV flow based local scale measure for texture discrimination. In T. Pajdla and J. Matas, editors, *Proc. 8th European Conference on Computer Vision (ECCV'04)*, volume 3022 of *LNCS*, pages 578–590. Springer, May 2004.
- [4] T. Brox, J. Weickert, B. Burgeth, and P. Mrázek. Nonlinear structure tensors. *Image and Vision Computing*, 24(1):41–55, 2006.
- [5] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain. Symmetric tensors and symmetric tensor rank. Technical Report SCCM-06-02, Stanford Scientific Computing and Computational Mathematics (SCCM), 2006.
- [6] P. Comon and B. Mourrain. Decomposition of quantics in sums of powers of linear forms. *Signal Processing*, 53(2):96–107, September 1996.
- [7] S. Di Zenzo. A note on the gradient of a multi-image. *Computer Vision, Graphics, and Image Processing*, 33:116–125, 1986.
- [8] C. Feddern, J. Weickert, and B. Burgeth. Level-set methods for tensor-valued images. In O. D. Faugeras and N. Paragios, editors, *Proc. Second*

- IEEE Workshop on Geometric and Level Set Methods in Computer Vision*, pages 65–72, Nice, France, October 2003.
- [9] W. Förstner and E. Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *International Society for Photogrammetry and Remote Sensing (ISPRS) Intercommission Conference on Fast Processing of Photogrammetric Data*, pages 281–305, Interlaken, 1987.
- [10] I. Galić, J. Weickert, M. Welk, A. Bruhn, A. Belyaev, and H.-P. Seidel. Towards PDE-based image compression. In N. Paragios, O. D. Faugeras, T. Chan, and C. Schnörr, editors, *Variational, Geometric, and Level Set Methods in Computer Vision, Third International Workshop (VLSM 2005)*, volume 3752 of *LNCS*, pages 37–48. Springer, 2005.
- [11] W. Gentleman. An error analysis of Goertzel’s (Watt’s) method for computing Fourier coefficients. *The Computer Journal*, 12(2):160–164, 1969.
- [12] M. Herberthson, A. Brun, and H. Knutsson. Pairs of orientations in the plane. In *Proceedings of the SSBA Symposium on Image Analysis*, pages 97–100, Umeå, Sweden, March 2006. SSBA.
- [13] F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1):164–189, 1927.
- [14] F. L. Hitchcock. Multiple invariants and generalized rank of a p-way matrix or tensor. *Journal of Mathematics and Physics*, 7(1):39–79, 1927.
- [15] M. Hlawitschka and G. Scheuermann. HOT-lines: Tracking lines in higher order tensor fields. In C. Silva, E. Gröller, and H. Rushmeier, editors, *Proceedings of IEEE Visualization 2005*, pages 27–34, Minneapolis, MN, USA, October 2005.
- [16] U. Köthe. Edge and junction detection with an improved structure tensor. In B. Michaelis and G. Krell, editors, *Pattern Recognition. 25th DAGM Symposium*, volume 2781 of *Lecture Notes in Computer Science*, pages 25–32. Springer, 2003.
- [17] A. Newbery. Error analysis for Fourier series evaluation. *Mathematics of Computation*, 27(123):639–644, 1973.

- [18] E. Özarslan and T. Mareci. Generalized diffusion tensor imaging and analytical relationships between diffusion tensor imaging and high angular resolution diffusion imaging. *Magnetic Resonance in Medicine*, 50:955–965, 2003.
- [19] E. Özarslan, B. C. Vemuri, and T. H. Mareci. Generalized scalar measures for diffusion MRI using trace, variance, and entropy. *Magnetic Resonance in Medicine*, 53:866–876, 2005.
- [20] W. H. Press, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C++: The Art of Scientific Computing, Second Edition*. Cambridge Univ. Press, Cambridge, 2002.
- [21] M. Rousson, T. Brox, and R. Deriche. Active unsupervised texture segmentation on a diffusion based feature space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 699–706, Madison, Wisconsin, USA, June 2003.
- [22] D. Tschumperlé and R. Deriche. Vector-valued image regularization with PDE’s: A common framework for different applications. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2003)*, pages 651–656, 2003.
- [23] J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner, Stuttgart, 1998.
- [24] J. Weickert and T. Brox. Diffusion and regularization of vector- and matrix-valued images. In M. Nashed and O. Scherzer, editors, *Inverse Problems, Image Analysis, and Medical Imaging*, volume 313 of *Contemporary Mathematics*, pages 251–268. AMS, Providence, 2002.

Below you find a list of the most recent technical reports of the Max-Planck-Institut für Informatik. They are available by anonymous ftp from [ftp.mpi-sb.mpg.de](ftp://ftp.mpi-sb.mpg.de) under the directory `pub/papers/reports`. Most of the reports are also accessible via WWW using the URL <http://www.mpi-sb.mpg.de>. If you have any questions concerning ftp or WWW access, please contact [reports@mpi-sb.mpg.de](mailto:reports@mpi-sb.mpg.de). Paper copies (which are not necessarily free of charge) can be ordered either by regular mail or by e-mail at the address below.

Max-Planck-Institut für Informatik  
 Library  
 attn. Anja Becker  
 Stuhlsatzenhausweg 85  
 66123 Saarbrücken  
 GERMANY  
 e-mail: [library@mpi-sb.mpg.de](mailto:library@mpi-sb.mpg.de)

---

MPI-I-2007-RG1-002	T. Hillenbrand, C. Weidenbach	Superposition for Finite Domains
MPI-I-2007-5-002	S. Bedathur, K. Berberich, T. Neumann, G. Weikum	A Time Machine for Text Search
MPI-I-2007-5-001	G. Ifrim, G. Kasneci, M. Ramanath, F.M. Suchanek, G. Weikum	NAGA: Searching and Ranking Knowledge
MPI-I-2007-4-004	C. Stoll	A Volumetric Approach to Interactive Shape Editing
MPI-I-2007-4-003	R. Bargmann, V. Blanz, H. Seidel	A Nonlinear Viseme Model for Triphone-Based Speech Synthesis
MPI-I-2007-4-002	T. Langer, H. Seidel	Construction of Smooth Maps with Mean Value Coordinates
MPI-I-2007-4-001	J. Gall, B. Rosenhahn, H. Seidel	Clustered Stochastic Optimization for Object Recognition and Pose Estimation
MPI-I-2007-2-001	A. Podelski, S. Wagner	A Method and a Tool for Automatic Verification of Region Stability for Hybrid Systems
MPI-I-2007-1-001	E. Berberich, L. Kettner	Linear-Time Reordering in a Sweep-line Algorithm for Algebraic Curves Intersecting in a Common Point
MPI-I-2006-5-006	G. Kasnec, F.M. Suchanek, G. Weikum	Yago - A Core of Semantic Knowledge
MPI-I-2006-5-005	R. Angelova, S. Siersdorfer	A Neighborhood-Based Approach for Clustering of Linked Document Collections
MPI-I-2006-5-004	F. Suchanek, G. Ifrim, G. Weikum	Combining Linguistic and Statistical Analysis to Extract Relations from Web Documents
MPI-I-2006-5-003	V. Scholz, M. Magnor	Garment Texture Editing in Monocular Video Sequences based on Color-Coded Printing Patterns
MPI-I-2006-5-002	H. Bast, D. Majumdar, R. Schenkel, M. Theobald, G. Weikum	IO-Top-k: Index-access Optimized Top-k Query Processing
MPI-I-2006-5-001	M. Bender, S. Michel, G. Weikum, P. Triantafilou	Overlap-Aware Global df Estimation in Distributed Information Retrieval Systems
MPI-I-2006-4-010	A. Belyaev, T. Langer, H. Seidel	Mean Value Coordinates for Arbitrary Spherical Polygons and Polyhedra in $\mathbb{R}^3$
MPI-I-2006-4-009	J. Gall, J. Potthoff, B. Rosenhahn, C. Schnoerr, H. Seidel	Interacting and Annealing Particle Filters: Mathematics and a Recipe for Applications
MPI-I-2006-4-008	I. Albrecht, M. Kipp, M. Neff, H. Seidel	Gesture Modeling and Animation by Imitation
MPI-I-2006-4-007	O. Schall, A. Belyaev, H. Seidel	Feature-preserving Non-local Denoising of Static and Time-varying Range Data
MPI-I-2006-4-006	C. Theobald, N. Ahmed, H. Lensch, M. Magnor, H. Seidel	Enhanced Dynamic Reflectometry for Relightable Free-Viewpoint Video
MPI-I-2006-4-005	A. Belyaev, H. Seidel, S. Yoshizawa	Skeleton-driven Laplacian Mesh Deformations
MPI-I-2006-4-004	V. Havran, R. Herzog, H. Seidel	On Fast Construction of Spatial Hierarchies for Ray Tracing

MPI-I-2006-4-003	E. de Aguiar, R. Zayer, C. Theobalt, M. Magnor, H. Seidel	A Framework for Natural Animation of Digitized Models
MPI-I-2006-4-002	G. Ziegler, A. Tevs, C. Theobalt, H. Seidel	GPU Point List Generation through Histogram Pyramids
MPI-I-2006-4-001	A. Efremov, R. Mantiuk, K. Myszkowski, H. Seidel	Design and Evaluation of Backward Compatible High Dynamic Range Video Compression
MPI-I-2006-2-001	T. Wies, V. Kuncak, K. Zee, A. Podelski, M. Rinard	On Verifying Complex Properties using Symbolic Shape Analysis
MPI-I-2006-1-007	H. Bast, I. Weber, C.W. Mortensen	Output-Sensitive Autocompletion Search
MPI-I-2006-1-006	M. Kerber	Division-Free Computation of Subresultants Using Bezout Matrices
MPI-I-2006-1-005	A. Eigenwillig, L. Kettner, N. Wolpert	Snap Rounding of Bézier Curves
MPI-I-2006-1-004	S. Funke, S. Laue, R. Naujoks, L. Zvi	Power Assignment Problems in Wireless Communication
MPI-I-2005-5-002	S. Siersdorfer, G. Weikum	Automated Retraining Methods for Document Classification and their Parameter Tuning
MPI-I-2005-4-006	C. Fuchs, M. Goesele, T. Chen, H. Seidel	An Empirical Model for Heterogeneous Translucent Objects
MPI-I-2005-4-005	G. Krawczyk, M. Goesele, H. Seidel	Photometric Calibration of High Dynamic Range Cameras
MPI-I-2005-4-004	C. Theobalt, N. Ahmed, E. De Aguiar, G. Ziegler, H. Lensch, M.A. Magnor, H. Seidel	Joint Motion and Reflectance Capture for Creating Relightable 3D Videos
MPI-I-2005-4-003	T. Langer, A.G. Belyaev, H. Seidel	Analysis and Design of Discrete Normals and Curvatures
MPI-I-2005-4-002	O. Schall, A. Belyaev, H. Seidel	Sparse Meshing of Uncertain and Noisy Surface Scattered Data
MPI-I-2005-4-001	M. Fuchs, V. Blanz, H. Lensch, H. Seidel	Reflectance from Images: A Model-Based Approach for Human Faces
MPI-I-2005-2-004	Y. Kazakov	A Framework of Refutational Theorem Proving for Saturation-Based Decision Procedures
MPI-I-2005-2-003	H.d. Nivelle	Using Resolution as a Decision Procedure
MPI-I-2005-2-002	P. Maier, W. Charatonik, L. Georgieva	Bounded Model Checking of Pointer Programs
MPI-I-2005-2-001	J. Hoffmann, C. Gomes, B. Selman	Bottleneck Behavior in CNF Formulas
MPI-I-2005-1-008	C. Gotsman, K. Kaligosi, K. Mehlhorn, D. Michail, E. Pyrga	Cycle Bases of Graphs and Sampled Manifolds
MPI-I-2005-1-007	I. Katriel, M. Kutz	A Faster Algorithm for Computing a Longest Common Increasing Subsequence
MPI-I-2005-1-003	S. Baswana, K. Telikepalli	Improved Algorithms for All-Pairs Approximate Shortest Paths in Weighted Graphs
MPI-I-2005-1-002	I. Katriel, M. Kutz, M. Skutella	Reachability Substitutes for Planar Digraphs
MPI-I-2005-1-001	D. Michail	Rank-Maximal through Maximum Weight Matchings
MPI-I-2004-NWG3-001	M. Magnor	Axisymmetric Reconstruction and 3D Visualization of Bipolar Planetary Nebulae
MPI-I-2004-NWG1-001	B. Blanchet	Automatic Proof of Strong Secrecy for Security Protocols
MPI-I-2004-5-001	S. Siersdorfer, S. Sizov, G. Weikum	Goal-oriented Methods and Meta Methods for Document Classification and their Parameter Tuning
MPI-I-2004-4-006	K. Dmitriev, V. Havran, H. Seidel	Faster Ray Tracing with SIMD Shaft Culling
MPI-I-2004-4-005	I.P. Ivrissimtzis, W.-. Jeong, S. Lee, Y.a. Lee, H.-. Seidel	Neural Meshes: Surface Reconstruction with a Learning Algorithm
MPI-I-2004-4-004	R. Zayer, C. Rössl, H. Seidel	r-Adaptive Parameterization of Surfaces
MPI-I-2004-4-003	Y. Ohtake, A. Belyaev, H. Seidel	3D Scattered Data Interpolation and Approximation with Multilevel Compactly Supported RBFs
MPI-I-2004-4-002	Y. Ohtake, A. Belyaev, H. Seidel	Quadric-Based Mesh Reconstruction from Scattered Data