Anisotropic Diffusion in Vector Field Visualization on Euclidean Domains and Surfaces

Udo Diewald, Tobias Preußer, and Martin Rumpf

Abstract—Vector field visualization is an important topic in scientific visualization. Its aim is to graphically represent field data on two and three-dimensional domains and on surfaces in an intuitively understandable way. Here, a new approach based on anisotropic nonlinear diffusion is introduced. It enables an easy perception of vector field data and serves as an appropriate scale space method for the visualization of complicated flow pattern. The approach is closely related to nonlinear diffusion methods in image analysis where images are smoothed while still retaining and enhancing edges. Here, an initial noisy image intensity is smoothed along integral lines, whereas the image is sharpened in the orthogonal direction. The method is based on a continuous model and requires the solution of a parabolic PDE problem. It is discretized only in the final implementational step. Therefore, many important qualitative aspects can already be discussed on a continuous level. Applications are shown for flow fields in 2D and 3D, as well as for principal directions of curvature on general triangulated surfaces. Furthermore, the provisions for flow segmentation are outlined.

Index Terms—Flow visualization, multiscale, nonlinear diffusion, segmentation.

1 INTRODUCTION

The visualization of field data, especially of velocity fields from CFD computations, is one of the fundamental tasks in scientific visualization. A variety of different approaches has been presented. The simplest method of drawing vector plots at nodes of some overlaid regular grid in general produces visual clutter because of the typically different local scaling of the field in the spatial domain, which leads to disturbing multiple overlaps in certain regions, whereas, in other areas, small structures such as eddies cannot be resolved adequately. This gets even worse if tangential fields on highly curved surfaces are considered.

The central goal is to come up with intuitively better receivable methods which give an overall, as well as a detailed, view on the flow patterns. Single particle lines only partially enlighten features of a complex flow field. Thus, we want to define a texture which represents the field globally on a 2D or 3D domain and on surfaces, respectively. Here, we confine ourselves to stationary fields. In the Euclidean case, we suppose \( \mathbf{v} : \Omega \rightarrow \mathbb{R}^n \) for some domain \( \Omega \subset \mathbb{R}^n \), whereas, in the case of a manifold \( \mathcal{M} \) embedded in \( \mathbb{R}^3 \), we consider a tangential vector field \( \mathbf{v} \). We ask for a method generating stretched streamline type patterns which are aligned to the vector field \( \mathbf{v}(x) \). Furthermore, the possibility of successively coarsening this pattern is obviously a desirable property. Methods which are based on such a scale of spaces and enhance certain structures of images are well-known in image processing analysis.

Actually, nonlinear diffusion allows the smoothing of gray or color images while retaining and enhancing edges [18]. Now, we set up a diffusion problem, with strong smoothing along integral lines and edge enhancement in the orthogonal directions. Applying this to some initial random noise image intensity, we generate a scale of successively coarser patterns which represent the vector field. Finite elements in space and a semi-implicit time stepping are applied to solve this diffusion problem numerically. Furthermore, a suitable modification of the approach allows the identification of topological regions.

Before we explain in detail the method, let us discuss related work on vector field visualization and image processing. Later on we will identify some of the well-known methods as equivalent to special cases or asymptotic limits of the presented new method, respectively.

2 RELATED WORK

The spot noise method proposed by van Wijk [25] introduces spot-like texture splats which are aligned by deformation to the velocity field in 2D or on surfaces in 3D. These splats are plotted in the fluid domain, showing strong alignment patterns in the flow direction. The original first order approximation to the flow was improved by de Leeuw and van Wijk in [6] by using higher order polynomial deformations of the spots in areas of significant vorticity. In an animated sequence, these spots can be moved along streamlines of the flow. Furthermore, in 3D, van Wijk [26] applies the integration to clouds of oriented particles and animates them by drawing similar moving transparent and illuminated splats.

The Line Integral Convolution (LIC) approach of Cabral and Leedom [4] integrates the fundamental ODE describing streamlines forward and backward in time at every
pixelized point in the domain, convolves a white noise along these particle paths with some Gaussian type filter kernel, and takes the resulting value as an intensity value for the corresponding pixel. According to the strong correlation of this intensity along the streamlines and the lack of any correlation in the orthogonal direction, the resulting texturing of the domain shows dense streamline filaments of varying intensity. Stalling and Hege [21] increased the performance of this method, especially by reusing portions of the convolution integral already computed on points along the streamline. Forssell [10] proposed a similar method on surfaces and Max et al. [17] discussed flow visualization by texturing on contour surfaces. Max and Becker [16] presented a method for visualizing 2D and 3D flows by animating textures. Shen and Kao [20] applied an LIC type method to unsteady flow fields. Recently, a method [2] has been presented which generates streakline type patterns by numerical calculation of the transport of inlet coordinates and inlet position. Interrante and Grosch [12] generalized line integral convolution to 3D in terms of volume rendering of line filaments.

In [24], Turk and Banks discuss an approach which selects a certain number of streamlines. They are automatically equally distributed all over the computational domain to characterize, in a sketch-type representation, the significant aspects of the flow. An energy minimizing process is used to generate the actual distribution of streamlines.

Especially for 3D velocity fields, particle tracing is a very popular tool. But, a few particle integrations released by the user can hardly scope with the complexity of 3D vector fields. Stalling et al. [22] use pseudorandomly distributed, illuminated, and transparent streamlines to give a denser and more receptible representation, which shows the overall structure and enhances important details.

Van Wijk [27] proposed the implicit stream surface method. For a stationary flow field, the transport equations \( v \cdot \nabla \phi = 0 \) are solved for given \( v \) and certain inflow and outflow boundary conditions in a precomputing step. Then, isosurfaces of the resulting function \( \phi \) are streamsurfaces and can be efficiently extracted with interactive frame rates, even for larger data sets.

Most of the methods presented so far have in common, that the generation of a coarser scale requires a recomputation. For instance, if we ask for a finer or coarser scale of the line integral convolution pattern, the computation has to be restarted with a coarser initial image intensity. In the case of spot noise, larger spots have to be selected and their stretching along the field has to be increased. The approach to be presented here will incorporate a successive coarsening as time proceeds in the underlying diffusion problem.

As already mentioned in the introduction, our method of anisotropic nonlinear diffusion to visualize vector fields is derived from well-known image processing methodology. Discrete diffusion type methods have been known for a long time. Perona and Malik [18] introduced a continuous diffusion model which allows the denoising of images together with the enhancing of edges. Alvarez et al. [1] established a rigorous axiomatic theory of diffusive scale space methods. Kawohl and Kutev [14] investigate a qualitative analysis of the Perona and Malik model. The recovering of lower dimensional structures in images is analyzed by Weickert [28], who introduced an anisotropic nonlinear diffusion method, where the diffusion matrix depends on the so-called structure tensor of the image. A finite element discretization and its convergence properties have been studied by Kacur and Mikula [13].

Concerning the application of diffusion type methods on surfaces, a general introduction to differential calculus on manifolds can be found for instance in the book by do Carmo [7]. Dziuk [8] presented an algorithm for the solution of partial differential equations on surfaces and, in [9], he discussed a numerical method for geometric diffusion applied to the surface itself which coincides with the mean curvature motion.

3 THE NONLINEAR DIFFUSION PROBLEM

Let us now derive our method based on a suitable PDE problem. At first, we confine ourselves to the case of planar domains in 2D and 3D. Here, nonlinear anisotropic diffusion applied to some initial random noisy image will enable an intuitive and scalable visualization of complicated vector fields. Therefore, we pick up the idea of line integral convolution, where a strong correlation in the image intensity along integral lines is achieved by convolution of an initial white noise along these lines. As proposed already by Cabral and Leedom [4], a suitable choice for the convolution kernel is a Gaussian kernel. On the other hand, an appropriately scaled Gaussian kernel is known to be the fundamental solution of the heat equation. Thus, line integral convolution is nothing else than solving the heat equation in 1D on an integral line parameterized with respect to arc length. On pixels which are located on different integral lines, the resulting image intensities are not correlated. Hence, the thickness of the resulting image patterns in line integral convolution is of the size of the random initial patterns, in general, a single pixel. Increasing this size, as has been proposed by Kiu and Banks [15], leads to broader stripes and, unfortunately, less sharp transitions across streamline patterns. As described so far, line integral convolution is a discrete pixel-based method. If we ask for a well-posed continuous diffusion problem with similar properties, we are led to some anisotropic diffusion, now controlled by a suitable diffusion matrix.

To begin with, let us at first introduce a general nonlinear diffusion method from image processing and then discuss the selection of the appropriate diffusion tensor and the righthand side. Here, we consider first the case of an image in Euclidean space either in 2D or 3D. In Section 6, we then generalize this with respect to textures on surfaces. We consider a function \( \rho : \mathbb{R}_+^+ \times \Omega \rightarrow \mathbb{R} \) which solves the parabolic problem

\[
\frac{\partial}{\partial t} \rho - \text{div}(A(\nabla \rho)\nabla \rho) = f(\rho) \quad \text{in} \ \mathbb{R}_+^+ \times \Omega,
\]

\[
\rho(0, \cdot) = \rho_0 \quad \text{on} \ \Omega,
\]

\[
\frac{\partial}{\partial \nu} \rho = 0 \quad \text{on} \ \mathbb{R}_+^+ \times \partial \Omega.
\]

for given initial density \( \rho_0 : \Omega \rightarrow [0, 1] \). Here, \( \rho_0 = \chi_\varepsilon * \rho \) is a mollification of the current density, which will later on turn
out to be necessary for the well-posedness of the above parabolic, boundary, and initial value problem. In our setting, we interpret the density as an image intensity, a scalar grayscale or—with a slight extension to the vector valued case—as a vector valued color. Thus, the solution \( \rho(t) \) can be regarded as a family of images \( \{\rho(t)\}_{t \in \mathbb{R}^+} \) where the time \( t \) serves as a scaling parameter. Let us remark that, by the trivial choice \( A = 1 \) and \( f(\rho) = 0 \), we obtain the standard linear heat equation with its isotropic smoothing effect. In image processing, \( \rho_0 \) is a given noisy initial image. The diffusion is supposed to be controlled by the gradient of the image intensity. Large gradients mark edges in the image which should be enhanced, whereas small gradients indicate areas of approximately equal intensity. Here, denoising, i.e., intensity diffusion, is considered. For that purpose we prescribe a diffusion coefficient

\[
A = G(\|\nabla \rho_0\|),
\]

where \( G : \mathbb{R}^+ \to \mathbb{R}^+ \) is a monotone decreasing function with \( \lim_{d \to \infty} G(d) = 0 \) and \( G(0) = \beta \), where \( \beta \in \mathbb{R}^+ \) is constant (cf. Fig. 1), e.g. \( G(d) = \frac{d}{1+d^2} \). If we would replace the mollified gradient \( \nabla \rho_0 \) as argument of \( G \) by the true gradient \( \nabla \rho_0 \) which leads to the original Perona Malik model, we would, in general, obtain a backward parabolic problem in areas of high gradients which is no longer well-posed [14]. The invoked mollification avoids this shortcoming and comes along with a desirable presmoothing effect. Nevertheless, the enhancing of steep gradients and, thereby, edges in the image, known from backward diffusion, is retained if we adjust the mollification carefully.

A suitable choice [13] for this mollification is a convolution with the heat equation kernel, i.e., we define \( \rho_0 = \bar{\rho}(t = \epsilon^2/2) \) where \( \bar{\rho} \) is the solution of the heat equation with initial data \( \rho_0 \). Then, \( \epsilon \) is the width of the corresponding Gaussian filter. Fig. 2 gives an example of such an image smoothing and edge enhancement by nonlinear diffusion. The function \( f(\cdot) \) may serve as a penalty which forces the scale of images to stay close to the initial image, e.g., choosing \( f(\rho) = \gamma(\rho_0 - \rho) \), where \( \gamma \) is a positive constant.

Now, we incorporate anisotropic diffusion. For a given vector field \( v : \Omega \to \mathbb{R}^n \), we consider linear diffusion in the direction of the vector field and a Perona Malik type diffusion orthogonal to the field. Let us suppose that \( v \) is continuous and \( v \neq 0 \) on \( \Omega \). Then, there exists a family of continuous orthogonal mappings \( B(v) : \Omega \to SO(n) \) such that \( B(v)v = \|v\|e_0 \) where \( \{e_i\}_{i=0,...,n-1} \) is the standard base in \( \mathbb{R}^n \) (cf. Fig. 3). We consider a diffusion matrix \( A = A(v, \nabla \rho_0) \) and define

\[
A(v, d) = B(v)^T \left( \frac{\alpha(\|v\|)}{G(\|d\|)} \right) B(v),
\]

where \( \alpha : \mathbb{R}^+ \to \mathbb{R}^+ \) controls the linear diffusion in the vector field direction, i.e., along streamlines, and the edge-enhancing diffusion coefficient \( G(\cdot) \) introduced above acts in the orthogonal directions. Here, \( \text{Id}_{n-1} \) is the identity matrix in dimension \( n-1 \). We may either choose a linear function \( \alpha \) or, in the case of a velocity field which spatially varies over several orders of magnitude, we select a monotone function \( \alpha \) (cf. Fig. 4) with

\[
\alpha(0) > 0 \quad \text{and} \quad \lim_{s \to \infty} \alpha(s) = \alpha_{\max}.\]

In general, it does not make sense to consider a certain initial image. As initial data \( \rho_0 \), we thus choose some random noise of an appropriate frequency range. This can, for instance, be generated by running a linear isotropic diffusion simulation on a discrete white noise for a short time. Hence, patterns will grow upstream and downstream, whereas the edges tangential to these patterns are successively enhanced. Still, there is some diffusion perpendicular to the field which supplies us for evolving time with a scale of progressively coarser representation of the flow field. If we run the evolution for vanishing righthand side \( f \), the image contrast will, unfortunately, decrease due to the diffusion along streamlines. The asymptotic limit would turn out to be an averaged gray value. Therefore, we strengthen the image contrast during the evolution, selecting an appropriate function \( f : [0,1] \to \mathbb{R} \) (cf. Fig. 4) with

\[
\|v\|e_0
\]

\[
\|v\|e_n
\]

\[
\|v\|e_n
\]

\[
\|v\|e_n
\]
If we—at first glance—neglect the diffusive term in the equation, one realizes that perturbations below the average value 0.5 are pushed toward the zero value and, accordingly, values above 0.5 are pushed toward 1. Well-known maximum principles ensure that the interval of gray values [0, 1] is not enlarged running the nonlinear diffusion. Here, the first property of \( f \) is of great importance. Finally, we end up with the method of nonlinear anisotropic diffusion to visualize complex vector fields. We thereby solve the nonlinear parabolic problem

\[
\frac{\partial}{\partial t} \rho - \text{div} (A(v, \nabla \rho, \nabla \rho)) = f(\rho),
\]

starting from some random initial image \( \rho_0 \), and obtain a scale of images representing the vector field in an intuitive way (cf. Fig. 5).

The corresponding variational formulation is obviously given by

\[
(\partial (\rho, \theta)) + (A(v, \nabla \rho, \nabla \rho)) \nabla \theta = (f, \theta),
\]

for all \( \theta \in C^\infty(\Omega) \), where \((,\) denotes the \( L^2 \) product on the domain \( \Omega \). Our later finite element implementation will be based on this formulation by restriction to finite dimensional function spaces.

If we ask for pointwise asymptotic limits of the evolution, we expect an almost everywhere convergence to \( \rho(\infty, \cdot) \in \{0, 1\} \) due to the choice of the contrast enhancing function \( f(\cdot) \). Analytically, 0.5 is a third, but unstable, fix point of the dynamics. Thus, numerically, it will not turn out to be locally dominant. The space of asymptotic limits significantly influences the richness of the developing vector field aligned structures. We may ask how to further enrich the pattern which is settled by anisotropic diffusion. This turns out to be possible by increasing the set of asymptotic states. We no longer restrict ourselves to a scalar density \( \rho \), but consider a vector valued \( \rho : \Omega \to [0, 1]^m \) for some \( m \geq 1 \), and a corresponding system of parabolic equations. The coupling is given by the nonlinear diffusion coefficient \( G(\cdot) \) which now depends on the norm \( \|\nabla \rho\| \) of the Jacobian of the vector valued density \( \nabla \rho \) and the righthand side \( f(\cdot) \). We define

\[
f(\rho) = h(\|\rho\|)\rho
\]

with \( h(s) = f(s)/s \) for \( s \neq 0 \), where \( f \) is the old righthand side from the scalar case and \( h(0) = 0 \). Furthermore, we select an initial density which is now a discrete “white” noise with values in \( B_1(0) \cap [0, 1]^m \). Thus, the contrast enhancing now pushes the pointwise vector density \( \rho \) either to the 0 or to some value on the sphere sector \( S^{m-1} \cap [0, 1]^m \) in \( \mathbb{R}^m \) (cf. Fig. 6). Again, a straightforward application of the maximum principle ensures \( \rho(t, x) \in S^{m-1} \cap [0, 1]^m \) for all \( t \) and \( x \in \Omega \).

Fig. 7 shows an example for the application of the vector valued anisotropic diffusion method applied to a 2D flow field from a MHD simulation convective flow field. Furthermore, Fig. 8 shows results of this method applied to several time steps of a convective flow field. An incompressible Bénard convection is simulated in a rectangular box with heating from below and cooling from above. The formation of convection rolls will lead to an exchange of temperature. We recognize that the presented method is able to nicely depict the global structure of the flow field, including its saddle points, vortices, and stagnation points on the boundary. Fig. 9 shows results for the same data sets obtained by line integral convolution (here, we used the implementation of Stalling and Hege [21]). Finally, Fig. 10 shows a different application to a porous media flow field.
The anisotropic nonlinear diffusion problem has been formulated in Section 3 for arbitrary space dimension. It results in a scale of vector field aligned patterns which we then have to visualize. In 2D, this has already been done in a straightforward manner in the above figures. In 3D, we have somehow to break up the volume and open up the view to inner regions. Otherwise, we must confine ourselves with some pattern close to the boundary representing solely the shear flow.

5 APPLICATION IN 3D

The anisotropic nonlinear diffusion problem has been formulated in Section 3 for arbitrary space dimension. It results in a scale of vector field aligned patterns which we then have to visualize. In 2D, this has already been done in a straightforward manner in the above figures. In 3D, we have somehow to break up the volume and open up the view to inner regions. Otherwise, we must confine ourselves with some pattern close to the boundary representing solely the shear flow.

6 ANISOTROPIC DIFFUSION ON SURFACES

In the above sections, we have discussed anisotropic diffusion in vector field visualization on domains which are subsets of two and three-dimensional Euclidean space.
In what follows, we will outline how to carry over this methodology to display tangential vector fields on surfaces. Important examples are results from meteorological simulations, flow fields on streamsurfaces, or vector fields in differential geometry. The applications presented here will focus on the latter case and present multiscale textures on surfaces representing the principal directions of curvature. Based on the well-established intrinsic differential calculus on manifolds [7], we can pick up the same diffusion problems with an appropriate reinterpretation of the operators. Thus, let us first briefly review the basic notation of manifolds, differential calculus, and geometric diffusion. For a detailed introduction to geometry and differential calculus, we refer to [7] and [5, chapter 1]. For the sake of simplicity, we assume our surfaces to be compact embedded manifolds without boundary. Thus, we consider a smooth manifold \( M \), which we suppose to be embedded in \( \mathbb{R}^3 \). Let \( x : \Omega \to M; \xi \mapsto x(\xi) \) be a coordinate map from an atlas of \( M \). For each point \( x \) on \( M \), the embedded tangent space \( T_x M \) is spanned by the basis \( \left\{ \frac{\partial}{\partial x^i}, \frac{\partial}{\partial x^j} \right\} \). By \( TM \), we denote the tangent bundle. On \( M \), the metric \( g(\cdot, \cdot) \) as a bilinear form on \( TM \times TM \) is prescribed by the metric tensor \( g_{ij} \) with

\[
g_{ij} = \frac{\partial x^i}{\partial \xi^k} \frac{\partial x^j}{\partial \xi^l} \cdot \frac{\partial x^k}{\partial \xi^m} \frac{\partial x^l}{\partial \xi^n},
\]

where \( \cdot \) indicates the scalar product in \( \mathbb{R}^3 \). The inverse of \( g \) is denoted by \( g^{-1} = (g^{ij})_{ij} \). Based on the metric, we can define the integration of a function \( f \) on \( M \). We split up an integral over \( M \) into separate integrals over subsets which are in the image \( x(\Omega) \) of some coordinate map \( x \) and define

\[
\int_{x(\Omega)} f := \int_\Omega f(x(\xi)) \sqrt{\det g} \, d\xi.
\]

Integrating either a product of two functions \( f, g \) on \( M \) or the product of two vector fields \( v, w \) on \( TM \), we obtain the following scalar products on \( C^0(M) \) and \( C^0(TM) \), respectively:

\[
(f, g)_M := \int_M fg \, dx,
\]

\[
(v, w)_{TM} := \int_M g(v, w) \, dx.
\]

Next, we have to introduce the fundamental intrinsic gradient and divergence operators on \( M \). The gradient \( \nabla_M f \) of \( f \) is defined as the representation of \( df \) with respect to the metric \( g \). We obtain, in coordinates,

\[
\nabla_M f = \sum_{ij} g^{ij} \frac{\partial (f \circ x)}{\partial x^j} \frac{\partial x^i}{\partial \xi^k} \frac{\partial x^k}{\partial \xi^l}.
\]

Furthermore, we define the divergence \( \text{div}_M v \) for a vector field \( v \in TM \) as the dual operator of the gradient by

\[
\int_M \text{div}_M v \, \phi \, dx := -\int_M g(v, \nabla_M \phi) \, dx
\]
for all $\phi \in C^\infty_0(M)$.

Finally, with these differential operators at hand, we can discuss a general and intrinsic diffusion on a manifold in analogy to diffusion in Euclidean space: We ask for a solution $\rho : \mathbb{R}^+ \times M \to \mathbb{R}$ of the parabolic equation

$$\frac{\partial}{\partial t} \rho - \text{div}_M(A \nabla_M \rho) = f(\rho)$$

on $\mathbb{R}^+ \times M$ for given initial data $\rho(0, \cdot) = \rho_0$ on $M$. Here, we suppose $A$ to be some positive definite symmetric endomorphism on $TM$. Testing with any function $\phi \in C^\infty(M(t))$ and integrating over $M$, we obtain the variational formulation

$$(\partial_t \rho, \theta)_M + (A \nabla_M \rho, \nabla_M \theta)_TM = (f(\rho), \theta)_M.$$

Now, we consider our actual goal, which is the generation of a texture by nonlinear anisotropic diffusion to represent a given vector field $v \in TM$ on the surface. Thus, we suppose $A$ to depend on the vector field $v$ and the norm of the gradient of a convoluted intensity $\rho_0$:

$$A = A(v, \|\nabla_M \rho_0\|).$$

For no vanishing $v$, let $w \in T_xM$ be some unit vector normal to $v$, i.e., $g(v, w) = 0$. Hence, $\{\frac{v}{\|v\|}, w\}$ is a basis of $T_xM$ and, with respect to this basis, we define, as before in the Euclidean case,

$$A(v, d) = \begin{pmatrix} \alpha(\|v\|) & \frac{\partial}{\partial \|d\|} \\ \frac{\partial}{\partial \|d\|} & G(\|d\|) \end{pmatrix}.$$

As righthand side $f(\cdot)$, we pick up the one already introduced in Section 3 and again assume $\rho_0$ to be a random noise, either scalar or vector valued, but now prescribed on the surface $M$. Furthermore, we have to give a suitable definition of the regularizing presmoothing to obtain $\rho$, from the original intensity $\rho$. Again, we proceed in analogy to the Euclidean case and define $\rho$, as the result of the above diffusion problem with $A = \text{Id}$ at time $t = \frac{T}{2}$ and for initial data $\rho$.

Finally, the resulting family $\{\rho(t)\}_{t \geq 0}$ of intensities on $M$ gives a multiscale of representations of the given vector field $v$. Fig. 13 and Fig. 14 show results on different surfaces. We consider the principal directions of curvature as tangential vector fields on which we apply the anisotropic diffusion method. On the underlying triangular grids, the shape operator, whose eigenvalues are the principal curvatures, is approximated as follows: Locally, we regard a single triangle $T$ and all the neighboring triangles which have a nonzero intersection with $T$ as a graph over the plane containing $T$ and calculate the $L^2$ projection of this piecewise linear graph onto the set of quadratic graphs which are tangential to the plane. Then, we evaluate the constant shape operator on this graph. Let us emphasize that the $L^2$ projection is always defined, although the local graph property of the triangular grid might not hold in certain degenerate cases.

### 7 Discretization in 2D and 3D

In what follows, we discuss the discretization and implementation of the field aligned diffusion method. We will first focus on domains in 2D and 3D Euclidean space. For this purpose, a finite element discretization in space and a semi-implicit backward Euler or second order Crank－Nicholson method in time is considered. The choice of the finite element method is guided by the need to approximate the diffusion process as accurately as possible. The semi-implicit Crank－Nicholson scheme is chosen for its stability properties, which are crucial for the convergence of the numerical solution.
Nicolsson scheme in time are considered. Here, we have restricted ourselves to regular grids in 2D and 3D generated by recursive subdivision. On these grids, we consider bilinear, respectively, trilinear, finite element spaces. Numerical integration is based on the lumped masses product $(\cdot, \cdot)^L$ [23] for the $L^1$ product $(\cdot, \cdot)$ in the variational formulation and a one point quadrature rule for the bilinear form $(A \nabla \cdot, \nabla \cdot)$. Semi-implicit means, for the schemes considered here, that the nonlinearity $A(\cdot)$ is evaluated at the old time. Finally, in each step of the discrete evolution, we have to solve a single system of linear equations. We obtain, for a backward Euler discretization,

\[(M^k + \tau L^k(A^k))\rho^{k+1} = M^k \rho^k + \tau M^k \tilde{\rho}^k.\]

Here, $\tilde{\rho}^k = (\tilde{\rho}_i^k)$, is the vector of nodal intensity values at time $t^k = kt \tau$, where $\tau$ is the selected time step size.

Furthermore, if we denote the "hat shaped" multilinear basis functions by $\Phi_i$ and the diffusion tensor with respect to the discrete intensity at time $t^k$ by $A^k$, we immediately verify

\[
M^k := \left( (\Phi_i^k, \Phi_j^k) \right)_{ij},
\]

\[
L^k(A^k) := \left( (A_i^k \nabla \Phi_i^k, \nabla \Phi_j^k) \right)_{ij},
\]

are the lumped mass matrix and nonlinear stiffness matrix, respectively. Finally, the components of the right-hand side $f^k$ are evaluated by $(f^k)_i = f(\tilde{\rho}_i^k)$.

The global matrices $M^k$ and $L^k(A^k)$ are assembled from local matrices $m^k$ and $l^k$ with respect to a single element. Their entries correspond to all pairings of local basis functions. Due to the applied lumped mass integration, we immediately verify

\[
m^k_{ij} = \frac{1}{2h_i} \delta_{ij}[E],
\]

where $[E]$ is the volume of the rectangular element $E$ and $\delta_{ij}$ the usual Kronecker symbol. For the nonlinear stiffness matrix we obtain

\[
l^k_{ij}(A) = \|E\| \left[ \alpha(||V||) \left( \nabla \Phi_i^j \cdot \frac{V}{||V||} \right) \left( \nabla \Phi_j^i \cdot \frac{V}{||V||} \right) + G(||D||) \left( \nabla \Phi_i^j \cdot \frac{V}{||V||} \right) \left( \nabla \Phi_j^i \cdot \frac{V}{||V||} \right) \right] .
\]

8 Discretization on Surfaces

The discretization of the proposed anisotropic diffusion method on surfaces is completely analogous to the above Euclidean case. We only have to replace the discrete differential operators and bilinear forms by their intrinsic geometric counterparts. We suppose the surface $M$ to be approximated with a sufficiently fine triangular grid $M_h$ consisting of nondegenerate triangles $T$ with maximal diameter $h$. Thus, we only focus on the computation of the local mass matrix $m^T$ and the local nonlinear stiffness matrix $l^T(A)$, respectively. We obtain again by lumped mass integration

\[
m^T_{ij} = \frac{1}{3} \delta_{ij}[T],
\]

where $[T]$ is the area of the triangle $T$. Next, let us consider, for every triangle $T$, the reference triangle $\hat{T} \subset \mathbb{R}^2$ with independent variables $\xi_1, \xi_2$ and nodes $\xi^0 = (0, 0)$, $\xi^1 = (1, 0)$, and $\xi^2 = (0, 1)$. Then, an affine coordinate mapping $X$ maps $\hat{T}$ onto $T$ and its nodes $\xi_i^o$ onto the corresponding nodes $P_i$ of $T$ on the discrete surface in $\mathbb{R}^3$. Hence, the corresponding metric tensor is as in the continuous case given by $g_{ij} = \partial \xi_i / \partial x_j = \partial \xi_i / \partial x_j$, where $\partial \xi_i / \partial x_j = P_i^0 - P_i^0$. Hence, we can evaluate gradients of the linear basis functions $\Phi^j$ corresponding to the nodes $P^j$ by

\[
\nabla_{M_h} \Phi^j = \sum_{i,j} g_{ij} \frac{\partial \Phi^j}{\partial \xi^i} (P^i - P^0),
\]

where the derivatives of $\Phi^j$ with respect to the reference coordinates $\xi$ are

\[
\left( \frac{\partial \Phi^j}{\partial \xi^1} \right) = \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix}, \left( \frac{\partial \Phi^j}{\partial \xi^2} \right) = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.
\]

Finally, we calculate the local nonlinear stiffness matrix

\[
l^T_{ij}(A) = \|T\| \left[ \alpha(||V||) \left( \nabla_{M_h} \Phi^j \cdot \frac{V}{||V||} \right) \left( \nabla_{M_h} \Phi^i \cdot \frac{V}{||V||} \right) + G(||D||) \left( \nabla_{M_h} \Phi^j \cdot \frac{V}{||V||} \right) \left( \nabla_{M_h} \Phi^i \cdot \frac{V}{||V||} \right) \right] .
\]

where $V = v(c_T)$ for the center of mass $c_T$ of $T$, $D$ the geometric gradient of the presmoothed discrete intensity on $T$, and "\cdot" still indicates the scalar product in $\mathbb{R}^3$. 
9 COMPARISON TO OTHER METHODS

So far, we have introduced a novel approach which provides us with an intuitive understanding of complex vector fields. We have discussed a variety of important properties and advantages. Let us now rank this method among other visualization methods and compare it with different techniques. Here, we especially pick up the line integral convolution method and the spot noise approach.

For stationary vector fields, we obtain similar results by all methods. Thin field aligned patterns are generated. Line integral convolution leads to comparable results with the essential difference that the PDE-based method carries a nice scale space property, i.e., evolving a longer time in the anisotropic diffusion method, we obtain a successive coarsening of the resulting pattern representing the vector field.

Furthermore, in a restricted sense, line integral convolution (LIC) and spot noise can be regarded as special cases of the anisotropic nonlinear diffusion method. LIC with Gaussian filter kernel can be identified as the asymptotic limit of the latter method for a concentration of the edge enhancing function $G(\cdot)$ at 0. Other filter kernel shapes correspond to different, in general, nonlinear diffusion processes along streamlines. Further on, generating a single deformed spot on the computational domain, as proposed in [6], can be regarded as an early time step in the diffusion starting with initial data, that is, a characteristic function of a circular disk. If we release a bunch of such disks as initial data in such a way that the evolving patterns do not overlap, then the resulting image is comparable to spot noise. Thus, the original spot noise technique can be regarded as a parallel version of short time diffusive vector field visualization.

10 TOWARD FLOW SEGMENTATION

The above applications already show the capacity of the anisotropic nonlinear diffusion method to outline the flow structure not only locally. Indeed, especially for larger evolution times in the diffusion process, the topological skeleton of a vector field becomes clearly visible. We will now investigate a possible flow segmentation by means of the anisotropic diffusion. Let us restrict this to the two-dimensional case of an incompressible flow with vanishing velocity $v$ at the domain boundary $\partial \Omega$. Then, topological regions are separated by homoclinic, respectively, heteroclinic, orbits connecting critical points in the interior of the domain and stagnation points on the boundary. Critical points, by definition, points with vanishing velocity $v = 0$, may either be saddle points or vortices. Furthermore, we assume critical points to be nondegenerate, i.e., $\nabla v$ is regular. Saddle points are characterized by two real eigenvalues of $\nabla v$ with opposite sign, whereas, at vortices, we obtain complex conjugate eigenvalues with vanishing real part. Stagnation points on $\partial \Omega$ are similar to saddles. For details we refer to [11]. In each topological region, there is a family of periodic orbits close to the heteroclinic, respectively, homoclinic, orbit. This observation gives reason for the following segmentation algorithm. At first, we search for critical points in $\Omega$ and stagnation points on $\partial \Omega$. We calculate the directions which separate the different topological regions. In the case of saddle points, these are the eigenvectors of $\nabla v$. Next, we successively place an initial spot in each of the sectors and perform an appropriate field aligned anisotropic diffusion. Let us suppose that a single sector is spanned by vectors $\{s_+ , s_- \}$, where the sign $\pm$ indicates incoming and outgoing direction. The method presented in Section 3 would lead to a closed pattern along one of the above closed orbits for time $t$ large enough. To fill out the interior region, we modify the diffusion as follows: Up to now, the Perona Malik diffusions enhance edges of the current image in both directions normal to the velocity. Henceforth, we select an orientation for a “one sided” diffusion (cf. Fig. 15), i.e., we...
select a unique normal $v^\perp$ to $v$ and consider the diffusion matrix

$$A(v, \nabla \rho) = B(v)^T \left( \alpha G((\nabla \rho_\epsilon \cdot v^\perp)_+) \right) B(v),$$

where $\alpha$ is a positive constant and $(s)_+ := \max\{s, 0\}$. Furthermore, we consider a nonnegative, concave function $f : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ with $f(0), f(1) = 0$ as a source term in the diffusion equation. If the orientation of $\{s_+, s_-\}$ coincides with that of $\{v, v^\perp\}$, then linear diffusion in the direction toward the interior will fill up the complete topological region. A segmentation of multiple topological regions at the same time is possible if we carefully select the sectors to release initial spots. Fig. 16 shows different time steps of the segmentation applied to a convective incompressible flow. So far, we have seen that anisotropic diffusion has strong provisions for flow segmentation as well. In a certain sense, we thereby identify the complement of what is usually extracted in topology recognition. An outstanding advantage of the new method is its numerical stability and its self-sharpening effect due to the edge enhancing strategy. We pay for this by a higher computational complexity. If we apply a standard implementation on a uniform grid of size $n^2$, the segmentation cost is at least $O(n^2)$ compared to an $O(n)$ count of grid cells met by the direct ODE integration to compute the homoclinic and heteroclinic orbits corresponding to the critical points. Fig. 17 shows an adaptive quadtree which allows the same resolution quality for the segmentation function $\rho$ as on a full grid, but now at a much lower cost. We thereby consider a piecewise linear and continuous finite element space on the adaptive quadtree.

11 Conclusions

We have introduced a new method based on the solution of a nonlinear anisotropic diffusion problem for the post processing of vector data. The method can be applied on 2D or 3D domains, as well as on two-dimensional surfaces embedded in $\mathbb{R}^3$. From a mathematical point of view, one of the major advantages is that it is based on a physically intuitive continuous model, i.e., streamline aligned diffusion. Most of the properties can be discussed on this level. Finally, it is discretized in an appropriate way making use of recent and efficient numerical algorithms.

From the authors’ point of view, exciting future research directions are further investigations of flow visualization in 3D. The exploiting of adaptive finite element paradigms and ordering strategies for the unknowns will be especially key issues to reducing the computing costs.

Furthermore, a visualization approach based on anisotropic diffusion and applicable for time dependent vector fields is a challenging topic. Finally, the anisotropic flow segmentation also carries provisions for the identification of interesting flow regions in 3D, such as recirculation zones and vortex cores.

Further results and the algorithm running on an $n \times n$ 2D vector array is available as a source code at http://www.iam.uni-bonn.de/FktAna_NumMath/Num_Vis/projekte/flow_visualization/.

Acknowledgments

The authors would like to acknowledge Karol Mikula and Jarke van Wijk for inspiring discussions and many useful comments on image processing and flow visualization. Furthermore, they thank Eberhard Bänisch from Bremen University, Klaus Johannsen from Stuttgart University, Konrad Polthier from the Technical University at Berlin, and Wolfram Rosenbaum from Bonn for providing the incompressible flow data sets, the porous media simulation data, the minimal surface data, and the MHD simulation data sets.

References


Udo Diewald studied mathematics and physics at the University at Kaiserslautern and at Bonn University. He spent one year at the University of Wales at Aberystwyth. Currently, he is working on his Diploma thesis. His interests are geometric evolution problems and nonlinear diffusion methods in image and surface processing.

Tobias Preußer studied mathematics at the University of Bonn, Germany, and at the Courant Institute at New York. He received his Diploma degree in 1999. Currently, he is working on his PhD thesis in the field of anisotropic diffusion methods in multiscale feature extraction. While a student, he worked for the National Research Center for Information Technology (GMD). In the spring of 2000, he received a grant from the European Community for his work at the Italian CINECA supercomputing center.

Martin Rumpf received his Diplom degree and his PhD in mathematics from Bonn University in 1989 and 1992, respectively. He has been a professor of applied mathematics at Bonn University since 1996. His research interests are numerical methods for nonlinear partial differential equations, adaptive finite element methods, and computer vision. Furthermore, he is concerned with new flow visualization techniques and efficient multiscale methods in scientific visualization.