On Moving Least Squares Based Flow Visualization

Harald Obermaier[1,2], Martin Hering-Bertram[3], Jörg Kuhnert[2], and Hans Hagen[1]

1 **Fraunhofer ITWM, Germany**
2 **University of Kaiserslautern, Germany**
3 **Rhine-Waal University of Applied Sciences, Germany**

Abstract

Modern simulation and measurement methods tend to produce meshfree data sets if modeling of processes or objects with free surfaces or boundaries is desired. In Computational Fluid Dynamics (CFD), such data sets are described by particle-based vector fields. This paper presents a summary of a selection of methods for the extraction of geometric features of such point-based vector fields while pointing out its challenges, limitations, and applications.

Keywords and phrases Moving Least Squares, Approximation, Flow, Scientific Visualization

Digital Object Identifier 10.4230/OASIcs.VLUDS.2010.55

1 Introduction

Computational Fluid Dynamics simulations allow the mathematically and scientifically well-founded modeling of fluid flow. Results of such simulations include the generation of pressure, velocity and temperature fields in domains with pre-defined boundary geometry for given liquids. While previous simulation techniques are mainly based on grids to construct computational meshes, modern particle-based methods, such as Smoothed Particle Hydrodynamics (SPH) and Finite Pointset Methods (FPM) [25], avoid the use of an explicit neighborhood relation in the simulation domain to cope with dynamic, rapidly changing boundary geometry and free surfaces. Due to these capabilities, grid-less methods are able to overcome challenges of industrial simulations such as stirring or mixing of multi-phase or heterogeneous liquids, without the need of frequent updates of neighborhood structures. This new generation method of flow field data poses interesting questions to the visualization community, as common visualization techniques often rely on structured data for feature extraction and rendering. In this paper, we give an overview of the main challenges arising from the lack of a pre-defined neighborhood and summarize existing work on moving-least-squares-based [16, 29] grid-less vector field segmentation. The main focus lies on the extraction of geometric flow features [24, 27, 2, 10] that can be used to segment point-based vector fields, covering an important topic of flow visualization, as (topological) segmentation [19, 30] is a key question in many of the affected application areas of meshfree CFD. *Moving Least Squares* (MLS) as the chosen reconstruction method is well accepted in the area of surface, object, and point-cloud reconstruction [1] and is also suitable for the problems posed in flow field approximation.

Section 2 gives an overview of challenges in grid-less vector field processing, proposes appropriate solutions, and discusses the concept of integral feature extraction in meshfree vector fields. In sections 3 and 4 we detail existing work on MLS-based stationary and time-varying vector field segmentation. Section 5 briefly recapitulates application areas and results of the presented papers. This paper is concluded in Section 6.

Visualization of Large and Unstructured Data Sets– IRTG Workshop, 2010.
Editors: Ariane Middel, Inga Scheler, Hans Hagen; pp. 55–63
OpenAccess Series in Informatics
OASICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2 Challenges

Challenges in the context of geometric feature extraction from grid-less fields rise from two different fields, namely i) reconstruction of a continuous field function and ii) extraction of feature geometry. On the one hand, the lack of an explicit neighborhood structure in the data set itself poses questions about the method of field approximation and domain decomposition, on the other hand, the defined geometric segmentation structures are desired to provide means for binary classification and should therefore hold information in the form of tessellations. We regard the following challenges to be the main questions to be answered in the context of point-based vector field feature extraction:

- Interpolation/Approximation schemes (Section 2.1)
- Domain decomposition (Section 2.2)
- Robust and correct boundary treatment (Section 2.3)
- Efficient visibility querying (Section 2.4)
- Definition of grid-less feature extraction methods (Section 2.5)

These challenges are detailed and discussed in the following.

2.1 Field Reconstruction

In most standard flow fields, the given grid-based simulation data implies the use of trilinear, barycentric, or higher order geometric interpolation based on the underlying cell types [26]. In contrast, the absence of a computational mesh in point-based fields requires the use of scattered data interpolation schemes [29], which can be adapted for the interpolation or approximation of grid-less flow fields [17]. A common property of these methods is the use of implicit local neighborhoods in the form of appropriate weighting functions. As such, *Radial Basis Functions* (RBF) and related techniques [6] implicitly define a spherical neighborhood around the point of evaluation. One polynomial reconstruction method is the MLS approximation scheme [16], where the quadratic distance of a polynomial f of a given degree to a set of n discrete function-values f_i at $p = (x\ y\ z)^T$ is minimized with respect to a pre-defined weighting function ω:

$$min \left\{ \sum_{i}^{n} \omega(p, p_i) ||f(p_i) - f_i||^2 \right\} \tag{1}$$

The central advantage of grid-less approximation is the independence of a computational grid, i.e. that solutions are governed by the field's value rather than by the choice and characteristics of a (static) neighborhood structure. Moreover, concrete neighborhood structures for a field may not be defined in a unique way (cf.: decomposition of cubes into tetrahedra). However, faithful reconstruction of a data set is only possible, if the same interpolation method is used during creation, i.e. simulation of the phenomenon, and visualization of the data set. Reconstruction properties of MLS, for example, depend heavily on the chosen weighting function. In the case of ω being a two-dimensional Gaussian function, changes in the variance parameter, shape or radius of the smoothing function have great impact on the output with respect to scale space [28]. Especially in data sets with inhomogeneous particle densities, choice of the appropriate smoothing radius has an influence on the reconstructed function and is also a major factor contributing to increased computation times. The reconstructed functions are highly sensitive to changes in smoothing

length, as too small radii introduce noise or lead to singular systems, and too large radii blur important features of the field and reduce reconstruction accuracy.
One of the main reasons why grid-less approximation techniques are outperformed by grid-based methods is the computationally expensive gathering and weighting step of particle neighbors. As discussed in the next section, caching of particle data contributes to faster field approximation. In the case of MLS with a linear polynomial of the form $f(p) = c \cdot (1\ x\ y\ z)^T$, the following linear system of equations needs to be solved for c:

$$\left(\sum_i^n \omega(p, p_i) \begin{pmatrix} 1 & x_i & y_i & z_i \\ x_i & x_i^2 & x_i y_i & x_i z_i \\ y_i & x_i y_i & y_i^2 & y_i z_i \\ z_i & x_i z_i & y_i z_i & z_i^2 \end{pmatrix} \right) c = \sum_i^n \omega(p, p_i) \begin{pmatrix} 1 \\ x_i \\ y_i \\ z_i \end{pmatrix} f_i \tag{2}$$

We note, that the symmetric matrices resulting from a product of base-vectors on the left side of (2) is independent of the point of evaluation and is completely determined by the position of a data point p_i. Following this observation, matrix creation can be relocated to a pre-processing step, pre-computing one such matrix for each data point, thus removing the expensive matrix creation step from approximation. It is notable, that MLS matrices of higher order include those of lower order. Advantages of MLS over other scattered data approximation techniques are mathematical simplicity, and ease of control over polynomial degree and approximation error.

2.2 Domain Decomposition

Mesh-free data sets miss the data clustering property of a computational mesh that is available in grid-based fields. The use of local scattered data approximation techniques for function reconstruction suggests the utilization of suitable domain decomposition schemes to reduce computational complexity by restricting the processed point set during approximation [20, 9, 13]. The most important reason for using a domain decomposition scheme is locality during the neighbor gathering step of approximation with compactly supported kernels. An ideal domain decomposition scheme should have the following properties:

1. Fast identification of enclosing/neighboring cells
2. Good caching properties of it contents
3. A small memory footprint
4. Support of parallelization
5. Adaptivity with respect to particle densities

When used in time-varying vector fields, a fast reorganization of the decomposition scheme is desired as well. Properties 1.-4. can be guaranteed by a simple uniform grid layer. The memory footprint of the grid consists of storing the grid implicitly in the form of grid origin, a number of cells for each dimension and uniform values for cell width, height and length. Further memory is used by a list of indices of the points contained in every cell. Identification of relevant cells during field evaluation is as low as three division operations. Parallelization of particle advection or clustering of the domain is straight-forward in the case of the data set being shared between processes. Parallelization with distributed memory requires the inclusion of neighboring grid-cells in the form of ghost cells. Improved caching is obtained by reordering points' memory locations according to their cell membership.
This straightforward approach is however not able to fully incorporate varying particle

densities. While small variations in particle densities can be handled by reducing the width of the approximation kernel as long as the maximal radius does not exceed cell size, large variations lead to gathering of unnecessarily large particle sets in areas with high densities. Multiple layers of uniform grids, as in *Adaptive Mesh Refinement* [5], or even more adaptive data structures such as kd-trees [29] can be used in this context to allow adaptive cell sizes, if the increased time in cell identification is justified by shorter gathering times.

2.3 Boundary Treatment

There are different types of boundaries in point-based CFD [18]. Type one is the triangulated boundary geometry defined by the device or object which is subject to fluid simulation, such as parts of stirring machines, planes or vehicles. Triangulated boundaries are commonly used to represent more static behavior than free surfaces, especially their topology tends to be stationary. Type two are fluid boundaries, i.e. free surfaces, where the simulated fluid moves without being restricted by a solid geometry. This often appears in simulations of stirring devices, where liquid is moved by mixing blades and is bound to a device by gravity or viscous force only. The third type of boundary is a multi-phase or fluid interface between different liquids. Boundaries of the latter two cases are generally free surface boundaries/field discontinuities and require special treatment [18].
During feature extraction and field approximation, a position outside of the data set is detected either by a boundary collision or visibility test with a triangulated boundary as described in the next section, or by analysis of the neighboring data points. In the latter case, statistics [15] like a large empty angular segment in the neighborhood or a point (the domain of the filter kernel) indicates, that the current point of evaluation is outside of the data set, see Figure 1 (a). While this covers most convex boundary shapes, accuracy in boundary detection decreases near concave regions. Similarly, smoothness measures [11] can be used to approximate free boundaries.

2.4 Visibility Queries

In grid-based visualization, visibility queries in vector fields play a minor role, since the computational mesh defines data set boundaries and effectively separates the domain of the flow field from flow obstacles. However, in meshfree particle systems, visibility queries have to be executed prior to the particle gathering step during field approximation to avoid that particles located on opposing sides of obstacles influence each other [23] as shown in Figure 1 (b). Such visibility queries can either be performed during neighborhood-finding [4, 3] or rely partly on a pre-computed visibility data structure, subdividing the field volume into regions with homogeneous visibility properties, as provided by Binary Space Partitioning or related schemes [7]. The former method is based on run time ray-triangle intersections to determine visibility of a certain data point from a point of evaluation. As higher order approximation techniques require a large number of neighboring points, the amount of required visibility tests for one vector field approximation often has a large impact on performance. Thus, visibility tests have to be both reduced to a minimum number and optimized performance wise.

2.5 Extraction and Segmentation Techniques

Geometry extraction in vector fields is often motivated by industrial mixing applications, where domain experts need to be able to analyze mixing quality of different devices. While

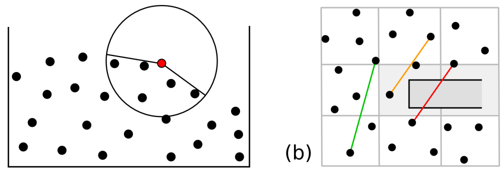

Figure 1 (a) Point of evaluation (red) outside of a free surface. (b) Visibility querying by ray testing. One query (green) is omitted, as affected cells do not contain obstacles. Two other rays are checked for intersection, one of which detects an obstacle (red).

certain other aspects of mixing, such as pressure and temperature are subject to optimization as well, velocity defines the central criterion for parameter optimization, as it directly influences the behavior of real and virtual material interfaces. Real interfaces between different liquids, as well as virtual interfaces defined by the user, device geometry or field topology and their evolution over time and space need to be analyzed in stationary fields and time-varying vector fields in order to be able to interpret the output of a complex mixing process. As demonstrated in the remainder of this paper, a selection of flow structures can be captured by the definition of integral flow features.
Integral flow features [12] are traces described by infinitesimal particles as they are advected through a vector field $f : \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$. Positions p on these traces are defined by the solution of the basic governing differential equation $\frac{dp}{dt} = f(p(t), t)$, with $p(t_0) = p_0$. Work summarized in the following sections is concerned with feature extraction using integral line or surface geometry S at time t as defined by (3).

$$S(r, s, t) = c(s) + \int_{t-r}^{t} f(S(r - (t - x), s, x), x)dx \tag{3}$$

where $c : \mathbb{R} \rightarrow \mathbb{R}^n$ is a univariate point seeding curve parametrized by s. For integral lines, c is a point. Individual instances of lines or stream particles are identified by their age parameter $r \in [0, t]$. The following sections summarize a selection of existing work in the context of MLS based grid-less vector field processing that rely on the methods presented in the previous sections.

3 Extraction and Segmentation I: Stationary Flows

The boundary of instantaneous vector fields is static by definition. The work by Obermaier et al. [23] focuses on the topological segmentation of stationary mixing processes, see Figure 2. It is observed, that in stationary fields of mixing processes, flow obstacles such as mixing blades are the primary source of flow separation due to the lack of critical points or lines in practical applications. Thus, separation and attachment lines on inner flow obstacles are detected on the projected two-dimensional flow field by the criteria defined by Kenwright et al. [14] and used as source for the generation of integral separation structures. As such, separation and attachment lines define lines, where flow meets an obstacle or separates from it, stream surfaces integrated in forward or backward direction at these rakes represent three-dimensional separatrices. Modifications [23] to previous stream surface integration techniques [12, 10] to accommodate data set boundaries that violate no-slip conditions to allow line integration on triangulations as well as surface splitting caused by different flow obstacles yield a set of such separation surfaces. Identification and processing of different surface-surface intersections allows the geometry-based computation of surface segmentation. The resulting sets of stream surface segments are reorganized and combined to form distinct, non-overlapping stream volumes that effectively segment the underlying domain into topologically homogeneous volumes.

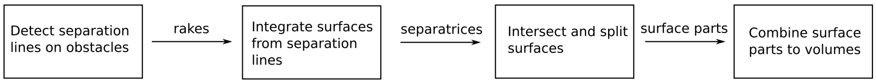

Figure 2 Steps of stationary stream volume extraction.

4 Extraction and Segmentation II: Time-Varying Flows

A more challenging and physically more realistic setting are time-varying data sets. In time-varying fields, changing data set boundaries and particle sets usually require the re-computation of domain decomposition or grid information for every time-step.

4.1 Two-Dimensional

In Obermaier et al. [22], a streak line based approach for the segmentation of two-dimensional, time-varying flow simulations into continuous areas, so called streak areas is introduced. Data set boundaries with non-vanishing velocities are incorporated in the streak line creation process, as they can be the source of streak line stretching, attaching and separation behavior. Streak lines are continuously seeded at user specified positions, indicating initial material boundaries. As the streak line is advected by the flow field, individual segments of the line are subdivided if either a maximal segment length is exceeded, or angles between neighboring segments fall below a given threshold, thus achieving curvature adaptivity and maintaining a pre-defined streak line resolution. Solid flow obstacles with non-zero boundary-velocity require special attention during the line generation process. In every time step, all line segments are tested for collisions with such obstacles. If a collision is detected at time step t_1, the affected segment is recursively split in half in $t_0 = t_1 - 1$ and the resulting two segments are advected and again tested for collision with a boundary in time t_1. This is repeated until no more segment-obstacle collisions are detected. Streak particles whose trajectory hits the boundary move along the projected field on the boundary, until they are released at separation points [14].

4.2 Three-Dimensional

In the three-dimensional case, streak surfaces provide means for segmenting a time-varying vector field when seeded at initial material interfaces or separation curves. A method to generate curvature adaptive streak surface geometry in grid-less flow is presented in [21], where MLS is used for vector field approximation as well as for surface curvature approximation. Particles seeded at a rake in a time-varying vector field naturally define a point-based streak surface. The true problem in accurate streak surface generation is to guarantee a certain desired surface mesh resolution. One solution to this adaptivity problem is to maintain a curved Delaunay-type surface mesh [8] by edge flipping as the surface is advected and deformed by the flow field. Triangle circumradii of this streak surface mesh serve as a measure of streak particle density and are used to insert new particles at regions with high surface curvature. As large Delaunay triangles reflect a low particle density, high MLS curvature at such regions indicates the need of additional particle insertion. In order to reduce surface artifacts resulting from bad particle insertion positions and increase accuracy of adaptive surfaces when compared to high resolution streak surfaces, particle insertion positions in highly curved regions are traced back in time until a time step with a sufficiently low surface curvature is reached, before actual particle insertion is performed. Figure 5 shows four streak surfaces at different time-steps, parting a time-varying vector field into two volumes.

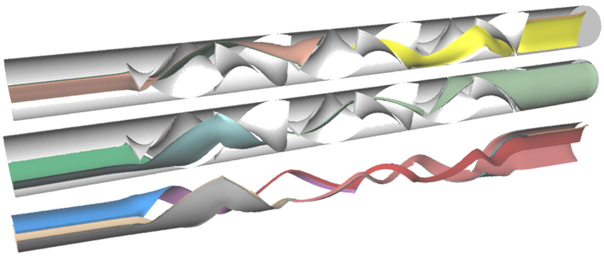

Figure 3 Visualization of two single stream volumes and a volume pair (bottom). The data set is segmented into a total of 64 distinct non-overlapping volumes.

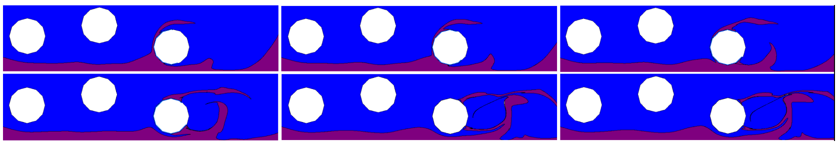

Figure 4 A sequence of six consecutive streak areas. The data set is segmented by a user specified material interface represented by a streak line. Intersections of the line with flow obstacles as well as cases where the streak line separates from the boundary geometry can be observed.

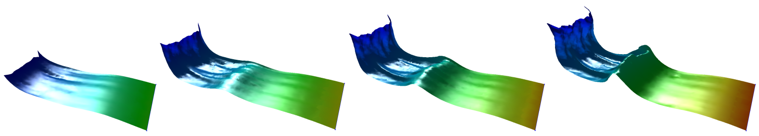

Figure 5 Four consecutive renderings of a segmenting streak surface colored according to seed time.

5 Results and Applications

The given examples demonstrate integral line and surface generation as tool to segment data sets for further analysis, processing and as general visualization method. Figure 3 depicts instantaneous flow in an industrial mixing process, however in reality time-dependent vector fields as shown in Figures 4 and 5 are more common, while sharing properties with stationary fields with respect to segmentation techniques. The shown results provide information about flow segmentation and allow domain experts to perform parameter optimization of mixing devices. The presented solutions to the main challenges of grid-less vector field segmentation are robust and applicable in stationary, as well as time-varying cases. Distinct segments of the data set that are bounded by integral feature geometry allow the computation of volumina, facilitate geometric intersection of different overlapping segments and form a basis for binary classification of points in the data set and voxel based visualization techniques such as volume rendering.

6 Conclusions and Outlook

We have given an overview of advances in the field of MLS-based grid-less flow visualization and pointed out its challenges and capabilities. It was shown, that segmentation of such vector-fields is of extreme importance for certain industrial applications, providing insights about mixing and interface evolution. While the advantages of grid-less flow fields such as meshless modeling of boundaries and interfaces and grid independence carry over to the field of visualization by avoiding cell identification and reconstruction artifacts caused by grid characteristics, the drawback of expensive field approximation and visibility querying remains a hurdle on the way to real-time surface generation in point-based vector fields. The presented challenges of grid-less vector fields are subject to ongoing research and new solutions are expected to improve expressiveness and performance of approximation, extraction, and visualization techniques. Further work is required in the field of domain decomposition and distribution techniques, furthermore, a promising direction of future research is the study of topology changes caused by parameter variations in scattered data interpolation schemes.

References

1 M. Alexa, J. Behr, D. Cohen-Or, S. Fleishmann, D. Levin, and C. T. Silva. Point set surfaces. In *Vis. '01*, pages 21–28, 2001.

2 B. G. Becker, N. L. Max, and D. A. Lane. Unsteady flow volumes. In *Proc. of the 6th Conf. on Vis. '95*, page 329, 1995.

3 T. Belytschko, Y. Krongauz, M. Fleming, D. Organ, and W. K. S. Liu. Smoothing and accelerated computations in the element free galerkin method. *J. Comp. Appl. Math.*, 74(1-2):111–126, 1996.

4 T. Belytschko, Y. Y. Lu, and L. Gu. Element-free galerkin methods. *Internat. J. for Numerical Methods in Engineering*, 37:229–256, 1994.

5 M. J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *J. Comp. Physics*, 82(1):64–84, 1989.

6 T. Bobach and G. Umlauf. Natural neighbor concepts in scattered data interpolation and discrete function approximation. In *GI LNI, Visualization of Large and Unstructured Data Sets*, pages 23–35, 2007.

7 D. Cohen-Or, Y. L. Chrysanthou, C. T. Silva, and F. Durand. A survey of visibility for walkthrough applications. *IEEE TVCG*, 9(3):412–431, 2003.

8 R. Dyer, H. Zhang, and T. Möller. Delaunay mesh construction. In *Eurographics Sym. on Geometry Processing*, pages 273–282, 2007.

9 M. S. Floater and A. Iske. Multistep scattered data interpolation using compactly supported radial basis functions. *J. Comp. Appl. Math*, 73:65–78, 1996.

10 C. Garth, H. Krishnan, X. Tricoche, T. Bobach, and K. I. Joy. Generation of accurate integral surfaces in time-dependent vector fields. *IEEE TVCG*, 14(6):1404–1411, 2008.

11 T. Gutzmer and A. Iske. Detection of discontinuities in scattered data approximation. *Numerical Algorithms*, 16:155–170, 1997.

12 J. P. M. Hultquist. Constructing stream surfaces in steady 3d vector fields. In *Proc. of Vis '92*, pages 173–175, 1992.

13 A. Iske and J. Levesley. Multilevel scattered data approximation by adaptive domain decomposition. *Numerical Algorithms*, 39:187–198, 2005.

14 D. N. Kenwright, C. Henze, and C. Levit. Feature extraction of separation and attachment lines. *IEEE TVCG*, 2(5):135–144, 1999.

15 S.-B. Kim. Eliminating extrapolation using point distribution criteria in scattered data interpolation. *Comp. Vis. Image Underst.*, 95(1):30–53, 2004.

16 D. Levin. The approximation power of moving least-squares. *Mathematics of Computation*, 67:1517–1531, 1998.

17 S. Li and W. K. Liu. *Meshfree Particle Methods.* Springer Berlin, 2007.

18 G. R. Liu and M. B. Liu. *Smoothed particle hydrodynamics: a meshfree particle method.* World Scientific, 2003.

19 K. Mahrous, J. Bennett, G. Scheuermann, B. Hamann, and K.I. Joy. Topological segmentation in three-dimensional vector fields. *IEEE TVCG*, 2(10):198–205, 2004.

20 L. B. Montefusco and C. Guerrini. A domain decomposition method for scattered data approximation on a distributed memory multiprocessor. In *EDMCC2: Proc. Dist. Memory Computing*, pages 274–282, 1991.

21 H. Obermaier, M. Hering-Bertram, J. Kuhnert, and H. Hagen. Generation of adaptive streak surfaces using moving least squares. In *Proc. of Dagstuhl Scientific Vis. Seminar (to appear)*, 2009.

22 H. Obermaier, M. Hering-Bertram, J. Kuhnert, and H. Hagen. Volume deformations in grid-less flow simulations. *Comp. Graph. Forum (Proc. of EuroVis)*, 28(3):879–886, 2009.

23 H. Obermaier, J. Kuhnert, M. Hering-Bertram, and H. Hagen. Stream volume segmentation of grid-less flow simulation. In *Topological Data Analysis and Vis.: Theory, Algorithms and Applications (Proc. of TopoInVis), (to appear)*, 2009.

24 T. Schafhitzel, E. Tejada, D. Weiskopf, and T.s Ertl. Point-based stream surfaces and path surfaces. In *Graph. Interface 2007*, pages 289–296, 2007.

25 S. Tiwari and J. Kuhnert. Finite pointset method based on the projection method for simulations of the incompressible navier-stokes equations. *Springer LNCSE: Meshfree Methods for Partial Differential Equations*, 26:373–388, 2002.

26 S.-K. Ueng, C. Sikorski, and K.-L. Ma. Efficient streamline, streamribbon, and streamlube constructions on unstructured grids. *IEEE TVCG*, 2(2):100–109, 1996.

27 W. von Funck, T. Weinkauf, H. Theisel, and H.-P. Seidel. Smoke surfaces: An interactive flow visualization technique inspired by real-world flow experiments. *IEEE TVCG*, 14(6):1396–1403, 2008.

28 J. Van De Weijer and R. Van Den Boomgaard. Least squares and robust estimation of local image structure. *Internat. J. of Comp. Vision*, 64:143–155, 2005.

29 H. Wendland. *Scattered Data Approximation.* Cambridge University Press, 2004.

30 A. Wiebel, X. Tricoche, and G. Scheuermann. Extraction of separation manifolds using topological structures in flow cross sections. In *Topology-Based Methods in Vis. II*, pages 31–44, 2009.