

Visualization of Multi-Variate Scientific Data

R. Fuchs^{1,2} and H. Hauser^{2,3}

¹Institute of Computer Graphics and Algorithms, Vienna University of Technology, Austria

²VRV is Research Center, Vienna, Austria

³Department of Informatics, University of Bergen, Norway

Abstract

In this state-of-the-art report we discuss relevant research works related to the visualization of complex, multi-variate data. We discuss how different techniques take effect at specific stages of the visualization pipeline and how they apply to multi-variate data sets being composed of scalars, vectors and tensors. We also provide a categorization of these techniques with the aim for a better overview of related approaches. Based on this classification we highlight combinable and hybrid approaches and focus on techniques that potentially lead towards new directions in visualization research. In the second part of this paper we take a look at recent techniques that are useful for the visualization of complex data sets either because they are general purpose or because they can be adapted to specific problems.

Keywords: visualization, scientific visualization, multi-variate data visualization, multi-dimensional data visualization, multi-field visualization, scalar data visualization, flow visualization, tensor visualization, illustrative visualization

ACM CCS: I.3.8 [Computer Graphics]: Computer Graphics Applications

1. Introduction

In the last decade, there has been enormous progress in scientific visualization [Max05]. Still the visualization of multi-variate continuous three-dimensional (3D) data, especially when the data is also time-dependent, remains a great visualization challenge. Recent publications have stated a shift in visualization research from a classical approach dealing with visualization of small, isolated problems to a new kind of challenge: visualization of massive scale, dynamic data comprised of elements of varying levels of certainty and abstraction [TC06]. It is a well-known fact that we are experiencing an increase in the amount and complexity of data generated that exceeds our ability to easily understand and make sense of it. To address this development, Lee et al. [LCG*02] have indicated the work with multi-variate data sets as one important task for future visualization research that will require significant advances in visualization algorithms. Munzner et al. [MJM*06] addressed top scientific research problems and identified multi-field visualization as one of the central questions for future research.

Kirby [KML99] defines the development of a visualization method as breaking the data into components, exploring the relationships among them and visually expressing both the components and their relationships. To visualize the complexity of multiple data components and the relationships between them researchers have sought to find ways to combine the advantages of different types of visualization techniques. In this article, we give an overview of existing work in scientific visualization that points in this direction. Intentionally, we leave out a vast amount of work focusing on solving specific visualization tasks which repeatedly have been discussed in other related reports.

By targeting the use of complex visualization techniques we do not speak in favour of a ‘more is better’ approach, we would rather like to stress the importance of feature- and knowledge-driven visualization. The aim of this report is to give an overview of current techniques from various disciplines dealing with complex multi-dimensional scalar, vector and tensor data sets with the goal in mind to apply them to situations where these types of data are present at the same

time in three dimensions. Hesselink et al. [HPvW94] give a short overview of research issues in visualization of vector and tensor fields that is still valid today. They declare four goals for future visualization research: feature-based representation of the data, comprising reduced visual complexity, increased information content and a visualization that matches the concepts of the application area. We see the three main advantages of hybrid (or multi-method) visualization techniques: first, improved effectiveness of visualization because each part of the data can be visualized by the most appropriate technique. Second, the ability to visualize multi-variate data sets while minimizing visual clutter at the same time. And third a separation between two questions that are too often intermingled: how to visualize versus what to visualize. An overview of current work on hybrid- and combined visualization algorithms can be a starting point for future visualization research in the direction of flexible and user-task driven visualization.

When talking of complex volumetric data we distinguish between different types of data. We distinguish between several ways for data to have multiple attributes. The terms multi-variate and multi-dimensional are related to the structure of the data and the relation of data items to the physical world.

- Multi-variate data is a general description for a type of information where each data item x is represented by an attribute vector such that $x = (a_1, \dots, a_n)$.
- Multi-dimensional data is a special case where some of the attributes are independent of each other and are related to physical dimensions such as space or time.

The terms multi-channel, multi-modal, multi-field and multi-valued are related to the method of data acquisition and representation. In the field of scientific visualization the attributes are most often samples from a continuous quantity inside some spatial domain. In this case additional questions such as the location of sample points and the goodness of interpolation functions are important for reconstruction of the underlying continuous field.

- Multi-channel data contains data from possibly different physical quantities (e.g. voltage, temperature and humidity) acquired through multiple measurement channels.
- Multi-modal data describes the result from data acquisition where the data attributes describe one physical object that has been scanned using multiple input modalities. The different modalities are not necessarily sampled on the same positions in space and the relative position of the coordinate systems for each modality can have error ranges due to the registration.
- Multi-field data contains multiple attributes that are grouped into multiple physical fields such that each com-

ponent of a group needs the others to be interpretable (e.g. a data set containing fluid velocities and tracer diffusion tensors). Again, each group can have individual sampling positions.

- Multi-valued data can describe two types of data: it is sometimes used synonymously to the term multi-variate. Other authors use the term multi-valued for data sets where for a quantity multiple values are given. This is the case, for example when multiple redundant sensors measure the same quantity but with possible measurement errors.

As an example we can think of a scan of the human chest using combined computed tomography (CT) and magnetic resonance imaging (MRI). The CT will capture the bones (e.g. ribs, spine) best, resulting in a single scalar field. The MRI scan is more accurate in measuring soft tissue, resulting in a second scalar field and with the use of a contrast agent it is possible to obtain vector information about the blood flow. Using multiple scanning modalities, we have obtained a multi-variate data set where the data elements are indexed by three spatial dimensions. At each grid point we have two scalars that describe the same physical situation and one vector with three components describing the x , y and z -direction of the flow. It is common for scientific data to have more spatial structure than general multi-variate data sets (e.g. census data or questionnaire results).

Even though there is a wealth of algorithms for automated data analysis, these are only applicable when we know what we are looking for in the first place. Automated processing cannot generate new features and understanding beyond what is already known and might even remove important aspects of the data. Ward's Mantra [War02] 'I'll Know it When I See it', stresses the fact that we often rely on our visual pattern recognition system to help us in gaining knowledge of the data. For this reason we need to discover new ways to visualize complex scientific data or adapt the existing ones to deal with the new situation of multi-field data sets.

Different tasks need specific qualities in a visualization method. For example when trying to gain overview of the data a continuous and smooth style might be most appropriate, whereas for information drill-down a researcher might want a specific aspect to be shown in a way as true to the data as possible (i.e. without smoothing, outlier removal, complex transfer functions, etc.). The application of combined hybrid visualization techniques will increase the value of any visualization application, therefore most of the techniques discussed in this paper will be useful for other types of data as well.

1.1. The visualization pipeline

The visualization pipeline is a model that describes how visual representations of data can be obtained following a procedure of well-defined steps [CDH05]. Figure 1 gives an

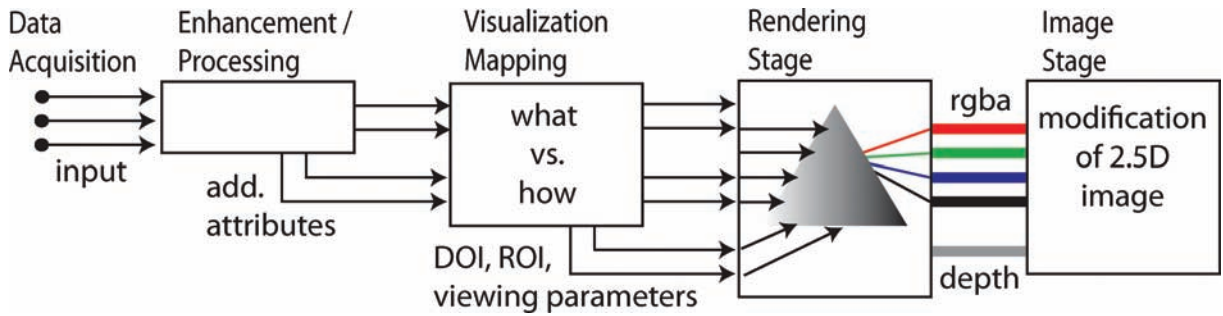


Figure 1: An abstract pipeline for multi-variate scientific visualization.

overview of this pipeline for multi-variate scientific data. The data acquisition step encompasses the measurement or generation of scientific data. In an *enhancement and processing* step the data background knowledge and data storage standards are applied to generate distributable and commonly usable data sets. For example the irregular data points stored internally by a computer tomograph are resampled onto standard grids internally using background knowledge the engineers have on the specifics of each machine. Another example are derived fields that can be computed from the original data, which are especially useful for visualization purposes. During *data filtering and visualization mapping* non-relevant data items are removed or aggregated and the abstract information in the data set is mapped to representatives. For example some types of tensor information (symmetric 3×3 matrices) could be mapped to ellipsoids whereas vectors are mapped to arrows. In the rendering stage the information inside the data and the corresponding representations are translated into an image. In the image stage the final image is manipulated to improve the rendering results.

Standard [see Figure 2(a)]: In traditional scientific visualization very often most of the acquired information is resampled onto a structured grid. During data filtering the user selects values or ranges of data attributes to be shown using a transfer function. In the visualization mapping stage the data elements are mapped to glyphs or volumes. In the rendering stage the data values and viewing parameters contribute to the resulting image. This image can undergo modifications such as colour enhancements or overdrawing (e.g. labels) to generate the final output of the visualization process.

Feature-based [see Figure 2(b)]: Multi-variate visualization using derived quantities uses additionally computed features to improve the visualization, e.g. by using these values in the transfer function design [KKH01] or for colour assignment [GGSC98]. Other important types of derived features are segmentation data, classification data and cluster information. These are very often generated in a (semi)automated fashion that also outputs uncertainty information for the generated information. Many feature-detection algorithms com-

pute spatial structures of domain-specific relevance. These additional fields can be used to improve the visualization of the data [HBH03].

Interactive Visual Analysis [see Figure 2(c)] The SimVis software is an example where recent approaches from the field of interactive visual analysis are integrated into one tool. The linking and brushing concept in SimVis [DGH03] uses different views and combines the users degree of interest specifications into a visualization mapping of the multi-variate data set.

Data intermixing [see Figure 2(d)]: The fourth type of multi-variate visualization combines the data coming from different modalities during the rendering step. The opacity is computed by using a combination operator to integrate the material properties into a single value [CS99]. Figure 2(e) Layering renders different data items separately and combines them in an image fusion step [WFK*02].

Goal [see Figure 2(f)]: A goal could be a visualization technique that would collect all relevant data, derive the inherent information, (interactively) detect all features of interest, match them to an appropriate rendering algorithm and combine the results cleverly to get a high-quality visualization.

1.2. Outline

In the first part of this paper (Sections 1–4) we will discuss existing techniques resulting from applications in various scientific fields such as meteorology, computational fluid dynamics (CFD) simulation, medical imaging and geology. Some types of data are of distinguished importance and appear in so many applications that visualization of these has become a research area in its own respect. Among these are flow, tensor and (time-dependent) scalar data. We will take a closer look at techniques that have been developed in these fields to deal with multi-variate data. In this context we will structure the following discussion relating to the type of data a publication mainly deals with and focus on scalar data in the first chapter, vectorial data in the second and tensorial

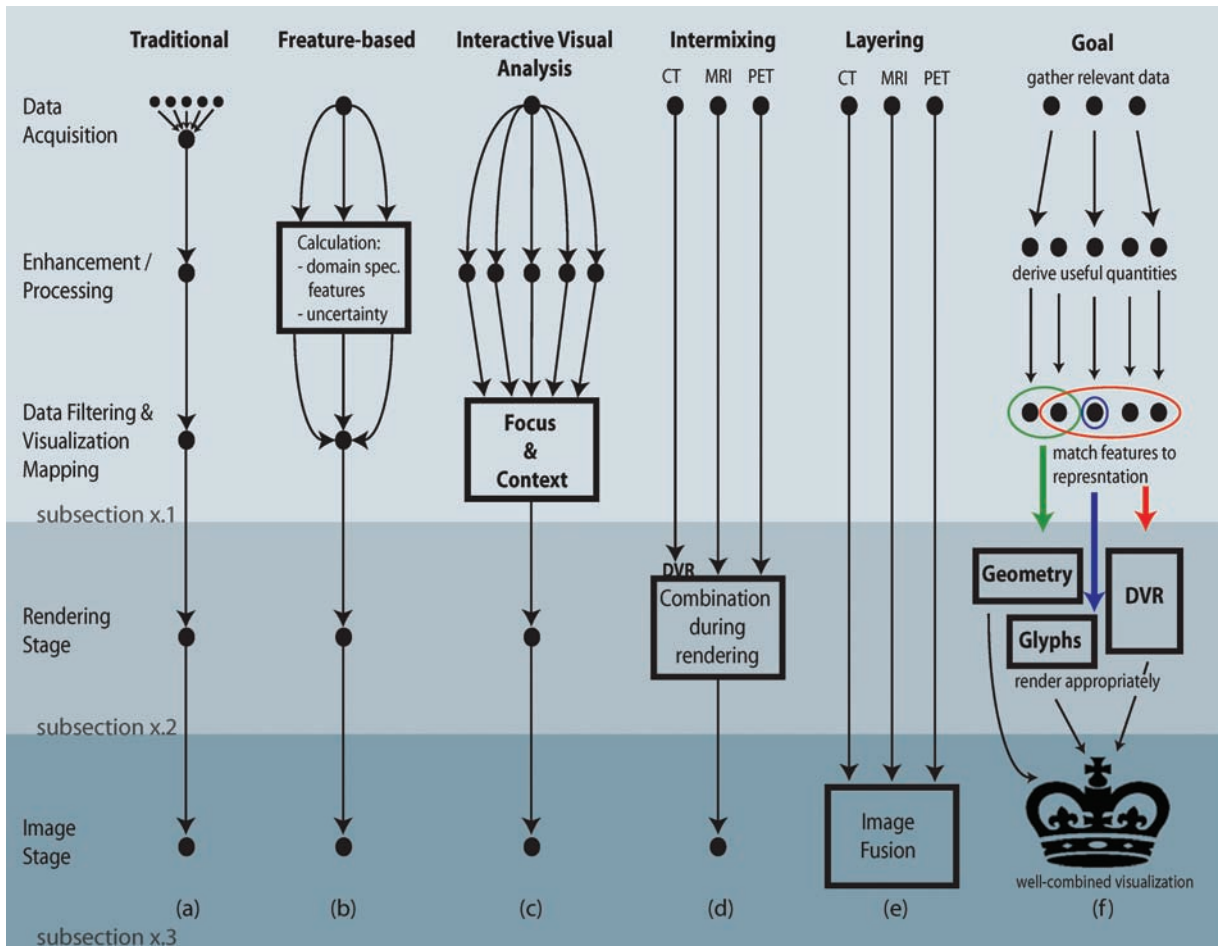


Figure 2: 6 examples of the visualization pipeline used to deal with multi-dimensional data sets and the three stages discussed in this paper (see also Section 1.1).

in the third. At the beginning of each chapter we will give references to survey articles that focus on the classical approaches concentrating on one or few data items of the respective data type at once. In the second part of this paper (Sections 5 and 6) we will give a short overview of existing techniques that may not have been developed in the context of multi-variate visualization but that we consider as highly applicable for this purpose.

In each chapter we classify techniques dealing with multi-variate data sets according to the point of the visualization pipeline where the multi-variate nature of the data is being tackled (see also Figure 2). In each section x we will begin with techniques related to data acquisition, processing and visualization mapping (section $x.1$), discuss techniques based on the rendering stage in the second subsection (section $x.2$) and techniques working on the image stage in the third (section $x.3$).

Even though this paper covers a relevant selection of related work, it lacks completeness due to space limitations.

2. Scalar Data

In this section, we outline multi-method or combinable visualization techniques for multi-variate scalar data.

Common sources for multi-variate scalar data are scanning devices as used in medical imaging and computational simulations. Other types of scalar data result from various measurement devices such as marine sonar, meteorology radar/satellite scans and photographic volumes. The acquisition devices used for medical imaging can be used for other purposes as well (e.g. industrial CT), but medical applications can be considered as one of the most common sources of regular scalar data sets. Because medical data sets are often

images obtained from different sources, the visualization of multi-valued data sets and their registration is a tightly coupled research area in medical image processing. Scalar data visualization is a central issue for medical data. See Engel et al. [EHK*06] for an extensive introduction.

Scalar data can become high dimensional quickly with the addition of various attributes. Scientific data sets are very often segmented or post-processed to extract regions containing different features that are of varying importance to the user—the location of a tumour might be of more interest than any other feature in the data set resulting in additional dimensions. An additional dimension for this kind of high-dimensional data sets results from the uncertainty that comes with automated registration and segmentation algorithms [DKLP02]. If we consider importance, uncertainty or level of interest as additional dimensions to a data set, multi-variate data become even more frequent.

2.1. Techniques in the processing, filtering and visualization mapping stage

It is important that the user is able to interpret the information presented in the visualization. This requirement sets a limit on the complexity of the function mapping data items to visual representations. By reducing the data to the relevant attributes or by computing expressive derived attributes this mapping from data values to visual attributes becomes simpler.

In this section, we discuss visualization techniques that reduce the number of variables before rendering. Two important approaches are feature extraction methods and region of interest (ROI) based methods. Feature extraction methods classify high-dimensional data into features like isosurfaces, topological structures or other application domain related features (such as blood vessels for example). They assign to each point in space a degree of membership to a feature of interest (e.g. a tumour) that can then be visualized using scalar rendering and colour coding. ROI based methods select data items according to their location in space. ROI selection can be considered complementary to attribute based feature detection because it is value independent. Injuries are an example: different types of affected tissue and bone belong to one ROI. The relevant portion of the body is well defined in terms of its spatial properties, but it is difficult to find value ranges describing this ROI.

Interactive visual analysis is commonly based on linking and brushing in multiple views [DMG*05]. Brushing means to select intervals of the data values by drawing selections onto the display area of a view [see Figure 3 (top)]. This way the user can specify a degree of interest in a subset of data items based on their attributes. The degree of interest functions from several linked views (scatterplots, histograms, etc.) are then accumulated using fuzzy-logic operators [DGH03]. The data elements that have attribute values inside these intervals, belong to the focus and are highlighted

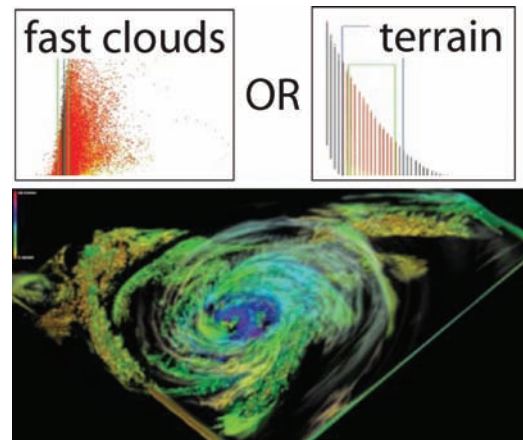


Figure 3: A DVR of a hurricane data set using interactive feature specification and focus + context visualization [DMG*05].

consistently in all views. In the 3D visualization, the features are visually discriminated from the rest of the data in a focus + context visualization style which is consistent in all views.

Tzeng et al. [TLM05] suggested an intelligent systems approach to brushing. The user brushes portions of the volume that are of interest. A machine learning classifier (a neural network or support vector machine) is built from this training set. Based on the classifier the system then determines for each sample whether it belongs to the region of interest or not.

In the field of medical imaging the integration of different volumes into a single visual representation is called data intermixing (this compares to data fusion and volume fusion in other fields). The different modalities (e.g. computed tomography, magnetic resonance imaging or positron emission tomography) can show different, complementary and partially overlapping aspects of the situation. Therefore most algorithms are designed to allow viewing of the original channels alone and more or less sophisticated combinations. A standard approach is to combine data sets based on segmentation information (e.g. the brain is visualized using MRI data, while the skull is shown based on data from the CT channel) combined with colour coding (see Figure 4).

Illumination stage intermixing takes place in the visualization mapping stage: to combine the different attributes in the multi-valued volume voxel V , a combination function takes the attribute values a_1, \dots, a_n directly as input:

$$\text{opacity}(V) := \text{opacity}(\text{combine}(a_1, \dots, a_n)).$$

This way only a single transfer function is necessary, but we have to define a combination function that deals with the different value ranges of the attributes (e.g. using a

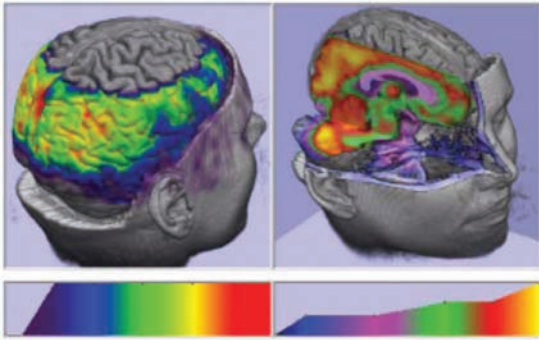


Figure 4: Using multiple transfer functions, region selection and colour coding to combine information from multiple channels [MFNF01] (Image courtesy of I. H. Manssour).

multi-dimensional transfer function). For example Kniss et al. [KKH01] developed a technique for visualization of multi-variate data by applying multi-dimensional transfer functions and derived quantities. In a case study [KHGR02] they apply this approach to meteorological simulation data using 3D transfer functions (for instance two axes map data values and the third the gradient magnitude). A drawback of this method is that multi-dimensional transfer function design is a complicated task and the results are hard to predict.

Another example for a hybrid rendering technique for scalar data was presented by Kreeger and Kaufmann [KK99]. Their algorithm combines volume rendering and translucent polygons embedded inside the volume. They apply their technique to combine an MRI-volume of a human head with an angiogram that visualizes blood vessels. Here the 'how' approach of the visualization (surfaces and volume) are matched to the 'what' context of the data (blood vessels and tissue).

Woodring and Shen [WS06] present a technique to visually compare different time steps of time-varying data sets using Boolean and other operations. The operators *over*, *in*, *out*, *atop* and *xor* compare two timesteps A and B at each voxel to derive a new field.

Another (rare) source of multi-modal data are photographic volumes. The visible human male data set contains vectorial (RGB) colour information at each voxel taken by photographing each slice. Volume rendering is difficult in this context, because a high-dimensional transfer function from 3D to opacity is necessary. Ebert et al. [EMRY02] show how to use a perceptually appropriate colourspace for transfer function design. Ghosh et al. [GPKM03] render multi-channel colour volumes consisting of CT, MRI and colour information on the hardware. Muraki et al. [MKN00] have presented a method to assign colour values to voxels from multi-modal data sets using a neuronal net trained on a photographic volume.

2.2. Rendering stage techniques

Cai and Sakas [CS99] present a ray casting technique that integrates the information of multiple volumes during rendering. Data intermixing is done in the rendering pipeline during accumulation. On the accumulation stage the different modalities are already mapped to opacity and intensity values by their own transfer functions. This means they have the same intensity and opacity range ([0, 1]). Intermixing on the accumulation stage can then be done by defining an additional opacity and intensity evaluation function taking as input the opacities of the different attributes a_1, \dots, a_n :

$$\text{opacity}(V) := \text{combine}(\text{opacity}(a_1), \dots, \text{opacity}(a_n))$$

The authors suggest to use linear or boolean operators for combination. There is a large amount of work in this direction. Ferre et al. [FPT04] for example discuss combination functions that take into account additional values, such as the gradient. Rössler et al. [RTF*06] present a GPU-based implementation of the intermixing technique working with 3D textures and shader programs. Each data volume is rendered separately using an individual shader program allowing for different render modes for the modalities. Then intermixing is done when volume slices are combined in back-to-front order.

The spectral volume rendering technique [NVS00] displays a multi-modal volume using a physics based light interaction model: each material interacts with the light in its specific way. For different modalities, the interaction with several materials at one point in space is simulated. Spectral volume rendering is probably the physically most realistic technique to do illumination stage intermixing in terms of light propagation.

Grimm et al. [GBKG04] developed methods that allow efficient visualization of multiple intersecting volumetric objects that is applicable in the situation of multi-modal volumes. They introduce the concept of V-Objects, which represent abstract properties like illumination, transfer functions, region of interest and transformations of an object connected to a volumetric data source.

2.3. Image stage techniques

Among the visual attributes that can represent data values are colour, transparency, contour lines, surface albedo, texture and surface height. Textures are a versatile medium, that can be computed with approaches such as spot noise, texture advection, bump-mapping and reaction-diffusion techniques. Shen and Interrante [SI05] discuss methods to combine colour and texture to represent multiple values at a single location. Taylor [Tay02] describes a layering system for visualization of multiple fields on the same surface using data driven spots. He also discusses problems that arise due to the layering process. Their finding is that visualizing

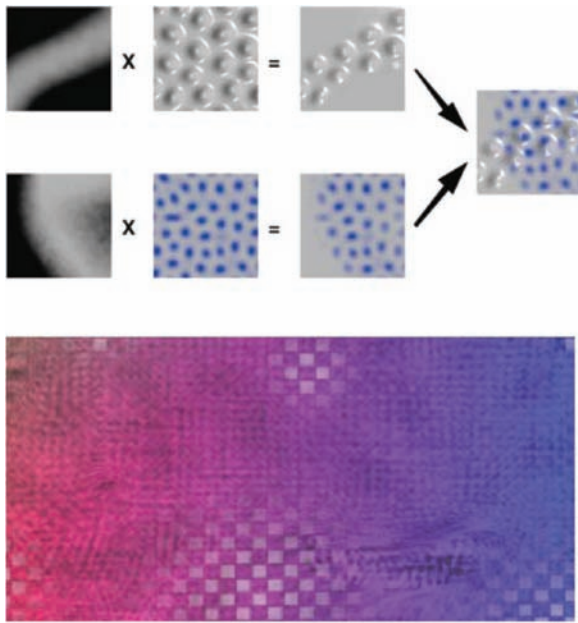


Figure 5: A combination of different bump-mapping styles to visualize multiple fields on the same layer [Tay02] (Image courtesy of R. M. Taylor).

multiple data sets using a different technique for each layer is limited to four different fields because the layers on top either mask or scramble the information below. Special care must be taken to keep the different layers distinguishable, for example by keeping the frequencies of the surface characteristics disjunct. In Figure 5, we see an example how bump-mapping and a reaction-diffusion texture are combined (left). On the right-hand side we see a resulting image using data driven spot textures. House et al. [HBW05] discuss optimal textures for information visualization, including a large user study design, that investigates layered textures for visualizing information located on a surface.

Common problems when dealing with multi-variate scalar data sets include:

1. Registration: Depending on the different resolution of different capturing modalities it is a common problem to register multiple data sets during preprocessing.
2. Scalability: The enormous size of data sets generated by recent acquisition devices such as in electron microscopy poses difficult problems for real-time visualization and interaction approaches.
3. Dimensions: Scalar data sets can now be generated with extremely high resolutions. Often it is difficult to retain the important structures after they are mapped to sub-pixel sizes during rendering.

3. Vector Field and Flow Visualization

In this section, we outline multi-method or combinable visualization techniques for multi-variate vectorial data, of course the cited works are by no means complete and not all important works could be included.

The velocity of a flow is represented by a vector field and a vector field can define a flow, therefore in many applications their visualizations can be considered equivalent [Max05]. Depending on the field of application there are additional variables of importance. In mechanical engineering, the pressure is always also important. Density and temperature are further additional variables. In many cases like the compressible Navier–Stokes equations, all these variables are necessary to describe the physics correctly, so visualizing the velocity alone can never give a complete picture.

Recent surveys and overview articles include: a classification of different flow visualization algorithms and a discussion on derived, second-order data by Hauser [Hau06] and the state of the art report on flow visualization focusing on dense and texture-based techniques by Laramée et al. [LHD*04]. Post et al. [PVH*03] give an overview of feature extraction methods for flow fields.

3.1. Techniques in the processing, filtering and visualization mapping stage

A basic technique in flow visualization is to match the attributes of a data set to physically appropriate representations (‘how’ matched to ‘what’). For example shock waves are mapped to surfaces, dispersed particles are mapped to particle traces or points. We will not repeat every application that uses combinations of standard flow visualization techniques such as lines [ZSH96, KM96], surfaces [Wij93], sub-volumes [SVL91] or dense techniques [IG97]. From the large body of work we can only mention a few examples.

Laramée et al. [LGSH06] discuss the application of texture advection on surfaces for visualization of vector fields defined at a stream surface. In this application tumbling motion of the flow in the combustion chamber of a diesel engine is visualized by seeding a surface that depicts the swirling motion of the flow. This is based on work by van Wijk and Laramée on image space advection [Wij03, LvWJH04]. In their approach parameterization of the surface is not necessary and advection is not computed for pixels occluded by other parts of the surface. The main steps are as follows:

1. Compute flow vectors at vertices of the surface mesh.
2. Project the vector field onto the image plane.
3. Advect texture properties according to the projected vector field.
4. Add shading to the image to convey shape information.



Figure 6: Combining texture advection and surface based flow visualization. Both the location of the iso-surface and its texture convey information about the flow [LGS06].

This approach allows interactive frame rates for animated flow textures. Both the shape of the surface and the texture can convey meaning to the user (see Figure 6).

Because topology based visualization techniques feature sparse and economic screen usage, there is ample space left for additional information. Hauser and Gröller suggest a two step approach [HG00]. In the first step topology information is computed. Examples are fixed points and their Jacobins and higher order attractors. This is the classical step in topology visualization and in most cases the second step is not very intricate: different types of topological elements are visualized by different glyphs representing attracting, repelling and saddle points and separation lines [HH91]. This second step is now augmented by showing a visualization of the flow structure in a neighbourhood of the critical point or visualizing the Poincaré map (see Figure 7).

There is a lot of work in how the components of multi-variate data can be visualized. Sauber et al. [STS06] present multifield-graphs that deal with the question how the correlations between the components in the data can be shown. They introduce derived correlation fields that describe the strength of correlation between two variables at each point in space. The user can visualize correlation between scalar fields and vector fields. This also shows that the inherent information in multi-variate field that groups several variables to vectors and tensors can be useful when deriving additional information.

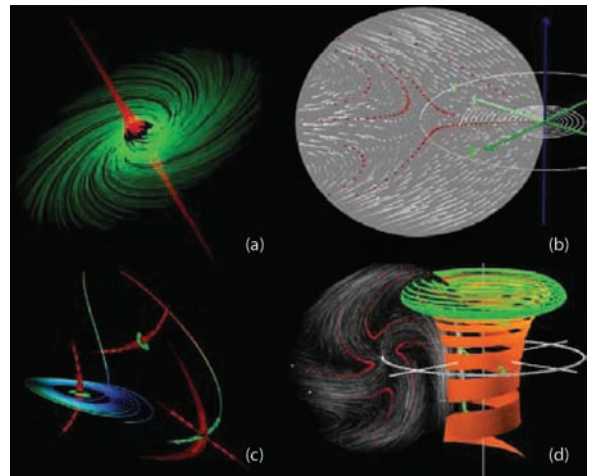


Figure 7: Enhanced topology visualization combining (a) streamline-based glyphs, (b) direct flow visualization, (c) solution trajectories and (d) streambands [HG00].

3.2. Rendering stage techniques

There is a number of flow visualization methods that render multi-valued flow data. Splatting is a very versatile technique that allows the integration of vector fields into scalar rendering by adding tiny vector particles into the splat texture [CM93]. The examples by Max et al. [MCW93] combine surface geometries representing cloudiness with coloured glyphs representing wind velocity. This is an example where a single rendering technique shows different types of data and still uses appropriate visualizations for the components. In a data type oriented manner the ground is rendered as a surface, while the clouds have a volumetric look giving a good feeling of orientation in space. Directions and altitude are visualized as coloured glyphs, showing that they do not represent physical objects in space (see Figure 8).

Treinish investigated how specialized visualizations can be used to effectively visualize weather data using views of varying complexity [Tre99] and presented a multi-resolution technique for complex weather data [Tre00] (see Figure 9).

Because many of the existing flow algorithms are derived from physical models based on particles, the combination of particle and texture based flow visualization is a natural approach. Erlebacher et al. [EJW05] developed a spatio-temporal framework that encompasses many aspects of time-dependent flow visualization. Weiskopf et al. [WSEE05] apply the spatio-temporal framework to unsteady flow visualization. In the context of dense flow visualization they identify two important types of coherence: spatial coherence that conveys the structure of a vector field within a single picture and frame-to-frame coherence that conveys the development of structures over time. They employ two

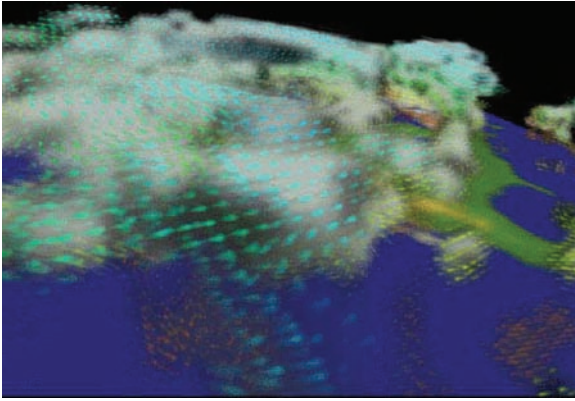


Figure 8: DVR combining both realistic cloud rendering and splatted directional glyphs and colour coding [CM93] (Image courtesy of R. A. Crawfis).

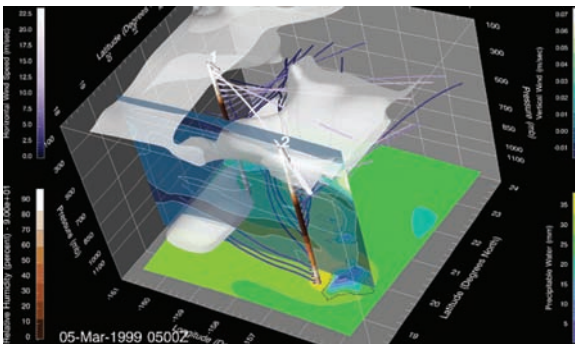


Figure 9: A weather visualization combining streamribbons, arrows, slices, contour bands and isosurfaces [Tre99] (Image courtesy of L. Treinish).

steps: the first step is basically a propagation of particles forward in time to construct a space-time volume of trajectories. The second step applies convolution along paths through the space-time volume that is done independently for each time step and texel. This hybrid particle and texture based approach combines advantages of particle-based representations with texture-based visualization. Particle systems are computational and memory efficient and allow accurate Lagrangian integration. Texture-based systems on the other hand have hardware acceleration for texture lookups and manipulations supported on modern graphic cards.

An approach that achieved not too much attention in the literature is to use more than one rendering system at the same time. Yagel et al. [YESK95] suggested the use of four different renderers on a single CFD data set. Each is specialized to a specific task. Interactions can be visualized using a fast hardware-accelerated algorithm, high magnification images employ a specialized anti-aliasing technique.

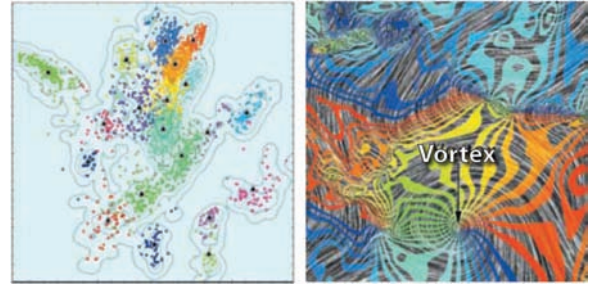


Figure 10: A layered combination of glyphs, colour coding and isolines (left) and a filigreed layered visualization of flow data combining texture advection and colour coding [WFK*02] (Image courtesy of P. C. Wong).

They use a ray casting algorithm specialized for the design of transfer functions while high-resolution and high-quality images are produced using a volumetric technique. Because today's computing machinery makes interactive manipulation of transfer functions, lighting parameters and other rendering attributes possible, the advantages of multiple combined renderers may be less obvious. Nevertheless the combination of different rendering approaches and smooth transition between these can improve the visual experience for the user. This is an open research problem. Also, the integration of multiple renderers (e.g. illustrative and volumetric) into a single image at the same time is not investigated in much detail today. New ways to integrate different rendering algorithms is a promising route for future research. Steinberg et al. [SMK05] present an application that uses several renderers for prototyping, comparison and educational applications.

3.3. Image stage techniques

Crawfis and Allison [CA91] very early recognized the power of compositing several images of rendered objects together to do scientific visualization. Their graphic synthesizer could combine several images to generate multi-variate representations of 2D data sets. Wong et al. [WFK*02] apply image compositing for visualization of multi-variate climate data. They present three image fusion techniques: opacity adjustments for see-through, filigreed graphics where portions of each layer are removed and elevation mapping where one scalar is mapped to the z -axis. In Figure 10, we see an example of layered glyph rendering (left) and a filigreed layering of colour coded rendering and advection layered flow visualization (right).

Kirby [KKL04] gives an introduction to art-based layered visualization in the context of 2D flow visualization (see Figure 11). A promising approach to visualize multiple aspects of high-dimensional data sets is the combination of art- and glyph-based rendering. They introduce the concept of layering in a similar way as done in oil-paintings:

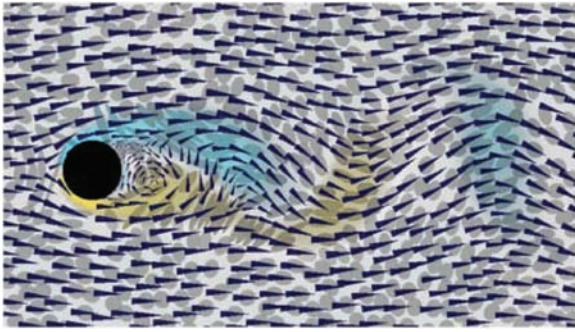


Figure 11: An visualization using multiple layers to visualize scalar, vectorial and tensorial information [KML99] (Image courtesy of M. Kirby).

underpainting contains a low-frequency and low colour-range colouring of a 1D scalar vorticity value. Then two data layers follow: ellipses and arrows, depicting the most important aspects of the data. A final mask layer gives context information to the obstacle in the simulation data set. By carefully selecting the order of layers it is possible to weight different aspects of the data differently and it can suggest a viewing order for different parts of an image. Sobel [Sob03] presents a descriptive language for modelling layered visualizations, to design and share visualization parameters for layering algorithms.

Common problems when dealing with multi-variate vectorial data sets include:

1. Interpolation, gradient estimation and feature extraction: Often a large amount of background information from CFD or mechanical engineering is necessary to select the appropriate technique for a given data set.
2. Unstructured grid handling: There can be large differences in the sizes of cells, degenerated cells and cells with non-planar boundaries. This can make point search, neighbour traversal and intersections numerically difficult and also complicates implementation of novel algorithms.
3. Complex data storage formats: There is a large number of data storage formats, which tend to be complex including subdivided regions, overlapping or moving meshes and the format definitions are also subject to changes and extensions.

4. Tensor Field Visualization

In this section, we outline multi-method or combinable visualization techniques for multi-variate tensorial data, of course the cited works are by no means complete and not all-important works could be included.

Visualization of multi-variate data containing only tensor information is a difficult problem already. The interpretation

of tensor information suffers if it is reduced to scalar information or if parts are visualized separately (e.g. in different images). Tensorial information has to be visualized fully or meaning and comprehensibility can be lost. When speaking of tensor field visualization we typically refer to second-order tensors (three by three matrices). Depending on the application these tensors can be symmetric or non-symmetric. From a symmetric tensor we can derive three orthonormal eigenvectors and corresponding eigenvalues. Non-symmetric tensor fields can be decomposed into a symmetric tensor and a vector field. Because of these properties most visualization applications focus on the visualization of symmetric tensor data—this already involves six variables at each point simultaneously. Because tensorial information is difficult to comprehend and structure, multi-style visualization techniques are common in this field. An example would be a layered visualization combining diffusion tensor glyphs and a CT reference image slice to show the organ geometry. It is also common to show basic geometry cues (e.g. the shape of the brain or the kidney) as context information in the form of a wire frame or silhouette rendering.

Important sources for tensor data are

- Medical applications working with measured MRI diffusion tensors. Their visualization is the field of Diffusion Tensor Imaging (DTI) and deals with symmetric tensors with positive eigenvalues.
- Materials science and geomechanics working with stress and strain tensor fields. Related tensors are symmetric with signed eigenvectors.
- Fluid dynamics where several properties are tensor valued. Examples are the vorticity tensor and the fluid momentum gradient tensor.
- General relativity theory simulations, where gravity is expressed as a rank two tensor and the electro-magnetic field tensor in special relativity.

Vilanova et al. [VZKL05] give an extensive introduction and a state of the art overview of diffusion tensor visualization. Wünsche [Wün99] gives a basic introduction into stress and strain tensor fields suitable for the computer scientist.

4.1. Techniques in the processing, filtering and visualization mapping stage

For tensor fields glyph-based visualization is the most common technique. Glyphs of stresses and strains is surveyed by Hashash et al. [HYW03]. One basic question that many publications state is ‘visualize all the information of one tensor in some places or only some part of it everywhere?’. The first would lead to some kind of glyph-based visualization where the information is visualized using a glyph that can represent all the degrees of freedom. Glyph-based visualization of tensor fields mainly uses the three eigenvectors (major, medium

and minor) to generate a shape showing the direction of the eigenvectors. The most common is the ellipsoid because it is possible to include all eigenvectors in a straightforward manner. Other glyphs are for example the Haber Glyph and the Reynolds Glyph [HYW03].

A classification of tensor shapes was given by Westin [WMM*02]. A diffusion tensor is isotropic when the eigenvalues are about equal ($\lambda_1 \approx \lambda_2 \approx \lambda_3$), planar anisotropic where two eigenvalues are about the same and larger than the third ($\lambda_1 \approx \lambda_2 \geq \lambda_3$) or linear anisotropic where one eigenvalue is larger than the others ($\lambda_1 \geq \lambda_2 \approx \lambda_3$). The corresponding ellipsoids are spherical, disk- or needle-shaped, respectively. Westin introduced the shape factors to measure which of these cases is dominant:

$$c_{linear} = \frac{\lambda_1 - \lambda_2}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}} \quad c_{planar} = \frac{2(\lambda_2 - \lambda_3)}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}$$

$$c_{spherical} = \frac{3\lambda_3}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}.$$

The three shape factors sum to one and define barycentric coordinates, that can be used for glyph geometry assignment [WMM*02], opacity mapping [KWH00], colour coding, or glyph culling [ZKL04].

One way to get from glyphs to hyperstreamlines [DH93] is to place ellipsoids close to another along the direction of the major eigenvector. From any seed point, three hyperstreamlines can be generated using one of the three eigenvector fields for the streamlines and the other two for the cross section. This leads to a connected line along the major direction that encodes the other two eigenvalues in the cross section of the streamline. A hyperstreamline visualizing a tensor field can be enhanced to show other properties by colouring. For non-symmetric tensors, the rotational components can be encoded as ‘wings’ along the main hyperstreamlines [ZP03].

Hyperstreamlines tend to overrepresent one eigendirection of the tensorfield. In regions where the one eigenvalue is clearly distinguished from the others it can be difficult to compute good hyperstreamlines. A very convincing hybrid approach is to place the glyph such that it becomes possible to follow the lines without actually connecting the glyphs. A placement algorithm for this approach was presented by Kindlmann et al. [KW06] (see also Figure 12).

Zhang et al. [ZKL04] use stream tubes and stream surfaces for the tensor aspect of the data and contours and volumes as anatomical landmarks. The authors do not use the different visualization techniques to visualize different components of the data but show data ranges differently. Regions of the brain of high linear anisotropy very often correlate with regions densely containing fibre tracks. Therefore tensors having high linear anisotropy are adequately visualized using stream tubes, while tensors of high planar anisotropy are visualized using stream surfaces. This way both techniques can be used for the type of data they work best for.

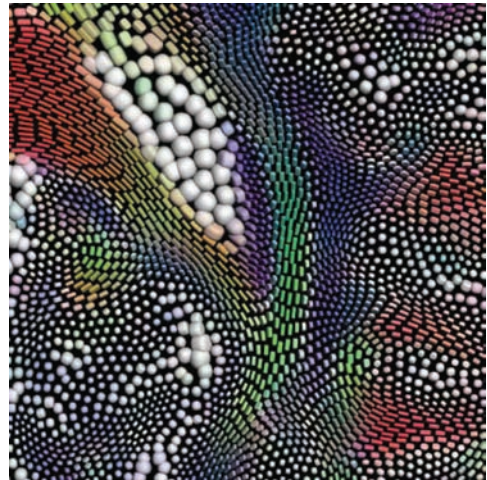


Figure 12: A sensible layout of glyphs can convey the location of fibre structures in the data. This shows good glyph placement can improve the visualization [KW06] (Image courtesy of G. Kindlmann).

In an adaption of image-based flow visualization Zhang et al. [ZHT07] visualize topological properties of tensor fields on curved surfaces. They discuss properties of critical points and an approach to extract flow directions to apply advection vectors. Furthermore they show applications to painterly rendering of 2D images.

Merhof et al. [MSE*06] present a hybrid rendering technique combining point sprites and triangle strips to display fibre tracts in the brain. They show that combining two rendering techniques can improve the comprehensibility of the visualization. This is an example how thinking about the ‘what’ part of the visualization (fibre tracts) can give clues to improving the ‘how’ approach.

4.2. Rendering stage techniques

Visualizing parts of the tensor information in a continuous fashion is done in volume rendering of tensor fields. Sigfridsson et al. [SEHW02] present a dense technique that filters noise along the eigenvector directions to get a continuous representation of tensor fields that produce results similar to line integral convolution. The basic tasks in volume rendering tensor fields—determining opacity, calculating shading and assigning material colour—can be done by specific mappings of tensor properties based on the shape factors. Opacity is determined by using a barycentric opacity map (e.g. high opacity for linear anisotropy). Lighting is determined by using a heuristic that refers to the shape an ellipsoid glyph would have in the same position: in case of planar anisotropy the lighting model is the same as with traditional surface modelling, in the linear case the lighting model is similar to lighting of illuminated streamlines. Cases in between are



Figure 13: (a) The reaction diffusion texture allows natural glyph placement and geometry. (b) Alternative combined use of the texture for colouring the volume [KWH00] (Image courtesy of G. Kindlmann).

interpolated. In the simplest setting, colour coding is done by using a coloured ball and choosing colour depending on the direction of the major eigenvector. This basic setting allows improvement using additional visualization techniques. Kindlmann et al. [KWH00] present a reaction-diffusion texture that can visualize a tensor field alone but also integrate it with the volume-rendered tensor-field visualization (see Figure 13). The idea of a reaction-diffusion texture is to simulate a system of two differential equations. One describes the diffusion governed by Fick's second law of two morphogens where the resulting concentration of these morphogenes determines the colour at each position. The other differential equation measures how much the two substances react and neutralize each other. The initial condition is that both have the same concentration everywhere. Applying diffusion relative to the given tensor field at each position generates a texture that can show information about the tensor field in its own right. The authors suggest colour modulation or bump mapping to combine volume rendering and the volumetric texture. The result is similar to a surface rendering of geometry combined with diffusion glyphs, but has several advantages. The most important is that the resulting ellipsoids are distributed more naturally and are packed in a way that represents features of the data. Also the empty space between tensor ellipsoids is reduced. Furthermore it avoids the common problem of gridded ellipsoid layout to give the false impression of structure in the data.

4.3. Image stage techniques

For tensor visualization an image stage technique has been published by Laidlaw et al. [LAK*98]. They show that a lot of information can be conveyed in a single image using brush based glyphs and layering. By combining varying brush strokes and layering it is possible to display many components of the data locally, while the underpainting can show form. Contrast is used to create depth. Stroke sizes, texture and contrast help to define a focus within each image (see Figure 14). In a recent publication Wenger et al. [WKZ04] combine volume rendering techniques and layering into a multilayer volume rendering approach. Their method is re-

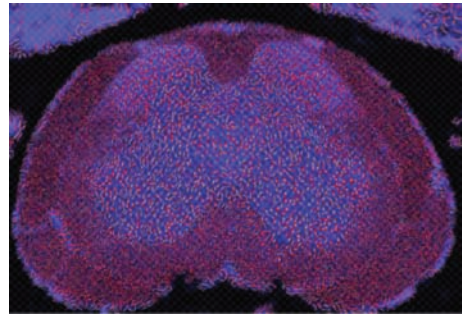


Figure 14: A visualization of the mouse spinal cord based on artistic techniques using multiple layers and glyphs [LAK*98] (Image courtesy of D. H. Laidlaw).

lated to two level volume rendering [HMBG00] which will be discussed in the second part of this paper. They combine densely packed threads of coloured and haloed streamlines with direct volume rendered context information. To deal with occlusion problems they designed interactive controls to change visualization parameters like thread length or opacity. Also, they heavily use transfer functions. This interesting publication is a good example of how to layer volumetric rendering successfully to visualize different portions of information.

Common problems when dealing with multi-variate tensorial data sets include:

1. Eigenvectors: One question for tensorial data is how to display all eigenvectors of a tensor without over-representation of the largest eigenvector.
2. Fuzzyness: In regions where the eigenvectors of the tensor have almost the same size standard approaches can be difficult to implement due to fluctuations of the direction of the largest eigenvector.
3. Noise: The data quality of measured tensors has not yet reached the detail and exactness of other imaging techniques.

5. General Approaches to Multi-Dimensional Visualization

In this section, we will give an overview of the techniques we have identified to deal with complex data sets, of course the cited works are by no means complete and not all-important works could be included.

It has become a widely accepted perspective to view visualization as a path from data to understanding [DKS*05]. We have identified a wide and diverse range of general approaches to multi-variate or complex data visualization. The following subsection cannot give a comprehensive

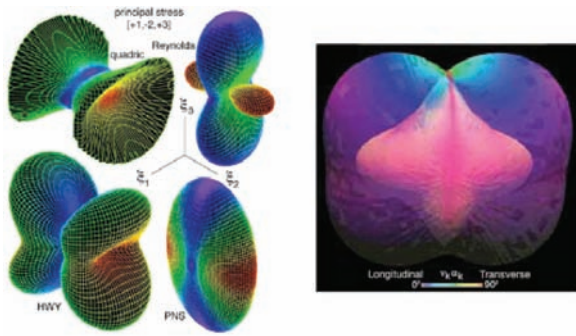


Figure 15: Advanced glyphs for stress tensor visualization using colour and geometry for information coding (left) and transparency (right) [KGM95] (Image courtesy of R.D. Kriz).

enumeration of all the related work, but is thought to be an introductory overview. We also do not distinguish between ‘how’ and ‘what’ approaches because several of the techniques can be used both ways.

Derivations or derived quantities are used because visualizing the measured data directly might not be useful for understanding it. Kirby et al. [KML99] show that in flow visualization additional components help understanding the situation, even if they are mathematically redundant. In flow visualization useful derived quantities are for example vorticity, the rate-of-strain tensor, the rate-of-rotation tensor, turbulent charge and turbulent current. Smoothing the data to remove noise or calculating gradients to improve lighting will very often result in more pleasing visualizations that are easier to work with. Jänicke et al. [JWSK07] present a derived quantity that measures statistical complexity. This is an interesting hint that the combination of information theory and visualization can be an important research direction in the future. Hauser [Hau06] discusses the use of differential information to improve scientific visualization. Conceptually derivations belong to the data processing stage in Figure 1.

Glyphs (also referred to as icons) are a powerful communication item. A large number of data dimensions can be incorporated into the attributes of a single shape or symbol (see Figure 15). The particular mappings may also be customized to reflect semantics relevant to specific domains to facilitate interpretation. Because glyphs are generally not placed in dense packings, the free space between them allows the visualization of additional information. They interact therefore well with other visualization algorithms and are frequently added to visualization applications. Wittenbrink et al. [WPL96] suggest glyphs for uncertainty in vector fields. Kindlmann and Westin [KW06] have presented a technique for packing glyphs in a way that their alignment conveys additional information. Hashash gives an overview of stress and strain tensor glyphs [HYW03]. Glyphs can be classified

to belong to the data processing stage in Figure 1. (See also for example [Kin04, War02, KGM95, WL93].)

Hybrid/multi-method visualization is the application of several visualization techniques for the same image. This is useful especially for segmented data sets where background information is applicable to choose the appropriate rendering technique for different subregions of the data [HMBG00]. There are many examples for this approach: Jesse and Isenberg [JI03] describe a hybrid rendering scheme that combines photorealistic and illustrative rendering to highlight parts of a volume for presentation. Kreeger and Kaufmann [KK99] describe a fast method to combine volume rendering and translucent polygons to render mixed scenes. Laramee et al. [LJH03, LvWJH04] and van Wijk [Wij03] present rendering algorithms to visualize flow on surfaces. Wegenkittl et al. [WGP97] combine surfaces, tubes and particles to visualize the behaviour of a dynamical system. Multi-method visualization means to use the information from a previous step in the abstract pipeline (Figure 1) to select between or combine multiple visualization techniques.

Interaction is probably the most important tool for understanding complex data. Interactions modify viewing parameters, transfer function manipulation, seeding point selection, culling, queries, graphical model exploration, region of interest selection and many others. An emerging trend is to use concepts from interactive visual analysis for data exploration. In Figure 16, we see an example of multiple linked views that work together to help understand the data. In the attribute view (c) and (d) linking is used to display how different attributes are related: the data elements selected by the brush are shown in red, while the elements selected in the other view are coloured yellow. Interaction can set parameters for the different stages, but in most cases it can be classified into the visualization mapping stage in Figure 1.

Layering and Fusion has been used extensively in scientific visualization to show multiple items. Fusion-based methods combine different rendering styles in image space [WFK*02]. Layering is a generalization of this approach where multiple layers of information are visualized on top of each other. This is most applicable for 2D visualization but there is work where transparent stroked textures show surfaces without completely obscuring what is behind them [IFP97, Int97] (see Figure 17 left). Several other layering techniques have been discussed in the first section of this paper, see [CA91, LAK*98, WFK*02, KKL04]. Layering and fusion are commonly part of the image stage in Figure 1.

Two-level volume rendering Hauser et al. [HMBG00] and Hadwiger et al. [HBH03] present a two-level approach that combines different rendering methods for volume rendering of segmented data sets (see Figure 18). Each of the segmented regions can be rendered using a specific rendering method like non-photorealistic rendering (NPR), direct volume rendering (DVR) or maximum intensity projection (MIP) during

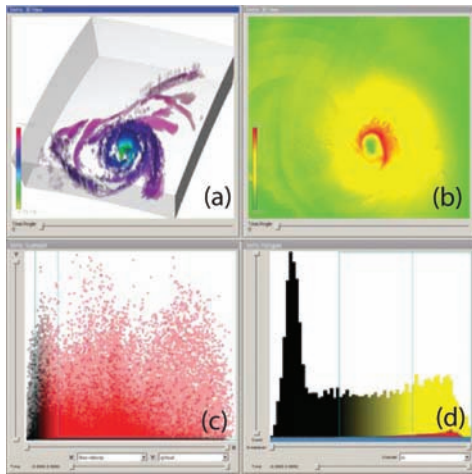


Figure 16: An example of combined attribute and volumetric views. The 3D view (a) shows the location of data points in space with pressure mapped to colour. A 2D slice (b) shows the velocity close to the eye of the storm. Two attribute views [scatterplot of velocity vs. cloud density (c) and a histogram of temperature (d)] are used to select which cells are shown.

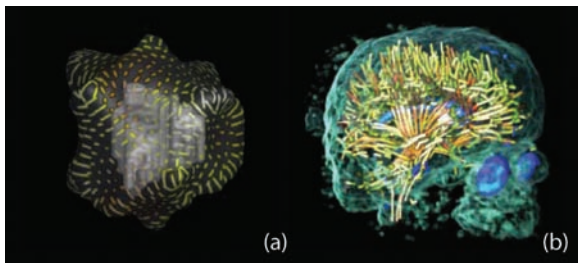


Figure 17: (a) Transparent surfaces can allow layered visualization for 3D images [IG97] (Image courtesy of V. Interrante) (b) Non-photorealistic rendering of tensor information using line-glyphs and DVR of context information [WKZ04] (Image courtesy of A. Wenger).

ray accumulation. Because most users perceive 3D scientific data sets as built up from individual objects, the authors use the segmentation information to generate images that take this into account. To compute the representative values for the objects different rendering techniques can be used. The authors also use the technique to visualize dynamical systems. This gives a hint at a more general applicability of their approach. Because the decision of what rendering method to choose is given to the user, it becomes possible to use the most adequate in the given moment. This approach is well suited to visualize multi-dimensional data sets by combining different rendering methods that are most appropriate for different features inside the data (see also [WKZ04] and

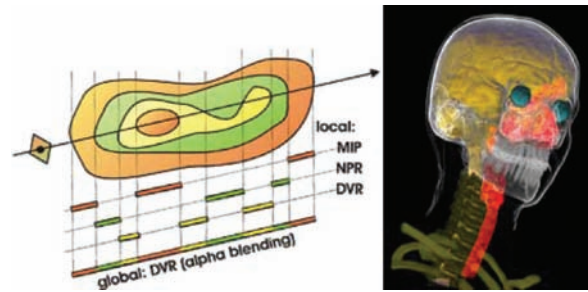


Figure 18: Two-level volume rendering can combine multiple rendering techniques using different compositing methods locally (left). A multi-level volume rendering of a human head using tone shading (brain), contour enhancement (skin), shaded DVR (eyes and spine), unshaded DVR (skull, teeth and vertebrae) and MIP (trachea) [HBH03] (Image courtesy of M. Hadwiger).

Figure 17). Two-level volume rendering belongs to the rendering stage in Figure 1.

Machine Learning can help to generate meaningful classifications for a multi-variate data items with only few user-specified examples. Ferré et al. [FPT06] for example discuss a specialized decision tree technique to classify multi-modal data sets to materials. Further examples where machine learning is brought to use for visualization purposes are discussed by Ma [Ma07]. These approaches allow the user to select samples of relevance from which a machine learning algorithm (e.g. neuronal network or support vector machine) builds a classifier which then segments the rest of the data (see also [TLM05, TM05]). Machine learning tends to be used in conceptually as a preprocessing step such that the computed classification is used as input to the rendering algorithm (see Figure 1).

Multiple Views present the information in several different views that encourage comparison, give contrast and help to generate a correct understanding. Roberts [Rob00] describes the generation and presentation of multi-form visualizations in an abstract way and gives an introduction to multi-view visualization. Yagel et al. [YESK95] discuss grouping volume renderers that have different quality and rendering speed trade offs. Van Wijk and van Liere's hyperslicing approach uses multiple views to display a large set of possible projections of the data [WL93]. Multi-view approaches can be classified into the image stage in Figure 1.

n-D viewing is based on defining hyperplanes on the high-dimensional volume or direct projection. This is done very often for time-varying data sets, where time-coherency can be exploited for compression and acceleration. The major issue for projections from n-D is to determine occlusion because a front-to-back ordering is not uniquely defined after projecting more than one dimension down. Feiner and Beshers [FB90]

suggest the World within World approach to drill down on the data by iteratively slicing away dimensions (see also [WWS03, NM02, BPRS98, WL93] and references therein). Blaas et al. [BBP07] have developed a framework that uses interactive projection parameter specification for mapping multi-variate data values to scatterplots. We can think of n-D viewing as generalized projection methods during the rendering stage in Figure 1.

Probing is a general visualization approach for multi-variate data visualization. The user can state interest in a specific location or data range. Then a reduced amount of data is shown everywhere and for subsets of the data a local and more complex visualization conveys details. This avoids clutter and occlusion, is computationally efficient and helps the user to focus on specific aspects of the data. Examples for local-detail, global-overview techniques are: focus and context visualization [DGH03], magic lenses [Kea99], level-of-detail [CMS98], clipping-probes [WEE03] or zooming. Probing belongs to the visualization mapping stage of visualization (see Figure 1).

Reduction of dimension and de-noising can remove unwanted details in the data and remove obscuring structures that hinder the process of understanding. Also, presenting views that contain a reduced amount of information and clipping are examples for data reduction tools. The importance of data reduction is very well expressed in the saying that in the future the main question will not be what to show, but what not to show. There is a trend to include attribute views (such as scatterplots, parallel sets, etc.) for interactive visual analysis of the attributes of the data-set. These views can benefit strongly from having strong clustering, reduction and projection algorithms available [Won99]. Reduction normally happens during data processing or visualization mapping (see Figure 1).

6. Illustrative Rendering

In this section, we will give an overview of the techniques we have identified to visualize complex data illustratively, of course the cited works are by no means complete and not all important works could be included. The visualization of multiple features and the accentuation of important structures and information has gained special attention in scalar volume rendering, especially in illustrative rendering. Illustrative rendering employs abstraction techniques to convey relevant information. In the context of scientific visualization illustrative rendering refers to the adaption of techniques that have been developed by artists for the generation of synthetic imagery. In the context of scientific visualization Stoppel et al. [SLM02] explore the use of illustrative techniques to visualize multi-dimensional, multi-variate data sets (see also Figure 19). Bruckner et al. [BG05] have developed the VolumeShop framework for direct volume illustration.

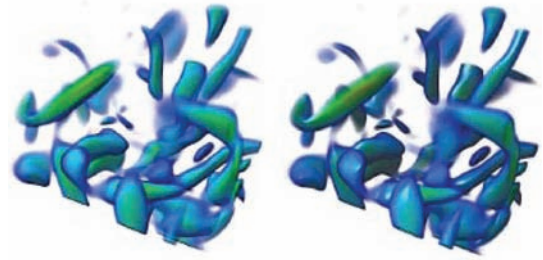


Figure 19: Visualizing flow data can benefit from using illustrative techniques. The image shows a closeup of turbulent vortex flow using silhouette and shading (left) and additionally gradient and depth enhancement (right) [SLM02] (Image courtesy of K.-L. Ma).

Depth colour Cues Svakhine and Ebert [SE03] describe depth based colour variation. It gives intuitively understandable cues of the relative positions of different features in the data set. Distance colour blending dims sample colours as they recede from the viewer. At the front of the volume, the voxel colour remains unchanged. As the screen-depth value increases the colour is gradually blended with the background colour:

$$\text{Colour} = (1 - \text{depth}) \cdot \text{Colour}_{\text{original}} + \text{depth} \cdot \text{Colour}_{\text{background}}$$

Silhouette enhancement and boundary enhancement

Silhouette lines are particularly important in the perception of surface shape and the perception of volumetric features. In order to strengthen the cues provided by silhouettes, one increases the opacity of volume samples where the gradient is close to perpendicular to the view direction. Using a dark silhouette colour can be effective for outlining of features. Levoy [Lev88] proposed to scale opacity using the magnitude of the local gradient. Many applications also use the local gradient as a transfer function parameter [KKH01]. Ebert and Rheingans [ER00] suggest to add scaling parameters to boundary enhancement such that the gradient-based opacity o_g of the volume sample becomes

$$o_g = o_v \left(k_{gc} + k_{gs} (|\nabla_f|)_{ge}^{k_{ge}} \right)$$

depending on the data o_v (original opacity), ∇_f (the gradient of the volume at the sample) and on user specified parameters k_{gc} (scales influence of original opacity), k_{gs} (scales influence of gradient enhancement) and the exponent parameter k_{ge} that allows the user to adjust the slope of the opacity curve.

Enhanced transparent surfaces and stippling Because many features can be visualized using surfaces, transparent surface rendering offers a good possibility to show the spatial relationship between two superimposed features. To improve

shape recognition, Interrante [Int97] uses principal direction-driven LIC [IG97]. The surface shape and curvature are enhanced using a stroke texture. This enables the visualization of additional variables using coloured strokes and varying stroke width and density. The surface information is stored in a texture and needs recomputation if stroke parameters are changed. Nagy et al. [NSW02] have developed a technique to do stippling of surfaces in real time using preprocessed principal directions. This allows to change stroke positions and appearance interactively but it needs an additional pass to render the strokes.

Feature halos In line drawings it is common to depict the depth ordering of two crossing lines by inserting a short gap into the line behind the other such that the lines do not touch. Therefore halos are an important technique for visualization of streamlines. For scalar volumes Ebert and Rheingans [ER00] propose a method for creating halo effects for features during the illumination process. Halos are created primarily in planes orthogonal to the view vector by making regions just outside features darker and more opaque. They calculate the halo-intensity by weighting the neighbour samples gradient magnitude and halo-direction. Bruckner et al. [BG07] present an approach using an interactively defined halo transfer function to classify structures of interest based on data value, direction and position.

Shadows and indirect illumination Even though a feature in the volume (e.g. a vortex) may not cast a shadow in reality, a shadow can very well show spatial relationships between features in the data. Especially for glyphs and disconnected features, lighting is a powerful way to provide better perceptual cues [GP06].

Cutaways and Ghosting Feiner and Seligmann [FS92] introduced cut-away views and ghosting to 3D visualization. Cut-away views are a common tool of illustrators to show important structures in a volume that would be occluded otherwise. Viola et al. [VKG05] introduced importance-driven volume rendering to determine which parts of the volume are to be cut away (this notion of importance is also referred to as degree-of-interest (DOI) in other publications). Ghosting [GBKG04, BGKG05] is a technique very often combined with cutaway views (see Figure 20). Instead of removing the obscuring region completely they keep the most important parts (e.g. edges) of the occluding structures to conserve context information.

Motion blur is a widely used technique to show information from time-dependent data sets. For example Neophytou and Mueller [NM02] use a motion blurring effect to give hints on how the data will evolve.

7. Conclusion

We have discussed new developments in visualization research and have outlined recent trends in the literature. Based on these trends we have presented an abstracted goal

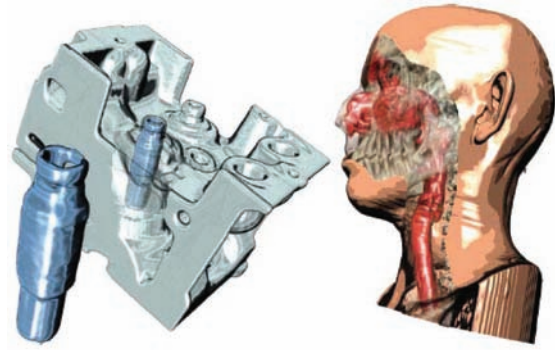


Figure 20: Techniques from traditional illustrative imaging can increase the information content in volume rendered images. Ghosting can be used to show important features in detail while cutaway views keep structural information independent of the viewing direction [BGKG05] (Image courtesy of S. Bruckner).

for multi-field visualization algorithms that highlights open questions on different stages of the visualization pipeline:

- How can we capture all the relevant information?
- How can we combine relevant quantities of information belonging to the same location in space?
- What quantities can be derived from the data, that will help the user to understand the data?
- What are the features in the data and how can we combine automated and interactive techniques for finding them?
- Which visualization technique is most appropriate for the given data?
- How can we combine different visualization techniques for efficient and effective visualization?

In the first part of this paper, we have discussed applications and algorithms employing one or several useful techniques to visualize complex scientific data sets in two and three dimensions. In the second part we have given an overview of modern approaches for multi-variate data visualization and analysis. We believe the discussed work provides valuable insight, but is by no means complete and much important work was not included in the discussion.

Conclusively, in the literature we found researchers taking three general approaches to deal with complex, high-dimensional data sets. These approaches are highly promising and research in how these can be combined and improved will increase the value of visualization systems:

- *Interaction:* The connection between elements of the data can be explored interactively. This approach assumes that

the user will understand the configuration of a data set using different views and different levels of detail interacting with visualization parameters. Exploration of correspondences between different views on the data by visual comparison and interactive drill-down are important aspects of current analysis tools. Even though we do not know what ‘understanding’ really is, there is an agreement that the visual system in many scenarios allows to see what the data means. This is an important key to understand complex multi-dimensional data and is used extensively in related fields like information visualization as well.

- *Complexity-adjusted rendering*: Due to the growth in computation power, a valid approach to deal with the rising complexity is to increase the computational efforts put into visualizing this data. Rendering complex visual cues (e.g. lighting models, shadows and distance cues) can make spatial interpretation easier for the user. We have seen that visualization research has produced tools to decrease clutter and to visualize complex and intertwined 3D data features.
- *Multi-method visualization*: Is the approach to use multiple or hybrid visualization techniques and each only for the range or type of data it works best for. In the scope of this article we have discussed several publications following this approach and we believe a tight integration of multiple techniques in a visualization gives a key advantage towards understanding the data.
- *Automated techniques*: These provide powerful tools when working with complex multi-variate data sets. Two important types of automated tools are feature classification algorithms and automated optimizations of visualization parameters. Classification, feature extraction and segmentation information are useful to render data in a more appropriate way and to ease the interaction with the data. Also, they may help to extract information from the data. Automated parameter optimizations (e.g. for seed point placement, colour selection or viewpoint parameters) can reduce clutter, speed up rendering and help to extract meaning from the data.

An alternative title for this report was ‘Multi, Multi, Multi!’—this describes the key message we want to transport: to deal with the new research challenges we need to visualize multi-variate data using multi-method visualizations combining multiple views, multiple scales and multiple data representations to show all aspects of the data.

Acknowledgements

We thank Meister E. Gröller for his input and support of this work. We thank the reviewers for their helpful comments. This work has been funded the FWF PVG project supported by the Austrian Science Fund (FWF) under grant no. P18547-

N04 and the SimCT project (FFG) under grant no. 812136-SCK/KUG.

References

- [BBP07] BLAAS J., BOTHA C. P., POST F. H.: Interactive visualization of multi-field medical data using linked physical and feature-space views. In *Proceedings of EuroVis '07* (2007), pp. 123–130.
- [BG05] BRUCKNER S., GRÖLLER M. E.: Volumeshop: An interactive system for direct volume illustration. In *Proceedings of IEEE Visualization '05* (2005), pp. 671–678.
- [BG07] BRUCKNER S., GRÖLLER E.: Enhancing depth-perception with flexible volumetric halos. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1344–1351.
- [BGKG05] BRUCKNER S., GRIMM S., KANITSAR A., GRÖLLER M. E.: Illustrative context-preserving volume rendering. In *Proceedings of the 7th Joint IEEE VGTC—EUROGRAPHICS Symposium on Visualization (VisSym '05)* (2005), pp. 69–76.
- [BPRS98] BAJAJ C. L., PASCUCCI V., RABBIOLO G., SCHIKORC D.: Hypervolume visualization: a challenge in simplicity. In *VVS '98: Proc. of the 1998 IEEE symposium on Volume visualization* (1998), pp. 95–102.
- [CA91] CRAWFIS R. A., ALLISON M. J.: A scientific visualization synthesizer. In *Proceedings IEEE Visualization '91* (1991), pp. 262–267.
- [CDH05] CHARLES D. HANSEN, JOHNSON C. R.: *The Visualization Handbook*. Elsevier, 2005.
- [CM93] CRAWFIS R. A., MAX N.: Texture Splats for 3D Scalar and Vector Field Visualization. In *Proceedings IEEE Visualization '93* (1993), pp. 261–267.
- [CMS98] CIGNONI P., MONTANI C., SCOPIGNO R.: A comparison of mesh simplification algorithms. *Computers & Graphics* 22, 1 (1998), 37–54.
- [CS99] CAI W., SAKAS G.: Data intermixing and multi-volume rendering. *Comput. Graph. Forum* 18, 3 (1999), 359–368.
- [DGH03] DOLEISCH H., GASSER M., HAUSER H.: Interactive feature specification for focus + context visualization of complex simulation data. In *Proceedings of the 5th Joint IEEE VGTC—EUROGRAPHICS Symposium on Visualization (VisSym '03)* (2003), pp. 239–248.
- [DH93] DELMARCELLE T., HESSELINK L.: Visualizing second-order tensor fields with hyperstreamlines. *IEEE*

- Computer Graphics and Applications* 13, 4 (1993), 25–33.
- [DKLP02] DJURCILOVA S., KIMA K., LERMUSIAUXB P., PANG A.: Visualizing scalar volumetric data with uncertainty. *Computers and Graphics* 26, (2002), 239–248.
- [DKS*05] DENNIS B., KOCHERLAKOTA S., SAWANT A., TATEOSIAN L., HEALEY C. G.: Designing a visualization framework for multidimensional data. *IEEE Computer Graphics and Applications* 25, 6 (2005), 10–15.
- [DMG*05] DOLEISCH H., MAYER M., GASSER M., PRIESCHING P., HAUSER H.: Interactive feature specification for simulation data on time-varying grids. In *Proceedings of the Conference Simulation and Visualization (SimVis '05)* (2005), pp. 291–304.
- [EHK*06] ENGEL K., HADWIGER M., KNISS J. M., REZK-SALAMA C., WEISKOPF D.: *Real-Time Volume Graphics*. A K Peters Ltd., 2006.
- [EJW05] ERLEBACHER G., JOBARD B., WEISKOPF D.: Flow Textures: High-Resolution Flow Visualization. In *The Visualization Handbook*. Elsevier, 2005, pp. 279–293.
- [EMRY02] EBERT D. S., MORRIS C. J., RHEINGANS P., YOO T. S.: Designing effective transfer functions for volume rendering from photographic volumes. *IEEE Transactions on Visualization and Computer Graphics* 8, 2 (2002), 183–197.
- [ER00] EBERT D., RHEINGANS P.: Volume illustration: non-photorealistic rendering of volume models. In *Proceedings IEEE Visualization 2000* (Los Alamitos, CA, USA, 2000), IEEE Computer Society Press, pp. 195–202.
- [FB90] FEINER S. K., BESHES C.: Worlds within worlds: metaphors for exploring n -dimensional virtual worlds. In *UIST '90: Proceedings of the Third Annual ACM SIGGRAPH Symposium on User Interface Software and Technology* (1990), pp. 76–83.
- [FPT04] FERRÉ M., PUIG A., TOST D.: A framework for fusion methods and rendering techniques of multimodal volume data. *Computer Animation and Virtual Worlds* 15, (2004), 63–77.
- [FPT06] FERRÉ M., PUIG A., TOST D.: Decision trees for accelerating unimodal, hybrid and multimodal rendering models. *The Visual Computer* 22, 3 (2006), 158–167.
- [FS92] FEINER S., SELIGMANN D. D.: Cutaways and ghosting: satisfying visibility constraints in dynamic 3d illustrations. *The Visual Computer* 8, 5–6 (1992), 292–302.
- [GBKG04] GRIMM S., BRUCKNER S., KANITSAR A., GRÖLLER M. E.: *V-Objects: Flexible Direct Multi-Volume Rendering in Interactive Scenes*. Tech. Rep. TR-186-2-04-06, Institute of Computer Graphics and Algorithms, Vienna University of Technology, 2004.
- [GGSC98] GOOCH A., GOOCH B., SHIRLEY P., COHEN E.: A non-photorealistic lighting model for automatic technical illustration. In *SIGGRAPH '98* (1998), pp. 447–452.
- [GP06] GRIBBLE C. P., PARKER S. G.: Enhancing interactive particle visualization with advanced shading models. In *APGV '06: Proceedings of the Third Symposium on Applied Perception in Graphics and Visualization* (2006), pp. 111–118.
- [GPKM03] GHOSH A., PRABHU P., KAUFMAN A. E., MUELLER K.: Hardware assisted multichannel volume rendering. In *Computer Graphics International* (2003), pp. 2–7.
- [Hau06] HAUSER H.: Visual analysis of differential information. In *Proceedings of the International Conference of Applied Mathematics* (2006).
- [HBH03] HADWIGER M., BERGER C., HAUSER H.: High-quality two-level volume rendering of segmented data sets on consumer graphics hardware. In *Proceedings IEEE Visualization '03* (2003), p. 40.
- [HBW05] HOUSE D. H., BAIR A., WARE C.: On the optimization of visualizations of complex phenomena. In *Proceedings IEEE Visualization '05* (2005), p. 12.
- [HG00] HAUSER H., GRÖLLER M. E.: Thorough insights by enhanced visualization of flow topology. In *Proceedings of the Ninth International Symposium on Flow Visualization* (2000).
- [HH91] HELMAN J. L., HESSELINK L.: Visualizing vector field topology in fluid flows. *IEEE Computer Graphics and Applications* 11, 3 (1991), 36–46.
- [HMBG00] HAUSER H., MROZ L., BISCHI G.-I., GRÖLLER M. E.: Two-level volume rendering - fusing MIP and DVR. In *Proceedings IEEE Visualization 2000* (2000), pp. 211–218.
- [HPvW94] HESSELINK L., POST F. H., VAN WIJK J. J.: Research issues in vector and tensor field visualization. *IEEE Computer Graphics and Applications* 14, 2 (March 1994), 76–79.
- [HYW03] HASHASH Y., YAO J., WOTRING D.: Glyph and hyperstreamline representation of stress and strain tensors and material constitutive response. *International Journal for Numerical and Analytical Methods in Geomechanics* 27, (2003), 603–626.
- [IFP97] INTERRANTE V., FUCHS H., PIZER S. M.: Conveying the 3D shape of smoothly curving transparent surfaces

- via texture. *IEEE Transactions on Visualization and Computer Graphics* 3, 2 (1997), 98–117.
- [IG97] INTERRANTE V., GROSCH C.: Strategies for effectively visualizing 3d flow with volume lic. In *Proceedings IEEE Visualization '97* (1997), pp. 421–424.
- [Int97] INTERRANTE V.: Illustrating surface shape in volume data via principal direction-driven 3d line integral convolution. In *SIGGRAPH '97: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques* (1997), pp. 109–116.
- [JI03] JESSE R., ISENBERG T.: Use of hybrid rendering styles for presentation. In *WSCG* (2003), pp. 57–60.
- [JWSK07] JÄNICKE H., WIEBEL A., SCHEUERMANN G., KOLLMANN W.: Multifield visualization using local statistical complexity. *IEEE Transactions on Visualization and Computer Graphics (Proceedings Visualization/Information Visualization '07)* 13, 5 (September–October 2007), 1384–1391.
- [Kea99] KEAHEY T.: Visualization of high-dimensional clusters using nonlinear magnification. In *Proceedings of the SPIE Visual Data Exploration and Analysis VI* (1999), pp. 236–243.
- [KGM95] KRIZ R. D., GLAESSGEN E. H., MACRAE J. D.: Eigenvalue-eigenvector glyphs: Visualizing zeroth, second, fourth and higher order tensors in a continuum. In *Proceedings of the Workshop on Modeling the Development of Residual Stresses During Thermoset Composite Curing* (1995).
- [KHGR02] KNISS J., HANSEN C., GRENIER M., ROBINSON T.: Volume rendering multivariate data to visualize meteorological simulations: a case study. In *Proceedings of the 4th Joint IEEE VGTC—EUROGRAPHICS Symposium on Visualization (VisSym '02)* (2002), pp. 189–194.
- [Kin04] KINDLMANN G.: Superquadric tensor glyphs. In *Proceeding of The Joint Eurographics—IEEE TCVG Symposium on Visualization* (2004), pp. 147–154.
- [KK99] KREEGER K., KAUFMAN A.: Mixing translucent polygons with volumes. In *Proceedings IEEE Visualization '99* (1999), pp. 191–198.
- [KKH01] KNISS J., KINDLMANN G., HANSEN C.: Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *Proceedings IEEE Visualization '01* (2001), pp. 255–262.
- [KKL04] KIRBY M., KEEFE D., LAIDLAW D. H.: Painting and visualization. In *Visualization Handbook*. Academic Press, June 2004.
- [KM96] KNIGHT D., MALLINSON G.: Visualizing unstructured flow data using dual stream functions. *IEEE Transactions on Visualization and Computer Graphics* 2, 4 (1996), 355–363.
- [KML99] KIRBY R. M., MARMANIS H., LAIDLAW D. H.: Visualizing multivalued data from 2D incompressible flows using concepts from painting. In *Proceedings IEEE Visualization '99* (1999), pp. 333–340.
- [KW06] KINDLMANN G., WESTIN C.-F.: Diffusion tensor visualization with glyph packing. *Proceedings IEEE Visualization 12*, 5 (September–October 2006).
- [KWH00] KINDLMANN G., WEINSTEIN D., HART D.: Strategies for direct volume rendering of diffusion tensor fields. *IEEE Transactions on Visualization and Computer Graphics* 6, 2 (2000), 124–138.
- [LAK*98] LAIDLAW D. H., AHRENS E. T., KREMERS D., AVALOS M. J., JACOBS R. E., READHEAD C.: Visualizing diffusion tensor images of the mouse spinal cord. In *Proceedings IEEE Visualization '98* (1998), pp. 127–134.
- [LCG*02] LEE J. P., CARR D., GRINSTEIN G., KINNEY J., SAFFER J.: The next frontier for bio- and cheminformatics visualization. *IEEE Computer Graphics and Applications* 22, 5 (2002), 6–11.
- [Lev88] LEVOY M.: Display of surfaces from volume data. *IEEE Computer Graphics and Applications* 8, 3 (1988), 29–37.
- [LGS06] LARAMEE R. S., GARTH C., SCHNEIDER J., HAUSER H.: Texture advection on stream surfaces: A novel hybrid visualization applied to CFD simulation results. In *Proceedings of the Joint EUROGRAPHICS—IEEE VGTC Symposium on Visualization (EuroVis '06)* (2006), pp. 155–162.
- [LHD*04] LARAMEE R., HAUSER H., DOLEISCH H., VROLIJK B. H., POST F., WEISKOPF D.: The state of the art in flow visualization: Dense and texture-based techniques. *Computer Graphics Forum* 23(2), (2004), 143–161.
- [LJH03] LARAMEE R., JOBARD B., HAUSER H.: Image Space Based Visualization of Unsteady Flow on Surfaces. In *Proceedings IEEE Visualization 2003* (2003), pp. 131–138.
- [LvWJH04] LARAMEE R. S., VAN WIJK J. J., JOBARD B., HAUSER H.: Isa and ibfvs: Image space-based visualization of flow on surfaces. *IEEE Transactions on Visualization and Computer Graphics* 10, 6 (2004), 637–648.
- [Ma07] MA K.-L.: Machine learning to boost the next generation of visualization technology. *IEEE Computer Graphics and Applications* 27, 5 (2007), 6–9.

- [Max05] MAX N.: Progress in scientific visualization. *The Visual Computer* 21(12), (2005), 979–984.
- [MCW93] MAX N., CRAWFIS R., WILLIAMS D.: Visualization for climate modeling. *IEEE Computer Graphics and Applications* 13, 4 (1993), 34–40.
- [MFNF01] MANSSOUR I. H., FURUIE S. S., NEDEL L. P., FREITAS C. M.: A framework to interact and visualize with multimodal medical images. In *Volume Graphics* (2001), pp. 385–398.
- [MJM*06] MUNZNER T., JOHNSON C., MOORHEAD R., PFISTER H., RHEINGANS P., YOO T. S.: Nih-nsf visualization research challenges report summary. *IEEE Computer Graphics and Applications* 26, 2 (2006), 20–24.
- [MNK00] MURAKI S., NAKAI T., KITA Y.: Basic research for colouring multichannel MRI data. *Proceedings IEEE Visualization 2000*, (2000), 33.
- [MSE*06] MERHOF D., SONNTAG M., ENDERS F., NIMSKY C., HASTREITER P., GREINER G.: Hybrid visualization for white matter tracts using triangle strips and point sprites. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (2006), 1181–1188.
- [NM02] NEOPHYTOU N., MUELLER K.: Space-time points: 4d splatting on efficient grids. In *VVS '02: Proceedings of the 2002 IEEE Symposium on Volume Visualization and Graphics* (2002), pp. 97–106.
- [NSW02] NAGY Z., SCHNEIDER J., WESTERMANN R.: Interactive volume illustration. In *Vision, Modeling and Visualization 2002* (November 2002).
- [NVS00] NOORDMANS H. J., VOORT H., SMEULDERS A. W. M.: Spectral volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 6, 3 (2000), 196–207.
- [PVH*03] POST F. H., VROLIJK B., HAUSER H., LARAMEE R. S., DOLEISCH H.: The state of the art in flow visualisation: Feature extraction and tracking. *Computer Graphics Forum* 22(4), (2003), 775–792.
- [Rob00] ROBERTS J. C.: Multiple-View and Multiform Visualization. In *Visual Data Exploration and Analysis VII, Proceedings of SPIE* (2000), vol. 3960, pp. 176–185.
- [RTF*06] RÖSSLER F., TEJADA E., FANGMEIER T., ERTL T., KNAUFF M.: GPU-based multi-volume rendering for the visualization of functional brain images. In *SimVis* (2006), pp. 305–318.
- [SE03] SVAKHINE N., EBERT D. S.: Interactive volume illustration and feature halos. In *PG '03: Proceedings of the 11th Pacific Conference on Computer Graphics and Applications* (2003), p. 347.
- [SEHW02] SIGFRIDSSON A., EBBERS T., HEIBERG E., WIGSTRÖM L.: Tensor field visualisation using adaptive filtering of noise fields combined with glyph rendering. In *Proceedings IEEE Visualization '02* (2002), pp. 371–378.
- [SI05] SHENAS H. H., INTERRANTE V.: Compositing colour with texture for multi-variate visualization. In *GRAPHITE '05: Proceedings of the Third International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia* (2005), pp. 443–446.
- [SLM02] STOMPPEL A., LUM E. B., MA K.-L.: Visualization of multidimensional, multivariate volume data using hardware-accelerated non-photorealistic rendering techniques. In *PG '02: Proceedings of the 10th Pacific Conference on Computer Graphics and Applications* (2002), p. 394.
- [SMK05] STEINBERG M., MUELLER K., KELLY R.: Engine room: A comparative environment for volume rendering engines. In *Computer Graphics International 2005 Poster Proceedings* (2005).
- [Sob03] SOBEL J.: *SciVL: A Descriptive Language for 2D Multivariate Scientific Visualization Synthesis*. Tech. Rep. CS-03-24, Computer Science Department, Brown University, 2003.
- [STS06] SAUBER N., THEISEL H., SEIDEL H.-P.: Multifield-graphs: An approach to visualizing correlations in multi-field scalar data. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (2006), 917–924.
- [SVL91] SCHROEDER W., VOLPE C., LORENSEN W.: The stream polygon: A technique for 3D vector field visualization. In *Proceedings IEEE Visualization '91* (1991), pp. 126–132.
- [Tay02] TAYLOR R. M.: Visualizing multiple fields on the same surface. *IEEE Computer Graphics and Applications* 22, 3 (2002), 6–10.
- [TC06] THOMAS J. J., COOK K. A.: A visual analytics agenda. *IEEE Computer Graphics and Applications* 26, 1 (2006), 10–13.
- [TLM05] TZENG F.-Y., LUM E. B., MA K.-L.: An intelligent system approach to higher-dimensional classification of volume data. *IEEE Transactions on Visualization and Computer Graphics* 11, 3 (2005), 273–284.
- [TM05] TZENG F.-Y., MA K.-L.: Intelligent feature extraction and tracking for visualizing large-scale 4d flow simulations. In *SC '05: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing* (Washington, DC, USA, 2005), IEEE Computer Society, p. 6.

- [Tre99] TREINISH L. A.: Task-specific visualization design. *IEEE Computer Graphics and Applications* 19, 5 (1999), 72–77.
- [Tre00] TREINISH L.: Multi-resolution visualization techniques for nested weather models. In *Proceedings IEEE Visualization 2000* (2000), pp. 513–516.
- [VKG05] VIOLA I., KANITSAR A., GRÖLLER M. E.: Importance-driven feature enhancement in volume visualization. *IEEE Transactions on Visualization and Computer Graphics* 11, 4 (2005), 408–418.
- [VZKL05] VILANOVA A., ZHANG S., KINDLMANN G., LAIDLAW D. H.: Visualization and processing of tensor fields. *Proceedings of the Dagstuhl Workshop on Introduction to Visualization of Diffusion Tensor Imaging and Its Applications* (2005), Springer-Verlag.
- [War02] WARD M. O.: A taxonomy of glyph placement strategies for multidimensional data visualization. *Information Visualization* 1, 3–4 (2002), 194–210.
- [WEE03] WEISKOPF D., ENGEL K., ERTL T.: Interactive clipping techniques for texture-based volume visualization and volume shading. *IEEE Transactions on Visualization and Computer Graphics* 9, 3 (2003), 298–312.
- [WFK*02] WONG P. C., FOOTE H., KAO D. L., LEUNG R., THOMAS J.: Multivariate visualization with data fusion. *Information Visualization* 1, 3–4 (2002), 182–193.
- [WGP97] WEGENKITTL R., GRÖLLER M. E., PURGATHOFER W.: Visualizing the dynamical behaviour of wonderland. *IEEE Computer Graphics and Applications* 17, 6 (1997), 71–79.
- [Wij93] WIJK J.: Flow visualization with surface particles. *IEEE Computer Graphics and Applications* 13, 4 (1993), 18–24.
- [Wij03] WIJK J.: Image Based Flow Visualization for Curved Surfaces. In *Proceedings IEEE Visualization 2003* (2003), pp. 123–130.
- [WKZ04] WENGER A., KEEFE D. F., ZHANG S.: Interactive volume rendering of thin thread structures within multivalued scientific data sets. *IEEE Transactions on Visualization and Computer Graphics* 10, 6 (2004), 664–672.
- [WL93] WIJK J., LIERE R.: Hyperslice: visualization of scalar functions of many variables. In *Proceedings IEEE Visualization 1993* (1993), pp. 119–125.
- [WMM*02] WESTIN C. F., MAIER S. E., MAMATA H., NABAVI A., JOLESZ F. A., KIKINIS R.: Processing and visualization for diffusion tensor MRI. *Medical Image Analysis* 6, 2 (2002), 93–108.
- [Won99] WONG P. C.: Guest editor's introduction: Visual data mining. *IEEE Computer Graphics and Applications* 19, 5 (1999), 20–21.
- [WPL96] WITTENBRINK C. M., PANG A. T., LODHA S. K.: Glyphs for visualizing uncertainty in vector fields. *IEEE Transactions on Visualization and Computer Graphics* 2, 3 (1996), 266–279.
- [WS06] WOODRING J., SHEN H.-W.: Multi-variate, time varying, and comparative visualization with contextual cues. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (2006), 909–916.
- [WSEE05] WEISKOPF D., SCHRAMM F., ERLEBACHER G., ERTL T.: Particle and Texture Based Spatiotemporal Visualization of Time-Dependent Vector Fields. In *Proceedings IEEE Visualization '05* (2005).
- [Wün99] WÜNSCHE B.: The visualization of 3d stress and strain tensor fields. In *Proceedings of the Third New Zealand Computer Science Research Student Conference* (1999), pp. 109–116.
- [WWS03] WOODRING J., WANG C., SHEN H.-W.: High dimensional direct rendering of time-varying volumetric data. In *Proceedings IEEE Visualization '03* (Washington, DC, USA, 2003), IEEE Computer Society, p. 55.
- [YESK95] YAGEL R., EBERT D. S., SCOTT J. N., KURZION Y.: Grouping volume renderers for enhanced visualization in computational fluid dynamics. *IEEE Transactions on Visualization and Computer Graphics* 1, 2 (1995), 117–132.
- [ZHT07] ZHANG E., HAYS J., TURK G.: Interactive tensor field design and visualization on surfaces. *IEEE Transactions on Visualization and Computer Graphics* 13, 1 (2007), 94–107.
- [ZKL04] ZHANG S., KINDLMANN G., LAIDLAW D. H.: Diffusion tensor MRI visualization. In *Visualization Handbook*. Academic Press, 2004.
- [ZP03] ZHENG X., PANG A.: Interaction of light and tensor fields. In *Proceedings of the 5th Joint IEEE VGTC—EUROGRAPHICS Symposium on Visualization (VisSym '03)* (2003), pp. 157–166.
- [ZSH96] ZÖCKLER M., STALLING D., HEGE H.-C.: Interactive visualization of 3D-vector fields using illuminated stream lines. In *Proceedings IEEE Visualization '96* (1996), pp. 107–114.