“Flow Visualization” Juxtaposed With “Visualization of Flow”: Synergistic Opportunities Between Two Communities

Tiago Etiene[*], Hoa Nguyen[†], Robert M. Kirby[‡]

SCI Institute – University of Utah, Salt Lake City, UT, 07041, USA

Claudio T. Silva[§]

Department of Computer Science and Engineering, Polytechnic Institute of NYU, USA

Visualization is often employed as part of the simulation science pipeline. It is the window through which scientists examine their data for deriving new science, and the lens used to view modeling and discretization interactions within their simulations. We advocate that, as a component of the simulation science pipeline, visualization itself must be explicitly considered as part of the Validation and Verification (V&V) process. But what does this mean in a research area that has two “disciplinary” homes – “flow visualization” within the computer science / computational science visualization area and “visualization of flow” within the aeronautics community. Are aeronautics practitioners merely making use of algorithms developed within the visualization community that have now become “standard” through their incorporation into various visualization tools, or rather does one find *both* development of algorithms and their usage for studying fundamental and engineering fluid mechanics in both communities, with possibly different focus. By narrowing the distance between research and development, and use of visualization techniques, one is left with a fertile ground for insights, and for increasing the reliability of results through V&V. In this paper, we explore “flow visualization” from the perspective of the visualization community and “visualization of flow” from the perspective of the aeronautics community in an attempt to understand how both communities can interact synergistically to bring visualization into the simulation science pipeline. We provide a brief review of the *state-of-the-art* in flow visualization from the perspective of both communities, discuss advances in research areas such as color maps/perception and uncertainty visualization about which the AIAA community should be aware, provide some observations from both perspectives on visualizations currently published in two of the communities’ representative journals (*IEEE TVCG* and *AIAA Journal*), and then conclude by highlighting some areas of possible synergistic interaction that might benefit both communities.

I. Introduction

Flow visualization has been around in some form for as long as people have studied flows. In some cases, visualization was done explicitly – that is, with the expressed purpose of the viewer to highlight some feature of the flow. In other cases, it was done tacitly, as when a child looks out the window of an airplane to see the slip-stream over the wing generated upon take-off. Visualization has many roles, spanning from art to science. In this paper, we are focussed on visualization techniques used for the scientific exploration and explanation of flow phenomena. In particular, we are interested in how two communities – the AIAA community and the Visualization community – consider flow visualization. To accomplish this task, we have used the *AIAA Journal* and the *IEEE Transactions on Visualization and Computer Graphics (TVCG)* as

[*]Ph.D. candidate, SCI Institute – University of Utah, 50 S Central Campus Drive, Salt Lake City, UT 84112

[†]Ph.D. student, SCI Institute – University of Utah, 50 S Central Campus Drive, Salt Lake City, UT 84112

[‡]Associate Professor of Computer Science, SCI Institute – University of Utah, 50 S Central Campus Drive, Salt Lake City, UT 84112

[§]Professor of Computer Science, Department of Computer Science and Engineering, Polytechnic Institute of NYU

“representative” publication venues of the two communities, and have explored the papers published therein to try to glean how each community approaches visualization of flow, how they might differ from each other and how the two communities might complement each other.

The paper is organized as follows. In Section II we provide a review of the state-of-the-art in flow visualization, both from the perspective of the Visualization and well as the AIAA communities. Tools such as Tecplot[1] and Paraview[2] have implemented many standard flow visualization techniques such as LIC (line integral convolution), streamlines, stream ribbons, and more. As we will show, our review encompasses much of the current practices in flow visualization and also provide pointers to new developments. In the next two sections, we focus our attention on research advances made within the Visualization community that we think will, in time, have impact on flow visualization and on other application domains that use visualization as a means of both scientific exploration and explanation. In Section III we show how perception and user studies may impact flow visualization, and in particular, we focus on issues related to color maps. In Section IV, we then provide discussions on the current Visualization community research trends in Visualization Verification and Uncertainty Quantification. We have chosen these topics because they are all related to flow visualization. In Section V, we speculate on some of the opportunities for collaboration and more effective communication between the two communities, and we conclude in Section VI.

Table 1. Advances in flow visualization. This table is *not* meant to be comprehensive.

Class	Subclass	Technique	Reference
Direct	*Arrows*	Standard	Klasshen and Harrington[3]
		Hybrid	Color-coding and arrows[4]
		3D	Arrows in 3D space, 2-manifolds embedded in 3D[5]
		Enhancements	Large data,[5] resampling[6]
	Color coding	Standard	Color maps, volume rendering[7]
Geometry	*Curve*	Streamline	Turk and Banks[8]
		Seeding	User-assisted,[9] automatic,[10,11] and hierarchical[12]
		3D	2-manifolds embedded in 3D[13]
		Rendering	Illuminated,[14] streamtubes and streamribbon[15]
		Unsteady	Wiebel and Scheuermann[16]
	Surface	Stream surface	Hultquist[17]
		Enhancements	Seeding and placement,[18] accuracy[19]
		Unsteady	Schafhitzel *et al.*[20]
Texture	*LIC**	Standard	Cabral and Leedom[21]
		Performance	Improved algorithm, parallelism, real-time, GPU[22]
		3D	3D and 2-manifolds embedded in 3D[23]
		Rendering	Flow orientation cues, local velocity magnitude
		Unsteady	Li *et al.*[22]
	Spot Noise	Standard	van Wijk[24]
		Enhanced	It deals with highly curved/high velocity vector fields.[25]
		Performance	Parallel implementation.[26]
Feature	*VFT***	Standard	First-/High-order critical point tracking[27–29]
		Compression	Theisel *et al.*[30]
		Simplification	Weinkauf *et al.*[31]
		Streakline	Weinkauf and Theisel *et al.*[32]
	*STD****	Pathline	Theisel *et al.*[33]
	*LM*****	FLTE	Haller,[34] Garth *et al.*[35]

* Line Integral Convolution ** Vector Field Topology *** Space-Time Domain **** Lagrangian Method

II. Review of Flow Visualization Techniques

Vector field visualization is an important and vibrant subfield of both the Visualization and AIAA communities. The techniques developed for vector field visualization extend beyond these communities to fields such as medical imaging, meteorology, the automotive industry, and others. In the past two decades, visualization experts and practitioners have seen the development and improvement of many vector field visualization techniques. The contributions are numerous: the ability of handling different grid types (structured, unstructured, curvilinear, etc), high dimension data (2D, 2.5D, and 3D), time-dependent flow, seeding and placement of geometric primitives, improved performance, perception, rendering, among others. In this section, we review some of the developments inside the Visualization community and compare with current practices inside the AIAA community.

II.A. Preliminaries

Although the concept of flow visualization is well defined in both communities, we start by clarifying what is meant by flow visualization in this section. The difference between *computational flow visualization* and *flow visualization* is that the latter focus on visualization of flow behavior using experimental data (*e.g.*, flow in a wind tunnel), whereas the former visualizes flow from simulated or computed data. Some computational visualization techniques are inspired by techniques used in flow visualization, such as dye advection. Since the subject of this section only addresses computational flow visualization, we will refer to that topic simply as flow visualization.

For thoroughness, we also define some commonly used mathematical/physical terms used within the flow visualization literature. A *streamline* is the path traced by a massless particle in a steady flow. Streamlines are sometimes referred to as "instantaneous particle trace". A *streakline* is the path traced by massless particles seeded at the same position but at different times in a unsteady flow. *Stream surfaces* and *streak surfaces* are the 2-manifold analog of streamlines and streakline, where the seeding primitive is a curve instead of a point.

II.B. Classes of techniques

Flow visualization techniques can be classified as direct, geometric, texture-, and feature-based. Table 1 provides an overview of the classification and a subset of the available techniques within each class. The table provides a hierarchy of the flow visualization tools available. The *Subclass* column provides the main component of a given visualization techniques that can be found within the *Technique* column. One can find reference to extra material within the *Reference* column. For more details about the articles shown in Table 1 and others, we refer the interested reader to the excellent surveys by Hauser *et al.*[36] and Peng and Laramee[37] for an overview of the flow visualization field, Edmunds *et al.*[38] and McLoughlin *et al.*[39] for geometric flow visualization, Laramee *et al.*[40,41] for texture-based flow visualization, and Pobitzer *et al.*[42] for feature-based flow visualization. Next, we briefly go over each of the classes (see Figure 1).

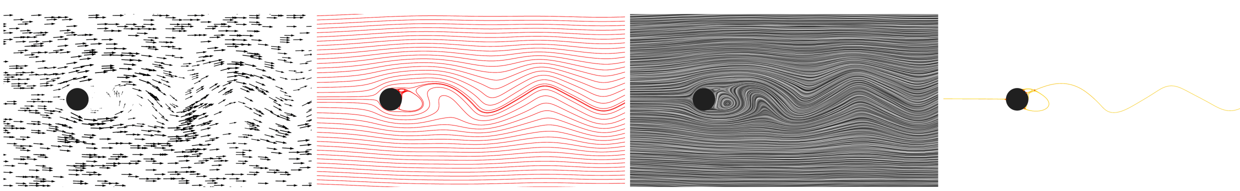

Figure 1. Examples of flow visualization using direct, geometry, texture-, and feature-based techniques, respectively.

Direct visualization Direct visualization techniques provide an intuitive and straightforward way of visualizing vector fields. In this approach, primitives of interest – such as arrows, glyphs, or lines – are placed at (often regularly-spaced) seed points. The primitives are then oriented according to the vector field. Optionally, the vector magnitude can be mapped to the primitives via scaling. Other flow properties, such as pressure and vorticity, can also be mapped using color maps. In the 3D case, volume rendering[7] is the natural choice for mapping flow properties into color and transparency. Although direct visualization provides an easy first approximation of the vector field, the visual complexity and occlusion may impair the interpretation of the results, especially in 3D datasets.

Geometric visualization In geometric visualization, curves and surfaces are used for summarizing flow behavior at particular seed points. Geometry-based approaches requires a more intensive processing of the data before the visualization than direct approaches. The main idea behind integration-based geometric flow visualization is to trace particles or curves through the vector field. By tracing particles (or respectively curves) one builds a 1-manifold (or respectively a 2-manifold) that can later be visualized. Geometric visualization techniques have a two steps: first, geometry computation; and secondly, rendering. Often, the rendering step is straightforward – *e.g.* rendering a polyline – in which case the algorithm collapses into one step. Streamlines are one of the most well-known representative visualization tools within this class. Although flow visualization using both curves and surface dates back over two decades, in recent years there has been constant research on the topic.[38] For curves, the main contributions of the past decade are related to rendering, seeding and placement of curves. Edmunds *et al.*[38] classifies the surface-based flow visualization into surface construction and rendering. Methods for surface construction are based on integral surface, implicit and topological construction. This is an area of intense research in the past few years. The authors present a variety of algorithm for both steady and time-dependent surfaces. Surface rendering methods involve the use of several techniques for improving the quality of the visualization of the flow over a surface of interest. Surface-based techniques can take advantages of direct or texture-based methods by including static/animated arrows over stream surfaces, shading for the evaluation of the shape of surfaces, placing streamlines over 3D surfaces, employing line-integral convolution (LIC) techniques, and/or non-photorealistic rendering techniques.

Feature-based visualization In feature-based flow visualization, the input vector field is segmented according to features of interest. As an example, consider a segmentation using classical vector field topology in 2D[27] (see also the right image in Figure 1). Let us assume that the features of interest are first order critical points, namely, focus source, focus sink, node source, node sink, and saddles. A segmentation is performed by building a topological skeleton through the computation of the vector field's separatrices. The final result provides a cleaner representation of the flow behavior in terms of the aforementioned features. The intensive processing of extracting features before visualization brings many advantages to the practitioner. First, feature-based techniques are valuable for visualization purposes: feature extraction provides an excellent level of abstraction of the data by removing undesired features and focusing the viewer on the important regions of the dataset. In addition, it can be used for vector field compressing, topological simplification, and even for building custom vector fields.[43] Topology-based approaches for feature-based visualization is not the only methodology available. In Lagrangian methods, the trajectories of particles are used to describe and segment the fluid flow. In particular, FLTE[34] methods have gained prominence as a research area within the last decade. One advantage of Lagrangian methods over traditional vector field topology is that they can naturally deal with unsteady flow.[42] Space-time domain techniques are another example of feature-based visualization. In this approach, in order to deal with the problems involved in unsteady flows, the problem of 2D and 3D flow visualization is moved to higher dimensions. As an example, time-dependent domains are merged into a single dataset where traditional techniques used for steady vector fields can be employed. A comprehensive survey on the topic can be found in the state-of-the-art report by Pobitzer *et al.*[42]

Texture-based visualization In texture-based flow visualization, the user replaces geometrical information with 2D texture mapped over surfaces. Line integral convolution (LIC) is a well-known (within the visualization community, at least) representative of the class. Texture-based techniques generate what is considered a dense visualization, *i.e.*, it covers the entire domain of interest, and it does not have to deal with the problem of finding appropriate seeding spots for streamlines. Texture-based techniques can be applied along with geometric or feature-based visualization; for instance, it can be used to render flow on 2-manifolds embedded in 3D spaces, or providing an overview of the flow behavior along with topological skeletons. The main issue with texture-based visualizations is the high computational cost associated with it. Nevertheless, the advances in both computer hardware and algorithms have granted to users the ability to handle large data sets and unstructured grid at interactive rates.[38,41]

II.C. Means to an end

In his position paper "On the death of visualization",[44] Lorensen argues for the need to bring visualization researchers closer to experts and practitioners. We have run a simple experiment in order to attempt to ascertain "the distance" between the Visualization and AIAA communities. We evaluated 78 articles published within the *AIAA Journal* over the period of Jan/2010-Oct/2012 containing at least one flow visualization image. Then, we simply counted the number of papers that contained at least one occurrences of the techniques shown in Table 1. We did not include the 2D color mapping and 2D isocontour visualizations as they appear quite often. Since multiple visualization techniques can be used in a single article, the percentages shown below are just the fraction of publications containing at least one particular type of visualization. Particle tracing using integration-based geometric visualization techniques for 2D vector fields is the most commonly used technique (42%), followed by 3D isocontouring (35%), 2D and 3D arrows and glyphs (33%), and 3D particle tracing (19%). Excluding isocontouring (which is mainly used for depicting scalar, instead of vector, data), 61% of the articles used at least one geometric approach to flow visualization, whereas 33% used a direct approach. Finally, 73% of the papers contained at least one visualization for 2D domains, whereas this number is 56% for 3D domains. The latter number drops to 22% if one considers only techniques for visualization of vector field data (*i.e.*, excluding 3D isocontouring).

Although the data is limited to a short window of time, it raises a few interesting points. With the exception of a handful of papers, most of the flow visualization appears to be using the standard form of the traditional visualization technique. As an example, consider some the papers that use streamlines for visualizing 3D flow. It may be the case that a subset of these paper can benefit from using stream ribbons,[15] which simultaneously encode the streamlines path and local flow vorticity, or from stream tubes,[15] which simultaneously encode the streamlines path and local cross flow divergence. Both stream ribbons and stream tubes are well-known, and commonly used visualization packages such as Paraview or Tecplot have them available within their tool options. Secondly, the preference for the two visualization techniques (direct and curve-based geometric visualization) shown in past three years is perhaps due to their simplicity and availability. The underrepresented methods in the same period of time are texture-, feature-, and surface-based flow visualization. Third, one could argue that the visualized datasets were "simple", and thus standard techniques worked well. Even though this may be the case for some datasets, some vector fields, especially in 3D, suffered from traditional problem of curves and arrows: cluttering, irregularly spaced streamlines, poor seeding, lack of depth cues, *etc.* These problems can make the detection of some flow features such as vortex more difficult. Direct visualization for 2D vector fields using glyphs can be improved by using, for instance, a resampling technique, such as shown in Laramee,[6] where the author introduce a user-driven approach for reducing visual clutter via resampling. Another way is to segment the flow using features of interest, *e.g.* critical points. Possible reasons for *not* using alternative techniques include that the technique might not be easily available, the technique might not improve the quality of the visualization, users not aware of their existence or find them difficult to use, or the AIAA community requires a different class of techniques, among other. Both communities would benefit from knowing the reasons for using one technique over another. The visualization community has, throughout the years, defined a set of priorities based on an interaction with researchers from different fields and their own experience. Some recurrent themes that are the focus of research are: a more comprehensive theory and techniques for dealing with unsteady 3D flows; improved rendering (for instance, by using techniques inspired in handcrafted illustrations[45]); handling of large data sets; and others. Together, the AIAA and Visualization communities should be able to define a set of priorities for their research agendas in order to address the concerns and issues raised.

III. Perception and Evaluation

An important aspect of the visualization research consists of the building of new visualization techniques and tools. Ideally, new techniques should be able improve the user cognitive process,[46] for instance, by allowing the visualization of data that has never been visualized before, or increasing ones ability to interact with, understand, and explore data. As visualization techniques are developed and improved, a question is raised: how can we compare and understand the differences between visualization techniques? The answer to this question leads us to a second important research topic: the need for rigorous evaluation of the strengths and weaknesses of visualization techniques. By "strength" and "weakness" we mean not only the evaluation of techniques according to traditional (computer science) metrics such as performance, memory footprint, ability to handle large datasets, *etc.*, but also in terms of the errors introduced through visualization, property of these errors, user perception, among others. In particular, questions involving perception and cognition are related to the *user*. In this section, we review two topics of interest for flow visualization from the point of view of perception and evaluation: the use of color maps for visualization of scalar properties and the representation of steady 2D vector fields, respectively.

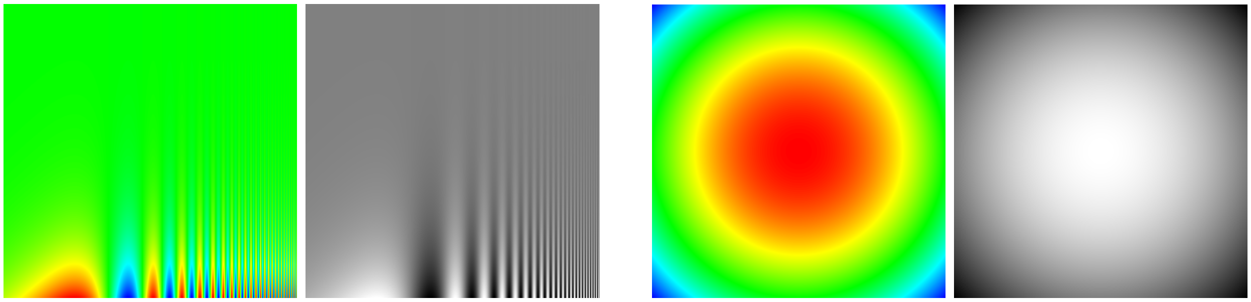

Figure 2. Left: the images show the color mapping of the spatial contrast sensitivity function. Frequency increases from left to right whereas contrast increases from the top to the bottom. The isoluminance of the rainbow color map obfuscate low contrast regions and small details, which can be seen using gray scale. Right: changes in color in the rainbow color map may be perceived as features in the data. The "boring" scalar field $f(x, y) = x^2 + y^2$ appears to have more features when rainbow color map is used than in the gray scale image.

III.A. Perception & color maps

The mapping between data and colors is ubiquitous and essential across the sciences. In the scientific pipeline, color maps are often used to study, explain, explore, and ultimately help experts to gain insight about a phenomenon of interest. Alas, color maps are not all equal, and depending on the choices made one can accelerate or impair scientific inquiry. Since they are just means-to-an-end, their impact on the underlying data should be as minimal as possible. In a myriad of choices, one color map has been shown to be a bad choice for virtually any type of visualization: the well-known and widely-used *rainbow* color map.[47–49]

The rainbow color map is built by varying hue in order to cover the whole spectrum of visible light, from red to purple or vice versa. In practice, many visualization tools use colors varying from red to blue because red and purple are very similar. It is the default map in several visualization / simulation software packages, such as Matlab®. Here we review three issues known to hinder visualizations, namely, lack of ordering, iso-luminance, and introduction of artifacts. Figure 2 shows examples for each of these issues. The first issue is due to the lack of a *natural sorting order.* Even though the rainbow color map is ordered from shorter to longer wavelength of light, users do not easily perceive it as such, which makes quantitative analysis more difficult.[47] In addition, the rainbow color map can *obscure* data. The problem arises for data containing high spatial frequency. Isoluminant maps can obfuscate these frequencies because our visual system perceives them through changes in luminance. This is illustrated in the left images in Figure 2. Note how details on the top half and left portions of the rainbow color mapped image were "removed" by the choice of the color map. Lastly, the rainbow color map can also add *artifacts* to the visualization.[50] The problem is that the gradient in color map creates the illusion of patterns where none exist. This is illustrated in the right image in Figure 2. In association with the lack of a natural sorting order, it becomes difficult to identify that patterns are not due to the underlying data but due to the color map. Although Figure 2 shows simple synthetic examples, there have also been user studies and analysis showing that these problems are also present in the visualization of real world scenarios.[50] Despite its disadvantages, the rainbow color map is widely used in the sciences. In the study by Borkin *et al.*,[51] participants reported that they liked it because they are "used to seeing", that the saturated colors are "easier to see", and it is the "most aesthetically pleasing". Another possible reason for its widespread use is that it is default in many popular simulation and visualization tools. Paraview is one of the tools that no longer uses the rainbow color map as the default option since the publication of Borland *et al.*'s "Rainbow Color Map (Still) Considered Harmful".[52] The author even suggest that a better name for it would be "misleading color map".

Table 2. Color maps in the AIAA journal

	Rainbow color map	Gray scale map	Other
2010	68.63%	13.73%	17.64%
2011	64.7%	15.69%	19.61%
2012	79.03%	8.65%	12.32%

In light of the many pitfalls of the rainbow color map, the visualization community has, in the past few years, been moving away from it. In 2005, 52% of the scientific publication using a color map at the IEEE Visualization Conference had at least one occurrence of the rainbow color map.[47] This number has

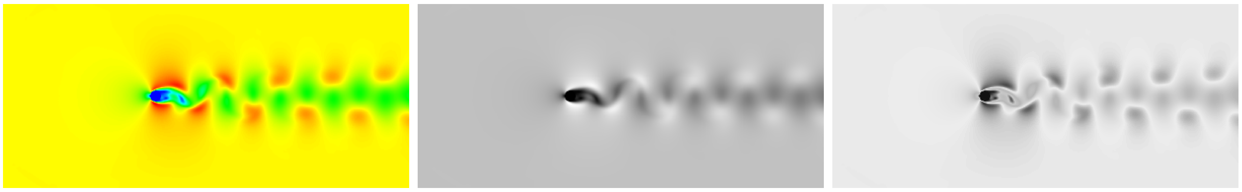

Figure 3. Velocity magnitude. Rainbow (left) and gray scale (middle) color maps were applied to a 2D flow simulation using a spectral element code for solving the incompressible Navier-Stokes Equations. Note how red regions on the rainbow color map are over emphasized while green regions "blur" details that are shown in the gray color map. The image on the right is the decolorized rainbow color map.

dropped to a single paper published at the *IEEE Transactions on Visualization and Computer Graphics* in 2011. Motivated by this experiment, we reviewed all publications from the *AIAA Journal* for the years of 2010, 2011, and 2012 that contained a color map and counted the number of papers that used the rainbow color map. Table 2 shows the obtained results. Note that we do not evaluate the potential problems caused by the rainbow color map. Nevertheless, we tried the methodology explained above for a flow simulation dataset. The left image in Figure 3 shows the results of a flow simulation. Note how some regions are over emphasized (shown in red) while details are blurred (shown in green). The problems with the rainbow color map can be avoided by simply switching to another color map, such as the gray scale color map shown in the middle image in Figure 3. The image to the right shows the decolorized rainbow color map: although some details are easier to see, the result is still very different from the gray scale color map.

The visualization community has also investigated what should constitute a "good" color map. Research on the topic of color selection can be found in the work by Treinish *et al.*,[50] Moreland,[52] Kindlmann *et al.*,[53] and others.[48,54] The AIAA community can benefit from a set of standard color maps suitable for visualization of typical simulation data such as pressure fields, angle fields, *etc.*

III.B. Evaluation & user studies

In recent years, the Visualization community has seen a substantial increase in the number of papers dealing with evaluation of visualization techniques published within *IEEE TVCG*. Figure 4 shows the number of such papers published per year within the *IEEE TVCG* journal. The data was obtained by searching the *TVCG* website for the keywords "evaluation", "user study", "design study", and "case study" in articles published in the period between 2002 and 2012. We then read the abstracts to make sure the papers were indeed relevant. From this corpora, 96% of the aforementioned articles were user studies.

As a representative example, we focus on a user study by Laidlaw *et al.*[55] comparing techniques for the visualization of steady 2D vector fields. The authors recruited five experts and 12 non-experts users to evaluate the efficacy of each of the six techniques displayed in Figure 5. The evaluation was measured by the user performance during the execution of several tasks of three types: critical point detection; critical points classification; and simulation of particle advection. The first two tasks are standard whereas the third task is motivated by the fact that often experts were interested in the global flow direction. The three tasks were chosen based on the authors interaction with fluid mechanics researchers. The authors built a collection of 500 vector fields for evaluation of the tasks. Among the results, they cite no significant difference between experts and non-experts regarding accuracy in the tasks or the response times. More interestingly, performance when using the standard method of arrows on a regular grid (GRID in Figure 5) falls below average for multiples tasks involving critical points location, classification and advection (which means that users required more time to complete the task and committed more errors). On the other end of the spectrum, user performance when using GSTR consistently scored above average. Another similar study compare the user performance when using line and tube integral curves (with monoscopic and stereoscopic viewing) for 3D vector field data.[56] User study can be a powerful tool for helping users choose the best tool for their needs and the visualization

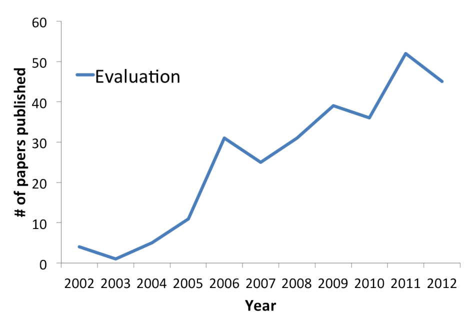

Figure 4. Evolution of the number of papers published on the topic of evaluation at TVCG.

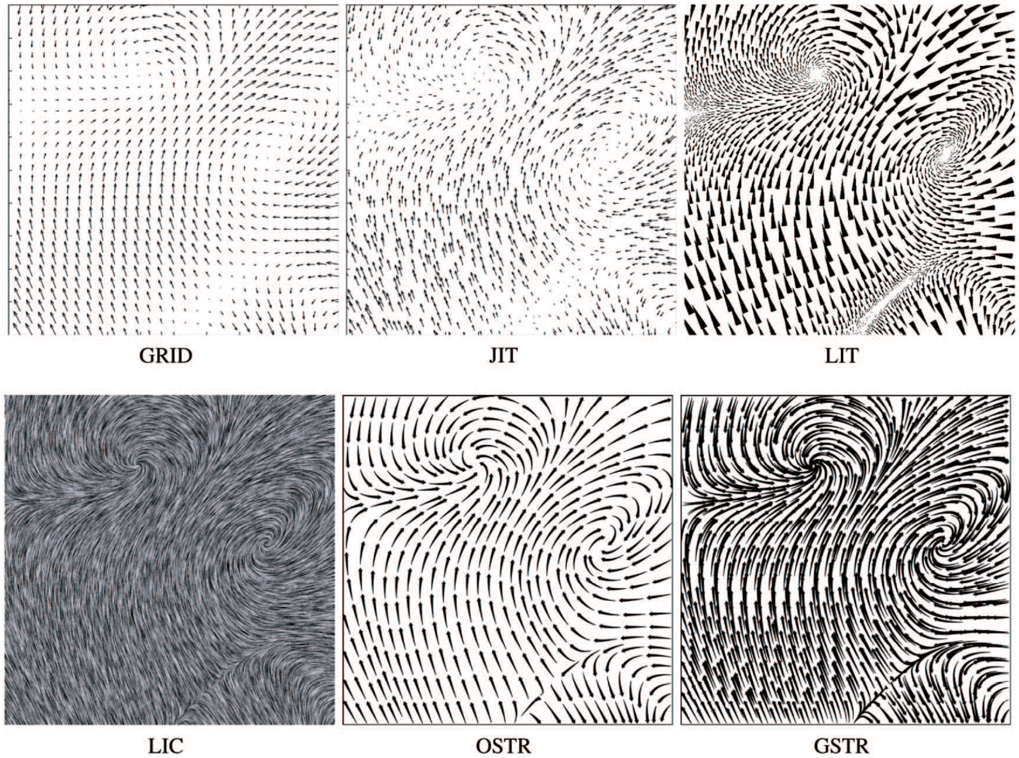

Figure 5. Comparing visualization methods for steady 2D vector fields. Top: standard arrow visualization, jittered arrow, icons using concepts borrowed from oil painting, respectively. Bottom: line-integral convolution, image-guided streamlines, streamlines seeded in a regular grid, respectively.

community has been working on evaluating and testing techniques as they become more widespread.

IV. Uncertainty and Verification

Uncertainty visualization and visualization verification are two important topics in the pursuit for reliable visualizations. The AIAA community is familiar with both topics. In this work, however, we present some of the recent advancements in this area from the point of view of the Visualization community. The goal is to increase the user confidence in the results of the visualization by answering questions such as: how can one visualize the inherent error sources in the visualization? or, how can one increase her/his confidence that an implementation of a visualization algorithm does what was intended? In the following sections we present some of the recent developments in uncertainty visualization and the verification of isosurface extraction techniques.

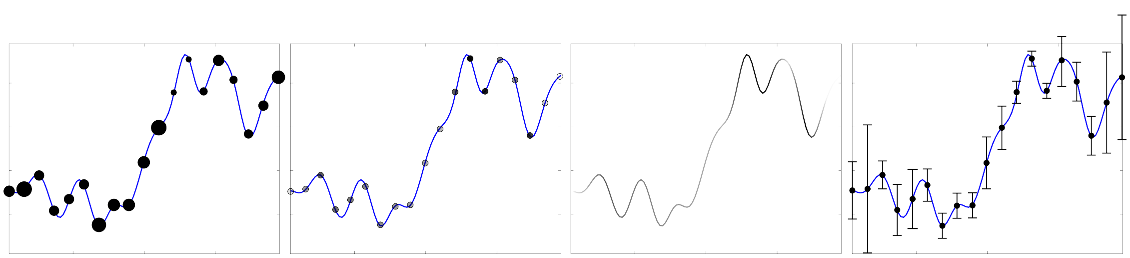

Figure 6. The four uncertainty visualization methods used by Sanyal *et al.*[57] in their user study. From left to right: glyphs size, glyphs color mapping, surface color mapping, and error bars.

IV.A. Uncertainty visualization

In the course of scientific inquiry, uncertainty is the norm. The visualization community has recently turned its attention to uncertain data, and is trying to solve problems on how to best compute and convey uncertainty information. Since 2010, around 30 papers were published at *TVCG* on the topic, with application on information visualization and scientific visualization. So far, the community has seen several different representation for uncertainty, varying from traditional method such as bars, glyphs, and colors, to texture, multi-layering, animations, and volume rendering. At the AIAA community, we analyzed ten papers since 2010 dealing with material uncertainty, uncertainty in flows, and fluid simulation. The visualization step, on the other hand, is restricted almost exclusively to error bars and charts.

In the user study conducted by Sanyal *et al.*,[57] the authors evaluate the effectiveness of four commonly used uncertainty visualization techniques: namely, glyphs size, glyphs color mapping, surface color mapping, and error bars (see Figure 6 for examples). The users performed two search tasks by identifying regions that are least and most uncertain, and two counting tasks where users counted the number of data and uncertainty features. The authors reported that, in general, users required more time and committed more mistakes when using error bars. The authors conjecture that a possible reasons for the poor performance displayed by error bars is due to the high density of the dataset used in their study. Nevertheless, a similar pattern can be found in the AIAA community (*e.g.*, see Figures 4 and 6 in Chassaing and Lucor[58]).

Several techniques for uncertainty visualization of vector fields are available. Botchen *et al.*[59] introduces a texture-mapping approach for uncertainty visualization of 2D vector fields. Hlawatsch *et al.*[60] introduces a new static visualization of unsteady vector fields with uncertainty based on a new type of glyph. Osorio and Brodlie[61] introduce a LIC-based method for uncertainty visualization. The work by Petz *et al.*[62] uses Gaussian random fields and takes into account spatial correlation of the data, which affects vector field features. Fout and Ma[63] presents a framework based on possibility theory for uncertainty visualization and as a case study, the authors use streamlines in 3D steady vector fields. Because many researchers have recently turned their attention to uncertainty visualization, this area of research is rapidly evolving.

IV.B. Verifiable visualization

Algorithm verification has recently attracted attention in the Visualization community. Although the need for verifying and validating image results dates back almost two decades, there is no systematic procedure for tackling the problem of verification in visualization. In particular, isosurface extraction has strong presence in AIAA journal for visualization of flow properties, and therefore, in this section we introduce two recent developments related to verification of isosurface extraction algorithm for the emerging field of verifiable visualizations.

We start our discussion on verifiable visualization by building a framework for the verification of isosurface extraction algorithms. Etiene *et al.*[64] borrowed the concept of the order of accuracy test from the CS&E community for assessing the quality of geometrical properties of isosurface extraction techniques. The authors manufacture solutions (using MMS) for which the behavior of each isosurface extraction technique could be analyzed, and then compare it against implementations. This framework requires a mathematical analysis of particular features of interest of each manufactured model in order to derive the *formal order of accuracy*, allowing one to compare the results produced computationally, *i.e.* the *observed order of accuracy*, to the one predicted by the analysis. By progressively refining the manufactured cases and analyzes and verifying that the numerical and analytical results are comparable, one can increase her/his confidence in the algorithm under scrutiny. For isosurfacing methods that generate simplicial approximations of smooth isosurfaces, the features of interest are geometric surface convergence, convergence of normals, area and curvature. By comparing numerically computed and analytical convergence rate the authors diagnoses and fixed problems within popular isosurfacing codes as well as better understand particular features of each technique, increasing the reliability on the methods under scrutiny. The practical impact of lacking of, say, area convergence is that, for some algorithms, the area error *increased* as the dataset was refined. By using a simple manufactured solution, the authors were able to reveal bugs that prevented the convergence of some mesh properties of two publicly available isosurfacing codes.

The authors extended their work on verification of geometrical properties of isosurface extraction algorithms to the evaluation topological properties of these techniques.[65] Unlike geometry verification, topology verification cannot be performed with order-of-accuracy tests due to the discrete nature of topological properties. This renders an approach similar to *state exploration*, used in the Computer Science literature, a more appropriate route. By *exploring* different topological configurations and comparing the expected results against the obtained through the algorithm under verification, one can verify correctness of the system or find a counter-example. The authors adapted machinery from both Stratified Morse Theory[66] and Digital Topology[67] to compute surface topological invariants directly from the grid that can later be compared against those results from the isosurface extraction algorithm under verification. As an example, the authors tested an implementation of Chernyaev's marching cubes 33,[68] a topologically-correct isosurface extraction algorithm, to their framework. Any implementation that preserves topology of the trilinear interpolant should be able to reproduce the case 13.5.2 of the extended marching cubes table.[69] The authors were able to find non-trivial bugs in the implementation and a non-obvious algorithm detail that is not discussed in either Chernyaev's or Lewiner's work.

V. Opportunities

Much of the early motivation for flow visualization in the visualization community came from the AIAA community, but over the last two decades it appears that a major gap has developed, and developments in the visualization community have been done much more independently of applications and new developments in the aeronautics area. This is in part due to the different needs of the many users of visualization techniques, including, the automotive industry, meteorology, medical imaging, geosciences, to cite a few. Summarizing decades of developments in the field of flow visualization and related areas is a nontrivial process. As an alternative, every year, a summary of recent relevant advances of visualization techniques could be published at the AIAA community; and conversely, the AIAA community could help the visualization community not only by providing expertise, but also research directions.[70] Yearly panels are held at the IEEE Vis conference, many of them with an applications focus. Consistent participation by the AIAA in these communities would help raise the level of awareness of current pressing issues. This gap between communities seems to be particular true in the need for validation and verification of visualizations techniques and codes, which over time seem to have lost track with the new rigor expected of computational codes. A related topic is the need for increasing the level of reproducibility of computational results, which cannot be simply accomplish by making codes available to other researchers.[71]

There is a natural progression from research idea within the visualization community to prototype tool, and from prototype tool to "hardened" user-available software. The challenge put forward to the visualization community to continue to seek out how to be relevant to collaborators such as our colleagues in the AIAA community, and the challenge of disseminating the advances made by the visualization community to application domains. Over the last twenty years, visualization techniques have merged as a key enabling technology for computation science by helping people explore and explain data through the creation of both static and interactive visual representations. Visualizations libraries such as Kitware's VTK contain a very large number of highly-complex visualization algorithms with thousand of lines of code implementing them. The most powerful of these algorithms are often based on complex mathematical concepts, *e.g.*, Morse-Smale complex, spectral analysis, and partial differential equations (PDEs). Robust implementations of these techniques require the use of non-trivial techniques. The overall complexity and size of these datasets leave no room for inefficient code, thus making their implementation even more complex. The complexity of the codes coupled with the new visualization techniques make it highly non-trivial for non-experts to use them, although, in principle, it should be "easier".

We believe better connections between the two communities have the chance to improve the adoption of new techniques. Furthermore, by working together, AIAA researchers can also help the Visualization community not only by providing new problems and datasets and be a major driver of problems to the community (such as they were when the visualization field was coming of age), but also by making sure the needs of the AIAA community are reflected in new research topics in Visualization.

VI. Conclusion

In this paper, we have briefly visited two decades worth of flow visualization. In particular, we first focused on vector field visualization. In this regard, we presented a classification of flow visualization seen from the perspective of the Visualization community and contrasted it with AIAA publications containing flow visualization over the last three years. By exposing the current advances in visualization, we have a starting point for building a common research agenda that can benefit both communities. In addition, we have also visited some topics related to flow visualization that have been attracting attention in the Visualization community, namely, evaluation of visualization techniques, perception, uncertainty visualization, and verifiable visualization. The common thread in all these topics is the need for improving visualization techniques in general via error mitigation, and understanding how visualization can improve the user cognitive process. We showed some of the recent work on each of these topics in the context of flow visualization. As we mentioned at the start, (computational) flow visualization is a research area that was birthed simultaneously in two communities, and early in its development benefited from strong interaction between the communities. It is our hope that a more tight coupling between the research needs/interests of the AIAA community and the research agendas of the Visualization community can be developed. This can only happen through cooperation, collaboration and communication. In part, we hope that this paper is the start of a dialog between the two communities.

Acknowledgments

The second and third authors acknowledge support by ARO W911NF-12-0375 (Program Manager Dr. Mike Coyle), NSF IIS-0914564, and Vietnam Education Foundation. The first and fourth authors acknowledge support by both NSF and DOE.

References

[1]Amtec Engineering Inc, *Tecplot, version 7 user's manual*, Amtec Engineering, Inc., 1996.

[2]Squillacote, A., *The ParaView Guide: A Parallel Visualization Application*, Kitware, 2007.

[3]Klassen, R. V. and Harrington, S. J., "Shadowed hedgehogs: a technique for visualizing 2D slices of 3D vector fields," *Proceedings of the 2nd conference on Visualization '91*, VIS '91, IEEE Computer Society Press, Los Alamitos, CA, USA, 1991, pp. 148–153.

[4]Kirby, R. M., Marmanis, H., and Laidlaw, D. H., "Visualizing multivalued data from 2D incompressible flows using concepts from painting," *Proceedings of the conference on Visualization '99: celebrating ten years*, VIS '99, IEEE Computer Society Press, Los Alamitos, CA, USA, 1999, pp. 333–340.

[5]Peng, Z., Grundy, E., Laramee, R. S., Chen, G., and Croft, N., "Mesh-Driven Vector Field Clustering and Visualization: An Image-Based Approach," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 18, No. 2, feb 2012, pp. 283–298.

[6]Laramee, R. S., "FIRST: a flexible and interactive resampling tool for CFD simulation data," *Computers & Graphics*, Vol. 27, No. 6, 2003, pp. 905 – 916.

[7]Engel, K., Hadwiger, M., Kniss, J. M., Lefohn, A. E., Salama, C. R., and Weiskopf, D., "Real-time volume graphics," *ACM SIGGRAPH 2004 Course Notes*, SIGGRAPH '04, ACM, New York, NY, USA, 2004, pp. 1–266.

[8]Turk, G. and Banks, D., "Image-guided streamline placement," *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '96, ACM, New York, NY, USA, 1996, pp. 453–460.

[9]Jobard, B. and Lefer, W., "Creating Evenly-Spaced Streamlines of Arbitrary Density," *EG Workshop on Visualization in Scientific Computing*, 1997, pp. 43–56.

[10]Mebarki, A., Alliez, P., and Devillers, O., "Farthest Point Seeding for Efficient Placement of Streamlines," *16th IEEE Visualization Conference (VIS 2005), 23-28 October 2005, Minneapolis, MN, USA*, IEEE Computer Society, 2005, p. 61.

[11]Li, L., Hsieh, H., and Shen, H., "Illustrative streamline placement and visualization," *Visualization Symposium, 2008. PacificVIS'08. IEEE Pacific*, IEEE, 2008, pp. 79–86.

[12]Jobard, B. and Lefer, W., "Multiresolution flow visualization," *WSCG (Posters)*, 2001, pp. 34–35.

[13]Spencer, B., Laramee, R., Chen, G., and Zhang, E., "Evenly Spaced Streamlines for Surfaces: An Image-Based Approach," *Computer Graphics Forum*, Vol. 28, No. 6, 2009, pp. 1618–1631.

[14]Mattausch, O., Theussl, T., Hauser, H., and Gröller, E., "Strategies for interactive exploration of 3D flow using evenly-spaced illuminated streamlines," *Proceedings of the 19th spring conference on Computer graphics*, SCCG '03, ACM, New York, NY, USA, 2003, pp. 213–222.

[15]Ueng, S.-K., Sikorski, C., and Ma, K.-L., "Efficient Streamline, Streamribbon, and Streamtube Constructions on Unstructured Grids," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 2, No. 2, jun 1996, pp. 100–110.

[16]Wiebel, A. and Scheuermann, G., "Eyelet particle tracing-steady visualization of unsteady flow," *Visualization, 2005. VIS 05. IEEE*, IEEE, 2005, pp. 607–614.

[17]Hultquist, J. P. M., "Constructing stream surfaces in steady 3D vector fields," *Proceedings of the 3rd conference on Visualization '92*, VIS '92, IEEE Computer Society Press, Los Alamitos, CA, USA, 1992, pp. 171–178.

[18]Peikert, R. and Sadlo, F., "Topologically relevant stream surfaces for flow visualization," *Proceedings of the 2009 Spring Conference on Computer Graphics*, SCCG '09, ACM, New York, NY, USA, 2009, pp. 35–42.

[19]Garth, C., Krishnan, H., Tricoche, X., Tricoche, T., and Joy, K. I., "Generation of Accurate Integral Surfaces in Time-Dependent Vector Fields," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 14, No. 6, nov 2008, pp. 1404–1411.

[20]Schafhitzel, T., Tejada, E., Weiskopf, D., and Ertl, T., "Point-based stream surfaces and path surfaces," *Proceedings of Graphics Interface 2007*, GI '07, ACM, New York, NY, USA, 2007, pp. 289–296.

[21]Cabral, B. and Leedom, L. C., "Imaging vector fields using line integral convolution," *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, 1993, pp. 263–270.

[22]Li, G.-S., Tricoche, X., and Hansen, C., "GPUFLIC: interactive and accurate dense visualization of unsteady flows," *Proceedings of the Eighth Joint Eurographics / IEEE VGTC conference on Visualization*, EUROVIS'06, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2006, pp. 29–34.

[23]Palacios, J. and Zhang, E., "Interactive Visualization of Rotational Symmetry Fields on Surfaces," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, 2011, pp. 947–955.

[24]Van Wijk, J., "Spot noise texture synthesis for data visualization," *ACM SIGGRAPH Computer Graphics*, 1991.

[25]De Leeuw, W. C. and van Wijk, J., "Enhanced spot noise for vector field visualization," *Proceedings of the 6th conference on Visualization*, 1995, pp. 45–52.

[26]Leeuw, D. and Van Liere, R., "Divide and conquer spot noise," *Supercomputing, ACM/IEEE 1997 Conference*, 1997.

[27]Helman, J. L. and Hesselink, L., "Representation and Display of Vector Field Topology in Fluid Flow Data Sets," *Computer*, Vol. 22, No. 8, aug 1989, pp. 27–36.

[28]De Leeuw, W. and Van Liere, R., "Visualization of global flow structures using multiple levels of topology," *Data Visualization*, Vol. 99, 1999, pp. 45–52.

[29]Scheuermann, G., Krüger, H., Menzel, M., and Rockwood, A. P., "Visualizing Nonlinear Vector Field Topology," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 4, No. 2, apr 1998, pp. 109–116.

[30]Theisel, H., Rssl, C., and Seidel, H.-P., "Compression of 2D Vector Fields Under Guaranteed Topology Preservation," *Computer Graphics Forum*, Vol. 22, No. 3, 2003, pp. 333–342.

[31]Weinkauf, T., Theisel, H., Shi, K., Hege, H.-C., and Seidel, H.-P., "Extracting Higher Order Critical Points and Topological Simplification of 3D Vector Fields," *Proc. IEEE Visualization 2005*, Minneapolis, U.S.A., October 2005, pp. 559–566.

[32]Weinkauf, T. and Theisel, H., "Streak Lines as Tangent Curves of a Derived Vector Field," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 16, No. 6, nov 2010, pp. 1225–1234.

[33]Theisel, H., Weinkauf, T., Hege, H.-C., and Seidel, H.-P., "Topological Methods for 2D Time-Dependent Vector Fields Based on Stream Lines and Path Lines," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 11, No. 4, jul 2005, pp. 383–394.

[34]Haller, G., "Distinguished material surfaces and coherent structures in three-dimensional fluid flows," *Phys. D*, Vol. 149, No. 4, mar 2001, pp. 248–277.

[35]Garth, C., Gerhardt, F., Tricoche, X., and Hans, H., "Efficient Computation and Visualization of Coherent Structures in Fluid Flow Applications," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 13, No. 6, nov 2007, pp. 1464–1471.

[36]Hauser, H., Laramee, R. S., and Doleisch, H., "The State of the Art in Flow visualization, part 1: Direct, Texture-based and Geometric Techniques," *IEEE Visualization*, 2003.

[37]Peng, Z. and Laramee, R., "Higher dimensional vector field visualization: A survey," *Theory and Practice of Computer Graphics*, The Eurographics Association, 2009, pp. 149–163.

[38]Edmunds, M., Laramee, R. S., Chen, G., Max, N., Zhang, E., and Ware, C., "Surface-based flow visualization," *Computers & Graphics*, Vol. 36, No. 8, 2012, pp. 974–990.

[39]McLoughlin, T., Laramee, R. S., Peikert, R., Post, F. H., and Chen, M., "Over Two Decades of Integration-Based, Geometric Flow Visualization," *Computer Graphics Forum*, Vol. 29, No. 6, 2010, pp. 1807–1829.

[40]Laramee, R., Hauser, H., Doleisch, H., Vrolijk, B., Post, F., and Weiskopf, D., "The State of the Art in Flow Visualization: Dense and Texture-Based Techniques," *Computer Graphics Forum*, Vol. 23, No. 2, 2004, pp. 203–221.

[41]Laramee, R., Erlebacher, G., Garth, C., Schafhitzel, T., Theisel, H., Tricoche, X., Weinkauf, T., and Weiskopf, D., "Applications of texture-based flow visualization," *Engineering Applications of Computational Fluid Mechanics (EACFM)*, Vol. 2, No. 3, 2008, pp. 264–274.

[42]Pobitzer, A., Peikert, R., Fuchs, R., Schindler, B., Kuhn, A., Theisel, H., Matkovi?, K., and Hauser, H., "The State of the Art in Topology-Based Visualization of Unsteady Flow," *Computer Graphics Forum*, Vol. 30, No. 6, 2011, pp. 1789–1811.

[43]Theisel, H., Rssl, C., and Weinkauf, T., "Topological Representations of Vector Fields," *Shape Analysis and Structuring*, edited by L. Floriani and M. Spagnuolo, Mathematics and Visualization, Springer Berlin Heidelberg, 2008, pp. 215–240.

[44]*On the Death of Visualization*, 2004.

[45]Brambilla, A., Carnecky, R., Peikert, R., Viola, I., and Hauser, H., "Illustrative Flow Visualization: State of the Art, Trends and Challenges," *EuroGraphics 2012 State of the Art Reports (STARs)*, 2012, pp. 75–94.

[46]Tory, M. and Möller, T., "Human Factors in Visualization Research," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 10, No. 1, Jan. 2004, pp. 72–84.

[47]Borland, D. and Taylor II, R. M., "Rainbow Color Map (Still) Considered Harmful," *IEEE Comput. Graph. Appl.*, Vol. 27, No. 2, mar 2007, pp. 14–17.

[48]MacDonald, L., "Using color effectively in computer graphics," *Computer Graphics and Applications, IEEE*, Vol. 19, No. 4, 1999, pp. 20–35.

[49]Silva, S., Santos, B. S., and Madeira, J., "Using color in visualization: A survey," *Computers & Graphics*, Vol. 35, No. 2, 2011, pp. 320 – 333.

[50]Treinish, L. et al., "Why Should Engineers and Scientists Be Worried About Color?" *IBM Thomas J. Watson Research Center, Yorktown Heights, NY*, 2009.

[51]Borkin, M. A., Gajos, K. Z., Peters, A., Mitsouras, D., Melchionna, S., Rybicki, F. J., Feldman, C. L., and Pfister, H., "Evaluation of Artery Visualizations for Heart Disease Diagnosis," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, 12/2011 2011.

[52]Moreland, K., "Default Color Map," *http://www.org//ParaView3//index.php//Default_Color_Map*, 2007.

[53]Kindlmann, G., Reinhard, E., and Creem, S., "Face-based Luminance Matching for Perceptual Colormap Generation," *Proceedings of IEEE Visualization 2002*, October 2002, pp. 299–306.

[54]Tominski, C., Fuchs, G., and Schumann, "Task-Driven Color Coding," *Information Visualisation, 2008. IV '08. 12th International Conference*, 2008, pp. 373–380.

[55]Laidlaw, D. H., Kirby, R. M., Jackson, C. D., Davidson, J. S., Miller, T. S., da Silva, M., Warren, W. H., and Tarr, M. J., "Comparing 2D Vector Field Visualization Methods: A User Study," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 11, No. 1, Jan. 2005, pp. 59–70.

[56]Forsberg, A., Chen, J., and Laidlaw, D., "Comparing 3D Vector Field Visualization Methods: A User Study," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 15, No. 6, Nov. 2009, pp. 1219–1226.

[57]Sanyal, J., Zhang, S., Bhattacharya, G., Amburn, P., and Moorhead, R., "A User Study to Compare Four Uncertainty Visualization Methods for 1D and 2D Datasets," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 15, No. 6, Nov. 2009, pp. 1209–1218.

[58]Chassaing and Lucor, D., "Stochastic Investigation of Flows About Airfoils at Transonic Speeds," *AIAA Journal*, Vol. 48, 2010, pp. 938–950.

[59]Botchen, R. P. and Weiskopf, D., "Texture-Based Visualization of Uncertainty in Flow Fields," *Visualization Conference, IEEE*, Vol. 0, 2005, pp. 82.

[60]Hlawatsch, M., Leube, P., Nowak, W., and Weiskopf, D., "Flow Radar Glyphs & Static Visualization of Unsteady Flow with Uncertainty," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 12, dec. 2011, pp. 1949–1958.

[61]Allendes Osorio, R. and Brodlie, K., "Uncertain flow visualization using LIC," *7th EG UK Theory and Practice of Computer Graphics: Proceedings*, 2009.

[62]Petz, C., Pthkow, K., and Hege, H.-C., "Probabilistic Local Features in Uncertain Vector Fields with Spatial Correlation," *Computer Graphics Forum*, Vol. 31, No. 3pt2, 2012, pp. 1045–1054.

[63]Nathaniel Fout, K.-l. M., "Reliable Visualization: Verification of Visualization based on Uncertainty Analysis," Tech. rep., University of California, Davis, 2012.

[64]Etiene, T., Scheidegger, C., Nonato, L. G., Kirby, R. M., and Silva, C., "Verifiable Visualization for Isosurface Extraction," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 15, No. 6, 2009, pp. 1227–1234.

[65]Etiene, T., Nonato, L. G., Scheidegger, C., Tienry, J., Peters, T. J., Pascucci, V., Kirby, R. M., and Silva, C. T., "Topology Verification for Isosurface Extraction," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 18, No. 6, June 2012, pp. 952–965.

[66]Goresky, M. and MacPherson, R., *Stratified Morse Theory*, Springer, 1988.

[67]Stelldinger, P., Latecki, L. J., and Siqueira, M., "Topological Equivalence between a 3D Object and the Reconstruction of Its Digital Image," *IEEE Trans. Pattern Anal. Mach. Intel.*, Vol. 29, No. 1, 2007, pp. 126–140.

[68]Chernyaev, E. V., "Marching Cubes 33: Construction of Topologically Correct Isosurfaces," Tech. rep., CERN, 1995.

[69]Lopes, A. and Brodlie, K., "Improving the Robustness and Accuracy of the Marching Cubes Algorithm for Isosurfacing," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 9, No. 1, 2003, pp. 16–29.

[70]Munzner, T., Johnson, C., Moorhead, R., Pfister, H., Rheingans, P., and Yoo, T. S., "NIH-NSF Visualization Research Challenges Report Summary," *IEEE Comput. Graph. Appl.*, Vol. 26, No. 2, March 2006, pp. 20–24.

[71]Silva, C. T., Freire, J., and Callahan, S. P., "Provenance for Visualizations: Reproducibility and Beyond," *Computing in Science and Engineering.*, Vol. 9, No. 5, Sept. 2007, pp. 82–89.