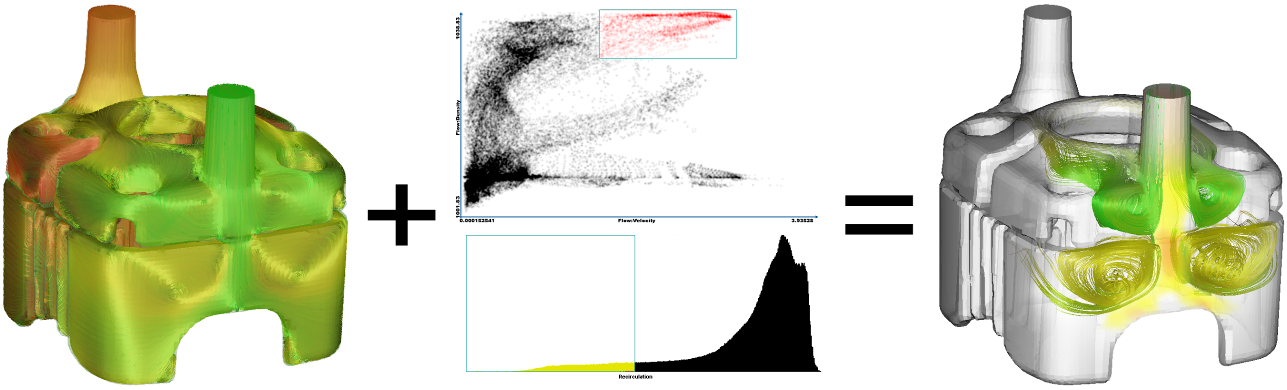
A Framework for Global Scope Interactive Visual Analysis of Large Unsteady 3D Flow Data

Martin Ilčík*

Faculty of Mathematics, Physics and Informatics
Comenius University, Bratislava, Slovakia
and
VRVis Research Center, Vienna, Austria



Abstract

We introduce a framework for interactive visualization of global flow features in large unsteady 3D flow fields. It is based on selective visualization using dense precomputed integral lines (streamlines, path lines) linked together with all other data attributes. This way we are able to provide an uniform and interactive environment for custom feature specification and visualization of global data aspects. These are related to the long term flow behavior, thus their description using only local properties is not possible. Furthermore we propose a strategy for recognition of global recirculation zones within 3D vector fields, judging on a self-proximity measure of integral lines.

Keywords: flow visualization, integral lines, global feature detection, recirculation

1 Introduction

Flow visualization deals with multivariate vector field data defined over a geometric mesh. Its application scope spans from automotive and aerospace industry through medicine right up to meteorology. Typically the data sets are produced by computational fluid dynamics simulations, with results containing over a million cells in several time steps, reaching the edge of present PCs processing capabilities. Thus special visual analysis tools are needed to understand the data as a whole. A good visualization of the flow field is very important for efficient creation of a mental model that helps the user to orientate in the data volume.

Given a 3D time-dependent vector field $v(x,t)$, we can trace the movement of a massless particle from a position x_0 at a time t_0 by solving the initial value problem for an ordinary differential equation

$$\frac{dx}{dt} = v(x,t), \quad x(t_0) = x_0 \tag{1}$$

Its trajectory can be represented by constructing a path line starting at position x_0 at time t_0

$$p_{x_0,t_0}(t) = x_0 + \int_0^t v\left(p_{x_0,t_0}(\tau), \tau + t_0\right) d\tau \tag{2}$$

Another class of lines is often used to visualize the tangent curves of v for a fixed time t. A streamline starting at position x_0 at time t_0 can be written as

$$s_{x_0,t_0}(t) = x_0 + \int_0^t v\left(s_{x_0,t_0}(\tau), t_0\right) d\tau \tag{3}$$

Hauser [7] gives a more detailed overview of integral curves. In order to solve the initial value problem and to construct the integral lines, we need to discretize the problem by rewriting the Eq. (1) to a discrete form. Using a substitution with finite steps Δx and Δt, we get

$$\Delta x = \Delta t \cdot v(x,t) \tag{4}$$

*ilcik@vrvis.at

Once a step length $\tau = \Delta t$ is chosen, one can express the particle's position after $n+1$ steps using a simple recurrence called Euler integration method

$$x_{n+1} = x_n + \tau \cdot v(x_n, t_n) \tag{5}$$

For a more accurate numerical integration, higher-order methods from the Runge-Kutta family with a fixed or adaptive step size can be used. The most popular are RK4, RK3(2) and the Fehlberg method, all explained in [17].

Integral lines belong to the geometric flow visualization, because they require to extract geometric objects related to the data in order to display them. Our approach concentrates on feature based visualization. Especially for large data sets it is important to allow a custom degree of abstraction showing only essential structural elements of the flow. In our framework we provide a set of tools allowing the user to specify interesting features using selections on various attributes. Data values, mesh connectivity and integral lines act as layers for our concept of uniform manipulation with local and global properties. By linking these layers together, we can significantly help to gain insight into a specific problem represented by the data.

The rest of this paper is organized as follows: Section 2 provides a brief overview of similar or related work to our concept. In section 3, we define the concept of a layered model for visual analysis of multivariate flow data. Moreover our contribution shows how to unify and link all these layers to an uniform environment. Section 4 presents our framework in detail. We propose a simple recirculation measure based on global flow analysis in section 5. A practical application of the layered model to industry data analysis is described in section 6. Finally we summarize our results and consider possible future work in section 7.

2 Related work

Utilizing particle traces for visualization of fluids has always been popular. Texture based visualization uses local integration for excellent interactive renderings of static and dynamic flow [10]. Feature based analysis and visualization of flow structures with help of enormous amounts of integral lines has gained attention only recently. Park et al. propose a visualization technique based on dense, uniformly distributed integral lines [15]. They address many important perceptual and performance issues. Chen et al. use a similarity measure of polylines for streamline placement [3]. Wischgoll and Scheuermann present a strategy for localization of closed lines in 3D [20] using the mesh topology to track line cycles reentering the same cells. A more general approach by Salzbrunn and Scheuermann [18] is heading in the direction of custom feature specification. It uses different data measures for the specification and computation of fuzzy streamline predicates, similar as we do in our global scope flow analysis approach. The selective visualization part of our framework and its usage for line data analysis comes very close to the approach by Shi et al. [19]. The authors present a set of path line measures suited for general flow analysis. For interactive examination of lines they use ComVis, a system oriented to information visualization. Unfortunately another system has to be used for computation and visualization of lines on the geometrical domain, preventing general interactivity, usage freedom, attributes linking and perception of spatial relations.

Detection of recirculation areas addressed in section 5 is related to the vortex detection topic, well described in the state of the art report by Post et al. [16] and also in context of SimVis by Bürger et al. [1]. Fuchs et al. [6] use parts of our framework for an improvement of vortex region detectors. They track the response to local vortex detectors along integral lines and use it for a global scope flow analysis.

The SimVis system

Our work presented in this article was implemented as a part of the SimVis system [4], a framework allowing the user to formulate own definitions of features describing significant objects, phenomena or structures, important for a particular problem. Each data item has a degree of interest (DOI) value ranging from 0 to 1. It is used to reveal the implicit data features by specifying selections on their values. To simplify the selection process of continuous quantities, smooth brushing is used [5]. For the DOI management, there is a hierarchical tree structure called feature definition language (FDL). DOI functions are stored in leafs, any superior node represents a logical combination of its children. SimVis uses multiple linked views to simultaneously display different visualizations of various data properties. Active views utilize InfoVis approaches and allow brushing, passive views show a 3D visualization on the geometrical mesh with the FDL root node mapped on the data. The views are linked together with a joint DOI function, so any user action causes changes in all of them. SimVis is based on the Focus+Context approach, thus all views have to use the DOI to discern focus from context. Focused regions are visualized with more emphasis and detail, context parts provide only supplemental overview.

In the next sections we will exploit the concept of SimVis in order to allow global scope flow analysis.

3 Flow driven analysis

The essential feature of our framework is the ability to provide a flexible and uniform environment for interactive visual analysis of global data aspects. Local feature detection methods investigate only a small geometrical neighborhood of a point. We use large amounts of precomputed integral lines as primitives for the global data analysis. The trajectories gained by following one or more particles in the vector field provide us important information on the flow behavior from a Lagrangian point of view. They can

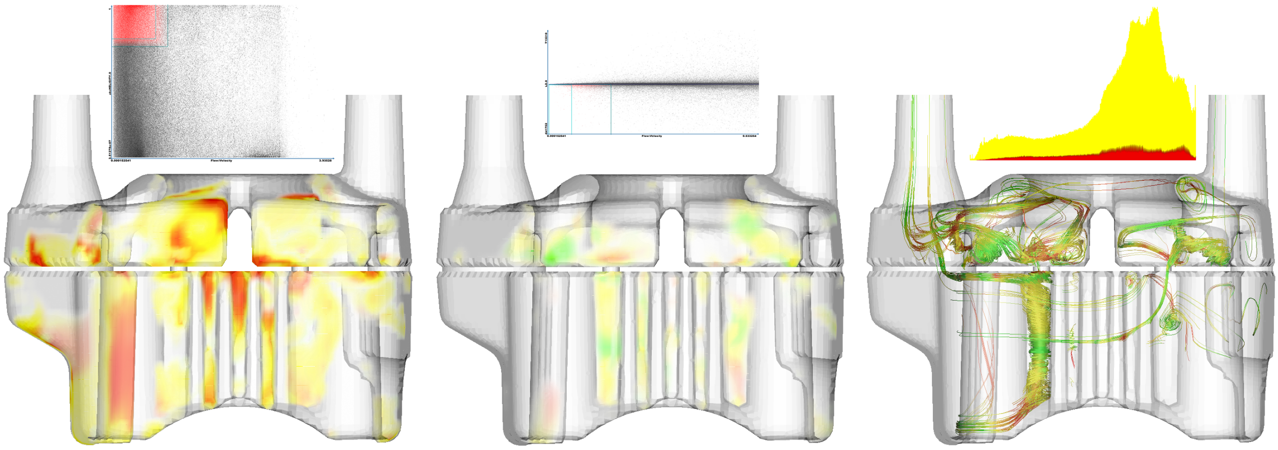


Figure 1: Layered analysis of a small cooling jacket. (left): The scatterplot shows velocity magnitude on the horizontal and vorticity magnitude on the vertical axis. The selection brushes regions with stagnant but swirling flow. High vorticity is shown in red. (middle): With help of local analysis a $lambda_2$ data channel is derived, substituting vorticity in the scatterplot. Now negative lambda values with low velocity are brushed and shown. The color still represents vorticity, thus we can see green regions with low values. (right): Global flow analysis allows us to show only the lines crossing the brushed cells. We have refined the previous selection by adding a recirculation condition (see sec. 5) brushed in the histogram. In a short time, we were able to isolate a long vertical vortex useful for the analysis of the cooling process.

span over a large portion of the spatio-temporal data domain, representing global properties of the flow. The line primitives provide us a new kind of complex structural information, that could never be obtained without the integration process. Many features like recirculation (see section 5) are imperceptible without a global scope analysis. This fact leads us to a clear separation of three layered approaches for flow data visual analysis used in our framework. All described layers can access the cell centered data storage and inferior layers for purposes of their own primitives computation.

Direct data analysis is restricted to the raw data values, without involving any geometrical knowledge of the underlying mesh. We can consider a data value of a cell as the basic entity for this layer. During the analysis it typically uses only a set of interval brushes in data attribute dimensions for selecting cells with certain interesting values like low pressure, high temperature etc.

Local flow analysis offers additional connectivity information involving the topological structure of the mesh. The flow is being observed from an Eulerian point of view, thus the basic entity on this stage is a set of neighboring cells. It enables the usage of local differential information for evaluation of regional data properties. Many well established local feature detection criteria are based on spatial or temporal differentials, e.g. different vortex detectors using the velocity gradient tensor [1, 2].

Global flow analysis extends the connectivity information to a wide area, but in our case we restrict it to the integral line seeded at the examined point. Here we change to the Lagrangian approach, using an integral line as the primitive. The main benefit for the exploration process is the possibility to investigate the flow behavior before it reaches and after it leaves the examined region. Furthermore it allows analysis based on local attributes of line points and global line properties. Many recent works use global scope flow analysis, e.g. [18, 19].

All of the present systems for interactive visual analysis of 3D flow data can handle only separate layers at once. There is no possibility of data exchange between different primitives. In our framework we introduce the interactive linking and brushing [5] of these three layers. The linking involves not only the access of the superior layers to the lower ones, but mainly the automatic bidirectional mapping of the DOI values between the layers. The cells act as an interconnection of the analysis primitives, thus a bijective transformation between the primitives of each type and the cell set is available. Using our approach, it is possible to combine selections on all the three layers to a uniform DOI rating of the cells. As shown on fig. 1, we could use the local layer to select the cells with a high gradient magnitude of a certain data channel. Then using the flow layer, one could refine the selection by omitting cells

Layer	Primitive	Source	Display
Direct	value	data channel	color coding
Local	set of cells	mesh topology	arrow glyph
Global	integral line	integration	colored lines

Table 1: Layers for interactive flow analysis

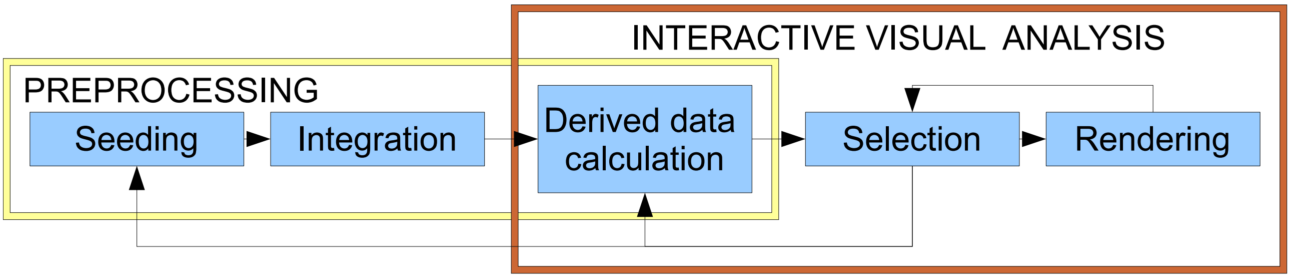


Figure 2: Pipeline for our framework.

that do not accomplish a certain global criterion defined for the line. The brushed cells immediately appear also in the focus region of active views managing other layers. Moreover all user actions directly affect passive views including the combined volumetric and lines renderer, so that the visual representation is instantly updated to be as true to the focused data as possible.

4 Framework overview

The set of proposed methods was designed and implemented as a part of the SimVis system, hence it closely relates to its concept and architecture. The data analysis layer is a core part of the system, we have incorporated the local scope analysis in the last years [1]. Since the global layer is the only one to require additional precomputation in order to maintain interactivity of the exploration process, we have designed two modules operating independently from each other. We can divide a typical usage scenario into several stages, realized by the modules as seen on the figure 2.

Preprocessing module

We use large amounts of precomputed integral lines as primitives for the global flow analysis. In order to create them, we need to set seed points or a seeding strategy and integrate from this positions using a continuous representation of the vector field. Setting the seed point at cell center is a good choice for the later linking, since the data values in our application are also sampled at cell centers. By default we create one representative line for each cell, but the user can decide to omit computation of unimportant lines by seeding only in cells with positive DOI.

For the numerical integration one has to choose between speed and accuracy. The fastest one is a direct Euler method with step size adapted to the underlying geometry. It is using nearest neighbor lookups for the velocity vectors. For a higher precision one can choose some of the adaptive Runge-Kutta methods with mean value coordinates interpolation [9]. All lines can be limited by a maximum length setting. Currently we offer two types of lines to be computed: streamlines and path lines. All generated lines are saved in the form of a new data set. Its size can be estimated by the upper boundary $cells_{selected} * maxsteps * 30$ bytes, hence it can easily reach problematic limits of several GB.

Additional derived data, mostly local and global measures operating on the lines geometry, can be also precomputed along with the integration or later during the exploration process. We will provide an example and discuss benefits of this feature later in section 6.

Visualization module

The interactive part of the pipeline allows the user as much freedom as possible to organize and guide her/his analysis. Following the principles of knowledge driven visualization, the SimVis system provides the experts with means to formulate their own feature definitions [4]. The main part of the interaction relates to various selections on active linked InfoVis views [5]. Storing the integral data as a common data set allows us to reuse the core modules of the SimVis and to exploit its selective visualization capabilities. The user can brush on the global scope level just like on the other layers. It is made possible by a linkage of cells and layer primitives. Lines are build up of segments, connecting pairs of succeeding points. We split line segments at cell boundaries, so we can easily construct a bijective mapping of segments to cells. Moreover each segment carries a reference to the seed segment of the line allowing the bijective mapping from lines to cells.

The actual DOI can be passed back to the preprocessing module as the seeding area for a new set of lines. One can also supplement the existing lines with additional attributes. It is possible to share the data across all the layers to involve it into various derived data calculations.

The most important type of view for the spatial orientation is the 3D renderer. It provides hybrid visualization of the data volume using multiple techniques simultaneously. Our lines rendering plugin can operate as a standalone renderer or as a part of a combined rendering pipeline. The principal task of visualization bound to a geometrical domain is to provide a clear overview of the flow volume and spatial relations between the flow features. Large amounts of lines as we produce them, can easily overwhelm the image space. Intersections and cluttering cause problems by depth perception and orientation, important structures can be hidden. A lot of research has been done in this field: lines density regulation by advanced seeding strategies [13, 11] and rendering enhancements for better visual separation in dense line fields [15]. Our renderer uses illuminated lines [12] that greatly help to improve the recognition of single lines in the image. Naturally we can map scalar attributes to the color channel, using build-in transfer functions. The lines density is implicitly adjusted by the data selections. Automatic linking of the DOI across the views immediately propagates its values from the cells to their line primitives, so that only curves belonging to the brushed cells are shown in the final image.

Framework performance

The visualization module runs fully interactively using common personal computer hardware. The number of displayed lines is limited by the graphics memory and fill rate, but it rarely happens that the user would select such a big number of line segments. Brushing performance is sometimes affected by the InfoVis views incorporated in the SimVis core, since our line data are one of largest the system ever had to deal with. The current limit given by the SimVis is a approximately 2 GB of line segments.

The preprocessing module uses numerical integration methods to compute the lines. Computation time depends on the selected integration and interpolation method. On the fly integration would not allow interactive analysis anymore. We have measured the average performance of our integration methods on unstructured meshes using a Pentium D processor with 2 GB of RAM:

Method	Interpolation	Segs / s	Cooling jacket
Euler	none	$\approx 10^5$	$\approx 25\,\text{min}$
RK2	mean value	$\approx 10^4$	$\approx 250\,\text{min}$

Table 2: Integration performance overview

5 Recirculation detection in steady flows

As stated in the survey of feature-based visualization [16], one of the areas with additional work needed is detection of new feature types, including recirculation zones. A lot of feature detection is done on a local basis, involving only a small geometrical neighborhood of investigated position. However it is impossible to detect a recirculation area using only local detectors. Our precomputed lines provide a global representation of the flow field, containing new connectivity information suitable for efficient feature detection. If a particle advected in the flow field returns closely to one of its previous positions, we speak of recirculation. Streamlines contain all positions of a particle moving in a steady flow, making them ideal for a fast and simple detection of recirculation.

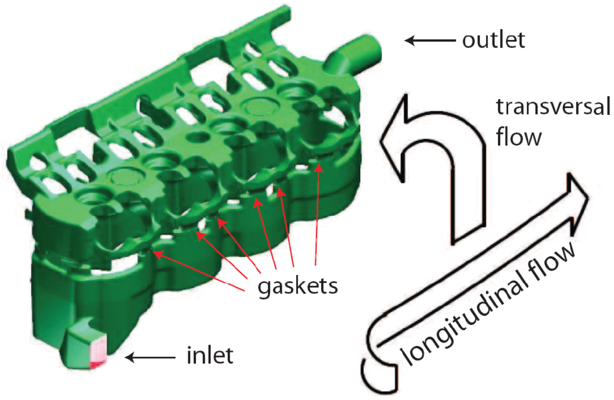


Figure 3: The major flow components inside of the cooling jacket include a longitudinal component from inlet to outlet and a traversal component in the upward-and-over direction through the gaskets. Image courtesy [8].

Given a line as an ordered sequence of points $c = x_0, \ldots, x_{n-1}$, we can precompute the length of the polyline $x_i, \ldots, x_j$ for all pairs of points i and j as

$$d_{ij} = d_{ji} = \sum_{i \le k < j} \|x_k - x_{k+1}\| \tag{6}$$

Due to the triangle inequality, the polyline length is always equal or larger than the euclidean distance of its endpoints.

$$\forall i \neq j : d_{ij} \geq \left\|x_i - x_j\right\| \tag{7}$$

We define our recirculation measure p_i for point i as

$$p_i = \min_{j \in \{0, \ldots n-1\} \setminus \{i\}} \left(\frac{\left\|x_i - x_j\right\|}{d_{ij}} \right) \tag{8}$$

Since we deal only with non-degenerate segments, $d_{ij} > 0$ always holds and $p_i \in (0, 1]$. Let us consider a particle at position x_i, moving in a vector field. As the particle moves forward passing through the points x_j, the polyline length d_{ij} of its trajectory will grow. But when the particle starts to draw near to its own line, the Euclidean distance of the points $\left\|x_j - x_i\right\|$ will decrease. The response will grow weaker with increasing polyline length and decreasing Euclidean distance.

6 Application example

We will demonstrate the possibilities of our framework by resuming the analysis of a cooling jacket from [2, 8]. Engineers from the automotive industry use computational fluid dynamics software to simulate possible designs of essential engine parts. The SimVis system provides them an interactive visual analysis environment for comparison and optimalization of the simulation results. Our subject is a cooling jacket of a four cylinder engine, sampled in 1.5 million cells. It consists of three main components: The cylinder head on the top, the cylinder block on the bottom and a thin gasket connecting the previous two components together (fig. 3)

The cylinder head is responsible for transferring the heat away from the intake and exhaust ports at the top of the engine block.

The cylinder block transfers the heat from the engine cylinders and distributes the flow evenly to the head.

The gasket consists of small but important ducts regulating the cooling liquid's passage from the block to the head.

The goal is to achieve an even distribution of the flow to each engine cylinder and to avoid overheating caused by recirculation of hot fluid in particular regions.

Turbulences causing recirculation

We know from the engineers, that a temperature of $363\,\mathrm{K} \approx 90°C$ is considered to be optimal. In our previous work, we were able to identify potential problematic regions by focusing on areas with critical temperatures and nearly stagnant flux. Our work flow was a typical example for direct data analysis. Afterwards we have refined our feature definition on the level of local analysis, using velocity gradient tensors to detect swirling motion. Some vortical areas preventing fast transport of overheated fluid have appeared mainly in the lower parts near the outlet. We have concluded that geometry in this areas should be improved in order to prevent the overheating.

With help of global scope analysis we can achieve even better results. Our recirculation criterion in connection with selections on other attributes, detects the lines returning to overheating areas, so we can easily brush and show the interesting lines (fig. 4). We were able to identify a strong recirculation region inside of a bridge between the exhaust ports of the fourth cylinder next to the outlet. It is probably caused by the thin passages in the surrounding geometry. On the figure 6 you can see that there are actually three vortices acting against each other. We were not able to recognize this detail in our previous work. The perceptually improved visualization of the vortex in the lower left component examined in [2] is shown on fig. 7.

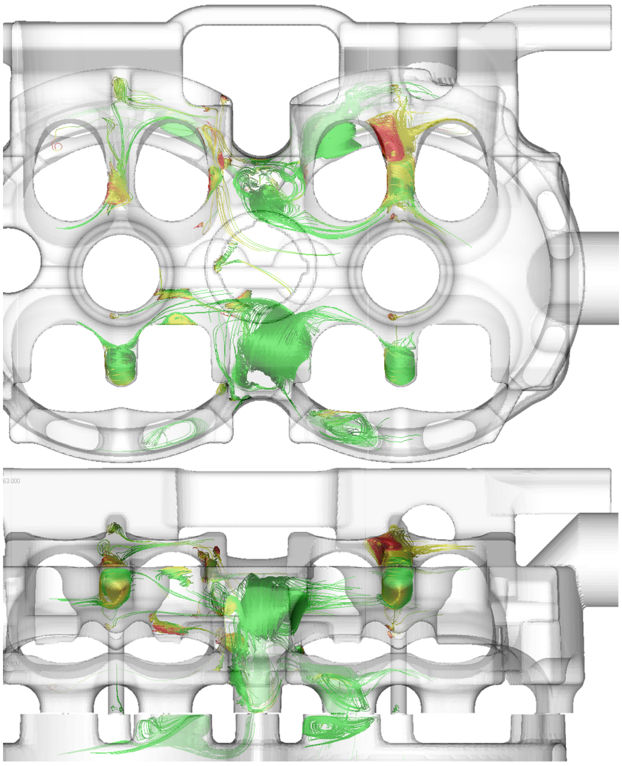


Figure 4: Overview of the recirculation regions around the fourth cylinder, top and side projection. Green regions symbolize low temperatures, red ones are critical. Most of the vortices are caused by the gasket connections, but in one region the fluid circulation causes overheating.

Turbulences accelerating the heat transport

Despite the impression from the previous analysis, most of the vortices found in the data set are caused by intentional design of jacket geometry to dissipate the hot fluid from the boundary and mix it with the colder rest. We have examined such regions also in our previous work. Now we are finally able to look at the swirling structures behind the ducts to confirm their positive effect on the heat transportation. On the figure 5 you can see one of these very fast rotating vortices indicated with red color over a junction of the block and the head part.

Looking at the global flow properties, it is much easier to perform a visual analysis of large data. In this example we have used temperature and velocity values together with a recirculation measure to select interesting line segments. To visualize the found structures, whole lines from the selected segments were drawn, using our rendering module in combination with the hybrid unstructured raycaster [14].

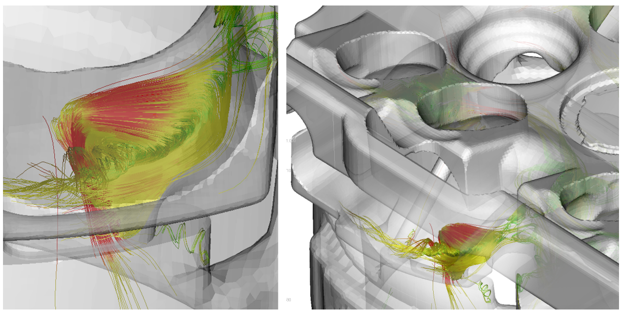


Figure 5: Detail and overview of a gasket vortex causing no problems. Now velocity magnitude is expressed with color. Red indicates fast swirling flow improving the cooling process.

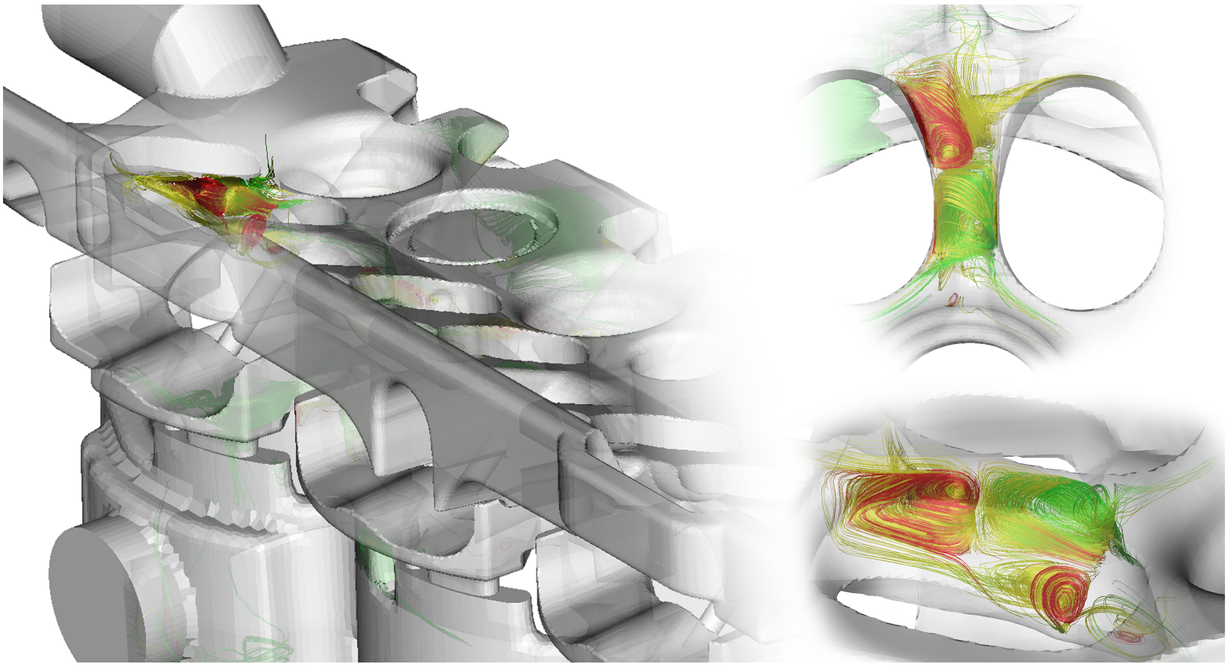


Figure 6: Visualization of an unwanted vortex with temperature mapped on the color of the streamlines. Critical values are shown with red, optimal with green. (left): An overview of the cooling jacket with a strong recirculation zone. Red color indicates overheating caused by blocked heat transport. (top): A top view of the recirculation zone caused by non-optimal jacked design. (bottom): We can recognize three separate vortices pushing against each other and conserving the flow.

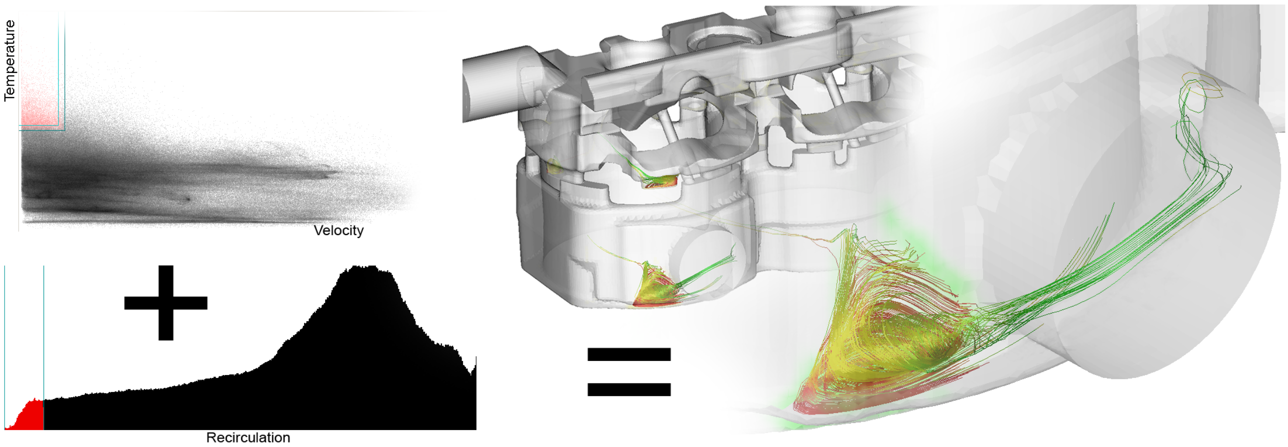


Figure 7: (top left): Brushing was used to select high temperatures and low velocities. (bottom left): Linked selection of recirculating line segments. The linked DOI displays only lines seeded in selected cells. (middle): Overview of the examined area under the outlet. (right): Detail of a recirculation zone delaying the cooling process displayed using a focus+context visualization. The temperature increases towards the center of the zone becoming red.

7 Conclusions

We have introduced a framework for interactive global flow analysis of large 3D and 4D multivariate flow data sets. The concept of linking all layers together has brought increased efficiency to the exploration process. Interactive examination of the flow structures from a Lagrangian point of view has shown to be very helpful for understanding of complex relations within the data. The intense preprocessing of integral lines has enabled the interactive manipulation with data, at the cost of high storage requirements. Nevertheless this tradeoff allows to run our framework on every common personal computer.

For future work there are many challenges following our concept. GPU computation of integral lines for large unstructured data sets is a very promising idea, incorporating compression of polylines would greatly help not only in the preprocessing stage. Our next work will probably focus on optimalization of the whole pipeline, allowing even larger data to be processed in a shorter time.

Acknowledgments

I would like to thank Raphael Fuchs and Philipp Muigg for their long lasting generous support during my work presented in this paper. I appreciate all their comments, advices and patience. Additional thanks go to Helwig Hauser, Andrej Ferko and Matej Novotný. This work has been realized at the VRVis Research Center in Vienna, Austria[1], funded by the Kplus program. The small cooling jacket CFD is courtesy of Institute for Internal Combustion Engines and Thermodynamics, TU Graz. The large one is courtesy of AVL List GmbH, Graz.

References

[1] Raphael Bürger, Philipp Muigg, Helmut Doleisch, and Helwig Hauser. Interactive cross-detector analysis of vortical flow data. In *Proceedings of CMV 2007*, pages 98–110. IEEE, 2007.

[2] Raphael Bürger, Philipp Muigg, Martin Ilčík, Helmut Doleisch, and Helwig Hauser. Integrating local feature detectors in the interactive visual analysis of flow simulation data. In *Proceedings of Eurographics/ IEEE-VGTC Symposium on Visualization 2007*, pages 171–178, 2007.

[3] Yuan Chen, Jonathan D. Cohen, and Julian Krolik. Similarity-guided streamline placement with error evaluation. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1448–1455, 2007.

[4] Helmut Doleisch, Martin Gasser, and Helwig Hauser. Interactive feature specification for focus+context visualization of complex simulation data. In *VISSYM '03: Proceedings of the symposium on Data visualisation 2003*, pages 239–248, 2003.

[5] Helmut Doleisch and Helwig Hauser. Smooth brushing for focus+context visualization of simulation data in 3d. In *WSCG*, pages 147–154, 2002.

[6] Raphael Fuchs, Bilal Alsallakh, and Meister Eduard Gröller. Delocalized unsteady vortex region detectors. Technical Report TR-2007-030, VRVis Research Centre, 2007.

[7] Helwig Hauser. Visual analysis of differential information. In *Proceedings of the International Conference of Applied Mathematics*, 2006.

[8] Helwig Hauser, Robert Laramee, and Helmut Doleisch. Topology-based versus feature-based flow analysis - challenges and an application. In *Topology-based Methods in Visualization (Proceedings of TopoInVis 2005)*. Springer-Verlag, 2007.

[9] Tao Ju, Scott Schaefer, and Joe Warren. Mean value coordinates for closed triangular meshes. volume 24, pages 561–566. ACM, 2005.

[10] Robert S. Laramee. *Interactive 3D Flow Visualization Based on Textures and Geometric Primitives*. PhD thesis, ICGA, Vienna University of Technology, 2004.

[11] Liya Li and Han-Wei Shen. Image-based streamline generation and rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(3):630–640, 2007.

[12] Ovidio Mallo, Ronald Peikert, Christian Sigg, and Filip Sadlo. Illuminated lines revisited. In *IEEE Visualization*, page 3, 2005.

[13] Oliver Mattausch, Thomas Theußl, Helwig Hauser, and Meister Eduard Gröller. Strategies for interactive exploration of 3d flow using evenly-spaced illuminated streamlines. In *SCCG '03: Proceedings of the 19th spring conference on Computer graphics*, pages 213–222. ACM, 2003.

[14] Philipp Muigg, Markus Hadwiger, Helmut Doleisch, and Helwig Hauser. Scalable hybrid unstructured and structured grid raycasting. *IEEE Trans. Vis. Comput. Graph.*, 13(6):1592–1599, 2007.

[15] Sung Park, Brian C. Budge, Lars Linsen, Bernd Hamann, and Kenneth I. Joy. Dense geometric flow visualization. In *Data Visualization 2005 (Proceedings of the EUROGRAPHICS - IEEE VGTC Symposium on Visualization 2005)*, June 2005.

[16] Frits H. Post, Benjamin Vrolijk, Helwig Hauser, Robert S. Laramee, and Helmut Doleisch. The state of the art in flow visualisation: Feature extraction and tracking. *Comput. Graph. Forum*, 22(4):775–792, 2003.

[17] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C++ - The Art of Scientific Computing - Second Edition*. 2002.

[18] Tobias Salzbrunn and Gerik Scheuermann. Streamline predicates. *IEEE Trans. Vis. Comput. Graph.*, 12(6):1601–1612, 2006.

[19] Kuangyu Shi, Holger Theisel, and Hans-Peter Seidel. Path line attributes - an information visualization approach to analyzing the dynamic behavior of 3d time-dependent flow fields. In *Topo-In-Vis 2007*, pages 60–74. Springer, 2007.

[20] Thomas Wischgoll and Gerik Scheuermann. Locating closed streamlines in 3d vector fields. In *VISSYM '02: Proceedings of the symposium on Data Visualisation 2002*, pages 227–ff. Eurographics Association, 2002.

[1] http://www.VRVis.at/