Preprint

M. Hlawatsch, F. Sadlo, H. Jang, and D. Weiskopf.
Pathline Glyphs.
Computer Graphics Forum, 33 (2): to appear, 2014.

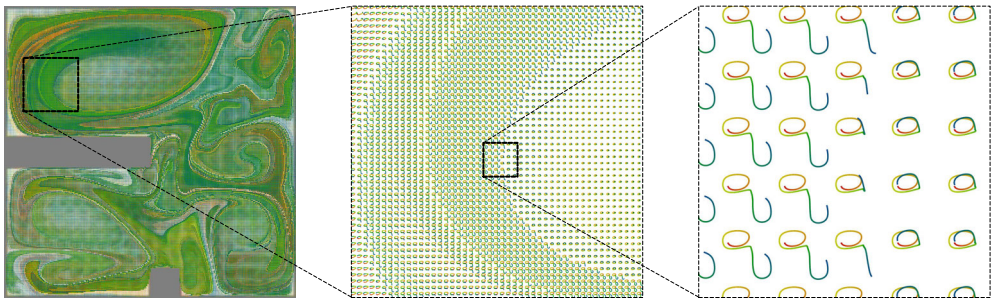
EUROGRAPHICS 2014 / B. Lévy and J. Kautz
(Guest Editors)

Volume 33 (2014), Number 2

Pathline Glyphs

M. Hlawatsch, F. Sadlo, H. Jang, and D. Weiskopf

Visualization Research Center (VISUS), University of Stuttgart, Germany



(a) (b) (c)

Figure 1: *Multi-scale visualization of unsteady flow with pathline glyphs. (a) Emergent patterns visible on the overview scale correspond to structures in the flow and provide the basis for detailed exploration, but individual pathline glyphs cannot be perceived. (b) On an intermediate scale, patterns remain present, but the shapes of the glyphs become visible. (c) A close-up facilitates detailed analysis of time-dependent transport: particles started in the left area later move downward (blue), while particles from the right area stay in the whirl.*

Abstract

Visualization of pathlines is common and highly relevant for the analysis of unsteady flow. However, pathlines can intersect, leading to visual clutter and perceptual issues. This makes it intrinsically difficult to provide expressive visualizations of the entire domain by an arrangement of multiple pathlines, in contrast to well-established streamline placement techniques. We present an approach to reduce these problems. It is inspired by glyph-based visualization and small multiples: we partition the domain into cells, each corresponding to a downscaled version of the entire domain. Inside these cells, a single downscaled pathline is drawn. On the overview scale, our pathline glyphs lead to emergent visual patterns that provide insight into time-dependent flow behavior. Zooming-in allows us to analyze individual pathlines in detail and compare neighboring lines. The overall approach is complemented with a context-preserving zoom lens and interactive pathline-based exploration. While we primarily target the visualization of 2D flow, we also address the extension to 3D. Our evaluation includes several examples, comparison to other flow visualization techniques, and a user study with domain experts.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—I.6.6 [Simulation and Modeling]: Simulation Output Analysis—

1. Introduction

The visualization of time-dependent vector fields is an important topic in flow visualization, since many phenomena can only be understood by looking at the temporal behavior of a flow. In particular, the direct visualization of pathlines is a common tool for analyzing unsteady flow. These lines represent the trajectories of massless particles transported by the flow. However, one of the main issues with this approach is that the resulting images quickly become cluttered with an increasing number of lines—amongst others due to

intersection. Interactive exploration of the flow field with a low number of pathlines reduces these issues, but analyzing the whole domain with this approach is time-consuming and does not provide a comprehensive view of the flow.

We present an approach to improving the pathline visualization of unsteady 2D flow. The first contribution is the "uncluttering" of such visualization. Our approach encodes the same data as a direct visualization by an arrangement of pathlines, but with less visual clutter. This is achieved by using downscaled pathlines as glyphs—*pathline glyphs*. The second contribution is that the resulting visualization allows the analysis on multiple scales (Figure 1). On the overview scale, emergent patterns become visible. We demonstrate that these correspond to significant flow structures. On smaller scales, individual pathlines can be analyzed and compared with respect to their neighborhood. We complement our approach with interaction techniques that allow context-preserving exploration of the flow. The evaluation of our method includes comparisons to other visualization techniques, several examples, and an assessment by domain experts.

2. Related Work

The huge variety of different techniques for flow visualization can be classified in different groups: glyph-based techniques [BKC*13], dense flow visualization [LHD*04, LEG*08], geometric flow visualization [MLP*10], and topology-based techniques [PPF*11].

In the field of geometric flow visualization, line placement and selection techniques are an important topic. Recent work on streamline placement methods include the method by Yu et al. [YWSC12], who use flow saliency to generate streamline bundles, and an approach to parallelize placement in 2D fields by Zhang et al. [ZWZ*13]. While placement methods adapt the seed points, selection methods reduce a set of given lines, e.g., McLoughlin et al. [MJL*13] reduce streamline sets for interactive seeding, while Tao et al. [TMWS13] treat streamline selection as a problem combined with viewport selection. A comprehensive overview of streamline placement and selection can be found, e.g., in the work by Günther et al. [GRT13].

In contrast to streamlines, pathlines can intersect; hence, placement and selection of them are more difficult and only few papers consider this topic. Günther et al. [GRT13] present a general method for sets of lines—including pathlines—which adapts the opacity to avoid the occlusion of important features. Pathline selection can also be achieved with the methods by Weinkauf et al. [WTS12], Salzbrunn et al. [SGSM08], and Falk et al. [FSf*10].

In contrast to those concepts, our approach avoids overlapping lines by downscaling and can be seen as a glyph-based technique. Glyph-based techniques use symbolic elements to represent properties at selected discrete locations.

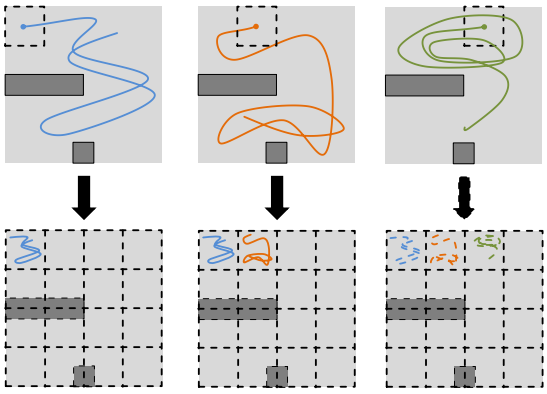


Figure 2: *Basic concept of pathline glyphs. The domain is partitioned into non-overlapping cells (dashed). Each cell represents the full domain and contains a single downscaled version of a pathline (colored).*

Their flexibility is one of their main advantages and allows encoding very different types of information. Peng et al. [PGL*12] use glyphs to encode local flow properties of their vector field clusters. An augmentation of non-scaled streamlines with glyphs is presented by Pilar and Ware [PW13]. The glyphs of De Leeuw and Van Wijk [dLvW93] show additional local flow properties like rotation and shear. The superquadric glyphs by Schultz and Kindlmann [SK10] can also be used for flow visualization and Kirby et al. [KML99] demonstrate how to combine different glyphs encoding flow properties. Flow Radar Glyphs by Hlawatsch et al. [HLNW11] provide a static representation of local velocities of unsteady flow. Those techniques share a local approach to flow visualization: they show the flow at points or local neighborhoods around those points. In contrast, our technique visualizes transport behavior in flow fields with a Lagrangian perspective on particle movement that includes their global motion patterns.

Our approach also exhibits similarities to some common concepts from information visualization. Drawing multiple plots into a single chart often leads to visual clutter; the small multiples approach [Tuf90] splits such a chart into a grid of multiple charts for better comparison. Sparklines [Tuf06] use downscaled plots for integration into continuous text. However, our approach employs spatial sampling of a domain that is of typically much higher density.

3. Pathline Glyphs

To achieve a pathline-based visualization with less visual clutter, we transform the pathlines into glyphs—pathline glyphs—by downscaling each of them (Figure 2). To this end, the domain is subdivided into cells, each representing the domain and containing a single pathline. The shape of the

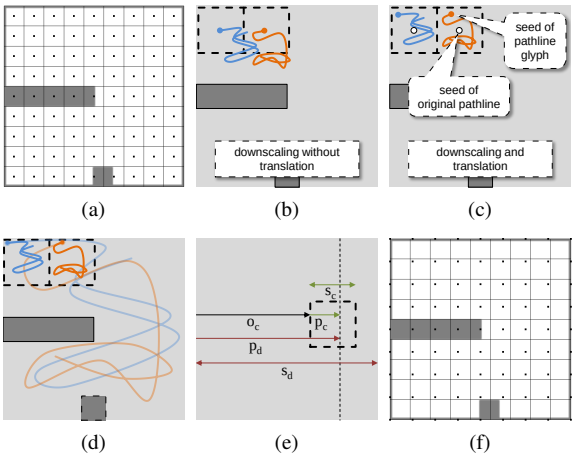


Figure 3: *Seeding strategies. (a) Using the center of each cell as seed point and downscaling the pathline without translation would result in an inconsistent visualization: (b) the cell would not represent the domain with respect to the downscaled pathline. (c) The respective translation would restore the alignment with the cell, but the seed point of the pathline glyph would not match the seed point of the original, non-scaled pathline. (d) The goal is now that the seed point of the pathline glyph is identical to the seed point of the original pathline (low saturation). (e) This requires that the relative position with respect to the domain (p_d/s_d) equals the relative position inside the cell (p_c/s_c). For simplicity, only the x-coordinate is illustrated; the scheme is identical for all coordinates. (f) Resulting seed points used in our approach.*

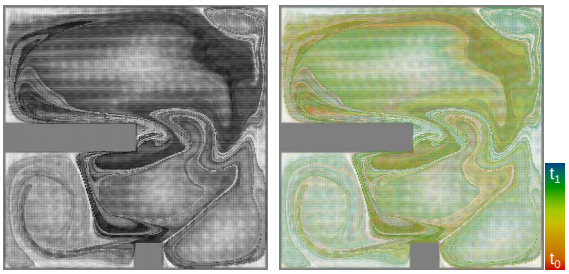


Figure 4: *Effect of color mapping. (a) Patterns are visible in densely seeded pathline glyphs due to their variation in local shape and, thus, texture. (b) Mapping additionally integration time to color enhances these patterns. The color map shown is used for all images, unless otherwise noted.*

pathline glyph inside each cell corresponds to the shape of the original, non-scaled pathline in the entire domain. Using non-overlapping cells avoids intersection of the lines; only self-intersection is possible, which is still allowed because we want to preserve the shape of individual pathlines.

Like for all glyph-based visualizations, different seeding strategies are possible. We concentrate on regular seeding grids because they are widely applied and easy to interpret. For regular seeding, a straightforward choice would be to use the center of each cell. However, this would be problematic, as Figure 3 illustrates. Simply downscaling the pathline would lead to an offset of the line with respect to the cell (Figure 3(b)) and an additional translation would introduce inconsistency between the seed points of the pathline glyph and the original pathline (Figure 3(c)). This would make a correct interpretation difficult, especially when combining the glyphs with interactively seeded pathlines (Section 4.2).

For these reasons, we use seed points that have the same relative position in the domain *and* the respective cell (Figure 3(d)). We derive the respective equations only for one dimension. They are directly applied to the other dimensions as well. To obtain such a seed with identical relative position, the ratio between the position p_d in the domain with size s_d and the position p_c inside the cell with size s_c must be equal: $p_d/s_d = p_c/s_c$ (Figure 3(e)). Furthermore, the position inside the domain must equal to the sum of cell offset o_c and cell position p_c, i.e., $p_d = o_c + p_c$. Deriving $p_c = s_c(p_d/s_d)$ from the first equation and inserting this in the second leads to $p_d = o_c + s_c(p_d/s_d)$. Solving this for p_d results in $p_d = \frac{o_c}{1-s_c/s_d}$. The resulting seed points can be seen in Figure 3(f). We can see from the last equation that the offset between neighboring points is constant.

Patterns are visible in densely seeded pathline glyphs due to their variation in shape (Figure 4(a)). This effect can be enhanced by color mapping. By default, color encodes the integration time (between t_0 and t_1) along the pathline glyph (Figure 4(b)), but other encodings are possible as well (Section 4.3). With the integration time mapped to color, we can see which time span is covered by the represented pathline, e.g., if they stop at domain boundaries. This exhibits similarities to the level set method by Westermann et al. [WJE00].

It becomes apparent that densely seeded pathline glyphs provide a multi-scale visualization of the flow data (Figure 1). On the overview scale, they make salient patterns in the data visible (Figure 1(a)). Regions with pathlines of similar behavior, both with respect to space and time, can be identified. Zooming-in allows us to analyze the respective flow behavior (Figure 1(b)). In this example, a region with pathlines of compact shape can be detected on the right, but more complex glyphs are also visible. Zooming-in further (Figure 1(c)) provides a detailed view on pathlines of two types of time-dependent flow behavior: while those on the right are part of a whirl over the entire time range, those on the left leave the whirl later in downward direction. This example shows that the visualization with pathline glyphs does not only work on different scales but also reveals qualitatively different information on them.

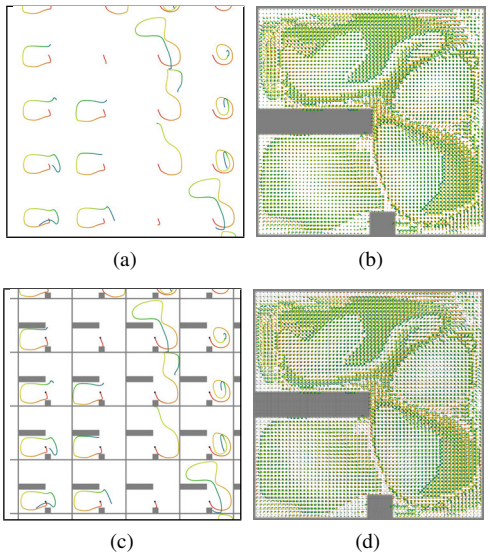


Figure 5: *Additional visual elements to support the understanding of pathline glyphs. (a) Displaying pathline glyphs alone makes it difficult to interpret the pathlines with respect to the domain. (b) On the overview scale, this is not problematic because the glyphs are usually too small to see details. (c) Displaying the boundaries of the data domain together with the respective seed points allows us to easily understand the shape of the pathline with respect to the domain. (d) However, this interferes with the recognition of patterns on the overview scale.*

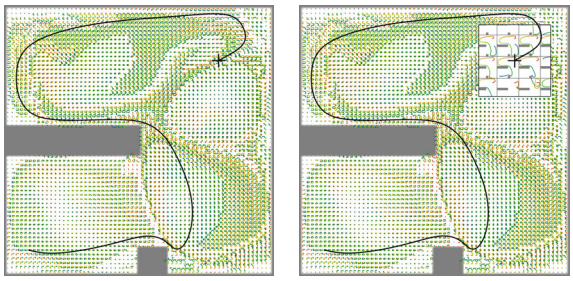


Figure 6: *Interaction for pathline glyphs. (a) An interactively controlled, full-sized pathline (black curve) enables the analysis of individual pathlines without leaving the overview scale. (b) An additional zoom lens, magnifying the area around the seeding point of the interactive pathline, provides the local context of the pathline.*

4. Extensions

Even though the presented concept reduces the visual clutter in pathline-based visualization and provides for flow analysis on multiple scales, we present extensions that can complement the work with our pathline glyphs.

4.1. Domain Grid

The shape of a pathline glyph is given by the corresponding pathline. Hence, the glyph shows the trajectory of a particle started at the respective seeding position. However, displaying just the glyphs makes it difficult to interpret the trajectory with respect to the domain (Figure 5(a)). This can be improved by additionally displaying information about the domain and the seeding grid. This supports the understanding of single glyphs (Figure 5(c)). The drawback is that this can interfere with the perception of patterns on the overview scale (Figure 5(d)). Therefore, the best approach is to display the grid only on zoomed-in scales.

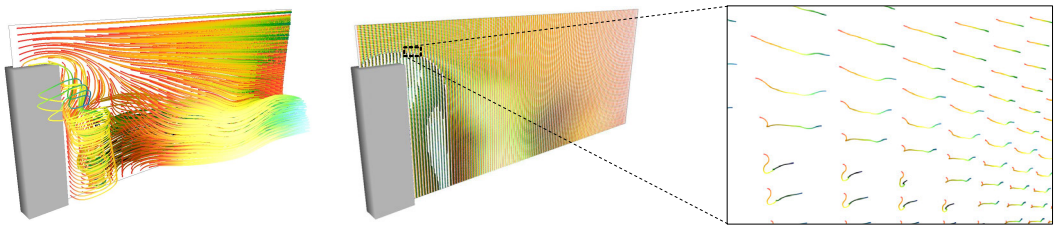
4.2. User-Controlled Pathline and Zoom Lens

Working with pathline glyphs typically requires frequently changing between scales: identifying interesting patterns on the overview scale, and then analyzing them in detail on a zoomed-in scale. We provide additional interaction techniques that can make this more efficient. Displaying additional full-sized pathlines that can be interactively controlled, e.g., with a computer mouse, allows us to analyze single pathlines without leaving the overview scale (Figure 6(a)). A region of interest can be explored by interactively seeding these pathlines. However, this alone would end up in a trial-and-error approach because the glyphs are usually too small on the overview scale to provide hints for the exploration. Zooming-in would not help because parts of the pathline can be located outside the zoomed-in area.

A solution for this problem is a zoom lens showing the area around the seed point of the pathline (Figure 6(b)). This provides the local context of the pathline and allows a more efficient exploration. It has to be noted that due to the pathline glyphs, the zoom lens provides the local context over the full time range. This can typically not be accomplished when using a zoom lens for classic pathline visualization. In that case, many neighboring pathlines could leave the zooming area and only an initial part of the time range would be visible. Figure 5 shows that an additional grid can help analyze pathline glyphs, but that it interferes on the overview scale. Hence, showing the grid only in the zoom lens is a good solution to improve the overall analysis with pathline glyphs.

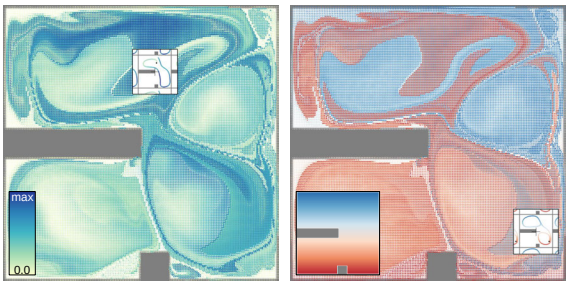
4.3. Quantity Encoding

So far, we have already demonstrated color coding applied to time, i.e., the mapping of physical time to color (Figure 4(b)). This is our default color coding for pathline glyphs because the temporal structure of pathlines is important for the analysis of unsteady flow. Nevertheless, other quantities can be mapped to color (Figure 8), depending on the underlying problem. If, e.g., areas of high velocity are of interest,



(a) (b) (c)

Figure 7: *Extension of pathline glyphs to 3D. (a) Traditional pathline visualization of 3D data suffers additionally from visual clutter due to projection, even when the lines are only seeded on a plane. (b) Our glyphs also reduce visual clutter in this case. Besides the pattern from the color mapping, the shading according to the distance to the seeding plane induces additional patterns. (c) A closer view shows the transition in the marked area. Parts of the upper lines stay in front of the seeding plane (brighter segments), while parts of the lower lines are behind the plane (darker segments).*



(a) (b)

Figure 8: *Possible encodings of quantities. (a) Velocity magnitude along the pathlines can be mapped to color. This mapping allows us to detect regions that give rise to pathlines with high velocities (dark blue). Looking at individual glyphs, we can see the velocity variation along the pathline. (b) Another possibility is a spatial mapping. In this example, we represent the y-coordinate with the color map (bottom left). In this case, the glyph visualization shows in which area the pathline travels most.*

it can be helpful to map flow velocity to color (Figure 8(a)). In that example, pathlines started in a large area in the upper part of the domain exhibit high velocities. If, however, the spatial context of the pathline is of interest, e.g., to analyze transport phenomena, the spatial position can be mapped to color (Figure 8(b)). The image shows that large groups of pathlines travel mainly in the area where they were seeded.

4.4. Extension to 3D

The basic concept of pathline glyphs—downscaling pathlines around their seed points—can be easily transferred to 3D applications. However, in this case, our glyph visualization will suffer from typical problems of 3D visualization, including issues with occlusion and depth perception. Furthermore, in contrast to the 2D case, the downscaling cannot avoid visual clutter resulting from projection. To address these issues, glyphs should only be seeded on a plane or surface. We exemplify this approach in Figure 7, showing visualizations of the time-dependent 3D flow behind an obstacle. We use shading to improve depth perception with respect to the seeding plane: the distance to the plane modulates the brightness of the vertices, i.e., parts of the line in front of the plane become brighter, parts behind the plane darker.

We can see that seeding traditional pathlines on a plane can already result in a cluttered image (Figure 7(a)), because lines that do not intersect can still overlap after projection. Due to the downscaling of the lines, our glyphs can avoid this issue. The image becomes less cluttered and patterns are better visible (Figure 7(b)). For example, due to our shading model, we can see the changes in the distance of the pathlines to the seeding plane. Zooming-in allows us to investigate this behavior in detail (Figure 7(c)), revealing that the upper lines stay in front of the seeding plane, while the lower lines are partly behind the plane.

5. Implementation

Our visualization tool was developed in C++, CUDA was used for computing the glyphs on the GPU, and OpenGL was used for rendering. All results were generated on a PC with an Intel Core2Quad CPU (2.4 GHz), 4 GB RAM, and an nVidia GeForce GTX 560 with 2 GB memory. Every pathline glyph can be computed independently from the others. Hence, parallel computation on GPUs is straightforward. We start one GPU thread for every glyph and employ the 4^{th}-order Runge-Kutta scheme with fixed step size for numerical integration. Our GPU implementation allows us to interactively change glyph parameters like the starting time, at least for shorter time ranges and lower seeding resolutions.

We were able to reuse existing code for pathline computation on the GPU, which shows the ease of implementation of our approach. We only had to add the proper scaling functionality. The glyphs are stored in two Vertex Buffer Objects,

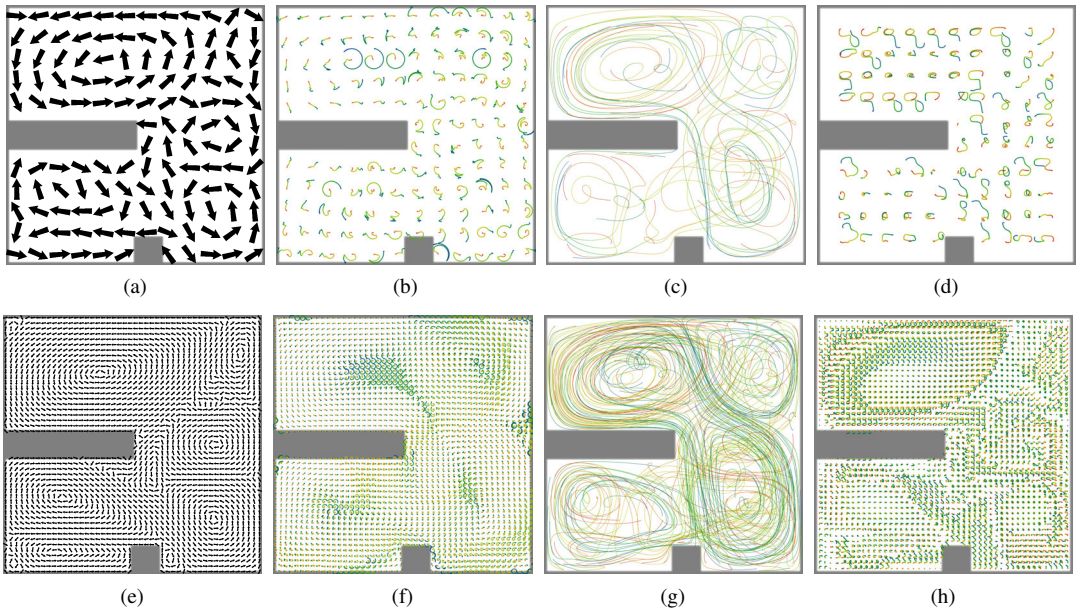


Figure 9: *Comparison with other flow visualization techniques for the Hotroom dataset. (a) While classic arrow glyphs provide only an instantaneous view, (b) flow radar glyphs [HLNW11] represent local flow direction over time. In contrast, (c) pathlines and (d) pathline glyphs provide a Lagrangian view on the flow. The lower row ((e)–(h)) shows the respective images for an increased seeding resolution.*

one for the vertex coordinates and one for additional quantities (integration time, velocity etc.) stored as texture coordinates for every vertex. Color mapping is applied in the fragment shader using this information. Hence, the user can switch between different color mappings without recomputation. For the zoom lens, the respective part of the frame buffer is overwritten with the rendering for zoomed-in camera settings. Drawing the glyphs with anti-aliasing improves the visibility of patterns at overview scales but drastically reduces the frame rate. Therefore, anti-aliasing is only applied when there currently is no user interaction.

Table 1 shows benchmark results for some images of this paper. The seeding resolution applies to the referenced figure and the viewport resolution was 1920×1200. GPU memory covers all used memory of the GPU including memory for the dataset, frame buffer, and operating system. The internal functionality of OpenGL was used for anti-aliasing (AA).

6. Comparisons and Examples

We applied our method to different unsteady 2D flow fields. The first dataset ("Hotroom", 101×101 resolution, 321 time steps) represents the simulated buoyant flow in a closed room with two obstacles and heated bottom and cooled top (Figure 9). In the following, we compare our method with other visualization techniques and exemplify temporal analysis of flow with this dataset. Afterward, we demonstrate our technique for two other qualitatively different datasets. Finally, we compare it with a pathline selection method.

6.1. Comparison to Other Visualization Techniques

In flow visualization, glyphs typically represent local flow direction (Figure 9(a) and 9(b)). Being based on pathlines (Figure 9(c)), our glyphs (Figure 9(d)) provide a complementary Lagrangian view and allow for the analysis of transport by the flow. With increasing seeding resolution, visual-

Table 1: *Benchmark results.*

dataset	seeding res.	GPU mem.	vertex count	glyph comp.	display rate [fps]	
		[MB]	[mio]	[ms]	with AA	w/o AA
Hotroom (Figure 10(a))	256×256	1,686	40	205	0.22	19
Kármán (Figure 12(d))	101×101	1,153	10	63	2.90	90
Obstacles (Figure 13(b))	410×256	1,642	40	234	0.28	19

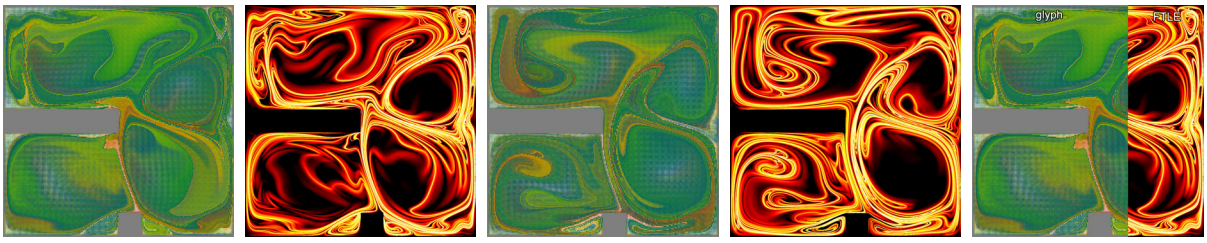


Figure 10: *Comparison with the FTLE for the Hotroom dataset. Pathline glyphs were computed in (a) forward and (c) backward time (seeding resolution 256×256). The FTLE fields (resolution 2048×2048—(b), (d)) for the same integration parameters show consistent structures. (e) The combined image of (a) and (b) shows that the shape and position of the structures match.*

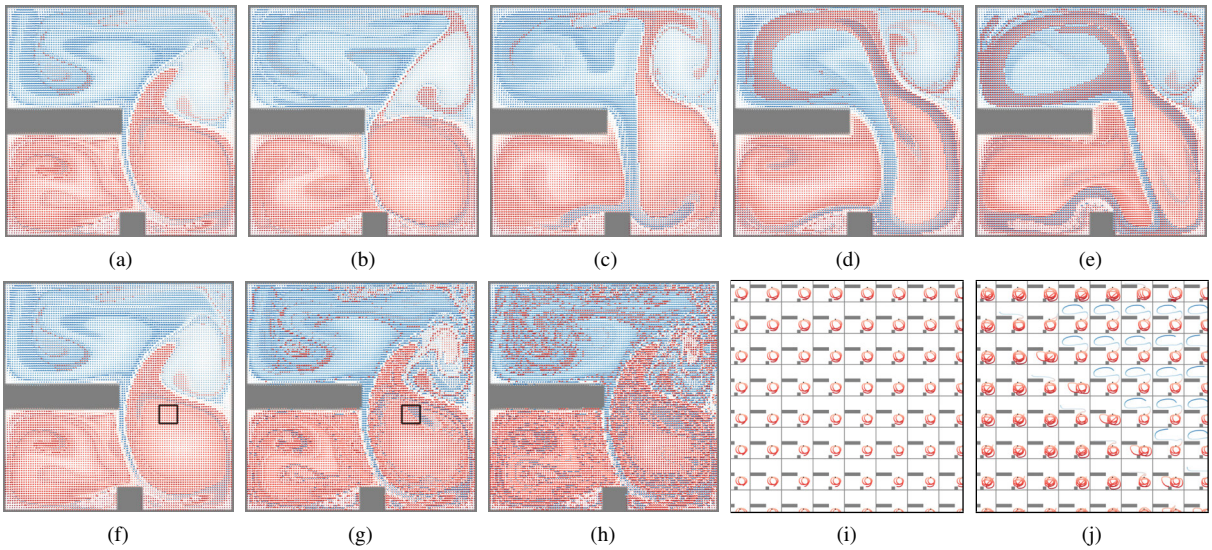


Figure 11: *Temporal analysis of the Hotroom dataset with glyphs for backward time. Spatial color coding is used (Figure 8(b)). Figures (a)–(e) cover the same time span (0.1) for different starting times (0.500, 0.525, 0.550, 0.575, 0.600). Figures (f)–(h) have the same starting time (0.5), but increasing integration time (0.1, 0.2, 0.3); the marked areas are shown in Figures (i)–(j).*

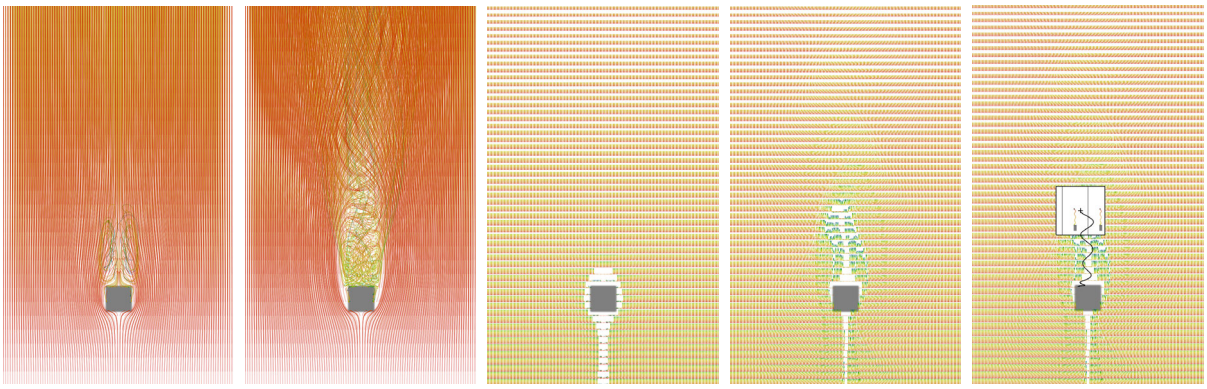
ization of pathlines typically becomes more cluttered (Figure 9(g)), while patterns emerge in the glyph-based visualizations (Figures 9(e), 9(f), and 9(h)). However, only the patterns in the pathline glyph visualization represent structures from transport processes in the flow.

6.2. Relationship to FTLE

Such structures are visible in densely seeded pathline glyphs on the overview scale. A comparison with the structures in the finite-time Lyapunov exponent [Hal01] (FTLE) fields is shown in Figure 10. Many of the visible structures correspond in shape and position (Figure 10(e)). Using temporal color coding results in different colors for pathlines of different length. Neighboring pathlines of different length also correspond to separation and therefore high FTLE values. Hence, an abrupt change in color of neighboring glyphs typically corresponds to high FTLE values. The similarity between the FTLE field and pathline glyphs can be understood from their principal computational approaches: both techniques focus on Lagrangian transport. However, FTLE only takes into account the maximum deviation of end positions, while pathline glyphs show the full trajectory. Therefore, some differences already occur at the overview scale. Of course, for more detailed views, the pathline glyphs become very different, as demonstrated in the other examples.

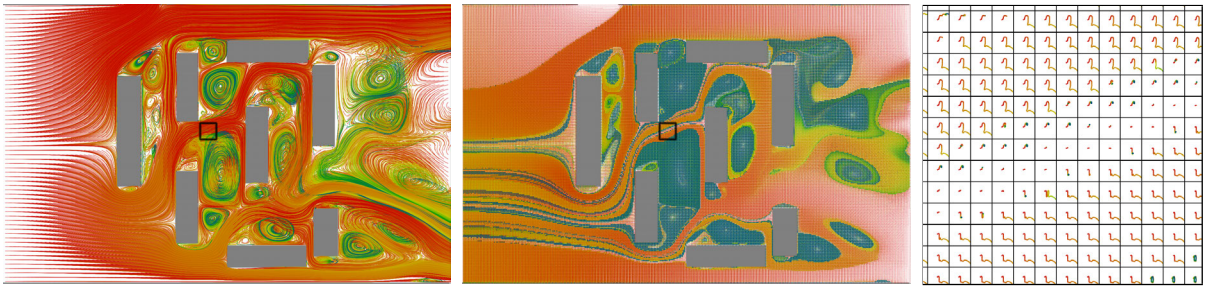
6.3. Temporal Analysis

By changing the starting time and time range of pathline glyphs, a temporal analysis of flow behavior is possible. We demonstrate this for heat transport in the Hotroom dataset (Figure 11). By using pathline glyphs for backward time and spatial color coding (Section 4.3), we can see from and



(a) (b) (c) (d) (e)

Figure 12: *Results for the Kármán dataset. Classic pathline visualization (seeding resolution 101×101) for two different starting times ((a): 0.0, (b): 0.2) and fixed integration range (0.2). The respective pathline glyphs are shown in Figures (c) and (d). (e) The interactive zoom lens and pathline seeding (Section 4.2) support detailed flow analysis on the overview scale.*



(a) (b) (c)

Figure 13: *Visualization of the full time range of the Obstacles dataset. (a) Pathlines from other areas hide the behavior of the particles started in the marked area (black frame). (b) Pathline glyphs show that the behavior there is not uniform. (c) Zooming into this area confirms that there are different groups of particles with similar paths in the flow.*

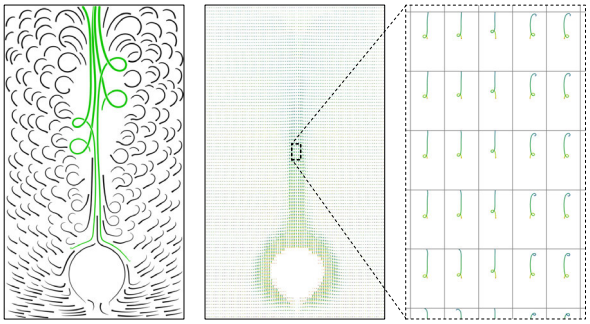
through which zones particles flow. By changing the starting time (Figures 11(a)–11(e)), we can analyze how the mixing of particles from both zones changes over time. Initially, most particles stayed only in one zone for most of the time. This is visible in their clearly separable color. At the end of the examined time range, the number of particles passing both zones increased considerably, visible in the mixing of both colors. With increasing time range (Figures 11(f)–11(h)), the probability that particles travel through both zones increases. This is also visible in the glyph shape (Figure 11(j)): some of the particles staying in the whirl in the lower zone came originally from the upper zone.

6.4. Von Kármán Vortex Street

In the "Kármán" dataset (101 × 301 spatial resolution, 801 time steps), flow from the bottom to the top passes a single obstacle in the lower part of the domain (Figure 12). Figure 12 shows a comparison of classic pathline visualization and pathline glyphs. Traditional pathlines (Figures 12(a) and 12(b)) provide an impression of the general flow behavior. However, it is difficult to determine the region of the flow influenced by the obstacle. The pathline glyphs (Figures 12(c) and 12(d)) show that the influenced area considerably increased for the second starting time and lost its symmetry. They also provide hints for the interactive exploration (Figure 12(e)): the small glyphs above the obstacle typically correspond to particles traveling against the main flow direction toward the obstacle, as the interactively seeded pathline in combination with the zoom lens shows.

6.5. Obstacles

The "Obstacles" dataset (Figure 13) has a resolution of 410 × 256 and 64 time steps. Flow from the left to the right passes several obstacles. There is an open boundary on the right and a small outflow at the upper boundary. Due to low temporal variation, this dataset is quite "pathline-friendly", as Figure 13(a) shows. There is not much visual clutter and the traditional pathline visualization provides a



(a) (b) (c)

Figure 14: *Comparison to pathline selection. Figure (a) was generated with the method by Weinkauf et al. [WTS12], which tries to select and display only pathlines that do not intersect with others. (b) Pathline glyphs do not require any selection algorithm and can densely cover the domain. (c) Close-up of the marked area.*

good overview of the flow behavior. However, even in this case, pathlines from other areas can hide the behavior in certain areas, as the comparison with our glyphs shows (Figure 13(b)). In the marked area, pathlines starting at the left domain boundary occlude the pathlines starting there. The glyphs reveal that the shape of the pathlines starting there is not uniform, but there are several groups of similar pathlines. The zoomed-in image confirms this (Figure 13(c)). Two groups of pathlines reaching the right domain boundary are separated by pathlines that stop inside the domain.

6.6. Pathline Selection

Pathline selection is one approach to reduce visual clutter in pathline visualizations (see Section 2). We therefore compare our pathline glyphs with the pathline selection method by Weinkauf et al. [WTS12]. We applied our method to one of the datasets used in their paper (Figure 14). The dataset contains the flow above a heated cylinder and has a spatial resolution of 41×70 and 241 time steps.

Their result (Figure 14(a)) contains only a few intersecting pathlines and has therefore low visual clutter; it provides a good impression of major pathline structures. However, due to the long pathlines (green lines) starting at the bottom, the upper part of the center area has very low pathline density. In contrast, pathline glyphs (Figure 14(b)) can cover the complete domain with equal density. Two major areas can be identified: the area around and above the cylinder contains long pathlines, while the pathlines in the remaining part of the domain are comparably short. However, zooming-in is required to see the individual shape of the pathlines (Figure 14(c)). For example, the asymmetry of the center area is visible: the pathlines on left side reach the domain boundary, while the ones on the right exhibit an additional whirl.

7. User Study

We conducted a qualitative user study with four domain experts from different departments of our university. They had no previous knowledge of pathline glyphs and were not involved in their development. All of them have good knowledge of computational fluid dynamics and work regularly with flow data and visualizations thereof. Classic pathline visualization with an additional, interactively controllable pathline and our pathline glyphs with the discussed interaction techniques (Section 4.2) were compared. Each session took between 40 and 50 minutes. The think-aloud method was used and the participants had to fill out a questionnaire afterward. They were asked to rate the methods on a Likert scale ranging from 1 (very helpful) to 5 (not helpful).

The participants were asked to explore three datasets and perform certain tasks: For the Kármán dataset (Figure 12), they should narrow down the area influenced by the obstacle and the area where particles flow back against the obstacle. In the Obstacles dataset (Figure 13), they should identify regions where particles end in vortices. And for the Hotroom dataset (Figure 9), they were asked to explore heat transport and mixing behavior. The average ratings for the three datasets and a general rating of both methods are shown in Table 2. The table indicates that the ratings of pathline glyphs are more robust with respect to the dataset. The suitability of the method seems to rely less on the analyzed dataset. Additionally, the participants had to decide which method was more suitable to solve the tasks. In all cases, all participants rated pathline glyphs as better suitable. However, because of the low number of participants, statistical inference is not possible and the study results can provide only an initial indication for the usefulness of pathline glyphs.

The experts had similar justifications for preferring pathline glyphs: The glyphs provide a clearer overview and visually separate the flow into different areas, offering a good basis for exploration. However, the experts emphasized the importance of interaction. To understand the flow structures, they think it is important to have the user-controllable pathline. All experts would like to use pathline glyphs in practice and confirmed that they may help accomplish their work more efficiently. Some were surprised by the clear structures visible with pathline glyphs. Others were distracted by the zoom lens and preferred to only use the interactively controllable pathline together with the pathline glyphs.

Table 2: *User study results. Average ratings with standard deviation in brackets.*

	pathlines	glyphs
Kármán	1.75 (0.83)	1.0 (0.0)
Obstacles	3.25 (0.83)	1.0 (0.0)
Hotroom	4.25 (0.43)	2.0 (0.71)
general	3.0 (1.0)	1.25 (0.43)

8. Conclusion and Future Work

We introduced pathline glyphs as a means of visualizing time-dependent flow using pathlines with reduced visual clutter. The results show that they provide insight into additional aspects of the flow compared to the direct visualization of pathlines. The flow domain can be densely covered, which allows for visual segmentation on the overview level. The presented interaction techniques facilitate the detailed exploration in small areas without losing the context.

Our method is a simple, easy-to-implement, and easy-to-use approach. Due to its simplicity and because it builds on well-established concepts and involves no parameters with implicit impact, it is straightforward to extend existing systems with our approach: the central step is the proper scaling of the individual pathlines.

Since it is difficult to use pathline glyphs for 3D fields, future work can target improvements for 3D applications. Furthermore, pathline glyphs are not restricted to uniform seeding. Seeding density may be locally varied to better cover the flow structure or clustering methods can be applied to pathline glyphs. Finally, the presented concept can be applied to other types of integral curves like streaklines.

Acknowledgments

The authors would like to thank the German Research Foundation (DFG) for financial support of the project within the *Cluster of Excellence in Simulation Technology* (EXC 310) at the University of Stuttgart. We thank Tino Weinkauf for the dataset and image for the comparison in Figure 14.

References

[BKC*13] BORGO R., KEHRER J., CHUNG D. H., MAGUIRE E., LARAMEE R. S., HAUSER H., WARD M., CHEN M.: Glyph-based visualization: Foundations, design guidelines, techniques and applications. In *Eurographics State of the Art Reports* (2013), pp. 39–63. 2

[dLvW93] DE LEEUW W. C., VAN WIJK J. J.: A probe for local flow field visualization. In *Proceedings of IEEE Visualization* (1993), pp. 39–45. 2

[FSf*10] FALK M., SEIZINGER A., ÜFFINGER M., SADLO F., WEISKOPF D.: Trajectory-augmented visualization of lagrangian coherent structures in unsteady flow. In *Proceedings of International Symposium on Flow Visualization* (2010). 2

[GRT13] GÜNTHER T., RÖSSL C., THEISEL H.: Opacity optimization for 3D line fields. *ACM Transactions on Graphics* (2013), 120:1–120:8. 2

[Hal01] HALLER G.: Distinguished material surfaces and coherent structures in three-dimensional fluid flows. *Physica D: Nonlinear Phenomena 149*, 4 (2001), 248–277. 7

[HLNW11] HLAWATSCH M., LEUBE P., NOWAK W., WEISKOPF D.: Flow radar glyphs—static visualization of unsteady flow with uncertainty. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (2011), 1949–1958. 2, 6

[KML99] KIRBY R. M., MARMANIS H., LAIDLAW D. H.: Visualizing multivalued data from 2D incompressible flows using concepts from painting. In *Proceedings of IEEE Visualization* (1999), pp. 333–340. 2

[LEG*08] LARAMEE R. S., ERLEBACHER G., GARTH C., SCHAFHITZEL T., THEISEL H., TRICOCHE X., WEINKAUF T., WEISKOPF D.: Applications of texture-based flow visualization. *Engineering Applications of Computational Fluid Mechanics 2*, 3 (2008), 264–274. 2

[LHD*04] LARAMEE R. S., HAUSER H., DOLEISCH H., VROLIJK B., POST F. H., WEISKOPF D.: The state of the art in flow visualization: Dense and texture-based techniques. *Computer Graphics Forum 23*, 2 (2004), 203–221. 2

[MJL*13] MCLOUGHLIN T., JONES M., LARAMEE R., MALKI R., MASTERS I., HANSEN C.: Similarity measures for enhancing interactive streamline seeding. *IEEE Transactions on Visualization and Computer Graphics 19*, 8 (2013), 1342–1353. 2

[MLP*10] MCLOUGHLIN T., LARAMEE R. S., PEIKERT R., POST F. H., CHEN M.: Over two decades of integration-based, geometric flow visualization. *Computer Graphics Forum 29*, 6 (2010), 1807–1829. 2

[PGL*12] PENG Z., GRUNDY E., LARAMEE R., CHEN G., CROFT N.: Mesh-driven vector field clustering and visualization: An image-based approach. *IEEE Transactions on Visualization and Computer Graphics 18*, 2 (2012), 283–298. 2

[PPF*11] POBITZER A., PEIKERT R., FUCHS R., SCHINDLER B., KUHN A., THEISEL H., MATKOVIC K., HAUSER H.: The state of the art in topology-based visualization of unsteady flow. *Computer Graphics Forum 30*, 6 (2011), 1789–1811. 2

[PW13] PILAR D., WARE C.: Representing flow patterns by using streamlines with glyphs. *IEEE Transactions on Visualization and Computer Graphics 19*, 8 (2013), 1331–1341. 2

[SGSM08] SALZBRUNN T., GARTH C., SCHEUERMANN G., MEYER J.: Pathline predicates and unsteady flow structures. *Visual Computer 24*, 12 (2008), 1039–1051. 2

[SK10] SCHULTZ T., KINDLMANN G.: Superquadric glyphs for symmetric second-order tensors. *IEEE Transactions on Visualization and Computer Graphics 16*, 6 (2010), 1595–1604. 2

[TMWS13] TAO J., MA J., WANG C., SHENE C.-K.: A unified approach to streamline selection and viewpoint selection for 3D flow visualization. *IEEE Transactions on Visualization and Computer Graphics 19*, 3 (2013), 393–406. 2

[Tuf90] TUFTE E. R.: *Envisioning Information*. Graphics Press, 1990. 2

[Tuf06] TUFTE E. R.: *Beautiful Evidence*. Graphics Press, 2006. 2

[WJE00] WESTERMANN R., JOHNSON C., ERTL T.: A level-set method for flow visualization. In *Proceedings of IEEE Visualization* (2000), pp. 147–154. 3

[WTS12] WEINKAUF T., THEISEL H., SORKINE O.: Cusps of characteristic curves and intersection-aware visualization of path and streak lines. In *Topological Methods in Data Analysis and Visualization II*, Peikert R., Hauser H., Carr H., Fuchs R., (Eds.). Springer, 2012, pp. 161–175. 2, 9

[YWSC12] YU H., WANG C., SHENE C.-K., CHEN J.: Hierarchical streamline bundles. *IEEE Transactions on Visualization and Computer Graphics 18*, 8 (2012), 1353–1367. 2

[ZWZ*13] ZHANG W., WANG Y., ZHAN J., LIU B., NING J.: Parallel streamline placement for 2D flow fields. *IEEE Transactions on Visualization and Computer Graphics 19*, 7 (2013), 1185–1198. 2