Eurographics Conference on Visualization (EuroVis) 2018
J. Heer, H. Leitte, and T. Ropinski
(Guest Editors)

Volume 37 (2018), Number 3

Explorative Blood Flow Visualization using Dynamic Line Filtering based on Surface Features

B. Behrendt[1,2] and P. Berg[1,3] and O. Beuing[1,4] and B. Preim[1,2] and S. Saalfeld[1,2]

[1]Research Campus STIMULATE
[2]Department of Simulation and Graphics, University of Magdeburg, Magdeburg, Germany
[3]Department of Fluid Dynamics and Technical Flows, University of Magdeburg, Magdeburg, Germany
[4]Institute of Neuroradiology, University Hospital Magdeburg, Magdeburg, Germany

Abstract

Rupture risk assessment is a key to devise patient-specific treatment plans of cerebral aneurysms. To understand and predict the development of aneurysms and other vascular diseases over time, both hemodynamic flow patterns and their effect on the vessel surface need to be analyzed. Flow structures close to the vessel wall often correlate directly with local changes in surface parameters, such as pressure or wall shear stress. Yet, in many existing applications, the analyses of flow and surface features are either somewhat detached from one another or only globally available. Especially for the identification of specific blood flow characteristics that cause local startling parameters on the vessel surface, like elevated pressure values, an interactive analysis tool is missing.
The explorative visualization of flow data is challenging due to the complexity of the underlying data. In order to find meaningful structures in the entirety of the flow, the data has to be filtered based on the respective explorative aim. In this paper, we present a combination of visualization, filtering and interaction techniques for explorative analysis of blood flow with a focus on the relation of local surface parameters and underlying flow structures. Coherent bundles of pathlines can be interactively selected based on their relation to features of the vessel wall and further refined based on their own hemodynamic features. This allows the user to interactively select and explore flow structures locally affecting a certain region on the vessel wall and therefore to understand the cause and effect relationship between these entities. Additionally, multiple selected flow structures can be compared with respect to their quantitative parameters, such as flow speed. We confirmed the usefulness of our approach by conducting an informal interview with two expert neuroradiologists and an expert in flow simulation. In addition, we recorded several insights the neuroradiologists were able to gain with the help of our tool.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture I.4.8 [Computer Graphics]: Scene Analysis—Shading

1. Introduction

For the study of vascular diseases, such as aneurysms, both morphological features and hemodynamic parameters as well as their complex interaction need to be evaluated. More precisely, physicians are often interested in exploring blood flow patterns that cause specific hemodynamic features, such as changes in pressure or wall shear stress, on the vessel wall. Studies have shown that these hemodynamics correlate with the rupture of aneurysms and are therefore vital for risk assessment [CDM*17,DCHS*17]. Since both the rupture as well as the treatment procedure can lead to severe consequences for the patient, improved risk assessment helps to optimize patient-specific treatment plans.

A common and accepted visualization for cerebral blood flow is to display the vessel morphology as a 3D model and convey flow patterns through either stream- or pathlines or map hemodynamic parameters directly onto the surface using a color scale. Pathlines are often filtered by their parameters, such as velocity magnitude or vorticity, to prevent occlusion. However, these filters require a general idea of which flow structures the user expects to find. For an explorative approach, where the physician wants to figure out what kind of flow causes a specific phenomenon on the vessel surface, their usefulness is limited.

In this paper, we present a set of techniques to interactively select and filter flow structures based on their effect on the vessel wall. Our approach combines the tasks of parameter visualization and pathline selection to create an intuitive and robust tool for explorative pathline filtering. By selecting regions on the vessel surface with hemodynamically interesting parameter values, such as local extrema in pressure or wall shear stress, underlying flow structures are automatically highlighted. These highlights can then be further refined by applying filters based on parameters such as velocity, pressure or residence time.

Our work was designed in cooperation with an experienced neuroradiologist to identify complex interactions between hemodynamic parameters in general. The user's goal is to find flow structures correlating to medically interesting surface features. A general workflow would therefore involve finding those surface features, selecting a subset of them for further investigation and extracting flow structures related to the selected features. Based on this workflow, we identified the following key requirements for our application in cooperation with our clinical partners:

Req. 1 It should support finding medically interesting surface regions through the visualization.
Req. 2 The user should be able to easily select one or more surface regions and explore the associated local flow structures.
Req. 3 To support exploration, the user should have control about what kinds of flow structures are extracted.
Req. 4 The flow structures extracted from each selected feature should be visually distinguished.
Req. 5 The user should be able to further explore the extracted flow structures based on their own features.

To evaluate our work, we specifically selected nine datasets, which yielded contradicting results using conventional analysis. The evaluation was carried out with two expert neuroradiologists and an expert of flow simulation. One of the neuroradiologists was involved in the design of our application, whereas the other is completely independent. In the evaluation section, we provide our exploration results for these datasets as well as an informal qualitative evaluation.

During the evaluation, we were able to show that our approach allowed for a systematic exploration and quantitative assessment of flow structures in aneurysms. Interesting structures such as vortices could be reliably detected and comprehensibly visualized, allowing the user to gain insights into the flow patterns both on a local and global scale.

2. Related Works

In this section, we discuss previous work on the visual exploration of medical flow with a focus on, but not restricted to, blood flow. Visual clutter is a main problem in 3D visualizations including stream- or pathlines. Interesting structures, such as vortices, are often hidden within other, more laminar flow. Therefore, a variety of methods have been developed to automatically or semi-automatically highlight important flow structures.

Van Pelt et al. introduced an implicit filtering technique by interactively selecting vessel cross sections as seeding planes [vBB*10]. By positioning these planes the user can gain insights into global flow patterns, such as splitting flow. However, focusing on specific flow structures within these patterns beyond color-coding the flow velocity is not possible.

Gasteiger et al. presented a focus-and-context visualization technique that allows exploring blood flow directly beneath an interesting area on the vessel surface [GNBP11]. By positioning an elliptical "lens" in screen-space, the user can clip away the vessel surface and reveal underlying flow. However, this is limited to the flow directly underneath the selected surface area. There is no way to trace the revealed pathlines to other interesting areas or even through the entire vessel. Additionally, since the lens is placed in screen-space, camera movement may result in an undesired change of the focal region.

Another approach of filtering lines based on screen-space was realized by Lee et al. [LMSC11]. They employed a filtering technique based on screen-space entropy and occlusion to determine whether a pathline should be shown. The calculated screen-space entropy can also be used to determine an optimal viewpoint. A drawback of this approach in an explorative scenario is its lack of real-time capabilities. Since the entropy is determined in screen-space, it has to be recalculated after each change of perspective.

Lawonn et al. used an automatic cut-away technique where the vessel surface is always removed when occluding any pathlines [LGV*16]. This allows for a simultaneous visualization of blood flow and parameters on the vessel wall, such as thickness. Since the vessel is usually completely filled with pathlines, the lines have to be animated. While this reduces the amount of surface area that needs to be culled, it also prevents the entire flow course from being visible at one time.

Oeltze et al. used clustering to reduce visual clutter in simulated cerebral bloodflow data [OLK*14]. Pathlines were clustered based on their geometry or attributes and visualized by a single representative for each resulting cluster. In a later publication, Oeltze et al. added dynamic seeding capabilities to better convey complex flow structures such as embedded vortices [OJCJP16]. While both approaches highlight existing flow patterns in a dataset, the reduction of each of these patterns to a single line may fail to capture its full structure and shape as well as its relation to the vessel surface.

Instead of highlighting all flow structures present in a cerebral flow dataset, Gasteiger et al. presented a method to specifically extract and visualize the inflow jet and impingement zone in aneurysms [GLv*12]. Van Pelt et al. extended this approach to allow for a comparative visualization of different stent configurations in the same aneurysm [vGL*14]. The inflow jets of all configurations are displayed simultaneously using multiple colored arrows. Glyphs are employed for the visualization of the impingement zones from all configurations. Since the resulting visualization is tailored to a very specific application, it is not suitable for a more general, explorative approach.

Zachow et al. used information visualization techniques to explore nasal airflow data [ZMH*09]. By linking a volume visualization with brush-based selection in parallel coordinates and scatterplots, the user is able to interactively highlight interesting parameter combinations, such as temperature or velocity. Our method allows for parameter selection based on scatterplots or parallel coordinates as well. However, the filtering capabilities of this approach are limited to parameters and cannot take the spatial position of certain flow phenomena into account.

Salzbrunn et al. introduced pathline predicates, a method of grouping vertices in a pathline based on their fulfillment of user-defined criteria [SS06]. Multiple predicates can be combined using Boolean algebra to create more complex criteria. Born et al. adapted line predicates to support the exploration of cardiac blood flow by designing a set of pre-defined predicates [BPM*13]. Users could adapt and combine these predicates to suit their specific needs for exploring cardiac flow data. In addition to predicates based on flow structures, such as vorticity or velocity, they also added a region-based predicate that would detect flow that passes by or originates in a certain anatomical area. However, these predicates were designed to work based on entire anatomical regions, such as a specific heart chamber. Therefore, the ability to assess local correlations between surface features and flow structures is restricted. A similar method for 4D PC-MRI data was presented by Broos et al. [BHK*16]. They employed a user-defined transfer function to determine possible seed points for pathlines from a flow field. In conjunction with a mostly automatic surface visualization, this alleviates the need for a segmentation of the underlying data. Köhler et al. implemented line predicates to extract vortices from cardiac blood flow data [KGP*13]. Although these vortices are reliably detected, this approach only allows for global filtering. Specifying a region of interest or focusing on only a single vortex that may correspond to a surface feature is not possible.

Meuschke et al. presented a combined visualization of hemodynamic flow and vessel surface information with a focus on near-wall flow [MVB*17]. To prevent occlusion, the vessel surface is mapped onto a 2D plane and displayed alongside the 3D visualization. This approach focuses on correlating multiple surface parameters, such as wall shear stress and wall thickness. Although wall-near flow is visualized as context information, there is no way to extract flow bundles directly corresponding to interesting local surface features.

A surface-based filtering approach was realized by Neugebauer et al. [NLB*13]. They employed an automatic detection of potentially interesting surface regions based on the surface geometry, although manual selection of a region is also possible. Instead of filtering existing pathlines according to their distance to the selected region of interest, they dynamically seed new lines close to the region. The generated lines are then classified based on a 2D representation. Further filtering of such a line bundle is not possible.

3. Medical Background

Cerebral aneurysms are pathologic dilations of brain-supplying arteries bearing the risk of rupture. Aneurysm rupture is mostly accompanied with subarachnoid hemorrhages that may cause fatal consequences for the patients. The treatment options comprise endovascular intervention and neurosurgical clipping. However, both endovascular and surgical therapy can cause severe complications. In the case of small aneurysms, the complication rate may even exceed the rupture rate [Wie03]. To minimize the risk for the otherwise healthy patients, careful pre-treatment assessment of the rupture risk is mandatory. In clinical practice, the most important rupture risk factors are the type of aneurysm (i.e., asymptomatic or symptomatic), age, sex, and aneurysm size and position [WvdSAR07]. Furthermore, morphological parameters, e.g., irregular shape, orientation and diameter [LEBB09] were correlated with rupture risk. However, the study of aneurysm-specific hemodynamics with computational fluid dynamics (CFD) plays an increasing role [XTSM14].

Recent studies reported a correlation between certain hemodynamic information (e.g., concentrated inflow jets, small impingement regions and increased wall shear stress) and prior aneurysm rupture [CDM*17, DCHS*17]. However, the clinical application is limited by the lack of an exploration tool that allows a systematic analysis of the complex intra-aneurysmal flow or a detailed, objective and reproducible correlation of qualitative and quantitative parameters. To understand local blood flow phenomena and their impact on local characteristics like pressure or wall shear stress, an adapted exploration technique, as presented in this paper, is required.

4. Pre-Processing

In this section, we provide a brief overview about the data acquisition and computational fluid dynamics simulation. Afterwards, we will discuss what parameters are derived from the simulation data.

4.1. Extraction of Simulated Blood Flow

The patient-specific datasets that are considered within this study were acquired using 3D digital subtraction angiography on an Artis Q angiography system (Siemens Healthcare GmbH, Forchheim, Germany). Segmentation was performed using a threshold-based segmentation with the open-source software MeVisLab 2.7 (MeVis Medical Solutions AG, Bremen, Germany). To account for multiple aneurysms, large vascular domains were considered. Hence, small artifacts such as melted vessels or holes, which occurred during the segmentation process, were manually removed on a sub-voxel level [GBNP15].

Before the hemodynamic simulations were carried out, each dataset was spatially discretized using tetrahedral as well as polyhedral elements. In order to account for the occurring velocity gradients, particularly close to the vessel wall, an appropriate grid size of $\Delta x = 0.1\ mm$ was chosen [JBB*13]. This resulted in a number of elements ranging from 5.3 to 8.9 million depending on the domain size.

The subsequent blood flow simulations were performed using the commercial fluid dynamics solver STAR-CCM+ 11 (Siemens Product Lifecycle Management Software Inc., Plano, TX, USA 75024). Here, the governing equations of mass and momentum conservation were solved. Regarding the boundary conditions, flow measurements of a healthy volunteer using 7T phase-contrast magnetic resonance imaging were applied at each inlet cross section [BSJ*14]. A healthy volunteer provides highly resolved and representative intracranial flow rates, which are adapted depending on the size of the vessel and the locations of interest. Specifically, flow rates are scaled by the corresponding ratio of inflow areas. Acquiring patient-specific measurements is not clinical practice at the moment. Once these measurements are readily available in the future, they can be easily applied as inflow boundary conditions.

All vessel walls were assumed to be rigid, since information about wall thickness and wall properties cannot be extracted from clinical data. The assumption of rigid vessel walls in the context

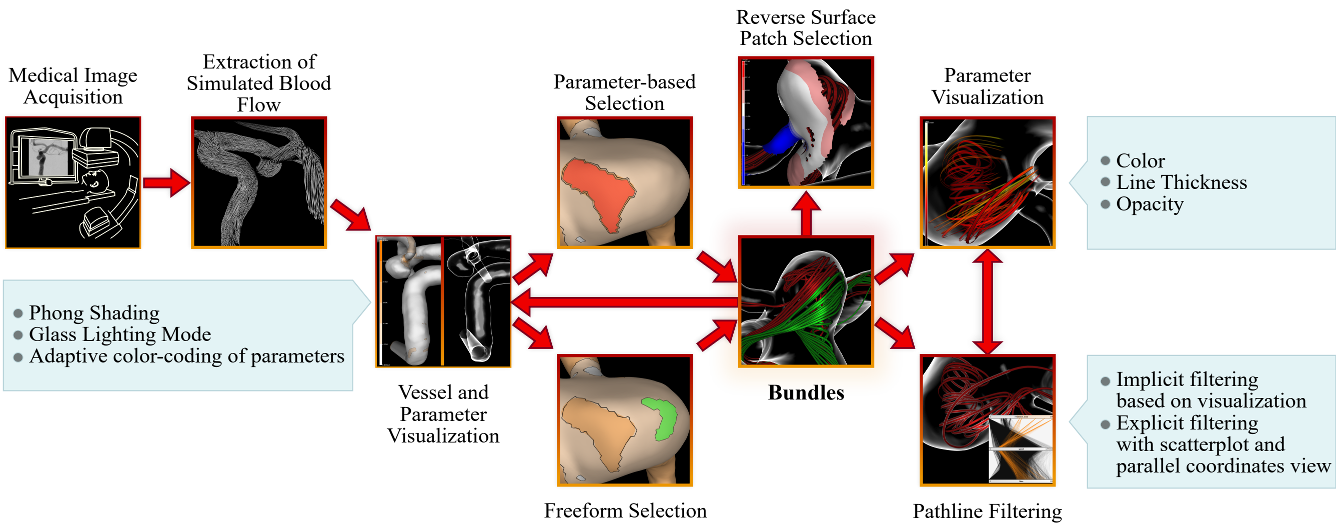

Figure 1: *Workflow for our application starting with the medical image acquisition.*

of cerebral vasculature is commonly used and well accepted. Compared to vessel movements close to the heart (e.g. the Windkessel effects within the aorta), intracranial arteries experience only small radial dilatations. However, cerebral aneurysms can possess differences regarding the local wall thickness and hence their rupture probability. Nevertheless, precise and reliable in vivo wall thickness measurement are not possible with recent imaging modalities.

At each outlet boundary, zero-pressure conditions were applied. Blood was considered as an incompressible ($\rho = 1055\ \frac{kg}{m^3}$), Newtonian ($\eta = 4\ mPa\ \cdot s$) and laminar fluid, which is appropriate in this range of vessel diameters. In total, three cardiac cycles with a time step size of $\Delta t = 0.001\ s$ were considered for each case. This allows for the generation of a periodic solution [BB17]. The analysis commonly considers one representative cardiac cycle. However, initialization effects can occur within the simulation. To avoid this, three cycles are calculated and the first two are discarded.

4.2. Parameter and Pathline Calculation

The hemodynamic simulations result in a time-resolved flow field. Some parameters, such as pressure, velocity or directional wall shear stress on the surface, are already calculated during the simulation. Other parameters, as well as the pathlines themselves, have to be derived.

While the directional wall shear stresses can be directly extracted from the simulated data, physicians are more interested in the wall shear stress magnitude. This value is therefore automatically generated from the directional wall shear stress for each surface vertex when loading a dataset. Since the numeric values of the wall shear stress are not suitable to compare multiple aneurysms in either the same or different datasets, we computed the normalized wall shear stress. This is done by normalizing the wall shear stress magnitude on the aneurysm surface with the average wall shear stress magnitude on the parent vessel.

Here, we also generate the oscillating shear index (OSI), which is a metric quantifying the alignment of the wall shear stress with the average wall shear stress vector over time. It is calculated using the instantaneous shear stress vector wss and the cycle period T, and yields values from 0 (strong alignment) to 0.5 (weak alignment).

$$OSI = \frac{1}{2} \cdot \left(1 - \frac{\left| \int_0^T wss \mathrm{d}t \right|}{\int_0^T |wss| \mathrm{d}t} \right)$$

The pathlines are integrated from evenly distributed seed points on the inlet planes using fourth order Runge-Kutta integration. To sample values from the flow field, we employ Shepard interpolation with 16 samples.

Another clinically important measure is the residence time of blood inside an aneurysm. To calculate this parameter, we performed a manual mesh segmentation by assigning a Boolean value to each surface vertex denoting whether it is part of the aneurysm or not. We then determine for each vertex from every pathline whether it resides inside a segmented aneurysm by searching the closest surface point using a KD tree and checking if that surface point belongs to an aneurysm. Whenever a pathline enters an aneurysm, i.e. the current vertex belongs to an aneurysm while its predecessor does not, we store the vertex ID and current time point of that vertex. On encountering the first vertex that does not belong to the aneurysm, we calculate the temporal difference between the current vertex and the previously stored vertex and assign the resulting residence time to all vertices in that aneurysm. Vertices that do not belong to an aneurysm are assigned a residence time of zero. As a side effect, this parameter makes it easy to filter pathline segments in an aneurysm, as they can be exclusively described as having a non-zero residence time.

5. Surface-based Pathline Filtering

In this section, we will give an overview over the intended workflow for our tool (Fig. 1). As we have already discussed the image acquisition and flow extraction, we will begin with the vessel visualization. Starting from there, the structure is as follows:

1. Visualization of the vessel surface with mapped surface parameters
2. Selection of one or more interesting surface regions
3. Extraction of pathlines related to the selected regions
4. Further filtering of extracted pathlines

For each of these steps, we will provide information about the user's interaction possibilities and explain our design decisions as well as the technical implementation.

5.1. Vessel Visualization

Initially, the user is presented with an empty visualization of the vessel surface. We employ Phong shading with a single headlight to convey the vessel shape. To prevent the surface from occluding the inner flow that the user will eventually add, it is always rendered semi-transparently. The amount of transparency can be configured, but defaults to an empirically determined value of 33%. While reducing the surface opacity does increase visibility of the inner flow, it also decreases the visibility of lighting effects and therefore reduces shape perception. We therefore decided to adopt a different strategy for applying lighting and transparency to the vessel surface that we call "glass lighting mode".

To emphasize the vessel boundaries even at higher transparency settings, we added a Fresnel effect to the lighting. Similar to [GNKP10], the lighting intensity is then multiplied with the vessel opacity for each vertex, although we consider both the Fresnel and the Phong lighting for this. Regions with strong lighting therefore appear more opaque, highlighting the vessel shape and creating an effect similar to looking through a glass bottle. The glass lighting mode is enabled by default, but can be disabled by the user in favor of using traditional Phong-shading with semi-transparent surfaces. A comparison between both modes can be seen in Figure 2.

To prevent visual clutter from overlapping parts of the vessel, the user can set the backfaces of the vessel to be always fully opaque, despite the previously mentioned transparency setting. This is disabled by default to prevent the user from missing details in the flow that otherwise may be hidden. Figure 2 shows an overview of the effect of this setting both in the traditional as well as the glass lighting mode.

To ensure correct image composition despite multiple, overlaying transparent fragments, we employ Order Independent Transparency (OIT) [Thi11]. Instead of rendering fragments directly into a framebuffer and resolving overlays using a depth test, we write their color and depth values into a shader storage buffer using a linked list structure. Fragments with an opacity of 1% or lower are discarded to reduce memory usage and GPU load during composition, as such fragments barely contribute to the visualization. The final image is composed by a separate fragment shader that is applied to a screen-filling quad, effectively being executed exactly once for each pixel on screen. Using the linked list from the previous rendering stage, the shader gains access to all fragments for the pixel and is therefore able to sort them according to depth and to perform appropriate alpha blending. This results in a correctly composed image generated entirely on the GPU without having to perform any pre-processing or ordering on the vertices prior to the rendering step.

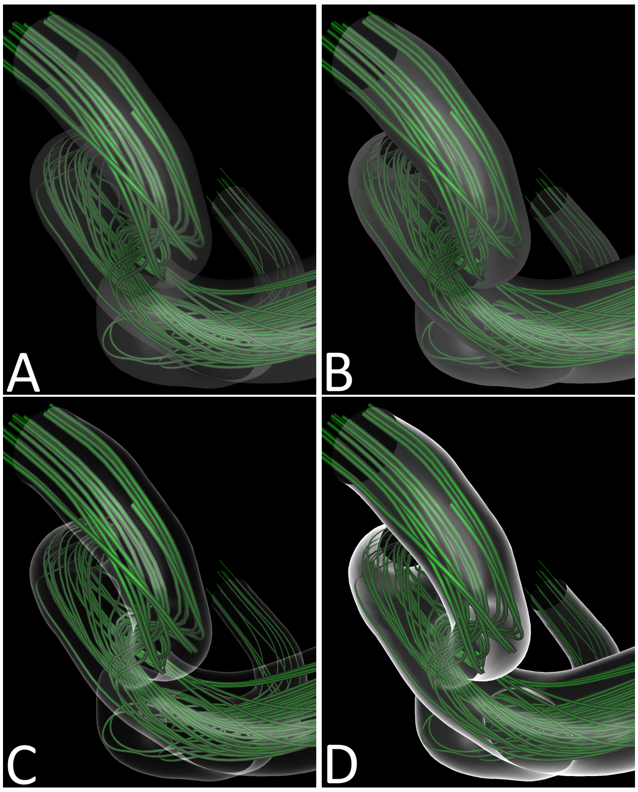

Figure 2: *Different rendering modes for the vessel surface in the pathline viewer; Disabled (A,B) and enabled (C,D) glass lighting, disabled (A,C) and enabled opaque backfaces.*

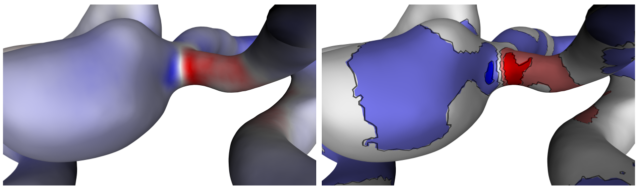

Figure 3: *Comparison of parameter visualization using a smooth (left) and discrete color scale with five shades (right).*

5.2. Parameter Visualization

To add pathlines to the visualization, the user has to select at least one area on the vessel surface based on surface parameters. When the user is selecting these features on the vessel surface, naturally the surface is considered as the focus object. Therefore, it is now rendered fully opaque and allows mapping parameters using a color scale. The glass lighting mode is not available during this selection.

To map the surface parameters onto a color scale, they are uploaded to the GPU as vertex attributes. The fragment shader re-

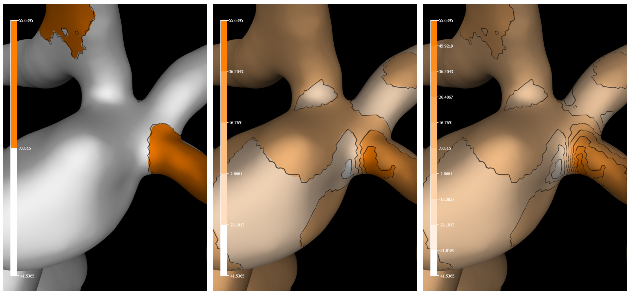

Figure 4: *Wall shear stress visualized on the vessel surface using a color scale with 2 (left), 5 (middle) and 10 (right) discrete shades ranging from white to orange.*

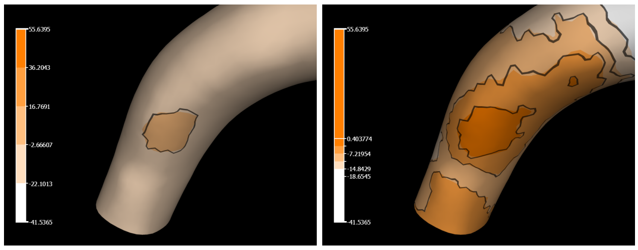

Figure 5: *Comparison of a color scale mapped to the entire parameter range (left) against only mapping the currently visible range of parameter values (right).*

ceives the value range of the active parameter as a uniform variable and calculates the fragment color based on the currently selected color scale. In order to highlight interesting hotspots, which are characterized by local extrema of surface parameter values, we employ discretized color scales. This makes it easier for the user to detect and estimate the size and extent of a hotspot (Fig. 3), therefore fulfilling requirement 1.

The user can freely choose from a set of pre-defined color scales and configure the amount of discrete shades. Figure 4 shows a comparison of different settings for the amount of shades. A higher amount of shades adds more details to the image, but can also lead to a cluttered visualization. To further emphasize the transition between shades, the boundaries are highlighted using black outlines. Our clinical partners were interested in specific parameter ranges, i.e. areas with a normalized wall shear stress value below 20%. To identify these regions, more than five different shades were rarely necessary.

Since the color scale and range settings need to be changed in real-time, the color scale is applied entirely in the fragment shader. Parameter values are normalized to a [0-1] range and uploaded to the GPU as vertex attributes. Settings such as value ranges or the selected color scale are stored as uniform variables for the shader. Therefore, setting changes can be applied without the need to perform any changes to the stored parameter values.

To convey the meaning of the selected color scale, a color legend showing the parameter ranges for each shade is permanently visible on the left side of the image (Fig. 4). Any change to the color scale, the amount of shades or parameter range is reflected on the legend in real time. The selected color scale's domain is initially determined based on the active parameter's value range and can later be adjusted. This is useful in case the parameter value distribution in the dataset does not fully cover the natural range of that parameter or if the user is only interested in a specific sub-range. The adjustment can be performed manually by simply entering new minimum and maximum values, or semi-automatically by basing the scale only on the currently visible surface area instead of the entire parameter range.

Using the latter approach increases the detail dynamically when only a smaller part of the dataset is visible on screen (Fig. 5). To achieve this, the fragment shader responsible for rendering the surface writes the parameter values it encounters into a buffer using atomic min/max operations. However, it may lead to overestimation of parameter differences, since smaller changes in the parameter value may lead to higher differences in the mapped color. To remedy this effect, the color legend will always show the entire parameter range, clearly indicating that the color scale currently only covers a part of the parameter range (Fig. 5). At any point, the user can fixate the current automatically determined range to prevent it from changing as a result of adjusting the camera.

5.3. Surface Patch Selection

When the user clicks on the vessel surface, we determine the vertex closest to the cursor position in screen-space. A simple way to select a feature on the surface would be to place a marker at the position of the closest vertex and then select all adjacent vertices in a specific distance. This type of selection is available in our toolkit, but it is not the default setting. We decided against this approach as the primary method to select patches for several reasons. Using a distance threshold based on user input would add another step to the interaction, which we want to keep as simple as possible, according to requirement 2. It may also lead to confusion whether the distance threshold refers to the distance on the surface (resulting in a circular selection) or in 3D space (resulting in a spherical selection around the marker). Additionally, this type of interaction would limit the user to selecting circular or spherical sections of the surface.

Instead, we decided to allow selecting arbitrary regions on the surface. Unlike the approach by Neugebauer et al. [NLB*13], we decided to base the region selection on hemodynamic instead of geometric features. We presume that a medically interesting region characterized by dissonant geometry would also be characterized by their hemodynamics. Therefore, we derive the selection shape directly from the surface parameter the user has enabled. We determine the color shade of the selected vertex and iteratively search for adjacent vertices with parameter values that would lead to the same color shade, effectively performing a flood fill on the surface (Fig. 6, top middle). Alternatively, the user can choose to also include "higher" or "lower" shades in the selection. This allows selecting arbitrarily shaped regions on the surface using a single click, whereas the parameter visualization itself works as a "selection preview".

We refer to these selections as "patches". It is possible to change

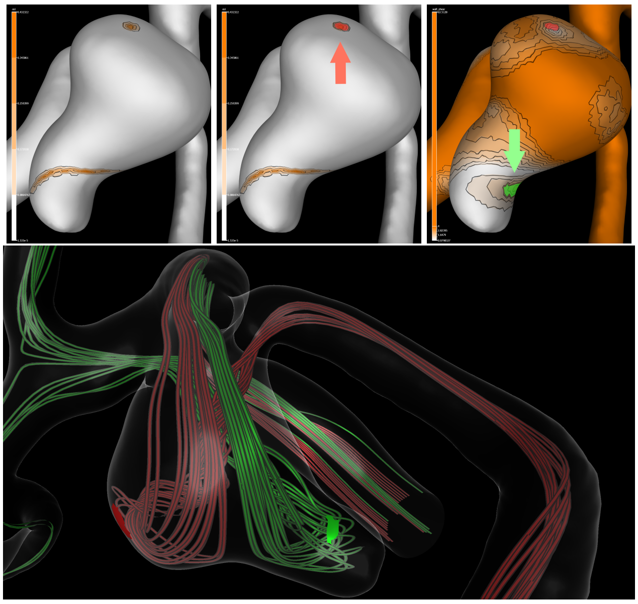

Figure 6: *Workflow of selecting surface patches; surface visualization without any selection (top left), selection of one patch (red arrow), second selection using a different parameter (green arrow); pathlines passing each of the selected patches highlighted with matching colors (bottom).*

the active surface parameter during the selection. Previously created patches will remain, whereas the placement of new patches will be based on the currently active parameter (Fig. 6, top right). We decided against using dynamic surface parameters, as they would likely make both the selection process and interpretation of the results more difficult.

By default, each singular selection creates a new patch. It is also possible to have multiple selections contribute to the same patch, even if the resulting surface is not coherent. If a selection based on surface parameters provides unsatisfactory results, the patch can manually be adjusted by drawing or erasing regions directly on the mesh. Visually, these patches are differentiated using pre-defined colors. Since the association of vertices to their respective patches is stored as flags in a 16 bit integral vertex attribute, it is possible for different patches to overlap. The total number of patches is therefore limited to 16. We assume this technical limitation to be unproblematic, as there would rarely be a situation where the user would need to create more than 16 different patches.

5.4. Pathline Visualization

After the user completes the selection of interesting surface regions, they have the option to select a distance from which to extract pathlines representing the associated blood flow. Only pathlines that come closer to the selected vertices than the distance threshold at least once during their course are selected. The chosen distance can be changed at any time, causing the pathline extraction to be repeated. The extraction is done by building a KD tree from the surface patch vertices and calculating the shortest distance for each pathline vertex from this patch. Only pathlines where at least one vertex is within the selected distance to the surface patch is included in the line bundle associated with that patch. We decided to use pre-integrated streamlines instead of dynamically seeding new lines close to the selected path due to the explorative nature of our application. Filtering existing pathlines only takes a few seconds, whereas creating new pathlines with dynamic seeding would take significantly longer.

If the user has selected multiple patches on the surface, the distance threshold can be configured individually for each patch. The resulting pathline bundles are colored according to the patches they belong to (Fig. 6, bottom), making them visually distinct (recall requirement 4). Pathlines belonging to different patches can be individually configured, such as by toggling their visibility or mapping parameters to their color, thickness or opacity. The patches themselves are visible on the vessel surface by default, but can also be individually hidden or rendered semi-transparently. This is useful for extensive patches that may otherwise create occlusions.

The pathlines are drawn as lines, then converted into view-aligned quads using the geometry shader. This allows the adjustment of the line on a per-vertex basis and also circumvents OpenGLs limitations on line width. Alongside each vertex, we store the integration time point as a vertex attribute. This allows animating the flow by mapping the temporal distance of the time point stored for each vertex with the current animation time point to opacity. The temporal range for which vertices are visible can be adjusted by the user.

5.5. Pathline Filtering

To further refine the previously selected lines, pathline bundles can be filtered based on their parameters, such as pressure or velocity. This gives the user the ability to restrict the visualization of a previously extracted line bundle to a certain combination of features (requirements 3 and 5).

One way of filtering the bundles is to map their hemodynamic parameters to line thickness or opacity, effectively reducing visibility of lines with certain high or low parameter values. By mapping the residence time of a pathline in an aneurysm to opacity and thickness, for example, it is possible to highlight pathlines that stay inside the aneurysm for a larger amount of time (Fig. 7).

Instead of implicitly filtering pathlines using thickness or opacity, the user can explicitly select parameter ranges in a scatterplot or parallel coordinated view of the current pathline bundle. The scatterplot displays two parameters from the currently selected pathline bundle and allows the user to draw a selection rectangle. To allow filtering based on more than two parameters at the same time, we included a parallel coordinates diagram. The user can select which parameters are shown in this diagram and change their order. For each enabled parameter, they can interactively specify a range to filter pathline vertices.

Both the scatterplot and parallel coordinates diagram are synchronized. When the user performs a range selection on one parameter in a diagram, the selection is propagated to the other. In the parallel coordinates diagram, all lines belonging to a selected vertex are highlighted.

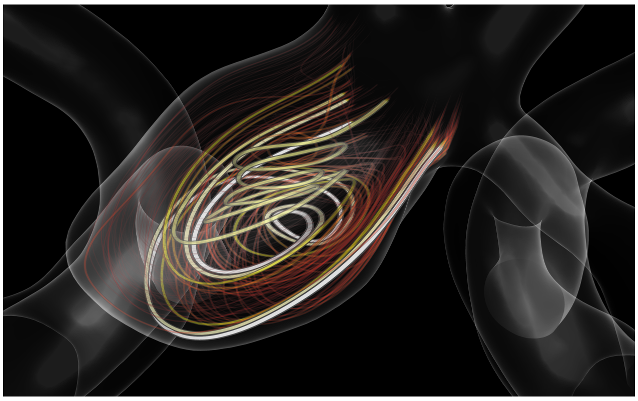

Figure 7: *Visualization highlighting long-residing flow in an aneurysm by mapping residence time on line width, opacity and color (temperature scale) at the same time.*

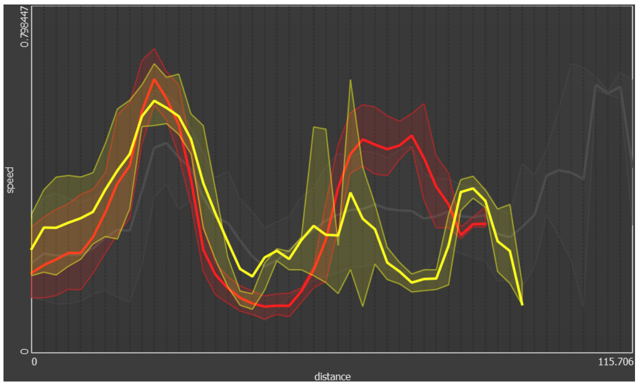

Figure 9: *Line chart comparing two pathline bundles with the global set of pathlines with respect to their speed and flow distance.*

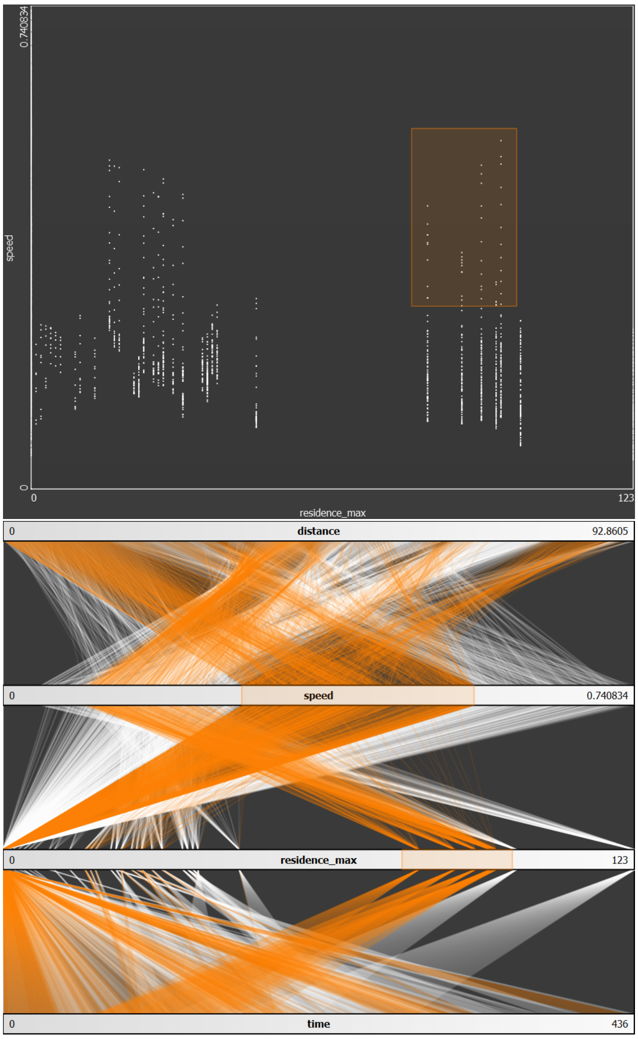

Figure 8: *Scatterplot and parallel coordinates view showing three parameters of the same line bundle with synchronized selection.*

To quantitatively compare two pathline bundles, they can both be plotted in a line chart (Fig. 9). Here, one parameter of each bundle (such as speed) is plotted either over time or flow distance with respect to their average, minimal and maximal value as well as their 25% quantile, median and 75% quantile. Each of these metrics can be individually toggled by the user. Figure 9 shows the area between the average speed as well as the 25% and 75% quantile of two line bundles (red and yellow) and the complete set of pre-integrated pathlines (grey) plotted over the flow distance. The graphs use the same color as the pathlines in the 3D visualization.

Once the user has performed a selection in any of the diagrams, the 3D view will be updated accordingly. There are several different ways in which highlighting or culling a certain parameter range can be performed (Fig. 10). The first method is vertex-based selection (Fig. 10, A and D). In this mode, only vertices matching the selected parameter ranges will be kept. While this approach represents the user's selection exactly, it tends to produce very short line segments in some areas (Figure 10, A). The line-based selection mode keeps an entire line if at least one vertex fits the parameter range (Fig. 10, B and E). This solves the problem of having very short line segments, but can lead to confusion as to which exact part of a pathline actually lies within the parameter range. The third mode is a combination of both previous modes. Like in the line-based mode, the entire line is kept. Additionally, vertices matching the parameter range are highlighted with white outlines (Fig. 10, C and F).

In addition to these three modes, the user can also choose how vertices or lines that do not match the parameter range are handled. They can either be removed from the visualization completely (Fig. 10, A-C) or be shown with strongly reduced opacity (Fig. 10, D-F). Completely removing them reduces visual clutter, but may also remove context information about the flow surrounding the selected areas. To convey these rather technical options to the user in an understandable way, they are presented in the user interface using expressive icons (Fig. 10, bottom).

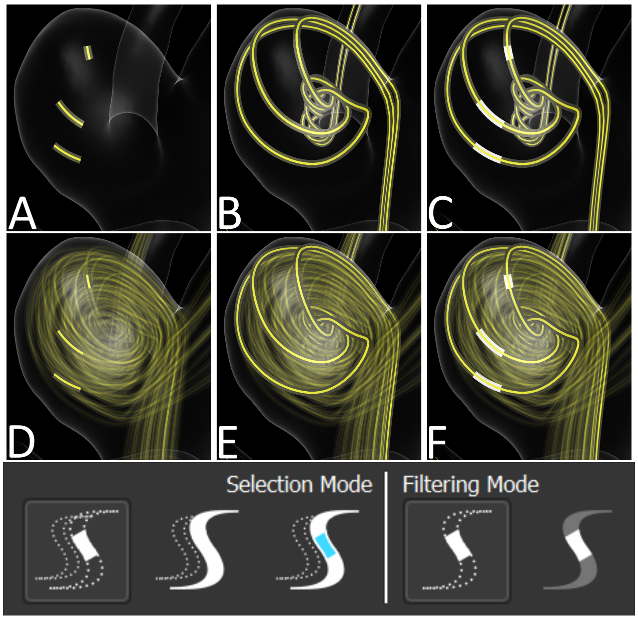

Figure 10: *The six different combinations of filtering settings; vertex-based selection (A,D), line-based selection (B,E) and line-based selection with vertex highlights (C,F). Vertices / lines outside the selected parameter range are removed completely (A,B,C) or have their opacity reduced to 20% (D,E,F). Icons used to represent these option in the user interface (bottom).*

5.6. Reverse Surface Selection

The usual workflow involves selecting an interesting surface region and then extracting pathlines that pass this section closely. In some situations, however, physicians would not only be interested in which flow structures cause a specific surface feature, but also what other surface features the selected flow structure may pass. Therefore, it is possible to select additional surface regions based on their distance to an already extracted pathline bundle (Fig. 11).

This is implemented similarly to the way the pathline bundles are selected, except now the pathline vertices are written to the KD tree and compared against the surface vertices. By default, the same distance threshold is used, although this can be adjusted. The user can then map any parameter to the generated surface to look for other interesting surface features (Fig. 11, right). Since we found the black outlines around the different shades of the color scale to be distracting in a view that also includes pathlines, we disabled them by default.

At any point during the described workflow, the user can go back to previous tasks without losing any information. For example, if the pathline filtering pointed the users to possibly interesting surface regions they have not yet selected, they can return to the surface patch selection mode and add additional patches.

6. Evaluation

To evaluate our methods, we asked two experienced neuroradiologists and an expert in flow simulation to apply it to nine aneurysms

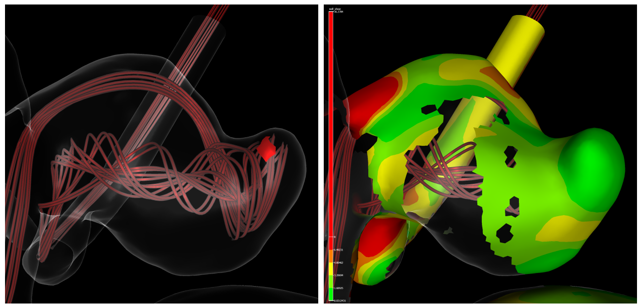

Figure 11: *Original surface patch used to select a pathline bundle (left); additional surface regions extracted using the same bundle, with wall shear stress mapped to the color scale (right).*

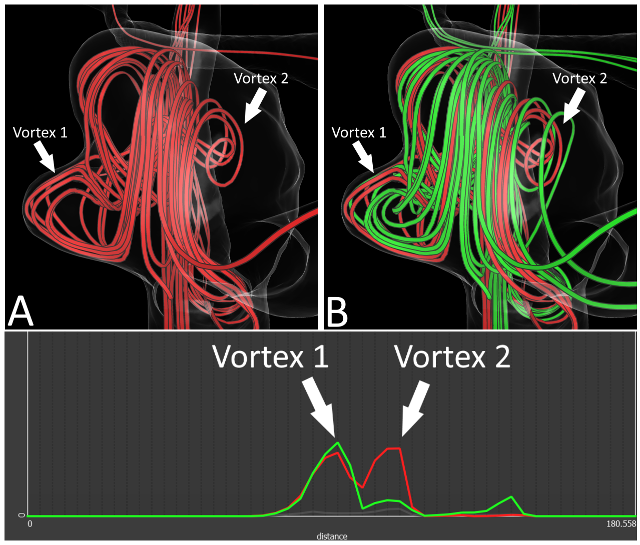

Figure 12: *Using the line chart plotting average residence time over flow distance to differentiate multiple vortices from a complex flow structure; the line chart (bottom) clearly shows multiple structures being present in a flow bundle (A) and helps creating a selection that only contains one of the structures (B, green lines).*

and recorded their findings. Two of these datasets were from a longitudinal study, acquired three years apart from each other. We also asked for general feedback in an informal interview afterwards.

The first neuroradiologist and the flow expert were able to use the tool themselves after a short introduction and demonstration on one dataset. The second neuroradiologist participated over the internet, using remote control. While she gave precise instructions on which patches to select, the actual interaction with the application was performed by us.

All three experts described our method as an advancement in the field of explorative flow visualization. They were able to quickly find interesting surface regions that almost always yielded interesting flow patterns such as vortices when selected. The color-coding proved especially useful for assessing which adjacent vessels a particular flow pattern drains into. According to the experts, a precise selection of specific flow patterns based on their relation to surface features has previously not been possible. They highly appreciated the visualization of splitting flow. Overall, our combination of interaction, visualization and filtering techniques allows for systematic exploration and qualitative assessment of flow structures.

The first neuroradiologist was primarily interested in patches with either high or low normalized wall shear stress or high OSI. His main goal was to correlate vortex structures in the flow with specific hemodynamic parameters on the vessel wall. To facilitate comparability between datasets, the expert used similar or identical parameter ranges for the placement of surface patches and extraction of pathlines. In a few cases, the expert made use of the function to manually draw patches, for example when a patch would otherwise "bleed" into the parent vessel. The expert also used the line chart to determine if a pathline bundle contains more than one actual vortex structure. To correlate features from the line chart with the 3D visualization, the expert used vertex-based filtering (Fig. 10, A). He also used the line chart to determine if a pathline bundle contains more than one actual vortex structure by plotting the residence time over flow distance. Figure 12 shows an example of such a situation from the evaluation. The red line bundle contains two different vortices, which is not instantly obvious due to the complex nature of the flow in the 3D visualization, but clearly visible in the line chart. With this information, the expert was able to add another patch that captures a flow bundle only passing through one of these vortex structures (green line bundle).

Placing a single patch took the expert between 24 and 110 seconds, depending on the complexity of the vessel geometry and if he had to manually draw a patch. Since settings such as mapped surface parameter, number of color scale shades or custom parameter ranges were reset to a default value when switching between datasets, he wished for a way to change the default values or create custom presets to accelerate the process of placing patches.

Figure 13 shows two pathline bundles the first neuroradiologist selected in an aneurysm. The red bundle was selected based on a local pressure minimum, the green bundle based on a wall shear stress minimum. Both the red and green vortices only appear after the blood flow hits the vessel wall. The flow decelerates when entering the aneurysm and accelerates when leaving it (Fig. 13, right). Although the aneurysm is located at a bifurcation, the flow from both vortices drains exclusively into only one of the adjacent vessels.

The second and third dataset were acquired from the same patient at different points in time. The neuroradiologist was therefore interested in visualizing the development of the aneurysm and flow. Since our application did not directly support the comparison of datasets, he improvised by running two instances at the same time and manually adjusting the camera to show a similar angle (Fig. 14). Selecting a patch at a similar location in both datasets allowed for a qualitative comparison of the changes in flow patterns.

The second, independent neuroradiologist was primarily interested in visualizing splitting flow in aneurysms for the purpose of optimal flow diverter placement. She stated that highlighting the splitting flow structures can provide decision support for the placement of flow diverters. Usually, the physician wants to place a flow diverter to reduce pressure from the aneurysm without covering neighboring vessels completely since this would stop blood supply via these vessels. According to the expert, experienced neuroradiologists are often able to infer this information from the wall geometry alone. However, visualizing the splitting flow could be a valuable help to less experienced neuroradiologists. Since the expert had limited interest in correlating flow structures with surface parameters, she mostly placed patches based on geometric features, such as bleps or the aneurysm dome. When filtering pathline bundles based on their hemodynamic parameters, this expert preferred line-based filtering with reduced opacity for filtered lines (Fig. 10, F) We did not record the time she took for patch selection since the interaction was not performed directly by the expert.

Our method proved to be stable in respect of the parameter chosen for the surface patch selection. Interesting flow structures often manifest in multiple surface parameter changes, either in different locations or different parameters. For example, the red line bundle in Figure 13 could have also been selected using the OSI parameter. In cases where a selection either did not yield an interesting flow structure or resulted in multiple structures at once, the resulting pathlines usually contained clues about more promising surface regions that could be selected instead. The ability to manually draw patches onto the surface without regard to the underlying parameters proved to be useful when the experts wanted to select a region based on vessel morphology that did not fully correlate to any surface parameter.

A point of criticism was that manually adjusting the surface color scales was often necessary. The color scale domain is initialized using the global parameter minimum and maximum, yet the physician is generally looking for local minima and maxima. These values may not always be visible initially due to the discrete nature of the color scale, therefore requiring manual adjustment. They requested various (possibly customizable) presets for these adjustments to be added to the application in order to save time when selecting surface patches with recurring parameter configurations.

All experts expressed their interest in being able to further quantify various aspects of our visualization. An example for that would be the ability to measure the size and extent of detected structures. More complex measures, such as the amount of flow that passes through a certain structure or directly underneath a surface patch, would be desirable as well. Adding quantitative measures for each line bundle would allow the physician to gain a deeper understanding of the flow patterns and also enable comparisons between the detected flow structures.

Another requested feature was the ability to place a plane into the parent vessel of an aneurysm and record the color and spatial positions of pathlines passing through it. This would generate a flow profile depicting which regions of the vessel cross section feed or drain into different flow structures. A potential use for this kind of information would be the optimization of stent placement.

7. Discussion

The feedback from all experts shows that our method can support the visual exploration of blood flow and its relation to surface features. According to their feedback, we were able to fulfill the requirements presented in Section 1. The use of a discrete color scale

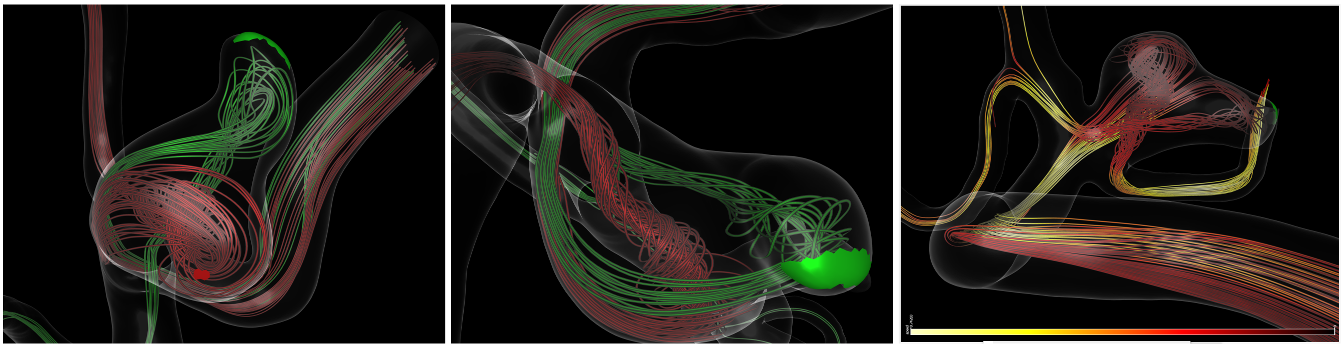

Figure 13: *Flow selection performed by one of the neuroradiologists from three different perspectives; the right image has the flow velocity mapped onto the pathlines using a temperature color scale.*

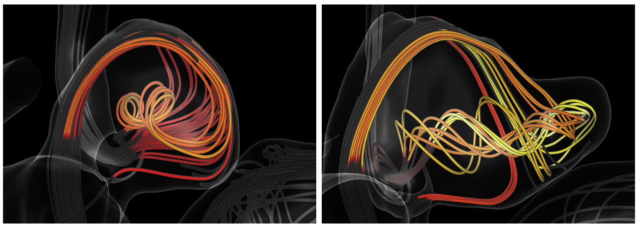

Figure 14: *Comparison of the same aneurysm acquired in 2011 (left) and 2014 (right); the residence time is mapped onto a temperature color scale.*

allows for a fast localization of extreme hemodynamic parameters. Unlike previous approaches such as [vBB*10,ZMH*09,GNBP11], the local flow structures associated to an interesting surface structure can be visualized with only a few mouse clicks. Further exploration of the resulting pathline bundles is possible by either mapping their parameters onto a color scale for filtering them in real-time using a parameter scatterplot, parallel coordinates view or line chart. Both the extraction of pathline bundles as well as the additional filtering can be performed in real-time. Color-coding the selected patches and associated pathlines allows for an easy visual assessment of the entire course of a bundle, similar to [vBB*10].

At present, our application's ability to perform quantitative analysis in addition to qualitative assessments is limited. Users can easily find interesting flow structures and visually compare their quantitative parameters using the line chart. To remedy the lack of additional, in-depth analysis features, the quantitative data from any pathline bundle can be exported as a CSV file. This allows the user to employ an external application of their choice to perform further analysis.

8. Conclusion & Future Work

We have presented a set of intuitive techniques to allow for an interactive exploration of local blood flow based on surface features. Both clinical and the flow simulation expert appreciated the local selection techniques to analyze blood flow characteristics in combination with surface parameters. In fact, both of them stated that they were missing this opportunity in their respective known tools. They also were interested in the visualization of the flow splitting and appreciated the presentation of the pathlines for the entire vessels. Furthermore, the sophisticated real-time filtering techniques including parameter-based filtering and usage of parallel coordinate views as well as the scatterplot could fulfill all of their requests regarding selection of specific blood flow characteristics.

Although tailored for the use on cerebral aneurysms, our methods can be easily adapted to other applications both in- and outside of the medical field. All that is required for our tool to work is a surface model and a set of arbitrarily generated pathlines. Having quantitative parameters mapped to them extends the filtering possibilities, but is not strictly required for our application to be used. Problems could arise when working with complex intertwining surface models, since occlusions might hinder the user's ability to select certain parts of the surface. Possible solutions in this scenario would be to use semi-transparent surfaces in combination with an automated selection algorithm, as presented by Mühler et al. [MTRP10].

For the scope of this paper, we limited our application to work with pre-integrated pathlines. This ensures that all interactions with our tool can be performed in real-time. For the future, we plan to add dynamic seeding capabilities to our application. If the users feel that a certain interesting area is under-detailed due to a lack of pathlines, they may dynamically add more detail by seeding additional pathlines.

At the moment, our tool is only focused on the exploration of a single dataset. There are, however, many scenarios in which physicians would like to compare different datasets. For example, a physician may want to see how a treatment procedure they have performed affected the blood flow in comparison to a dataset acquired before the procedure. To support these comparisons, further quantitative values in addition to the existing ones should be extracted, for example about the flow directly underneath a patch or the patch itself. Instead of simply showing multiple datasets side-by-side in isolated views, an integrated visualization would be desirable. This would require translating either the surface patches or the seed points for a selected pathline bundle to highlight how the flow has changed between datasets.

Acknowledgments

We would like to give thanks to Prof. Christina Iosif for her participation in our evaluation and the very useful feedback she provided. Additional thanks go to Stefanie Glaßer and George Reeder for their help with the accompanying video to this paper.

The work of this paper is funded by the European Regional Development Fund under the operation number 'ZS /2016/04/78123' as part of the initiative "Sachsen-Anhalt WISSENSCHAFT Schwerpunkte" as well as the German Ministry of Education and Research (13GW0095A) within the STIMULATE research campus.

References

[BB17] BERG P., BEUING O.: Multiple intracranial aneurysms: A direct hemodynamic comparison between ruptured and unruptured vessel malformations. *International journal of computer assisted radiology and surgery* (2017). 4

[BHK*16] BROOS A. J. M., HOON, NIELS H. L. C. DE, KONING, PATRICK J. H. DE, GEEST, ROB J. VAN DER, VILANOVA A., JALBA A. C.: A Framework for Fast Initial Exploration of PC-MRI Cardiac Flow, 2016. 3

[BPM*13] BORN S., PFEIFLE M., MARKL M., GUTBERLET M., SCHEUERMANN G.: Visual analysis of cardiac 4D MRI blood flow using line predicates. *IEEE transactions on visualization and computer graphics 19*, 6 (2013), 900–912. 3

[BSJ*14] BERG P., STUCHT D., JANIGA G., BEUING O., SPECK O., THÉVENIN D.: Cerebral blood flow in a healthy Circle of Willis and two intracranial aneurysms: Computational fluid dynamics versus four-dimensional phase-contrast magnetic resonance imaging. *Journal of biomechanical engineering 136*, 4 (2014). 3

[CDM*17] CHUNG B. J., DODDASOMAYAJULA R., MUT F., DETMER F., PRITZ M. B., HAMZEI-SICHANI F., BRINJIKJI W., KALLMES D., JIMENEZ C., PUTMAN C., CEBRAL J. R.: Angioarchitectures and Hemodynamic Characteristics of Posterior Communicating Artery Aneurysms and Their Association with Rupture Status. *American Journal of Neuroradiology 38*, 11 (2017), 2111–2118. 1, 3

[DCHS*17] DODDASOMAYAJULA R., CHUNG B., HAMZEI-SICHANI F., PUTMAN C. M., CEBRAL J. R.: Differences in Hemodynamics and Rupture Rate of Aneurysms at the Bifurcation of the Basilar and Internal Carotid Arteries. *American Journal of Neuroradiology 38*, 3 (2017), 570–576. 1, 3

[GBNP15] GLASSER S., BERG P., NEUGEBAUER M., PREIM B.: Reconstruction of 3D Surface Meshes for Bood Flow Simulations of Intracranial Aneurysms. In *Proceedings of Computer- and Robot-Assisted Surgery (CURAC)* (2015), pp. 163–168. 3

[GLv*12] GASTEIGER R., LEHMANN D. J., VAN PELT R., JANIGA G., BEUING O., VILANOVA A., THEISEL H., PREIM B.: Automatic Detection and Visualization of Qualitative Hemodynamic Characteristics in Cerebral Aneurysms. *IEEE transactions on visualization and computer graphics 18*, 12 (2012), 2178–2187. 2

[GNBP11] GASTEIGER R., NEUGEBAUER M., BEUING O., PREIM B.: The FLOWLENS: A focus-and-context visualization approach for exploration of blood flow in cerebral aneurysms. *IEEE transactions on visualization and computer graphics 17*, 12 (2011), 2183–2192. 2, 11

[GNKP10] GASTEIGER R., NEUGEBAUER M., KUBISCH C., PREIM B.: Adapted Surface Visualization of Cerebral Aneurysms with Embedded Blood Flow Information. In *VCBM* (2010), pp. 25–32. 5

[JBB*13] JANIGA G., BERG P., BEUING O., NEUGEBAUER M., GASTEIGER R., PREIM B., ROSE G., SKALEJ M., THÉVENIN D.: Recommendations for accurate numerical blood flow simulations of stented intracranial aneurysms. *Biomedizinische Technik. Biomedical engineering 58*, 3 (2013), 303–314. 3

[KGP*13] KÖHLER B., GASTEIGER R., PREIM U., THEISEL H., GUTBERLET M., PREIM B.: Semi-automatic vortex extraction in 4D PC-MRI cardiac blood flow data using line predicates. *IEEE transactions on visualization and computer graphics 19*, 12 (2013), 2773–2782. 3

[LEBB09] LALL R. R., EDDLEMAN C. S., BENDOK B. R., BATJER H. H.: Unruptured Intracranial Aneurysms and the Assessment of Rupture Risk based on Anatomical and Morphological Factors: Sifting Through the Sands of Data. *Neurosurgical focus 26*, 5 (2009), E2. 3

[LGV*16] LAWONN K., GLASSER S., VILANOVA A., PREIM B., ISENBERG T.: Occlusion-free Blood Flow Animation with Wall Thickness Visualization. *IEEE transactions on visualization and computer graphics 22*, 1 (2016), 728–737. 2

[LMSC11] LEE T.-Y., MISHCHENKO O., SHEN H.-W., CRAWFIS R.: View point evaluation and streamline filtering for flow visualization. In *IEEE Pacific Visualization Symposium* (2011), pp. 83–90. 2

[MTRP10] MÜHLER K., TIETJEN C., RITTER F., PREIM B.: The medical exploration toolkit: An efficient support for visual computing in surgical planning and training. *IEEE transactions on visualization and computer graphics 16*, 1 (2010), 133–146. 11

[MVB*17] MEUSCHKE M., VOSS S., BEUING O., PREIM B., LAWONN K.: Combined Visualization of Vessel Deformation and Hemodynamics in Cerebral Aneurysms. *IEEE transactions on visualization and computer graphics 23*, 1 (2017), 761–770. 3

[NLB*13] NEUGEBAUER M., LAWONN K., BEUING O., BERG P., JANIGA G., PREIM B.: AmniVis - A System for Qualitative Exploration of Near-Wall Hemodynamics in Cerebral Aneurysms. *Computer Graphics Forum 32*, 3pt3 (2013), 251–260. 3, 6

[OJCJP16] OELTZE-JAFRA S., CEBRAL J. R., JANIGA G., PREIM B.: Cluster Analysis of Vortical Flow in Simulations of Cerebral Aneurysm Hemodynamics. *IEEE transactions on visualization and computer graphics 22*, 1 (2016), 757–766. 2

[OLK*14] OELTZE S., LEHMANN D. J., KUHN A., JANIGA G., THEISEL H., PREIM B.: Blood Flow Clustering and Applications in Virtual Stenting of Intracranial Aneurysms. *IEEE transactions on visualization and computer graphics 20*, 5 (2014), 686–701. 2

[SS06] SALZBRUNN T., SCHEUERMANN G.: Streamline predicates. *IEEE transactions on visualization and computer graphics 12*, 6 (2006), 1601–1612. 3

[Thi11] THIBIEROZ N.: Order-independent transparency using per-pixel linked lists. *GPU Pro 2* (2011), 409–431. 5

[vBB*10] VAN PELT R., BESCÓS J. O., BREEUWER M., CLOUGH R. E., GRÖLLER M. E., TER HAAR ROMENIJ B., VILANOVA A.: Exploration of 4D MRI Blood Flow using Stylistic Visualization. *IEEE transactions on visualization and computer graphics 16*, 6 (2010), 1339–1347. 2, 11

[vGL*14] VAN PELT R., GASTEIGER R., LAWONN K., MEUSCHKE M., PREIM B.: Comparative Blood Flow Visualization for Cerebral Aneurysm Treatment Assessment. *Computer Graphics Forum 33*, 3 (2014), 131–140. 2

[Wie03] WIEBERS D. O.: Unruptured Intracranial Aneurysms: Natural History, Clinical Outcome, and Risks of Surgical and Endovascular Treatment. *The Lancet 362*, 9378 (2003), 103–110. 3

[WvdSAR07] WERMER M. J., VAN DER SCHAAF I. C., ALGRA A., RINKEL G. J.: Risk of Rupture of Unruptured Intracranial Aneurysms in Relation to Patient and Aneurysm Characteristics. *Stroke 38*, 4 (2007), 1404–1410. 3

[XTSM14] XIANG J., TUTINO V., SNYDER K., MENG H.: CFD: Computational Fluid Dynamics or Confounding Factor Dissemination? The Role of Hemodynamics in Intracranial Aneurysm Rupture Risk Assessment. *American Journal of Neuroradiology 35*, 10 (2014), 1849–1857. 3

[ZMH*09] ZACHOW S., MUIGG P., HILDEBRANDT T., DOLEISCH H., HEGE H.-C.: Visual exploration of nasal airflow. *IEEE transactions on visualization and computer graphics 15*, 6 (2009), 1407–1414. 2, 11