Eurographics Workshop on Visual Computing for Biology and Medicine (2017)
S. Bruckner, A. Hennemuth, and B. Kainz (Editors)

Concentric Circle Glyphs for Enhanced Depth-Judgment in Vascular Models

Nils Lichtenberg[1], Christian Hansen[2], Kai Lawonn[1]

[1]Institute for Computational Visualistics, University of Koblenz, Germany
[2]Computer Assisted Surgery group, University of Magdeburg, Germany

Abstract

Using 3D models of medical data for surgery or treatment planning requires a comprehensive visualization of the data. This is crucial to support the physician in creating a cognitive image of the presented model. Vascular models are complex structures and, thus, the correct spatial interpretation is difficult. We propose view-dependent circle glyphs that enhance depth perception in vascular models. The glyphs are automatically placed on vessel end-points in a balanced manner. For this, we introduce a vessel end-point detection algorithm as a pre-processing step and an extensible, feature-driven glyph filtering strategy.

Our glyphs are simple to implement and allow an enhanced and quick judgment of the depth value that they represent. We conduct a qualitative evaluation to compare our approach with two existing approaches, that enhance depth perception with illustrative visualization techniques. The evaluation shows that our glyphs perform better in the general case and decisively outperform the reference techniques when it comes to just noticeable differences.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Display algorithms

1. Introduction

Medical 3D models can be used for surgery planning. The vascular structures within these models, that are located in certain risk areas, can be of special interest for the surgeon. In a liver tumor resection scenario, for example, the vessels around a tumor may have a significant impact on the resection strategy that is defined by a surgeon [HZH*09, HZS*14]. Based on their spatial location, vessels might be included in a resection volume, or represent risk structures that need to be preserved. High-quality 3D models can be obtained via CT or MRI and help a surgeon to obtain a full overview of the vasculature. It is known that depth cues are crucial for the perception of 3D scenes [RHFL10]. Thus, augmenting a scene with supporting depth cues can help a surgeon with the interpretation of medical data.

The perception of a 3D object can be naturally enhanced by stereopsis or by rotating the object in an animation, taking advantage of motion parallax. When no interactive visualization is desired or possible, other perceptual cues are necessary. Such a situation may occur in an operation room, where an interactive application may be too distractive for surgeons. Furthermore, static images might be useful for physician-patient communication, medical documentation, and augmented reality visualization [RHD*06].

When displaying a 3D object on a 2D screen, the most significant property that is missing is the distance of a point on the surface to the viewer - or simply the depth of a point. This gap has to be filled in order to allow the observer to create a cognitive image of the 3D object. While perspective projection can be used in still images, it may however be insufficient to reveal small differences in depth. When no prior knowledge of the viewed structure is available, then perspective projection may even have no advantage at all, because single parts of the structure can be misinterpreted. For example, the projection of a circle with a larger radius, that is further away, may be as large as the projection of a smaller circle that is closer to the viewer.

In this work, we introduce a circle glyph, that is suitable to represent depth information. With this, we aim to augment static, as well as dynamic, 3D scenes with further depth information to help the viewer to get a thorough and precise understanding of a presented 3D object. We do this by the example of several vascular models, obtained from CT scans of the lung and liver. To cover the vasculature with glyphs, we come up with a graph-based selection method, that can take several properties into account. The selection method is view dependent and places more glyphs at depth-complex features, while maintaining an even distribution among the screen-space.

We examine the advantages and disadvantages of the new circle glyphs in a quantitative evaluation, that conducts a direct comparison with two existing methods. Our method is flexible and can easily be integrated into existing visualizations. The accuracy of our glyphs will be underlined by our evaluation. In summary, we make the following contributions:

1. A vessel end-point detection.
2. A feature-driven, filtered glyph placement strategy.
3. A circle glyph design, that allows for precise depth measurement in orthographic and perspective 3D visualizations.

2. Related Work

For this work, we have to cover two main topics. The first is the field of research that deals with depth perception in computer graphics, especially in the medical domain. Secondly, there has been intensive research in the area of glyph-supported visualization. The literature also addresses general guidelines for the proper design and placement of glyphs, which strongly depend on the data and task at hand.

2.1. Spatial Perception

Spatial perception is naturally given through stereopsis. Hardware like 3D monitors or head-mounted displays (HMDs) employ this to give the viewer an impression of depth. However, HMDs can only be used by one person at a time. Hence, they require specially designed applications, particularly if collaboration of several users is desired. Stereoscopic images are further not suitable for print. Thus, it is still desirable to offer monoscopic visualizations for specific tasks. In the context of vascular structures, Kersten-Oertel [KOCC14] has covered a range of depth cues in an evaluation. In their setup aerial perspective and pseudo-chromadepth performed as better depth cues than stereopsis. Aerial perspective is an atmospheric effect, due to scattered light. Objects that are further away are perceived with less contrast [Gib50]. An application of aerial perspective was tested for digitally reconstructed radiographs by Kersten et al. [KSTE06]. Chromadepth utilizes a similar approach. It has first been used by Steenblik [Ste87] and uses the visible color spectrum, rather than the contrast, to encode depth. An extension, pseudo-chromadepth, has been proposed by Ropinski et al. [RSH06]. Instead of the visible color spectrum, a range of colors from red to blue is utilized. The color-range has been chosen with respect to psychological foundations. Other approaches have used illustrative visualization techniques to enhance depth perception. For example, Ritter et al. have used illustrative shadows to emphasize the distance between vessels. Shadows between vessels of different distances can be distinguished by different numbers of hatching strokes. Generally, hatching can be employed to convey shape as done by Interrante et al. [IFP97]. Less salient visualizations use feature lines and focus on geometric properties to make a visualization more comprehensive, as done by Zhang et al. [ZHX*11]. Rendering geometry with such line-drawings can also help to address occlusion issues. A combination of line-drawings as mentioned above has been implemented in the work by Lichtenberg et al. [LSHL16]. Instead of modifying the appearance of a 3D model itself, ideas have been proposed that add supporting geometry to a scene. The first to mention is the work by Bichlmeier [BHFN09] who added a virtual mirror to a visualization. The virtual mirror allowed to add a second view perspective which can help to resolve problems with occluding geometry. Lawonn et al. used such supporting geometry as a shadow plane [LLPH15] or a cut-away object with depth information [LLH17] to support depth perception. Further techniques that address medical visualization with respect to different tasks can be found in the report by Preim et at. [PBC*16].

In scenes with many objects, or complex objects such a vessel trees, occlusion is often a problem. Then, transparency can be used to dynamically highlight structures of interest [EAT07]. The flexible management of order-independent-transparency renderings is feasibly, thanks to modern GPU architecture [VPF15].

2.2. Glyphs

Glyphs are a well accepted concept to display multivariate data. Ward et al. [War02] have first introduced a taxonomy on glyph placement. Their taxonomy distinguishes between *data-driven* and *structure-driven* placement. In *data-driven* placement, the glyph location depends on the data that it represents, e.g., in a 2D plot. An example for *structure-driven* placement would be the visualization of a graph, using glyphs to represent the graph nodes.
As Ward's taxonomy can not be applied to 3D images or surface data, Ropinski et al. [ROP11] introduced an additional placement strategy. Their *feature-driven* and *data set-driven* placement addresses glyph placement in the medical domain. Glyphs based on the iso-surface of a 3D image are *feature-driven*, while the placement on the nodes of a voxel-grid are *data set-driven*. Examples for Ropinski's feature-driven placement are given in [LLH17], [RSMS*07] and [MSSD*08], where glyphs are placed on an iso-surface, orientated based on surface normal. A data set-driven approach is given in [OHG*08], where glyphs are placed on nodes of a voxel grid or segments of the AHA-heart-model.

A third possible placement strategy mentioned in their work is based on filtering. Filtering aims for the reduction of possible glyph positions to emphasize certain properties of the underlying data and to guide the viewers attention. Our placement method can be classified as a *feature-driven*, filtered placement. Anyway, we think that this classification is very general and allows for a more fine grained subdivision that distinguishes between multiple features that can be combined. We return to this topic in Section 3.2.

Despite the regular appearance of glyph placement strategies in literature, information is rare on how to filter glyphs on the surface of a 3D mesh to achieve a well balanced and task-oriented glyph distribution. Generally, *data set-driven* (including meshes) approaches place glyphs randomly and relax the distribution [MSSD*08] or use a grid as an initial placement and avoid the underlying grid structure by jitter [LAK*98]. Mesh based approaches [VGL*14, MVB*17] apply a vertex selection method to evenly cover the surface at different zoom levels. To our knowledge, the filtered glyph placement strategy proposed in [LLH17] is the first to take features into account and to place glyphs at most representative locations. This approach can also be affiliated with Ropinski's *feature-driven* placement, but it is a much more restrictive example, where multiple features are combined. In the work by Rieder et al. [RRRP08] only a single circular glyph is placed at a point of interest. This glyph represents the depth of the point of interest along the view ray and is comparable to the approach that we are going to present in this work.
A thorough survey on the history and application of glyphs has

been published by Borgo et al. [BKC*13]. They also composed a wide range of guidelines for the design of glyphs.

2.3. Direct Foundation for this Work

The work proposed in this paper is directly related to the publications by Lawonn et al.: *Supporting Lines* [LLPH15] (see Fig. 8, center) and *Supporting Anchors* [LLH17] (see Fig. 8, left). Both approaches aim to enhance depth perception of vascular models by augmenting a 3D scene with supporting geometry. Selected locations on the vasculature are projected to simpler geometries that allow for an easier depth perception. A plane, situated beneath the mesh, is used for the supporting lines and a cylinder, surrounding a region of interest, is used for the supporting anchors. The vasculature casts a shadow on the plane and locations on the mesh are connected to their projections within the shadow by lines. Since the shadow is a natural depth cue, depth perception is supported in an intuitive way. For the supporting anchors, selected locations on the mesh are projected onto the closest point of the surrounding cylinder. Anchor shaped drawings are then used to build up a reference between the original location in the mesh and its projection. We also want to improve the depth perception by highlighting the depth of selected points on a mesh, but want to omit such additional geometry. Instead of building up a relation between surface points and their projections on simplified geometry, we place glyphs directly at the positions that these glyphs represent.

3. Method

The vascular data is represented by the mesh $\mathcal{M}$, containing vertices $\mathcal{V}$ and edges $\mathcal{E}$. An edge $e_{ij} \in \mathcal{E}$ exists if the vertices $\mathbf{v}_i, \mathbf{v}_j \in \mathcal{V}$ are connected. We aim to place glyphs at locations of vessel end-points, because these are the most representative locations on the structure when it comes to depth perception [LLH17]. Furthermore, we want that k glyphs should evenly cover the mesh, where k is a user-defined value. Our method is split into two main parts:

1. Detection of vessel end-points (Section 3.1)
2. Graph based glyph placement and filtering (Section 3.2)

The set $\mathcal{P} \subset \mathcal{V}$ represents possible vessel end-points, that we get as input. Then the detection of actual vessel end-points $\mathcal{C} \subset \mathcal{P}$ follows. This is done by examining the neighborhood of each $\mathbf{p} \in \mathcal{P}$, which is denoted by $\mathcal{N}_\mathbf{p}$, while considering the boundary vertices of a neighborhood, denoted by $\mathcal{B}_\mathbf{p} \subset \mathcal{N}_\mathbf{p}$.

From the vertices in $\mathcal{C}$ we then build a complete graph $\mathcal{G}$ for further filtering of the candidate glyph positions. Throughout this section, we use several distance measures. For readability, any label in a variation of the following letters refers to scalars or functions related to a certain distance measure: d (geodesic distance), h (screen-space distance) and z (depth).

3.1. Vessel End-Points

We want to detect vessel end-points of the given mesh for the placement of our glyphs. In this work, we refer to a vessel end-point as the location of a half-spherical ending of a tubular section of the vessel model. As a first step, we determine a set $\mathcal{P} \subset \mathcal{V}$ of probable end-points that we obtain with the algorithm proposed by Lawonn et al. [LLH17]. Their method incorporates the shape index and a connected component analysis for the end-point detection. However, the algorithm does not distinguish between vessel end-points and convex regions in general. Thus, we further classify these convex locations $\mathcal{P}$ to distinguish between vessel end-points and non-end-points.

We obtain sets of vertices $\mathcal{N}_\mathbf{p}$ that form a neighborhood around each end-point candidate $\mathbf{p} \in \mathcal{P}$.

$$\mathcal{N}_\mathbf{p} := \{\mathbf{v} \in \mathcal{V} \,|\, d(\mathbf{v},\mathbf{p}) \leq d_\mathbf{p}\} \tag{1}$$

where

$$d_\mathbf{p} = \operatorname*{clamp}_{[d_{min},d_{max}]} \frac{\pi}{2} \frac{m}{\bar{\kappa}_\mathbf{p}} \tag{2}$$

and $d(\mathbf{v},\mathbf{p})$ results in the geodesic distance between the points $\mathbf{p}$ and $\mathbf{v}$, for which we employ the heat method by Crane et al. [CWW13]. The parameter m is a factor to control the magnitude of $d_\mathbf{p}$ and $\bar{\kappa}_\mathbf{p}$ is the mean curvature at $\mathbf{p}$. Thicker vessels will have a lower curvature and consequently $d_\mathbf{p}$ will be larger because it increases with $1/\bar{\kappa}_\mathbf{p}$, yielding larger neighborhoods. As a default set of parameters, we use $m = 3$(dimensionless factor), $d_{min} = 3\ mm$, $d_{max} = 20\ mm$. Note that $d_{max} = 20\ mm$ assumes, that no vessel has a circumference larger than 40 mm, as will be clarified next.

We examine the graph topology of each neighborhood $\mathcal{N}_\mathbf{p}$ and store the boundary vertices in a set $\mathcal{B}_\mathbf{p} \subset \mathcal{N}_\mathbf{p}$. If the edges among the vertices in $\mathcal{B}_\mathbf{p}$ form exactly one cycle, we classify $\mathbf{p}$ as a vessel end-point and add it to the set of vessel end-points $\mathcal{C}$. The rationale behind this idea can be explained by the mesh illustration in Fig. 2 (right). If a set $\mathcal{N}_\mathbf{p}$ is located at a vessel end-point, it will basically have the shape of a cup (i.e., the topology is equal to a disc). Hence, the edges between the vertices in $\mathcal{B}_\mathbf{p}$ will represent one cycle. In contrast to this, a set $\mathcal{N}_\mathbf{p}$, that is located somewhere on a branch, will not have this property. As shown in Fig. 2 (right), example B covers a mid-section of a vessel. The corresponding $\mathcal{B}_\mathbf{p}$ contains two cycles, because of the two open ends. This method works only, if a set $\mathcal{N}_\mathbf{p}$, whose $\mathbf{p}$ is not located at a vessel end-point, covers the full circumference of the vessel. We ensure this with $d_\mathbf{p}$, which depends on the curvature (i.e., the approximated vessel radius) at $\mathbf{p}$. Further, the factor m introduces a large safe-margin by increasing the set size, while the parameters d_{min}, d_{max} control the minimum and maximum extent. A comparison of the locations in $\mathcal{P}$ and the end-points $\mathcal{C}$ detected by our method can be found in Fig. 1.

3.2. Multi-Feature Glyph Placement

After the vessel end-point detection, we have obtained a set $\mathcal{C}$ of possible glyph locations on the input mesh. The cardinality of $\mathcal{C}$ depends on the mesh (i.e. the organ) and on the parameters used for the computation. Vasculatures in lung data sets exhibit more branchings, and thus more vessel endings compared to, for example, liver data sets. Displaying glyphs at all locations quickly overloads the visualization. Hence, we want the user to be able to control the number k of glyphs that are visible. The selection of the k glyphs however, should be done automatically by an appropriate filter method.

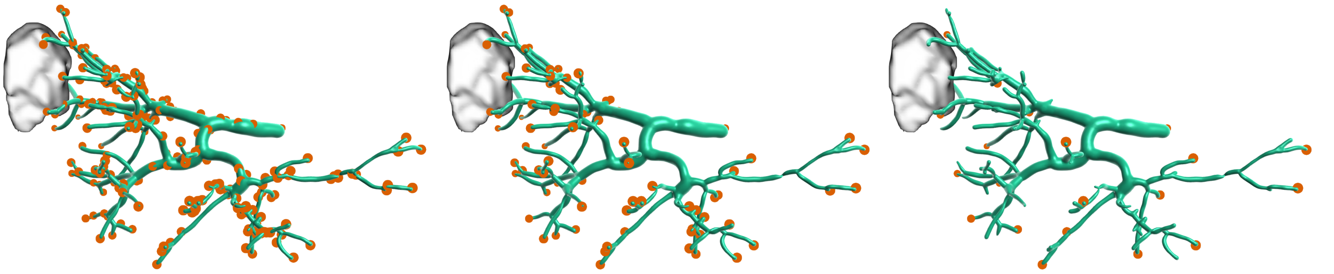

Figure 1: *All locations in $\mathcal{P}$, found by the method by Lawonn [LLH17] (left). Vessel end-points $\mathcal{C}$, detected by our method (center). Filtered glyph locations $\mathcal{S}$, for $k = 20$ (right).*

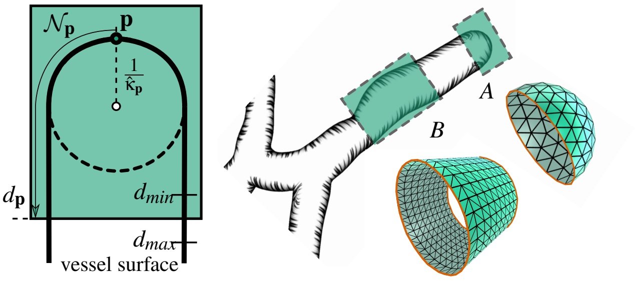

Figure 2: *Schematic depicting $\mathcal{N}_\mathbf{p}$ (left). Valid end-point A and invalid candidate example B (right, top). Both example sets are shown as a mesh section of the vascular model (right, bottom). The boundaries of the sections are highlighted in orange.*

Results of the quantitative evaluation in [LLPH15] have shown, that subjects found it most difficult to find the correct depth relation between two points, when the points were far away from each other (in screen space) and had a similar depth. Consequently, we would like to prefer such combinations of point pairs to show glyphs at. Another aspect we have to consider is the effect of size constancy, which is a well-known theorem in perception psychology. It says that the perceived size of an object remains constant, even though its projective image on the retina changes (i.e., an object moving away from the viewer). Due to the size constancy theorem, a thicker vessel might appear to be closer to the viewer than a thinner vessel, independent of the actual depth. For each vessel end-point $\mathbf{p} \in \mathcal{C}$ we know its approximated thickness in world space, given by

$$r_\mathbf{p} := \frac{1}{\bar{\kappa}_\mathbf{p}}. \tag{3}$$

This is even more important to consider when using orthographic projection which may lead to misinterpretation of the depth. To attribute for this, we want to prefer locations that represent either a thick vessel with a high depth value, or a thin vessel with a low depth value, to avoid false interpretations. Our objectives for the filter method are then:

- R1 Balanced screen space distribution, preferring locations far away from each other (camera dependent).
- R2 Depth aware distribution, preferring locations with similar depth (camera dependent).
- R3 Favor locations that are critical with respect to size constancy (geometry dependent).

Extracting k glyph locations from $\mathcal{C}$ is done by using a graph based algorithm. The graph build-up and the selection of k glyph locations based on that graph are described in the following sections.

3.2.1. Graph Buildup

At first, we build a weighted complete graph $\mathcal{G}$ that connects all possible glyph locations $\mathbf{p} \in \mathcal{C}$. Edge costs $\mathcal{G}_{ij}$ between two locations $\mathbf{p}_i$ and $\mathbf{p}_j$ are computed as a combination of three terms (see Eqs. 6,7), covering the aforementioned objectives. Main parameters for the computation are the depth and screen space position of a location, as well as the vessel thickness $r_\mathbf{p}$. Given the minimal and maximal depth in $\mathcal{C}$ for a given camera position, we map the actual depth z_i of a location $\mathbf{p}_i$ to the normalized range $\bar{z}_i \in [0,1]$.The normalized Euclidean depth distance of two possible glyph locations $\mathbf{p}_i$ and $\mathbf{p}_j$ is then obtained by $\hat{z}_{ij} = |\bar{z}_i - \bar{z}_j|$. A similar mapping is applied to the screen space coordinates of $\mathcal{C}$. After a minimal screen axis aligned bounding rectangle has been computed around all locations of $\mathcal{C}$, the screen space coordinates are normalized, such that the bounding rectangle's diagonal has length 1 (see Fig. 3, right). The maximum euclidean distance between two coordinates within that normalized rectangle is 1 and the distance between two possible glyph locations is given by $\hat{h}_{ij}$. To stick with the notation used by Lawonn et al. [LLPH15], we denote the screen-space and depth relation with the distance labels: FF, FN, NF, NN. Here, F means far and N means near, where the first capital refers to the screen-space distance $\hat{h}_{ij}$ and the second capital to the depth distance $\hat{z}_{ij}$. For $\hat{h}_{ij} \geq \frac{1}{2}$ we use the label F and N otherwise. The same is done for $\hat{z}_{ij}$. For example, a pair of locations where $\hat{h}_{ij} \geq \frac{1}{2}$ and $\hat{z}_{ij} < \frac{1}{2}$ would be notated with FN.

To compute the edge cost $\mathcal{G}_{ij}$, we further apply an approximated Heaviside function, which is a scaled sigmoid function, to the values obtained by $\hat{z}_{ij}$ and $\hat{h}_{ij}$. The Heaviside function resembles the partitioning by the F and N labels, because it can be used to map half of the parameter space to values close to zero (N) and the other half to values close to one (F). A general formulation for the approximated Heaviside function is:

$$f_{a,t}(x) = \frac{1}{1 + exp(-(x - \frac{a}{2})t)} \tag{4}$$

where the parameter t controls the sharpness of the approximated Heaviside function and $[0,a]$ is the range of the input parameter x. As we want to compute f for the values $\bar{z}_{ij}$ and $\bar{h}_{ij}$, we can fix $a = 1$:

$$z_t(i,j) := f_{1,t}(\bar{z}_{ij}), \;\; h_t(i,j) := 1 - f_{1,t}(\bar{h}_{ij}) \tag{5}$$

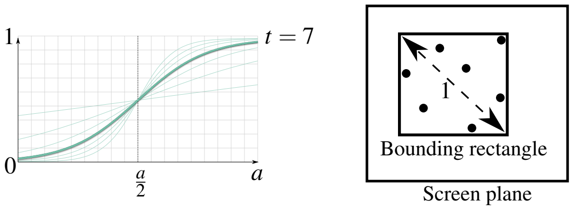

Figure 3: *The approximated Heaviside function for different values of t (left). Bounding rectangle of the glyph locations in $\mathcal{C}$ (right).*

With that we obtain small values for FN labelled pairs (i, j), and high values for NF labelled pairs. We set $t = 7$ as a default value in our implementation (see Fig. 3, left). The rather smooth transition does not completely binarize the input value a, thus allowing a differentiation inside the labels F and N. In $h_t(i, j)$ we invert the results, so that we obtain low values for high screen space distances. The edge cost $\mathcal{G}_{ij}$ is then defined as:

$$\mathcal{G}_{ij} = \frac{z_t(i,j) + h_t(i,j)}{2} \tag{6}$$

covering the requirements R1 and R2. The size constancy (R3) is taken care of with a bonus term

$$\mathcal{G}_{ij} \leftarrow \begin{cases} \mathcal{G}_{ij} - b, & \text{if } r_{\mathbf{p}_i} > r_{\mathbf{p}_j} \text{ and } z_i > z_j \\ \mathcal{G}_{ij} - b, & \text{if } r_{\mathbf{p}_i} < r_{\mathbf{p}_j} \text{ and } z_i < z_j \\ \mathcal{G}_{ij}, & \text{otherwise} \end{cases} \tag{7}$$

where b is the bonus. Thus, edges between nodes that are critical due to the size constancy get their cost reduced by b. In this work $b = \frac{a}{2} = 0.5$ was used as a default value, because this is the location of the transition in the Heaviside function.

3.2.2. Graph Filtering

The selection of k nodes from the graph is done as follows: As an initial point pair, we select the nodes n_i, n_j with the lowest edge cost in $\mathcal{G}$ and add these to the set of selected, visible nodes $\mathcal{S}$. These are typically far away from each other in screen space and have a similar depth, thus, their distance is labelled FN. In order to find the remaining $k-2$ locations, we do an iterative search, consecutively adding nodes that best fit to the nodes in $\mathcal{S}$. We add a parameter to the graph based selection condition, that controls the minimal screen space distance h_{min} between two nodes in $\mathcal{S}$.This parameter can be used to avoid overlapping glyphs and we set a default value of $h_{min} = 0.075$ (7.5% of the maximum distance in the normalized ROI). Our selection method can be summarized in 3 steps:

1. For every node $n_i \in \mathcal{G}$, compute the average edge cost e_i and the minimal screen space distance h_i to all nodes $n_j \in \mathcal{S}$
2. Pick node $n_i \in \{\mathcal{G} \,|\, \arg\min_i e_i\}$, ($i.e., n_i \in \mathcal{G}$ with the lowest average edge cost):
    a. If $h_i \geq h_{min}$, add n_i to the visible set $\mathcal{S} \leftarrow \mathcal{S} \cup n_i$.
    b. Remove n_i from the graph $\mathcal{G} \leftarrow \mathcal{G} \setminus n_i$.
3. Repeat steps 1 and 2 until $|\mathcal{S}| = k$ or $\mathcal{G} = \varnothing$.

Exemplar results of the filtering are shown in Fig. 1.

4. Glyph Design

For the design of our glyph, we have to take several considerations into account. From [ROP11] we know, that visual stimuli are processed in a pre-attentive and an attentive phase. Thus, we want to base our glyph on two main components, such that each component satisfies the conditions for either of these phases. Also they state, that glyph shapes should be unambiguously perceivable independent of viewing direction. Pop-out effects are described in literature, where color is the most significant channel, followed by size, shape and orientation. From [LLH17] we already know that pseudo-chroma depth does not support small differences very well. So, to take advantage of the pop-out effect without losing precision, we choose to address the pre-attentive perception of our glyph by its size. A less salient property is the angle or orientation of a glyph, which is also listed as part of the geometric channel in [BKC*13]. We want to use this as the precise information channel during the attentive phase. Summed up, we want our glyph to meet the following criteria:

1. Pre-attentive stimuli through glyph size.
2. Attentive stimuli through angle/orientation.
3. Perceivable independent of view direction

Keeping in mind that we only want to display one scalar value - the depth, we can come up with a very simple glyph to match the above conditions.

Pre-attentive Phase: We use a circle as a very basic shape (*base circle*). Mapping the depth to the radius of the base circle perfectly matches the size constancy theorem. We can take advantage of the size constancy effect to exaggerate depth differences in a perspective view, or to imitate the effect in orthographic view. This way, the user can quickly identify clusters of glyphs that are closer or further away and hence get an overview of the vessel structure.

Attentive Phase: In the attentive phase, we want to use an orientation attribute to allow for precise depth-judgment. We can combine this with an aspect that Ropinski et al. [RSMS*07] attribute to the pre-attentive phase: The continuous or discrete mapping of an attribute to the glyph can communicate information. The discrete mapping enables better discrimination of fewer values. The continuous mapping allows to distinguish between small, just noticeable differences. Here, we combine the continuous with the discrete mapping, by filling the base circle with three concentric circles. If z_{max} is the maximum and 0 the minimal depth in a scene, and z the depth of a glyph, then we display one full inner circle if $z = \frac{1}{3} z_{max}$, two full inner circles if $z = \frac{2}{3} z_{max}$ and three full inner circles if $z = z_{max}$. This yields a discrete glyph attribute, that supports the pre-attentive phase, but can also be useful in the attentive phase. As illustrated in Fig. 4, the depth range is subdivided into three bins reflected by the three circles. Furthermore, we cover the range between the discrete steps by gradually filling the inner circles, which is the continuous part of our attribute mapping. During the attentive phase, the circle fill-status allows the user to precisely compare depth relations. Since every circle covers only one third of the depth range, the fill-status is more sensitive to differences and thus easier to read. The glyph could also be designed with two, or more than three circles, but we observed that three gave a good balance between sensitivity and readability.

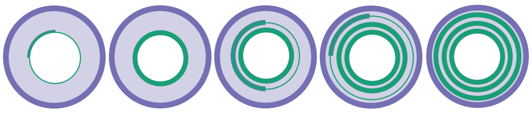

Figure 4: *Combined discrete and continuous mapping. Depth represented by the glyph components are (left to right): $\frac{z_{max}}{12}, \frac{4z_{max}}{12}, \frac{6z_{max}}{12}, \frac{9z_{max}}{12}, \frac{12z_{max}}{12}$. The base circle is displayed in purple.*

Further Information Channels: If it is desired to represent a second scalar value through the glyphs, where precise perception is not crucial, other visual channels can be applied. An example is given in Fig. 5, where the distance of a glyph location to the tumor is encoded in color and geometrical variations of the glyph. The geometric modification only requires an adjustment of the fragment coordinates (x, y) during rendering (compare Fig. 6, left and Eq. 8) The example shows, that three scalar values can be reflected by our glyph, with each scalar using a distinct visual channel.

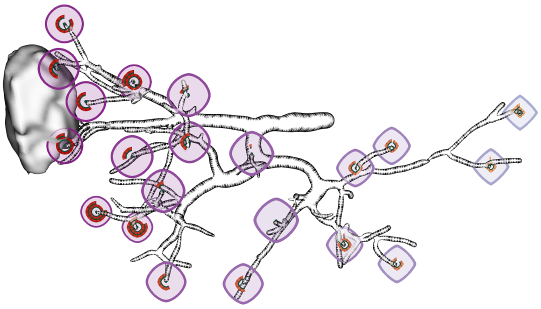

Figure 5: *Tumor distance encoded in additional geometric channels: Close: Red, circular shape; Far: Yellow, rectangular shape.*

Summary: The proposed glyph is designed with respect to aspects of the pre-attentive and attentive perception phases. The pre-attentive phase is attributed to by modifying the base circle size in dependence of depth and by the discrete mapping to the three inner circles. The attentive phase is covered by the circle fill-status. The filling contains an orientational attribute that can be used for a relative comparison of the mapped value. For rendering, the glyph can be implemented as a camera-aligned billboard and is thus independent of camera direction or lighting. Details on the implementation and further considerations to make the glyph feasible in the actual visualization are given in Section 5.2.

5. Implementation

In this section we describe implementation details for the glyph location selection as well as for the glyph visualization and the hatching technique that we apply to the vascular models.

5.1. Glyph Selection

The selection of glyph locations is split into two main parts. The offline pre-processing to find possible locations and the online filtering of glyphs that selects glyph locations with respect to the dynamic camera features. The pre-processing time scales linearly with the number of input candidates of the set $\mathcal{P}$ and further depends on the mesh resolution. Thus, we provide timings for two exemplar meshes to give a rough estimate: For a mesh with $30k$ vertices, we measured an averaged time of $0.33s$ per candidate, while the computation of the geodesic distances, using the heat method [CWW13], took 63% of that time. Another mesh, with $8k$ vertices, took $0.1s$ per candidate, with 71% consumed by the heat method. Note that the pre-processing can be potentially run in parallel for each candidate. In the current implementation of the online filtering, we recalculate the glyph positions for each frame. Thus, the k glyph locations may vary with each slight move of the camera. A thinkable modification would be to update the glyphs only after the camera rotation/movement exceeds a certain threshold, ensuring a stable visualization. The complete graph $\mathcal{G}$ is built up and the edge weights are computed (see Section 3.2.1, Eq. 6,7). Then we apply our filtering method to find k valid glyph locations (see Section 3.2.2) and send these to the OpenGL pipeline. The timings for the filtering depend on the number of possible locations $|\mathcal{G}|$ and the parameter k. For example, the graph buildup and the selection of $k = 15$ glyphs required 1.2 ms for a number of $n = 136$ possible glyph locations and 0.05 ms for $n = 25$ possible glyph locations, exhibiting real-time capabilities. The performance was tested on a desktop computer environment with a 4.00 GHz i7-6400 processor and 16GB RAM.

5.2. Glyph Visualization

Our glyphs can be added to an existing rendering pipeline in a simple way. As the world locations of the glyphs are given by $\mathcal{C}$, we can use point primitives to draw view-aligned quads. The size of the quads can be set in dependence of the depth of each location to account for perspective projection, or to imitate perspective distortion when using an orthographic camera. In the fragment shader, each fragment knows its location within the point primitive, given by *gl_PointCoord*. From the fragment coordinate within the quad, we can derive an angle α and a distance h_f to the quad's center (see Fig.6, left). The distance can be modified with:

$$h_f^2 = \left(\operatorname{sgn}(x) \cdot |x|^q\right)^2 + \left(\operatorname{sgn}(y) \cdot |y|^q\right)^2 \qquad (8)$$

where $q \in [0.5, 1]$ is the mapped distance of the glyph to a reference position (the tumor in Fig. 5). This modification implements the transition from circular glyphs ($q = 1$) to quadratic glyphs ($q = 0.5$). Extending the range of q allows for more shape variations, but then the inner circles become hard to read. Along with the depth z, these are the properties that are necessary to render the glyph (see Fig.6, right). Based on α, h_f and z, each fragment can be appropriately rendered. The base circle is filled with a purple transparent area. The concentric inner circles have two components. A thick, orange component with dashes and a green border between consecutive concentric circles. The dashes help to distinguish between glyphs of very similar depth. In our implementation, we integrated the glyph rendering into an existing order independent transparency pipeline. This allows us to react to situations, where glyphs are occluded by surrounding parts of the vessel structure as in Fig. 7, left. Here, vessels occluding a glyph are rendered with increased transparency. The transparency is unaffected at the

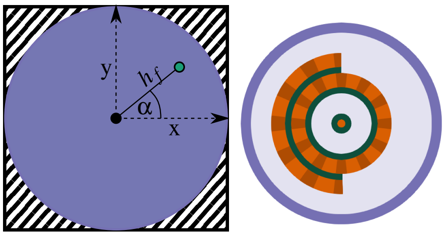

Figure 6: *Fragment (green dot) properties obtained from* gl_PointCoord *(left). Fragments in the striped area are discarded. Final glyph rendering with one and a half concentric circles (right). The dot at the glyph center can help to identify the actual glyph position on the mesh.*

glyph edge and increases towards the center. If the base circle (purple) is occluded, one can tell that it is actually behind the vessel. Hence, overlapping depth cues remain visible.

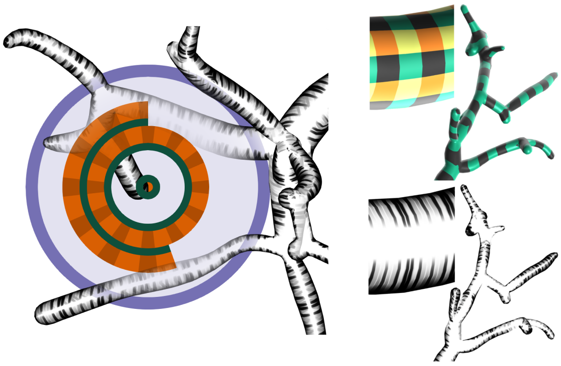

Figure 7: *Surrounding vessels that occlude the glyph have their transparency increased (left). Visualization of the continuous parameterization based on Knöppel [KCPS15] (right, top). Hatching applied (right, bottom).*

5.3. Hatching

In our visualization, the glyphs are desired to be the most prominent aspect. Therefore, we employ a less salient hatching technique that is based on a continuous parameterization, to draw the mesh. The parameterization is based on the work by Knöppel et al. [KCPS15]. Their method yields a globally continuous parameterization that is aligned with a given vector field. Using the principal curvature direction allows us to draw stripes with respect to the vessel geometry. Fig. 7 (right) shows the parameterization and the hatching results. A 4-colored patch (yellow, green, black, orange in Fig. 7, right top) represents texture coordinates in the range $[0,1]$. Using these texture coordinates, we can render multiple strokes with varying length and intensity (see Fig. 7, right bottom). The stroke variation is repeated for every 4-colored patch. We apply the hatching within a margin of the vessel contour that is nearly independent of the vessel thickness, by employing the method by Kindlmann et al. [KWTM03].

6. Evaluation

To assess the performance of our visualization, we conducted a qualitative, comparative, online study. Generally, a comparison to a standard non-depth-enhanced (e.g. phong shading) and a depth-enhanced (e.g. pseudo chroma-depth [KOCC14]) visualization would be feasible to reveal the benefit of a new technique. Such evaluations have already been performed in the works by Lawonn et al. [LLPH15, LLH17]. The results of their questionnaires have proven that their illustrative visualizations can outperform standard techniques. Consequently, our analysis focuses on a direct comparison with the methods by Lawonn et al. Our evaluation is formulated in the same way as the previous ones by Lawonn in order to be comparable with the aforementioned standard techniques.

We implemented the depth-enhancement strategies found in [LLPH15] (*supporting lines*) and [LLH17] (*supporting anchors*) along with our new technique. For five different data-sets, each including a vessel structure and tumor data, we randomly selected two locations to attach the supporting visualization. Then, images were captured for three camera positions for each of the three techniques. This totaled to 45 images that we used for the survey. Example images are shown in Fig. 8. The approaches by Lawonn et al. both create a relation between the vessel structure and a supporting structure. Therefore, these techniques are expected to have less complication with overlapping vessel structures. Contrarily, our circle glyphs are placed directly at the vessel location that they represent. Hence, we assume that overlapping vessel structures will have a negative impact on their performance. The three camera positions were chosen such that the following cases were matched for the two selected vessel end-points and thus yield three difficulty levels:

1. High distance difference (L1, label FF/FN)
2. Circle glyphs partly obstructed (L2)
3. Very low depth difference (L3, label NN/FN)

Case 1 is the simplest task, and inquiries how fast decisions can be made with the respective visualization. With case 2 we aim to learn if overlapping is really a problem for the circle glyphs. The last case probes for the ability of the glyphs to distinguish just noticeable depth differences. The survey was implemented as an online questionnaire, because this enabled us to measure exact timings during the process. However, this means that we did an unsupervised study and did not observe the subjects. Subjects were first introduced to their task in a learning phase. For each visualization technique an example image was shown. The example image contained a description of the technique and two selected vessel end-points (as in Fig. 8). Also, a solution to the question which vessel was further away, was included with an explanation. Then, subjects were shown two sample images per method and had to select the vessel that appeared to be further away. Answers during this training phase have not been included in the evaluation results.

After the training questions, participants were shown the 45 evaluation images in a fixed, but previously shuffled, order. Timings were measured from the point where an image was displayed until the subject selected an answer ((+) or (#) or *undecided*). Further, participants had to estimate their decision confidence with a 5-point Likert scale (1 = most inconfident, 5 = most confident). In the evaluation *undecided* answers were treated as a wrong answer, i.e., we were interested in the percentage of correct answers.

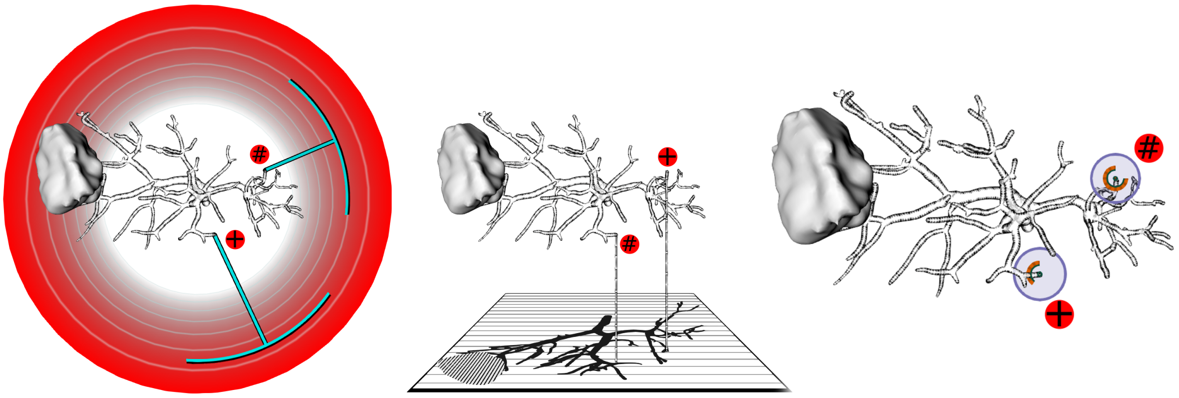

Figure 8: *Supporting anchors (left), supporting lines (center), circle glyphs (right). Vessel end-points in question are tagged with (+) and (#).*

Table 1: *Averaged results over all questions for each technique (Circle glyphs, Supporting Lines, Supporting Anchors).*

	Lines	Anchors	Circle
Correct answer (%)	92.2	87.2	93.6
Confidence (1-5)	4.08	4.21	4.35
Time (s)	6.09	6.42	5.92

7. Results

Our study was carried out with 24 participants (16 male, 8 female, average age 29.75 with a range of 22-57). Most of them (83%) came with a background in computer science and visualization. 29.1% of them had experience with visualizations of blood vessels. Nevertheless, we think that the results are also representative for expert users, since the depth perception is a general task and not only restricted to the medical context. Table 1 depicts the overall results of the study, averaged for each enhancement technique (15 questions each). Most correct answers were given with the proposed circle glyphs (93.6%), directly followed by the supporting lines (92.2%). The supporting anchors lead to 87.2% of right answers. Participants were most confident in their answers with the circle glyphs (4.35), succeeded by supporting anchors (4.21) and supporting lines (4.08). Hence, all averages are between the *very confident* (5) and *confident* (4) estimations. Subjects decided most quickly with our new method (5.92 s), closely followed by the supporting lines (6.09 s) and the supporting anchors (6.42 s) just behind. The timings for individual questions initially ranged from 1.8 to 126.2 seconds. We assume that some subjects were interrupted during the process, which lead to the high values. Hence, we computed the 75% quartile (7.735 s) and removed outliers that lay above three times the 75% quartile (23.205 s). This high tolerance is in order to cover the more difficult questions. The portion of outliers was 1.82%. As mentioned in Section 6, the questions of the survey were subdivided into three levels of difficulty. Charts in Figures 9, 10, 11 show the evaluation results for the individual task levels L1, L2 and L3. The percentage of correct answers (Fig. 9) is highest for the supporting lines in L1. This is likely due to the fact, that the shadow plane in the supporting lines visualization resembles depth cues in the most natural way. The confidence (Fig 10) in L1 is very similar among the techniques and higher than for the other levels. As L1 represented the easiest task, the timings (Fig. 11) are relatively low compared to L2 and L3. But we can observe, that the supporting anchors lack behind the other two methods. For task level L2 we expected our new method to perform worse than for L1. Recall, that L2 contained situations, where the circle glyphs were partially obstructed. Since the depth difference was also smaller in this case, we can observe a negative impact on all three methods in the estimation precision, the confidence and

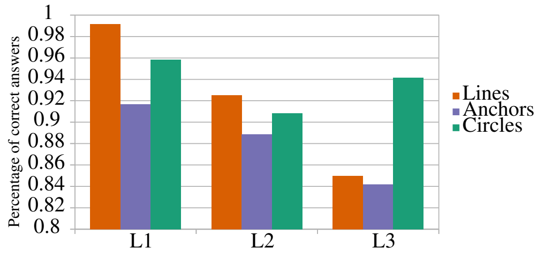

Figure 9: *Estimation precision.*

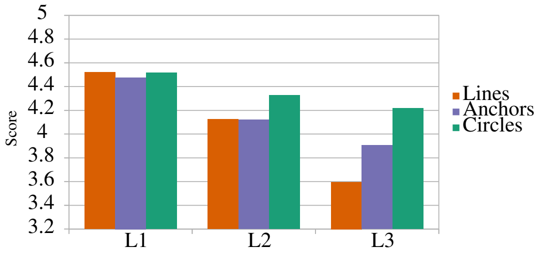

Figure 10: *Confidence.*

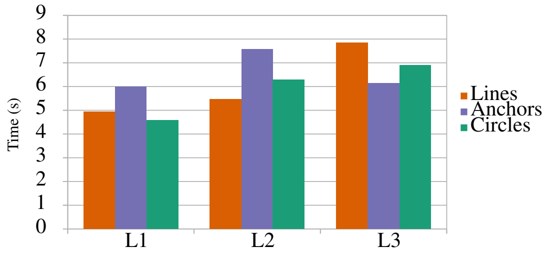

Figure 11: *Task duration.*

Table 2: *Glyph placement quality in terms of screen-space distribution.*

	μ	σ^2
Bipartite	0.090	0.00210
Random	0.078	0.00241
Ours	0.104	0.00085

the task duration. Contrarily to our assumption, the circle glyphs did not lose most precision (-5.56%) compared to supporting lines (-6.72%) and supporting anchors (-3.03%). Nevertheless, the fact that subjects were most confident with the circle glyphs but made the most correct choices with supporting lines in L2, underlines that overlapped glyphs were likely to be misinterpreted. The L3 task clearly shows that our approach outperforms the others in terms of accuracy. Here, 94.8% of answers were correct, while for the reference techniques 85.0% (supporting lines) and 84.1% (supporting anchors) were correct. This is accentuated by the participants' confidence. Surprisingly, the supporting anchors have now the lowest timings. A reason for this might be, that anchors were located very close to each other in L3. Thus, comparing them took less time than comparing two oppositely situated anchors.

In summary, and with the overall results in Table 1 at hand, we can state that all three techniques performed similarly well, with a slight advantage for the new circle glyphs. Only when it came to very small depth differences, our approach was able to outperform the others. We did not evaluate the performance of the glyph filtering technique with subjects but plan this for future work. The new approach does not require the solution of a bipartite graph matching as done for the supporting anchors [LLH17]. Their method tries to find a best matching of two positional properties: The distribution of glyphs around the depth-cylinder, and the depth distribution (refer to Hall's marriage theorem). The graph matching is computationally more intensive and less flexible than our strategy. For example, as described in Section 3.2.2, we use the variable h_{min} to avoid overlapping of glyphs. Such an extension to the filtering approach can not be simply added to the bipartite matching. However, as we place glyphs on top of the vascular model, it is crucial that glyphs do not overlap, i.e., that glyphs have enough space around them. Hence, we are interested in maximizing the minimal distance among the visible glyphs. For an objective measurement, around 1000 random camera positions were generated for four different vascular models. For each scene, we used our technique, the bipartite matching and a random selection, to pick 16 glyph positions. We then computed the minimal distance of each glyph position to the remaining positions, to obtain variance and mean values. The results shown in Table 2 are averaged over the values obtained for all camera positions. The maximum possible distance between two glyphs is 1 (compare Fig. 3, right). Also, for a uniform distribution of glyph positions, the minimal distance is maximized and equal for each sample. Thus, larger minimal distances (i.e., larger mean values) and a smaller variance can be interpreted as a hint for a more uniform distribution. From Table 2 it it can be observed, that the averaged minimal distance of our method is 16% larger than for the bipartite matching and 33% larger than for the random picking. More significant is the difference in variance σ^2. Here, σ^2 for the bipartite matching exceeds our approach by 145%, while the random picking yields a 181% larger value. Thus, we conclude that our technique results in a more balanced distribution across the screen-space, that is suitable for our glyphs. A comparison of glyphs placed with our technique and the bipartite matching is shown in Fig. 12.

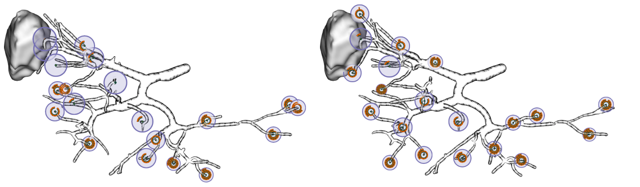

Figure 12: *Comparison of glyph filtering methods. Bipartite matching as described in [LLH17] (left). Our approach (right).*

8. Discussion and Future Work

We have proposed a vessel end-point detection algorithm as an offline pre-processing step. These locations are further filtered by an online feature-driven and graph-based approach to select a subset of vessel end-points that cover the vasculature in a representative way. To support spatial perception of the presented 3D model, we place circle glyphs at the filtered locations. The glyphs are designed with respect to aspects of the pre-attentive and attentive phase in human perception. Further, our parameter mapping employs a combination of a discrete and continuous mapping to allow for a precise reading of the mapped parameter. An advantage over the reference techniques *supporting lines* and *supporting anchors* lies in the simplicity of the glyph design. We do not need to add any supporting geometry (like the shadow plane or cylinder) to the scene. Our glyphs work independently and 'in place'. As we omit such additional geometry, we can use the glyphs at any zoom level. The shadow plane for the supporting lines in contrast, is only reasonable when the whole vasculature is visible. However, the circle glyphs come with a deficiency. The angular attribute of the inner concentric circles has no relation to natural depth perception. Thus, it requires additional cognitive effort to correctly interpret filled and empty glyphs. An issue with positioning glyphs right at the position they refer to is overlap or obstruction by other structures. We have approached this problem by rendering occluding geometry with increased transparency. While the glyph itself is then better perceivable, the impression of the overall visualization might suffer.

The graph-based glyph filtering method proposed in this work takes different features into account to select a specific subset of possible glyphs. This formulation is flexible, as further features could be added to modify the resulting edge costs in $\mathcal{G}$. We conducted a quantitative evaluation and were able to show that the proposed circle glyphs are suitable for a precise and quick estimation of relative depth. We expect that the glyphs perform similar for representation of scalar fields in general. A deficiency of the evaluation is that we only tested for scenes with two glyphs visible. In a situation where multiple glyphs (10-15) are shown to allow for a general overview of the data in question, the results might be different. In this case the supporting lines and anchors probably overlap in many locations and it becomes very hard to obtain a satisfactory estimation of depth.

The example in Fig. 5 shows, that our glyph can be modified with low effort to represent three scalar magnitudes. However, the performance of the these extended glyphs has to be assessed in another evaluation. Furthermore, a subsequent study should investigate the discretization of the (so far) continuous mappings to geometrical features. E.g., the depth dependent glyph size could be mapped to ordered bins, so that an ordering of multiple glyphs is perceivable, even if the actual (un-binned) depth differences are very low. Apart from this, our glyphs embody a simple to implement, effective and flexible representation for scalar values on arbitrary meshes.

Acknowledgements This project was partly funded by the DFG: LA 3855/1-1 and HA 7819/1-1. We would like to thank the subjects of our evaluation for their contribution.

References

[BHFN09] Bichlmeier C., Heining S. M., Feuerstein M., Navab N.: The virtual mirror: a new interaction paradigm for augmented reality environments. *IEEE Transactions on Medical Imaging 28*, 9 (2009), 1498–1510. 2

[BKC*13] Borgo R., Kehrer J., Chung D. H. S., 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. 3, 5

[CWW13] Crane K., Weischedel C., Wardetzky M.: Geodesics in heat: A new approach to computing distance based on heat flow. *ACM Transactions on Graphics*, 3 (2013), 10. 3, 6

[EAT07] Elmqvist N., Assarsson U., Tsigas P.: *Employing Dynamic Transparency for 3D Occlusion Management: Design Issues and Evaluation*. Springer Berlin Heidelberg, 2007, pp. 532–545. 2

[Gib50] Gibson J. J.: *The Perception Of The Visual World*. Boston: Houghton Mifflin, 1950. 2

[HZH*09] Hansen C., Zidowitz S., Hindennach M., Schenk A., Hahn H., Peitgen H.-O.: Interactive Determination of Robust Safety Margins for Oncologic Liver Surgery. *International Journal of Computer Assisted Radiology and Surgery 4*, 5 (2009), 469–474. 1

[HZS*14] Hansen C., Zidowitz S., Stravrou G., Oldhafer K. J., Hahn H. K.: Impact of model-based risk analysis for liver surgery planning. *International Journal of Computer Assisted Radiology and Surgery 9*, 2 (2014), 473–480. 1

[IFP97] Interrante V., Fuchs H., Pizer S. M.: Conveying the 3d shape of smoothly curving transparent surfaces via texture. *IEEE Transactions on Visualization and Computer Graphics 3*, 2 (1997), 98–117. 2

[KCPS15] Knöppel F., Crane K., Pinkall U., Schröder P.: Stripe patterns on surfaces. *ACM Transactions on Graphics 34*, 4 (2015), 39:1–39:11. 7

[KOCC14] Kersten-Oertel M., Chen S. J. S., Collins D. L.: An evaluation of depth enhancing perceptual cues for vascular volume visualization in neurosurgery. *IEEE Transactions on Visualization and Computer Graphics 20*, 3 (2014), 391–403. 2, 7

[KSTE06] Kersten M., Stewart J., Troje N., Ellis R.: Enhancing depth perception in translucent volumes. *IEEE Transactions on Visualization and Computer Graphics 12*, 5 (2006). 2

[KWTM03] Kindlmann G., Whitaker R., Tasdizen T., Moller T.: Curvature-based transfer functions for direct volume rendering: Methods and applications. In *Proceedings of the 14th IEEE Visualization 2003* (2003), pp. 67–. 7

[LAK*98] Laidlaw D. H., Ahrens E. T., Kremers D., Avalos M. J., Jacobs R. E., Readhead C.: Visualizing diffusion tensor images of the mouse spinal cord. In *Proceedings of the Conference on Visualization '98* (1998), pp. 127–134. 2

[LLH17] Lawonn K., Luz M., Hansen C.: Improving spatial perception of vascular models using supporting anchors and illustrative visualization. *Computers and Graphics 63* (2017), 37–49. 2, 3, 4, 5, 7, 9

[LLPH15] Lawonn K., Luz M., Preim B., Hansen C.: Illustrative visualization of vascular models for static 2D representations. In *Lecture Notes in Computer Science*, vol. 9350. Springer International Publishing, 2015, pp. 399–406. 2, 3, 4, 7

[LSHL16] Lichtenberg N., Smit N., Hansen C., Lawonn K.: Sline: Seamless Line Illustration for Interactive Biomedical Visualization. *Eurographics Workshop on Visual Computing for Biology and Medicine* (2016), 133–142. 2

[MSSD*08] Meyer-Spradow J., Stegger L., Döring C., Ropinski T., Hinrichs K.: Glyph-based SPECT visualization for the diagnosis of coronary artery disease. *IEEE Transactions on Visualization and Computer Graphics 14*, 6 (2008), 1499–1506. 2

[MVB*17] Meuschke M., VoÃ§ S., Beuing O., Preim B., Lawonn K.: Glyph-Based Comparative Stress Tensor Visualization in Cerebral Aneurysms. *Computer Graphics Forum* (2017). 2

[OHG*08] Oeltze S., Hennemuth A., Glasser S., Kühnel C., Preim B.: Glyph-Based Visualization of Myocardial Perfusion Data and Enhancement with Contractility and Viability Information. In *Eurographics Workshop on Visual Computing for Biomedicine* (2008), The Eurographics Association, pp. 11–20. 2

[PBC*16] Preim B., Baer A., Cunningham D., Isenberg T., Ropinski T.: A Survey of Perceptually Motivated 3D Visualization of Medical Image Data. *Computer Graphics Forum 35*, 3 (2016), 501–525. 2

[RHD*06] Ritter F., Hansen C., Dicken V., Konrad O., Preim B., Peitgen H. O.: Real-time illustration of vascular structures. *IEEE Trans. Vis. Comput. Graphics 12*, 5 (2006), 877–884. 1

[RHFL10] Reichelt S., Häussler R., Fütterer G., Leister N.: Depth cues in human visual perception and their realization in 3d displays. In *Proc. SPIE* (2010), vol. 7690, pp. 76900B–76900B–12. 1

[ROP11] Ropinski T., Oeltze S., Preim B.: Survey of glyph-based visualization techniques for spatial multivariate medical data. *Computers and Graphics 35*, 2 (2011), 392–401. 2, 5

[RRRP08] Rieder C., Ritter F., Raspe M., Peitgen H. O.: Interactive visualization of multimodal volume data for neurosurgical tumor treatment. *Computer Graphics Forum 27*, 3 (2008), 1055–1062. 2

[RSH06] Ropinski T., Steinicke F., Hinrichs K.: Visually supporting depth perception in angiography imaging. In *Smart Graphics: 6th International Symposium, SG 2006* (2006). 2

[RSMS*07] Ropinski T., Specht M., Meyer-Spradow J., Hinrichs K., Preim B.: Surface glyphs for visualizing multimodal volume data. *Proceedings of the 12th International Fall Workshop on Vision Modeling and Visualization D* (2007), 3–12. 2, 5

[Ste87] Steenblik R. A.: The chromostereoscopic process: A novel single image stereoscopic process. In *Proc. SPIE* (1987), vol. 0761, pp. 27–34. 2

[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

[VPF15] Vasilakis A. A., Papaioannou G., Fudos I.: K-buffer: An efficient, memory-friendly and dynamic K-buffer framework. *IEEE Transactions on Visualization and Computer Graphics 21*, 6 (2015), 688–700. 2

[War02] Ward M. O.: A Taxonomy of Glyph Placement Strategies for Multidimensional Data Visualization. *Information Visualization 1*, 3-4 (2002), 194–210. 2

[ZHX*11] Zhang L., He Y., Xia J., Xie X., Chen W.: Real-time shape illustration using laplacian lines. *IEEE Transactions on Visualization and Computer Graphics 17*, 7 (2011), 993–1006. 2