Computers & Graphics (2019)

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

EvalViz – Surface Visualization Evaluation Wizard for Depth and Shape Perception Tasks

Monique Meuschke[a,b,*], Noeska N. Smit[c,d], Nils Lichtenberg[e], Bernhard Preim[a,b], Kai Lawonn[e]

[a]*Department of Simulation and Graphics, University of Magdeburg, Universitätsplatz 2, 39106 Magdeburg, Germany*
[b]*Research Campus STIMULATE, Universitätsplatz 2, 39106 Magdeburg, Germany*
[c]*Department of Informatics, University of Bergen, Thormøhlensgt. 55, 5020 Bergen, Norway*
[d]*Mohn Medical Imaging and Visualization Centre, Haukeland University Hospital, Jonas Lies vei 65, 5021 Bergen, Norway*
[e]*University of Koblenz - Landau, Universitätsstraße 1, 56070 Koblenz, Germany*

ARTICLE INFO

Article history:
Received June 14, 2019

Keywords: Computers and Graphics, Formatting, Guidelines

ABSTRACT

User studies are indispensable for visualization application papers in order to assess the value and limitations of the presented approach. Important aspects are how well depth and shape information can be perceived, as coding of these aspects is essential to enable an understandable representation of complex 3D data. In practice, there is usually little time to perform such studies, and the establishment and conduction of user studies can be labour-intensive. In addition, it can be difficult to reach enough participants to obtain expressive results regarding the quality of different visualization techniques.

In this paper, we propose a framework that allows visualization researchers to quickly create task-based user studies on depth and shape perception for different surface visualizations and perform the resulting tasks via a web interface. With our approach, the effort for generating user studies is reduced and at the same time the web-based component allows researchers to attract more participants to their study. We demonstrate our framework by applying shape and depth evaluation tasks to visualizations of various surface representations used in many technical and biomedical applications.

1. Introduction

In the past decade, many surface visualization techniques have been developed in order to support efficient exploration, analysis, and interpretation of data for a variety of domains. To demonstrate the benefits of novel techniques, user studies can be performed to investigate task performance and user experience. However, designing and creating user studies is a time- and resource-intensive process where problems such as lack of objectivity and reproducibility may arise.

In this paper, we focus on supporting the evaluation of surface visualization techniques, which are used in a variety of technical and biomedical applications. In medical visualization, a typical scenario is to depict blood vessels using surface visualization techniques to support the analysis of vascular diseases. Moreover, vascular structures are also important in case of other pathologies, where damage to vessels needs to be minimized. Due to their elongated and branching character, they present perceptual challenges, in particular in case of high curvature or partial occlusion. Effective visualization techniques can help to better understand the shape of anatomical and pathological structures and their spatial relationships.

Basic requirements due to the visualization of complex surface models are that spatial relationships and the distances between structures should be made apparent. To check whether a new visualization technique allows for adequate or improved depth and shape perception of 3D surfaces compared to existing methods, participants of a user study are asked to perform

*Corresponding author: Tel.: +49-391-675 2759; fax: +49-391-671 1164 ;
e-mail: meuschke@isg.cs.uni-magdeburg.de (Monique Meuschke)

specific judgment tasks. However, there are three major issues when performing such studies. First, the tasks have to be created manually by selecting suitable landmarks and camera positions, which is a time-consuming, and subjective process. Second, it is necessary to capture specific information throughout a study, including the required time and accuracy of participant task completion. Third, the interactive evaluation application has to be made available for a wide range of study participants.

To ease the evaluation process for surface visualization techniques, we previously presented a framework that supports automatic generation of web-based user studies to evaluate depth perception in vascular surface visualizations [1]. In this paper, we extend the previous work by two fundamental aspects. First, we integrate a method for generating task-based user studies to evaluate shape perception in addition to depth perception. Second, we overcome the limitation to vascular surfaces by extending our framework to arbitrary surfaces. The resulting experiments are performed via a web interface, and we provide automatic statistical reporting of the results. Thus, the effort to create user studies is reduced, and the web-based solution helps researchers to attract more participants by offering remote access. We demonstrate our framework on the basis of depth and shape judgment tasks in visualizations of different surfaces. In summary, we make the following contributions:

- We extend our framework focusing on the evaluation of depth perception described in our previous work to additionally include the evaluation of shape perception during the study preparation, conduction and reporting.
- We extend the generation of task-based experiments on the basis of vascular surface models to arbitrary 3D surfaces.
- Automatic preparation of web-based user studies using the generated tasks, including statistical analysis and reporting of the results.

2. Related Work

The work associated with our approach includes concepts for evaluating scientific visualizations. In addition, visualization techniques to support depth and shape perception, especially in the biomedical field, will be presented.

2.1. Perceptual Experiments in Visualization

The goal of perception-based experiments is to determine the relationship between a physical stimulus, and perceptions of its effects [2]. They provide empirical evidence of the effectiveness of a new technique by examining aspects of human perception. For this purpose, a *stimulus* is presented to participants who are asked to perform a certain task [3]. In the context of surface visualizations, images or videos are usually employed as *stimuli*, which are generated either with different visualization methods or with different parameter values for a method. Moreover, two types of variables are distinguished: *independent* and *dependent* variables. An independent variable, also called *factor*, represents the object to be examined, which is generated by systematic alteration of the stimuli. A dependent variable measures an effect in the behavior of the participant that is to be influenced by the independent variable.

These effects can be measured quantitatively and qualitatively. Quantitative measurements involve the collection of objective data in the form of numerical values. Such measures are used as input for a statistical analysis to validate how effective a factor is. In contrast, qualitative measurements usually include nominal data in the form of oral reports that examine subjective aspects such as the participant's preferences or the acceptance of a factor.

We focus on perception-based experiments by collecting quantitative and qualitative information. In scientific visualization, these types of experiments can be used to test how well interesting structures can be perceived and compared to demonstrate the benefits of a novel method [4]. For this purpose, common tasks such as comparing, associating, discriminating, ranking, grouping, correlating, or categorizing can be performed by the user [5]. With these tasks, a multitude of visualization-based research questions can be examined, ranging from abstraction [6] to the perception of spatial relationships [7] to decision-making [8]. Perceptual experiments are rarely performed as they require extensive preparation [9].

Therefore, software solutions were developed to support the preparation of user studies. In neuroscience, the *PsychToolbox* [10] is a widely used set of functions to generate visual and auditory stimuli for performing cognitive experiments. *TouchStone* [11] is an open-source platform to support design, execution and analysis of human-computer interaction experiments. Aigner et al. [12] proposed *EvalBench*, a software library to support the evaluation of lab-based experiments. Okoe and Jianu introduced *GraphUnit* [13], a framework to evaluate graph visualizations using crowdsourcing. Englund et al. [14] developed a web-based system to prepare and conduct quantitative evaluations of scientific visualizations. While they integrated an automatic sampling of parameter ranges influencing the results, a calculation of suitable viewpoints and automatic label placement to evaluate depth perception is missing. To the best of our knowledge, there is no framework that allows a simple evaluation of existing and novel surface visualization techniques through the automatic generation of task-based experiments on depth and shape perception.

2.2. Depth Perception

The investigation of depth perception has become increasingly important in visualization research, especially for biomedical applications [15]. The visual coding of depth determines how precisely and quickly complex 3D scenes can be perceived.

There are *monoscopic* and *stereoscopic* depth cues. For the former, an open eye is sufficient to view the scene, where shadows, perspective projection, partial occlusion, and shading are important cues. Stereoscopic cues are a natural way to provide depth information via visual perception using both eyes. However, there are situations where static images are desired. Examples are print-outs or cases where dynamic visualizations would require a high degree of interaction (e.g., during a surgery). In these cases, additional depth cues are essential. Further sub-

categories of depth cues are motion-, surface- and illumination-based cues. Common techniques are color scales, glyphs or illustrative line drawings [16, 17]. These cues can help to reconstruct the 3D structure of an object perceived by projection onto a 2D image plan.

The analysis of complex data such as biomedical information requires an appropriate visualization of spatial relationships. *Chromadepth* [18] uses the visible color spectrum to encode the depth often applied to vascular structures. In contrast, pseudo-chromadepth [19] uses only a color palette from red to blue inspired by the scattering of light in the atmosphere. Red colors are perceived as closer than blue colors. Similar to the chromadepth is the air perspective, where distant objects are perceived with less contrast [20]. Kersten-Oertel et al. [21] evaluated several depth cues for vascular visualizations in which air perspective and pseudo-chromadepth exceeded stereopsis.

Applying chromadepth to a 3D surface makes it difficult to additionally encode attributes on the surface. Therefore, Behrendt et al. [22] used the Fresnel term to combine chromadepth and additional parameters. Illustrative techniques were also used to improve depth perception. Ritter et al. [23] used illustrative shadows to emphasize the distance between vessels. To further support depth perception of vascular structures, Joshi et al. [24] used *toon shading* and *halos*.

However, the visualization is not limited to what a 3D model looks like. Rendering supporting geometry also allows to interpret a 3D scene. The virtual mirror introduced by Bichlmeier et al. [25] adds a second perspective to solve problems with occluding geometry. An additional shadow plane supports perception of depth in a natural way [7]. Reference objects whose depth is easy to interpret can also aid the perception of complex structures. Lawonn et al. [26] combined a cylindrical cutaway view with supporting anchors to provide depth cues. Lichtenberg et al. [27] used camera-oriented disc-shaped glyphs to represent depth relations at vessel endpoints. Recently, Kreiser et al. [28] introduced *Void Space Surfaces*, where the empty space between vessel branches is used to encode depth.

These techniques have their strengths and weaknesses. Usually, methods that are able to convey the depth distribution of an entire model fail to detect subtle depth differences. For example, the pseudo-chromadepth easily covers an entire mesh, but the perception of small differences is challenging due to the smooth color changes. Information at discrete surface points can be visualized using glyphs [29]. The derivation of information about surface positions that are not covered by glyphs can require a high cognitive effort. We are not aware of any comprehensive study on these aspects that could provide guidance for task-oriented decisions. However, the proposed framework supports the preparation of such studies.

2.3. Shape Perception

While evaluating depth perception is straightforward, the quality assessment concerning shape perception is quite challenging. In general, shape perception means the overall impression of the model, including spatial relations between structures, e.g., distances of vessel end points, as well as to locally get a spatial impression, e.g., is a certain part on a surface more roundish than others. In this work, we focus on the latter aspect.

Usually, to evaluate how well shape can be perceived, the user is asked to determine the surface normal on a model. In case the user can correctly estimate the surface normal based on a specific rendering technique, the hypothesis that she/he has a good spatial impression is valid for this visualization [30]. Stevens [31] introduced the task of placing gauge figures to assess the shape perception, which was used to improve or justify visualization techniques. Koenderink et al. [32] employed gauge figures on photographs. The user mentally constructs a surface that matches the photographs and is then asked to adjust a gauge that corresponds to the surface normal. Sweet and Ware [33] evaluated parallel lines on surfaces. They extracted surfaces from height fields and applied Phong shading to them. The surfaces were additionally covered with different line textures, which are aligned in certain directions. The task was to orient a gauge such that it fits the mentally imagined surface normal. A notorious problem is the scale of gauge figures: since they occlude the surface exactly where its normal is estimated, they should be small. However, a small gauge is hardly recognizable. Sweet and Ware thus use an additional display to enlarge the gauge with the same orientation as the small gauge embedded in this figure. Finally, they analyzed for which line direction the angular deviation could be reduced. O'Shea et al. [34] evaluated the suitability of several light conditions to perceive shape correctly. For this purpose, they used different models and different light positions. Again, the user was asked to adjust the gauge concerning the surface normal. It was confirmed that shape perception works best if the light position is above the view direction. Bernhard et al. [35] compared monoscopic and stereoscopic displays with respect to shape perception by measuring the deviations of slant angles to a ground truth. For this purpose, the gauge figure task was used applied to various well-defined objects.

Regarding illustrative techniques, Cole et al. [36] performed a user study to explore how well line drawings communicate the shape of a surface. Different line renderings were applied to 3D surface representations. The general idea of such approaches is to convey the shape of a surface by just covering it by a small number of lines. However, the use of a few lines generally leads to an enormous loss of information, which raises the question whether certain techniques nevertheless enable shape perception. To investigate this, users were again asked to adjust a gauge corresponding to the surface normal. Šoltészová et al. [37] presented a novel technique to enhance important structures by employing chromatic shadows. Similar to previous studies, accuracy of shape perception was evaluated with a gauge task. Baer at al. [8] evaluated a visualization technique to visualize blood flow data in the context of a surface depiction representing the morphology of an aneurysm. The visualization technique was designed to retain the perception of shape by simultaneously depicting hidden structures through additional transparency. The challenge was to adequately represent both, the surface and the internal blood flow. Again, the spatial impression was evaluated with gauge tasks.

Inspired by the previously mentioned evaluations, we extended our framework to create task-based evaluation concerning shape perception by manually adjusting a gauge.

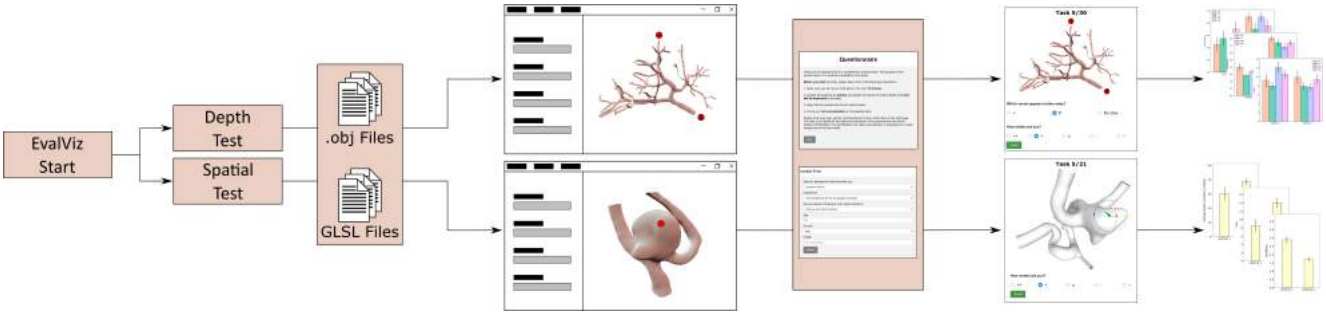

Fig. 1. Our framework asks the creator whether to create a depth or a spatial test. Afterward, surfaces, shaders, and user-defined parameters are provided. Based on this, it generates labeled as well as unlabeled images as stimuli for depth and shape perception studies, respectively. Then, a web-based study is generated, and the results are reported automatically via statistical summary charts.

3. Requirements Analysis

The current state of the art in quantitative user studies for the evaluation of surface visualizations has motivated us to develop the proposed framework. It is based on two observations by Isenberg et al. [9] which explain the lack of quantitative evaluations in scientific visualization. First, quantitative user studies require an enormous expenditure of time and resources. Secondly, it is difficult to acquire a sufficient number of participants for a meaningful study, especially when these participants need to have domain knowledge.

Based on the guidelines by Forsell [38], Englund et al. [14] identified three main phases in conducting quantitative evaluations using crowdsourcing for scientific visualizations. In the first phase, the study is prepared by generating experiments. In the second phase, the study is conducted and response data is collected, which requires a sufficient number of participants. Finally, the data is analyzed and the results are reported.

Based on these phases, we introduce the framework *EvalViz* – Evaluation Visualization Wizard – to support visualization researchers in all three stages of perceptual task-based studies. We focus EvalViz on supporting the evaluation of surface visualization methods via depth and shape judgment tasks. It is designed for web-based user studies, which makes it easier to obtain a sufficient number of participants since the study can be conducted at any place and time.

In the aforementioned three phases of performing quantitative evaluations, the first phase involves the generation of depth and shape judgment tasks. Concerning depth perception, such a task could, for example, consist of determining which of two marked positions in an image is closer to the viewer. In contrast, a typical task for shape perception is to adjust a gauge, which should estimate the surface normal at a specific point on the surface. Besides the selection of predefined visualization techniques to encode depth or shape, the creator of the study can add novel visualization techniques that should be considered for task generation. To support the study creator in this task, the following requirements have to be met:

- The framework should be able to generate an arbitrary number of tasks and should support user-defined visualization techniques.
- The framework should allow for custom surface shader specification to evaluate novel visualization techniques.
- The framework should automatically generate *representative* images for depth and shape judgment tasks as stimuli.

In the second phase, the generated images are used as input for a web interface to conduct the experiments. The participants can use their own web browser to perform the study instead of using a system installed in a local environment such as a lab. In order to support the study creator in the second phase, the following requirements have to be met:

- The framework should allow participants to take part in the study via any web browser.
- The framework should record task performance via answers given by the participants, as well as measured information such as time.

After conducting the experiment, the final phase, analysis and reporting of results, needs to be supported by our framework as well. Here, the requirements are:

- The framework should report results via the automatic generation of textual summary reports and charts which describe basic statistical information.
- The framework should allow exporting of study results for further detailed statistical analysis in dedicated statistical analysis frameworks.

4. EvalViz

This section presents *EvalViz* - a framework to prepare, conduct, and analyze task-based user studies concerning depth and shape perception. To this end, we analyzed the previous routine how such studies are manually carried out and identified three major steps for both types of perception:

1. Development of novel surface visualization techniques to enhance depth or shape perception.
2. Re-implementation of existing visualization methods.
3. Conducting a user study to assess the impact of new techniques.

In the first step a novel technique is developed. Applying illustrative techniques [7, 23, 26], glyphs [27], or add an additional layer of information [22, 28] can improve depth perception, whereas methods aiming at improving shape perception of surfaces use, e.g., Phong shading [34] or line drawings [33]. The last step reveals the potential benefits of the novel technique. Here, a scene is generated showing the surface representation. This scene is generated with different visualization techniques. Afterward, concerning depth perception, two labels are placed near certain positions on the surface. Then, the user has to decide which label appears closer to him. In contrast, for the shape judgment, an adjustable gauge is generated at a certain surface position. Based on this, the user has to estimate the surface normal at this location. In combination with the ground truth for both types of perception, the study analyzes the task performance of the evaluated methods. This raises several questions, e.g., where the labels or gauge will be placed, how the study will be conducted and how the results will be evaluated.

Based on these observations, we developed an evaluation system consisting of three major components, see Figure 1:

1. Preparation of depth or shape judgment tasks (see Section 4.1).
2. Generation of a web-based user study (see Section 4.2).
3. Statistical analysis and reporting of study results (see Section 4.3).

For the preparation of depth and shape judgment tasks, we developed a framework written in *C#* and *OpenTK* - a wrapper for *OpenGL*. As the main input, the creator of the study loads geometry files of desired surfaces. Based on the generated tasks, a web interface is built up that consists of two main parts: a front-end which presents the user study to the participants, and a back-end that controls the recording of the task results. To create the web interface, and to record the participants' answers as well as to measure the time for completing a task, HTML and PHP are used. The task performance results are then stored in a CSV file which will be passed to the final analysis step. Here, the user task performance of the chosen visualization techniques is investigated, and the results are reported to the study creator.

4.1. Preparing a Perception-Based Study

This section presents the automatic generation of task-based experiments to evaluate depth or shape perception. The method consists of the following three steps for both study types:

1. Determining study creator input (Section 4.1.1).
2. Constructing judgment tasks concerning depth and shape perception (Section 4.1.2).
3. Placing labels (Section 4.1.3).

Based on the prepared tasks, the study can be conducted and analyzed afterward, which is explained in Section 4.2 and Section 4.3.

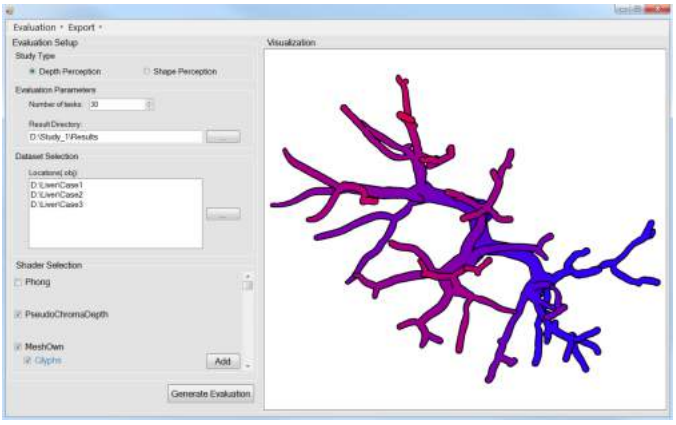

Fig. 2. User interface to set up a perception study. The creator has to define settings such as the study type, number of desired tasks as well as the desired data sets and visualization techniques.

4.1.1. Creator-Defined Input Parameters

First, the creator has to define general evaluation criteria via the user interface, see Figure 2. Initially, a selection of the type of perception to be evaluated must be made. The remaining settings are the same for both depth and shape perception studies. This includes a decision of how many tasks should be generated. Moreover, a directory has to be selected where the resulting stimuli images are stored. Another conceivable setting would be the choice of study design. There are two design categories: *Within-* and *Between-Group* studies. The *Between-Group* design uses different participants for each condition to be evaluated. When every participant evaluates all conditions, this is called a *Within-Group* design and this is the standard in perception experiments. The first design requires that each condition is evaluated by the same number of participants. Therefore, the creator would have to know the number of participants before starting the web study, which is likely not the case. Thus, we decide to design only *Within-Group* studies. However, with this design, we have to pay attention to the task order to avoid memory effects.

Next, the creator has to select the data sets that should be considered for task generation. By clicking the button next to the data set selection, a folder dialog is opened and the creator can select the directory of a desired data set. If the data set is processed for the first time, an algorithm is applied to detect candidates on the surface, where the perception tasks should be performed (see Section 4.1.2). Then, the indices of the detected points are automatically stored as a text file in the same folder as the surface mesh and can be used in follow-up perception studies.

Finally, the creator has to select the desired visualization techniques to evaluate. The framework provides Phong shading [39] and pseudo-chromadepth [19] which can be selected, see Figure 2. These are baseline techniques against which new techniques should be compared. Additional surface visualization techniques can be integrated by clicking the "Add" button. This opens another file dialog to select shader files for the surface depiction. Moreover, we provide the possibility to select shaders for rendering glyphs [27, 40], see Figure 3. Information

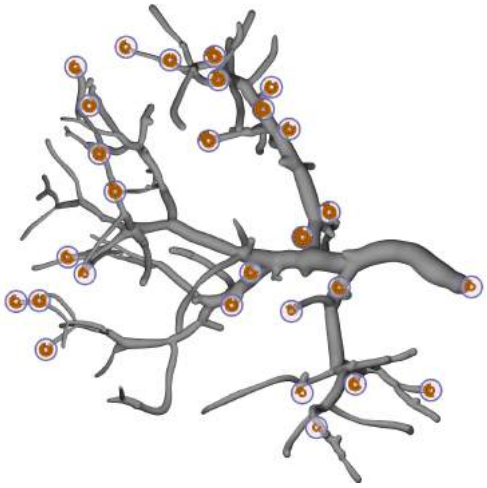

Fig. 3. ***EvalViz*** **is able to consider glyph-based techniques. Here, the method by Lichtenberg et al. [27] was used, where glyphs are shown for a subset of the detected endpoints on a liver vessel tree.**

about shaders that have been loaded once is displayed for selection the next time the framework is started. If glyphs are available for rendering, a second checkbox appears in the display. If the setup is finished, clicking the button "Generate Evaluation" starts the automatic calculation of the task-representing image stimuli, which is explained in the next section.

4.1.2. Construction of Judgment Tasks

To construct judgment tasks, we need to define locations on the surface, where the assessment is performed. We call them judgment points. Thus, we first need to acquire a set of candidates. In the second step, based on the set of candidates, we need to select the actual judgment points, which are discussed in the appropriate section and rely on a specific task. Finally, a viewpoint needs to be calculated, which serves as a basis for the online evaluation. In short, the construction depends on the following steps, which will be explained in more detail in the next section:

1. Candidate detection on the surface.
2. Selection of a subset of candidates, called judgment points.
3. Calculation of viewpoints.

Candidate Detection. The suitability of a surface position for the assessment of shape or depth perception is application-specific. Therefore, we offer the creator several options where the candidates should be placed on the surfaces. For example, if the creator wants to test depth perception for vessel trees, ideal candidates are the endpoints of the trees [7, 23]. To detect the vessel endpoints, we employ the method by Lichtenberg et al. [41]. Thus, the first option the creator can choose is *convex features*. In general, the algorithm of Lichtenberg et al. [41] works for convex structures and is therefore a perfect candidate for this option, see Figure 4. The second option to choose is *negative Gaussian curvature*. This option serves as a way to select candidates on saddle regions. Note, that with this option the creator obtains regions on the surface instead of single points. Nevertheless, we use all points within the region as candidates because judgment points are selected in a post-processing step, which prevents that neighbored points are chosen. Next option to choose is *brushed region*. Here the creator is asked to brush surface regions, where the corresponding vertices are used as candidates. Finally, the last option is *all*, which means that all surface vertices are candidates.

Judgment Point Selection.

After we determine possible candidates, we need to select judgment points for placing labels. For this, we distinguish between judgment point selection for:

- depth judgment tasks, and
- shape judgment tasks.

Depth judgment tasks:

For a meaningful study, tasks with different degrees of difficulty (DoD) are needed. This depends on the distance of two candidates in depth and screen space. The obvious choice would be to select two candidates randomly, but following Lawonn et al. [7], we group the candidates according to two distance measures. A pair of two candidates with a given camera setting is an element of the set $\mathscr{C} = \{NN, NF, FN, FF\}$. The letters N, F stand for 'near' and 'far', respectively. The first entry of the pair of letters relates to the distance of the candidates in screen space, i.e., the Euclidean distance after an orthogonal projection of both points onto the view plane. The second entry of the pair of letters relates to the distance of the candidates in depth, i.e., the Euclidean distance after an orthogonal projection of both points onto the view plane's normal. For example, in a pair of NF-categorized candidates, the distance in screen space is near, but the depth distance is far (see Figure 15).

Next, we need to define for what distances a pair of points is to be used as 'far' or 'near'. For this purpose, we calculate the Euclidean distances of all pairs of candidates $(\mathbf{p}_i, \mathbf{p}_j)$ and determine the maximum occurring distance $D = \max_{i,j} \|\mathbf{p}_i - \mathbf{p}_j\|$. We take the 90 % quantile of D, defined as D', to exclude surface parts with a large distance to all other parts. Otherwise, only a few pairs could be considered for the FF condition, which could be too few depending on the required amount of tasks. If the distance of two candidates in screen space (or depth) is less than $D'/2$, we assign N.

Lawonn et al. [7] assigned N for distances less than half of diagonal of the screen size. Lichtenberg et al. [27] used the diagonal of the bounding box of the geometry in screen space. Again, if the distance was less than half, N was assigned. We used the distance D' to be consistent in the definition and as it simplifies further calculations.

Depending on the Euclidean distance of a pair of points, we can exclude them of being a certain element of $\mathscr{C}$. Let d_s be the Euclidean distance of two points in screen space (on the view plane), d_d be the Euclidean distance in depth and d_E be the Euclidean distance in 3D world space, then: $d_E^2 = d_s^2 + d_d^2$, see Figure 5. This yields the distinction, see also Figure 6:

$$d_E = \begin{cases} < \frac{1}{2}D' & \text{then } NN \\ \in [\frac{1}{2}D', \frac{\sqrt{2}}{2}D') & \text{then } NF, FN, \text{ or } NN \\ \geq \frac{\sqrt{2}}{2}D' & \text{then } NF, FN, \text{ or } FF. \end{cases} \quad (1)$$

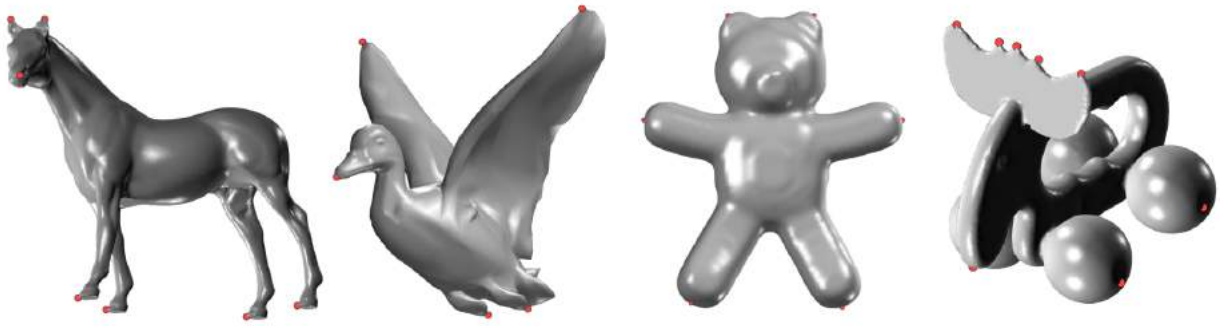

Fig. 4. Exemplary results for the detection of convex features for various surface models using the method by Lichtenberg et al. [41]. The features are represented by red spheres.

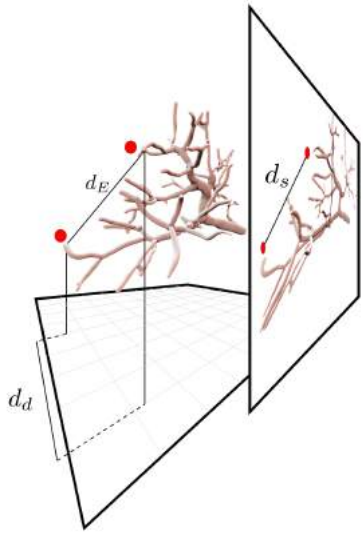

Fig. 5. The distance d_E corresponds to the Euclidean distance, d_d to the depth distance, and d_s to the screen space distance.

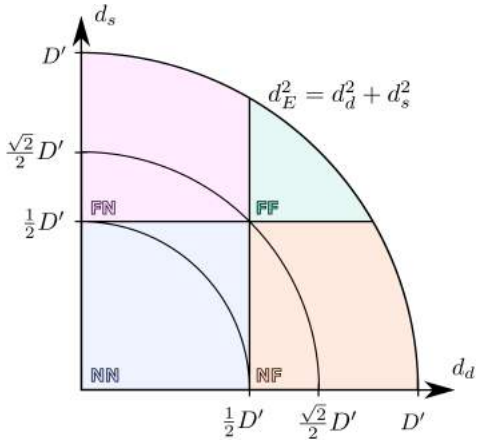

Fig. 6. The conditions for which the screen and depth distances yield the Euclidean distance and, therefore, the categories.

Therefore, if we need to determine a scene for the category *NN*, we can randomly pick a pair with a distance smaller than $\frac{\sqrt{2}}{2}D'$.

Shape judgment tasks:

In comparison with depth perception tasks, the computation of shape perception tasks is more straightforward. All identified candidates determined on the basis of the input of the creator serve as judgment points. In related work, we could not identify other strategies to select judgment points other than random selection. However, we have limited this by providing user input to select suitable areas, e.g., by brushing.

Viewpoint Calculation. The viewpoint calculation must also be differentiated w.r.t.:

- depth judgment tasks, and
- shape judgment tasks.

Depth judgment tasks:

For each category $\mathscr{C}$, we have to determine as many scenes as required tasks (see Figure 7) for generating a scene. Just picking a pair could lead to occlusion of a candidate within a scene. Thus, we need to calculate a viewpoint such that:

1. Occlusions of the pair of candidates do not occur.
2. The category of $\mathscr{C}$ is kept.

To meet requirement 1, we limit camera movement to translations and rotations such that it does not violate requirement 2. We use an orthographic projection to avoid depth hints due to perspective distortion. The following calculations are done in camera space with a view vector $\mathbf{v} = (0,0,-1)^T$. Independent of the category, we translate the surface model such that the first endpoint $\mathbf{p}_i$ (of the pair of candidates $\mathbf{p}_i, \mathbf{p}_j$) lies in the origin. Then, we rotate the object around the origin with the rotation axis of $\mathbf{v} \times (\mathbf{p}_j - \mathbf{p}_i)$ such that $\mathbf{p}_j$ lies in the (x,y)-plane. Finally, a rotation around $\mathbf{v}$ is performed such that $\mathbf{p}_j$ lies on the x-axis. The new coordinates of $\mathbf{p}_i, \mathbf{p}_j$ are $(0,0,0)$ and $(d_E,0,0)$, respectively. Depending on the category, we determine random variables such that the basis configuration of the mesh is arranged. First, we describe the calculation of d_s, d_d such that the category is fulfilled, for which mathematical proofs are given in Section 8. Then, we describe the rotation.

◦ ***NN*:** For this case, the distance of the two points needs to fulfill $d_E < \frac{1}{2}D'$, see Figure 6. Then, we determine a uniform random variable r in the interval $[0, D'/2)$ and set $d_s = r$. This yields a depth distance of $d_d = \sqrt{d_E^2 - d_s^2} < D'/2$. Based on the distances, this results in the category *NN*.

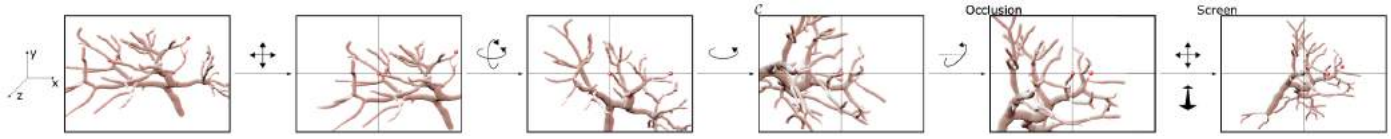

Fig. 7. First, we translate the object such that the first candidate lies in the origin, afterward consecutive rotations result in a desired category $\mathscr{C}$. Then, the occlusion problem is resolved and finally the object is fit to the screen.

∘ ***NF/FN*:** For this case, the distance of the two points needs to fulfill $d_E \geq \frac{1}{2}D'$. Without loss of generality, we assume that we want to determine *NF* first. The case *FN* is similar. We determine a uniform random variable r in the interval $\left[0, \sqrt{d_E^2 - D'^2/4}\right]$ and set $d_s = r$. This yields a depth distance of $d_d = \sqrt{d_E^2 - d_s^2} \geq D'/2$. Based on the distances, this results in the category *NF*. For the case *FN*, we change the distances and set $d_d = r$.
∘ ***FF*:** For this case, the distance of the two points needs to fulfill $d_E \geq \frac{\sqrt{2}}{2}D'$. We determine a uniform random variable r in the interval $\left[0, -D'/2 + \sqrt{d_E^2 - D'^2/4}\right]$ and set $d_s = D'/2 + r$. This yields a depth distance of $d_d = \sqrt{d_E^2 - d_s^2} \geq D'/2$. Based on the distances, this results in the category *FF*.

∘ **Rotation.** To finalize the configuration, we rotate the surface model around the y-axis with an angle of $\arccos \frac{d_s}{d_E}$. Now, we achieve a setting of the mesh such that a given configuration is fulfilled. Nevertheless, the current state does not guarantee that the surface candidates $\mathbf{p}_i, \mathbf{p}_j$ are visible. It is still possible that another surface part occludes one or both points. Considering three possible rotations around the main axes x, y, z, rotating around the x-axis and y-axis may violate the configuration $\mathscr{C}$, rotating around the z-axis does not affect the configuration, but occlusion will still occur. Rotating around the $\mathbf{p}_j - \mathbf{p}_i$ axis will not influence the configuration, but may solve occlusion problems. Therefore, we rotate around $i \cdot 2\pi/120$ with $i \in \{0, 1, \ldots, 119\}$. We iterate over i until we find a camera setting such that both judgment points are visible. For every rotation, we render the surface model and compare the fragment's depth value with the depth value of the judgment points. If they coincide, the points are visible. If the depth of the fragment is less than the depth value of a judgment point, it is occluded by a fragment in front of the point. Moreover, we randomly rotate the surface model around the z-axis with a random angle $[0, 2\pi)$ to avoid that the judgment points are always lying on the x-axis. Finally, we translate the mesh such that the whole model is seen in the scene. For this purpose, we determine the bounding box of the model in the screen and translate the midpoint to the origin. Afterward, we determine the maximum x, y-coordinate and scale the model such that it fits the screen.
Shape judgment tasks:
The goal is to generate an image of the surface model, where we have to export the normal vector for a visible point on the mesh as well as its resulting pixel position. This information is then used as input for performing the web-based user study, where the gauge is placed at the determined pixel position. Since the stimuli for shape perception only depend on a single position on the surface model, no distinction of different DoD is necessary.

The detection of suitable candidates, see Section 4.1.2, provides individual regions on the surface up to the entire surface, depending on what the study creator selects. In case of more than one region, we determine an initial view so that the visible area of the selected regions is maximized using the method by Meuschke et al. [42]. Similar to the depth judgment tasks, we use an orthographic projection. To ensure that the whole model is visible in the scene, we apply translation and scaling operations similar to the construction of the depth judgment tasks.

Besides the initial selection of a view on the model, we have to determine judgment points used for shape evaluation. The goal is to choose as many positions on the surface as the number of required tasks. The principle idea is to generate a sequence of pseudo-random numbers, whose length corresponds to the number of required judgment points. Then, these numbers are sorted in ascending order. During the rendering of the surface model, we use the generated sequence to determine judgment points in the fragment shader. For each visible pixel belonging to one of the selected surface regions, a counter is increased using the OpenGL *Atomic Counters*. If the value of the counter is equal to the value of the first sequence element, we store the actual pixel position as well as the normal of the corresponding surface position in a texture. For storing the normals, we use the principal concept of *normal mapping*, where per fragment normals are converted to RGB-values, which are written into a texture. Then, the atomic counter is further increased until its value is equal to the value of the next sequence element. This procedure is repeated for all elements of the sequence. Finally, pixel positions and normals of the judgment points are extracted from the textures on the CPU, where the normals are transformed back in the range of $[-1, 1]$ for the x-, y-, and z-component. This information is stored as CSV files as input for web-based study conduction. Figure 8 shows two exemplary results of surface positions, where the selection was based on high curvature (left) and random selection (right).

For the generation of the random number sequences, we have to define an interval. The lower boundary is set to 1, whereas the upper boundary is set to the number of visible pixel within the surface regions on the selected camera view. To determine the number of visible pixels, another render pass is needed. Thus, after the first render pass, we know the number of pixels belonging to the selected surface regions, which is used to define the upper boundary and in the second render pass, the determination of pixel positions and normals is done.

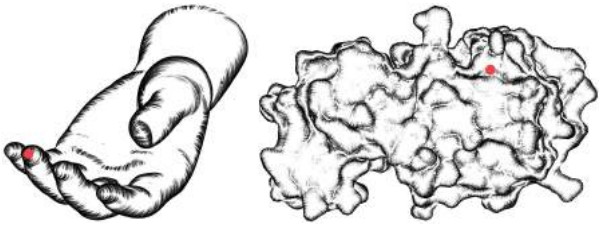

Fig. 8. Exemplary results for the selection of surface positions based on high curvature (left) and random selection (right) as input for the generation of shape judgment tasks.

4.1.3. Label Placement for Depth Judgment Tasks

In contrast to the shape perception task, where a gauge is placed on the surface to estimate the normal, the depth judgment task needs more consideration for the label placement. This is because during the study participants are asked to estimate which candidate appears closer. For this, the considered candidates need to be labeled, e.g., with circles that are denoted with '#' and '+'. The problem arises where to place the labels such that they are not:

1. Occluding the candidates.
2. Mistakenly assigned to an unintended candidate, e.g., in case of endpoints on a vessel tree, by the viewer.

Therefore, the creator can choose different label placement options:

- Void space labeling,
- bullseye labeling, and
- anchor labeling.

In the following, we discuss the various options.

Void Space Labeling. This method is optimized for branching structures such as vessel trees. Here, the judgment points are mostly the endpoints of the branches and thus, consistent to previous work, e.g., Lawonn et al. [7, 26], Lichtenberg et al. [27], we want to place the label next to the endpoint. For this purpose, we apply a gradient descent approach that automatically finds a reasonable position. On the final image, we place a circle with radius $r = 10$ on the judgment point $\mathbf{q} = (x, y)$ and count the pixels within the circle that contribute to the surface $p_v = \#\{\mathbf{p} \,|\, \|\mathbf{p} - \mathbf{q}\| \leq r \text{ and } \mathbf{p} \in Surface\}$ and the pixels that contribute to the background $p_b = \#\{\mathbf{p} \,|\, \|\mathbf{p} - \mathbf{q}\| \leq r \text{ and } \mathbf{p} \in Background\}$. Afterward, we shift $\mathbf{q}$ first in x-direction and then in y-direction (the shift is three pixels). With this, we apply the gradient descent and iteratively determine the new position:

$$\mathbf{q}_{i+1} = \mathbf{q}_i + \lambda \nabla p_b, \tag{2}$$

with $\mathbf{q}_0 = \mathbf{q}$ and $\lambda = \frac{2}{\|\nabla p_b\|}$. This scheme ensures to find a position that fulfills our first constraint that the candidate should not be occluded. Unfortunately, the second constraint is neglected with this. To consider this, we calculate the Voronoi diagram of the candidates in screen space. For each iteration, we test if the midpoint of the circle would leave the Voronoi area of its starting candidate. If this is the case, we restrict the movement to stay in the area, see Figure 9. The algorithm stops if the ratio of p_b and p_v exceeds 95%, specifically $\frac{p_b}{p_v + p_b} > 0.95$. In case the gradient descent method cannot find an appropriate position, we continue with another scene.

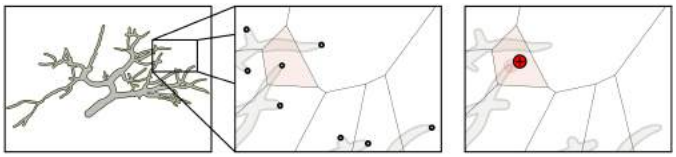

Fig. 9. For every candidate on the surface, we determine the Voronoi diagram. Afterward, a gradient descent approach is applied to find an appropriate position for the label.

Bullseye Labeling. In case the creator selects regions on the surface for judgment points, placing labels on the closest background would not make any sense. Therefore, we place the glyph directly on the position, but to avoid occlusion problems, we use *bullseye* glyphs. This means that we employ standard geometrical shapes, e.g., a square and a circle, but we only use the contour. Both shapes are placed on the surface and the web study is changed accordingly, see Figure 11 right.

Anchor Labeling. The last option places the labels at the boundary of the image. For every judgment point, the closest distance to the boundary in x, y direction is determined in screen space. Afterward, the label is placed and the judgment point is connected with the label by a thin line, see Figure 10.

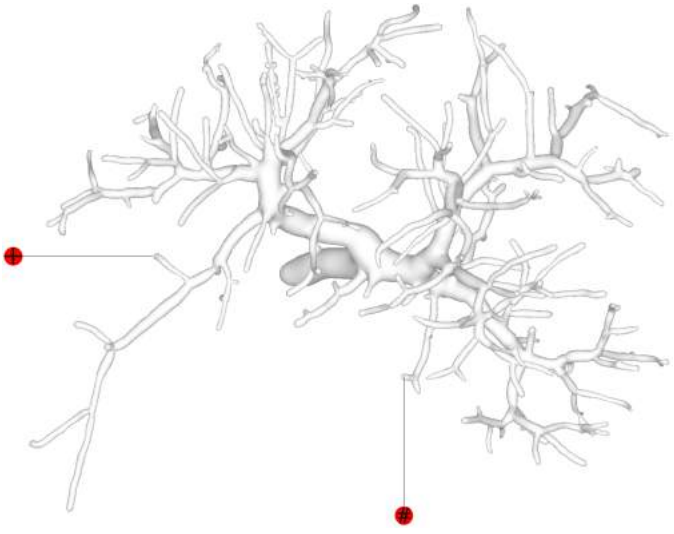

Fig. 10. Example of anchor labeling.

4.2. Conducting a Perception Study

After we can generate tasks in the form of images, we need to make sure that the order of the scenes with the different shading techniques bias the results depending on the type of study chosen. For this purpose, we save the images in a counterbalanced sequence such that the visualization techniques alternate, but the same scene occurs much later to avoid memory effects. To reach a large audience, we decide to offer a web-based study.

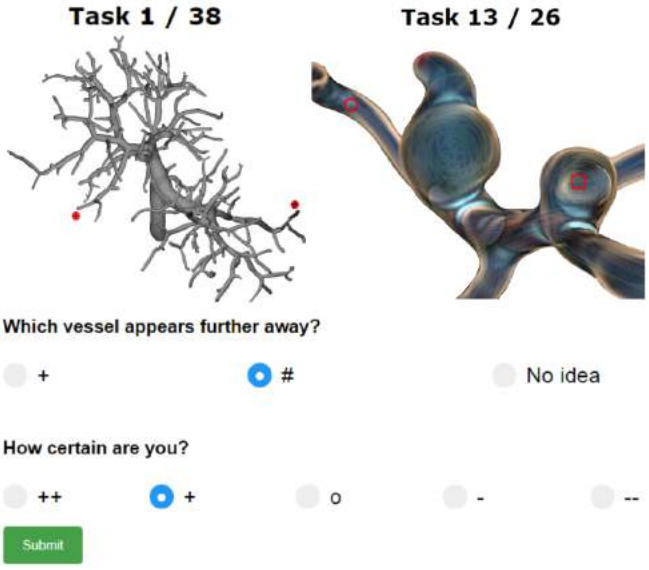

Fig. 11. Two exemplary images of the web-based conduction of a depth study. On the left a vessel tree is shown and on the right a study inspired by the visualization by Lawonn et al. [43] is depicted. Concerning the depth perception, the user has to specify which of the labeled candidates is closer to the viewer. Moreover, the degree of certainty should be selected.

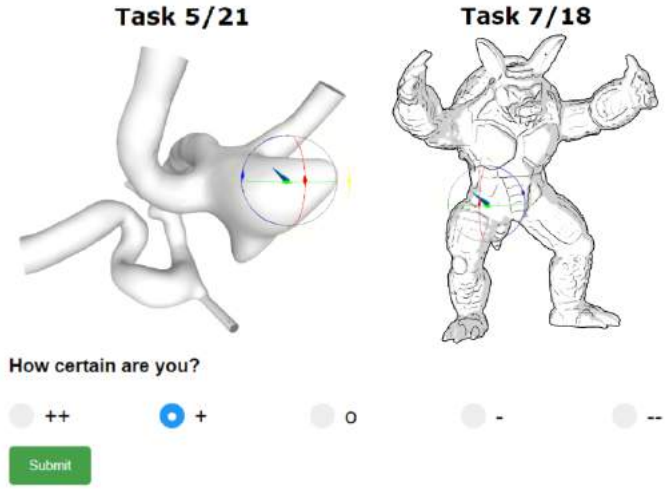

Fig. 12. Example for a spatial test inspired by the study by Baer et al. [8] (left) and Cole et al. [36] (right) with respect to the visualization techniques employed. The gauge has to be adjusted regarding the actual surface normal. Moreover, the degree of certainty should be selected.

For this, we provide PHP and HTML files that read the images from the folder initially selected by the creator, and generate the web interface for the evaluation. Moreover, we use WebGL to place the gauge on the generated image. In order to perform an evaluation, only the files provided by our framework and generated images must be copied to a server. Then, the web address can be shared, and participants can take part in the study. On the first page of the study, we ask participants to read the instructions carefully and take the time to participate in the evaluation. On the second page, we ask for information about the participant, e.g., age, gender, professional background, experience in scientific visualization, and color vision deficiencies. Concerning a depth perception study, we ask which label appears closer for each task, see Figure 11. Regarding, the perception of shape, the user adjusts the depicted gauge, see Figure 12. Then, we ask for the confidence of the participant in his answer or adjustment, respectively, from very uncertain to very certain using a five-point Likert scale ($--$,$-$,$\circ$,$+$,$++$). We measure the time it takes to answer which label appears closer. After all tasks are performed, we provide the participant with the opportunity to leave remarks on the study. Finally, we save the results of the evaluation in a CSV file as input for the subsequent reporting.

4.3. Reporting a Perception Study

After a user study is conducted, we provide automatic statistical analysis and reporting support for the evaluation results. For this purpose, our framework also provides an option to load an already performed study. To do this, the creator has to navigate to the folder of the desired evaluation session.

Given a CSV file containing the recorded study responses, as well as a ground truth CSV file with the correct answers to the tasks as input, we provide a *Python* script that automatically generates a summary to report on the study findings integrated into the framework. First, a summarizing text for the evaluation is generated consisting of the number of participants, their gender, age range, professional background, reported color vision deficiencies, and experience in scientific visualization. In the case of a depth perception study, charts show the mean and standard error in the ratio of correct answers, the reported confidence, and timings, categorized per scene type and visualization method. In the example in Figure 13, bar charts automatically created from synthetically generated depth perception evaluation results are visible, similar to the reporting charts used in the work by Lawonn et al. [7]. For a shape perception study, bar charts were automatically generated that show the average angle deviation of the normal estimates compared to the original surface normals, the reported confidence, and timings, categorized per visualization method, see Figure 14. After interpreting the results presented in the bar charts, users can carry out a further detailed statistical analysis either via Python, or dedicated software such as *R* or *SPSS* by simply importing the CSV file. An example of such a statistical analysis is the *Analysis of Variance* (ANOVA) to examine differences among group means and significance, which may be revealed by the charts.

5. Results and Evaluation

The automatic preparation of depth judgment tasks for web-based studies is one of the core functions of our framework. This includes two major components: the calculation of appropriate view points in the four conditions *NN*, *FN*, *NF* and *FF*, as well as the placement of labels to compare two judgment points. In Section 5.1, we present results of the automatically calculated stimuli. Moreover, we conducted a web-based study to evaluate the void space labeling as this is the only labeling method where misalignments may occur on the part of the user. The results of this study are presented in Section 5.2. Finally, we interviewed domain experts to assess the suitability of *EvalViz*. Their feedback is presented in Section 5.3.

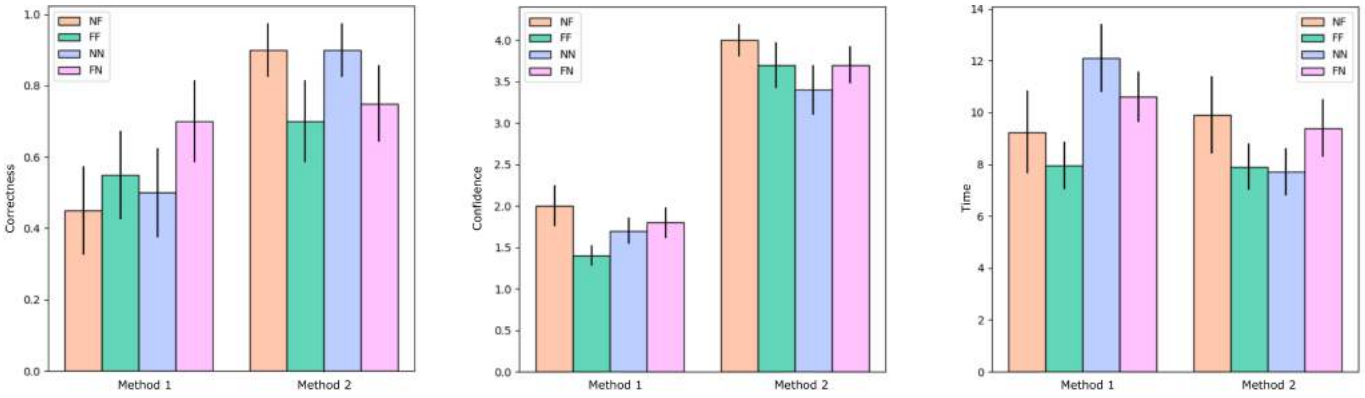

Fig. 13. Automatically generated charts for a depth perception study based on the output of our framework. Since answers, confidence, and time are recorded for every task, statistical summaries showing the mean and standard error in bar charts can be generated.

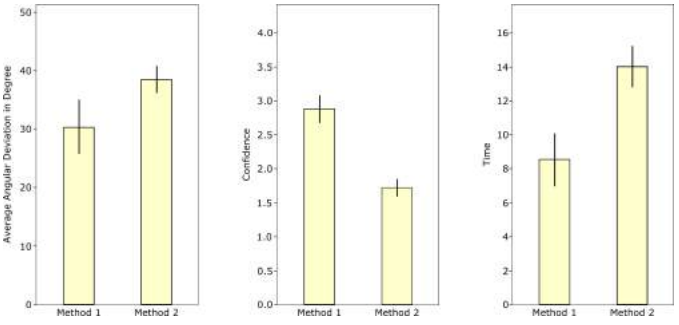

Fig. 14. Automatically generated charts for a shape perception study based on the output of our framework. Since angular differences, confidence, and time are recorded for every task, statistical summaries showing the mean and standard error in bar charts can be generated.

5.1. Results of Depth Judgment Task Generation

We applied *EvalViz* to calculate appropriate scenes to four data sets of liver vessel trees. The number of detected endpoints varies between 60 and 82 and the resulting number of pairs of endpoints for which images have been calculated varies between 1770 and 3240. For each endpoint pair, an image is calculated depending on the condition it fulfills. Our testing system uses an Intel Core i5 CPU with 2.8 GHz, 16 GB RAM, and an NVidia GeForce GTX 1080 Ti. The computation time per image depends on the number of applied rotations as well as on the number of gradient descent steps and varies between 0.21 and 9.8 s with 0.87 s on average. Regarding shape judgment tasks, the generation of images is much faster and varies between 0.09 and 0.15 s with 0.1 s on average per image since no complex calculations are necessary.

We qualitatively analyzed the resulting images to check if our method leads to reasonable results. Figure 15 shows exemplary results for the four conditions. For each case, the considered endpoints are labeled with '#' and '+'. To investigate the generated differences in depth between two endpoints, we encode depth using pseudo-chromadepth [19]. Our method leads to appropriate results: depending on the condition category, two endpoints have either a small or large distance in image space as well as either a small or large depth distance.

However, two aspects could make it more difficult to perform depth judgment tasks. First, the visibility of a vascular branch may be limited when a small branch of a vessel appears behind a larger branch. As soon as the endpoint of the smaller vascular branch is visible, our algorithm goes over to the calculation of the associated label. However, most of the smaller branches may then be occluded by the larger branch, which can make it difficult to compare depths with another endpoint. In our tests, such cases occurred in about 4 % of the images. Secondly, the random determination of the variable r, which defines the rotation of the vessel tree (see Section 4.1.2), can cause small differences in the depths for the conditions *NN* and *FN*. Depending on the used depth encoding, such small differences may not be visually perceivable.

5.2. User Study to Evaluate Void Space Label Placement

To obtain expressive results from a depth perception study, it is important that the participants know which judgment points are to be compared. This depends on the position of the associated labels. To check the quality of our void space labeling algorithm, we conducted a user study with 14 participants (6 female, 8 male; age range from 23 to 35). Among them were 11 participants with a background in computer science, one in engineering, one in mathematics, and one in medicine. Six participants stated they have no experience in scientific visualization, while eight stated they are experienced. None of the participants had any known color vision deficiencies.

To perform the evaluation, we adopted our framework to generate a web-based study, as described in Section 4.2. From the set of images for each data set (see Section 5.1), we randomly selected 10 images for each of the four vessel tree data sets. We kept only the first label and removed the second one. Besides, we have colored the associated vessel endpoint, as well as the two nearest vessel endpoints based on their screen space distance, see Figure 16. The task for the user is then to decide which vessel endpoint the label belongs to, based on the smallest distance in screen space. Only one response can be selected for every task. In addition, we recorded the time and indicated confidence of the participant per task. In total, each participant had to perform 40 tasks.

The summary statistics for this study can be seen in Table 1. The mean percentage of correct answers is 92 % ($SD = 27$ %),

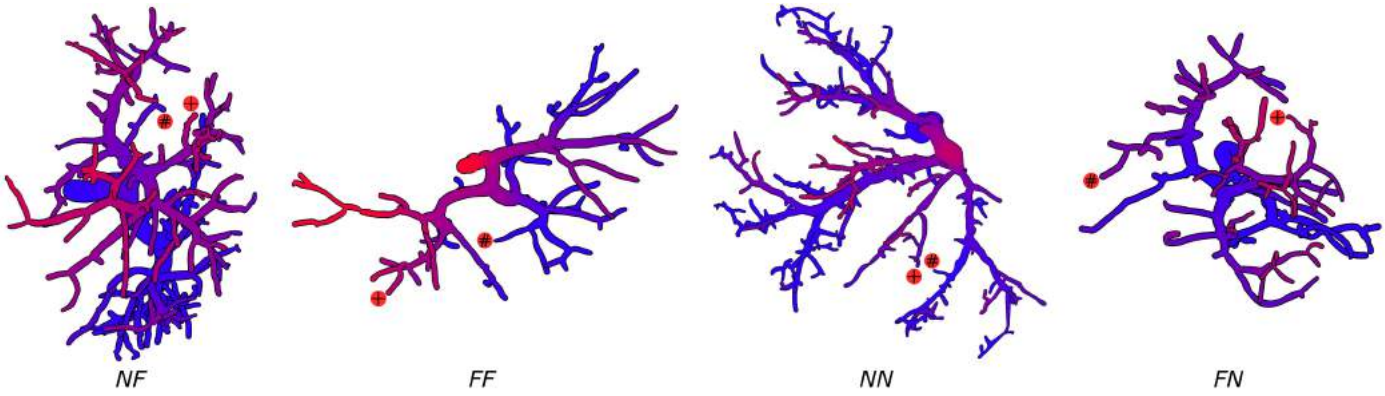

Fig. 15. Exemplary results for the four conditions. To encode depth, pseudo-chromadepth [19] is used. For each case, the considered candidates are labeled with '#' and '+'.

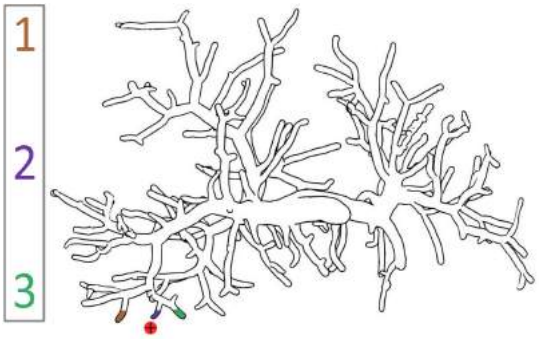

Fig. 16. Example image from the user study to evaluate the void space label placement. The vascular branch of the labeled endpoint and the closest two endpoints are colored. The participant has to decide which vessel end the label belongs to, based on the smallest distance in screen space. Here, the correct answer would be 2.

Table 1. Statistical summary of the label placement study.

	M	SD
Correctness ratio	0.92	0.27
Confidence	4.08	0.05
Time	6.44	9.72

where most of the participants where quite confident ($M = 4.08, SD = 0.05$, on a scale of 1 (very uncertain) to 5 (very confident)), and fast ($M = 6.44, SD = 9.72$ seconds, median of 2 seconds) with their decisions. The results demonstrate that our algorithm to place labels calculates positions that can be correctly assigned to the correct vessel endpoint.

5.3. Qualitative Expert Feedback

To assess the quality of our proposed framework, we conducted an informal interview with experts, who are familiar with scientific visualization, and particularly in the field of depth and shape perception. For this, we asked three visualization experts E1, E2, E3 (age 29,32,37; all male), who contributed in the field of depth and shape perception, and who also designed evaluations to assess the effect of their visualization techniques. First, we asked them about their previous studies. All participants stated that they had to manually generate images for the novel visualization techniques as well as related visualization methods. Concerning depth evaluations, they had to place the labels manually in a graphics editor afterward, e.g., *Adobe Photoshop*. E3 stated that this is a tedious work. Afterward, we showed the experts our framework and inquired about the usefulness and the effectiveness. All participants agreed that our framework helps to generate the images "very fast and easy" (E3) for both types of perception. E2 positively remarked that it is very simple to include new shaders and to edit the source code for more flexibility. "Even if the framework is not used to implement new visualization techniques, the saved file with camera positions and label positions is a great support for image generation" (E1).

The experts also had several comments and ideas for additional features. E1 and E2 asked for including a tutorial in the web study. Currently, we assume that the participants know the visualization techniques and that they can immediately participate in the evaluation. E1 and E2 stated that it would be helpful to generate an example and that the study creator could add explanations how the visualization technique works. Afterward, two easy test questions should be asked to assess if the participant understood the method. Furthermore, E2 demanded an information button for every task. In case the participant is suddenly unsure about the technique, it would be helpful to reread the tutorial. E3 suggested adding videos in the evaluation. These videos should show small rotations such that the study can also be conducted on rotating objects to avoid static images only. This would also facilitate the adjustment of the gauge. Furthermore, he asked to create follow-up studies to assess the results for long-term studies.

6. Discussion

Our framework to generate perceptual task-based user studies is based on observations from previous user studies. Preparing such studies manually is time-consuming and it is difficult to acquire a sufficient number of participants. Furthermore, a manual task setup may be prone to human bias, which may favor certain depth perception visualizations or put others at disadvantage. With our system, we overcome these limitations. After the evaluation setup, the image calculation, creation of the web study, and reporting is performed fully automatically.

The generation of a user study to evaluate depth perception as carried out by Lichtenberg et al. [27] (15 tasks, 3 visualization techniques) takes approximately 45 minutes. During this time, the researcher who creates the study can focus on other aspects, such as the generation of additional qualitative evaluation methods, e.g., questionnaires. In contrast, the manual preparation of a study of this scope can take multiple hours depending on the experience of the study creator. It is also not ensured that tasks for different visualizations or input structures are prepared with the same constraints and task difficulty. With our proposed technique, the study setup and report are done in an objective and repeatable manner. The impact of task difficulty on the evaluation results becomes evident in the comparison done by Lichtenberg and Lawonn [44]. For example, conducting depth judgment tasks with Phong shading, similar to the example setup in Figure 10, resulted in correctness ratios ranging from 26 % to 73 % across different studies. Such discrepancies could be avoided with our framework. Additionally, studies could be extended in a follow-up survey and be produced under the exact same circumstances as the original setup.

The user interface provides an easy-to-use option for considering own visualization techniques for task generation. Here, the creator has to select the corresponding shader files. Note that newly integrated shaders do have to fulfill specific criteria. With regard to the surface visualization, the positions and normals of the surface points are transferred to the shader as *OpenGL Vertex Buffer Objects* (VBOs). Similar to this, the positions and normals of the candidates are expected to be transferred to the GPU as VBOs, in case of shaders for rendering glyphs are selected. However, for more advanced visualization techniques it could be necessary to transfer more information to the GPU. Therefore, we will offer the possibility to customize our framework by making the source code freely available on an open access repository. As an alternative to adapting the framework, we also offer the possibility to export all calculated information per image based on the provided standard visualization techniques. For each image, a text file is written, consisting of the modelview and projection matrix, the indices of the considered candidates and the label positions. These could then be loaded into custom tools to render the images.

Providing a web interface for conducting the user studies facilitates the acquisition of a higher number of participants as they do not have to come to a lab. The web-based character, however, also has limitations. The study creator has no control over the character of the display, the lighting conditions, and the attention of the users. These aspects clearly may influence shape and depth perception. Despite the larger number of participants in the web-based study, the results may be of limited validity. However, our framework does not exclude the realization of studies in controlled environments. The creator could automatically prepare the study using our method and then run it in a lab setting. Another positive aspect of lab-based experiments is the increased control over the selection of participants, which is limited for web-based studies, and could induce a selection bias. However, web-based studies could also be helpful in integrating more experts into the study, who often have little time to participate in lab-based scenarios.

Domain Applications. Besides liver surgery, there are other possible application scenarios for our framework. The visualization of vascular structures also plays an essential role in oncological pelvic [45] and thoracic surgery [46], where for the latter, also the bronchial tree has to be visualized. For planning surgical brain interventions, fiber tracts have to be visualized to damage as little healthy tissue as possible during operation [47]. Another application of depth and shape perception-based visualizations is the education of students [48]. Depending on the scenario, different structures have to be visualized simultaneously, which requires an adequate visualization of spatial relations and the shape of anatomical structures. Besides medical data, depth and shape cues are applied to biological data for visualizing molecular structures [49] and proteins [50]. *EvalViz* could be used in this area to find suitable techniques for encoding depth and shape information. The extended detection of surface candidates (see Section 4.1.2) allows the determination of depth and shape judgment tasks to be applied directly to these scenarios.

7. Conclusion and Future Work

In this paper, we extend our previous framework [1] to prepare, conduct and analyze user studies for perception-based evaluations of scientific visualizations with minimal effort. In addition to the automatic generation of task-based experiments to evaluate depth perception, we also integrated an automatic generation of stimuli to evaluate shape perception in surface visualizations. Moreover, we extended our framework to handle arbitrary surfaces instead of just vascular surfaces. We presented the strength of *EvalViz* using different surface representations, and discussed other potential applications. To set up an evaluation, we designed a user interface, where appropriate images for task-based experiments are calculated fully automatically based on the defined settings. With just a few mouse clicks, extensive studies can be created. The obtained expert feedback confirms that our framework supports visualization researchers in creating user studies in multiple ways. First, the automatic generation of appropriate stimuli saves significant time. Second, conducting studies via a web interface provides the possibility to acquire a large set of participants. Third, the automatic study generation and analysis of evaluation results based on many responses and many techniques allows for a fair and objective comparison of task performance for a variety of visualization techniques.

At the moment, *EvalViz* is focused on task-based experiments. Concerning depth perception, two labeled positions are compared according to their depth. However, there are other task-based experiments such as the *depth profile test* [51] that could be integrated into EvalViz. Here, the user has to estimate the depth profile along a line or set of points that are placed on a surface. Besides, currently, four categories $\mathscr{C} = \{NN, NF, FN, FF\}$ for generating depth judgment tasks are distinguished, which was inspired by existing studies [26, 27]. A valuable improvement would be to integrate more user flexibility into this process. This means that the user is given the possibility to divide the depth and space distance into any number of intervals instead of limiting it to two. This could produce

a finer gradation of difficulty levels, and it could be evaluated to what extent this affects the depth perception. Moreover, we plan to integrate 3D models of the surface object that are rendered within the web-based user study for placing the gauge, which can be interactively explored by rotation and zooming. This would allow a more in-depth evaluation of shape perception. Furthermore, tasks could be integrated that aim even more strongly at understanding the relation between the surface model such as the vascular tree in the context of other objects such as organ morphology, or tumors. A possible design of such tasks could be to show three branches indicated by three labeled points and ask the user whether the second or third branch is the supplying branch of the first one. With such experiments, one could examine even more precisely to what extent the visualization techniques influence real intra-operative decisions.

Besides task performance, there are other evaluation methods, such as interviews, questionnaires, and the think-aloud method that can provide interesting data. In the future, we plan to combine our framework with such qualitative methods in order to be able to carry out more in-depth evaluations, even if these qualitative methods would probably be performed with a lower number of participants due to their more elaborate character. As an extension to the presented web-based studies, also crowdsourcing platforms could be used to increase the number of participants further. Another interesting point for future work would be the automatic generation of artificial surface models such as vessel trees [52]. Domain experts would not have to generate input data to evaluate new visualization techniques. Besides, EvalViz may be extended to support depth perception studies in AR and VR. Depth perception in AR and VR have unique properties [53, 54] that require different strategies and solutions compared to 3D visualizations on a desktop. Moreover, we want to integrate special tests to check whether a proband has color vision deficiencies, e.g. red-green blindness or color blindness. Depending on the weakness to be checked, different user inputs and interactions have to be integrated.

As it stands, our framework supports researchers in creating, conducting, and analyzing task-based user studies, and may be employed not only for assessment of novel visualization techniques but also for replication studies.

8. Acknowledgments

This work was partially funded by the Federal Ministry of Education and Research within the Research Campus STIMULATE (grant number '13GW0095A'), by the German Research Foundation (DFG) project LA 3855/1-1, and by the Trond Mohn Foundation (grant number 'BFS2016TMT01').

Appendix

This section provides proofs for the constraints of the categories $\mathscr{C}$.

∘ ***NN*:** In this case, $d_E < \frac{1}{2}D'$ and $d_s = r \in [0, D'/2)$ holds. This yields a depth distance of $d_d^2 = d_E^2 - d_s^2 < \frac{1}{4}D'^2 - d_s^2 \in (0, \frac{1}{4}D'^2]$, which shows $d_d < D'/2$.

∘ ***NF*:** In this case, $d_E \geq \frac{1}{2}D'$ and $d_s = r \in \left(0, \sqrt{d_E^2 - D'^2/4}\right]$ holds. This yields a depth distance of $d_d^2 = d_E^2 - d_s^2 \in [D'^2/4, d_E^2)$, which shows $d_d \geq D'/2$. Furthermore, the condition $d_E < \frac{\sqrt{2}}{2}D'$ yields $d_s < D'/2$.

∘ ***FF*** In this case, $d_E \geq \frac{\sqrt{2}}{2}D'$ and $d_s = D'/2 + r \in \left[D'/2, \sqrt{d_E^2 - D'^2/4}\right]$ holds. This yields a depth distance of $d_d^2 = \sqrt{d_E^2 - d_s^2} \in [D'^2/4, d_E^2 - D'^2/4]$, which shows $d_d \geq D'/2$, since $d_E^2 - D'^2/4 \geq D'^2/2 - D'^2/4 = D'^2/4$.

References

[1] Meuschke, M, Smit, NN, Lichtenberg, N, Preim, B, Lawonn, K. Automatic generation of web-based user studies to evaluate depth perception in vascular surface visualizations. In: Proc. of EG VCBM. 2018, p. 33–44.
[2] Fechner, G. Elements of psychophysics; vol. 1. New York; 1966.
[3] Cunningham, DW, Wallraven, C. Experimental design: From user studies to psychophysics. CRC Press; 2011.
[4] Healey, CG, Enns, JT. On the use of perceptual cues & data mining for effective visualization of scientific datasets. In: Proc. of Graphics Interface; vol. 98. 1998, p. 177–184.
[5] Plaisant, C. The challenge of information visualization evaluation. In: Proc. of Advanced Visual Interfaces. 2004, p. 109–116.
[6] Lawonn, K, Baer, A, Saalfeld, P, Preim, B. Comparative evaluation of feature line techniques for shape depiction. In: Proc. of VMV. 2014, p. 31–38.
[7] Lawonn, K, Luz, M, Preim, B, Hansen, C. Illustrative visualization of vascular models for static 2D representations. In: Proc. of MICCAI. 2015, p. 399–406.
[8] Baer, A, Gasteiger, R, Cunningham, D, Preim, B. Perceptual evaluation of ghosted view techniques for the exploration of vascular structures and embedded flow. Comput Graph Forum 2011;30(3):811–820.
[9] Isenberg, T, Isenberg, P, Chen, J, Sedlmair, M, Möller, T. A systematic review on the practice of evaluating visualization. IEEE Trans Vis Comput Graph 2013;19(12):2818–2827.
[10] Kleiner, M. Visual stimulus timing precision in psychtoolbox-3: Tests, pitfalls and solutions. In: 33rd European Conference on Visual Perception. 2010, p. 189.
[11] Mackay, WE, Appert, C, Beaudouin-Lafon, M, Chapuis, O, Du, Y, Fekete, JD, et al. Touchstone: exploratory design of experiments. In: Proc. of Human Factors in Computing Systems. 2007, p. 1425–1434.
[12] Aigner, W, Hoffmann, S, Rind, A. Evalbench: a software library for visualization evaluation. Comput Graph Forum 2013;32(3pt1):41–50.
[13] Okoe, M, Jianu, R. Graphunit: Evaluating interactive graph visualizations using crowdsourcing. Comput Graph Forum 2015;34(3):451–460.
[14] Englund, R, Kottravel, S, Ropinski, T. A crowdsourcing system for integrated and reproducible evaluation in scientific visualization. In: Proc. of IEEE Pacific Visualization Symp. 2016, p. 40–47.
[15] Preim, B, Baer, A, Cunningham, D, Isenberg, T, Ropinski, T. A survey of perceptually motivated 3D visualization of medical image data. Comput Graph Forum 2016;35(3):501–525.
[16] Lawonn, K, Preim, B. Feature Lines for Illustrating Medical Surface Models: Mathematical Background and Survey; chap. Visualization in Medicine in Life Sciences III. 2016, p. 93–132.
[17] Lawonn, K, Viola, I, Preim, B, Isenberg, T. A survey of surface-based illustrative rendering for visualization. Comput Graph Forum 2018;37:205–234.
[18] Steenblik, RA. The chromostereoscopic process: A novel single image stereoscopic process. In: Proc. of SPIE; vol. 0761. 1987, p. 27–34.
[19] Ropinski, T, Steinicke, F, Hinrichs, K. Visually supporting depth perception in angiography imaging. In: Smart Graphics; vol. 4073. 2006, p. 93–104.
[20] Gibson, JJ. The Perception Of The Visual World. Boston: Houghton Mifflin; 1950.
[21] Kersten-Oertel, M, Chen, SJS, Collins, DL. An evaluation of depth enhancing perceptual cues for vascular volume visualization in neurosurgery. IEEE Trans Vis Comput Graph 2014;20(3):391–403.

[22] Behrendt, B, Berg, P, Preim, B, Saalfeld, S. Combining pseudo chroma depth enhancement and parameter mapping for vascular surface models. In: Proc. of EG VCBM. 2017, p. 159–168.
[23] Ritter, F, Hansen, C, Preim, B, Dicken, V, Konrad-Verse, O. Real-time illustration of vascular structures for surgery. IEEE Trans Vis Comput Graph 2006;12:877–884.
[24] Joshi, A, Qian, X, Dione, D, Bulsara, K, Breuer, C, Sinusas, A, et al. Effective visualization of complex vascular structures using a non-parametric vessel detection method. IEEE Trans Vis Comput Graph 2008;14(6):1603–1610.
[25] Bichlmeier, C, Heining, SM, Feuerstein, M, Navab, N. The virtual mirror: a new interaction paradigm for augmented reality environments. IEEE Trans Med Imaging 2009;28(9):1498–1510.
[26] Lawonn, K, Luz, M, Hansen, C. Improving spatial perception of vascular models using supporting anchors and illustrative visualization. Computers and Graphics 2017;63:37–49.
[27] Lichtenberg, N, Hansen, C, Lawonn, K. Concentric circle glyphs for enhanced depth-judgment in vascular models. In: Proc. of EG VCBM. 2017, p. 178–188.
[28] Kreiser, J, Hermosilla, P, Ropinski, T. Void space surfaces to convey depth in vessel visualizations. arXiv:180607729 2018;.
[29] Ropinski, T, Oeltze, S, Preim, B. Survey of glyph-based visualization techniques for spatial multivariate medical data. Computers and Graphics 2011;35(2):392–401.
[30] Gibson, JJ. The perception of visual surfaces. Am J Psychol 1950;63(3):367–384.
[31] Stevens, K. Slant-tilt: The visual encoding of surface orientation. Biological cybernetics 1983;46:183–195.
[32] Koenderink, JJ, Van Doorn, AJ, Kappers, AM. Surface perception in pictures. Perception & Psychophysics 1992;52(5):487–496.
[33] Sweet, G, Ware, C. View direction, surface orientation and texture orientation for perception of surface shape. In: Proc. of Graphics Interface 2004. 2004, p. 97–106.
[34] P. O'Shea, J, Banks, M, Agrawala, M. The assumed light direction for perceiving shape from shading. In: Proc. of the Symposium on Applied Perception in Graphics and Visualization. 2008, p. 135–142.
[35] Bernhard, M, Waldner, M, Plank, P, Soltészová, V, Viola, I. The accuracy of gauge-figure tasks in monoscopic and stereo displays. IEEE Comput Graph Appl 2016;36(4):56–66.
[36] Cole, F, Sanik, K, DeCarlo, D, Finkelstein, A, Funkhouser, T, Rusinkiewicz, S, et al. How well do line drawings depict shape? ACM Trans on Graph 2009;28(3):28:1–28:9.
[37] Šoltészová, V, Patel, D, Viola, I. Chromatic shadows for improved perception. In: Proc. of Non-Photorealistic Animation and Rendering; vol. 2011. 2011, p. 105–116.
[38] Forsell, C. A guide to scientific evaluation in information visualization. In: Proc. of Information Visualisation. 2010, p. 162–169.
[39] Phong, BT. Illumination for computer generated pictures. Communications of the ACM 1975;18(6):311–317.
[40] Rieder, C, Ritter, F, Raspe, M, Peitgen, HO. Interactive visualization of multimodal volume data for neurosurgical tumor treatment. Comput Graph Forum 2008;27(3):1055–1062.
[41] Lichtenberg, N, Lawonn, K. Parameterization and feature extraction for the visualization of tree-like structures. In: Proc. of EG VCBM. 2018, p. 145–155.
[42] Meuschke, M, Engelke, W, Beuing, O, Preim, B, Lawonn, K. Automatic Viewpoint Selection for Exploration of Time-dependent Cerebral Aneurysm Data. In: In Proc. of Bildverarbeitung für die Medizin. 2017, p. 352–357.
[43] Lawonn, K, Gasteiger, R, Preim, B. Adaptive Surface Visualization of Vessels with Animated Blood Flow. Computer Graphics Forum 2014;33(8):16–27.
[44] Lichtenberg, N, Lawonn, K. Auxiliary tools for enhanced depth perception in vascular structures. In: Rea, PM, editor. Biomedical Visualisation; chap. xx. Springer International Publishing; 2019, p. in print. doi:10.1007/978-3-030-14227-8.
[45] Smit, N, Lawonn, K, Kraima, A, DeRuiter, M, Sokooti, H, Bruckner, S, et al. Pelvis: Atlas-based surgical planning for oncological pelvic surgery. IEEE Trans Vis Comput Graph 2017;23(1):741–750.
[46] Dicken, V, Kuhnigk, JM, Bornemann, L, Zidowitz, S, Krass, S, Peitgen, HO. Novel CT data analysis and visualization techniques for risk assessment and planning of thoracic surgery in oncology patients. In: Proc. of CARS; vol. 1281. 2005, p. 783–787.
[47] Svetachov, P, Everts, MH, Isenberg, T. DTI in context: illustrating brain fiber tracts in situ. Comput Graph Forum 2010;29(3):1023–1032.
[48] Tietjen, C, Isenberg, T, Preim, B. Combining silhouettes, surface, and volume rendering for surgery education and planning. In: Proc. of IEEE/Eurographics Symp on Visualization. 2005, p. 303–310.
[49] Tarini, M, Cignoni, P, Montani, C. Ambient occlusion and edge cueing for enhancing real time molecular visualization. IEEE Trans Vis Comput Graph 2006;12(5).
[50] Weber, JR. Proteinshader: illustrative rendering of macromolecules. BMC Struct Biol 2009;9(1):19.
[51] Todd, JT, Mingolla, E. Perception of surface curvature and direction of illumination from patterns of shading. J Exp Psychol Hum Percept Perform 1983;9(4):583.
[52] Galarreta-Valverde, MA, Macedo, MM, Mekkaoui, C, Jackowski, MP. Three-dimensional synthetic blood vessel generation using stochastic l-systems. In: Proc. of Medical Imaging; vol. 8669. 2013, p. 86691I.
[53] Jones, JA, Swan II, JE, Singh, G, Kolstad, E, Ellis, SR. The effects of virtual reality, augmented reality, and motion parallax on egocentric depth perception. In: Proc. of Applied Perception in Graphics and Visualization. 2008, p. 9–14.
[54] Drascic, D, Milgram, P. Perceptual issues in augmented reality. In: Stereoscopic displays and virtual reality systems III; vol. 2653. 1996, p. 123–135.