

Automatic Generation of Web-Based User Studies to Evaluate Depth Perception in Vascular Surface Visualizations

M. Meuschke¹, N.N. Smit^{2,3}, N. Lichtenberg⁴, B. Preim¹, K. Lawonn⁴

¹Department of Simulation and Graphics, University of Magdeburg, Germany

²Department of Informatics, University of Bergen, Norway

³Mohn Medical Imaging and Visualization Centre, Haukeland University Hospital, Norway

⁴Institute of Computational Visualistics, University of Koblenz - Landau, Germany

Abstract

User studies are often required in biomedical visualization application papers in order to provide evidence for the utility of the presented approach. An important aspect is how well depth information can be perceived, as depth encoding is important to enable an understandable representation of complex data. Unfortunately, in practice there is often little time available to perform such studies, and setting up and conducting user studies may be labor-intensive. In addition, it can be challenging to reach enough participants to support the contribution claims of the paper.

In this paper, we propose a system that allows biomedical visualization researchers to quickly generate perceptual task-based user studies for novel surface visualizations, and to perform the resulting experiment via a web interface. This approach helps to reduce effort in the setup of user studies themselves, and at the same time leverages a web-based approach that can help researchers attract more participants to their study. We demonstrate our system using the specific application of depth judgment tasks to evaluate vascular surface visualizations, since there is a lot of recent interest in this area. However, the system is also generally applicable for conducting other task-based user studies in biomedical visualization.

CCS Concepts

• **Human-centered computing** → Visualization systems and tools; Visualization design and evaluation methods; • **Computing methodologies** → Scientific visualization;

1. Introduction

In the past decade, many novel visualization techniques for biomedical applications have been developed. The goal is to support biomedical research, medical education or clinical practice by providing an efficient exploration, analysis and interpretation of the data. In order to demonstrate the benefits of new visualization techniques, user studies are performed that investigate the performance and the experience of the user. However, designing user studies is a time- and resource-intensive process where problems such as lack of objectivity and reproducibility may arise.

This paper focuses on the evaluation of biomedical visualizations of vascular trees in particular. To illustrate the importance of this topic, we briefly describe a major application, namely oncological liver surgery. Surgical removal of a liver tumor must be carefully planned to minimize the risk for the patient [HZH*09, HZP*14]. During surgery, it is important to understand the morphology of the organ and internal vasculature to determine the resection volume. To support decision making, physicians need visualizations that depict anatomical structures and their spatial relations to each other, where high-quality 3D models of organs and vessels can be

reconstructed from medical image data. An appropriate solution is to provide a set of static images that show all relevant information with an adequate visualization of depth conditions to avoid misinterpretation of spatial relations. These images can be printed out or displayed on a monitor during surgery.

Vascular trees are complex due to their highly curved and branching nature, which complicates an appropriate depiction. Therefore, novel techniques to visualize such structures have been developed and refined. Basic requirements are that spatial relationships between objects and their distance to each other should be made apparent. User studies are then carried out to check whether a new visualization technique allows for adequate or improved shape and depth perception in complex vascular structures compared to existing methods. For this purpose, participants are asked to perform specific tasks with regard to shape and depth perception based on different visualization techniques. However, there are three major issues when performing such studies. First, the tasks are created manually by selecting suitable landmarks and camera positions, which is a time-consuming, tedious and subjective process. Second, it is necessary to capture specific information throughout a study. This includes the required time and accuracy with which

a participant completes a task. Third, interactive test applications have to be made available for a wide range of participants.

To ease this process, we present a system to quickly generate perceptual task-based user studies for surface visualizations. The resulting experiments are performed via a web interface, and we provide automatic statistical reporting of the results. This approach reduces the effort to create user studies, and the web-based solution helps researchers to attract more participants by offering remote access. We demonstrate our approach based on the specific application of depth judgment tasks in vascular surface visualization, but the system is more generally applicable. In summary, we make the following contributions:

- Automatic generation of task-based experiments to evaluate depth perception in new visualization techniques compared to existing methods on the example of vascular surface models.
- Automatic preparation of web-based user studies using the generated tasks, including statistical analysis and reporting of the results.

2. Related Work

Related work to our approach comprises concepts to evaluate scientific visualizations. Moreover, we outline visualization techniques to support depth perception, in particular in the biomedical domain.

2.1. Perceptual Experiments in Visualization

The goal of perceptual experiments is to investigate the relationship between a physical stimulus, and perceptions of its effects [Fec66]. They provide empirical proof 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 specific task [CW11]. In the context of biomedical 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 a *factor*, represents the object that should be investigated, which is generated by systematically changing the stimuli. A dependent variable measures an effect in the participant's behavior that is to be influenced by the independent variable.

These effects can be measured quantitatively and qualitatively. Quantitative measurements include acquiring 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 include nominal data mostly in the form of verbal reports, which are used to investigate subjective aspects such as the participant's preferences or acceptance of a factor.

We focus on perceptual 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 in order to demonstrate the benefits of a novel method [HE98]. For this purpose, general tasks such as compare, associate, distinguish, rank, group, correlate or categorize can be performed by the user [Pla04]. With these tasks, a

multitude of visualization-based research questions can be investigated ranging from abstraction [LBSP14] to spatial relation perception [LLPH15] to decision making [BGCP11]. However, perceptual experiments are rarely carried out due to the large amount of preparation required [IIC*13].

Therefore, software solutions have been developed to support the preparation of user studies. In neuroscience, the *PsychToolbox* is a widely used set of functions to generate visual and auditory stimuli for performing cognitive experiments. *TouchStone* [MABL*07] is an open-source platform to support designing, conducting and analyzing human-computer interaction experiments. Aigner et al. [AHR13] proposed *EvalBench*, a software library to reduce the implementation effort to evaluate lab-based experiments. Okoe and Jianu introduced *GraphUnit* [OJ15], a framework to evaluate graph visualizations by leveraging crowdsourcing. Englund et al. [EKR16] 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 appropriate viewpoints and automatic label placement to evaluate depth perception is missing. To our knowledge, no framework exists that enables an easy evaluation of existing and novel visualization techniques by generating depth perception task-based experiments automatically.

2.2. Depth Perception

Studying depth perception has gained increasing importance in visualization research, especially for biomedical applications [PBC*16]. The visual encoding of depth determines how accurately 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 methods. Stereoscopic cues are a natural way of providing 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 huge amount of interaction (e.g., during an operative intervention). In these cases, additional depth cues are important. Further subcategories of depth cues are motion-, surface- and illumination-based cues. Frequently used techniques are color scales, glyphs or illustrative line drawings [LP16, LVPI18]. These cues can help to reconstruct the 3D structure of an object that is perceived through a projection onto a 2D image plane.

2.3. Depth Perception in Biomedical Visualizations

For analyzing biomedical data, an appropriate visualization of spatial relations is needed. *Chromadepth* [Ste87] uses the visible color spectrum to encode depth, which was frequently applied to vascular structures. An extension to chromadepth is pseudo-chromadepth [RSH06], which uses a color palette from red to blue instead of the complete visible color spectrum. This is inspired by the scattering of light in the atmosphere, as objects in red colors are perceived to be closer than blue colors. Similar to chromadepth is air perspective, where objects that are further away are perceived

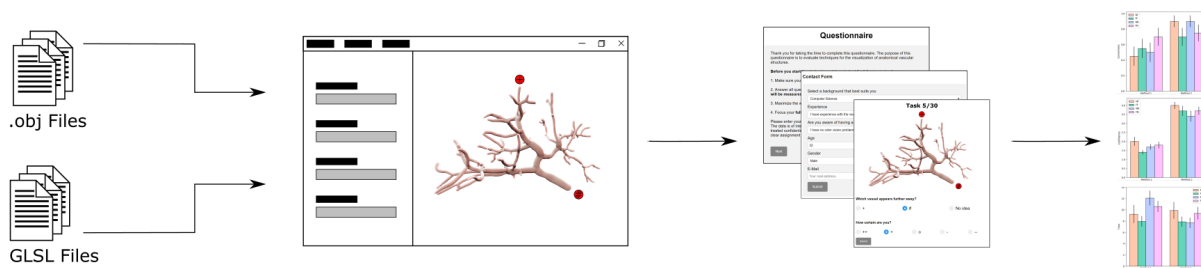


Figure 1: Our system takes surfaces, shaders, and user-defined parameters as input. Based on this input, it detects endpoints and generates labeled images as stimuli, generates a web-based study, and automatically reports study results via statistical summary charts.

with less contrast [Gib50]. Kersten-Oertel et al. [KOCC14] evaluated multiple depth cues for vascular visualization, where aerial perspective and pseudo-chromadepth outperformed stereopsis.

When chromadepth is applied to 3D surfaces, it becomes difficult to encode additional information on the surface. Behrendt et al. [BBPS17] tackled this problem by using the Fresnel term to combine chromadepth and additional parameters. Illustrative techniques have also been applied to enhance depth perception. Ritter et al. [RHP*06] used illustrative shadows to emphasize the distance between vessels. To further support depth perception of vascular structures, Joshi et al. [JQD*08] used *toon shading* and *halos*.

Other work showed that the visualization is not restricted to modifying the appearance of the target 3D model. Rendering supporting geometry also allows to interpret a 3D scene. The virtual mirror introduced by Bichlmeier et al. [BHFN09] adds a second view perspective to resolve problems with occluding geometry. Adding a shadow plane aids the perception of depth in a natural way [LLPH15]. Reference objects, whose depth can be easily interpreted, may also aid the perception of more complex structures. Lawonn et al. [LLH17] used a cylinder to provide depth information. Lichtenberg et al. [LHL17] used camera-oriented disc-shaped glyphs to provide depth information at vessel endpoints. Recently, Kreiser et al. [KHR18] introduced *Void Space Surfaces*, where the empty space in between vessel branches is used to encode depth.

The above-mentioned techniques have their strengths and weaknesses. Usually, methods that are able to convey the depth distribution of a model in its entirety fail at uncovering subtle depth differences. For example, pseudo-chromadepth easily covers a whole mesh, but perceiving small differences is difficult due to the smooth color changes. Information at discrete surface points can be faithfully visualized using glyphs [ROP11]. Deriving information towards surface locations that are not covered by glyphs may be impossible or require a lot of cognitive effort from the viewer. We are not aware of a broad study that tackles such different aspects that could be used as a guidance for more task-oriented decisions. However, the proposed system supports the creation of such studies.

3. Requirements Analysis

The current state of the art in quantitative user studies to evaluate biomedical visualizations has motivated us to develop the proposed system. It is based on two observations by Isenberg et al. [IIC*13] which explain the lack of quantitative evaluations in scientific vi-

ualization. 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 [For10], Englund et al. [EKR16] 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 a system to support biomedical visualization researchers in all three phases of perceptual task-based studies. We focus our system on supporting the evaluation of surface visualization methods via depth judgment tasks. The proposed system 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 judgment tasks. Such a task could, for example, consist of determining which of two marked positions in an image is closest to the viewer. Besides the selection of predefined visualization techniques to encode depth, e.g., pseudo-chromadepth, the creator of the study can add novel visualization techniques that should be considered for task generation. In order to support the study creator in this task, the following requirements have to be met:

- The system should be able to handle user input parameters, such as how many tasks should be created, and which visualization techniques should be taken into account.
- The system should allow for custom surface shader specification to evaluate novel visualization techniques.
- The system should automatically generate representative images for depth judgment tasks as stimuli.

In the second phase, the generated images are used as input for a web interface, which is used 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 user study creator in the second phase, the following requirements have to be met:

- The system should allow participants to take part in the study via any web browser.

- The system 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 system as well. Here, the requirements are:

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

4. System

This section presents a systematic approach to prepare, conduct and analyze depth perception-based user studies. To this end, we analyzed the previous routine how such studies are manually carried out and identified three major steps:

1. Development of novel techniques to enhance depth 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 aiming at improved depth perception using illustrative techniques [RHP*06, LLPH15, LLH17], via glyphs [LHL17], or by adding an additional layer of information [BBPS17, KHR18]. The last step reveals the potential benefits of the novel technique. Here, a scene is generated showing the vessel tree. This scene is generated with the different visualization techniques. Then, labels are placed near a certain position on the vessel. Finally, the user has to decide which label appears closer to him. In combination with the ground truth, the study shows the task performance of the evaluated methods. Nevertheless, this raises several questions, e.g., where to place the labels, how to conduct the study, and how to evaluate the results.

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

1. Preparation of depth 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 the depth 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 Depth Perception Study

This section presents the automatic generation of task-based depth experiments. The method consists of the following steps:

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

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

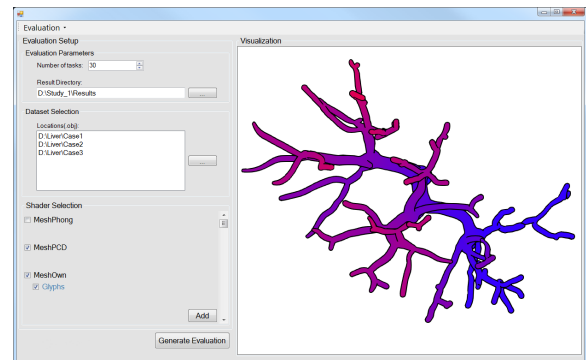


Figure 2: User interface to set up a depth perception study. The creator has to define settings such as the number of desired tasks as well as the desired data sets and visualization techniques.

4.1.1. Creator-Defined Input Parameters

First, the creator has to define several general evaluation settings using our provided user interface, see Figure 2. This includes a selection 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. This folder must contain at least the surface mesh of the vessel tree. If the data set is processed for the first time, an algorithm is applied to detect endpoints of the vessel tree (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 will be used in follow-up studies.

Finally, the creator has to select the desired visualization techniques to evaluate. The framework provides Phong shading [Pho75]

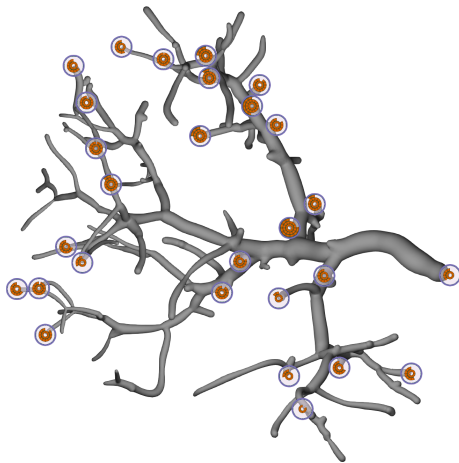


Figure 3: We provide the possibility to consider glyph-based techniques. Here, the method by Lichtenberg et al. [LHL17] was used, where glyphs are shown for a subset of the detected endpoints.

and pseudo-chromadepth [RSH06] 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. Moreover, we provide the possibility to select shaders for rendering glyphs [RRR08, LHL17]. Information about shaders that have been loaded once are stored in an XML file, and the corresponding visualizations are 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 Depth Judgment Tasks

For a meaningful study, tasks with different degrees of difficulty (DoD) are needed. This depends on the distance of two endpoints in depth and screen space. We developed an automatic approach to prepare tasks with different DoD consisting of three major steps:

1. Endpoint detection of the vessel tree.
2. Selection of appropriate endpoint candidates.
3. Calculation of viewpoints.

In the following, each of these steps is explained in more detail.

Endpoint Detection. Following previous studies about depth perception for vessel trees [RHP*06, LLPH15], we observe that the labels were placed at the endpoints of the tree. As it is difficult to estimate the correct position of a label if it is placed on the vessel itself rather than an endpoint, the estimation of the depth might be more challenging. The reason for this might be that the participants are confronted not only with the interpretation of the visualization technique, but additionally with the estimation of the label location. Therefore, we use the vessel endpoints as possible locations for the labels. A further distinctive position besides the endpoints might be branches, but we focus on endpoints only.

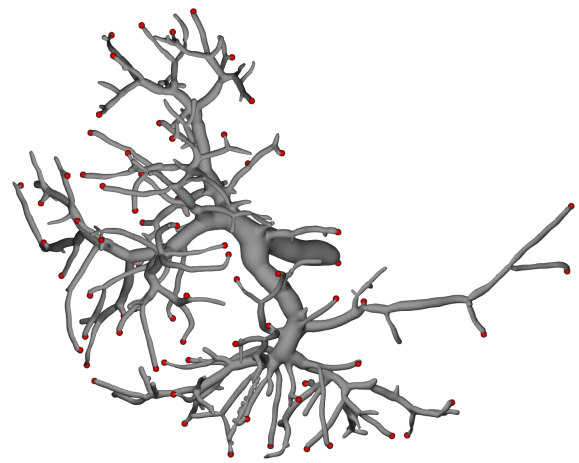


Figure 4: An exemplary result for the detection of the endpoints of a liver vessel tree using the method by Lichtenberg et al. [LHL17]. The endpoints found are represented by red spheres.

To detect the vessel endpoints, we employ the method by Lichtenberg et al. [LHL17]. First, they detected a set of candidates based on the shape index [LLH17]. Regions with a high shape index are cap-shaped and, thus, are vessel endpoint candidates. This set is further processed to eliminate outliers by considering regional topology. With this, they could distinguish between caps and cylinders, yielding an improved set of vessel endpoints.

Candidate Selection. After we determine the vessel endpoints, we need to select candidates for placing labels. The obvious choice would be to randomly select two endpoints, but following the study by Lawonn et al. [LLPH15], they grouped the endpoints according to two distance measures. A pair of two endpoints with a given camera setting is an element of the set $\mathcal{C} = \{NN, NF, FN, FF\}$. The letters N , F stand for 'near' and 'far', respectively. The first entry of the pair of letters relates to the distance of the endpoints 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 endpoints 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 endpoints, the distance in screen space is near, but the depth distance is far (see Figure 9).

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 endpoints ($\mathbf{p}_i, \mathbf{p}_j$) 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 vessel branches with a large distance to all other branches. 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 endpoints in screen space (or depth) is less than $D'/2$, we assign N .

Lawonn et al. [LLPH15] used the diagonal of the screen size to define the first letter. If the distance was less than half, N was assigned. Lichtenberg et al. [LHL17] used the diagonal of the bounding box of the geometry in screen space. Again, if the distance was

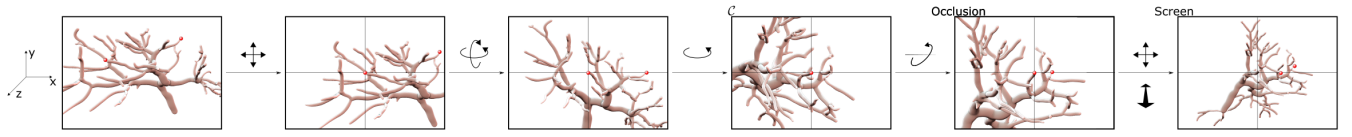


Figure 5: First, we translate the object such that the first endpoint lies in the origin, afterwards consecutive rotations result in a desired category \mathcal{C} . Then, the occlusion problem is resolved and finally the object is fit to the screen.

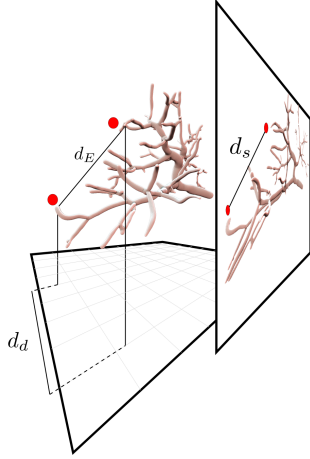


Figure 6: The distance d_E corresponds to the Euclidean distance, d_d to the depth distance, and d_s to the screen space distance.

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 \mathcal{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, then: $d_E^2 = d_s^2 + d_d^2$, see Figure 6. This yields the distinction,

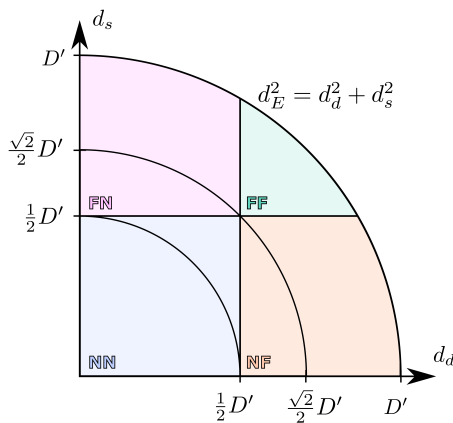


Figure 7: The conditions for which the screen and depth distances yield the Euclidean distance and, therefore, the categories.

see also Figure 7:

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

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

Viewpoint Calculation. For each category \mathcal{C} , we have to determine as many scenes as required tasks (see Figure 5) for generating a scene. Just picking a pair could lead to occlusion of an endpoint within a scene. Thus, we need to calculate a viewpoint such that:

1. Occlusions of the pair of endpoints do not occur.
2. The category of \mathcal{C} is kept.

In order to fulfill requirement 1, we constrain the 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 vessel tree such that the first endpoint \mathbf{p}_i (of the pair of endpoints $\mathbf{p}_i, \mathbf{p}_j$) 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 7. 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 .

◦ **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 $[0, \sqrt{d_E^2 - D'^2/4}]$ 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 $[0, -D'/2 + \sqrt{d_E^2 - D'^2/4}]$ 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 vessel tree around the y -axis with an angle of $\arccos \frac{d_s}{d_E}$. Now, we achieve a setting of the vessel tree such that a given configuration is fulfilled. Nevertheless, the current state does not guarantee that the vessel endpoints $\mathbf{p}_i, \mathbf{p}_j$ are visible. It is still possible that one or both points are occluded by a branch. Considering three possible rotations around the main axes x, y, z , rotating around the x -axis and y -axis may violate the configuration \mathcal{C} , rotating around the z -axis has no effect on 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 $i \in \{0, 1, \dots, 119\}$. We iterate over i until we find a camera setting such that both endpoints are visible. For every rotation, we render the vessel tree and compare the fragment's depth value with the depth value of the endpoints. If they coincide, the points are visible. If the depth of the fragment is less than the depth value of an endpoint, it is occluded by a fragment in front of the endpoint. Moreover, we randomly rotate the vessel tree around the z -axis with a random angle $[0, 2\pi)$ to avoid that the endpoints are always lying on the x -axis. Finally, we translate the vessel tree such that the whole tree is seen in the scene. For this purpose, we determine the bounding box of the tree in the screen and translate the mid point to the origin. Afterwards, we determine the maximum x, y -coordinate and scale the vessel tree such that it fits the screen.

4.1.3. Label Placement

During the study, the participants are asked to estimate which vessel branch appears closer. For this, the considered endpoints 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 endpoints.
2. Mistakenly assigned to an unintended endpoint by the viewer.

To solve this problem, 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 endpoint $\mathbf{q} = (x, y)$ and count the pixels within the circle that contribute to the vessel $p_v = \#\{\mathbf{p} \mid \|\mathbf{p} - \mathbf{q}\| \leq r \text{ and } \mathbf{p} \in \text{Vessel}\}$ and the pixels that contribute to the background $p_b = \#\{\mathbf{p} \mid \|\mathbf{p} - \mathbf{q}\| \leq r \text{ and } \mathbf{p} \in \text{Background}\}$. Afterwards, we shift \mathbf{q} first in the x -direction and then in the y -direction (the shift is three pixels). With this, we apply the gradient descent and iteratively determine the new position:

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

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 endpoint should not be occluded. Unfortunately, the second constraint is neglected with this. To consider this, we calculate the Voronoi diagram of the endpoints in screen space. For each iteration, we test if the midpoint of the circle would leave the Voronoi area of its starting endpoint. If this is the case, we restrict the movement to stay in the area, see Figure 8. 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.

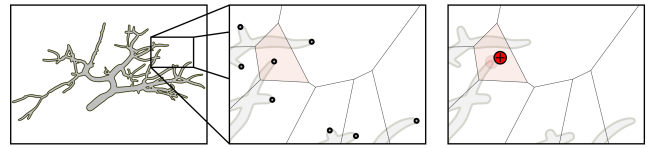


Figure 8: For every endpoint on the vessel tree, we determine the Voronoi diagram. Afterwards, a gradient descent approach is applied to find an appropriate position for the label.

4.2. Conducting a Depth Perception Study

After we are able to generate tasks in the form of labeled images, we need to ensure that the order of the scenes with the different shading techniques does not bias the results. 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. In order to reach a large audience, we decide to offer a web-based study. 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. To conduct an evaluation, all that is needed is to copy the files and generated images provided by our system 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 carefully read the instructions and to 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 vessel visualization, and color vision deficiencies. For each task, we first ask which label appears closest. Then, we ask for the confidence of participant in their answer, from very uncertain to very certain using a five-point Likert scale (—, —, o, +, ++). We measure the time it takes to answer which label appears closest. After all tasks are performed, we provide the participants 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 Depth 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 provides also 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 system. 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 vessel visualization. Moreover, 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 10, bar charts automatically created from synthetically generated evaluation results are visible, similar to the reporting charts used in the work by Lawonn et al. [LLPH15]. After interpreting the results presented in the bar charts, users are able to

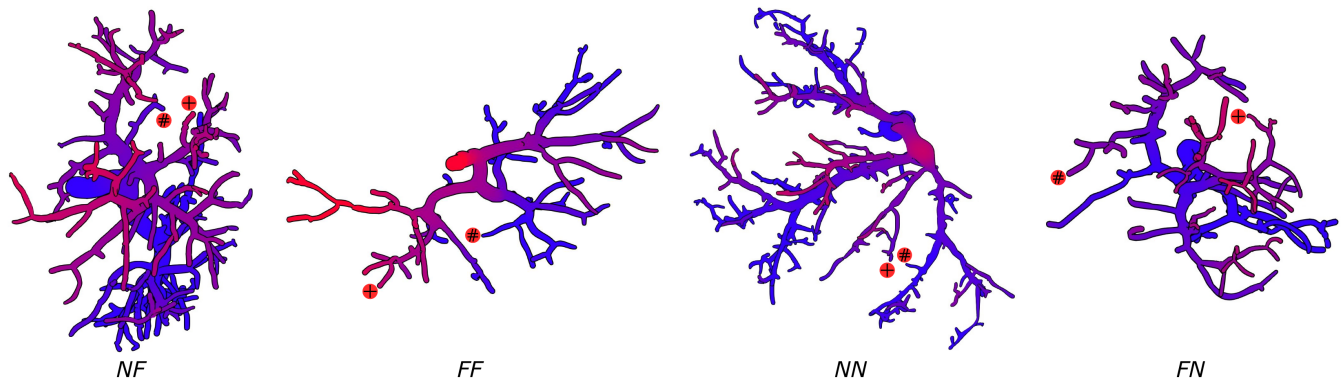


Figure 9: Exemplary results for the four conditions based on four data sets. To encode depth, pseudo-chromadepth [RSH06] is used. For each case, the considered endpoints are labeled with '#' and '+'.

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 endpoints. In Section 5.1, we present results of the automatically calculated stimuli. Moreover, we conducted a web-based study to evaluate the label placement at the vessel endpoints. The results of this study are presented in Section 5.2. Finally, we interviewed domain experts to assess the suitability of our system, and their feedback is presented in Section 5.3.

5.1. Results of Depth Judgment Task Generation

We applied our method 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 amount of applied rotations as well as on the number of used gradient descent steps, and varies between 0.21 and 9.8 s with 0.87 s on average.

We qualitatively analyzed the resulting images to check if our method leads to reasonable results. Figure 9 shows four exemplary results for the four conditions based on the four data sets. 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 [RSH06]. 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, there are two aspects that 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 branch may then be covered 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 Label Placement

To obtain meaningful results from a depth perception study, it is important that the participants know which vessel ends are to be compared with each other. This depends on the position of the associated labels. To check the quality of our 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 background in computer science, 1 in engineering, 1 in mathematics, and one in science. 6 participants stated they have no experience in vessel visualization, while 8 stated they are experienced. None of the participants had any known color vision deficiencies.

To perform the evaluation, we adopted our system 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 data sets. We kept only the first label and removed the second one. In addition, we have colored the associated vessel endpoint, as well as the two nearest vessel endpoints based on their screen space distance, see Figure 11. 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 record the time and indicated confidence of the participant per task. In total, each participant had to perform 40 tasks.

The mean percentage of correct answers is 92 % ($SD =$

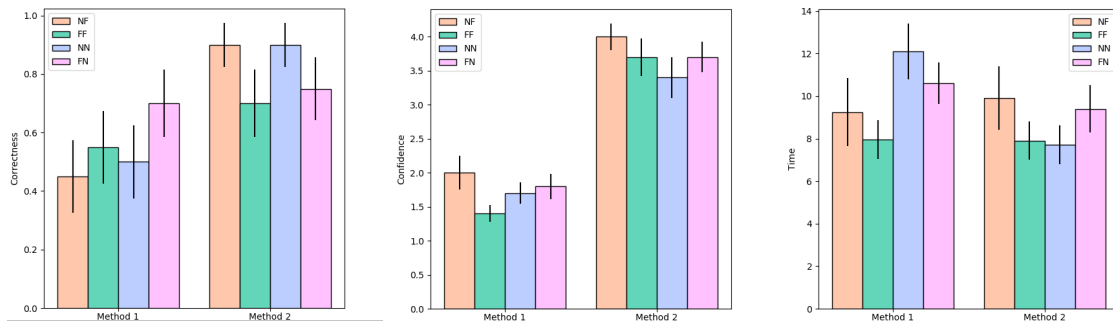


Figure 10: Automatically generated charts based on the output of our evaluation system. Since answers, confidence, and time are recorded for every task, statistical summaries showing the mean and standard error in bar charts can be generated.



Figure 11: Example image from the user study to evaluate the 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.

27 %), where most of the participants were 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 vessel visualization, and particularly in the field of depth perception. For this, we asked three visualization experts E1, E2, E3 (age 29,32,37; all male), who contributed in the field of depth 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. Afterwards, they had to place the labels manually in a graphics editor, e.g., *Adobe Photoshop*. E3 stated that this is tedious work. Afterwards, 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). 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. Afterwards, 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 read the tutorial again. E3 suggested to add 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. 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. 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 as carried out by Lichtenberg et al. [LHL17] (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 generation of additional qualitative evaluation methods, e.g., questionnaires. In contrast to this, the manual preparation of a study of this scope can take multiple hours depending on the experience of the study creator.

The user interface provides an easy-to-use option for considering own visualization techniques for task generation. Here, the creator just has to select the corresponding shader files. Note that newly integrated shaders do have to fulfill certain structure 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 vessel endpoints are expected to be transferred to the GPU as VBOs, in case 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 endpoints and the label positions. These could then be loaded into custom tools to render the images.

Currently, four categories $\mathcal{C} = \{NN, NF, FN, FF\}$ for generating tasks are distinguished, which was inspired by existing studies [LHL17, LLH17]. 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.

Providing a web interface for conducting the user study facilitates the acquisition of a higher number of participants as they do not have to come to a lab. However, conducting experiments in a lab also has advantages. The creator has full control over the monitors and lighting used during the study and can create the same conditions for all participants. This is not the case for web-based studies, which could negatively influence the results. 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 many other possible application scenarios for our tool. The visualization of vascular structures also plays an important role in oncological pelvic [SLK*17] and thoracic surgery [DKB*05], where for the latter, also the bronchial tree has to be visualized. For planning surgical brain interventions, fiber tracts have to be visualized in order to damage as little healthy tissue as possible during operation [SEI10]. Another application of depth perception-based visualizations is education of students [TIP05]. Depending on the scenario, different structures have to be visualized simultaneously, which requires an adequate visualization of spatial relations. Besides medical data, depth cues are applied to biological data for visualizing molecular structures [TCM06] and proteins [Web09]. Our tool could also be used in this area to find suitable techniques for encoding depth information. For this purpose, the determination of candidates used to generate the depth judgment tasks may need to be adapted, depending on the application. For tree structures, the method by Lichtenberg et al. [LHL17] can be used to detect endpoints, but for molecular or protein structures other methods would probably be needed. Apart from this, no additional adjustments are necessary.

7. Conclusion and Future Work

In this paper, we introduced an interactive system that makes it possible to prepare, conduct and analyze user studies for the evaluation of depth perception in scientific visualizations with minimal effort. We presented the strength of our system using liver vessel trees as an example, and have discussed other potential applications. To setup 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 system supports biomedical 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 analysis of evaluation results based on many responses and many techniques may lead to new insights for the design and suitability of visualization techniques.

At the moment, our system is focused on task-based experiments, where depth information is compared between two labeled positions. However, there are other task-based experiments such as the *depth profile test* [TM83] that could be integrated in our system. Here, the user has to estimate the depth profile along a line or set of points that is placed on a surface. Furthermore, tasks could be integrated that aim even more strongly at understanding the structure of the vascular tree in the context of organ morphology, and other structures such as 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 as discussed in Section 1.

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 system 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 further increase the number of participants. Another interesting point for future work would be the automatic generation of synthetic vessel trees [GVMMJ13]. Domain experts would not have to generate input data to evaluate new visualization techniques.

As it stands, our system 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'), and by the Bergen Research Foundation (grant number '811255').

Appendix

This section provides proofs for the constraints of the categories \mathcal{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

- [AHR13] AIGNER W., HOFFMANN S., RIND A.: Evalbench: a software library for visualization evaluation. 41–50. 2
- [BBPS17] 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), pp. 159–68. 3, 4
- [BGCP11] 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* 30(3) (2011), 811–20. 2
- [BHFN09] BICHLMEIER C., HEINING S. M., FEUERSTEIN M., NAVAB N.: The virtual mirror: a new interaction paradigm for augmented reality environments. *IEEE Trans Med Imaging* 28, 9 (2009), 1498–1510. 3
- [CW11] CUNNINGHAM D. W., WALLRAVEN C.: *Experimental design: From user studies to psychophysics*. CRC Press, 2011. 2
- [DKB*05] DICKEN V., KUHNIGK J.-M., BORNEMANN L., ZIDOWITZ S., KRASS S., PEITGEN H.-O.: Novel ct data analysis and visualization techniques for risk assessment and planning of thoracic surgery in oncology patients. In *Proc. of Congress Series* (2005), vol. 1281, pp. 783–87. 10
- [EKR16] 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), pp. 40–47. 2, 3
- [Fec66] FECHNER G.: Elements of psychophysics. vol. i. 2
- [For10] FORSELL C.: A guide to scientific evaluation in information visualization. In *Proc. of Information Visualisation* (2010), pp. 162–69. 3
- [Gib50] GIBSON J. J.: *The Perception Of The Visual World*. Boston: Houghton Mifflin, 1950. 3
- [GVMMJ13] GALARRETA-VALVERDE M. A., MACEDO M. M., MEKKAOUI C., JACKOWSKI M. P.: Three-dimensional synthetic blood vessel generation using stochastic l-systems. In *Proc. of Medical Imaging* (2013), vol. 8669, p. 86691I. 10
- [HE98] HEALEY C. G., ENNS J. T.: On the use of perceptual cues & data mining for effective visualization of scientific datasets. In *Proc. of Graphics Interface* (1998), vol. 98, pp. 177–84. 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. *J Comput Assist Radiol Surg* 4, 5 (2009), 469–74. 1
- [HZP*14] HANSEN C., ZIDOWITZ S., PREIM B., STAVROU G., OLD-HAFER K. J., HAHN H. K.: Impact of model-based risk analysis for liver surgery planning. *J Comput Assist Radiol Surg* 9, 3 (2014), 473–80. 1
- [IIC*13] ISENBERG T., ISENBERG P., CHEN J., SEDLMAIR M., MÖLLER T.: A systematic review on the practice of evaluating visualization. *IEEE Trans Vis Comput Graph* 19, 12 (2013), 2818–27. 2, 3
- [JQD*08] JOSHI A., QIAN X., DIONE D., BULSARA K., BREUER C., SINUSAS A., PAPADEMETRIS X.: Effective visualization of complex vascular structures using a non-parametric vessel detection method. *IEEE Trans Vis Comput Graph* 14, 6 (2008), 1603–10. 3
- [KHR18] KREISER J., HERMOSILLA P., ROPINSKI T.: Void space surfaces to convey depth in vessel visualizations. *arXiv:1806.07729* (2018). 3, 4
- [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 Trans Vis Comput Graph* 20, 3 (2014), 391–403. 3
- [LBSP14] LAWONN K., BAER A., SAALFELD P., PREIM B.: Comparative Evaluation of Feature Line Techniques for Shape Depiction. In *Proc. of VMV* (2014), pp. 31–38. 2
- [LHL17] LICHTENBERG N., HANSEN C., LAWONN K.: Concentric Circle Glyphs for Enhanced Depth-Judgment in Vascular Models. In *Proc. of EG VCBM* (2017). 3, 4, 5, 9, 10
- [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. 3, 4, 5, 10
- [LLPH15] LAWONN K., LUZ M., PREIM B., HANSEN C.: Illustrative Visualization of Vascular Models for Static 2D Representations. In *Proc. of MICCAI* (2015), pp. 399–406. 2, 3, 4, 5, 7
- [LP16] LAWONN K., PREIM B.: *Feature Lines for Illustrating Medical Surface Models: Mathematical Background and Survey*. 2016, ch. Visualization in Medicine in Life Sciences III, pp. 93–132. 2
- [LVPI18] LAWONN K., VIOLA I., PREIM B., ISENBERG T.: A survey of surface-based illustrative rendering for visualization. *Comput Graph Forum* 37 (2018), in print. 2
- [MABL*07] MACKAY W. E., APPERT C., BEAUDOUIN-LAFON M., CHAPUIS O., DU Y., FEKETE J.-D., GUIARD Y.: Touchstone: exploratory design of experiments. In *Proc. of Human Factors in Computing Systems* (2007), pp. 1425–34. 2
- [OJ15] OKOE M., JIANU R.: Graphunit: Evaluating interactive graph visualizations using crowdsourcing. In *Comput Graph Forum* (2015), vol. 34, pp. 451–60. 2
- [PBC*16] PREIM B., BAER A., CUNNINGHAM D., ISENBERG T., ROPINSKI T.: A survey of perceptually motivated 3d visualization of medical image data. In *Comput Graph Forum* (2016), vol. 35, pp. 501–25. 2
- [Pho75] PHONG B. T.: Illumination for computer generated pictures. *Communications of the ACM* 18, 6 (1975), 311–17. 4
- [Pla04] PLAISANT C.: The challenge of information visualization evaluation. In *Proc. of Advanced Visual Interfaces* (2004), pp. 109–16. 2
- [RHP*06] RITTER F., HANSEN C., PREIM B., DICKEN V., KONRAD-VERSE O.: Real-Time Illustration of Vascular Structures for Surgery. *IEEE Trans Vis Comput Graph* 12 (2006), 877–84. 3, 4, 5
- [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. 3
- [RRRP08] RIEDER C., RITTER F., RASPE M., PEITGEN H. O.: Interactive visualization of multimodal volume data for neurosurgical tumor treatment. *Comput Graph Forum* 27, 3 (2008), 1055–62. 5
- [RS06] ROPINSKI T., STEINICKE F., HINRICHS K.: Visually supporting depth perception in angiography imaging. In *Smart Graphics*, vol. 4073. 2006, pp. 93–104. 2, 5, 8

- [SEI10] SVETACHOV P., EVERTS M. H., ISENBERG T.: Dti in context: illustrating brain fiber tracts in situ. In *Comput Graph Forum* (2010), vol. 29, pp. 1023–32. [10](#)
- [SLK*17] SMIT N., LAWONN K., KRAIMA A., DERUITER M., SOKOOTI H., BRUCKNER S., EISEMANN E., VILANOVA A.: Pelvis: Atlas-based surgical planning for oncological pelvic surgery. *IEEE Trans Vis Comput Graph* 23, 1 (2017), 741–50. [10](#)
- [Ste87] STEENBLIK R. A.: The chromostereoscopic process: A novel single image stereoscopic process. In *Proc. of SPIE* (1987), vol. 0761, pp. 27–34. [2](#)
- [TCM06] TARINI M., CIGNONI P., MONTANI C.: Ambient occlusion and edge cueing for enhancing real time molecular visualization. *IEEE Trans Vis Comput Graph* 12, 5 (2006). [10](#)
- [TIP05] 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), pp. 303–10. [10](#)
- [TM83] TODD J. T., MINGOLLA E.: Perception of surface curvature and direction of illumination from patterns of shading. *J Exp Psychol Hum Percept Perform* 9, 4 (1983), 583. [10](#)
- [Web09] WEBER J. R.: Proteinshader: illustrative rendering of macromolecules. *BMC Struct Biol* 9, 1 (2009), 19. [10](#)