Visualisierung medizinischer Daten 2 2020
C. Wurditsch and S. Reisinger
Final Project
Volume 01 (1982), Number 3

Final Project Cardiovascular Visualization

C. Wurditsch[1] and S. Reisinger[1]

[1]Institute of Visual Computing & Human-Centered Technology, TU Wien, Austria

Abstract

This paper introduces and describes a pure java-script web-based implementation of an interactive visualization of the human heart in 3D with educational purposes, that runs completely in any modern web-browser. The visualization is done by surface rendering of a mesh-model consisting of anatomical structures of the human heart with a recent library. All anatomical structures can be selected via menu and are then highlighted in the rendering result. Structures can be hidden by mouse click to overcome the problem of occlusion. Additional fact information from Wikipedia, that can be read in English and German, is presented when selecting an anatomical structure. The visualization supports zoom and rotation and can be helpful for an interested layman for understanding the human heart's anatomy.

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

1. Introduction and Motivation

Visualizations of cardiovascular structures play an important role in diagnosis as well as in treatment of cardiovascular diseases [KBT*12] [BGP*11]. These visualizations are frequently used for disease diagnosis in radiology [AMB*13]. Different technological approaches have been developed. We have described the most frequently used ones in our state-of-the-art report for the course Medical Visualization 2 and have tried to present them in a categorized, taxonomic manner.

Of course, cardiovascular visualizations can also be helpful for educational purposes. Especially, due to the fact that vascular structures of the human heart are quite complex, three-dimensional (3D) visualizations can be helpful to understand anatomy. 3D visualizations could also be very helpful and provide a useful learning resource for medical students for anatomy education. According to Keenan and Ben Awadh "the appreciation of 3D concepts is one of the most demanding areas for medical student learning of anatomy" [KbA19]. They also claim, that the ability of interpretation of three-dimensional anatomical features from two-dimensional cross-sectional images can be problematic [KbA19]. Visualizations may be superior to two-dimensional images in anatomical atlases, if the user can interact with them.

In our report we have not identified and reported gaps in literature explicitly, but we have encountered the fact, that there is no web-based implementation of a visualization of the human heart, which is easily accessible and provides the ability to upload any data set and visualize it. In spite of this fact, it would be quite demanding to set-up such a website, for example because of compatibility issues with different data formats, and unfortunately this task is out of scope for the final project of this course. For this reason we decided to build a web-based-visualization of an example data set of the human heart and vascular structures, which is easily accessible and understandable by a layman, who wants to get more information about the heart and its vessels. We provide the functionality to select different anatomical structures, which then get emphasized in the visualization. As a data set we make use of a set of mesh models of human cardiovascular structures.

Nevertheless working on an implementation, which would provide the functionality to upload and use an arbitrary medical data set as mentioned above, would be an impressive example for future work.

Our visualization is implemented in javascript and makes use of the three.js (https://three.js.org) 3D library. It does not need any client tools to be installed on the viewers computer locally. The only requirement is a recent version of a web browser.

2. Related Work

Beside implementing the visualization that this paper describes, we also conducted a literature search and a search for other visualizations, that have the purpose to facilitate anatomy learning, to identify prior work in this field of research and found and identified a few papers relevant to this specific application of interactive visualization.

Concerning the motivation for this work, a systematic literature review was conducted by Triepels et. al., which aims to assess, whether 3D visualizations improve anatomy knowledge of medical students compared to traditional methods [TSN*20]. They found that 12 of 21 articles "showed that, according to the users' test results, using a 3D visualization method was significantly more effective than using traditional methods" [TSN*20].

Matthew Hackett and Michael Proctor also reviewed articles that had assessed 3D technology in anatomical education with focus on knowledge gains, perceptions of students and cognitive load [Hac16]. Their review, which was conducted in 2016, shows that the majority of studies, 74 % to be exact, had concluded that 3D technology brings a benefit to anatomical education [Hac16].

Nevertheless there are also different results of research, that report that anatomy learning utilizing 3D visualization models is not superior than anatomy education using conventional methods. Azer et. al. did a screening of 30 reports on the impact of 3D anatomy models on learning. They concluded that there was "no solid evidence that the use of 3D models is superior to traditional teaching" [AA16]. But they add that the studies examined by them varied in research quality and that further research was necessary to examine the short- and long-term impacts of 3D models on learning [AA16].

There are already multiple Human Body Visualization programs. For example BioDigital (https://www.biodigital.com) provides "an interactive 3D software platform for visualizing anatomy, disease and treatment." [Bio20] The software solutions provided by this platform also allow creation and authoring of 3D visualizations of parts of the human body [Bio20]. The solution appears to be commercial.

Zhang et. al. also developed a visualization platform for the human heart utilizing multiple visualization methods [ZWY*16]. In their paper they propose advanced volume rendering methods [ZWY*16].

Kenhub [ken20] offers an anatomy learning platform, but only offers videos and atlases, and no interactive 3D visualizations.

VisibleBody [vis20] offers Apps and 3D models of the human body for education and learning for medical students and professionals. These apps and 3D models are commercial.

Doccheck Flexikon (https://flexikon.doccheck.com/de/Herz) provides an advanced, interactive 3D visualization of the heart which also features labelling of different anatomic structures, zoom and rotation [doc20]. A screenshot, which we have included because of the very high quality and usability of the visualization concerning the purpose of anatomy learning, can be examined in figure 1.

3D 4medical [3d420] by the Elsevier scientific publisher provides excellent 3D heart visualizations and even features animation and allows for examining the heart beat of the animated heart at an adjustable heart rate. This goes beyond pure anatomy and even includes physiological aspects, and might also be very useful for learning anatomy, but only provides complete access when purchasing the product [3d420].

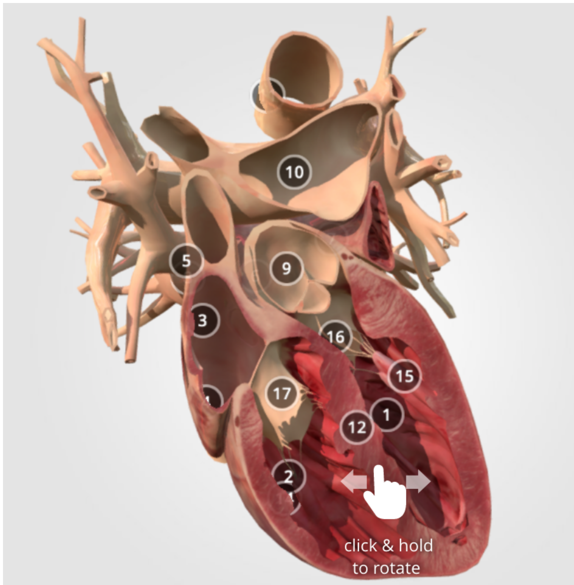

Figure 1: *This shows a screenshot of the 3D visualization of the model at https://flexikon.doccheck.com/de/Herz*

3. Data

In this section we want to shortly describe the data which was used for the visualization. Medical images of the heart of humans originate from Computed Tomography (CT) or Magnetic Resonance Imaging (MR) modalities or from the two contrast-enhanced procedures using the same imaging modalities, called Computed Tomography Angiography (CTA) and Magnetic Resonance Angiography (MRA) or from Ultrasound. CTA and MRA enhance the contrast of the vessels by intravasal application of a contrast agent. These procedures are described in more detail by Gabriel Mistelbauer [Mis10].

These real data sets of course provide the advantage of real anatomical data of a specific patient, but of course are highly-sensitive medical data in terms of privacy and availability of CT or MR data set of real patients might be a problem. The second drawback of the usage of these data sets for applications with educational purposes is, that anatomical structures for interest would have to be segmented properly and labeled first to be able to emphasize structures of interest in the visualization upon their selection.

3.1. Mesh model used as a data set

Primarily for this reason we decided to use a mesh-model of the human heart as a data set, which is part of the project BodyParts3D developed by The Database Center for Life Science https://lifesciencedb.jp/bp3d/.

This data set already provides representations of the different parts of the heart as separate labeled .obj-files in rich detail. For example in Figure 2 the model representation of the anterior cardiac vein from this data set is shown. The data set is licensed under CC Attribution-Share Alike 2.1 Japan (The license regulations of this Creative commons license can be accessed at https://creativecommons.org/licenses/by-sa/2.1/jp/deed.de).

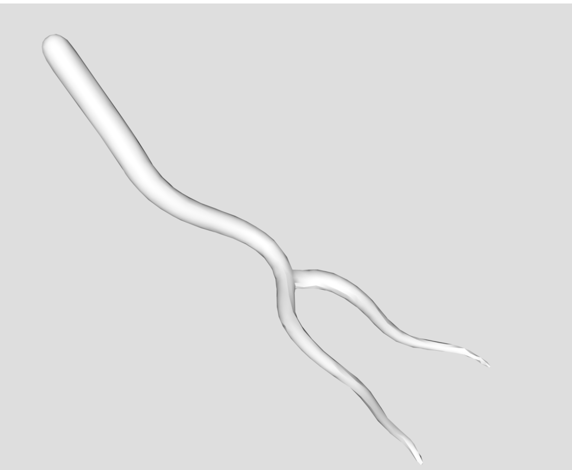

Figure 2: *This shows the mesh model from https://lifesciencedb.jp/bp3d/ of the anterior cardiac vein.*

The data set has some minor issues, which are briefly described in the following section. Nevertheless The Database Center for Life Science also informs developers or users of the data set about this on their web page and also claims that "Some parts were made from scratch by artists or distorted to fit into the environment" [bod20].

3.2. Discovered issues of the data set

Not all objects are complete 3D Objects that have a designated inside and outside. Some are just open 3D surfaces. Therefore both sides are rendered for all objects to circumvent visibility problems.

The following figures 3 to 6 (including) illustrate issues of the data set. One problem of the data set is that some of the structures are divided into multiple files. So in the final result the position where the two objects are put next to each other can be seen very clearly (see figures 3 and 5). They were also not cleaned very well, hence many of them have artifacts (see figure 6). To mention another minor issue for completeness sometimes the surface is not really smooth (see figure 4).

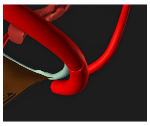

Figure 3: *Two parts of one blood vessel do not fit perfectly together*

Figure 4: *Surface is not smooth*

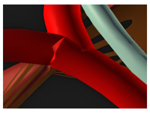

Figure 5: *Two parts of one blood vessel do not fit perfectly together*

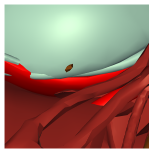

Figure 6: *There is a small object that does not belong there.*

Apart from using this mesh-model data set, it would also be possible to extend the rendering pipeline and use well segmented anatomical structures from a real CT or MR data set, which would of course increase the possibilities of the usage of our visualization.

4. Implementation

In this section we describe our implementation starting out with the discussion of basic design aspects focusing on an explanation why we decided to choose certain methods, followed by a technical description of the implementation of the visualization.

4.1. Basic Design Aspects

In this subsection we give a short explanation why we chose to implement the specific type of visualization.

In general there are multiple technological approaches to accomplish a visualization of medical image data sets, including approaches based on surface-rendering (indirect volume rendering) and (direct) volume-rendering. We have described them in detail in our report. For this use-case we make use of surface-rendering since we use mesh-models of structures as a basis. We decided to implement a 3D visualization, because 2D visualizations project all data onto one plane, which leads to loss of anatomical information [BGP*11]. 3D visualizations preserve anatomical information better, but lead to occlusion, if they are not interactive and the user cannot rotate them to look at the results from all perspectives [BGP*11].

With this aspects in mind we consider it as a huge benefit if the user can zoom in and out and rotate a visualization. The visualization developed is fully interactive and allows the viewer to zoom in and out from, as well as rotate the rendered result and highlight structures of interest. All parts of the human heart from the data set can be selected from a menu and then are highlighted utilizing a bloom shader.

One further aspect that was important for us, when choosing the underlying framework, was, that the visualization should be web-based and run in any recent version of a web browser without the requirement to install any software or tools on the viewer's computer. For this reason we opted for the three.js JavaScript 3D library (https://three.js.org) as a rendering framework, which uses WebGL to render a visualization. WebGL is "a cross-platform, royalty-free web standard for a low-level 3D graphics API" [web20] and does not require any plugins to be installed [web20]. It is derived from OpenGL ES, another free cross-platform API for rendering [ope20] [web20].

A further advantage is that the three.js 3D library is also open source software freely available to the public.

We also had in mind, that further additional information on the structure currently selected, could be useful and should be displayed when a structure is selected by the user.

4.1.1. Browser compatibility

According to the three.js library's documentation the browsers Google Chrome 9+, Firefox 4+, Opera 15+, Safari 5.1+, Internet Explorer 11 and Microsoft Edge are supported [thr20]. The only requirement is enabled java-script execution in the browser's properties.

We have tested the visualization with different browsers on different platforms, and can report good compatibility. The browsers used for testing can be looked up from table 1. Although the visualization is not optimized for mobile devices and browsers and there are known problems concerning display of the menu, which is too long for some mobile device screens, and the info panel together with the rendering of the visualization, the visualization itself is also runnable on mobile devices. On some of the test devices we encountered performance problems for the visualization rendering though.

Operating system and Browser	Test result
Google Chrome 83.0.4103	No issues
Firefox 77.0.1	No issues
Opera 68.0.3618.165	No issues
Microsoft Edge 83.0.478.45	No issues
Safari 13.1.1	No issues
Internet Explorer 11.0.195	Not Supported
Safari (iOS 13/iPadOS)	Partially works
Firefox 26.0 (iOS 13)	Partially works
Opera Touch (iOS 13)	Major problems
Dolphin (Android)	Performance problems
Samsung Internet 8.2.01.2	Performance problems
Sony TV (Android 8)	Not supported

Table 1: *Browsers selected for testing the final result of the visualization*

4.2. Technical Description

This section briefly describes the major aspects of the implementation from loading of the objects (.obj-Files), the color mapping scheme, rendering to highlighting and hiding of anatomical structures. Afterwards the additional inclusion and display of anatomical and physiological facts for anatomic structures from Wikipedia is described.

4.2.1. Object Loading

For loading all the .obj-Files of the data set local copies of the .obj-files of the data set stored in a separate directory are loaded by making use of the three.js class `OBJLoader2Parallel` [OBJ]. To find all the .obj-Files needed and the colors for the rendering, a constant array holding the filenames of all files to load, the desired colors of the rendering of the structures and mappings of their corresponding Wikipedia articles, which are mapped manually to the structure represented by an object (see section 4.2.7) has been defined. The material as well as the texture of the .obj files are therefore not used. `OBJLoader2Parallel` extends the basic `OBJLoader2` with the ability to execute the parser in a web worker. This allows the loading of all the objects being run in the background independently of the rest of the script and so the website never freezes. After the loading of the mesh is completed a `callbackOnLoad` event is triggered. This assigns it a material and adds it to the scene.

The objects are originally positioned off-center. Therefore they are shifted by hard-coded values.

To be able to scale, move or rotate all the anatomical structures at once all parts are not directly loaded to the scene but we use a `Pivot` object where all the parts of the heart are added as descendants.

4.2.2. Color Scheme

The color scheme of our approach is, that each kind of anatomical structure of the heart is colored in just one color. We used this more abstract coloring scheme proposed by the authors of the data set because this project focuses on learning the different parts of the organ and their relative position to one another and less on the realism of the visualization result.

Regarding the vessels we have to point out, that the coloring follows the oxygenation of the blood they carry and not their anatomical classification, whether they are arteries or veins, resulting in red for all arteries except the pulmonary artery, as they carry oxygenated blood, and blue all veins except the pulmonary veins, because they carry deoxygenated blood. The pulmonary artery is blue, while the pulmonary veins are red.

The detailed color scheme can be looked up from table 2.

Organ strucutres	Color
Coronary arteries	red
Pulmonary vein	red
Aorta	red
Coronary veins	blue
Pulmonary arteries	blue
Vena cava superior and inferior	blue
Trabecular muscle	saddlebrown
Myocard	brown
Right and left Auricle	#ffc0a1
Heart valves	lightblue

Table 2: *Color scheme for the different kinds of anatomical structures*

4.2.3. Interactions

The current implementation of this project uses `TrackballControls` for controls. As the heart is in the middle (0,0,0) of the plane it is interesting for users to be able to see the object from all directions easily. This can be achieved by dragging the mouse courser over the canvas. For translation of the object the CTRL-key has to be pressed on Mac OS. On mobile devices translation of the object can be achieved by using two fingers. For hiding objects the user has to click on them with the mouse. To gain knowledge about which structure in the 3d space the mouse is over we use `THREE.Raycaster` [Ray]. It checks all intersections between the ray cast, from the mouse in the direction of sight and the objects in the scene. The actual hiding of the selected 3d object is implemented by setting the attribute `visible` to `false`. This functionality is not available for mobile devices (see section 4.2.6).

There are two drop-down menus for highlighting and showing further information (see section 4.2.5).

4.2.4. Rendering

The scene is illuminated by 3 light sources. It has an `THREE.AmbientLight` and two `THREE.DirectionalLights` from different directions to get a better 3 dimensional feeling of the anatomical structures.

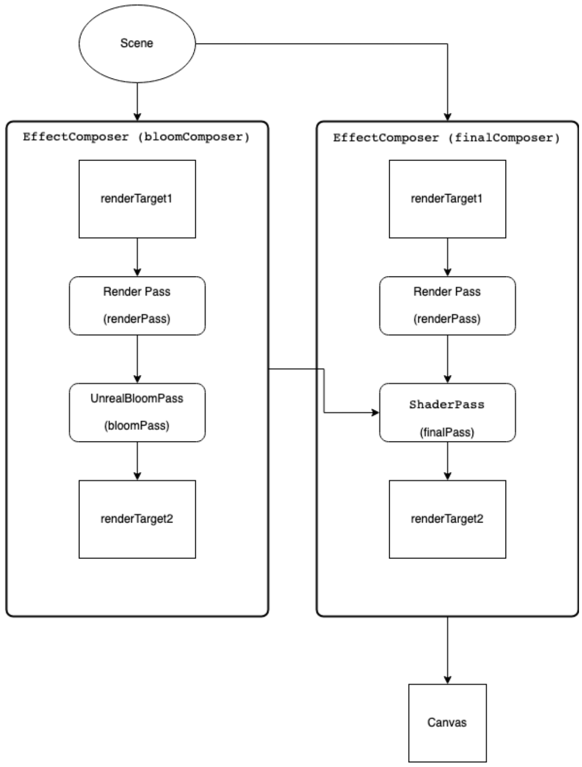

Figure 7: *Rendering Pipeline: From 3D scene to canvas*

For rendering we use the `THREE.WebGLRenderer`. It displays the scene using WebGL. To add the bloom effect the renderer is passed to the `EffectComposers`. They are designed to manage the post-processing step. Multiple `EffectComposers` can be used and the final pass to the last composer is then rendered to screen. For the glow effect we use two `EffectComposers`. The first one just renders the bloom effect (`bloomComposer`). As the first pass the `RenderPass` is added. It is used to render the scene with the camera. We also added the textttUnrealBloomPass. It only renders the glowing objects by changing the material color of the objects that should not glow to black. The whole rendering pipeline can be examined by having a look at 7

The second composer (`finalComposer`) adds up the result of the first composer and the rendering of the objects from the `RenderPass` up. For this task we added a `ShaderPass`. It gets as an input parameter a `THREE.ShaderMaterial` which uses a custom vertex and fragment shader. In the fragment shader the color from the normal rendering of the scene and the bloom texture are added up and then passed to the canvas.

4.2.5. Highlighting

To enable the viewer to select the anatomical structures and be able to see where exactly they are located, we designed a feature for highlighting of selected structures. For selection we implemented a menu, from which the viewer can select each anatomical structure. The highlighting of the selected organ is done with a bloom shader. It "gives all brightly lit regions of a scene a glow-like effect" [blo20]. The effect is illustrated in Figure 8 with the example of the selection of the wall of left auricle of the heart.

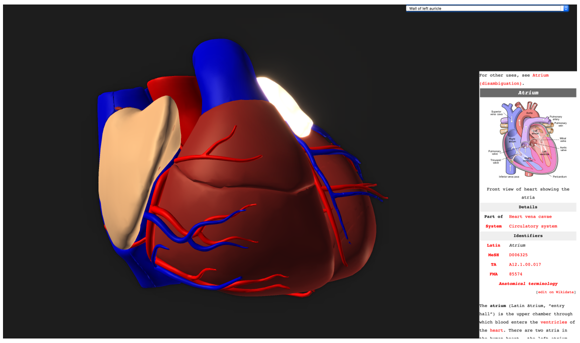

Figure 8: *Highlighted wall of the left auricle*

4.2.6. Selective hiding of structures

The problem with visualizing and highlighting of structures inside the heart is, that they can be occluded by outer structures that are in between the camera position and inner structures of interest. For this reason we implemented the functionality to hide anatomical structures by simply performing a mouse click on them. Figure 9 shows the heart opened up with multiple hidden structures providing non-occluded view of the anterior papillary muscle of the left ventricle highlighted in this figure.

Of course we also provide an option to revert the rendering to it's original visibility state with all anatomical structures visible by selecting "show all" from the highlighting selection menu for all anatomical structures.

4.2.7. Presentation of additional facts for selected anatomical structures

In order to present further anatomical and physiological information to the user, we decided to display an information panel, that shows basic information from Wikipedia for the structure currently selected. To display content from Wikipedia the Wikipedia API that is described in [wik20] is used. An example of such an info panel for the posterior interventricular artery is shown in figure 10. Figure 8 also

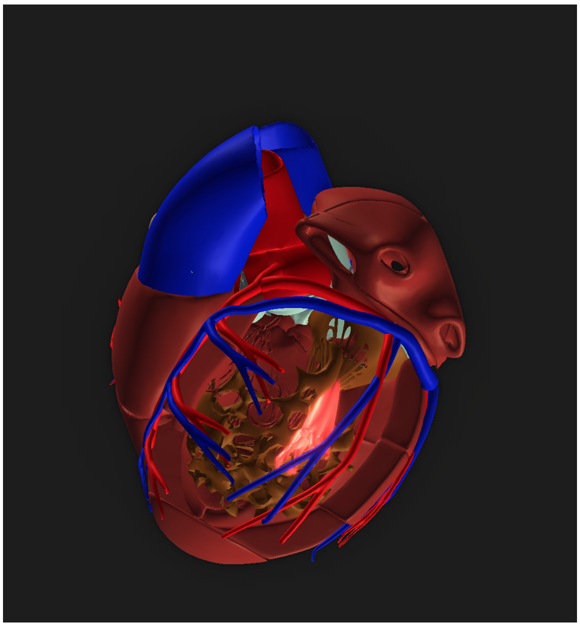

Figure 9: *Non-occluded view of the anterior papillary muscle of the left ventricle by hiding multiple different structures*

shows a Wikipedia info panel together with the 3D rendering after the selection of an anatomical structure, in this case the wall of the left auricle. The information from Wikipedia can be displayed in two languages, English and German. Because there is no automatic linking between articles concerning the same subject in different languages on Wikipedia, the right articles were linked manually. There also exists an option to hide the Wikipedia panel and solely view the visualization rendering without displaying further informations.

5. Results

Interactive 3D visualizations can help in learning anatomy and in medical education. Several papers exist that have examined and compared anatomy learning by using 3D visualizations to traditional anatomy learning methods. The majority of them suggests that there are significant benefits.

Quite a few platforms, web pages and apps exist that provide 3D visualizations for human anatomy learning, partially for medical students and professionals, partially for layman. Quite a few of them provide the visualizations at least partially on a commercial basis or restrict access in the free versions, some of them are freely accessible by everyone or embedded in a freely accessible web page.

In this project a 3D visualization of the human heart was developed with a pure-JavaScript library called three.js. It is interactive and allows the user to zoom and rotate the rendering, as well as hide each anatomical structure individually. This enables the user to get a non-occluded view of all

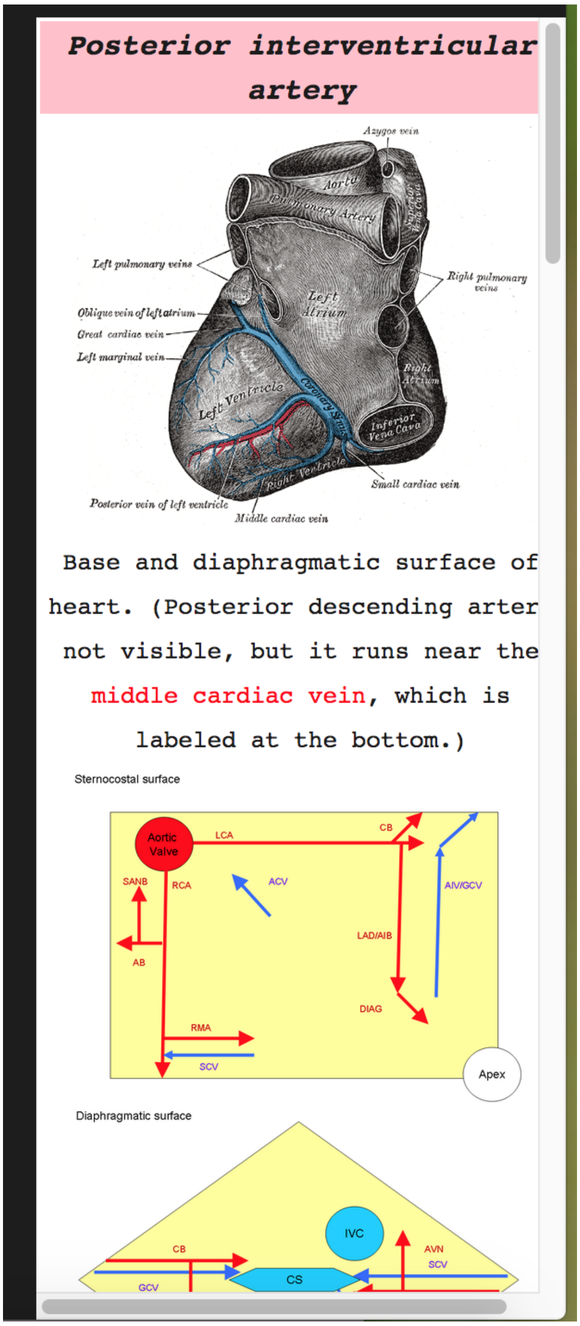

Figure 10: *Information panel with infos from Wikipedia for the posterior interventricular artery*

structures inside the heart. Each kind of anatomical structure is colored in one color to make it easier to identify and distinguish them.

The user can select an anatomical structure of interest from a menu, which then gets highlighted in the visualization, in addition a panel with anatomical and physiological information from Wikipedia gets displayed beside the 3D rendering of the visualization. The Wikipedia information can be displayed in two languages, English and German.

The visualization runs in any recent version of the most modern web browsers with JavaScript execution enabled for the site, on which the web page, that controls the rendering, is hosted. No client software is needed on the viewer's computer.

As the project does not use php it is possible to host it on github.io and therefore an online demo of the final result of our visualization can be found and explored at: https://ippon1.github.io/Medical_Vis. A showcase video was made and can be downloaded as well from https://github.com/ippon1/Medical_Vis/tree/master/screenshots.

We have tested the final visualization with some of the most common browsers on different platforms and can report good compatibility. The visualization runs without major problems in most of the recent versions of the most popular browsers. The detailed results are given in table 1.

Figure 11 shows an example screenshot of the developed visualization and gives an overview of the 3D objects rendered from an approximate frontal perspective. Figure 12 shows a different perspective of the heart, rotated approximately 90 degrees to the front, which provides a good view of the aortic valve from the perspective of the arcus aortae. The anterior cusp of the aortic valve is highlighted. Another example of highlighting of anatomical structures together with the Wikipedia information panel displayed can be examined in figure 8 for the wall of the left auricle.

6. Discussion and outlook to future work

The visualization presented is interactive and provides 3D views of an anatomical model of the human heart. Structures of interest can be selected from a menu and are then highlighted in the rendered result.

The project is optimized for desktop computer usage. If the width of the screen is too low the panel that shows the Wikipedia information will overlap most of the 3D objects. A possible starting-point for future work would be to develop a mobile version, that would fit onto the screens of mobile devices, which would preserve the free, non-occluded view of the visualization as well access to the Wikipedia information panel. This could be interesting for students who want to learn on the go.

Another possibility to gain better insight of the inner structures of the human heart could be a cross-section.

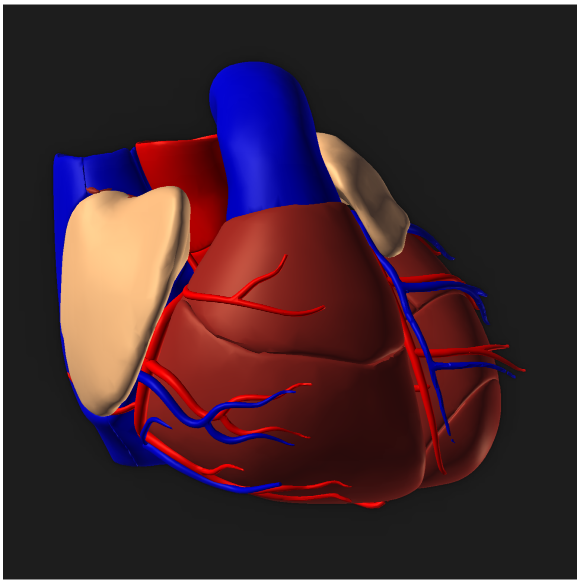

Figure 11: *Screenshot from the visualization from the front. One directional light coming from the top left side of the object.*

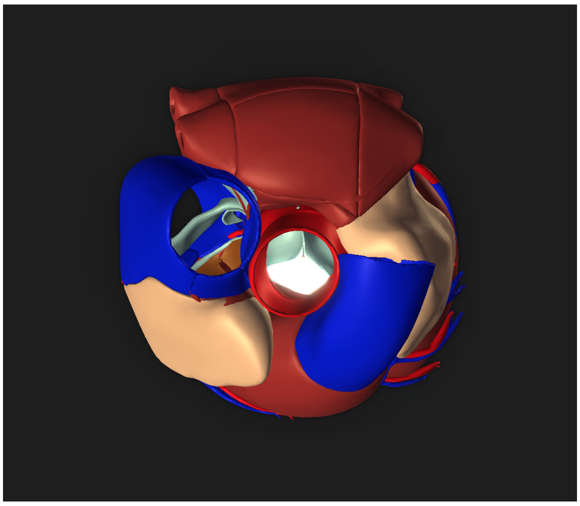

Figure 12: *Axial view with ventral part on the bottom, dorsal part on the top (rotated approximately* 90° *to the front) with the anterior cusp of the aortic valve emphasized*

To learn the names of the structures of the heart easier one interesting way could be to implement a text field that shows the name of the structure which the mouse is currently hovering over.

Another point of discussion we encountered is the problem how to properly show and highlight structures that are inside the heart or otherwise occluded from the viewers current perspective upon selection. In our visualization it is possible to hide each anatomical structure by performing a mouse click on it to be able to see structures inside the heart properly. One possible more advanced solution would be the implementation of automatic clipping of the rendered volume or hiding of structures with respect to the structure which is currently selected with the goal, that all structures that occlude the vision of the anatomical structure of current interest are hidden or clipped away automatically. This is of course far more complex since it would require logical awareness of where which structure is located in the volume rendered.

A further possible extension would be to provide expert anatomical information from scientific sources instead of Wikipedia articles in the panel displayed upon selection of a structure. This would leverage the value for medical students in anatomical education. Of course apart from technical issues the usage of this information would be subject to licensing terms of the scientific specialist publishers in the field of medicine.

Of course as mentioned in section Data, an extension of the rendering pipeline shown in figure 13, to be able to use data from CT(A) or MR(A) imaging, would be interesting. From figure 13 it can be concluded, that due to the usage of mesh models we only perform actions that belong to the step rendering (and due to color and transparency mapping of anatomical structures to a little extent to the step mapping) in this project and do not address or make use of the first two steps of the pipeline. Utilizing well segmented and labelled surfaces from a real CT or MR data set instead of mesh models would also add value to this visualization of the human heart.

Figure 13: *The rendering pipeline of a medical visualization from the slides of the course Medical Visualization 2, originally from [HM90]*

References

[3d420] [online]06 2020. URL: https://3d4medical.com/apps/complete-heart [cited 2020-06-13]. 2

[AA16] Azer S. A., Azer S.: 3d anatomy models and impact on learning: A review of the quality of the literature. *Health Professions Education 2*, 2 (2016), 80 – 98. URL: http://www.sciencedirect.com/science/article/pii/S2452301116300281, doi:https://doi.org/10.1016/j.hpe.2016.05.002. 2

[AMB*13] Auzinger T., Mistelbauer G., Baclija I., Schernthaner R., Köchl A., Wimmer M., Gröller M. E., Bruckner S.: Vessel visualization using curved surface reformation. *IEEE Transactions on Visualization and Computer Graphics 19*, 12 (2013), 2858–2867. 1

[BGP*11] Borkin M., Gajos K., Peters A., Mitsouras D., Melchionna S., Rybicki F., Feldman C., Pfister H.: Evaluation of artery visualizations for heart disease diagnosis. *IEEE Transactions on Visualization and Computer Graphics 17*, 12 (2011), 2479–2488. 1, 4

[Bio20] [online]06 2020. URL: https://www.biodigital.com [cited 2020-06-12]. 2

[blo20] [online]06 2020. URL: https://learnopengl.com/Advanced-Lighting/Bloom [cited 2020-06-13]. 6

[bod20] [online]06 2020. URL: https://lifesciencedb.jp/bp3d/ [cited 2020-06-12]. 3

[doc20] [online]06 2020. URL: https://flexikon.doccheck.com/de/Herz [cited 2020-06-13]. 2

[Hac16] Three-dimensional display technologies for anatomical education: A literature review. *Journal of Science Education and Technology 25*, 4 (2016), 641–654. URL: https://doi.org/10.1007/s10956-016-9619-3, doi:10.1007/s10956-016-9619-3. 2

[HM90] Haber R. B., McNabb D. A.: Visualization idioms: A conceptual model for scientific visualization systems. *Visualization in scientific computing 74* (1990), 93. 8

[KbA19] Keenan I. D., ben Awadh A.: *Integrating 3D Visualisation Technologies in Undergraduate Anatomy Education*. Springer International Publishing, Cham, 2019, pp. 39–53. URL: https://doi.org/10.1007/978-3-030-06070-1_4. 1

[KBT*12] Kretschmer J., Beck T., Tietjen C., Preim B., Stamminger M.: Reliable adaptive modelling of vascular structures with non-circular cross-sections. *Computer Graphics Forum 31* (06 2012), 1055–1064. doi:10.1111/j.1467-8659.2012.03098.x. 1

[ken20] [online]06 2020. URL: https://www.kenhub.com/de [cited 2020-06-13]. 2

[Mis10] Mistelbauer G.: *Automated processing and visualization of vessel trees*. 2010. 3

[OBJ] Objloader2parallel. https://threejs.org/docs/#examples/en/loaders/OBJLoader2Parallel. Accessed: 2020-06-16. 4

[ope20] [online]06 2020. URL: https://www.khronos.org/opengles/ [cited 2020-06-09]. 4

[Ray] Raycaster. https://threejs.org/docs/#api/en/core/Raycaster. Accessed: 2020-06-16. 5

[thr20] [online]06 2020. URL: https://threejs.org/docs/index.html#manual/en/introduction/Browser-support [cited 2020-06-09]. 4

[TSN*20] Triepels C. P. R., Smeets C. F. A., Notten K. J. B., Kruitwagen R. F. P. M., Futterer J. J., Vergeldt T. F. M., Van Kuijk S. M. J.: Does three-dimensional anatomy improve student understanding? *Clinical Anatomy 33*, 1 (2020), 25–33. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/ca.23405, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/ca.23405, doi:10.1002/ca.23405. 2

[vis20] [online]06 2020. URL: https://www.visiblebody.com/de/ [cited 2020-06-16]. 2

[web20] [online]06 2020. URL: https://www.khronos.org/webgl/ [cited 2020-06-09]. 4

[wik20] [online]06 2020. URL: https://www.mediaWiki.org/Wiki/API:Main_page [cited 2020-06-12]. 6

[ZWY*16] Zhang L., Wang K., Yang F., Lu W., Wang K., Zhang Y., Liang X., Han D., Zhu Y. J.: A visualization system for interactive exploration of the cardiac anatomy. *Journal of Medical Systems 40*, 6 (2016), 135. 2