Handheld Web-based Augmented Reality: Visualizing an MRI Dataset of a Frog

Ragger Jonkers - 10542604
Steven Raaijmakers - 10804242

October 2018

Abstract

In this project we will make use of a MRI dataset annotated for tissue of a frog. We will visualize this dataset in Handheld Augmented Reality to show that it is possible to create an interactive 3D visualization that can be used as a surrogate for real frog dissection, due to the freedom of exploration, created perception of depth and ease of access.

1 Introduction

In 1993 the Lawrence Berkeley National Laboratory conducted an experiment called the "Whole Frog Project" to show how a 3D visualization can be used as a curriculum tool (Robertson, Johnston, & Nip, 1995). A website was created on which it was possible to interact with a 3D visualization of a frog and its different parts. It allowed for study of the anatomy of an actual frog, without the need of frog cadavers. In this project we take it a step further, aiming at an augmented-reality web-visualization, accessible for everyone with a smartphone without the need of an app, only requiring a web browser. Using this visualization we hope to answer the upcoming research question with respect to the anatomy of the frog.

1.1 Research question

1. How can MRI data of a frog effectively be visualized?
   - (a) Which visualization techniques contribute to this perceptual effectiveness?
   - (b) How can we create a visualization that can easily be requested without the need of advanced equipment?

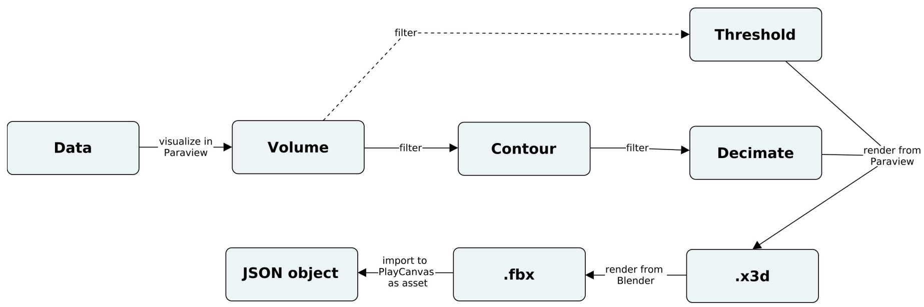

Figure 1: Visualization pipeline

2 Background

In this project we use a the dataset of Lawrence Berkeley. The dataset consists of two parts. One part is the MRI values of the frog, which are the raw measured values that describe the time it takes for the frog molecules to return to equilibrium state, hence highlighting the contrast between the tissues.

The second part of the dataset is the manually annotated part of the frog tissue. Per pixel is determined exactly which tissue it is part of. The values per data point are discrete and can be interpreted as categories, unlike the other part of the dataset. This second part excludes the skin as a category.

3 Method

In order to answer the research question we will describe the steps of visualizing the frog in handheld augmented reality. This process consist of two categories: converting the MRI data into *.x3d* and *.fbx* files and visualizing the latter in augmented reality.

3.1 Converting the MRI data

The converting process of the MRI data is schematically drawn in figure 1. It shows that we used Paraview[1] as our initial tool to explore the dataset by visualizing it in 3D. Specifically, the skin is extracted using a contour filter. Then a decimate filter is applied, not only to reduce the size of the object, but also to eliminate many of the steep edges that are present in the frog, because of the MRI slices it is segmented in. For the other tissues we extracted desired tissue by eliminating values in the view that would exceed the threshold value of the tissue category. On these selected tissues it is not possible to use a contour filter, due to their discrete values. Consequently the output could not be made polygonal to serve as an input for the decimate filter.

[1] `https://www.paraview.org/`

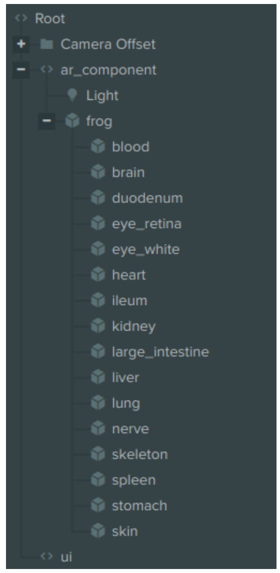

Figure 2: Structure of application in PlayCanvas

For the next step, all objects are rendered as a .x3d file through Paraview and converted to a .fbx file in Blender[2]

3.2 Visualizing

To visualize the created .fbx files we use PlayCanvas[3]. PlayCanvas is a web-oriented application engine in which augmented reality (AR) objects can be made. Applications created in PlayCanvas are web-based, meaning that they are accessible through a web-browser.

Within PlayCanvas we make use of the framework provided by the UvA [4]. In this framework we find an AR component which consist of an object that is mapped onto a marker, which is an image of the kanji-sign in our case. We replace the object by our own object: the frog. This frog object is made by importing the .fbx files, which are then being converted by PlayCanvas into .json files. These files are then being grouped into one model called "frog". The structure can be found in figure 2.

The tissues are assigned to different materials thus the viewer has a clear view of the distinct tissues. For example the blood-tissue is assigned to a red material, while the skin-tissue is assigned to a green shiny material, which we made opaque in order to minimize occlusion. We also added a light source, which we positioned above the frog. The light source will generate shadow on

[2] `https://www.blender.org/`
[3] `https://playcanvas.com`
[4] https://playcanvas.com/project/546615/overview/uva-framework–development

Figure 3: Frog in AR

the frog itself and on the ground the frog is projected on.

In the UI the viewer has the ability to manually show or hide certain tissues. Therefore the viewer will have the ability to study different parts of the frog in a clearer environment.

4 Results

The described method gave the result that can be found in figure 3. In this figure we see all the tissues of the frog being visualized at the same time. Due to the skin being opaque we can still see the contours of the skeleton and the organs of the frog.

We can also clearly see the influence of the light source on top gives, which gives a perception of depth. This can be seen in the shadows, underneath the frog, but also in the light that is reflecting on the frog skin. This behaviour is best observed when using the application.

We can use the menu of the UI to hide the skin to get a better overview of the inside of the frog. The result can be seen in figure 4.

The visualized frog can be zoomed and panned by changing the position of the camera with respect to the marker. The frog changes dynamically on this movement. This movement generates a continuous animation on a desktop computer. However, when using a device with lower specifications (like smartphones) the animation fluidness will deteriorate. This is due to the fact that a smartphone has a less powerful graphical component and thus needs more time to load the components. For example the skin-tissue is over 20 megabytes. Besides that, the impact of the light source also requires relatively heavy computational power.

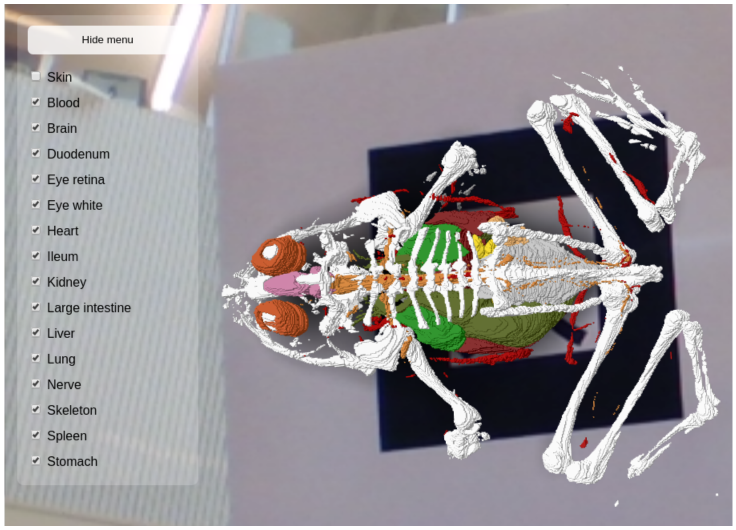

Figure 4: Frog without skin

5 Conclusion

The goal is to draw a conclusion of the effectiveness of the produced visualization. This is always tricky, because there is no hard measure to define this. However, we believe that the success in effectiveness can primarily be attained to the freedom of exploration that a handheld AR visualization provides. This, in combination with perception of depth caused by shadowing, and colorization of different tissues, allows the user to analyze and compare frog tissues rapidly. Furthermore, the AR application is able to run on any device with a web browser. This visualization technique could therefore also be adopted for educational purposes.

6 Discussion

In future work we could make use of lower resolution objects in order to facilitate rendering on smartphones.

Beside this the decimation filter is only applied to the frog skin, because this was the only data that we could successfully apply the contour filter to. Therefore the frog skin has a more rounded shape in contrast to the other tissues, which seem to be more angular.

A clipping/slicing plane is something we would have added, had there been more time. It improves the freedom of exploration, coming closer to the experience of "real life" frog dissection.

A Appendix

The interactive frog can be found on PlayCanvas: `https://playcanv.as/p/WxrJnepz/` The created user interface can be found in the framework under the folder called "UI". The script which handles the frog and its tissues can be found under "scripts > frogModel.js".

References

Robertson, D., Johnston, W., & Nip, W. (1995). Virtual frog dissection: interactive 3d graphics via the web. *Computer networks and ISDN systems*, *28*(1-2), 155–160.