



AXOLOTL

Team

1. Nicholas Verrochi
2. Vidhya Sree Narayanappa
3. Sairam Bandarupalli
4. Andy Duverneau

Client

McCusker Lab at Umass Boston

Teaching Assistant

Kiran

Professor

Daniel Haehn

Final Project Documentation v1.0 05/20/2023

1 Introduction

The Axolotl Project is an innovative endeavor aimed at assisting the McCusker lab in their research on the regenerative capabilities of Axolotls, specifically focusing on measuring changes in the layers of skin during limb regeneration. This project involves the development of specialized software that will enable the lab to analyze high-resolution, zoomable scans of Axolotl skin and accurately measure the boundaries between different skin layers.

1.1 Purpose

Assist the McCusker lab: The primary purpose is to provide the McCusker lab with a specialized software tool that will aid in their research on Axolotl limb regeneration. By developing software capable of analyzing high-resolution scans of Axolotl skin, the project aims to support the lab in measuring the changes that occur in the layers of skin during limb regeneration. This will help researchers gain a deeper understanding of the regenerative process and potentially uncover key insights into the mechanisms involved.

1.2 Scope

The creation of high-resolution images is one aspect of the Axolotl Project, but it is not the sole focus. The main scopes of the project include:

Software Development: The project involves the development of specialized software that will enable the McCusker lab to analyze the layers of skin in Axolotls during limb regeneration. This software will provide a viewer with zooming capabilities, allowing researchers to examine high-resolution scans of the skin layers in detail.

Measurement Capabilities: The software will have measurement functionalities that allow researchers to precisely measure the boundaries between different layers of Axolotl skin. This feature will simplify the process by allowing researchers to click on the specific locations where the layers begin and end, eliminating the need for manual measurements.

User-Friendly Interface: The software will be designed to be user-friendly, ensuring that researchers can easily navigate and utilize its features. The interface will be intuitive, making it accessible to both experienced and novice users in the field of regenerative research.

Data Analysis Efficiency: The project aims to streamline the data analysis process by automating measurements and eliminating manual and time-consuming procedures. The software will enhance efficiency, accuracy, and consistency in analyzing the skin layers of Axolotls, allowing researchers to focus more on the interpretation of data and the exploration of regenerative mechanisms.

Contribution to Regenerative Medicine: The ultimate goal of the project is to contribute to the field of regenerative medicine. By providing a powerful tool for analyzing Axolotl skin, the project aims to generate valuable data and measurements that can advance our understanding of regenerative processes. This knowledge may have implications for the development of regenerative therapies in humans.

1.3 Overview

Our goal is to create software to help the McCusker lab measure the changes in the layers of skin of Axolotls as they regenerate lost limbs. Our final product will allow the lab to pull up high-resolution, zoomable scans in a viewer and take measurements by simply clicking where the layers of skin begin and end.

1.4 Existing Works

In 2022, Allen Dai created the original Axolotl project where he used OpenSeadragon to create zoomable views of the Axolotl scans, as well as taking measurements from the views.

Web Viewer: <https://haehn.github.io/axo/web/>

Project Report: https://github.com/allendai1/cs460student/blob/main/axo/CS460_Final_Project_Report.pdf

Source code: <https://github.com/allendai1/cs460student/tree/main/axo/>

1.5 Limitations

Sample Availability and Variability: The availability of Axolotl specimens for research purposes may be limited. Obtaining an adequate number of specimens with consistent regenerative abilities and suitable for high-resolution imaging can be challenging. Additionally, individual variation among Axolotls in terms of genetics, age, and health could influence the regeneration process and introduce variability in the data.

Imaging Challenges: Capturing high-resolution, zoomable scans of Axolotl skin may present technical difficulties. Ensuring consistent image quality, minimizing artifacts, and addressing issues like motion blur or tissue distortion during imaging could be challenging. These factors may impact the accuracy and reliability of the measurements obtained from the images.

Tissue Preparation and Staining: Preparing Axolotl skin samples for imaging and analysis may involve specific procedures, such as tissue fixation, embedding, sectioning, and staining. Each step introduces the potential for artifacts or tissue alterations that may affect the accuracy of the measurements or distort the representation of the skin layers.

Image Analysis Algorithms: Developing robust algorithms to analyze the layers of Axolotl skin in high-resolution images can be complex. Automating the detection and measurement of skin layer boundaries accurately and reliably is a challenge that may require sophisticated image processing and analysis techniques. Ensuring that the algorithms are optimized, validated, and adaptable to different types of images and variations in skin structures can be demanding.

Interpretation and Generalization: While the project aims to provide insights into Axolotl limb regeneration, the findings may be specific to Axolotls and may not directly apply to other species or regenerative processes. Care must be taken when generalizing the results to other contexts or species, including humans, as there may be biological differences that affect regenerative mechanisms.

User Expertise and Training: The software developed for analyzing the images and measuring skin layer boundaries may require training and expertise to use effectively. Ensuring that researchers are adequately trained and supported in utilizing the software's capabilities can impact the consistency and accuracy of the measurements obtained.

2 Requirements

The user will be able to zoom in and take the measurements via clicking in the zooming viewer. Start and end points of measurements will be clearly displayed, and the viewer will use a coordinate system to describe where these points are within the high-resolution image. The user will be able to save their measurements as well as reset the ones they've already taken. Users will be able to annotate images as well. The application will load images quickly and perform annotations and measurements with no noticeable delay.

1. Accuracy and Precision: The software and measurement tools should provide accurate and precise measurements of the boundaries between skin layers. The algorithms used for measurements should be validated and capable of handling variations in image quality, noise, and other potential challenges

2. Image Data : we used some Image (Young scans) data <https://haehn.github.io/axo/web/>

3 Specifications

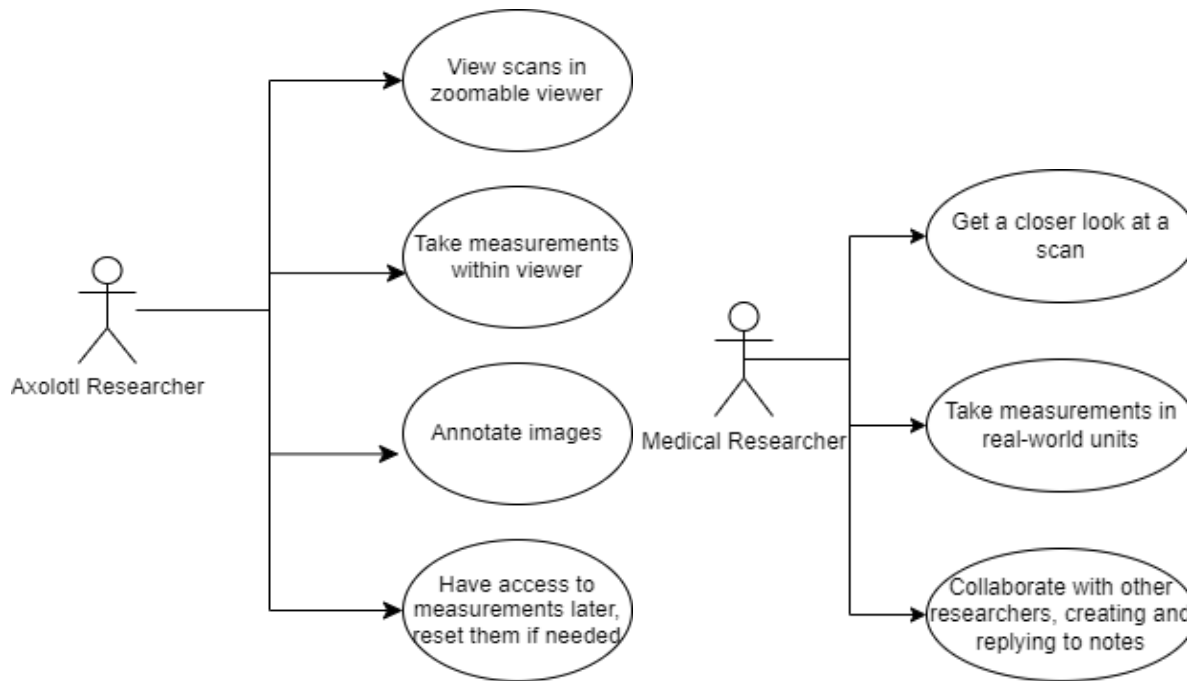
We will use JavaScript libraries to create a client-side web application that requires no installation. The only algorithms needed are the calculation of distance between two points, and the conversion from pixels to real-world units using the scale of the image. Image sets will reside on the server, and measurements will be stored using local storage for persistence.

1. High-Resolution Imaging : The project involves capturing high-resolution images of Axolotl skin samples.
2. Zoomable Image Viewer: This enables closer inspection and analysis of specific areas or features.
3. Measurement Tools: A length tool on a high-resolution image is a specific measurement tool that allows users to measure the length or distance between two points accurately.
4. User-Friendly Interface: The software should have an intuitive and user-friendly interface that enables easy navigation, selection of measurement tools, and visualization of results. The interface should be designed with usability in mind, considering the needs of both experienced researchers and new users.

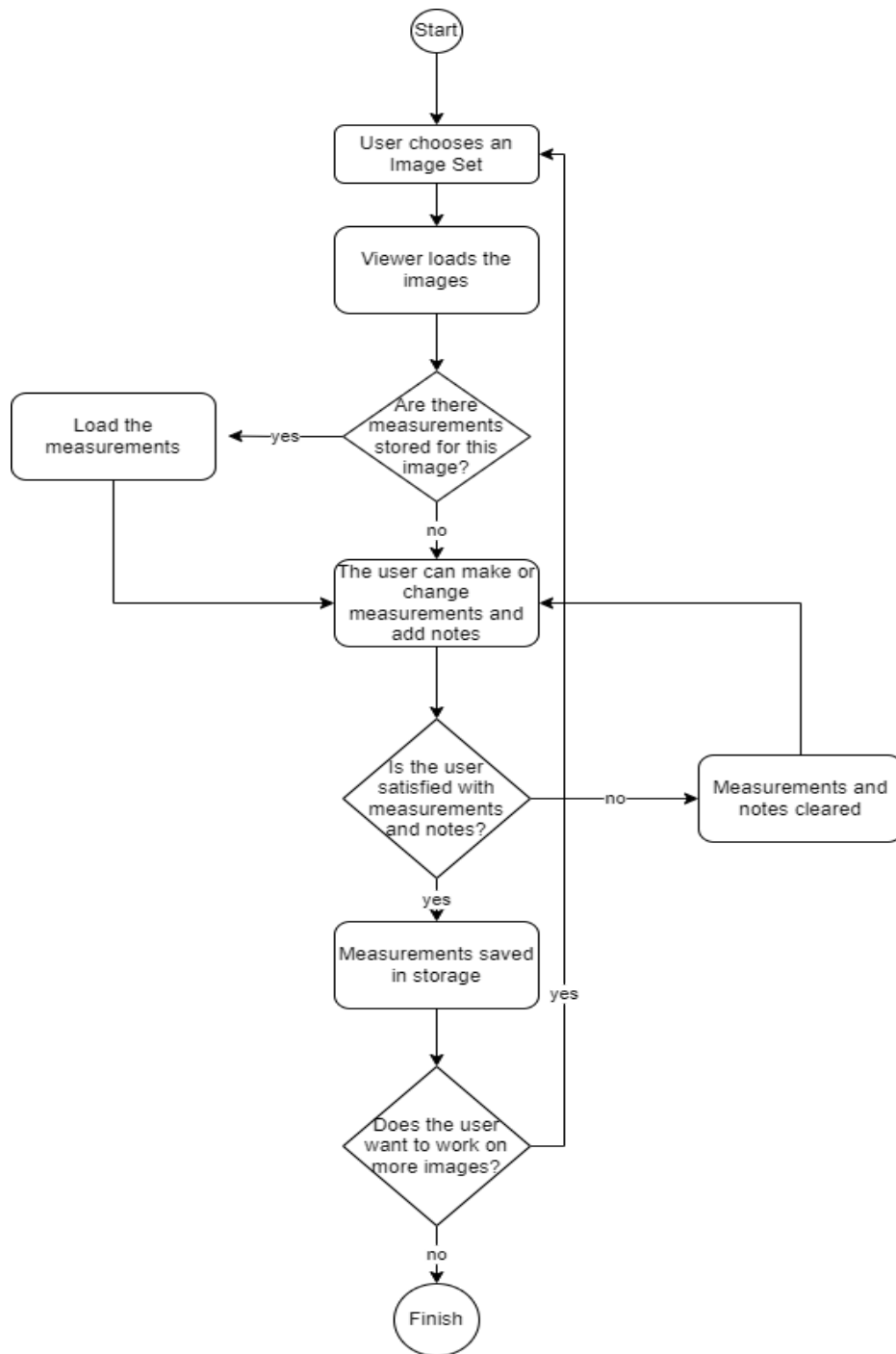
4 Design

We will use typical HTML and CSS to create the user interface and JavaScript for the logic. For the image viewer, we tried to combine the high-resolution, zoomable views of the OpenSeadragon library with the Cornerstone tools for medical imaging which has features such as measuring and annotation, but were unable to combine them in a timely manner. We instead used Annotorious for annotations and Fabric.js to render measurements on the image and perform calculations ourselves.

4.1 Use cases

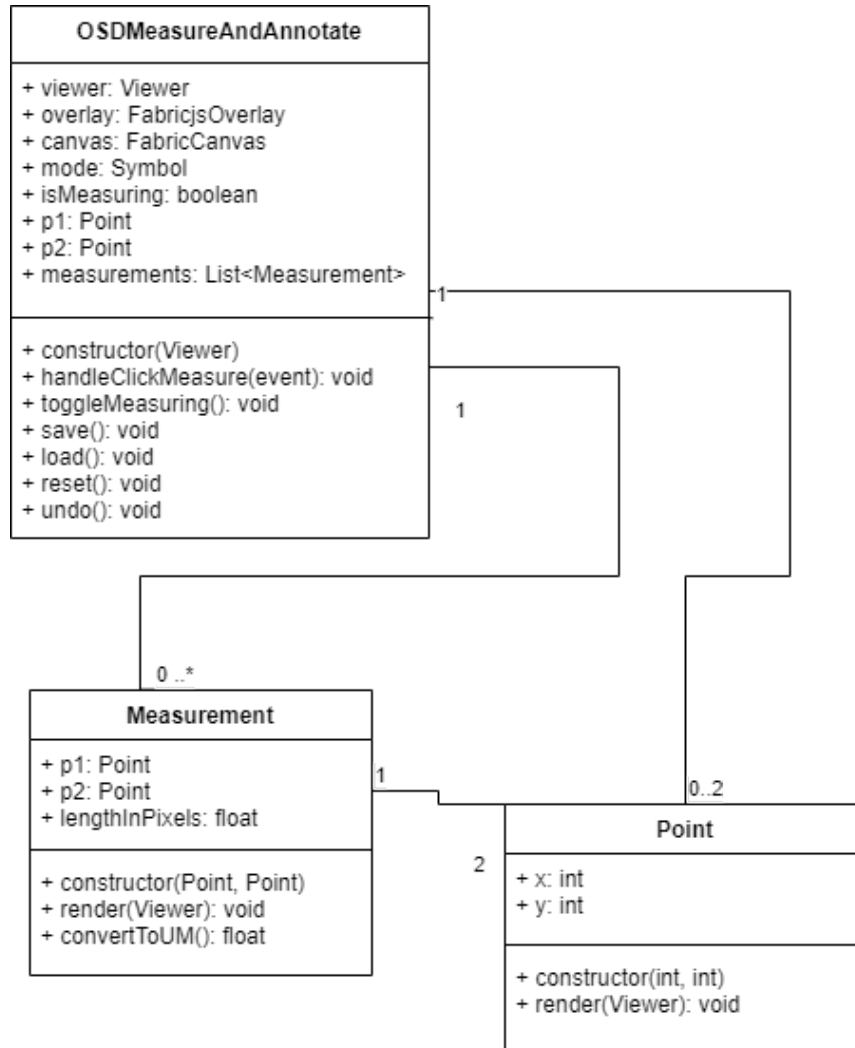


These diagrams represents the capabilities the application offers researchers who may be using the app to view their scans. Besides Axolotls, our software will work with other data as well, and could be useful in medicine and other sciences. Users can access several features with ease and make their work much easier.



This above diagram describes the order in which the user interacts with the program. Our application’s flow is user-driven, whenever the user makes a mistake or isn’t satisfied with their notes and measurements, they can go back and make the changes they need. If they’ve already started working on a scan, their work will be saved and loaded again should they need to work more on that scan.

4.3 Class Diagram



This diagram below describes how we structured the plugin we used to add measuring and annotation functionality to the OpenSeadragon viewer. We took a modular approach to the plugin for faster, more organized development. We used classes to represent points and measurements so we can store and re-render them as the viewer zooms or if the user wants to save their data. The library we used for annotations already had an interface for accessing annotation data, so we used that.

University of Massachusetts Boston

5 Technologies and Prototypes

OpenSeadragon: an open-source Javascript library developed by Ian Gilman to create zooming viewers for high-resolution images.

Annotorious: a Javascript library to annotate images.

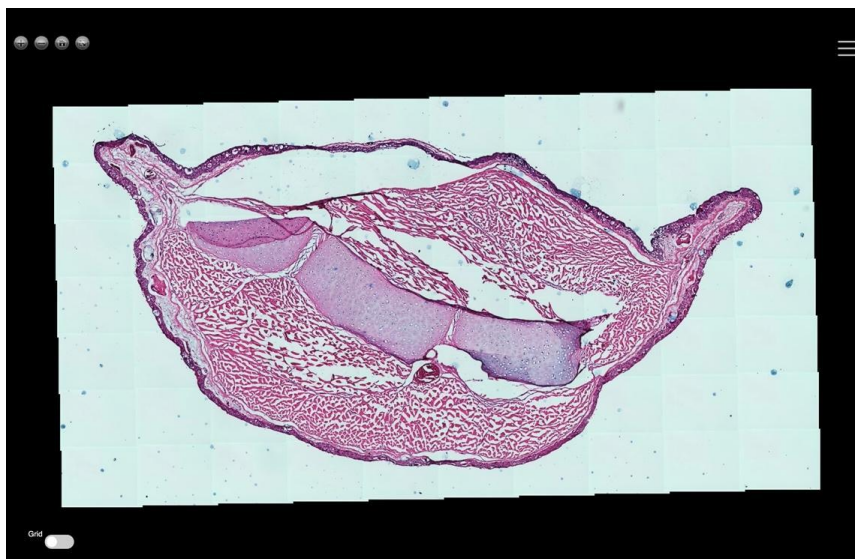
Fabric.js: a Javascript library that allows creation of objects on HTML canvases.

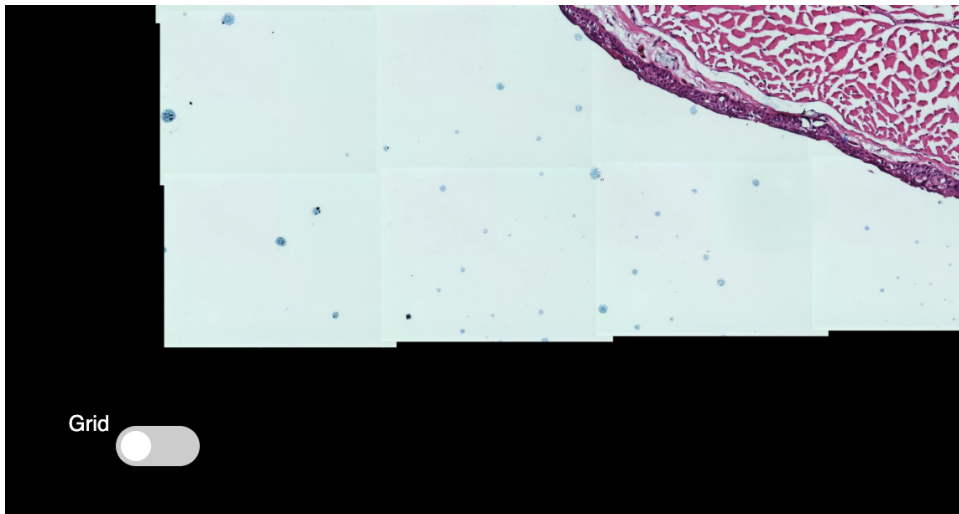
5.1 Zoom In



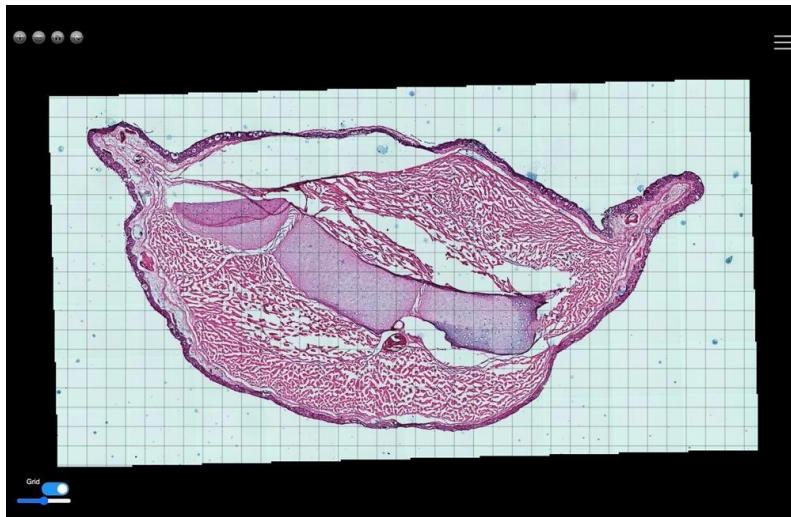
The zoom-in option for high-resolution images allows users to examine the image at a more detailed level by increasing the magnification. This feature is particularly useful when working with high-resolution images, as it enables users to explore fine details and intricate structures that may not be visible at a normal viewing level. As the user zooms in, the image expands, revealing finer details.

5.2 Zoom Out





When grid is on and off



University of Massachusetts Boston

A grid toggle button for a high-resolution image allows users to overlay a grid pattern onto the image for various purposes, such as measurement, alignment, or reference. This feature can be particularly useful when working with precise measurements or analyzing specific features within the image.

Here's how a grid toggle button for a high-resolution image might function:

Grid Overlay: When the user activates the grid toggle button, a grid pattern is superimposed over the image. The grid consists of evenly spaced horizontal and vertical lines that divide the image into a series of cells or squares.

Grid Customization: The software or viewer may provide options to customize the grid, such as adjusting the spacing between the lines or changing the color and transparency of the grid overlay. This allows users to tailor the grid to their specific needs and preferences.

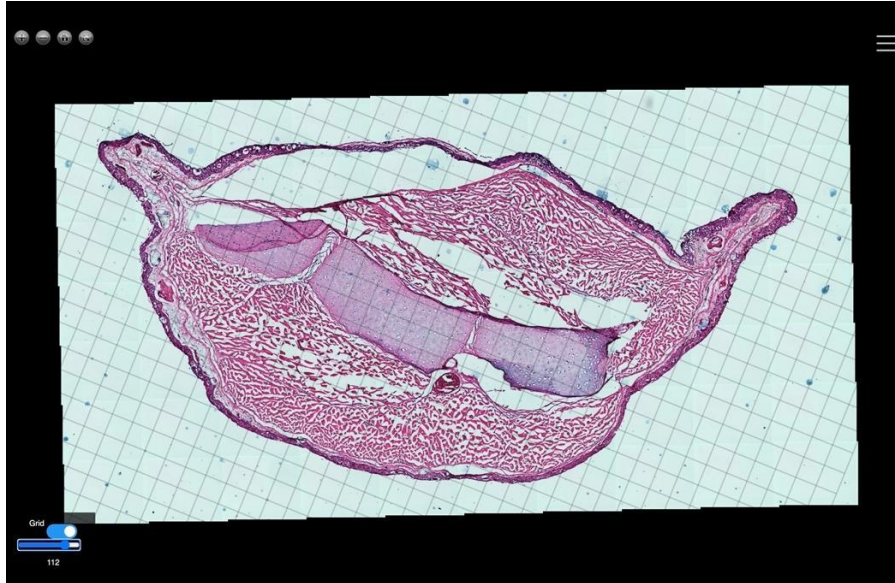
Measurement Reference: The grid overlay serves as a visual reference for measurements within the image. Users can align objects or features within the image with the grid lines to obtain accurate measurements or compare sizes and distances.

Grid Visibility: The toggle button allows users to turn the grid overlay on and off, providing flexibility in displaying or hiding the grid as needed. This ensures that the grid does not obstruct the view or interfere with other image analysis tasks when not required.

Grid Alignment: The grid overlay can assist with aligning objects or structures within the image. By aligning key features with the grid lines, users can ensure precise positioning and alignment when comparing or analyzing different regions of the image.

Grid Snap or Lock: In some cases, the software or viewer may offer a snap or lock feature that enables objects or annotations to align with the grid lines automatically. This helps maintain consistent alignment and measurement accuracy throughout the analysis process.

By incorporating a grid toggle button for a high-resolution image, the software or viewer enhances the user's ability to perform accurate measurements, align objects, and make precise observations. It provides a visual framework that aids in quantitative analysis and improves the overall usability of the image analysis tool.



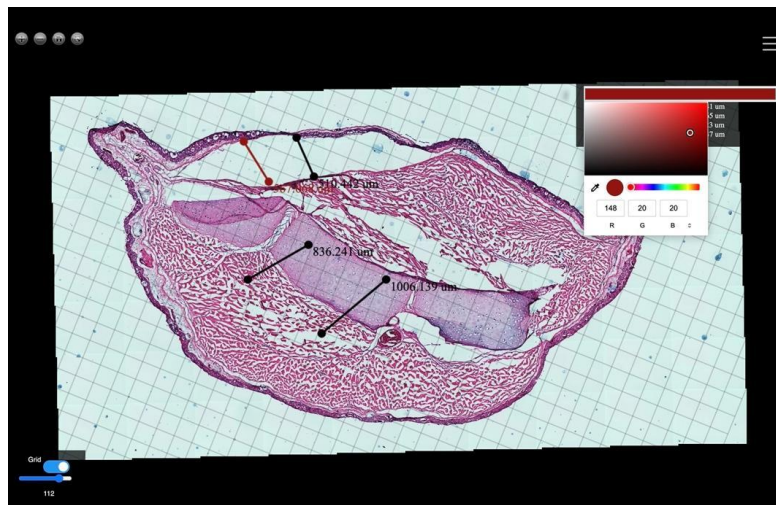
Angle Selection: The user can input the desired angle of rotation or adjust it using controls such as a slider or input field. The angle can be specified in degrees or other relevant units.

Grid Rotation: Upon selecting or adjusting the angle, the software rotates the grid overlay accordingly. The grid lines now align with the specified angle within the image.

Measurement and Alignment: The rotated grid overlay can assist in measuring angles or aligning objects within the image accurately. Users can compare the orientation of features or structures with the rotated grid lines to obtain precise angle measurements or ensure proper alignment.

Grid Visibility and Interaction: The software allows users to toggle the visibility of the rotated grid overlay on and off as needed. Additionally, users may have the ability to interact with the grid, such as adjusting the size, spacing, or transparency of the grid lines to optimize visibility and usability.

5.5 Measurements



University of Massachusetts Boston

A length tool on a high-resolution image is a specific measurement tool that allows users to measure the length or distance between two points accurately.

Multiple Length Measurements: Users can make multiple length measurements within the same image session. This allows for efficient and simultaneous measurement of multiple objects or distances.

Length Tool Selection: Users select the length measurement tool from the software or viewer's options or toolbar.

Color Selection: Instead of a single default color, the software or viewer allows users to choose different colors for their length measurements. This can be done through a color palette, color picker, or a predefined set of colors.

Placing Length Measurements: Users click or drag on the image to define the starting and ending points for the length measurement. As each measurement is made, it is assigned a specific color.

Color Differentiation: Each length measurement is displayed with its assigned color, creating visual differentiation between the measurements. The lines representing the measurements are displayed in their respective colors.

This can aid in clarity, organization, and interpretation of the measured distances within the high-resolution image, particularly when dealing with complex or numerous measurements.

7 Milestones

Zooming factor doesn't affect the measurements: This was a huge problem when we tried to connect Cornerstone and OpenSeadragon - the measurements would not adjust with the zoom, and stay in the same location in the window rather than on the image.

Obtaining image coordinates to calculate measurements: This was luckily made easy by OpenSeadragon's API, using the coordinates we could make accurate measurements.

Good annotations that allow for collaboration and aren't affected by zoom: This was important as we had many issues with annotations, in particular there are keyboard shortcuts for OpenSeadragon that are difficult to override, with Annotorious we didn't have to worry.

Saving and loading data client-side in local storage: Although Cornerstone would have been nice as it provides direct access to measurements and annotations, we built class models for measurements to easily store them, and Annotorious gives direct access to annotations. We used the JSON format to store an array of objects as a string within a single entry to local storage.

University of Massachusetts Boston

8. References

- 1) <https://github.com/allendai1/cs460student/tree/main/axo> (Allen's source code)
- 2) <https://haehn.github.io/axo/web/> (Working version of the original project)
- 3) <https://openseadragon.github.io/> (OpenSeadragon's official page)
- 4) <http://fabricjs.com/> (Fabric.js official site)
- 5) <https://annotorious.github.io/> (Annotorious official site)

GitHub Links:

The project repository: <https://github.com/verrochi92/axolotl>

The OSDMeasure plugin: <https://github.com/verrochi92/OSDMeasure>

GitHub pages: <https://verrochi92.github.io/axolotl/>