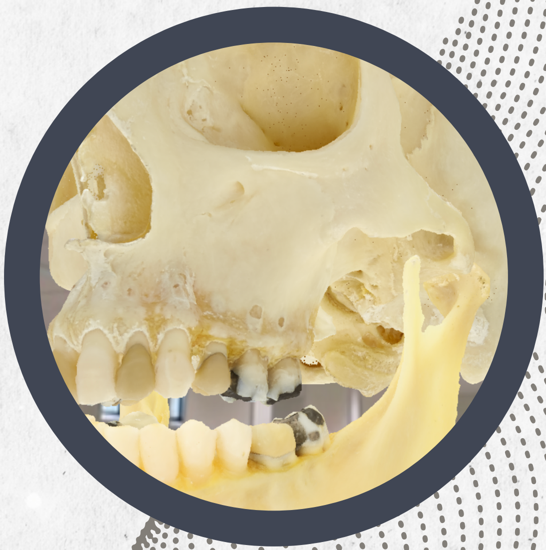
TU Delft, Faculty of Architecture and the Built Environment

SYNTHESIS REPORT

Project 4: Pointcloud based anatomy



Project members:

Vidushi Bhatt (5862124)
Sharath Chandra Madanu (5722101)
Shen Qiwei (5687500)
Susanne Epema (5655730)
Gees Brouwer (4726693)

TU Delft supervisors:

Edward Verbree & Peter van Oosterom

Company supervisors (Enatom):

Bastiaan Hofsteenge

https://github.com/GEO1101-Synthesis-Group4/Selection-Annotation-Repo

November 2023

Abstract: The declining availability of practical hours for medical anatomical education has prompted Enatom to develop a digital anatomical platform, utilizing the open-source WebGL-based point cloud renderer Potree. This platform, which employs detailed point cloud scans of anatomical structures, aims to offer a dynamic and interactive educational experience. Although Enatom's focus is not directly on geomatics, the techniques employed in this project have strong parallels with those used in geomatics, thereby enabling a symbiotic exchange of expertise. This interdisciplinary approach enhances the development of Enatom's digital platform, with the potential to contribute to the field of geomatics. To address existing user experience challenges, this project has added a lasso-selection tool tailored for Potree, advanced annotation capabilities, and methods for sensitive data anonymization within the point cloud. The project's outcomes will be available in an open-source format at https://github.com/GEO1101-Synthesis-Group4/Selection-Annotation-Repo. This project exemplifies the versatile application of geomatics expertise beyond its traditional scope, demonstrating its potential in enhancing diverse domains such as medical education.
Key words: Medical Education, Human Anatomy, Geomatics, Point Cloud Interaction, User Experience in Educational Tools, Potree, Lasso Tool, Annotation.

Contents

List of Figures

1. Introduction

This chapter summarizes the project, including the problem and what has been worked on. Afterwards the relevance of point clouds in geomatics is mentioned, and lastly a reading guide for this report is given.

1.1 Summary of project

Medical care and education are under a lot of pressure to provide fast, flexible access to adequate knowledge. Indispensable knowledge in this field is human anatomy. Many medical education institutions are not able to provide enough practical hours for students to study anatomy, or any practical education at all. Traditional methods require the use of cadavers, which is immensely resource-intensive, but also face logistical challenges.

The company Enatom aims to provide a solution to this problem. This solution makes use of dense point cloud scans of anatomical structures, and is therefore as close to reality as possible. A 3D, web-based application, utilizing the open-source WebGL based pointcloud renderer Potree is developed, which students can use to see every detail of the real human body without the need for physical specimens.

While Enatom's operations are not directly related to geomatics, the techniques they use to realize this 3D model bear resemblance to those utilized in the field of geomatics. Although geomatics focuses on spatial and earth-related data, the underlying principles of visualization, data acquisition and analysis can be applied to other domains. Therefore, our expertise in geomatics can support the development of the model Enatom has been working on. Moreover, the knowledge gained from this project could be applied within the field of geomatics, enhancing proficiency in both areas.

This collaboration with Enatom aims to optimize this application by mainly focusing on enhancing the user experience. Based on our understanding of Enatom's need we discovered a number of potential add-ons that could be developed. Given the limitation of time, we focused on a few of these under this project:

- A lasso tool for precise region selection within the point cloud
- An advanced 3D annotation system, to make the platform even more efficient and informative.
- Exploring anonymization methods for areas containing sensitive information in the point cloud.

1.2 Point cloud applications in geomatics

Point cloud representations play a pivotal role in reshaping how we model Earth's surface and urban structures in geomatics. Their precision in capturing extensive spatial information enables the creation of detailed digital models that faithfully reflect the complexities of our physical environment. The comprehensive geomatics research approach, from advanced data acquisition technologies like LiDAR and photogrammetry to sophisticated algorithms for processing and analysis, transforms raw point cloud data into actionable intelligence. This versatile process fosters applications across diverse industries, showcasing the broad impact of point cloud representations.

And autonomous driving is one of the most prominent fields applying point cloud data. For autonomous driving, accurate environmental perception and precise localization are essential[Li et al., 2020]. Point cloud data, with its precision and direct acquisition capabilities, plays a vital role in this domain. Current research in this field is extensively focused on deep learning-based methods for scene understanding and object detection using point cloud data. These methods leverage the high-dimensional nature of point clouds to extract detailed spatial features, enabling more accurate object recognition and scene interpretation. Advanced neural network architectures, such as PointNet++[Qi et al., 2017], Ponint Transformer[Zhao et al., 2021] and their variants, have shown remarkable ability in processing point cloud data efficiently.

Furthermore, state-of-the-art research on point cloud is also happening at TU Delft. For instance, Recently van Oosterom et al. did research on organizing and visualizing point clouds with continuous details. They presented a method based on optimized distribution of points over continuous levels, avoiding visualization shocks [Oosterom et al., 2022]. Dardavesis et al. recently worked on indoor location tracking using LiDAR point clouds and images of the ceilings [Dardavesis et al., 2023] where they had observed encouraging results for users in emergency while the user data acquisition was static.

1.3 Reading guide

This report starts off with defining the problem that was worked on. Following is an extensive chapter on the theoretical background needed and methods to develop the tools needed to solve the issues described in the problem definition. The chapter after this describes the results obtained, which are mainly the developed tools within the Potree environment. The report ends with a concluding chapter, summarizing what has been achieved, and the link to geomatics, and finally a short description of future study is given. The appendix contains the project organisational structure and rich picture of the project.

2. Problem definition

In this chapter, we delineate the current hurdles encountered in the Enatom application development, as identified through discussions with the company. The objectives are outlined in Section **??**, explicitly detailing the issues addressed in the course of this project. In Section 2.3 we delineate the project timeline and the broad methodology adopted for project execution in the eight weeks.

2.1 Description of challenges

Currently the capabilities and applications of photogrammetry and corresponding point clouds are more popular in the domains of built environment, robotics, manufacturing and design. Consecutively, the software and practices have been developed to cater needs and specifications for these sectors. The primary problem definition being worked on in this synthesis project is:

"Exploring efficient methods of visualizing, annotating and interacting with the objects of human anatomy using its point cloud representation"

Capturing the details of human anatomy through photogrammetry is a less explored application of Geomatics. Given this nascent application there are a number of challenges that the client (Enatom) has been working on. Under this project we examined each of these challenges and worked on developing solutions for some:

- Spaces between points in the point cloud: The inherent point cloud with points captured through photogrammetry have gaps in between when viewed closely. This is because the objects visualized have been captured through pixels of images stitched together.
- Loading of points is heavy: Millions of captured points have to be loaded for visualization of each frame (angle) at which one is observing. This is a machine-intensive task that can potentially cause lag or crashing.
- Classification of points in the point cloud: This is challenging for multiple reasons:
  - Dense point clouds: Since the anatomical objects are dense point clouds (and not discrete objects), it is harder to make selections. Point Clouds are inherently unstructured without any topological relationships between points (unlike vectors that can have topological relationships). Each point exists independently of other points.
  - Multiple Hierarchies: The human anatomy is structured in a way that there exist multiple levels of body organs. This makes selecting the exact classification level and/or classification category harder.
  - Specialized terms: The terminologies and objects for classification are known to medical students who collaborate with Enatom. There is yet insufficient training data for utilizing ML for automated classification.
- Dynamic Point Cloud: We wanted to explore the possibility of adding dynamicity (temporal dimension) in point cloud. For example, point cloud model of a hand emulating the movement of hand. This is also called 4D point cloud with 4th dimension as time.
- User capabilities (clicking and knowing which body part/subpart): The client hopes to add additional user capabilities to the existing platform.

- Privacy/Ethics (Anonymising face and tattoos): There are privacy concerns which call for anonymising of the individuals who are subjects of anatomical study.
- Encryption of the pointcloud for safe transfer of the sensitive data.
- Canvas remarks (currently in 2D)
- Measurements in point cloud - volume

Due to time limitations in and the scope of this project, not all identified problems can be worked on. Thus, certain problems to tackle have been decided on of which completing the tasks seem feasible during this project. Furthermore, there is not necessarily a need to complete the functionalities to its full potential. A starting point for the tools can be developed, but since they serve the user experience, user input needs to be taken into account to further improve the functionalities.

2.2 Objectives

The synthesis project in the geomatics program lets students work on a problem defined by a company that is not involved in the geomatics program, in our case Enatom. This project aims to give our team members insights in the running of a project in which multiple stakeholders are involved, to gain knowledge in point cloud visualisation via potree, and experience in the development of web-based tools. Furthermore, knowledge from the first-year courses of the geomatics program can be applied to this project and expanded upon.

The development process in this synthesis project mainly focuses on improving the annotation process of the point cloud, and on the development of an improved selection tool. Therefore, the research questions become

1. How can an object selection mechanism be developed to extract a set of points from point cloud and assign classification label to them?
2. How can a web-based annotation function be designed and implemented to enable multi-perspective content plane visibility, exportability, and other possible features within a web viewer platform?

Besides this, Enatom is dealing with sensitive data. When displaying a point cloud of someone's face, the face should not be recognizable to protect the privacy of this person. Therefore, an anonymization tool should be developed, which preferably can be toggled on and off by authorized users, and still have as much detail as possible. Therefore, the final objective is:

3. How can sensitive information be anonymized while losing as little anatomical information as possible?

2.3 Project Organization and Timeline

This section lays out the process followed under the synthesis project in terms of timeline, collaboration, implementation, and feedback.

At the initiation of the project, a broad structure for the process, the guidelines and the context had been laid down by the supervisors at the beginning of the project. The client (Enatom in our case) provided us with the knowledge about existing platform, its limitations, and their vision about the capabilities they wanted to add to their platform.

A constant flow of discussions among the team members and between the team and supervisors ensured that the objectives laid out for the project were being carried out in the correct way.

Prioritization

The project was planned such that the priorities were decided upon in advance using the MoSCoW prioritization method [“must have,” “should have,” “could have,” and “won’t-have (this time)”]. This meant recognition of the scope of work and the limitations in advance and staying focused on realistic goals.

Timeline

In the course of eight weeks, the project was implemented such that sufficient time was allocated to identification of the issues, followed by a detailed assessment of the technical aspects around the project and finally the implementation and improvement of the tools developed.

A weekly summary of the activities focused on in each week has been provided in image 2. The initial two weeks focused on the assessment of project fundamentals and finalization of a problem statement. Weeks three and four forayed into the technical possibilities for meeting the objectives identified initially. Weeks five and six were focused on actual implementations of the tools developed while the final two weeks focused on debugging and finalizations (Refer Appendix A.4 for more detailed explanation timeline of this project).

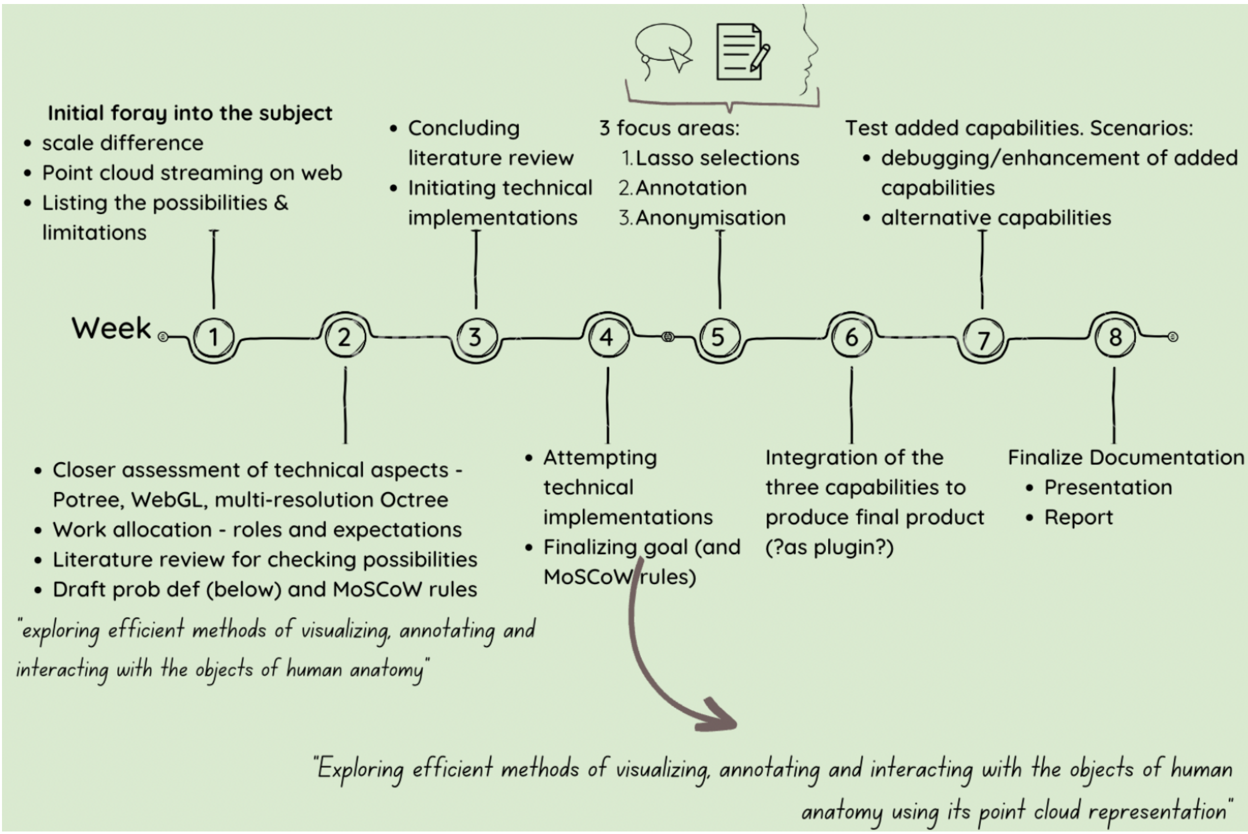


Figure 2: Project Timeline and Planning

3. Methodology and Theoretical Background

This section will elaborate on how this project will attempt to answer the research questions. The current version of the 3D point cloud model by Enatom has to be understood completely to be able to build the necessary tools on top of it. The model uses Potree to visualize the point cloud, and thus firstly this point cloud renderer will be discussed in detail. Then the three topics of the research questions, thus the annotation tool, the selection tool and the anonymization process of identifiable features will be expanded upon in detail.

3.1 Potree

Developed by Markus Schütz, Potree is an open-source WebGL-based point cloud renderer that excels in managing large and complex point cloud datasets [Schuetz, 2016]. Potree makes use of Multi-Resolution Octree (Multi-Res-Octree) algorithms, which allows it to maintain a consistent frame rate while rendering substantial point cloud datasets. This feature ensures that users can smoothly explore and analyze intricate 3D data without performance degradation. At the heart of Potree lies the powerful Three.js framework, which is itself based on WebGL — a web technology rooted in OpenGL, as shown in Figure [**?**]. This architecture leverages the capabilities of modern web browsers to provide high-performance 3D graphics rendering. By building upon Three.js and ultimately on WebGL, Potree taps into the vast potential of GPU acceleration, making it possible to visualize and manipulate point cloud data directly within web browsers. The Multi-Res-Octree structure employed by Potree is pivotal to its performance. This hierarchical data structure organizes the point cloud data into octree nodes of varying resolutions, allowing Potree to load and render only the necessary data for the current view. As a result, Potree can maintain a consistent frame rate even when dealing with extensive point cloud datasets. Users can seamlessly zoom, pan, and rotate through their data, enjoying a responsive and fluid experience.

The three primary factors that contribute to managing of slow internet connection on Potree are:

- File Format
- Progressive loading and rendering
- Point-wise adaptive point sizes

To improve the rendering speed, Potree deploys **Poisson-disk subsampling** method. Poisson-disk sampling is commonly used in computer graphics and point cloud processing to select samples in a way that ensures a minimum distance between the selected points. In Potree, it is applied as a subsampling (since points are already sampled first based on the octree hierarchy) to produce evenly spaced subsets with reduced occlusion from overlapping points.

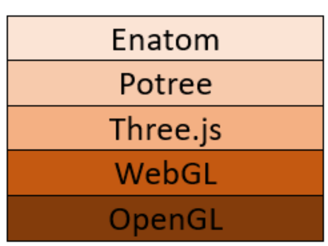


Figure 3: The structure of dependencies of the project

3.1.1 Development process with Potree

One of the key advantages of working with Potree is its use of JavaScript as the primary development language. JavaScript is a widely adopted and accessible language, making it easier for our team to write code, collaborate, and iterate swiftly. Additionally, the availability of comprehensive documentation for Three.js, upon which Potree is based, offers invaluable support in understanding the underlying architecture and maximizing the potential of our software.

Our approach to extending this Potree-based software involves building upon existing functionalities, implementing new features, and tailoring the user experience to our project's unique needs. With a foundation in Potree and the extensive resources provided by the Three.js documentation, we are well-equipped to tackle these development tasks effectively.

The following subsection 3.1.2 discusses all the capabilities that Potree offers on web platforms in general. It also explains how the general tools in Potree are limited in their applicability to Enatom. This is followed by, subsection 3.1.3 which highlights the existing usage of Potree renderer by Enatom.

3.1.2 Existing user capabilities on Potree-based web platforms

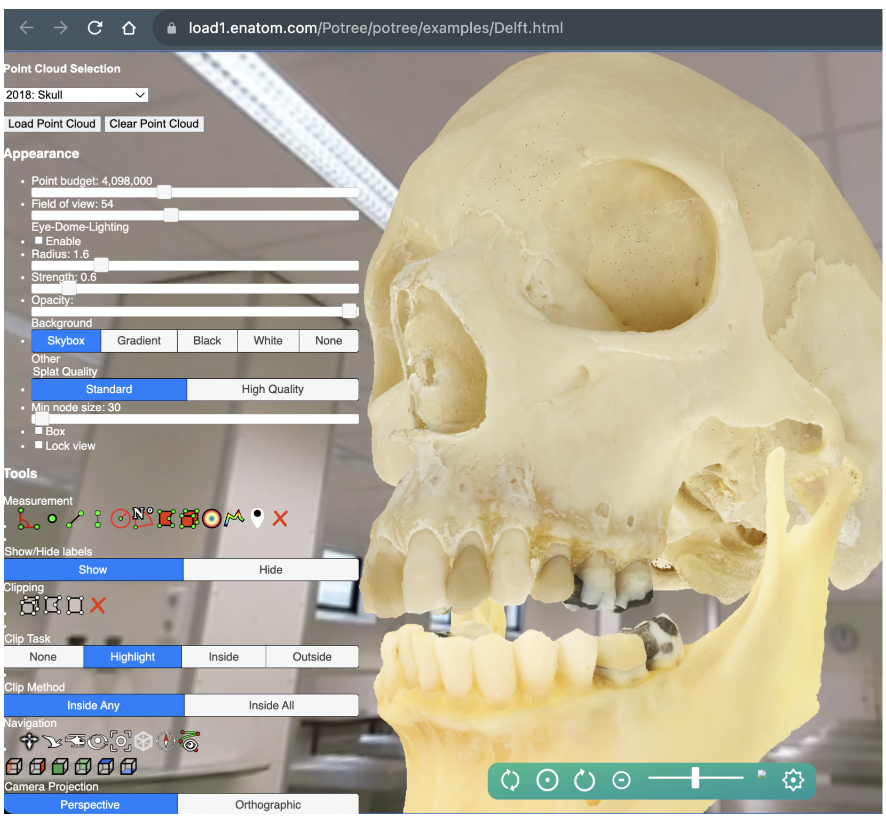


Figure 4: Interface of a Potree based web application

There are a number of tools that Potree offers for web-based 3D point cloud visualization and interaction.

- **Appearance** - A number of user capabilities to modify the appearance of point cloud are offered by Potree and included in the Enatom web application's interface. These include
  - Point budget - A point budget restricts the number of points that are both loaded and displayed concurrently, effectively tailoring the performance demands to match the varying capabilities of different hardware configurations. On Potree (and Enatom web app) it ranges from 100,000 to 10,000,000
  - Field of view (FOV) - It refers to the extent of the observable space that can be seen by the viewer. Potree (and Enatom web app) allow a range of 20-100 FOV
  - Eye-Dome Lighting illumination - this tool is used to enhance the depth perception of a scene and make identification of object shapes easier. It has been incorporated into Potree (and thereby on Enatom web app) as originally developed by Christian Boucheny. Once enabled, the Eye-Dome Lighting offers further options for the extent of lighting illumination - radius, strength, and opacity of the illumination effect.
  - Background - Potree (and Enatom web app) allow users to select a background for viewing the 3D point cloud. The options include skybox, gradient, black, white, and none.
  - Splat quality - the rendering quality of points on the viewer. This includes selecting size of nodes and having a 3D grid displayed.
- **Tools** - Potree also offers some inherent measurement and label tools options on the viewer. These include
  - **Measurement tool** - A number of 3D measurement tools are available that allow measuring angles and distances in 3D spaces. The drawback with Potree's inherent measurement tools in that they operate as standalone tools irrespective of which points the user might want them to be latched on to.
  - **label** - The only option available on Potree is to view or hide labels, i.e. there is no option currently to add labels to the point cloud. This makes sense and Potree was created with point cloud visualisation as the focus and not for processing. However, in this project we looked into the possibilities of adding the labels to the pointcloud after selecting them with lasso selection. The actual implementation and the extent of success achieved under this has been covered under Chapter 4
  - **clip task** and clip method - these tools allow a way of selecting or highlighting points within one of the clipping tool shapes. However, the selection/highlight is limited by these shapes and there is no method to modify the selection.

Limitations of Potree for use by Enatom
Potree is a revolutionary viewer for fast rendering and viewing of point clouds over the web. It allows Enatom (and many other applications) to be accessed by users easily and without having to load original data on their devices. However, there are a few limitations of Potree with respect to its use by Enatom. Some of these were worked on by us under this project while some have been mentioned under scope for future work.

- A selection tool with more flexibility - Enatom's web application requires a more sophisticated tool for selecting points that can freely select any shape of points within the point cloud, such as a lasso-tool.

- Label tool to add labels - Since Enatom's target user has anatomy students, they would greatly benefit from an option to add custom labels to the point clouds and/or classify selected point clouds in different categories. In our implementation, we were able to achieve the creation of scribble-based and text-based annotations that offer more flexibility.

- Anonymisation of cadavers through blurring - Since the point clouds on Enatom are actually cadavers that have been photographed for anatomical studies, it is important to blur out parts that can lead to personnel identification - such as tattoos or distinct facial features. The current application has no option to anonymize specific sections or to anonymize the pointcloud as a whole. Currently, the only option in the tool created by us is to change the colour of selected pointcloud, either by using a pre-defined gradient scheme or by giving the whole selection one colour.

3.1.3 Existing Potree usage on Enatom

Enatom uses the Potree point cloud renderer which currently utilizes only a few of the several capabilities that a Potree-based web platform inherently offers. This is because Enatom is working on creating tools more specific to their application with restricted access control to the original point cloud.

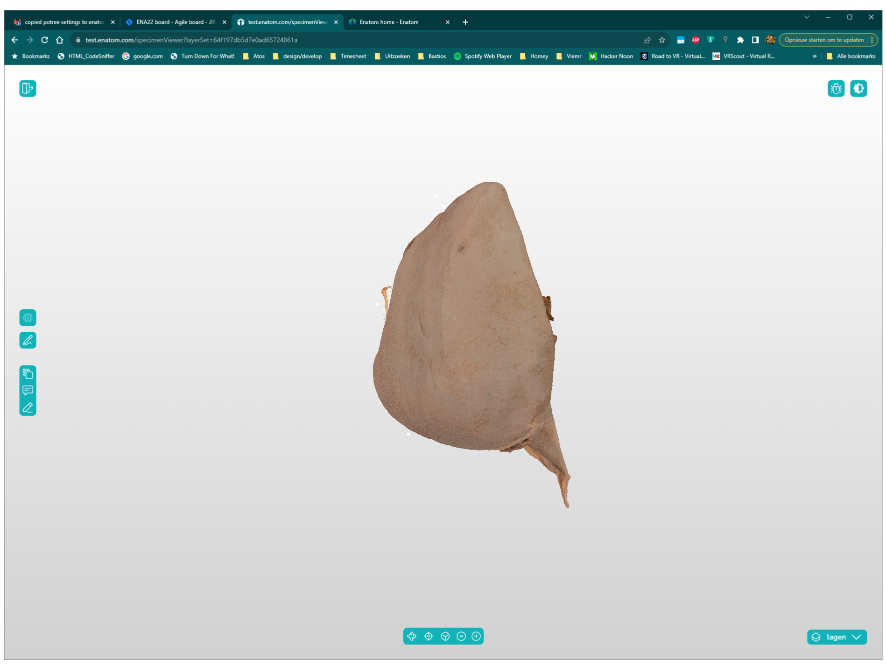


Figure 5: Current interface of Enatom web application

Under the Synthesis project, we worked on creating tools tailored to Enatom's specifications. (However, it is important to note that the access control aspect was not specifically considered during this process, primarily for two reasons: 1) This project is an open-source exercise, the results of which are to be made public, and 2) In the limited time we had, it was decided to focus on development of the tools themselves)

Figure 5 is a demo of Enatom's web-based Potree platform. A pointcloud model of a human lever.

Overview of technologies being applied

- The current web application gives 3D visualization through a point cloud created by photographs of anatomical objects
- PotreeConverter generates an octree LOD structure for streaming and real-time rendering of massive point clouds. The point cloud can then be viewed on web browsers using Potree. There are significant changes from version 1.7 to version 2.0 of PotreeConverter, where instead of tens of thousands of mini files of octree only three files are produced. However, Enatom prefers to use the old format of generating octree LOD because it gives flexibility to know to which node is the point of interest in the point cloud.
- Enatom had initially developed their application on Unity (a popular game development platform) and now migrating to the web.
- existing practice for anonymization - preprocessing of point cloud on CloudCompare to color (in grey) the points in those areas of the 3D model which can lead to identification.
- for the model itself - Enatom did consider having a digital model (mesh) as the end product but found that dense point clouds created by photogrammetry produce more realistic visualization

3.2 Selection tool and layers control system

The selection of the subset of the point cloud data is one of the bases when analyzing that data [Burgess et al., 2015]. Utilizing a selection tool permits the user to classify the point cloud data into distinct categories. By combining the sequenced analysis or statistics operation, more intrinsic information of the point cloud data could be extracted. Therefore, enabling the selection and extraction of different body tissues directly from point cloud data presents a significant pedagogical advantage for medical education. This capability grants medical students a unique opportunity to explore anatomical structures in an immersive way. Through the precise discrimination and selection of diverse body tissues, students can acquire a virtual hands-on experience that nurtures a profound and nuanced comprehension of human anatomy.

Moreover, beyond the selection tool, a layers control system is indispensable for managing user-selected point cloud data subsets. When coupled with this system, users can label various anatomical structures post-selection. This not only augments the learning experience but, when paired with a data export function, can supply valuable classified data for deep learning endeavors in medicine.

However, these capabilities present challenges. Among them are the efficient selection of point cloud data in 3D space through 2D interactions, the design of the layer control system from scratch, and the determination of the optimal data format for exporting point cloud data subsets.

In the subsequent section, we first elucidate the methodology underpinning the selection tool, followed by a proposed layer control system mechanism. Concludingly, the data export procedure will be detailed. An illustrative overview of this selection process can be seen in figure 6.

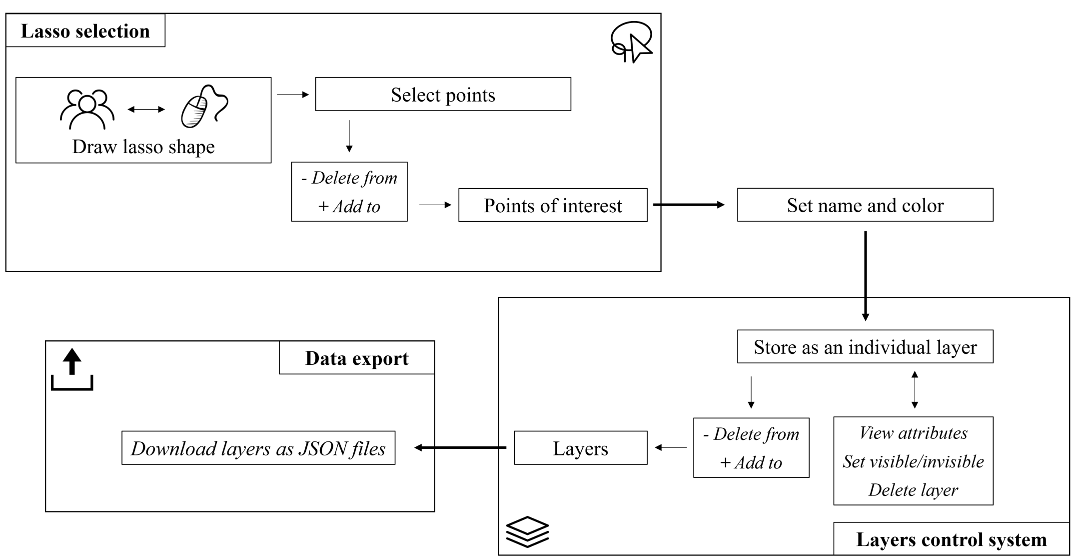


Figure 6: Overview of the workflow of selection tool and layers control system

3.2.1 Development framework

Numerous point cloud selection or labeling tools, such as RViz Cloud Annotation Tool [RMonica, 2023] and AnnotationTools [Xiong, 2023], are primarily constructed using C++. While these tools harness the computational power intrinsic to C++, they often grapple with user accessibility challenges and encounter difficulties in establishing smooth interactions with Potree. As a result, such tools, despite their potential efficiencies, have been set aside.

Considering an integrative approach, one might contemplate creating a bridge between a C++ programme and Potree. However, Potree's inherent Octree data structure and its Level of Detail mechanism pose significant barriers to this union. The complexities of ensuring full point cloud data transmission, especially when managing massive data, become computationally intensive, making this fusion approach less viable.

Given these circumstances, and notwithstanding the scarcity of existing web-based tools developed with JavaScript, the decision to adopt JavaScript in crafting a tool from scratch is well-founded. Tools developed in this paradigm not only ensure enhanced usability (as it is web-based) but also guarantee seamless integration with Potree.

3.2.2 Selection tool

In the primary phase of the selection tool, users are empowered to create a polygon for point delineation. Drawing insights from the field of art design, we have incorporated the lasso tool mechanism to facilitate this polygon generation (Fig. 7). To facilitate this, a plane is established parallel to the camera plane and positioned one unit forward, serving as the drawing canvas. The lasso shape is traced by monitoring the user's mouse trajectory once the lasso selection function is activated, as illustrated in Fig. 8.

Figure 7: The lasso selection tool
Picture source: www.adobe.com

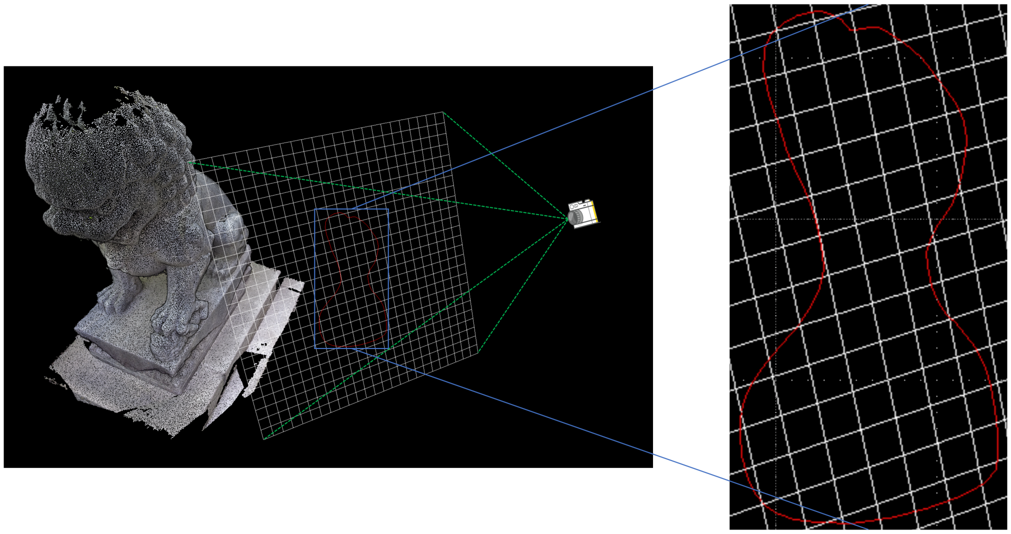


Figure 8: The canvas plane (white) and the lasso shape (red)

In the second phase, the aim is to identify points within the lasso's boundary from the camera's position. Given the depth variations between the point cloud data and the canvas plane in 3D space, the ray-casting algorithm is employed. This algorithm determines the nearest object intersecting the ray originating from its origin. This mechanism is depicted in Fig. 9.

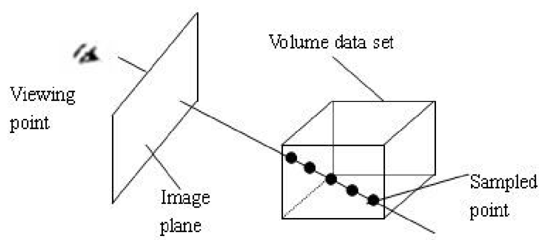


Figure 9: Overview of the workflow of selection tool and layers control system [Appa, 2015]

In our research, we have implemented a method for dynamic grid ray-casting using the Raycaster class in Three.js. In this approach, an invisible density grid overlaying on the canvas plane, tailoring to the dimensions of a user-defined lasso shape, is generated first. The grid cell size, set as a hyper-parameter with a default value of 10 screen pixels, is adjustable to meet different needs. Then, the grid points outside its boundary are excluded. The rays are projected from the camera viewpoint, directed towards the remaining grid points, and aimed at the target point cloud object. This process is illustrated in Figure. 10, where the grid on the canvas plane is depicted in white, and the density grid remains invisible. The green dot represents the camera's origin, the yellow dots mark the grid points on the invisible grid, and the yellow lines indicate the paths of the rays.

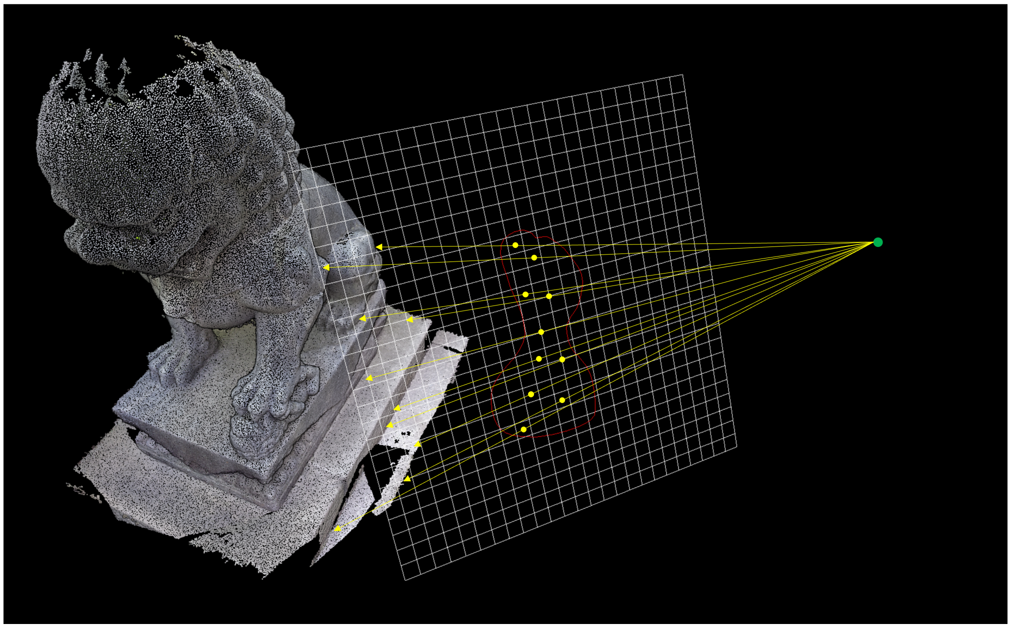


Figure 10: The grid ray-casting

In traditional ray-casting systems, each ray is limited to detecting points that are precisely aligned with its trajectory. This limitation can result in the exclusion of points within the lasso-defined area, particularly when the grid's resolution is low. To surmount this challenge, our approach incorporates a dynamic buffer system into the ray-casting process. Each ray is augmented with a square buffer, capable of capturing points within a specified proximity. As illustrated in Figure 11, this buffer dynamically adjusts in size relative to the grid cell size, thereby ensuring comprehensive detection of points within the lasso shape area, regardless of the grid's resolution.

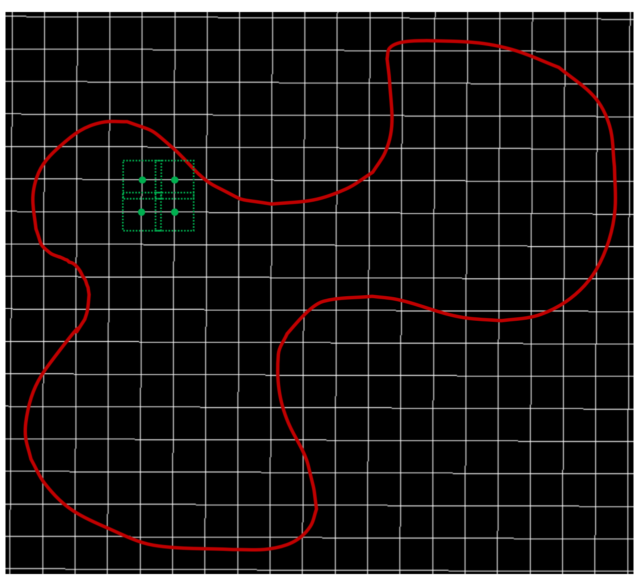
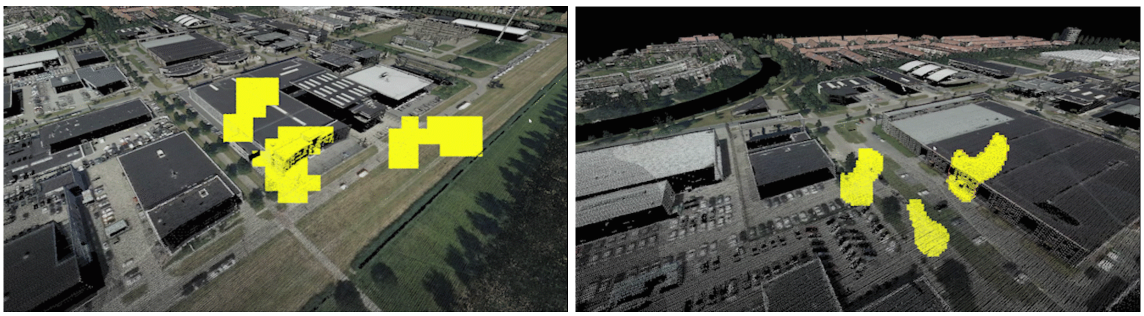


Figure 11: The illustration of ray's buffer (The white grid is the density grid, the green dots are the direction points of the rays, and the green dash squares are the buffers)

The decision to make the grid cell size a hyper-parameter was driven by the need to balance computational load with accuracy in the selection area. A smaller grid cell size increases the number of grid points and consequently, the number of rays, leading to higher computational demands. Conversely, a larger grid cell size could result in pronounced serrated edges, as the larger buffer might extend beyond the lasso boundary, creating irregularities due to its square shape (Fig.). The user has the flexibility to select an appropriate grid cell size based on the size of their point cloud data and the desired precision of their selection.



(a) 50 screen pixels *(b) 10 screen pixels*

Figure 12: The effect of different grid cell size

Furthermore, to facilitate modifications in the selected point set (either adding or removing points), we have provided a user-configurable flag (add/remove). If set to "remove", the selected points will be subtracted from the existing point set, provided they are part of it. Conversely, if set to "add", the selected points will be incorporated into the existing set, with duplicate points being eliminated.

3.2.3 Layers control system

Managing the selected point sets is crucial, especially in medical visualization where students might need to differentiate between multiple anatomical structures. Inspired by some Geo-information system platforms, such as QGIS and Google Earth Engine, a layer control system is identified as an optimal solution (Fig.13). Such systems offer the capability to view attributes like point count and color, adjust visibility, and delete specific layers. Implementing a layer control system can thus significantly enhance operational efficiency and precision. We therefore propose the integration of a comprehensive layer control system for refined point sets management.

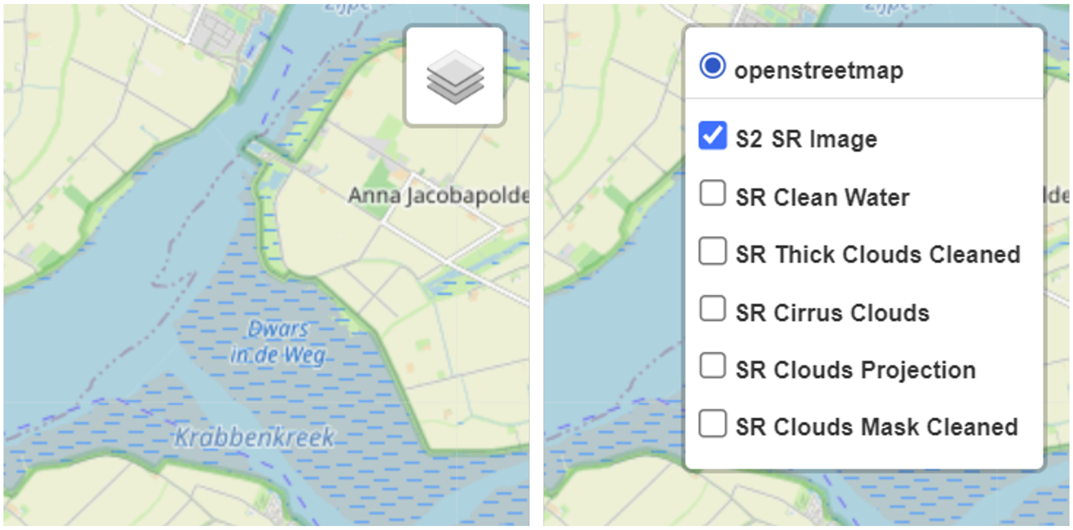


Figure 13: The Layers control system of Google Earth Engine

3.2.4 Data export

The ability to export data is indispensable. By facilitating the export of selected point sets, users can seamlessly access, share, and integrate specific anatomical structures for collaborative research, advanced analysis, or assimilation into diverse platforms. We have elected to utilize the plain JSON format for this purpose due to its widespread acceptance as a data interchange medium, which ensures the seamless incorporation of the exported datasets across a plethora of applications, amplifying collaborative potential. What is more, with the ascendance of JSON as the predominant standard for web APIs, it is much more suitable for our web-based tool.

3.3 Annotation tool

This subsection will outline the foundational principles that have guided the development of an advanced "scribble" annotation tool, emphasizing its role in fulfilling the pedagogical need for interactive and multi-dimensional learning strategies in medical education. Subsequent subsections will detail the cognitive importance of note-taking and annotation, the limitations of the current annotation system employed by Enatom, and explore homography as a pivotal methode to enhance the annotation tool's functionality.

3.3.1 Importance of Note-taking and Annotation in Learning

Note-taking and annotation are active learning strategies that have been shown to significantly affect learning outcomes. It engages learners in processing information deeply by summarizing, paraphrasing, organizing, and synthesizing; activities that are linked to better comprehension and memory retention [Kiewra, 1989]. For medical students, whose curriculum includes the mastery of large and complex subject matter, the ability to annotate directly on learning materials - much like using a highlighter in a textbook — serves as a crucial cognitive tool. It allows them to interact with the content more profoundly than through passive 'reading' alone. The active engagement facilitated by 'scribble' annotations can help in forming robust mental representations of anatomical structures, essential for clinical reasoning and problem-solving in medical practice.

3.3.2 Limitations of the Current Annotation System

The existing annotation system within the Enatom application offers a basic level of functionality by allowing users to overlay 2D annotations on 3D models. However, it falls short in supporting the iterative and multi-perspective exploration that is characteristic of medical study. Once an annotation is made, it becomes fixed to a specific orientation, hindering the user's ability to interact with the model from various angles while maintaining access to the annotated information. This constraint not only disrupts the continuity of study but also restricts the educational experience to a single perspective, counter to the three-dimensional nature of the subject matter.

The current system's limitation extends to emerging technologies such as virtual reality (VR), which is becoming an increasingly important tool in medical education for its immersive capabilities and its potential to enhance spatial reasoning and learning.

3.3.3 Scribble based annotation tool

To address these limitations, this project proposes the development of an advanced annotation tool that maintains the educational benefits of 'scribble' annotations while offering dynamic interaction with 3D models. The enhanced tool will allow annotations to be visible and editable from any orientation, thereby not restricting the medical student to the initial viewpoint.

The methodological approach to achieving this involves employing homography, a concept from the field of computer vision that refers to the transformation of images between different planes, to rotate and position the annotation plane congruently with the 3D anatomical model. By doing so, the annotations will effectively maintain their spatial relevance, allowing students to seamlessly integrate notes and visual references as they examine different facets of the model.

We plan to incorporate functionality that will enable users to toggle/remove the annotations, adjust their depth relative to the point cloud while keeping the same perceived dimensions from the original perspective, and reorient the camera to the original perspective of annotation creation when necessary. These features will be integrated within a user-friendly interface, ensuring that the tool enhances the learning process.

3.3.4 Typing based annotation tool

Three.js is a popular JavaScript library used for creating 3D graphic and interactive 3D applications in web browsers. It provides high-level abstraction for working with WebGL, allowing developers to create 3D scenes and animations with ease. Three.js is often used for creating web-based 3D experiences, such as games, simulations, architectural visualization, and more by creating 3D objects, scenes, cameras, lighting, and other elements. Potree is a specialized library that extends the capabilities of Three.js to handle the unique requirements of massive point cloud data in web application.

The basic idea is to develop a simple text editor using three.js objects, which can be used to take notes on Enatom's webpage, and this notes has to allow user to add text to it. Given an input point where user wants this annotation tool to be, the annotation canvas pops up allowing him to make notes.

The basic features include user modifiable title, notes, editor's background colour, font's colour and size, editor's width and height. To add to it most importantly, editor should have the functionality of movement in 3D space, because if user thinks plane is obscuring points behind, or if they just prefer to have text plane to be on one side of the screen, they should be able to do it.

To meet this end, three.js's *PlaneGeometry* and *TextGeometry* objects are used to create meshes, and let us call them *PlaneMesh* and *TextMeshs*. *PlaneMesh* serves as the background for the *TextMeshs*. Any modification user wants to make to the annotation tool, in the background any or both of meshes will be triggered and modified. For instance, changing the background colour of the annotation tool, only modifies *PlaneMesh's* colour.

Strangely, three.js TextGeometry object doesn't provide the feature of multi-line text. Meaning, for each line that we see on the screen it has to be loaded as different TextMesh object. Based on the background size (width), font size and text input, we have to calculate number of words each line can take to not overflow beyond the boundaries of the canvas.

Firstly, for each mesh that assists text annotation tool, their respective centers of translation and rotation are to be known, and these are defalut centers of mesh objects created in three.js:

- The background plane, *PlaneMesh*, has its centre at where diagonals meet.
- Each *TextMesh*, centre is at left side bottom corner of that mesh.

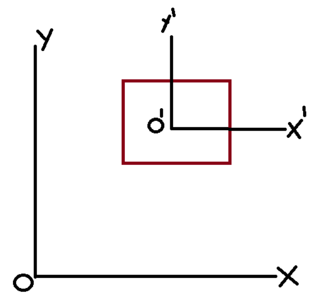


Figure 14: Depiction of center of rotation (O') of a text editor in 2D

All of this adds extra layer of complexity, because now for any 3D transformation that has to be applied on the text annotation editor in the viewer, all of the TextMeshes including the background PlaneMesh has to be transformed.

So, for each TextMesh, i.e., each line that is created has to be offsetted in both the directions parallel to the background plane's adjacent sides. The TextMesh has the offset of $-widthOfPlane/2$ and $heightOfPlane/2 - numOfLines * fontsize - lineSpacing$.

Now, considering the whole text editor as a single entity or group, we have to define the the center of translation (with respect to which point should the translations occur) and center of rotation (with respect to which point and axis should rotations occur).

- For translation, center is the given point itself.
- For rotation, the center is considered as the center of PlaneMesh. It makes more sense because the user can expect and is intuitive to see plane rotating with center of rotation on the plane itself (Figure 14).

3.4 Anonymization

In ensuring the protection of the individual's privacy and confidentiality, it is crucial to anonymize identifiable attributes, including facial characteristics, fingerprints, and any discernible tattoos.

The anonymization technique should take certain things into consideration. A facial point cloud will both have a structure and colour aspects that make it identifiable. Distortions in both, or either, could result in different extends of anonymization. For areas containing tattoos or fingerprints, an abstraction in colour may be sufficient to avoid identifiability.

There will be individuals authorized to access the entire point cloud, in contrast to the typical Enatom application user, for whom knowledge of the identity of the deceased individual whose body contributed to the scan is unnecessary. A toggle option for the anonymization method should therefore eventually be implemented.

While successfully developing such a tool is critical, it is imperative to acknowledge the associated risks of the anonymization process, particularly the potential for circumvention leading to the rediscovery of personal information. Employing encryption techniques for the point cloud (or only for specific section of the point cloud) could ensure protection.

Firstly, the legal motivation behind anonymization will be explored. The sections following this propose potential methods to anonymize regions of the anatomical point cloud model of Enatom.

3.4.1 Legal motivation

When working with personal data in the EU, the General Data Protection Regulation (GDPR) applies. This directive enhances an individual's control and rights over their personal data, including the transfer of such data.

A special category of personal data in the GDPR is biometric data. The GDPR defines biometric data as:

"Personal data resulting from specific technical processing relating to the physical, physiological or behavioural characteristics of a natural person, which allow or confirm the unique identification of that natural person, such as facial images or dactyloscopic data;" [European parliament, 2016].

Hence, an argument can be made that facial point cloud models should be classified as biometric data. The directive does not specify exactly what facial features need to be removed to comply, thus leaves room for interpretation.

Enatom is involved with handling personal data; however, the data in question pertains to a deceased individual. According to Recital 27 of the GDPR, the regulations do not directly apply to deceased persons, although member states may establish additional guidelines for processing the personal data of deceased individuals [European parliament, 2016]. Relevant implications arise only when the personal data of the deceased individual directly impacts that of a living individual. Enatom works together with the University Medical Centre Groningen. The centre provides the cadaver that is used to complete the scan to produce the pointcloud. They use the bodies of people that have donated their body to science. They state on their website that if a person decides to donate their body to science via the UMCG, the individual has no say in what exactly their body is used for [UMCG, 2023]. They do provide a list of possible applications, which includes the digital model being developed by Enatom. The UMCG makes a couple statements on their website regarding the protection of personal data when donating your body to science :

"The UMCG shall exercise the utmost care to anonymize the images so that they cannot be traced back to a person."
"In addition, those who gain access to your body have a duty of confidentiality, in the unlikely event that your body and/or your data are recognized."

The UMCG promises to ensure the protection of the personal data. Consequently, although the data utilized by Enatom may not fall under the purview of the GDPR directive, there remains a necessity to anonymize and safeguard the identities of the deceased individuals who have donated their bodies to science, facilitating the creation of scans for Enatom.

To what extend the data needs to be anonymized is up for interpretation. In spatial data processing, data protection officials agree on that if a spatial object is represented in an abstract manner only or if the object is obfuscated, no interest of individuals are violated [M. Kada, 2009].Thus in the case of following such an approach, methods to enhance abstraction levels of the pointclouds or methods that hide or blur certain parts of the pointcloud can be considered. There is ofcourse a difference between spatial datasets and (possible more sensitive) anatomical datasets, which could imply that protection methods for spatial datasets will not be sufficient for anatomical datasets.

3.4.2 Downsampling

A sparse pointcloud will have less detail than a dense one. Downsampling the pointcloud then could be a possible solution to reduce identifiability. While the random removal of points may create holes, the structural integrity of the face remains largely unaffected, as the points themselves remain stationary without any displacement.

Specific parts of the face can be removed to erase identifiable features. An anonymization method for facial scans in healthcare for the purpose of developing a customized face-mask, anonymizes by removing certain parts of the face via an automatic anonymization algorithm [Rustici, 2020]. They specifically keep parts of the face that are needed to produce the face-mask, and reduce the quality of the pointcloud as much as possible to comply to the GDPR (as they work with living individuals). Figure 15 shows the results of this algorithm. Given the objective of achieving this level of anonymization in the point cloud, this approach stands as a viable choice, as evidenced by the findings of this study demonstrating the successful anonymization of the facial model through this method.

For the purposes of this project, this would not be the best option. This method would result in holes in the point cloud, which would be distracting and aesthetically undesirable. Enatom aims to develop a method that minimizes the loss of detail of the point cloud, while this technique almost maximizes this loss.

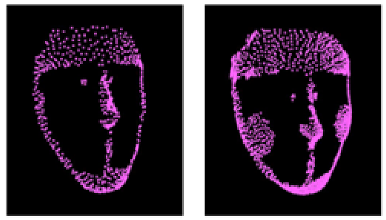


Figure 15: The anonymization results of a pointcloud developed with the purpose to produce a medical facial mask [Rustici, 2020]. The pointcloud has a lot of holes in it, but the method is succesfull in anonymizing.

Downsampling could also be used as a pre-processing step in the creation of a mesh to replace or hid sensitive areas, as described in the following section (3.4.3).

3.4.3 Mesh mask

Enatom opts out of utilizing a mesh representation over a point cloud due to the loss of intricate details incurred during the involved processing steps. For the anonymization process, we could take advantage of this. Features that need to be less recognizable can be meshed can be made from the pointcloud separately, but loaded with a toggle tool in potree. A mesh produced from specifically a dense pointcloud could still have too much detail and thus be easily identifiable, especially when the colourscheme of the pointcloud is applied to it. The structure of the face could be very similar. To distort and simplify this structure, a downsampled pointcloud could be used to produce the mesh. Replacing a pointcloud with a mesh from a downsampled pointcloud simplifies the shape of the object and results in an increase of the level of abstraction.

The client mentioned an attempt at anonymization by changing the colour of the selected area to grey by using external software. While effective, this method is not optimal from an aesthetic standpoint, as it would conspicuously contrast with the overall point cloud. Replacing the selected area by a mesh with a single colour could result in the same issue. Thus, this method should also consider the colourscheme of the mesh mask.

3.4.4 Colour distortion

As mentioned before, Enatom's attempt at anonymizing was by setting the colour of the points to grey. Colouring a face grey cause a great loss of anatomical accuracy. Improvements can be made by at least partially preserving the colour of the section in question. This could be achieved by taking an average colour and using this instead of plain grey. A colourpalette could be defined using the colours of the points in the section and randomly assigning each point a colour from this palette.

More sophisticated approaches would preserve the location of the colour within the section. Blurring filters such as a 3D Gaussian filter applied to colour value would take the RGB value of nearby points into account. A problem with this method is that Gaussian blur usually is applied to images, in which pixels have a specific pattern over which the filter can convolve. Point clouds do not have this structure. Such blurring should then rely on finding nearest neighbours.

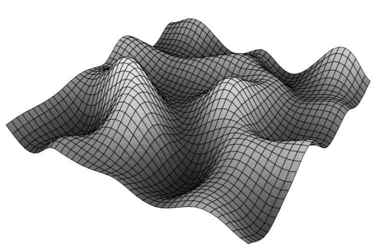
Another approach can be to give a section of neighbouring points one colour by taking the average of this section, creating a patchy pattern within it.

3.4.5 Deformation

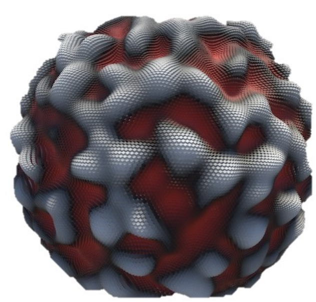
To achieve a less recognizable point cloud, one possible approach could involve distorting the structure of the point cloud. By adding spatial noise to the points, the structure of the point cloud will change. A proposed method to do this is described by Florent Poux [Poux, 2023]. This method uses unsupervised planar segmentation to section the point cloud, which obtains categories without the need of annotation. The algorithm needs to produce features that are compact enough to form distinct clusters. These clusters then are instantiated and used to differentiate between segments. To each of these segments, deformable noise with a specific pattern is attached. This noise pattern is randomly chosen per segment. It is ensured that points in segments do not collide. The result is the same point cloud, but with each section being a bit warped in shape.

The problem this project tries to solve consists of anonymizing sections defined beforehand. Planar segmentation does not seem to have any need for this if we already select a small region. A simpler solution can then be used to deform the point cloud, and that is by attaching noise to the spatial location of each point. This will distort the point cloud and should reduce identifiability.

Potential types of deforming noise can be a random offset, or offset following a specific pattern, as Poux describes in his method. Potential options for offset following a specific pattern are Perlin noise and its variant Simplex noise. These are techniques employed in computer graphics that introduce a naturalistic randomness and spatial distortion to objects. These methods add complexity and organic variation, which are often used to simulate natural phenomena, such as terrains or patterns in a visually appealing manner. Perlin noise involves the interpolation of pseudo-random gradient vectors, which produces a smooth, continuous gradient field. Simplex noise was developed as an improvement for Perlin noise, lowering the computational complexity, but maintaining similar characteristics to Perlin noise.



(a) Terrain mesh produced using Perlin noise [scratchapixel.com, 2023]



(b) Simplex noise on a 3D sphere [Flick, 2021]

Figure 16: Perlin noise and simplex noise visualisations

3.5 Encryption

Enatom uses pointclouds to visualize data that contains sensitive information. Therefore, the safety of the transfer of the data itself from the company to the user should be ensured. This can be achieved by encrypting the pointcloud. Encryption methods for 3D data involve techniques to secure the data through the process of transforming it into an unreadable format using cryptographic algorithms. These methods aim to protect sensitive information from unauthorized access or modification.

A review of three-dimensional objects encryption algorithms mentions the current techniques used to encrypt pointclouds and other 3D objects [Manal Mizher, 2023]. These techniques utilize the concept of chaos and tend to be based on cat mapping. Mathematically, chaos is a seemingly random motion in deterministic dynamical systems. Chaos, even though in its nature is unpredictable and random, has regularity. A cat map is a chaotic mapping method for repeated folding and stretching transformation in a finite region.

Chaochuan Jia et al. propose two encryption methods for pointclouds that utilize chaotic cat mapping [C. Jia, 2019]. The first method is a permutation using 2D cat map. Since a point in 3D space has three coordinates, three permutation matrices are randomly generated by 2D cat mapping. These are then used to shuffle each coordinate. The other method proposed in this paper is based on 3D cat map. A 3D point's position is altered by a reversible transformation matrix.

Xin jin et al. propose two schemes of chaotic mapping to encrypt 3D pointclouds [Jin et al., 2016]. The first uses 3 random sequences that are generated by the logistic cat mapping. Then each random vector is sorted to randomly shuffle the coordinates of the 3D pointclouds. The second scheme utilizes a random 3x3 invertible rotation matrix and a 3x1 translation vector which are generated by the logistic mapping. Each 3D point is projected to another location using these.

Xin Yang's method utilizes deformed fringe and off-axis digital Fresnel hologram. The z-coordinate of the 3D object is phase coded and subsequently transformed to deformed fringe. A conventional off-axis digital Fresnel holography method was utilized to encrypt and record the information, using the deformed fringe and gray image of the 3D object [Yang and Zhang, 2016].

One of the main challenges involved with the encryption process if maintaining the quality. Extra concern has to be taken into account when working with Virtual Reality (VR). VR enables the user to emerse themselves in the simulated environment and interact with the 3D model.

4. Results

This chapter discusses the tools and methods the team was able to implement within the time frame of this project. Firstly, the selection tool is discussed. After that, there are two options for annotation; one scribble-based and one text-based. Lastly, anonymization methods are implemented.

4.1 Selection

4.1.1 Operation panel

Leveraging the foundational structure of the Potree sidebar, we have crafted a dedicated operation panel for selection. This interface empowers users to activate the selection feature, adjust pertinent parameters, and export the chosen datasets. Figure 17 delineates this enhanced operation panel.

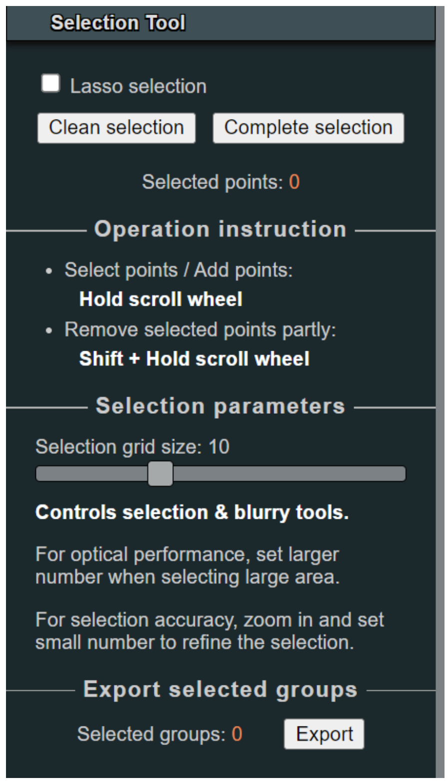


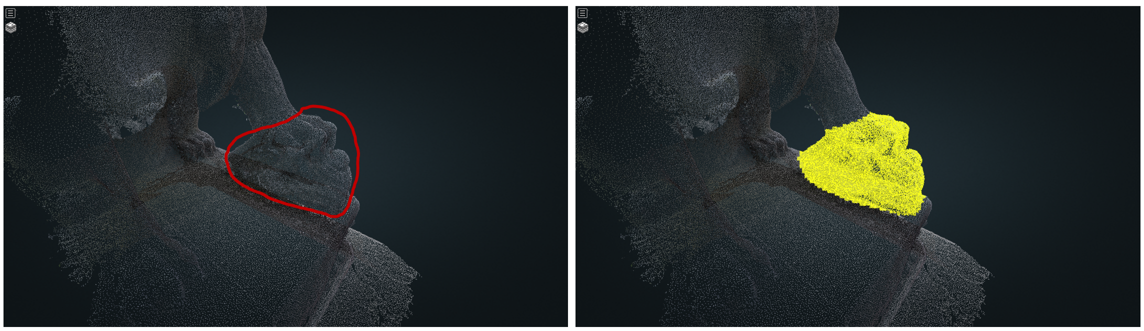
Figure 17: The operation panel

4.1.2 Selection tool

Upon determining the appropriate size for the invisible grid used in lasso selection, the user can initiate the creation of a lasso line by employing the mouse's middle button (as depicted in Figure 18a). The completion of the lasso shape, marked by the release of the mouse button, simultaneously activates the dynamic grid ray-casting process. During this phase, points falling within each ray's buffer zone are selected and highlighted, as shown in Figure 18b.

It is important to note that Potree utilizes the "Level of Detail" technique, a strategy designed to enhance performance and facilitate real-time rendering, and the point cloud data are organized within an octree structure [Schütz et al., 2020]. During the lasso selection, not all levels of the octree are accessed, thus, only the points that are rendered and hence visible are eligible for selection.

Furthermore, in the default configuration of Three.js's Raycasting, there is no depth limitation imposed on the rays. Consequently, each ray possesses the capability to select multiple points along its path. This characteristic ensures that all visible points within the confines of the lasso shape are selected, encompassing points at varying depths as perceived from the camera's perspective.



(a) The lasso line *(b) The selected points*

Figure 18: The lasso selection tool

4.1.3 Layers control system

Drawing inspiration from the layers control system employed by Google Earth Engine, as depicted in Figure 13, we have designed a layers control panel, as illustrated in Figure 19.

Name	Points Nums	Color	Visible	Delete
Trees	2116		☑	Delete
Buildings	9673		☐	Delete
Roads	1486		☑	Delete
Water bodies	4829		☐	Delete

Figure 19: The layers control system panel

Upon completing their selection for a given group, users can elect to save the point set as a distinct layer within the layers control system by clicking the **Complete selection** button. Subsequent customization options, including layer naming and color designation, will be presented to the user through an ensuing pop-up window, shown in Figure 20.

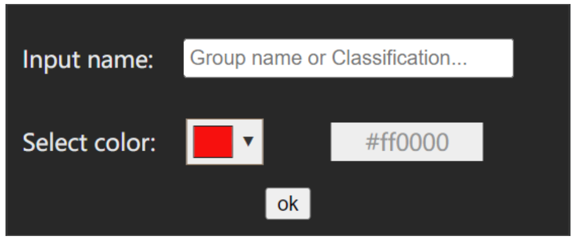


Figure 20: Set group name and custom color

A comprehensive view of the layers control system functionalities is delineated in Figure 21.

Figure 21: Overview of Layers control system

4.1.4 Data export

Upon completing the selection process and necessitating the export of delineated or classified point sets, users can conveniently initiate this by selecting the **Export** button situated at the operation panel's base. A subsequent pop-up window, depicted in Figure 22. Once the **Download** button is activated, all point sets are transmuted into a plain JSON format and autonomously initiated for download by the browser. The resultant JSON file structure is presented in Figure 23.

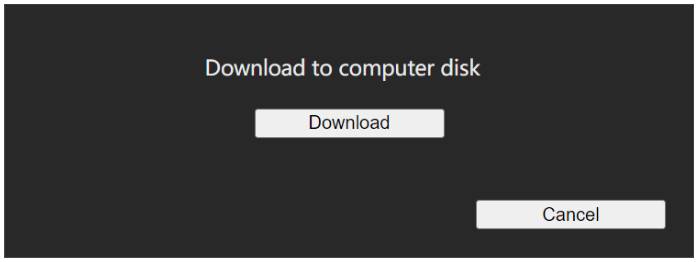


Figure 22: The export window


```json
{
    "asd": {
        "metadata": {
            "version": 4.5,
            "type": "Object",
            "generator": "Object3D.toJSON"
        },
        "geometries": [
            {
                "uuid": "1755DD3F-9F1D-48F5-8207-6A6BD7567CF2",
                "type": "BufferGeometry",
                "data": {
                    "attributes": {
                        "position": {
                            "itemSize": 3,
                            "type": "Float32Array",
                            "array": [
                                0.45975935459136963,
                                -1.4654335975646973,
                                5.485752105712891,
                                0.4767593443393707,
                                -1.4734336137771606,
                                5.51575231552124,
                                0.47478705644607544,
                                -1.472406029701233,
                                5.519765853881836,
                                0.483759343624115,
                                -1.475433588027954,
                                5.535752296447754,
                                0.4807593524456024,
                                -1.4744336605072021,
                                5.535766124725342,
                                0.4787593483924866,
                                -1.4724336862564087,
                                5.551752090454102,
                                0.46975934505462646,
                                -1.4704335927963257,
                                5.558752059936523,
                                0.4897593557834625,
                                -1.475433588027954,
                                5.579751968383789,
```


Figure 23: The downloaded JSON file

4.2 Annotation

4.2.1 Scribble annotation

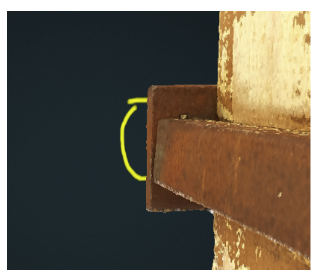
We present a "scribble"-based annotation system embedded within the Potree viewer interface. This system affords medical students a seamless transition from 2D drawing interactions to 3D spatial annotations, supporting the creation, management, and contextual retrieval of user-generated markings designed to enhance the learning experience in the study of complex anatomical structures. It facilitates the emulation of traditional annotation methods, such as highlighting and notation, within a virtual spatial context.

Within a dedicated "Annotations" panel users can activate the annotation mode via a button interface. This action overlays a 2D drawing canvas atop the 3D point cloud scene, capturing the user's "scribbles" in real-time.

The drawing interface allows for free-form input, simulating the experience of using a highlighter or pencil to underscore features of interest. The "scribble" analogy is deliberately employed to invoke the familiar experience of manual note-taking and annotation within a digital medium.

Upon completion of the 2D "scribble", the canvas is transposed into the three-dimensional space directly before the camera's current viewpoint. This ensures the spatial relevance of the "scribble" to the user's perspective at the time of creation, anchoring the annotations within the 3D environment, as shown in Figure 24.

(a) Initial "scribble" annotation capturing a 2D free-form circle around a screw in 3D space, viewed from the perspective of the user at the moment of creation.



(b) View of the annotated point cloud rotated 180 degrees, illustrating the persistence and spatial anchoring of the "scribble" annotation, which now appears partially obscured by the point cloud structure.

(c) Alternative angle showcasing the "scribble" annotation in relation to the screw, demonstrating the distance maintained between the annotation layer and the point cloud to ensure clarity and legibility from varying perspectives.

Figure 24: A 'scribble' annotation in 3D space.

Annotations maintain a consistent size relative to the user's view, with the ability to be repositioned along the camera's original view axis to achieve the desired proximity to the anatomic structures, as shown in Figure 25. They persist within the scene and are indexed within the side panel, allowing for immediate spatial repositioning, removal, or revisitation, as shown in Figure 26.

Figure 25: "Scribble" annotations in 3D space at various depths (left), and the convergence of these annotations into a single apparent "Scribble" when viewed from the original annotation perspective (right).

Another feature of the annotation list is the option to "jump" the camera to the specific orientation and location from which a particular "scribble" was drawn. This facilitates a context-rich review of the annotations, as each "scribble" recalls the precise anatomical viewpoint originally intended by the annotator.

The integration of ray casting techniques facilitated the editing of 3D "scribble" annotations from various perspectives. Additionally, experiments were conducted on spatial repositioning of annotations using transformation controls. Despite the technical feasibility of these methods, they were found to contribute minimally to the utility of the "scribble" tool. The experimentation revealed that these editing techniques were time-consuming and did not align with the users' inclination for quick and effortless annotation. In the context of creating "scribbles", which are intended to be rapid and intuitive annotations, these methods introduced an unnecessary complexity. Consequently, the most efficient approach for editing in this scenario was determined to be the simple removal and re-creation of a "scribble". This method aligns with the tool's core objective of facilitating quick and easy annotations on medical point cloud data.

The integration of the "scribble" annotation tool within the point cloud viewer has demonstrated potential to significantly enhance user engagement and cognitive interaction with anatomical structures. This tool enables rapid annotation and the creation of spatial markers, similar to "sticky notes", facilitating the swift conveyance of thoughts and highlighting of specific areas on medical objects. Initial testing suggests that this capability can contribute to more efficient and comprehensive spatial understanding. As highlighted in section 3.3.1, the ability to mark, recall, and contextualize individual viewpoints results in a comprehensive spatial understanding that is critical in medical education. While the current findings are indicative of the tool's utility for quick idea expression and information retention in complex anatomical contexts, broader testing is recommended to further substantiate its effectiveness.

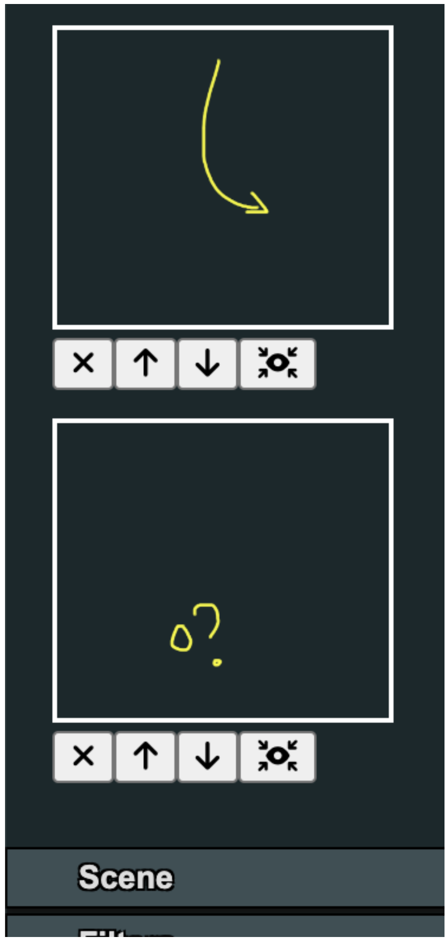


Figure 26: 'Annotation list.

4.2.2 Typing Based Annotation

A dedicated "Text Annotation Tool" panel is designed where users can interact with text annotation tool (Figure 27). User has to give the location's (X, Y, Z) where they want to have the text annotation plane. Along with it, the annotation's title and notes should be added, and then click save button. This creates a white background of size (100, 100) units, and text is added in black colour (Figure 28).

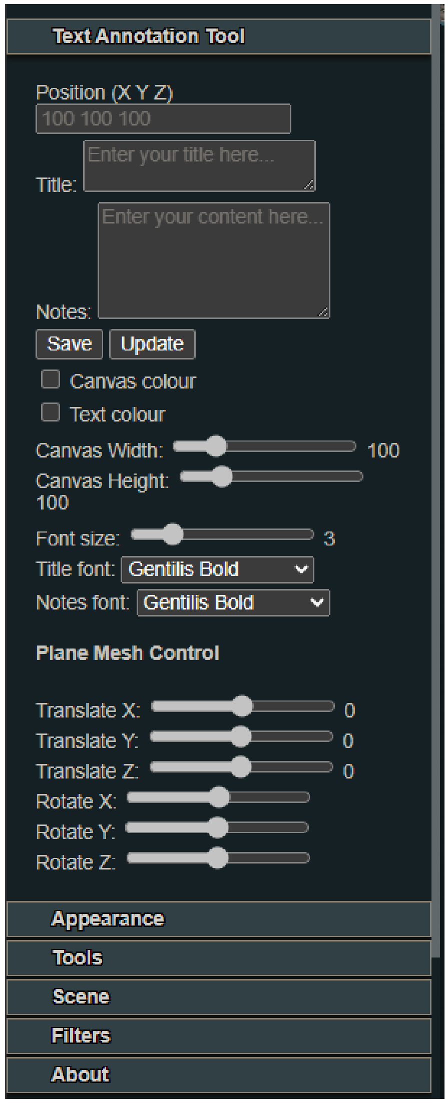


Figure 27: Text Annotation Tool user interface

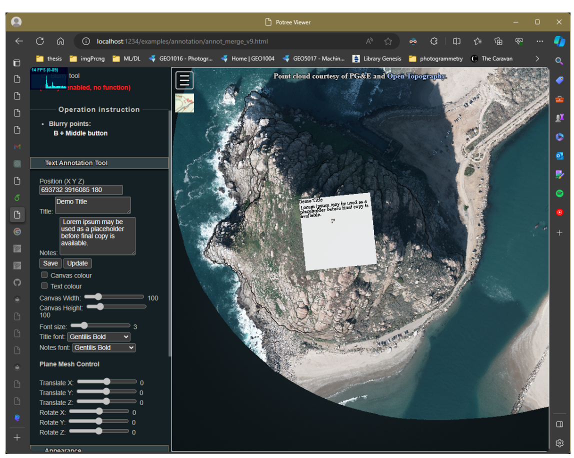
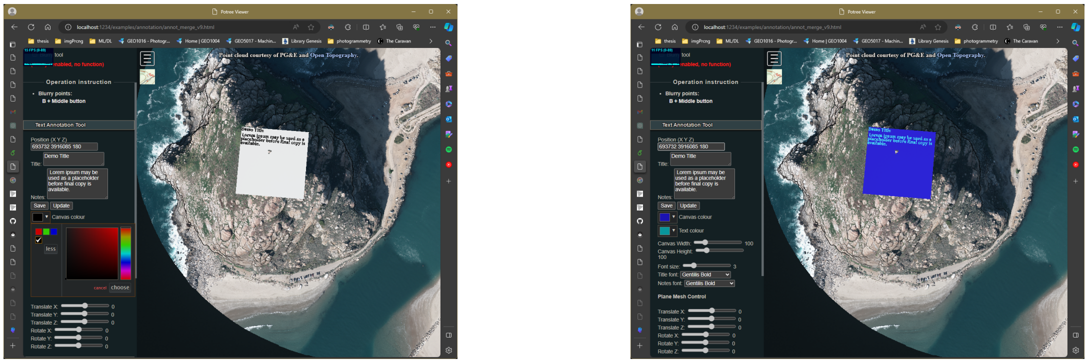


Figure 28: First annotation with basic white background and black font

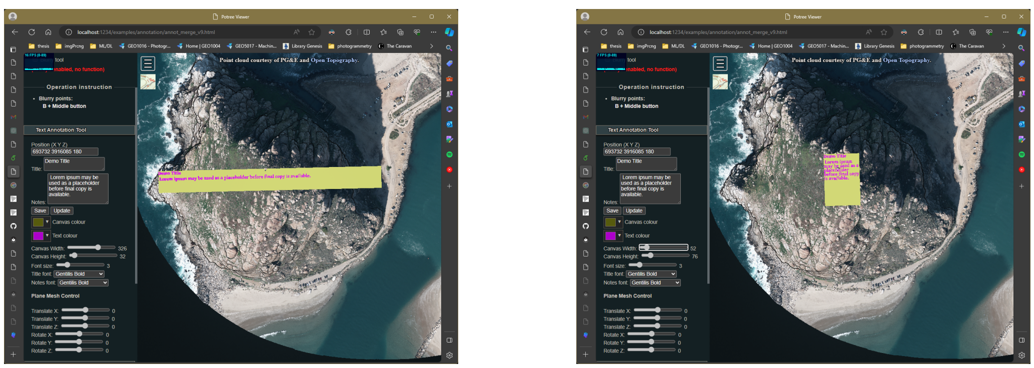
Once the basic annotation tool is ready, then the user can choose the colours for both background and the text (Figure 29).



(a) Choice of colours for canvas and text *(b) Colours applied to background and text*

Figure 29: Colour choice for text annotation

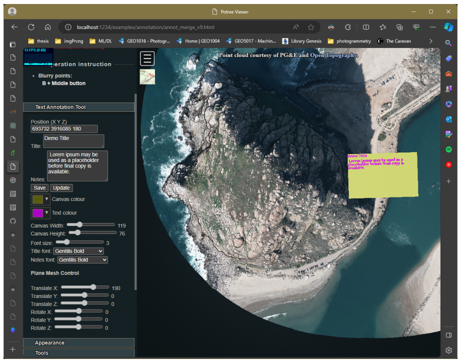
One of the important features is to let the shape of the annotation plane be defined by user input. If the amount of text is limited, then then it desirable to have a small annotation plane occupying less space in the 3D scene, and viceversa. Figure 30 gives an example of this.



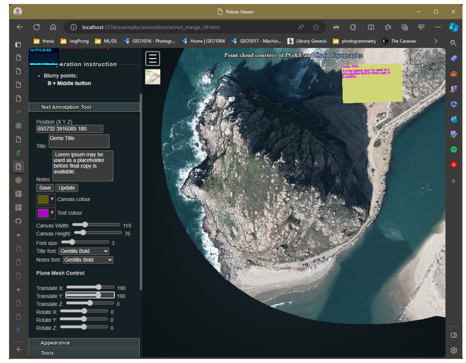
(a) Low width high length annotation plane *(b) Low width low length annotation plane*

Figure 30: Width and length modification of text annotation

When we have an editor which contains text on it, this occludes points behind it, which is an unintended and unpleasant consequence. To solve this issue, two straightforward apparent methods are either to make the background (Semi-)transparent, or to give the user flexibility regarding moving the plane within the 3D scene. We chose the latter approach because if the annotation is text heavy, then the problem still remains. Figures 31 and 32 show the possible movements that the user can apply to the annotation plane.

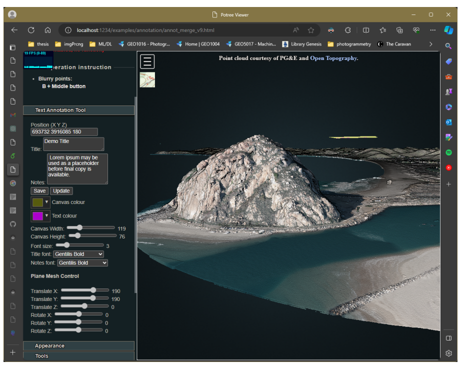


(a) Translation in x-direction

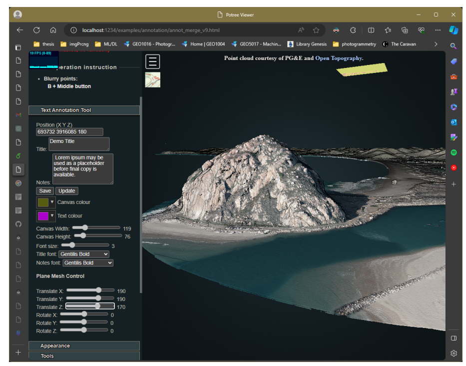


(b) Translation in x- and y-direction

Figure 31: Translation of annotation plane in xy-directions



(a) No translation in the z-direction



(b) Plane lifted in the z-direction

Figure 32: Translation of plane in the z-direction

The figure below (Figure 33) shows the combination of translations in all directions and rotation along the x-direction.

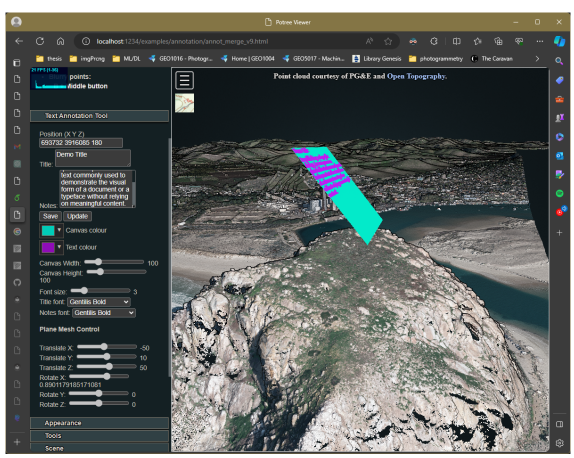


Figure 33: Translation along all directions and rotation along the x-axis

4.3 Anonymization

Within the selection tool, a blurring tool to decrease identifiability is developed. This tool has two aspects to it. The first one concerns the colour of the point cloud.

Colour of the selected region is replaced by another colour, randomly generated. Random values for r, g and b are given within the interval between 0 and 1. A problem with complete random colour generation is that it stands out a lot from the rest (Fig. 34a). Applying random colour generated within bounds shows less contrast (Fig. 34b).

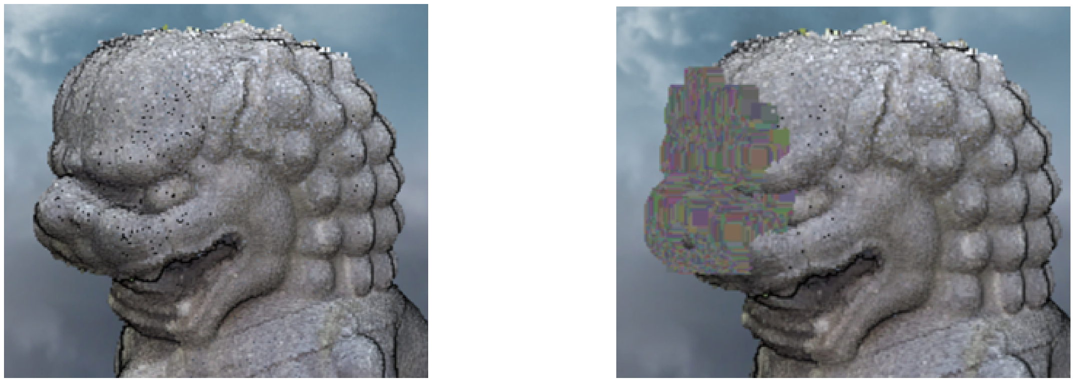
(a) Random colour (values between 0 and 1)

(b) Random colour within bounds (values between 0.3 and 0.5)

Figure 34: Blurring tool examples: colour distortion.

Presently, the colour pattern chosen is defined within the code. The selection tool does not give access to the colour of the points, but replaces it by one uniform color. Bounds for random colour selection are manually chosen, but should be able to be user-defined in future developments.

The second aspect concerns the structure of the point cloud. It causes spatial distortion as points within the selection obtain a random offset. Figure 35 shows the difference between the point cloud and the point cloud with the face blurred. The blurred region is slightly changed in position, and the colour is distorted.



(a) Normal point cloud *(b) Point cloud with colour and spatial distortion*

Figure 35: Blurring tool examples: spatial distortion

For full protection and anonymization, the whole face or area that needs to be secure can be removed from the point cloud and replaced by the mesh.

A problem with this method lays in when you want to see the whole pointcloud. The mask can be removed, but a hole in the pointcloud will be seen. A solution to this is to make a toggle tool that either shows the mesh, or it shows the pointcloud. Another option is to load the whole pointcloud underneath the mask.

A problem arises when the face of the pointcloud is not removed before visualizing in potree, and the mask is set at the same position as the pointcloud. Some of the points from the pointcloud can have a higher position than the mask. This is solved by altering the position of the mask a little so that it hoovers above the pointcloud. Figure 36 shows an example implementation of this method in Potree.

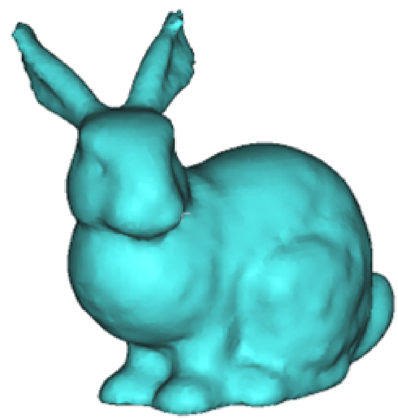
(a) Pointcloud and mesh mask both toggled on. *(b) Pointcloud toggled on, mesh mask toggled off* *(c) Pointcloud toggled off, mesh mask toggled on.*

Figure 36: Pointcloud of the bunny with a mesh mask to hide its face, visualized in Potree.

For this example implementation, external software (CloudCompare) was used to produce the mesh object, specifically the Poisson surface reconstruction plugin. Firstly, the point cloud object was downsampled to approximately 10 percent of the points to obtain later obtain a mesh that has more of a smoothed and simplified surface. Figure 37 shows the difference between a mesh from the full point cloud and one from a downsampled point cloud. The surface is smoother and has a loss of detail in the downsampled object. This abstraction in spatial structure can aid the reduction of identifiability.

CloudCompare was then used to cut out the piece that serves as a mask and that is removed from the point cloud. The adjusted pointcloud is then converted to the correct format using the Potree converter and loaded into potree. Ideally, this process should take place within the Potree application itself. The selection tool should select a region, the points within this region are downsampled, and this downsampled piece of point cloud is used to create a mesh that hoovers above the full point cloud.

(a) Mesh from the full point cloud



(b) Mesh from the downsampled pointcloud

Figure 37: Difference between a mesh from a full point cloud and a downsampled point cloud

The texture of the mesh then could be replaced by an image with a matching colourscheme of the point cloud to make it blend in more while still distorting the colours.

Instead of using a single colour to represent the area in need of protection, a colourscheme close to the point cloud can be used to allow the area to blend in. JavaScript allows you to wrap a self-defined texture around mesh. Figure 38 shows an example of this. An image of a rabbit's fur is wrapped around a mesh representation of a rabbit.

(a) Fur

(b) Grey

(c) Bunny meshes visualized in Potree; one with a texture applied (.jpg image of rabbit fur) and one with a gray colour

Figure 38: Two coloured meshes of a bunny

5. Conclusion

The aim of this project was “Exploring efficient methods of visualizing, annotating and interacting with the objects of human anatomy using its point cloud representation”.
Point clouds have been selected for visualization as they present the actual object in 3D with a high level of detail, similar to what one would observe in reality. Adding functionalities to 3D point clouds such as selection, classification, or annotation, can further enrich the user experience - in this case for anatomy students in their interaction with the 3D point clouds of anatomical subjects being studied.

With the aim to add functionalities to 3D point cloud on a web platform (Enatom’s platform in particular), we developed a Potree-based selection tool that extracts distinct sections from a point cloud, we also developed (partially) two possible solutions for annotation function within the web viewer platform, and we explored and implemented methods that anonymize sensitive information while losing as little anatomical information as possible.

The necessary background information on Potree, technicalities, and motivation for the development of the tools have been discussed in the report. While there were a number of possibilities and aspects we brainstormed on, the plan of action was dynamically adjusted to accommodate the realities of implementation.

5.1 Implementation and results in medical application

A lasso tool for Potree was developed using a ray casting algorithm that can export sections of the point cloud as a different las file. Within the selection tool, there are options to change the colour and structure of the section. The possibilities of adding a mesh mask have been explored.

Two annotation methods for making annotations that are affixed to the point cloud have been realized, one being scribble-based and the other being text-based.

The developed 3D ’scribble’ annotation tool represents an approach to interactive learning within the Potree viewer, offering medical students an intuitive mechanism for identifying and revisiting points of interest in complex anatomic point clouds. While the utility of perspective-based editing was less pronounced than expected, the overall functionality enriches the educational experience by blending traditional note-taking with 3D visualization technologies.

The anonymization process was not fully developed, as the priority of this project lay in the selection and annotation tools. Nonetheless, anonymization methods have been explored and implemented in Potree.

Integration of the developed tools was also not implemented entirely. The eventual steps for using of these tools could be combining selection tool features with anonymisation tool on the backend (for Enatom to select surfaces to be anonymised) and with the annotation tools in the front end (for students to be able to make a selection and add notes). A more detailed explanation of future directions for the developments made under this project has been provided under section 6. Future study.

5.2 Broader Implications and Future Prospects

This project represents a significant intersection between geomatics and medical education, highlighting the versatile application of spatial data analysis in various domains. By leveraging the capabilities of Potree, a WebGL-based point cloud renderer, for the detailed visualization of human anatomy, we have not only addressed a need in medical education but also expanded the potential applications of geomatics techniques.

5.2.1 The use of point clouds

Traditionally, geomatics has been associated with earth sciences, focusing on spatial data related to geography, earth environments, and urban planning. This project takes a pivotal step in applying point cloud technology, a staple in geomatics, to the field of medicine. The dense point cloud scans used to create detailed anatomical structures demonstrate that spatial data can be effectively utilized beyond its conventional scope.

5.2.2 The selection tool

The selection tool developed for precise region selection within the point cloud is not limited to medical applications. In the Geomatics field, this tool can be instrumental for tasks requiring high precision, such as delineating land parcels, environmental features, or urban infrastructures within large spatial datasets. The accuracy and ease of use provided by this tool exemplify how geomatics tools can be tailored to meet specific needs in different fields.

What is more, the selection tool, encompassing lasso selection, layers control, and data export functionalities, could serves as an alternative method for 3D geo-data labeling in preparing training datasets for deep learning applications. Its web-based nature and integration with Potree make it particularly efficient for handling large point cloud datasets, offering researchers a streamlined approach to point cloud data processing.

5.2.3 3D annotation in spatial analysis

The project's contribution to 3D annotation systems marks a substantial advancement in how spatial data can be documented and analyzed. In geomatics, such enhanced annotation capabilities can facilitate more effective communication among researchers, planners, and decision-makers, particularly in collaborative projects or complex spatial analyses. As demonstrated in Figure 37, users can label different elements within a point cloud, such as distinguishing buildings from vegetation, thereby facilitating more accurate and efficient data analysis in geomatics projects.

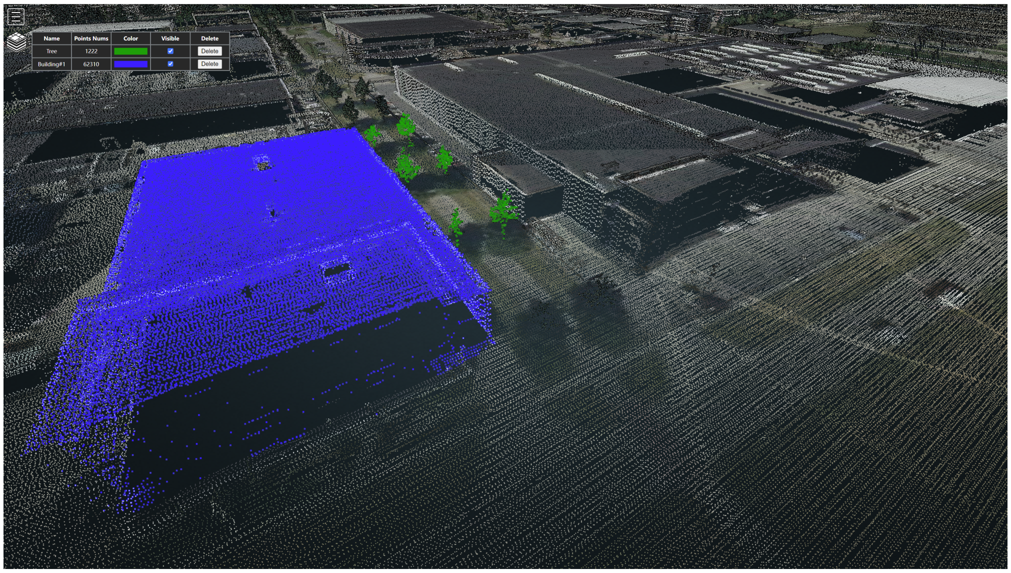


Figure 39: Label point cloud from AHN

5.2.4 Anonymization techniques in spatial data

The methods developed for anonymizing sensitive areas within point clouds have significant implications for both medical and geomatics fields. In geomatics, this addresses a growing concern over privacy and security of spatial data, especially when dealing with urban mapping or sensitive environmental areas. These techniques ensure that while the integrity and utility of the data are maintained, personal or sensitive information is protected.

5.2.5 Future directions and broader impacts

The knowledge and tools developed in this project have implications far beyond their immediate application. They set a precedent for how geomatics methodologies can be adapted to diverse fields, encouraging innovation and interdisciplinary collaboration. Future studies can explore further integration of these tools in other areas of geomatics, such as environmental monitoring, heritage conservation, and urban development, thereby broadening the impact and utility of spatial data analysis.

By extending geomatics principles to medical education through this project, we have opened new pathways for the application of spatial data technology. This not only enriches the field of geomatics but also paves the way for its application in various other disciplines, demonstrating the universality and adaptability of geomatics techniques.

6. Future study

The synthesis project was an 8 weeks' exercise wherein we, a group of 5 students, tried to understand the needs of the company Enatom and develop tools to help improve their platform. While the first few weeks were focused on understanding the technologies being used and the gaps to be filled, we attempted to utilize the remaining weeks for developing the three tools. However, there are a number of aspects that can be looked into in the future, especially if someone decides to take up similar endeavours.

The selection process is done by creating a separate point cloud besides the original one. A connection between the two would be desirable. The new point cloud has the same coordinates as the original point cloud, thus this information could be used to make this connection. Space filling curves could be used to create a unique index. Through indexing, the option to add attributes (intelligence) to points in the point cloud could be developed.

The extent of anonymization has a level of subjectivity to it. Methods have been developed, but not applied to a medical point cloud. Thus, the level of anonymization needed has to be determined. The methods discussed need to be subjected to testing to see if they acquire the desired level of anonymity.

Methods to anonymize have been described in this report, but not fully implemented. A fully functioning tool within the Potree environment still has to be developed. Colour distortion via random colour has been shown, but needs to be automated to extract the bounds of random colour generation from the selected region in the point cloud. Replacing the section by a mesh was described, but it could be desirable to develop this as a tool within Potree. Encryption methods have to be applied to ensure protection of sensitive information.

The annotation tools are currently a rudimentary version of what we had in mind as there are a number of ideas that we were unable to implement but can be looked into in the future.

Ray-casting could be used to decide the point of annotation attachment by the user. Alternatively, the annotation tools could be linked with current selection tool that would allow the users to select a number of points and add annotation/classification to those. This user generated classification might even pave the path for automated classification through machine learning algorithms which currently doesn't have enough training data to allow for ML classification in anatomy.

The annotated objects (text with plane or the scribbled drawing) could be positioned in a way that it does not occlude the points of original point cloud. This means the text plane and/or the drawing plane could be aligned with the best fitting plane of selected points.

For the list in the user interface of the scribble-based annotations, the user experience can be improved by not only displaying the annotation itself but also where the annotation is positioned on the point cloud.

One of the limitations of the current version of the text-based annotation tool is that it can handle all the translations in any combination, and also all the rotations individually with any permutation of translations, but the combination of any two rotations do not work. This is because, as discussed in Methodology, all the text lines of the text annotation tool are individual meshes. Add the centers of each mesh are not aligned and have different offsets in X, Y, Z directions. Developing a robust transformation matrix for the next version of the tool should solve this problem. Further, to track the movement of annotation plane, an arrow joining its initial position to its current position will also help.

References

[Appa, 2015] Appa, S. (2015). Ray casting for 3d rendering – a review.

[Burgess et al., 2015] Burgess, R., Falcão, A. J., Fernandes, T., Ribeiro, R. A., Gomes, M., Krone-Martins, A., and de Almeida, A. M. (2015). Selection of large-scale 3d point cloud data using gesture recognition. In *Technological Innovation for Cloud-Based Engineering Systems: 6th IFIP WG 5.5/SOCOLNET Doctoral Conference on Computing, Electrical and Industrial Systems, DoCEIS 2015, Costa de Caparica, Portugal, April 13-15, 2015, Proceedings 6*, pages 188–195. Springer.

[C. Jia, 2019] C. Jia, T. Yang, C. W. B. F. F. H. (2019). Encryption of 3d point cloud using chaotic cat mapping. *3D Display Research Center, Kwangwoon University and Springer-Verlag GmbH Germany, part of Springer Nature 2019*.

[Dardavesis et al., 2023] Dardavesis, I., Verbree, E., and Rafiee, A. (2023). Indoor localisation and location tracking in indoor facilities based on lidar point clouds and images of the ceilings. *AGILE: GIScience Series*, 4:4.

[European parliament, 2016] European parliament, c. o. t. E. U. (2016). Regulations regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation). *Official Journal of the European Union*.

[Flick, 2021] Flick, J. (2021). Simplex noise.

[Jin et al., 2016] Jin, X., Wu, Z., Song, C., Zhang, C., and Li, X. (2016). 3d point cloud encryption through chaotic mapping. In Chen, E., Gong, Y., and Tie, Y., editors, *Advances in Multimedia Information Processing - PCM 2016*, pages 119–129, Cham. Springer International Publishing.

[Kiewra, 1989] Kiewra, K. A. (1989). A review of note-taking: The encoding-storage paradigm and beyond. *Educational Psychology Review*, 1:147–172.

[Li et al., 2020] Li, Y., Ma, L., Zhong, Z., Liu, F., Chapman, M. A., Cao, D., and Li, J. (2020). Deep learning for lidar point clouds in autonomous driving: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 32(8):3412–3432.

[M. Kada, 2009] M. Kada, M. Peter, D. F. O. S. C. H. (2009). Privacy-enabling abstraction and obfuscation techniques for 3d city model (position paper).

[Manal Mizher, 2023] Manal Mizher, Salam Hamdan, M. M. A. A. M. H. M. A. M. H. (2023). Three dimensional objects encryption algorithms: A review. *International Conference on Information Technology (ICIT)*.

[Oosterom et al., 2022] Oosterom, P., van Oosterom, S., Liu, H., Thompson, R., Meijers, M., and Verbree, E. (2022). Organizing and visualizing point clouds with continuous levels of detail. *ISPRS Journal of Photogrammetry and Remote Sensing*, 194:119–131.

[Poux, 2023] Poux, F. (2023). Pointcloud unsupervised segmentation.

[Qi et al., 2017] Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.

[RMonica, 2023] RMonica (2023). rviz_cloud_annotation. original-date: 2017-09-04T07:50:23Z.

[Rustici, 2020] Rustici, P. (2020). Anonymization of 3d face models for gdpr compliant outsourcing to 3rd party companies.

[Schuetz, 2016] Schuetz, M. (2016). Potree: Rendering large point clouds in web browsers.

[Schütz et al., 2020] Schütz, M., Ohrhallinger, S., and Wimmer, M. (2020). Fast out-of-core octree generation for massive point clouds. In *Computer Graphics Forum*, volume 39, pages 155–167. Wiley Online Library.

[scratchapixel.com, 2023] scratchapixel.com (2023). Perlin noise part 2.

[UMCG, 2023] UMCG, U. M. C. G. (2023). Umcg information on donating your body to science.

[Xiong, 2023] Xiong, l. (2023). Annotationtools. original-date: 2019-10-14T08:11:19Z.

[Yang and Zhang, 2016] Yang, X. and Zhang, H. (2016). Encryption of 3d point cloud object with deformed fringe. *Advances in Optical Technologies*, 2016:1–9.

[Zhao et al., 2021] Zhao, H., Jiang, L., Jia, J., Torr, P. H., and Koltun, V. (2021). Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268.

A. Project organisation

A.1 Team and responsibilities

The table below shows an overview of the teammembers.

Name	**Background**	**Email**
Susanne Epema	BSc in Liberal Arts & Sciences, major in mathematics and physics	s.a.epema@student.tudelft.nl
Sharath Chandra Madanu	BTech in Civil Engineering	s.c.madanu@student.tudelft.nl
Vidushi Bhatt	BPlan	v.bhatt-1@student.tudelft.nl
Gees Brouwer	Bsc in Computer Science and Engineering	g.d.brouwer@student.tudelft.nl
Qiwei Shen	Bsc in Land Resource Management	q.shen-2@student.tudelft.nl

The team has two supervisors from TU Delft, Edward Verbree and Peter van Oosterom. The main contact person from Enatom is Bastiaan Hofsteenge, but due to his absence in week 2, Lusanne Tehupuring was available instead.

The supervisors from Enatom are there to provide us with information of missing functionalities, and improving points for their application. They are there to help us understand their process and to answer questions regarding this process.

Our TU Delft supervisors are there to help us during weekly meetings with the challenges we face and can provide guidance throughout this project. Drafts of deliverables will be sent before the deadline, and will be reviewed by them, which we will use to improve the reports.

Organizational: Project manager	**Deliverables: Report manager**	**Technical: Technical managers**
Person in charge of all organizational components.	Person in charge when writing reports and when putting together the presentations.	Person in charge during the technical phase and the research phase.
Plan meetings, In charge of contact between team and supervisors, Meeting agenda, Lead meetings, Notes of meetings, Hold people accountable to their tasks, Make sure deadlines are met (deliverables and agreed upon deadlines within the team).	Needed components (template), Set up online environment to write deliverables in, Quality of reports, Quality of presentation, Define and divide tasks when writing deliverables, Make sure deadlines are met for deliverables.	Define and divide tasks during research phase, Software development framework, Define and divide tasks during development phase

Vidushi will be our project manager, Susanne will be our report manager, and Gees, Sharath & Qiwei will be our technical managers. The technical part of this project is divided into three parts. We have established 2 main tasks (the must-haves, annotation improvements and the lasso tool) and multiple additional tasks (the could-have and should-haves). Qiwei is in charge of the development of the selection and layers control system, and the implementation of the anonymization task.

Sharath and Gees lead the development and research on the annotation improvements.

The technical managers divide the technical tasks among all the group members, thus also the project and report managers. The software development is the main task of this project and is shared amongst all team members.

A.2 Meetings & communication

The group will meet with the supervisors Edward Verbee and Peter Van Oosterom every Monday afternoon at 15:30 to discuss progress and clarify doubts. The group will prepare an agenda beforehand to make use of this time efficiently.

The group will also meet every Monday (post-discussion with supervisors), Wednesday (post-lunch) and Friday (11:00) for a general meeting. For each of the four weekly discussions, there will be a brief agenda prepared in advance. The notes on agenda, status update and future task distribution will be taken by Vidushi and accessible to all through the Google Drive folder on Communications and Files.

Mon	**Tue**	**Wed**	**Thur**	**Fri**
11:30 Group meeting		Group meeting after lunch		11:00 Group meeting
15:30 Supervisor discussion with Edward and Peter				

A.3 Relevant courses

Our academic background in the MSc Geomatics program equips us with a robust understanding of various key principles that directly apply to the current project. The following courses have provided us with valuable theoretical foundations that will prove instrumental in enhancing Enatom's anatomic scans platform:

- GEO1001 - Sensing Technologies
  Link to Project: This course has equipped us with a comprehensive understanding of various sensing technologies, enabling us to better understand how the point clouds used in the project are captured.

- GEO1015 - Digital Terrain Modelling
  Link to Project: The principles of digital terrain modeling are directly applicable to the representation of anatomical structures in three dimensions. Our proficiency in this area will allow us to refine the visualization and interaction aspects of the scans, ensuring an informative and intuitive learning experience.

- GEO1006 - Geo Database Management Systems
  Link to Project: This course has equipped us with the knowledge and skills necessary to efficiently manage and manipulate large datasets. Given the density and complexity of point cloud scans, this expertise will be invaluable in optimizing data storage, retrieval, and processing for Enatom's platform. While we won't directly come in contact with the storage and streaming part of the project, this knowledge could be useful. In this course, we were introduced to Potree.

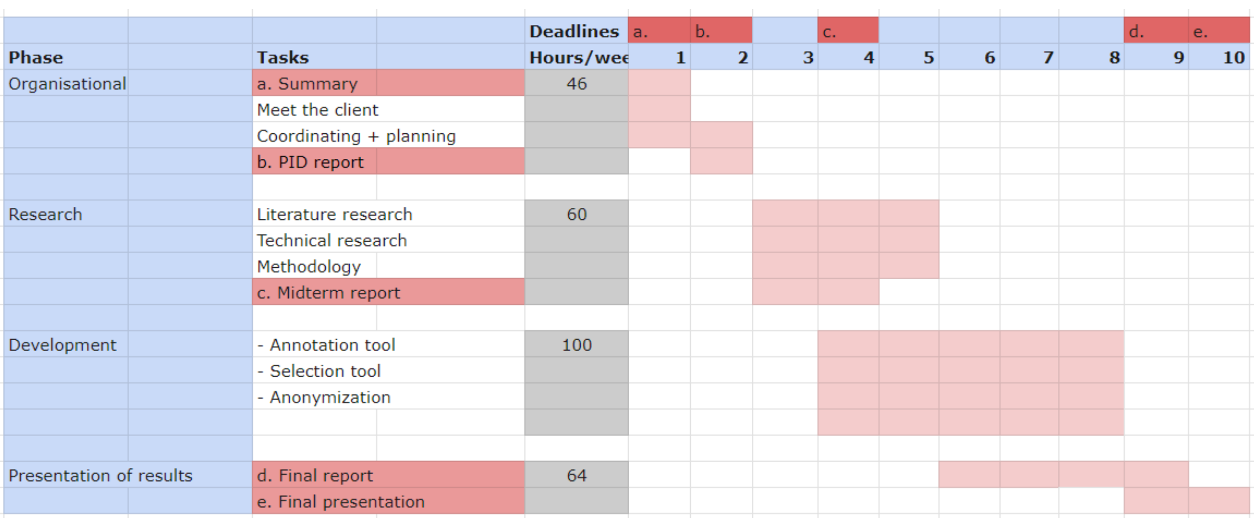
- GEO1016 - Photogrammetry and 3D Computer Vision
  Link to Project: This course has provided us with an understanding of techniques for deriving three-dimensional information from imagery. Leveraging this knowledge, we can optimize the processing pipeline for generating and visualizing high-quality anatomic scans.

- GEO1009 - Geo-information governance
  Link to Project: This course taught us the governing aspects of handling geodata, thus also concerning the protection of personal data. The anatomical point clouds used by Enatom can contain personal data that can not be shared with everyone.

- GEO1007 - Geoweb Technology
  Link to Project: This course discussed the principles and applications of generic web-services. Software tools for the visualization of 3D geo-data were taught. This course served as an introduction to HTML and JavaScript to some of the group members, which are used extensively in this project.

A.4 Gannt chart

Figure 40 shows the planning of this project in a Gannt chart.



		Deadlines	a.	b.		c.					d.	e.
Phase	**Tasks**	**Hours/wee**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Organisational	a. Summary	46										
	Meet the client											
	Coordinating + planning											
	b. PID report											
Research	Literature research	60										
	Technical research											
	Methodology											
	c. Midterm report											
Development	- Annotation tool	100										
	- Selection tool											
	- Anonymization											
Presentation of results	d. Final report	64										
	e. Final presentation											

Figure 40: Gannt diagram describing the planning of this project

B. Rich picture

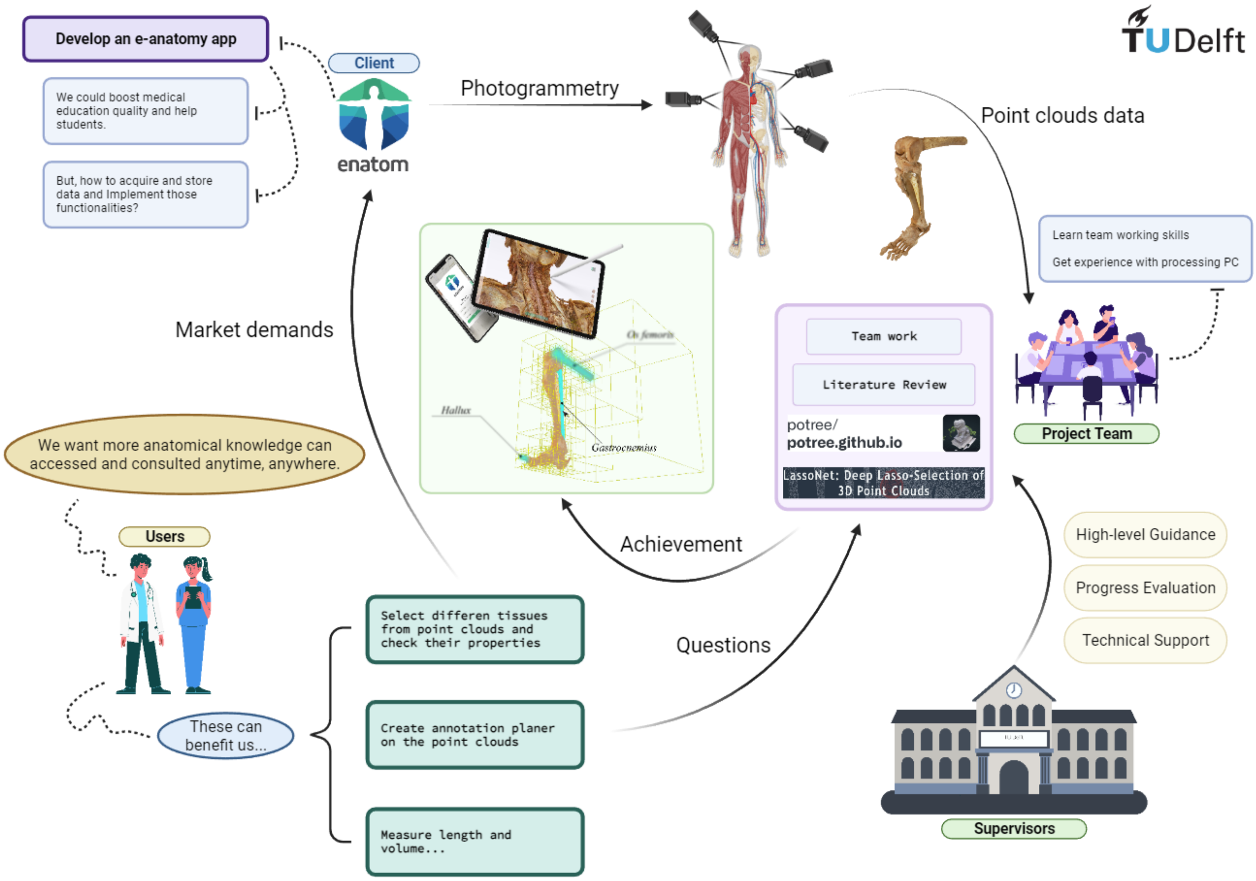


Figure 41: Rich picture of the project