Master of Computing CP5103 Project Report

Segmentation for Cochlear Implant Surgery

ZHANG YIXIN

Supervisor: Associate Professor LEOW Wee Kheng

2025-11-09

1 Introduction

Hearing loss represents a significant and growing global public health challenge. Currently, over 1.5 billion people live with some degree of hearing impairment, among whom hundreds of millions experience severe or profound hearing loss that substantially affects daily functioning[1]. For individuals with severe to complete deafness, cochlear implantation has been established as an effective intervention for hearing rehabilitation and quality-of-life improvement[2]–[4]. In recent years, both its indications and eligible candidate population have continued to expand.

The procedure involves mastoidectomy and posterior tympanotomy to access the inner ear, followed by electrode array insertion via the round window or a cochleostomy, with an emphasis on minimizing trauma to surrounding structures[5]. Given the close proximity of critical anatomical features such as the facial nerve, semicircular canals, and internal auditory canal, surgical precision is crucial to avoid complications and ensure optimal auditory outcomes[6].

The primary challenge in cochlear implantation stems from the abnormal or highly variable anatomy of the inner ear[7]–[9]. Structural differences in the cochlea and adjacent bone regions often complicate surgical access and electrode insertion, leading to greater variability in surgical process. To overcome these challenges, cochlear implant surgery simulation has become an effective tool for preoperative planning, surgical training, and risk assessment, enabling surgeons to visualize patient-specific anatomy and practice the procedure before actual surgery[10].

The overall simulation system consists of two main components: 3D model construction and surgery simulation. The first part focuses on reconstructing accurate three-dimensional models of the patient's temporal bone and inner ear structures from medical imaging data, typically CT scans. The second part builds on these models to simulate the surgical process, including drill path planning, electrode insertion, and visualization of potential risks. Since the simulation relies on the geometric and topological correctness of the input models, the fidelity of the reconstructed anatomy has a direct effect on how realistic and useful the simulation results will be.

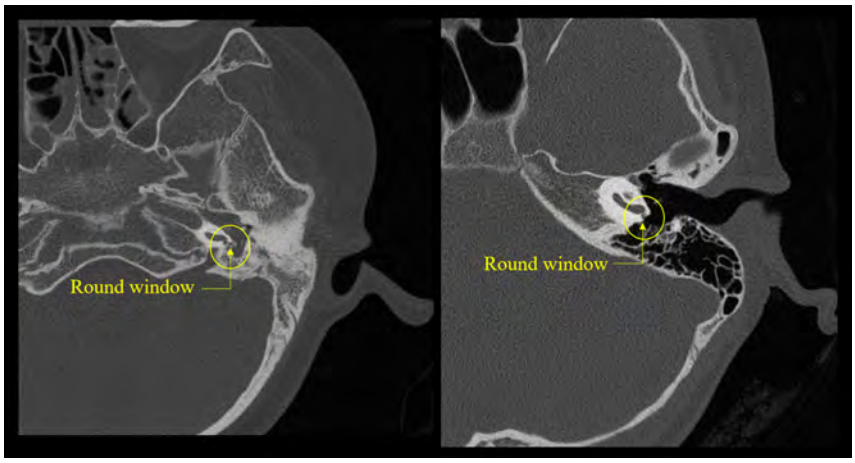

Figure 1: Comparison of temporal bone and cochlear anatomy between two patients. Significant structural differences can be observed between the two, highlighting the need for patient-specific surgical planning and simulation.

This project concentrates on the 3D model construction pipeline, which forms the foundation for all downstream simulation tasks. High-quality anatomical modeling of the cochlear and surrounding structures is critical for personalized surgical planning and outcome prediction. However, current segmentation tools face several challenges when applied to cochlea implant surgery planning. Many existing medical image processing applications struggle to capture fine anatomical details within the cochlea and facial recess, leading to incomplete or inaccurate 3D reconstructions. Some tools require extensive manual correction, which is time-consuming and operator-dependent. Moreover, temporal-bone anatomy shows large inter-patient variability in cochlear size and orientation, degree of ossification, and local bone morphology, which further complicates automated segmentation. As a result, deep-learning models trained on limited or curated datasets frequently underperform on routine clinical CTs, particularly when confronted with atypical anatomy, imaging artifacts, or lower resolution scans.

To address these shortcomings, this project aims to develop an improved segmentation and 3D reconstruction workflow that enhances both accuracy and efficiency. The proposed system integrates flexible DICOM image handling, interactive volume-of-interest (VOI) selection, and threshold-based segmentation with real-time visualization and 3D mesh generation. By streamlining the modeling process and improving anatomical precision, the tool enables the creation of high-fidelity, patient-specific 3D ear models suitable for surgical simulation and training.

2 Literature Review

Accurate segmentation of temporal-bone and cochlear structures from clinical CT is a necessary first step for patient-specific 3D reconstruction and any downstream surgical simulation for cochlear implantation[7], [11]. Clinical CT offers reliable bone contrast but limited resolution for the fine intracochlear anatomy (e.g. scalae, basilar membrane), and is subject to artifacts such as beam hardening and metal streaks[11]. As a result, a wide range of segmentation methods have been developed to convert CT volumes into usable surface models: simple intensity-based algorithms, deformable/contour-based techniques, and atlas or shape-prior approaches[12]. Each family of methods trades off ease of implementation, robustness to image quality, and the amount of required manual correction. The practical choice in clinical and research pipelines is often influenced by compatibility with existing toolkits (ITK/VTK), available computing resources, and the desired balance between automation and interpretability.

Intensity-based methods (thresholding, region growing, morphological filters) are the most commonly used primitives in temporal-bone work. They are straightforward to implement, fast to run, and easy to understand—qualities that make them attractive for initial bone segmentation and for integration into VTK/ITK pipelines. In regions where cortical bone contrasts sharply with surrounding tissue, thresholds and connected-component analysis produce reliable coarse masks. However, these methods perform poorly when structures have similar intensities (for example, fluid-filled scalae versus adjacent bone) and are highly sensitive to noise, partial-volume effects and CT artifacts and thus needs necessary manual editing to recover accurate intracochlear geometry.

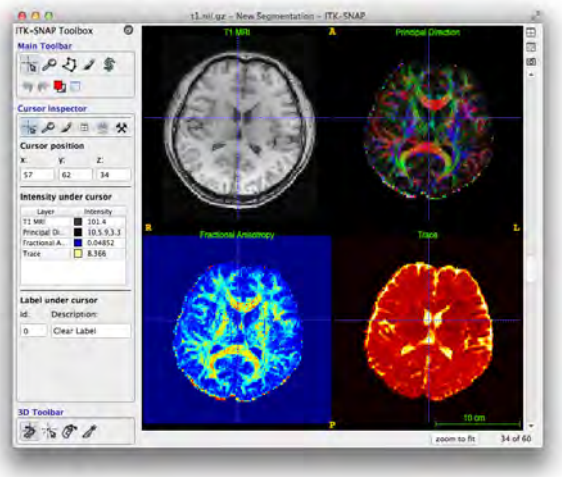

Figure 2: Interface of ITK-SNAP.

ITK-SNAP [13] is an open-source interactive tool designed for medical-image segmentation, widely used for both MRI and CT datasets. It provides region-growing and active-contour ("snake") algorithms combined with real-time 3D visualization, allowing users to seed and refine structures such as the cochlea and semicircular canals. Its intuitive interface and integration with ITK/VTK make it a convenient choice for research-grade temporal-bone segmentation. However, accuracy depends heavily on manual seed placement and intensity contrast, and fine intracochlear structures often require post-processing to achieve anatomical fidelity.

3D Slicer [14] is a comprehensive medical-image analysis platform that includes multiple segmentation modules—thresholding, region growing, morphological operations, and manual painting tools. It supports plug-ins and Python scripting, allowing customized workflows for cochlear and ossicular-chain modeling. The major strength of 3D Slicer lies in its flexible visualization and segmentation editor, enabling step-wise refinement of bone and soft-tissue boundaries. Nonetheless, it can be computationally heavy for large high-resolution temporal-bone datasets, and reproducibility depends on user experience and parameter tuning.

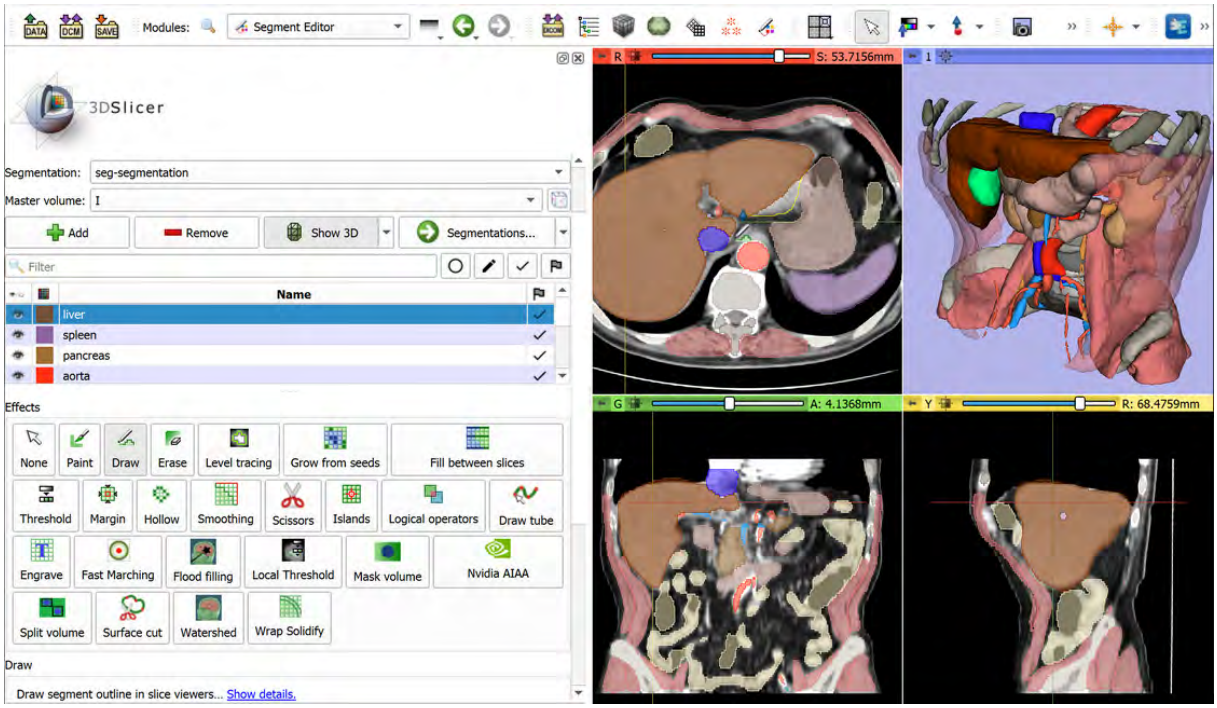

Figure 3: Interface of 3D Slicer.

Amira/Avizo (Thermo Fisher Scientific) are commercial 3D visualization and analysis suites frequently used in clinical and industrial CT research. Their segmentation tools combine intensity-based thresholding with interactive 3D surface editing and morphological filters, offering high rendering quality and efficient manipulation of complex structures. In cochlear-implant research, Amira/Avizo are valued for producing publication-grade surface models and for their support of precise measurement and landmarking. However, their proprietary nature and limited automation make them costly and less adaptable for customized segmentation pipelines compared to open-source alternatives.

Deformable-model segmentation has a strong foundation in early works such as Kass et al. [15]on Snakes and later Sethian [16] and Osher & Fedkiw [17] on Level-Set Methods. Modern implementations are available in frameworks such as ITK and SimpleITK, including Geodesic Active Contour and Chan–Vese algorithms. These methods evolve a surface or contour to minimize an energy functional that balances image gradients and smoothness constraints, enabling them to capture complex 3D structures such as the cochlear turns with higher topological flexibility than pure thresholding. However, practical use in temporal-bone segmentation is still limited because parameter tuning and initialization strongly affect convergence, and high-resolution CT data significantly increase computational demand. As a result, these methods are often integrated into semi-automatic

workflows, which is used to refine surfaces obtained from threshold-based masks rather than as fully autonomous segmenters.

Atlas-based segmentation is another method widely explored in anatomical regions such as the brain and later extended to the temporal bone and cochlea Open-source tools like ANTs (Advanced Normalization Tools) and Elastix [18], [19] provide robust registration frameworks that can align a labeled atlas to patient CT scans, transferring anatomical labels to guide segmentation. Statistical Shape Models (SSMs) [7], [12] implemented in libraries such as Scalismo and ShapeWorks further generalize this idea by learning shape variability from a population dataset and fitting the model to new images through optimization. These methods can greatly reduce manual effort and enforce anatomical plausibility even in low-contrast or partially missing regions. Nonetheless, they require well-curated training sets and perform poorly on scans with abnormal anatomy or surgical alterations, which limits their universality in clinical cochlear-implant planning.

In summary, existing CT segmentation tools for temporal-bone and cochlear modeling demonstrate complementary strengths but also notable limitations. Intensity-based methods remain fast and intuitive but struggle with fine intracochlear detail and often require extensive manual correction. Deformable and active-contour approaches improve boundary smoothness and adaptability to complex shapes, yet they are sensitive to initialization, parameter tuning, and computationally demanding for high-resolution data. Atlas-based and statistical shape models introduce valuable anatomical priors and improve consistency, but their reliance on representative training sets limits the generalizability to patients with atypical anatomy or surgical changes. Moreover, most current toolchains are monolithic and tightly integrated, which makes it difficult to adapt or extend individual modules for customized workflows. Consequently, there remains a need for a segmentation framework that is both accurate and modular, balancing automation with flexibility to accommodate diverse patient anatomies and research requirements.

3 Segmentation Tool

3.1 Introduction

This project presents a comprehensive medical image segmentation and visualization tool which is specifically designed to support cochlear implant surgical planning workflows. The system addresses critical segmentation challenges in medical image analysis by providing an integrated platform that combines DICOM image loading and visualization, precise region-of-interest selection, threshold-based segmentation, and 3D mesh generation capabilities, creates a seamless workflow that guides users from raw DICOM data through processed segmentation results to detailed 3D anatomical models.

The development of this tool originated from Panax, a medical visualization application originally implemented in C++. While Panax provided valuable functionality for medical image processing, its C++ codebase presented some maintenance challenges and deployment complexities. The migration to Python was motivated by several key factors: Python's extensive ecosystem of scientific computing libraries provides robust support for medical image processing without requiring low-level implementation details; Python's readability and simplicity significantly reduce the learning curve for new developers, facilitating team collaboration and long-term maintenance; Python's cross-platform compatibility simplifies deployment across different operating systems and clinical environments.

With key design principles of modularity, extensibility, and user experience, the system adopts a Model-View-Controller (MVC) architectural pattern, separating data management, visualization logic, and user interface components. This modular design not only improves code organization but also facilitates future enhancements.

Our implementation was developed using Python 3.10. Table 1 shows the Python packages and corresponding minimum versions required for our experiments. The system was tested with these versions and is compatible with newer versions as specified in the requirements.

Table 1: Software Dependencies

Package	**Version**
Python	3.11
PyQt5	$\geq$ 5.15.0
VTK	$\geq$ 9.0.0
NumPy	$\geq$ 1.20.0
pydicom	$\geq$ 2.0.0
Pillow	$\geq$ 8.0.0
trimesh	$\geq$ 3.9.0

3.2 Core feature

3.2.1 DICOM Image Loading and Visualization

The system provides robust support for standard, uncompressed DICOM formats using VTK's native vtkDICOMImageReader. It automatically processes DICOM directory structures, identifies individual slices within a series, and organizes them into a coherent 3D volume, forming the foundation for subsequent segmentation and visualization tasks.

However, a significant limitation of native vtkDICOMImageReader is its inability to handle compressed DICOM formats, such as JPEG 2000 Lossless Only. This format is commonly used in modern imaging systems to reduce storage requirements while maintaining diagnostic quality. In this project, all cochlear CT datasets were stored in this compressed format, making direct loading incompatible with the default reader. To address this issue while retaining the advantages of the VTK pipeline, an auxiliary Python pre-processing module was developed using the pydicom library. This module automatically detects and decompresses JPEG 2000–encoded DICOM files into a VTK-compatible format prior to loading. The uncompressed images are then passed to vtkDICOMImageReader, preserving its robust metadata handling and smooth integration with downstream visualization and segmentation components.

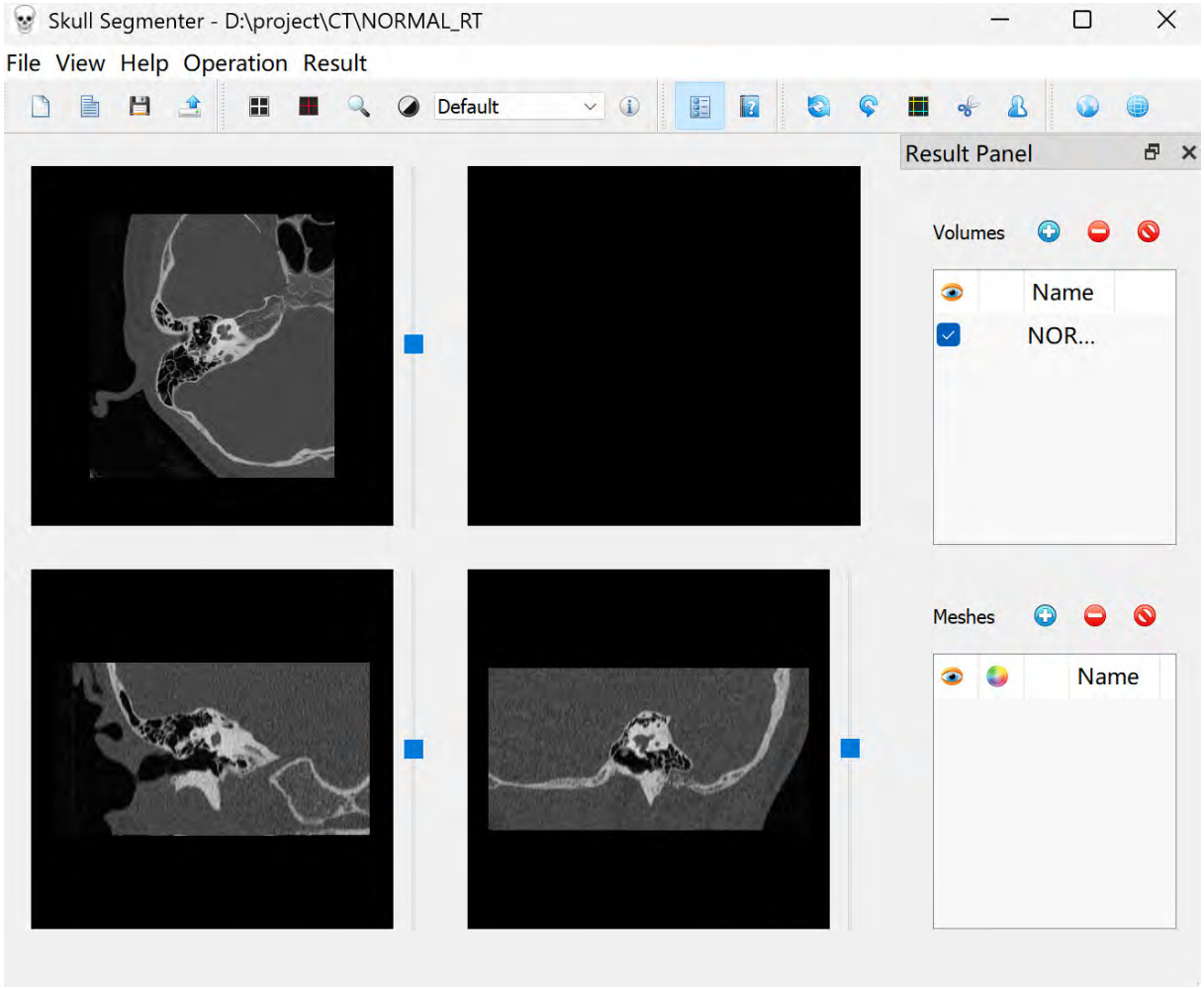

Figure 4: Overall GUI of the segmentation tool. After successfully reading normal right ear CT file, the corresponding slices can be seen in three 2d plane windows.

The three-view display system provides comprehensive visualization of 3D medical volumes through simultaneous display of three orthogonal planes, enabling users to understand spatial relationships and navigate through the volume efficiently. The system displays three orthogonal views, each representing a different slice through the volume:

- Transverse View (XY Plane): Displays axial slices, perpendicular to the superior-inferior axis. This view is commonly used in CT imaging as the primary acquisition plane and provides cross-sectional views of anatomical structures.
- Coronal View (XZ Plane): Displays coronal slices, perpendicular to the anterior-posterior axis. This view provides front-to-back cross-sections, useful for visualizing structures that extend vertically through the volume.

- Sagittal View (YZ Plane): Displays sagittal slices, perpendicular to the left-right axis. This view provides side-to-side cross-sections, complementary to the coronal view for understanding lateral anatomical relationships.

Each view is implemented using VTK's vtkImageViewer2, which efficiently renders 2D slices from 3D volume data. Besides, the system integrates several advanced visualization and interaction features to enhance 3D medical image exploration and analysis.

The application provides three synchronized orthogonal views (axial, coronal, and sagittal) to ensure spatial correspondence across all planes. When users navigate through slices in one view—via sliders or mouse scrolling—the corresponding slices in the other two views automatically update to maintain consistent 3D alignment. This synchronization is achieved through slice index calculation, and event-driven updates.

The system also supports real-time window and level (contrast and brightness) adjustment, allowing users to optimize visualization for tissues with varying intensity ranges. Adjustments can be made interactively through mouse or slider controls, or by selecting from preset protocols (e.g., CT Bone, CT Brain, CT Skin). Each view supports independent settings, enabling simultaneous optimization for different anatomical regions. This feature is implemented using VTK's vtkImageMapToWindowLevelColors, which dynamically maps voxel intensities to display colors without altering the original data, thus preserving quantitative accuracy for later analysis.

3.2.2 Set/Crop VOI

The Volume of Interest (VOI) selection feature enables users to interactively define regions of interest within the 3D volume through synchronized boundary line adjustments in the 2d views. Users can independently position top, bottom, left, and right boundary lines in each view, with real-time visual feedback showing the selected region across all three orthogonal planes. The system extracts the selected region as a new volume dataset using VTK's vtkExtractVOI filter, preserving spatial relationships from the original data to ensure accurate anatomical representation. A notable capability is the support for multi-level cropping, allowing users to apply additional VOI selections on previously cropped volumes, which is particularly useful for progressively refining anatomical focus or working with nested structures.

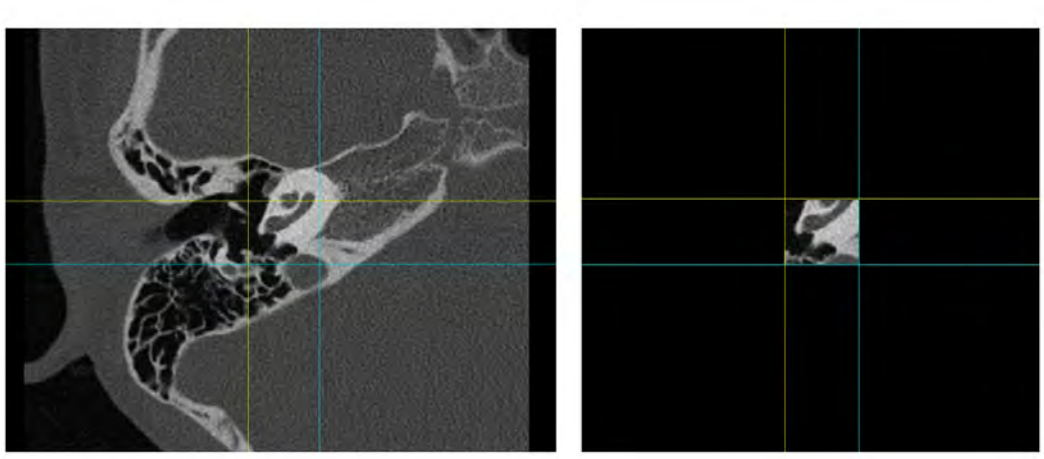

Figure 5: The demonstration of VOI-related functions. By selecting VOI in the box and clicking the Crop VOI button, users can perform subsequent segmentation operations in finer areas.

The implementation uses interactive line widgets (VoiWidget) that handle mouse-based boundary positioning, with automatic coordinate transformations between world coordinates and voxel indices to ensure accurate volume extraction regardless of the original data's spatial properties. Upon cropping, the three views are automatically updated to display the extracted volume, maintaining the synchronized navigation and visualization capabilities of the original workflow. Additionally, when a VOI has been previously defined, the thresholding tool receives the cropped volume as input through the extract_voi filter's output port, creating a pipeline chain where VOI selection and thresholding work together, enabling users to first narrow the anatomical region and then apply precise intensity-based segmentation.

When using set voi to select voi regions, the cross hair function can provide assistance. It is a dynamic crosshair system that provides visual feedback in all three views, marking the current slice position in other planes. The horizontal and vertical lines use VTK's vtkLineSource and vtkActor components as overlays for rendering and automatically update during navigation. Users can switch the crosshairs as needed to facilitate flexible viewing and precise anatomical localization in diagnostic or segmentation workflows.

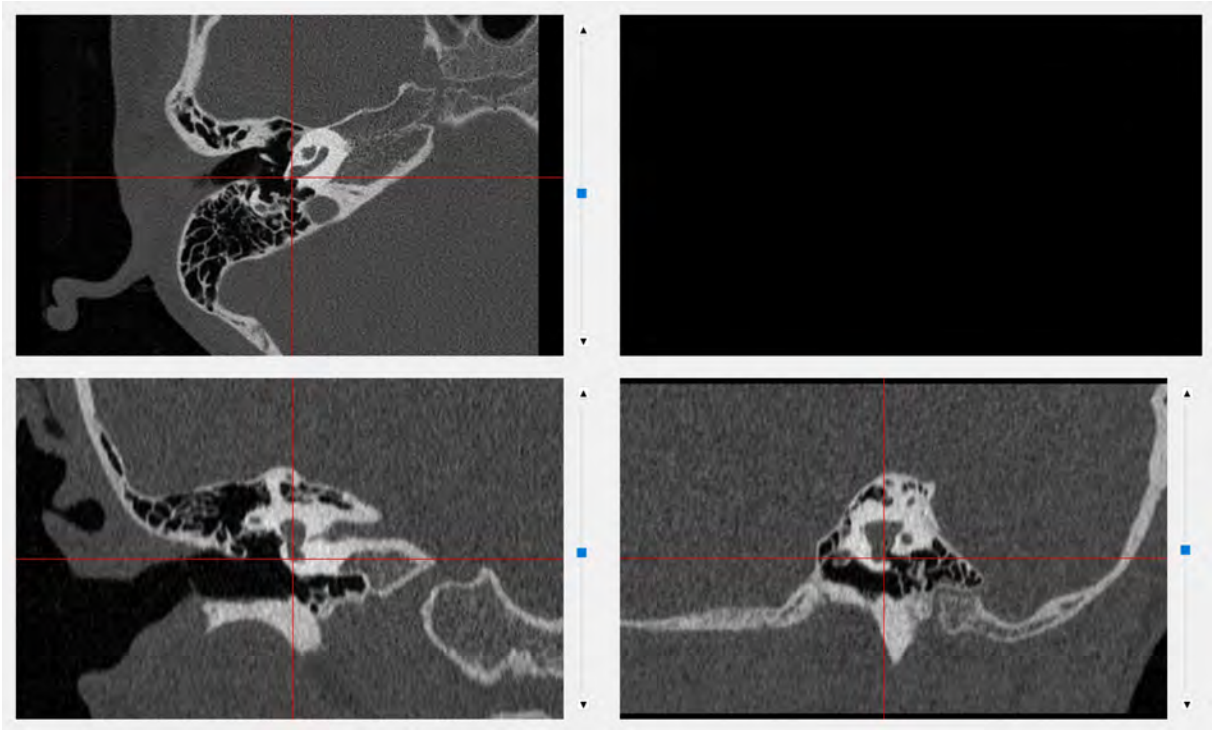

Figure 6: Using the cross hair function on the transverse view of normal right ear to quickly locate the round window area.

3.2.3 Threshold-Based Segmentation

Threshold-based segmentation is a fundamental technique for isolating anatomical structures in medical images based on intensity values. Our system implements an interactive thresholding tool that provides flexible parameter control and real-time visualization, enabling users to efficiently segment structures of interest such as bone, soft tissue, or specific anatomical regions within the cochlear implant context.

The thresholding tool supports three distinct threshold modes, each suited for different segmentation scenarios:

- Lower Threshold Mode: Preserves all pixels with intensity values greater than the specified threshold value. This mode is particularly useful for segmenting structures with high intensity values, such as bone in CT images, where the threshold defines the minimum intensity required for inclusion in the segmentation result.

- Between Threshold Mode: Preserves pixels with intensity values within a specified range, defined by lower and upper bounds. This mode enables segmentation of structures with intermediate intensity values, such as soft tissues or contrast-enhanced regions, by excluding both low-intensity background and high-intensity structures.

- Upper Threshold Mode: Preserves all pixels with intensity values less than the specified threshold value. This mode is useful for segmenting low-intensity structures or for excluding high-intensity artifacts from the segmentation result.

Figure 7: Adjust the lower threshold to achieve smoother and more complete skin. After selecting binaried, both the skin edges and internal fragments become more distinct.

Users can switch between threshold modes through radio button selection in the threshold dialog, with each mode automatically configuring the appropriate VTK filter parameters. The threshold values can be adjusted in real time, with immediate visual feedback in the three orthogonal views, enabling rapid iteration to achieve optimal segmentation results.The threshold segmentation module offers several advanced parameters to refine segmentation precision beyond the basic mode selection. Users can interactively adjust upper and lower threshold bounds through synchronized sliders and spinboxes, enabling both coarse and fine control based on image intensity characteristics. Additionally, an optional binarization mode converts the segmentation output into a binary mask, assigning 1 to segmented voxels and 0 to all others.

Overall, the threshold-based segmentation is implemented using VTK's vtkImageThreshold filter, which operates directly on vtkImageData structures and integrates seamlessly into the system's data processing pipeline. The three threshold modes are implemented through corresponding VTK methods dynamically selected based on user mode selection, while threshold values are updated in real time. The integration with the overall workflow is managed through the system's output_filter, which receives the threshold filter's output port connection, ensuring thresholded data automatically flows into sub-

sequent processing steps. Advanced features such as fill value assignment and binarization are handled through VTK filter configuration (SetInValue, ReplaceInOn), while state management tracks threshold application status and preserves parameter values, enabling iterative refinement throughout the segmentation workflow.

3.2.4 Mesh Generation

The mesh generation module converts segmented volumetric data into high-quality 3D surface meshes through a multi-stage pipeline that combines the Marching Cubes algorithm with several optimization techniques.

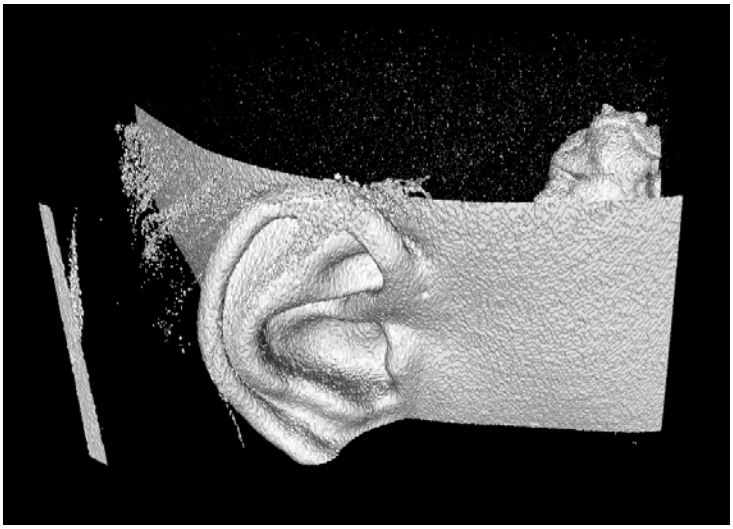

Figure 8: Original mesh generated by the segmentation tool.

The process begins with the Marching Cubes algorithm, which extracts an isosurface from the input volume based on a user-defined intensity threshold. Users can interactively adjust the intensity threshold to control which tissues are included in the surface. Lower thresholds are suitable for soft-tissue surfaces such as skin, whereas higher values capture denser bone structures.

Following initial extraction, the raw mesh often contains excessive polygons and surface irregularities. To address this, the system applies a two-stage mesh optimization pipeline comprising decimation and surface smoothing. Decimation reduces mesh complexity using VTK's vtkDecimatePro filter based on an edge-collapse algorithm. The simplification is governed by a ratio parameter (0.0–1.0), where a default value of 0.2 retains approximately 80% of the original triangles. Lower ratios (0.0–0.1) maintain fine anatomical details, while higher values (0.3–0.5) improve computational efficiency.

Subsequently, surface smoothing is applied using a Laplacian smoothing algorithm (vtkSmoothPolyDataFilter) to remove residual noise and reconnect discontinuous regions. The smoothing factor (0.0–1.0) controls vertex displacement, with values around 0.4–0.5 typically providing a balance between noise reduction and shape preservation. The process is iterative, generally running 1–30 cycles, where early iterations merge small surface fragments and later iterations refine overall smoothness. This gradual adjustment effectively bridges minor gaps and yields a continuous, high-quality surface suitable for visualization and downstream processing.

3.3 Post process

The models obtained from thresholding and marching cube generation algorithms in our segmentation tool often require further processing to achieve better display results. As shown in Figure 9-11, to reduce noise and smooth the model surface, MeshLab's Remove Isolated Pieces, Laplacian Smoothing, and Screened Poisson Reconstruction were applied to each model initially generated from the segmentation tool. In addition, features such as Fill Holes, Deleting selected vertices have also proven very useful in post-processing.

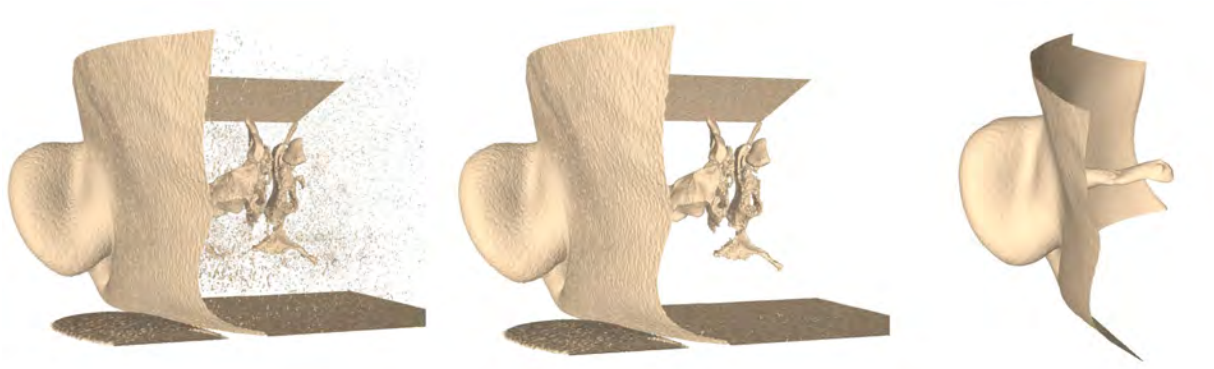

Figure 9: Use MeshLab to post-process the generated skin model. The unprocessed skin model retained a large amount of internal tissue debris, which needed to be removed. The rightmost image shows the result after filtering Poisson reconstruction and removing some invalid tissue.

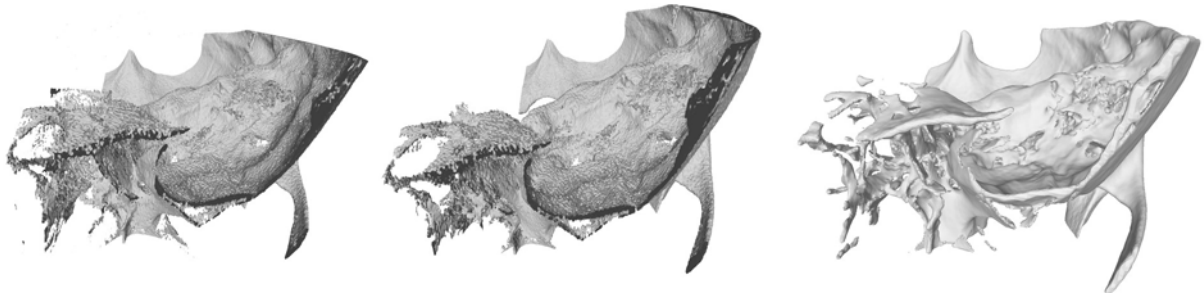

Figure 10: Use MeshLab to post-process the generated skull model. The challenge with skull models isn't removing inner fragments, but rather that Screened Poisson Reconstruction sometimes incorrectly processes parts of the skeleton. In practice, this issue is addressed by reconstructing individual skeleton sections separately.

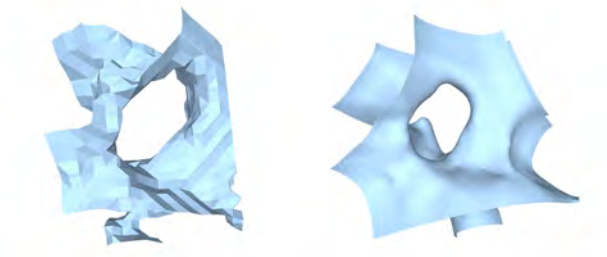

Figure 11: Use MeshLab to smooth the generated round window.

4 Test Results

4.1 Cochlear Segmentation

For the extraction of the outer ear surface and surrounding soft tissue, the basic setting of threshold is between -400HU and -300HU, which ensures that the skin surface segmentation provides a clear outer boundary and maintains the integrity of the overall head contour for all patients. The results confirm that the threshold-based approach reliably separates low-density tissue from high-density bone regions.

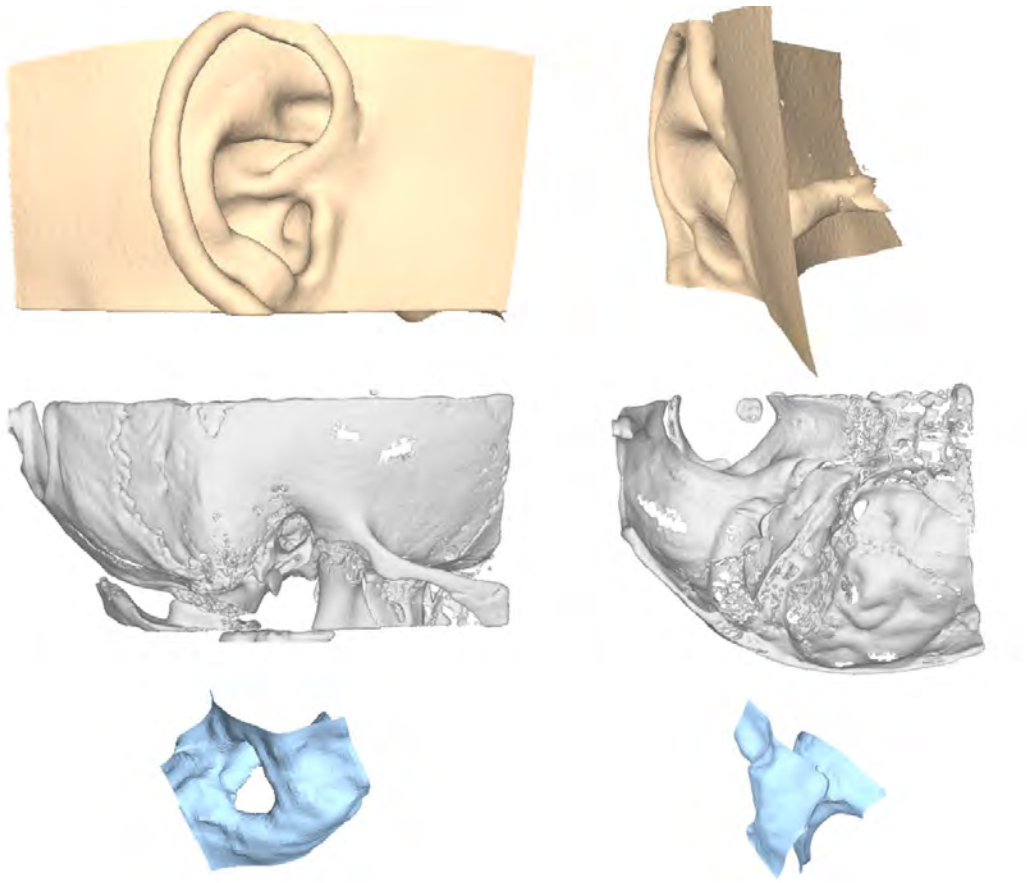

Figure 12: The reconstruction of normal right ear.

In the skin segmentation results, regions near the external auditory canal showed clear differences between the normal and abnormal ears. In the normal ear skin segmentation results, regions were captured around the external auditory canal. This part appeared as irregular, patch-like structures extending inward from the outer canal wall. Their intensity values were close to the range of the skin threshold, suggesting they might represent transitional zones between soft tissue, partially air-filled bone, and cortical bone. The segmented pattern gradually diminished toward the deeper temporal bone. In

the contrast, these regions did not appear on the abnormal ear skin model.

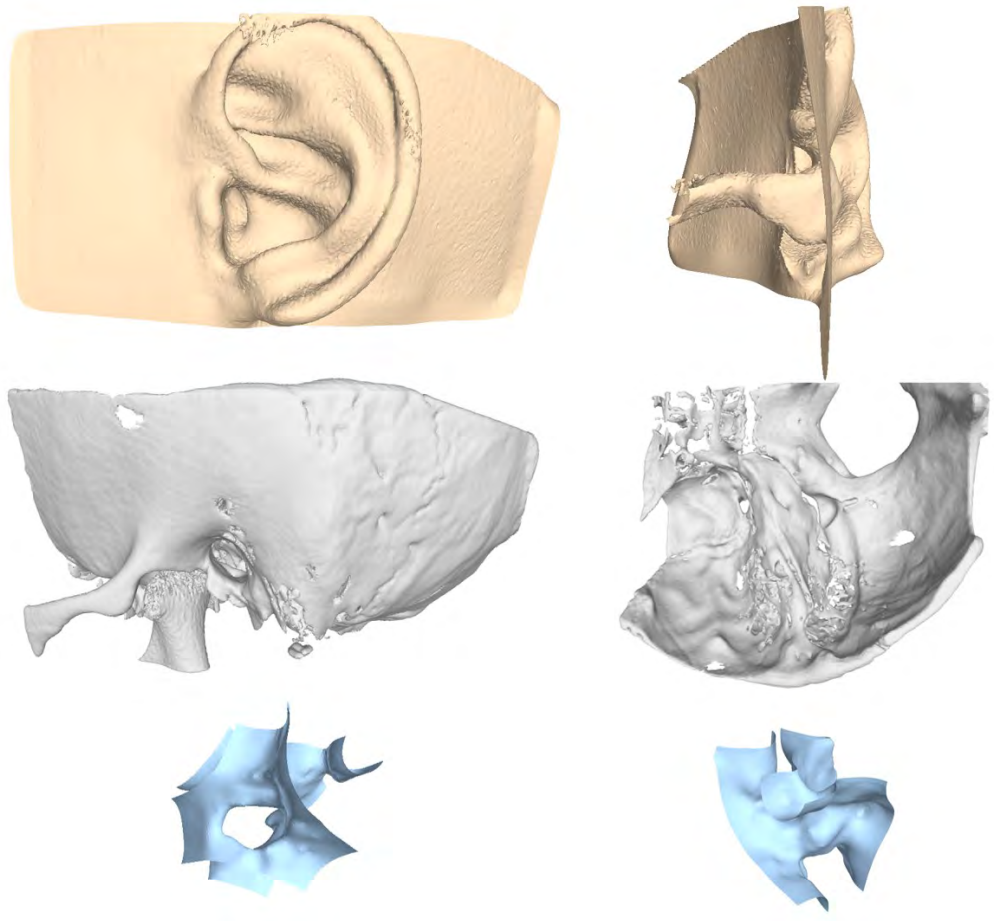

Figure 13: The reconstruction of normal left ear.

The temporal skull region was segmented using a higher intensity threshold of approximately 500HU - 600 HU, effectively capturing the dense cortical bone structures. The resulting meshes clearly depict the bony contours and the internal cavities surrounding the cochlear. Although full temporal-bone modeling is not strictly required for cochlear implantation procedures, the present reconstruction provides a clearer anatomical context for understanding the surrounding skeletal structures.

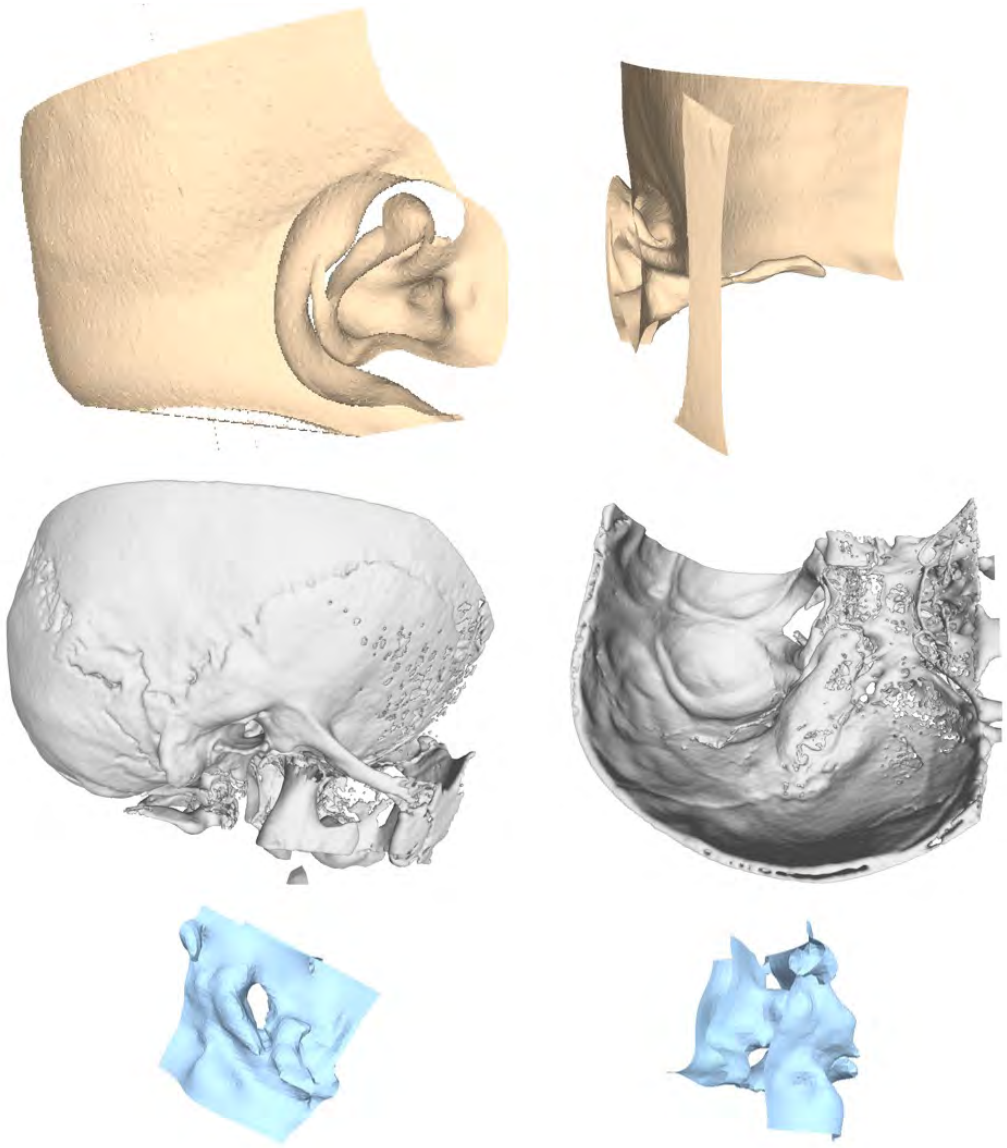

Figure 14: The reconstruction of abnormal right ear. The incomplete skin was due to the CT input itself.

In normal ears, the overall bony morphology appears thin, smooth, and well-defined, with a clearly identifiable external auditory canal opening positioned along the expected lateral direction. In contrast, the abnormal ear models show noticeably thicker and bulkier outer bone regions, with the canal opening located in a deeper and more medial position. These differences highlight the morphological variability of the temporal bone and its potential influence on surgical access and implant trajectory planning.

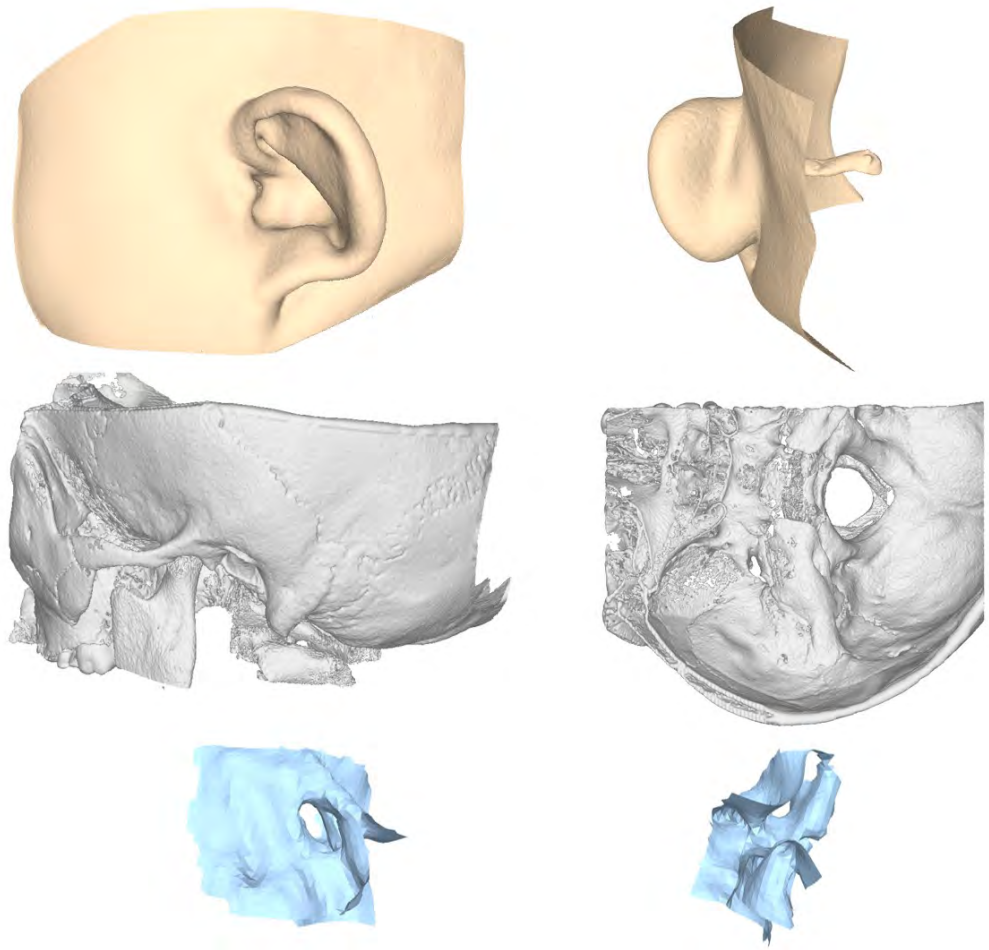

Figure 15: The reconstruction of abnormal left ear.

The round window was segmented as a small, low-intensity region located on the posterior wall of the cochlea. Although it lies within a low-density area corresponding to soft tissue and fluid, it is enclosed by dense bone in CT images. In this context, the segmentation of the round window was guided by the surrounding bony boundaries, since the thin membranous structure itself is not directly visible on standard-resolution CT scans. The precise position of the round window was determined by progressively narrowing the Volume of Interest (VOI), which allowed focused inspection of the target region.

It can be observed separately that in normal patients, the round window exhibits a smoother and more regular morphology with a consistent orientation, making the process of trimming the VOI simpler. In contrast, the abnormal cases show narrower, irregular openings that deviate from the typical direction, requiring more meticulous judgment.

4.2 Cochlear Simulation

After implementing mesh generation, we tried putting the model into the simulation tool, where we attempted to simulate the cochlear implantation surgery process. This generally consists of two steps: the first step is to use a skin cutter to cut open the skin, and the second step is to use a diamond burr to drill the skull. In the normal cases, the round window is easier to locate — its orientation and shape are more regular, facing in a predictable direction that aligns well with the expected drilling path. In contrast, the abnormal ears show irregular or tilted round window openings, sometimes partially covered by surrounding bone, which makes it much harder to identify and plan the electrode insertion path accurately, as shown in Fig 16-19.

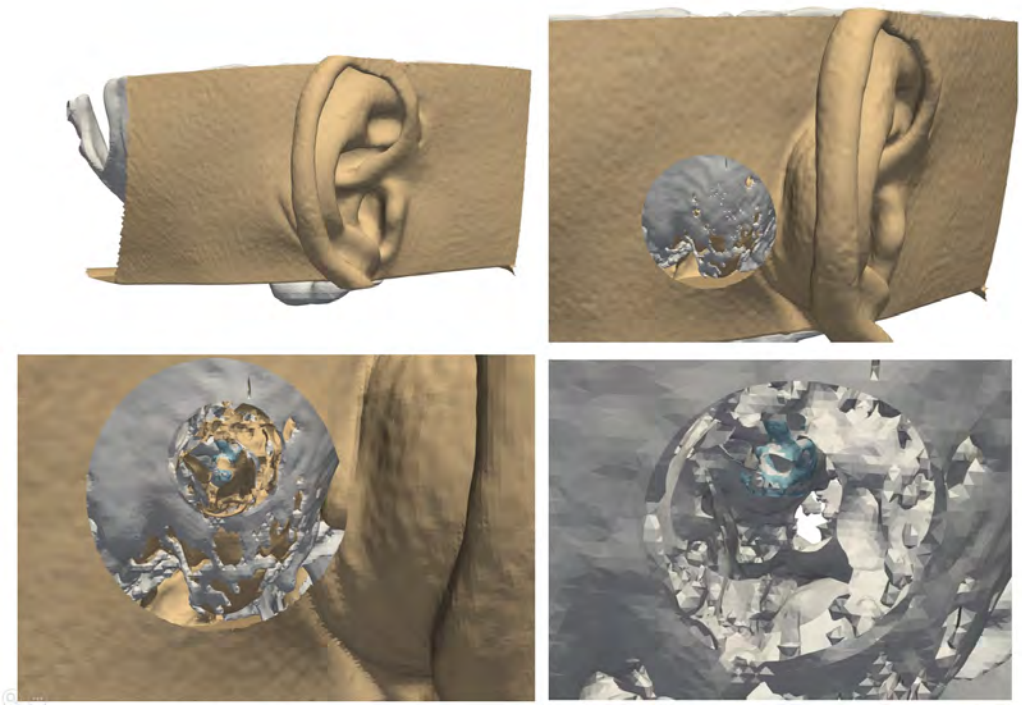

Figure 16: Schematic diagram of different cutting and drilling stages of the 3D normal right ear model.

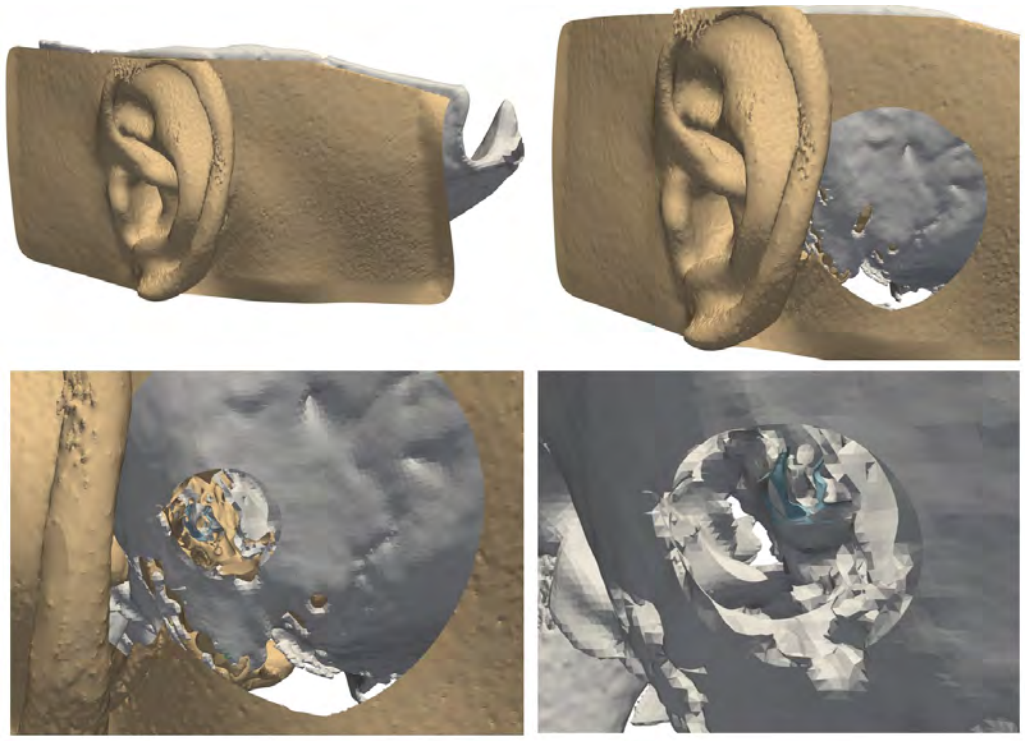

Figure 17: Schematic diagram of different cutting and drilling stages of the 3D normal left ear model.

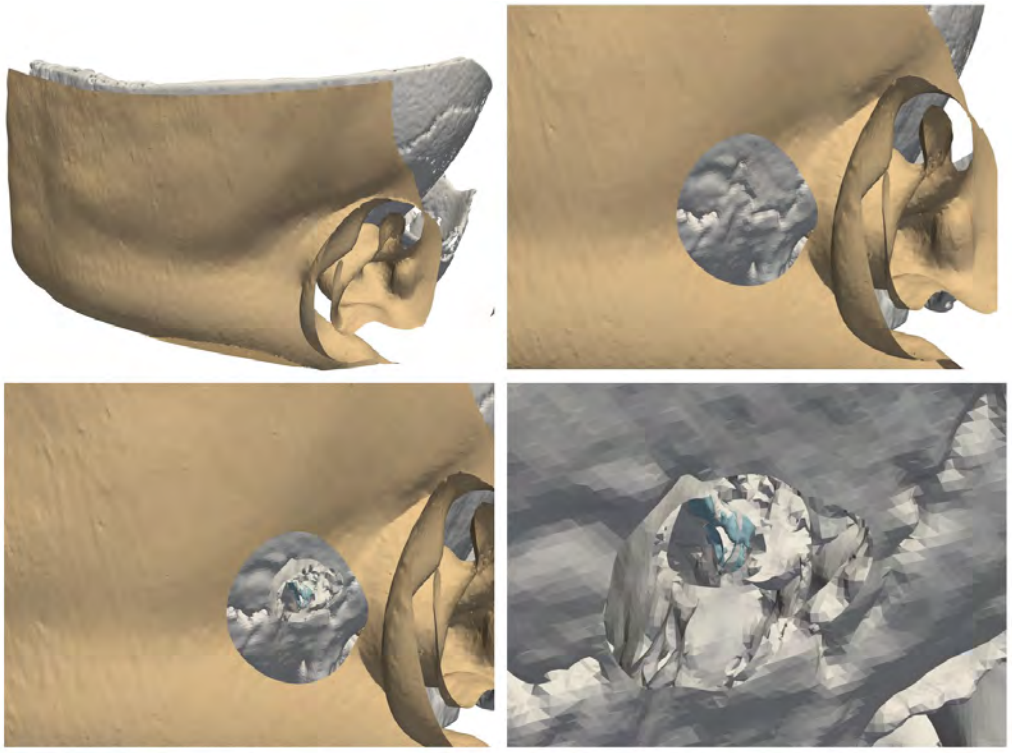

Figure 18: Schematic diagram of different cutting and drilling stages of the 3D abnormal right ear model.

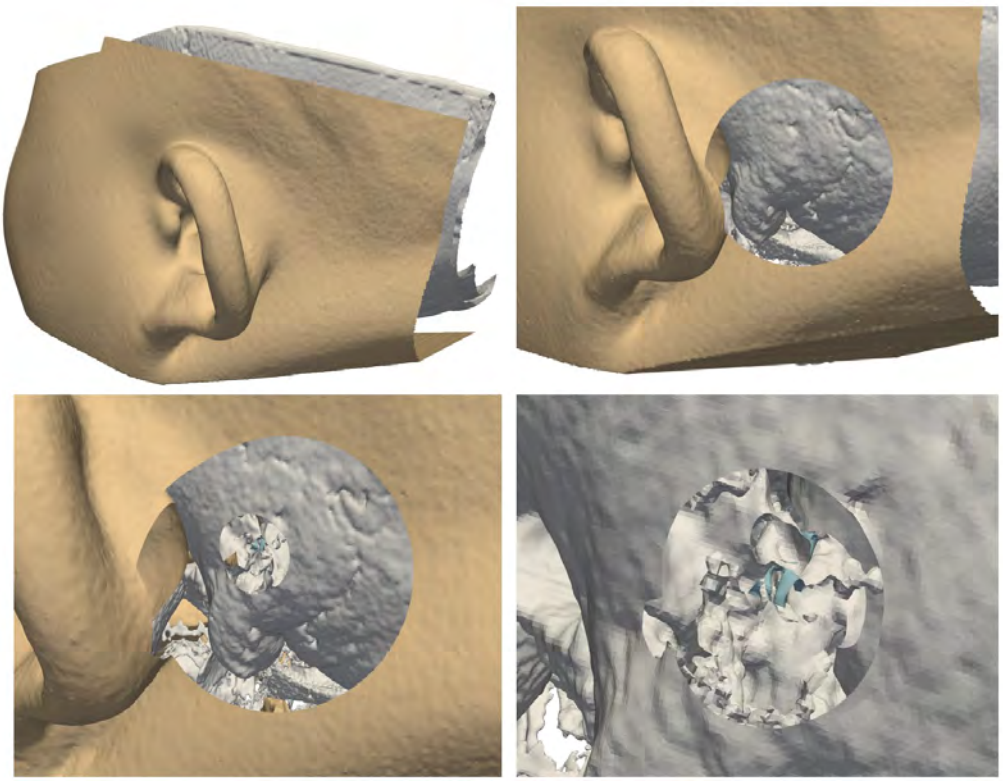

Figure 19: Schematic diagram of different cutting and drilling stages of the 3D abnormal left ear model.

Although no quantitative comparison was performed, both the original CT slices and the reconstructed 3D models clearly demonstrate the structural complexity and variability of the cochlear region. The differences between normal and abnormal ears are primarily reflected in the distribution of inner tissue, the thickness and continuity of the bony structures, and the shape and orientation of the round window. These abnormalities make it difficult to interpret the underlying anatomy directly from CT images, as many fine structures overlap or exist ambiguous intensity boundaries. Through 3D reconstruction, the spatial relationships among different components, such as the cochlea, surrounding bone, and surgical access routes become much clearer. The reconstructed models provide an intuitive understanding of individual anatomical variations and help identify regions that may pose challenges during surgery. Overall, the modeling process not only enhances structural interpretation but also lays a solid foundation for the simulation and surgical planning tasks.

5 Conclusion

While the current segmentation tool provides comprehensive functionality for medical image analysis, several enhancements could further extend its capabilities and improve its applicability in clinical and research settings.

The segmentation tool is designed as the first component of a comprehensive cochlear implant surgery assistant system, with simulation functionality originally planned as the subsequent phase. Due to time constraints in the current implementation, the simulation component has not yet been integrated but is used as a standalone tool as a separate tool. However, the author believes the integration is highly feasible given the tool's modular architecture. Specifically, the generated mesh models can serve as input for simulation analysis, while the platform's modular design, including clear separation of data management, rendering, and segmentation logic, enables simulation functionality to be integrated as an additional module, leveraging existing interfaces and state management.

Additionally, the integration of automated segmentation algorithms, particularly deep learning-based methods, will present a significant opportunity to enhance the tool's efficiency and accuracy. Current segmentation and reconstruction workflows rely primarily on manual or semi-automatic methods, which can be highly customized but also time-consuming due to the difference between CT and patients. Deep learning models, trained on large datasets of medical images, can provide rapid and consistent segmentation results for anatomical structures and likely to speed up the reconstruction processing flow.

In conclusion, the segmentation tool presented in this work successfully addresses the critical requirements for cochlear implant surgical planning by providing comprehensive functionality for segmenting cochlear-related anatomical structures. It enables precise anatomical delineation even in cases with complex anatomical variations or image artifacts. The modular architecture and Python-based implementation provide a solid foundation for future enhancements, particularly integration with simulation functionality, while maintaining the tool's accessibility and usability for clinical practitioners. As the first component of a comprehensive cochlear implant surgery assistant system, this segmentation tool will provide a reliable foundation for accurate 3D cochlear anatomical modeling.

References

[1] World Health Organization, *World report on hearing*, 2021. [Online]. Available: https://www.who.int/publications/i/item/9789240020481.

[2] P. e. a. Blamey, "Factors affecting auditory performance of postlinguistically deaf adults using cochlear implants: An update with 2251 patients," *Ear and Hearing*, vol. 34, no. 6, e1–e17, 2013. DOI: 10.1097/AUD.0b013e31829b53e7.

[3] C. A. e. a. Buchman, "Cochlear implant candidacy in adults: A review of recent changes and considerations," *Otology & Neurotology*, vol. 41, no. 7, pp. 897–904, 2020. DOI: 10.1097/MAO.0000000000002660.

[4] T. A. e. a. Zwolan, "Cochlear implants for adult medicare beneficiaries who met criteria for implantation," *JAMA Otolaryngology–Head & Neck Surgery*, vol. 146, no. 10, pp. 933–941, 2020. DOI: 10.1001/jamaoto.2020.2286.

[5] G. B. e. a. Wanna, "Cochleostomy versus round window insertion: A meta-analysis of outcomes in cochlear implantation," *Otology & Neurotology*, vol. 35, no. 3, e85–e93, 2014. DOI: 10.1097/MAO.0000000000000241.

[6] M. L. e. a. Carlson, "Cochlear implantation: Current outcomes and complications," *Otolaryngologic Clinics of North America*, vol. 45, no. 1, pp. 187–201, 2012. DOI: 10.1016/j.otc.2011.08.015.

[7] J. H. Noble, B. M. Dawant, and R. F. Labadie, "Statistical shape model segmentation and frequency mapping of the cochlea," *Medical Image Analysis*, vol. 16, no. 3, pp. 632–638, 2012. DOI: 10.1016/j.media.2012.01.010.

[8] W. e. a. Wimmer, "Cochlear duct length and variability: Implications for cochlear implantation," *Otology & Neurotology*, vol. 36, no. 7, pp. 1137–1144, 2015. DOI: 10.1097/MAO.0000000000000780.

[9] M. e. a. Ceresa, "Patient-specific segmentation of the cochlea and inner ear structures in clinical ct for individualized ci simulation," *Physics in Medicine and Biology*, vol. 61, no. 2, pp. 790–807, 2016. DOI: 10.1088/0031-9155/61/2/790.

[10] M. e. a. Frendø, "Cochlear implant surgery: Virtual reality simulation training and transfer of skills to cadaver dissection—a randomized controlled trial," *Journal of International Advanced Otology*, vol. 18, no. 3, pp. 219–224, 2022. DOI: 10.5152/iao.2022.21234.

[11] D. L. Pham, C. Xu, and J. L. Prince, "Current methods in medical image segmentation," *Annual Review of Biomedical Engineering*, vol. 2, pp. 315–337, 2000. DOI: 10.1146/annurev.bioeng.2.1.315.

[12] T. Heimann and H.-P. Meinzer, "Statistical shape models for 3d medical image segmentation: A review," *Medical Image Analysis*, vol. 13, no. 4, pp. 543–563, 2009. DOI: 10.1016/j.media.2009.05.004.

[13] P. A. Yushkevich, J. Piven, H. C. Hazlett, *et al.*, "User-guided 3d active contour segmentation of anatomical structures: Significantly improved efficiency and reliability," *NeuroImage*, vol. 31, no. 3, pp. 1116–1128, 2006. DOI: 10.1016/j.neuroimage.2006.01.015.

[14] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, *et al.*, "3d slicer as an image computing platform for the quantitative imaging network," *Magnetic Resonance Imaging*, vol. 30, no. 9, pp. 1323–1341, 2012. DOI: 10.1016/j.mri.2012.05.001.

[15] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988. DOI: 10.1007/BF00133570.

[16] J. A. Sethian, *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 1999, ISBN: 9780521645578.

[17] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2003, ISBN: 9780387004301.

[18] B. B. Avants, N. J. Tustison, G. Song, P. A. Cook, A. Klein, and J. C. Gee, "A reproducible evaluation of ants similarity metric performance in brain image registration," *NeuroImage*, vol. 54, no. 3, pp. 2033–2044, 2011. DOI: 10.1016/j.neuroimage.2010.09.025.

[19] S. Klein, M. Staring, K. Murphy, M. A. Viergever, and J. P. W. Pluim, "Elastix: A toolbox for intensity-based medical image registration," *IEEE Transactions on Medical Imaging*, vol. 29, no. 1, pp. 196–205, 2010. DOI: 10.1109/TMI.2009.2035616.