International Journal of Applied Technology in Medical Sciences
Vol 3 Issue 1 (2023)
Pages (17 - 25)

Available at www.emiratesscholar.com

Interactive GUI for Advanced Visualization and Volume Rendering of MRI Data: A MATLAB - Based Approach

Noora Saeed Alhajeri
n.s3eedalhajeri@gmail.com
Department of Biomedical Engineering, Philips Healthcare, Abu Dhabi, UAE

ARTICLE INFO

Published on 11th of June 2024.
Doi: 10.54878/rxzd0j10

KEYWORDS

MATLAB GUI, MRI visualization, Volume rendering, medical imaging, Maximum Intensity Projection, Minimum Intensity Projection, Isosurface, Slice Planes, Gradient Opacity, Orthogonal Slice View.

HOW TO CITE

Interactive GUI for Advanced Visualization and Volume Rendering of MRI Data: A MATLAB - Based Approach. (2024). *International Journal of Applied Technology in Medical Sciences*, *3*(1), 17-25. https://doi.org/10.54878/rxzd0j10

ABSTRACT

This paper presents the development and evaluation of a graphical user interface (GUI) for visualizing and volume rendering MRI volumetric data using MATLAB. The GUI incorporates various techniques such as Maximum Intensity Projection, Minimum Intensity Projection, Isosurface, Slice Planes, Gradient Opacity, Orthogonal Slice View, and Slice View to enhance the usability of the system for different viewing styles. The project aims to facilitate the visualization of MRI data for clinical staff, offering functionalities to load abdominal, brain tumor, and heart images and apply desired visualization techniques. Through MATLAB code and the GUI, the project successfully achieved the desired results, meeting the expectations outlined in the project objectives. Challenges encountered during GUI design were addressed, and potential areas for improvement and future research were identified. The paper highlights the significance of the developed GUI in modern medicine, emphasizing its potential to improve diagnostics, treatment planning, and medical education. Overall, this project contributes valuable insights into MATLAB GUI development and its applications in medical imaging, laying the groundwork for further advancements in this field.

1. Introduction

This study delves into the evolutionary trajectory of volume rendering and visualization techniques, alongside the development and outcomes of our MRI Graphical User Interface (GUI). The overarching objective is to enhance the usability of volume rendering tools within medical imaging. The advent of novel medical imaging technologies has ushered in an era of unprecedented data generation. Consequently, a pressing need arises for the continuous innova- tion of analytical and visualization systems. Volume rendering, as a pivotal method for visualizing im- age data, has emerged as a cornerstone in medical imaging. Despite its potential to revolutionize cur- rent practices, volume rendering remains confined to a select few radiology facilities globally. Realiz- ing its transformative potential necessitates substan- tial enhancements and the development of intuitive, user-centric systems. In pursuit of this objective, we present a Graphical User Interface (GUI) tailored to the continuous visualization and volume render- ing of MRI data, with a specific focus on medical imaging applications. This endeavor is poised to not only streamline clinical workflows but also em- power both practitioners and patients by facilitating accurate diagnoses and treatment decisions. The suc- cessful implementation of our system holds profound implications for healthcare institutions and individuals alike, potentially catalyzing paradigm shifts in medical practice. However, the imperative nature of this undertaking is compounded by security considerations and the imperative for rigorous validation. Notably, the recruitment of personnel for system development and testing mandates stringent vetting processes to safeguard against security breaches and erroneous conclusions. At its core, the principal aim of this project is to cultivate an MRI data GUI ca- pable of comprehensively reviewing and evaluating medical records, discerning health statuses, and fur- nishing a robust platform for scholarly inquiry and analysis.

1. Literature Review

Volume Rendering Kajiya and Herzen pioneered the development of volume rendering with their seminal work on Ray Casting [11]. Surface representations, which involve the drawing of geometric primitives, resulting in polygonal meshes and the loss of information from one dimension, have tradition- ally been used to construct three-dimensional objects. This technique, based on mathematical modeling of light behavior equations, introduced the concept of the volume rendering integral, marking a significant advancement in the field [12]. Ray casting algorithms, such as Early Ray Termination and Empty Space Skipping, emerged as effective means to accelerate the rendering process, optimizing computational resources [13]. Early ray termination is a process that limits the volume so that subsequent samples do not contribute to the pixel's value.

The inherent parallelizability of ray casting algorithms makes them well-suited for GPU architectures, leveraging the parallel processing capabilities of modern graphics hardware [14]. GPU-based volume rendering techniques, facilitated by 3D texture storage and fragment shader programs, have demonstrated notable efficiency gains compared to traditional CPU-based rendering approaches [15]. However, the increased complexity of volume rendering methods poses challenges in handling large datasets and optimizing rendering times, necessitating the development of optimized rendering strategies [16, 17].

Visualization of Medical Images Since image inter- pretation can lead to clinical intervention, medical visualization is a complex scientific field. As a result, quality and rapid interactive response are critical in this domain. Recent advances in medical imaging technology and applications have made it possible to share imaging data online between clinical and research centers, as well as between clinicians and patients. Concerns about connectivity, security, and resource heterogeneity all have an impact on the development of these applications. As demonstrated in Figure 1, on-server rendering is a partial solution for Medical Imaging [18]. Several web-based volumet- ric visualization solutions have also been presented, though many of these solutions require third-party systems to allow visualization, or their scalability is limited by the rendering server [19].

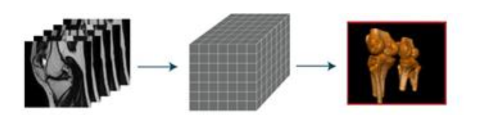

Figure 1: Extracting 3D bone from MRI of knee.

Besides that, the majority of fundamental volume visualization algorithms are divided into two groups:

Direct Volume Rendering (DVR) algorithms and Surface Fitting (SF) algorithms. SF algorithms fit (usually planar) surface primitives like polygons or patches to constant-value contour surfaces in volumetric data sets. After the user selects a threshold value, the geometric primitives are automatically fitted to the volume's high contrast contours that match the threshold. SF algorithms include opaque cubes, marching cubes, dividing cubes, and marching tetrahedra [20]. SF methods are typically faster than DVR methods because they only traverse the volume once to extract the surfaces. After the surfaces have been extracted, rendering hardware and other rendering methods can be used to quickly render the surface primitives whenever a viewing or lighting parameter is changed by the user. Changing the SF threshold value requires revisiting all of the cells to extract a new set of surface primitives, which takes time. SF methods have several flaws, including occasional false positive and negative surface pieces, as well as incorrect handling of small features and data branches. Furthermore, it only shows surfaces that meet a threshold density and the surface closest to the fictitious viewer. Ulti- mately, high-quality rendering and real-time interac- tion are critical goals in medical data visualization.

Recent studies in the field of volume rendering and medical image visualization have focused on addressing challenges related to real-time interac- tion, scalability, and security. For example, [21] pro- posed a novel GPU-accelerated rendering approach to improve real-time visualization of large medical datasets, while [22] investigated the use of blockchain technology for secure and decentralized sharing of medical imaging data. Additionally, advancements in deep learning techniques have shown promise in enhancing the quality and efficiency of medical image visualization [23].

2. Methodology

A friendly user interface using MATLAB's App De- signer was employed to develop the graphical user interface (GUI) for MRI data visualization and volume rendering. This approach facilitated the creation of a user-friendly GUI by providing a drag-and-drop interface for designing buttons, axes, and menus, eliminating the need for extensive MATLAB expertise. The GUI was then coded and programmed in MATLAB to ensure proper functionality and alignment with the project's objectives.

The GUI was designed to prepare users for MRI data visualization and volume rendering, enabling the utilization of MRI data from at least two different organs. The workflow began with the loading of MRI data, typically consisting of 3D data composed of several slices. Subsequently, a montage of the MRI data was displayed to facilitate visualization.

Various visualization techniques were implemented within the GUI to enable users to explore MRI data from different perspectives and angles. These techniques included:

1. Maximum Intensity Projection (MIP): This technique involves projecting the voxel with the highest attenuation value across the volume onto a 2D image at each view [3].
2. Minimum Intensity Projection (MinIP): It is a data visualization technique that detects low-density structures in a given volume [4].
3. Volume Rendering: A technique for data visualization that produces a three-dimensional representation of data [5].
4. Isosurface: A 3D surface representation of points in a 3D data distribution with equal values. It is the 3D equivalent of a contour line [6].
5. Slice Planes: A surface colored based on the volume data values in the region where the slice is positioned [7].
6. Gradient Opacity: A rendering style that uses localized transparency when sequential voxels have similar intensities [8].
7. Orthogonal Slice View: Displays the axial, coronal, and sagittal planes intersecting the currently selected coordinate [9].
8. Slice View: Perceives data in cross sections [10].

The primary objective throughout the GUI development process was to ensure a user-friendly experience, characterized by seamlessness, faultlessness, and visual appeal. User-centric design principles were prioritized to address user needs effectively. Considerations such as cost, public health, safety, and environmental impact were deemed less rele- vant given the project's focus on user experience.

The GUI development process was conducted digitally using MATLAB and MATLAB App Designer.

Therefore, the GUI was designed to meet specific specifications, including:

- Ease of use and implementation in MATLAB.
- Capability to load 3D MRI data composed of multiple slices.
- Ability to display a montage of MRI data for visualization.
- Support for visualization of MRI data from different angles using various techniques.

3. Visualization and Volume Rendering of MRI Data System (GUI)

The user-friendly interface depicted in Figure 2 facilitates the loading of MRI data, visualization of a montage, and application of various rendering techniques from different perspectives.

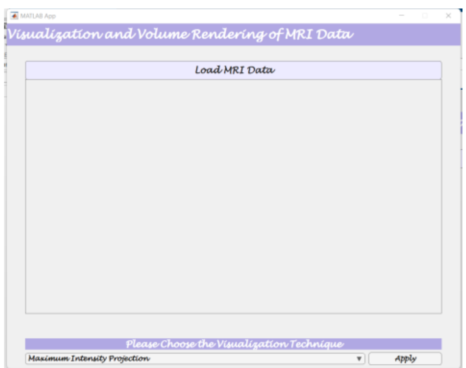

Figure 2: Visualization and Volume Rendering of MRI Data System.

4. Component Design

The project's primary objective is the development of a user-friendly GUI for MRI data visualization and volume rendering. To achieve this goal, the following features were implemented:

- Two axes were integrated to display a montage of selected MRI data and the output after applying a visualization technique.
- A button was provided to enable the loading of desired MRI images into the interface.
- Separate buttons were incorporated to facilitate the application of average intensity projection, maximum intensity projection, and volume rendering techniques.
- Additionally, a button was included to allow users to view the rendered output from different angles, enhancing the flexibility of data analysis.

5. Code Development

The development of the GUI commenced after the layout phase using MATLAB's App Designer. A fully functioning code was obtained through iterative processes, ensuring the GUI's efficacy in serving its intended purpose.

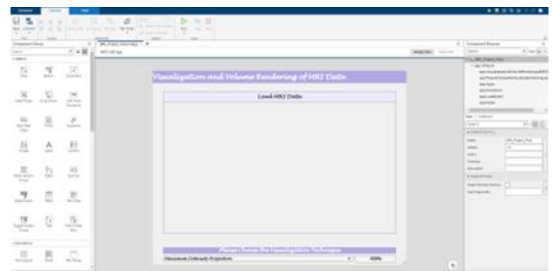

Figure 3: The MATLAB App Designer's design view window.

In the GUI design process, several key steps were implemented:

1. GUI elements were selected from the component library and placed onto the canvas, including a load button, a drop-down list for technique selection, and an apply push button (see Fig- ure 3). Techniques such as Maximum Intensity Projection, Minimum Intensity Projection, Volume Rendering, Isosurface, Slice Planes, Gradient Opacity, Orthogonal Slice View, and Slice View were incorporated into the drop-down list.
2. The Inspector window was utilized to adjust formatting and interactivity settings of the GUI components.
3. Callbacks were constructed for edit fields, switches, and buttons to add functionality to the GUI.
4. The code view was accessed to further refine the GUI's functionality and appearance (see Fig- ure 4).
5. Properties were created, with access set to public, facilitating the sharing of variables such as the 'volume' variable.
6. The 'Load Button' functionality was implemented to load volumetric data of the NIFTI format, such as Abdomen MRI image, Brain Tu-

mor MRI image, and Heart MRI image, and display it in a panel using the SliceViewer() function.

7. A switch case was constructed in the apply push button callback to handle technique selection and visualization.

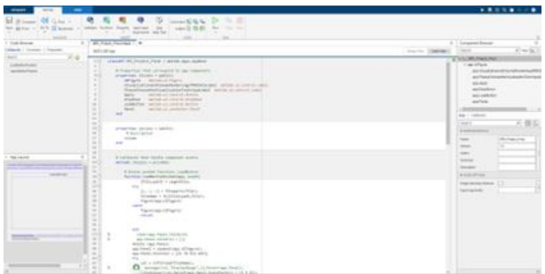

Figure 4: The MATLAB App Designer's code view window.

These steps resulted in a functional GUI capable of loading MRI data and applying various visualization techniques, providing users with a versatile tool for exploring volumetric medical imaging datasets.

6. System Overview

The principal objective of this project is to utilize MATLAB's App Designer to develop a graphical user interface (GUI) tailored for MRI data visualization and volume rendering. This system aims to empower users by providing intuitive tools to load MRI data, comprising 3D data slices, and employ visualization techniques for comprehensive data analysis. The system encompasses several key functionalities, including the generation of a montage of MRI data and the visualization of techniques such as Average Intensity Projection, Maximum Intensity Projection, and Volume Rendering applied to MRI data.

This framework serves to elucidate the impact of various visualization techniques on MRI data from diverse perspectives, facilitating a deeper understanding of their applicability and significance. The system operates seamlessly within MATLAB's GUI environment, offering users an accessible platform to explore and analyze MRI data with ease.

The holistic overview of the system architecture is depicted in Figure 5, illustrating the interconnected components and functionalities designed to streamline MRI data visualization and analysis processes.

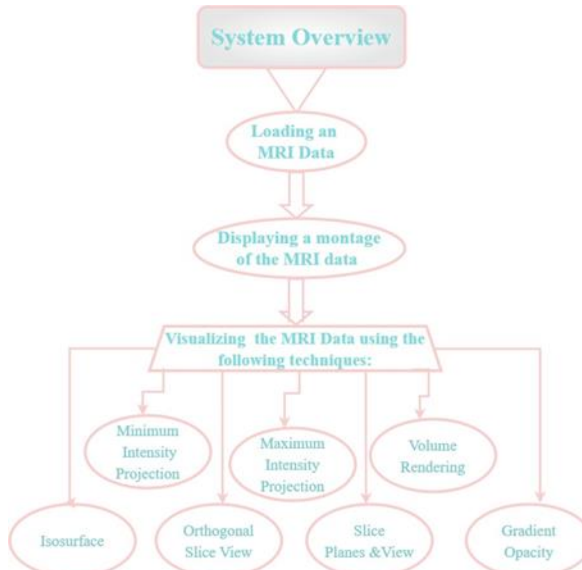

Figure 5: System Overview.

7. Results and Discussion

7.1. Experimental Testing

Three tests were conducted to assess the functionality of the GUI design and the generated codes.

7.1.1. First Test:

- Test Name: GUI Functionality with Volume Rendering of an Image.
- Test Description: Evaluation of volume rendering functionality.
- Procedure: Running the code, inserting a figure, and checking outputs.
- Observations: Successful volume rendering of the image.
- Acceptance Criteria: Clear volume rendering at different angles.
- Result: Success.

7.1.2. Second Test:

- Test Name: GUI Functionality with Maximum Intensity Projection.

- Test Description: Assessment of maximum intensity projection functionality.
- Procedure: Running the code, selecting the maximum intensity projection option, and observing outputs.
- Observations: Clear visualization of nerves.
- Acceptance Criteria: Observation of nerves.
- Result: Success.

7.1.3. Third Test:

- Test Name: GUI and Functionality of Slice Viewing.
- Test Description: Evaluation of slice viewing functionality.
- Procedure: Running the code and observing slice outputs.
- Observations: Successful visualization of selected image slices.
- Acceptance Criteria: Viewing of slices.
- Result: Success.

7.2. Analysis and Interpretation of Data

The project aimed to develop an application for visu- alizing and volume rendering MRI data, incorporat- ing various visualization techniques to enhance the system's usability for different viewing styles.

Selecting Any Figure: To select any volume MRI data, the user must first push the load image button of a. nii extension. The image is directly viewed as a sequence of several slices on a panel where the user can move the slider to view each slice alone as can be seen in Figure 6 below.

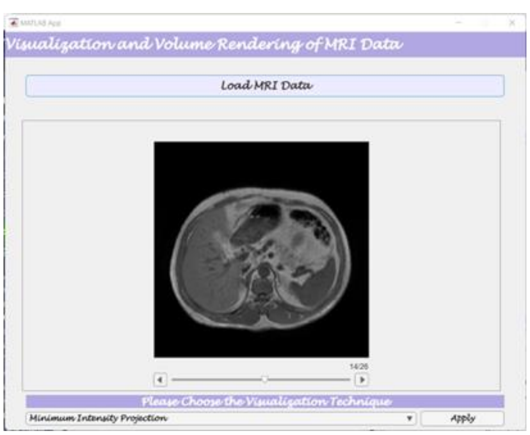

Figure 6: Abdomen MRI image slice viewer slider.

Drop Down Menu: The user can have multiple visualizations of MRI data using a drop-down menu.

The user can choose and apply the visualization tech- nique by clicking on the button apply. The user can choose between maximum intensity projection as shown in Figure 7, minimum intensity projection as shown in Figure 8, volume rendering as in Figure 8, Isosurface as in Figure 9, Slice View as in Figure 10, gradient opacity as in Figure 11, and orthogonal slice view as in Figure 12. Also, the user has the option to go back to the slice view slide that appeared right after uploading the image.

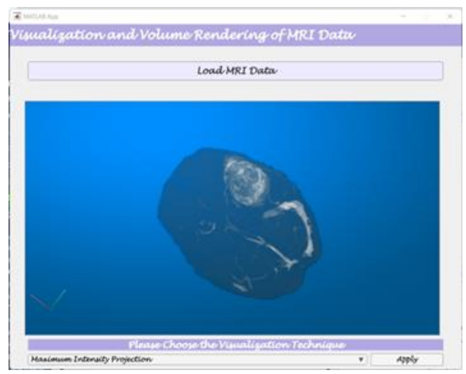

Figure 7: Brain Tumor MRI image visualized using MIP.

Figure 8: Abdomen MRI image visualized using minimum intensity projection

International Journal of Applied Technology in Medical Sciences
Vol 3 Issue 1 (2023) Pages (17 - 25)

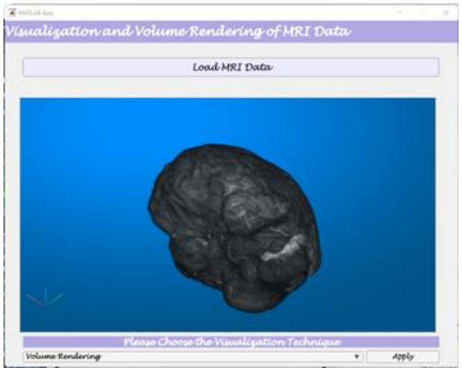

Figure 9: Brain Tumor MRI image visualized using volume rendering.

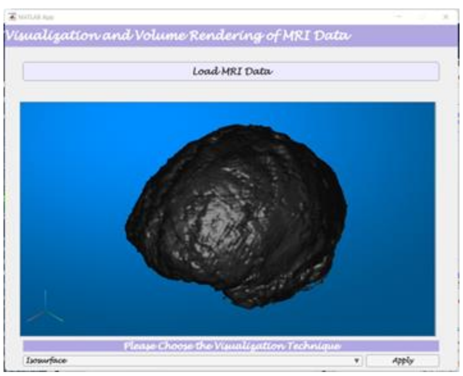

Figure 10: Brain Tumor MRI image visualized using Isosurface.

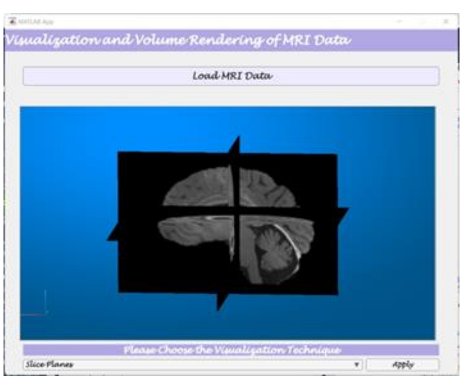

Figure 11: Brain Tumor MRI image visualized at several slices.

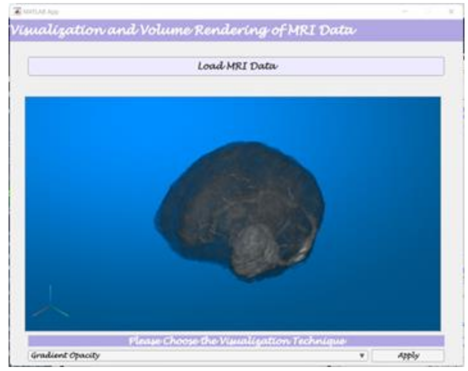

Figure 12: Brain Tumor MRI image visualized using gradient opacity.

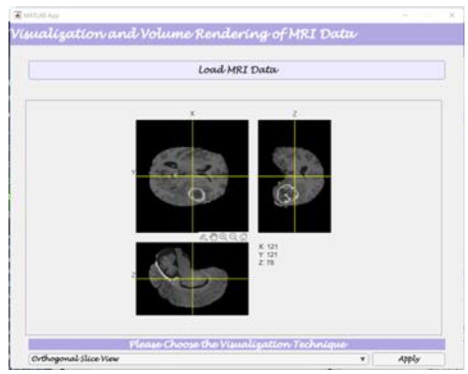

Figure 13: Brain Tumor MRI image visualized using orthogonal slice view.

The results obtained align precisely with the expected outcomes, demonstrating the successful functionality of the GUI. This robustness allows for diverse MRI data visualization techniques, aiding both students and medical professionals in their diagnos- tic and analytical endeavors.

8. Conclusion

The project aimed to develop and simulate a graphical user interface (GUI) for visualization and volume rendering using MRI volumetric data within MATLAB, encompassing various techniques such as Maximum Intensity Projection, Minimum Intensity Projection, Isosurface, Slice Planes, Gradient Opacity, Orthogonal Slice View, and Slice View. Leveraging MATLAB facilitated the attainment of results. The incorporation of a user-friendly GUI, particularly for clinical personnel, marked a significant advancement, enabling the loading of diverse MRI datasets such as abdominal, brain tumor, and heart images. The chosen techniques were deployed as a response to user inputs, yielding the desired outcomes. The project fostered a comprehensive understanding of MATLAB and the MATLAB Designer App, augmenting skillsets applicable in future coursework and professional endeavors. Notwithstanding, several challenges surfaced during GUI design, necessitating further elaboration on functionalities and specifications tailored to clinic applications. Moreover, the study primarily explored conventional computer-based interactions, warranting the exploration of alternative methods like haptic devices. Anticipated advancements in user interaction methods will likely herald novel applications of volume rendering in medical imaging, enhancing clinical decision-making and surgical planning. The envisioned design embodies both positive and impactful outcomes, particularly in healthcare settings, where it holds the potential to mitigate errors and save lives. Environmental impact remains negligible owing to MATLAB simulations, while stringent privacy measures ensure data security. As a crucial facet of modern medicine, continued exploration of MRI visualization and volume rendering is indispensable for advancing research and clinical practice.

References

1. “MathWorks - Makers of MATLAB and Simulink,” Mathworks.com, 2022. https://www.mathworks.com/?s_tid=gn_logo.
2. "MATLAB App Designer,” Mathworks.com, 2024. https://www.mathworks.com/products/matlab/app-designer.html.
3. A.Murphy,“Maximum intensity projection,” Radiopaedia, Mar. 23, 2023. https://radiopaedia.org/articles/ maximum-intensity-projection#:~:text= Maximum%20Intensity%20Projection%20(MIP)%20consists,onto%20a%202D%20image%201.
4. A. H. Duran, M. N. Duran, I. Masood, L. M. Maciolek, and H. Hussain, “The Additional Diagnostic Value of the Three-dimensional Volume Rendering Imaging in Routine Radiology Practice,” Cure¯us, Sep. 2019, doi: https://doi.org/10.7759/cureus.5579.
5. Navdeep Singh Gill, “3D and Multi- dimensional Data Visualization Tech- nique,” Xenonstack.com, May 03, 2022. https://www.xenonstack.com/insights/multidimensional-data-visualization.
6. “Isosurface | Scientific Volume Imaging,” Svi.nl, 2022. https://svi.nl/Isosurface#:~:text=An%20isosurface%20is%20a%203D,with% 20a%20given%20CoLocalization%20level.
7. “VisibleBreadcrumbs,” Mathworks.com, 2022. https://www.mathworks.com/help/matlab/visualize/exploring-volumes-with-slice-planes. html.
8. “VisibleBreadcrumbs,” Mathworks.com, 2022. https://www.mathworks.com/help/matlab/visualize/exploring-gradient-opacity. html.
9. “Research Imaging Institute — Mango,” Mangoviewer.com, 2022. http://mangoviewer.com/userguide.html.
10. M. E. Munns, C. He, A. Topete, and M. Hegarty, “Visualizing Cross-Sections of 3D Objects: Devel- oping Efficient Measures Using Item Response Theory,” Journal of intelligence, vol. 11, no. 11, pp. 205–205, Oct. 2023, doi: https://doi.org/ 10.3390/jintelligence11110205.
11. X. Huang, Q. Zhang, Y. Feng, H. Li, X. Wang, and Q. Wang, “HDR-NeRF: High Dynamic Range Neural Radiance Fields,” Thecvf.com, pp. 18398–18408, 2022, Accessed: May 09, 2024. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2022/html/Huang_HDR-NeRF_High_Dynamic_Range_Neural_Radiance_Fields_CVPR_2022_paper.html
12. Bao, Xueyi, Jun Han, and Chaoli Wang. "VolumeVisual: Design and evaluation of an educational software tool for teaching and learning volume visualization." 2021 ASEE Virtual Annual Conference Content Access. 2021.
13. Y. Shin, B.-S. Sohn, and H. Kye, “Acceleration techniques for cubic interpolation MIP volume rendering,” Multimedia tools and applications, vol. 80, no. 14, pp. 20971–20989, Mar. 2021, doi: https://doi.org/10.1007/ s11042-021-10642-4.
14. J. Li et al., “Hiplot: a comprehensive and easy- to-use web service for boosting publication- ready biomedical data visualization,” Briefings in bioinformatics, vol. 23, no. 4, Jul. 2022, doi: https://doi.org/10.1093/bib/bbac261.
15. K. M. Tabor et al., “Brain-wide cellular resolution imaging of Cre transgenic zebrafish lines for functional circuit-mapping,” eLife, vol. 8,

Feb. 2019, doi: https://doi.org/10.7554/elife.42687.

16. A. S. Mady and Samir Abou El-Seoud, “An Overview of Volume Rendering Techniques for Medical Imaging,” International journal of online and biomedical engineering, vol. 16, no. 06, pp. 95–95, May 2020, doi: https://doi.org/10.3991/ijoe.v16i06.13627.
17. Q. Zhang, “Web-based medical data visualization and information sharing towards application in distributed diagnosis,” Informat- ics in medicine unlocked, vol. 14, pp. 69–81, Jan. 2019, doi: https://doi.org/10.1016/j.imu.2018.10.010.
18. K. O. Lewis, V. Popov, and Syeda Sadia Fatima, “From static web to metaverse: reinventing medical education in the post-pandemic era,” Annals of medicine (Helsinki)/Annals of medicine, vol.56, no. 1, Jan. 2024, doi: https://doi.org/10.1080/07853890.2024.2305694.
19. Congote, “MEDX3DOM: MEDX3D for X3DOM.” Accessed: May 09, 2024. [Online]. Available: http://cadcamcae.eafit.edu.co/documents/draft_MEDX3DOM.pdf
20. Alper Sahistan et al., “Ray-traced Shell Traversal of Tetrahedral Meshes for Direct Volume Visualization,” Oct. 2021, doi: https://doi.org/10.1109/vis49827.2021.9623298.
21. Ayca Kirimtat and Ondrej Krejcar, “GPU-Based Parallel Processing Techniques for Enhanced Brain Magnetic Resonance Imaging Analysis: A Review of Recent Advances,” Sensors, vol. 24, no. 5, pp. 1591–1591, doi: https://doi.org/10.3390/s24051591.
22. Randolph, Jiyoun, "Blockchain-based Medical Image Sharing and Critical-result Notification" (2022). Master of Science in Information Technology Theses. 11. https://digitalcommons.kennesaw.edu/msit_etd/11
23. Zakaria Rguibi, Hajami Abdelmajid, and Dya Zitouni, “Deep Learning in Medical Imaging:A Review,” ResearchGate, Apr. 05, 2022. https://www.researchgate.net/publication/359754529_Deep_Learning_in_Medical_Imaging_A_Review.