The Framework of Inspection Layers of CT and MRI Human Brain Datasets by Bimanual Gesture Interaction

Yiyi Deng, Zeqing Fu, Xin Jia, Bin Gao, and Yanlin Luo[✉]

College of Information Science and Technology, Beijing Normal University, Beijing, China
luoyl@bun.edu.cn

Abstract. We proposed methods for 3D visualization and inspection of different layers corresponding to different tissues. First, we give pipeline of extraction the MRI human brain datasets. Then we present the design of trapezoid opacity transfer function and present the framework of visualization for the CT and segmented MRI datasets, based on CUDA-based real-time volume ray-casting. Furthermore, we inspect its inner layers via virtual lenses by intuitive bimanual gesture interaction, achieving the focus+context visualization. Finally, the performance of our framework verified effective by our experiments.

Keywords: CT (Computed Tomography) · MRI (Magnetic resonance imaging) · Volume visualization · CUDA- based volume ray-casting · Bimanual gesture interaction

1 Introduction

MRI and CT are two powerful techniques for assessing the brain. MRI can provide cross-sectional and longitudinal high-resolution images of soft tissue in brain with obvious contrast, while CT can display the cross sections which are perpendicular to the body long axis and present bone and muscle better. What's more, MRI is more time-consuming and costly than CT. Therefore, they differ in functions for medical study, and their datasets are always processed in different methods.

3D visualization and inspection of the human brain volume dataset is the further critical step, which is also essential to the modern medicine and its application. It helps users efficiently understand both the shape characteristic and the internal geometry structure of the volumetric datasets [1]. Currently, most studies for volume visualization focus on real-time rendering algorithm, which achieved particularly rapid development as the GPU innovation [2].

In volume rendering it's difficult to simultaneously visualize interior and exterior structures, and it's necessary to design layer inspection to show each organization individually. Rezk-Salama et al. put forward an opacity peel rendering algorithm basing on depth peeling [3]. This algorithm solved the problem of occlusion in volume data. Basing on this algorithm, Liang Ronghua et al. put forward a delamination volume rendering algorithm. Liang's algorithm improved the accuracy of stratification and got great draw peeling effects [4, 5].

In the meanwhile, rendering performance alone is not sufficient to ensure that users understand volume visualizations intuitively [6]. Actually providing effective interaction techniques to ensure that users understand volume visualizations intuitively has become a hot research topic [7]. There are two basic fashions for volume interaction. One employs a 2D or 3D controller device, the other is touchless fashion which allows the user to manipulate on the display with their bare hands. Touchless interface is a primary paradigm of natural user interface and quantities of application need it to support the specified scenario, by which users apperceive and interact with volume data in a more natural and intuitive way.

In this paper, we mainly make four contributions. First, we present method of the extraction of human brain MRI dataset. Second, we design the framework of visualization of MRI and CT respectively. Then, we propose layer inspection of the MRI and CT datasets by intuitive bimanual gesture interaction via virtual lenses. Finally, some experiments are conducted to analyze the differences of visualization and inspection results between MRI and CT. The paper is structured as follows. Section 2 describes our method. Section 3 shows our experiment result. Section 4 gives some conclusions and ideas of future work.

2   Methods

There are several differences of visualization and interaction methods between MRI and CT. The first one is that CT datasets don't need to be processed from NIFTI dataset into RAW dataset. According to their differences we propose the following methods to achieve our framework for two kinds of datasets.

2.1   The Extraction of MRI Human Brain Datasets

We apply the FSL (FMRIB Software Library v5.0) to automatically extract the brain tissues and divide the brain image. The extraction method consists of following four steps.

Step 1. Extract the brain from the whole dataset

We used automated brain extraction method for human brain MRI data by BET method (Brain Extraction Tool)in FSL, which deletes non-brain tissue from an image of the head after inputting T2 images.

Step 2. Segment the brain into different tissues

The extracted brain images are spatially normalized using FAST method (FMRIB's Automated Segmentation Tool), which segments the 3D image into different tissue types, whilst also correcting for spatial intensity variations. Thus the human brain was segmented into some tissues such as grey matter, white matter and CSF (cerebrospinal fluid). The density threshold of these brain tissues are overlapped in some intervals.

Step 3. Take linear transformation

By statistics of the three tissues, we get the density of cerebrospinal fluid ranging from 1 to 1892, the grey matter ranging from 16 to 1129 and the white matter ranging from 286 to 1361. Hence we took linear transform for the density values of three tissues with non-overlapping region. The density values of CSF remain in [1,1892], the grey matter mapped from [16,1129] to [1900,3013] and the white matter mapped from [286,1361] to [3020,4095].

Step 4. Merge

After the linear transformation, we merge the data of three tissues into a result of the brain and finally input the data into MIPAV by which converting the MRI volume NIFTI dataset to MRI volume RAW dataset.

2.2   Transfer Functions Designed for the CT and Segmented MRI Volume Datasets

In direct volume rendering, it's essential to show different part of volume dataset, by build a mapping relationship from voxel to the corresponding optical characteristics.

This critical step is implemented by transfer function, which might be simple slope, piecewise continuous function, or in any other forms. The transfer function affects the final quality of output image directly. Meanwhile, intuitive transfer function improves the efficiency of adjusting parameters.

In the early transfer functions, scalar threshold was used as the basis for dividing the materials, mapping to different color and opacity value. This early method is simple and particle, but doesn't perform well in boundary, because the scalar threshold of some different materials too close to be categorized. To make up for this imperfection of one-dimensional tracker function, some researchers proposed multi-dimensional transfer function, which is based on scalar value, gradient value, curvature, and spatial association information. However, as the dimension increasing, it can cause high complexity and heavy computation. So we propose an improved one-dimensional transfer function, a trapezoid-shaped function which has a better performance for layer control.

Transfer Function Designed for CT Datasets. In CT scalar datasets, materials can't be divided precisely only based on single scalar threshold, and there must be the scalar threshold coincidence of several data. As for the trapezoid transfer function, the trapezoidal waist corresponds to the overlapping threshold interval of multiple materials. To study the different expression in linear form in accordance with the actual distribution of material in the region is the key to solve the problem.

We described the transfer function as multiple trapezoid, and each one represents one classification such as boon, skin, and muscle. Assume that the number of categories is K, and the trapezoid is expressed as $B_j = [b_{j,0}, b_{j,1}, b_{j,2}, b_{j,3}]$ with color value C_j and maximum opacity A_j. In fact, there's an overlapping area between two neighboring materials, as we illustrated in Fig. 1.
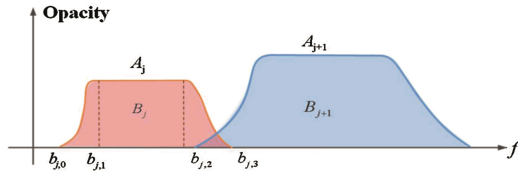
Fig. 1. Illustration of overlapping area

Take a ray V starting from viewpoint as an example, let f_i be the scalar value correspond to sampling point p_i on V, $fi \in Bm(m = 1, 2, \ldots, K-1)$, the weight w_i is defined by the following equation

$$
w_i = \begin{cases}
g\left(\dfrac{f_i - b_{m,0}}{b_{m,1} - b_{m,0}}\right), & f_i \in [b_{m,0}, b_{m,1}] \\
1, & f_i \in [b_{m,1}, b_{m,2}] \\
1 - g\left(\dfrac{f_i - b_{m,2}}{b_{m,3} - b_{m,2}}\right), & f_i \in [b_{m,2}, b_{m,3}]
\end{cases}
\tag{1}
$$

where g(t) is the cubic function of the value in [0,1], $g(t) = t^2(3 - 3t)$.

According to Eq. (1), we can get the opacity value α_i of p_i

$$
\alpha_i = \begin{cases}
g\left(\dfrac{f_i - b_{m,0}}{b_{m,1} - b_{m,0}}\right) A_m, & f_i \in [b_{m,0}, b_{m,1}] \\
A_m, & f_i \in [b_{m,1}, b_{m,2}] \\
\dfrac{\left[1 - g\left(\dfrac{f_i - b_{m,2}}{b_{m,3} - b_{m,2}}\right)\right] A_m + g\left(\dfrac{f_i - b_{m+1,0}}{b_{m+1,1} - b_{m+1,0}}\right) A_{m+1}}{A_m + A_{m+1}}, & f_i \in [b_{m,2}, b_{m,3}]
\end{cases}
\tag{2}
$$

where the last part is the opacity value of the overlapping $[b_{j,1}, b_{j,2}]$.

As supposed the color c_i of p_i can be got from the color of B_m, $c_i = \alpha_i T_m$, so we can get accumulated value of its opacity and color according to conventional ray-casting algorithm by the following equation

$$
\begin{cases}
c_i^* = c_{i-1}^* + (1 - \alpha_{i-1}^*)\alpha_i c_i \\
\alpha_i^* = \alpha_{i-1}^* + (1 - \alpha_{i-1}^*)\alpha_i
\end{cases}
\tag{3}
$$

where α_i^*, c_i^* are the accumulated opacity and color corresponding to one viewing ray through the pixels into the field volume implemented by one thread in CUDA.

Transfer Function Designed for the Segmented MRI Volume Datasets. For MRI datasets, we illustrated it in Fig. 2 in which one trapezoid expresses one band or layer for the segmented human brain MRI data. Each band has no overlapping region. There are three bands corresponding to CSF, the white matter and the grey matter.
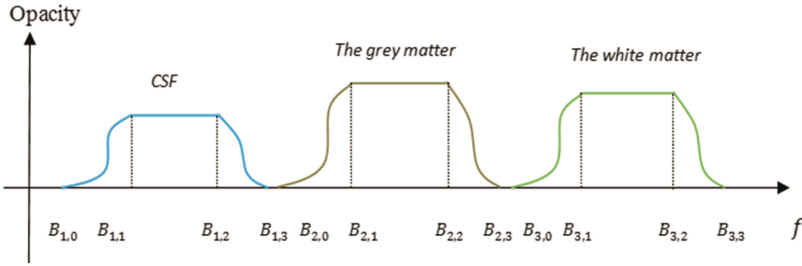
Fig. 2. Illustration of three bands

B_j is defined as above. Because different bands are corresponding to different separated tissues, so B_j and B_{j+1} $(j = 1, 2, \dots, K - 1)$ has no overlapping region. The opacity α_i of p_i is defined by the following equation

$$\alpha_i = \begin{cases} g\left(\dfrac{f_i - b_{m,0}}{b_{m,1} - b_{m,0}}\right)A_m, & f_i \in [b_{m,0}, b_{m,1}] \\ A_m, & f_i \in [b_{m,1}, b_{m,2}] \\ [1 - g\left(\dfrac{f_i - b_{m,2}}{b_{m,3} - b_{m,2}}\right)]A_m & f_i \in [b_{m,2}, b_{m,3}] \end{cases} \qquad (4)$$

We can get the accumulated opacity and color α_i^*, c_i^* according to conventional raycasting algorithm by the Eq. (3).

We use the above transfer function in which more details are shown to get better effects of volume rendering and effective layer control when performing volume exploration using the layer filter interaction tool.

2.3 The Framework for Visualization of CT and MRI Human Brain Datasets

The framework for visualization is described in Fig. 3. As for the MRI datasets, we process it as Sect. 2.1 and then input the volume datasets into data cache. As for the CT data, we can input the dataset into the data cache directly.

The framework for visualization is similar to our previous work depending on different transfer function design [8], and here we integrate the CT and segmented MRI datasets. It exploits the CUDA framework and the hierarchical structures such as octree for both compression of volume data and speed optimization of ray casting process. In the beginning of raycaster, we need to determine the thread index in CUDA, and then resample in the volume if it intersects with the viewing ray. Data mapping means from data properties to optical properties specifying the RGBA. We adjusted the opacity and color according the tools specification if it is inside the ROI, else take data mapping from our transfer function defined in Sect. 2.2. Phong illumination model is used for shading before accumulating the color. Because all threads run at the same time and process the data, the performance is greatly increased and real time.
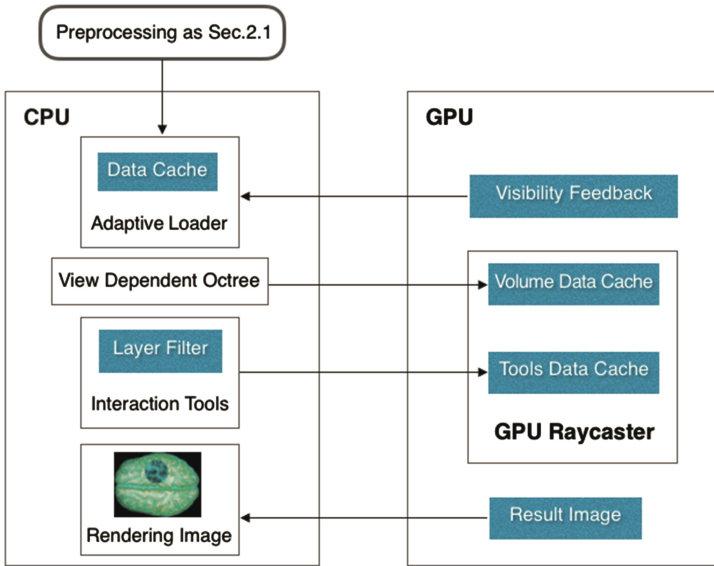
Fig. 3. The framework of visualization

2.4 Inspection by Bimanual Gesture Interaction via Virtual Lenses

In order to achieve better inspection of human brain structure by our visualization framework mentioned above, we design bimanual gesture to manipulate the volume model with layer filter tool.

Layer Filter Tool. For facilitating inspection of interior and exterior tissues, we define the virtual lenses as layer filter for 2D inspection. According to above simplified model, the volume dataset is divided into two parts including 3D ROI described as shaded parts in Fig. 4 which is the intersection of a cone with the window of virtual lenses as base and the viewpoint, and the whole volume datasets. In ROI, we modify the opacity α_i of p_i in (0, 1] for selected band and 0 for others bands before compositing as Eq. (3). Outside ROI and inside the view frustum the conventional CUDA-based volume rendering is used.
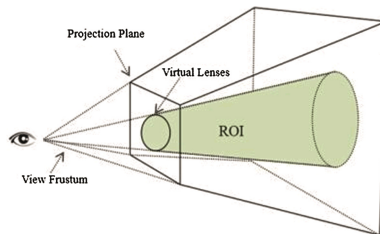

Fig. 4. Diagram of virtual lenses

Bimanual Gesture Design. In order to simultaneously visualize interior and exterior structures and get the region of interest, the layer filter and intuitive hand gestures are need to manipulate the human brain volume datasets directly. We develop natural and intuitive exploring fashion. The concept of natural means various, intuitive, memorable and practical, which is argued through mimicking aspects of the real world or drawing on our existing tendencies in the areas of communicative, gesticulate, deictic, and manipulative behaviors and actions [9].

We propose the bimanual gesture for operating human brain volume model and the interactive tool simultaneously. Using an open hand pose as shown in Fig. 5(a). The user can rotate and move the left hand as the non-dominant hand with natural hand pose to achieve a 6DOF motion of the human brain volume model. The right hand is regarded as the dominant hand to control an interactive tool as shown in Fig. 5(b), and can change the visible layer by moving in a circular motion. When interacting with both hands, the working space is limited, so finger movements are considered as much as possible to achieve the manipulation of the volume datasets.

(a)   (b)

Fig. 5. Bimanual gesture interaction for both volume model and layer filter. (a) Natural open bending pose for 6DOF manipulating the volume model. (b) Point pose to control the layer filter

3 Experiment Result

We get the human brain CT and MRI T1 imaging using a Siemens Trio Tim scanner at Beijing MRI Center for Brain Research. It is a $168 \times 206 \times 128$ with 16 bit volumetric dataset. We have implemented our algorithm on an Intel Core 2 Quad CPU Q9400 @ 2.66 GHz equipped with 4 GB of RAM on Windows 7 using OpenGL and CUDA 3.0. The graphics card is a NVIDIA GeForce GTX 760 with 1 GB of RAM memory. We use a Leap Motion sensor to track hand motion data and provide 3D object coordinates with millimeter accuracy at 120 Hz.The system is written in C++ using SDK V2.3.1 for Windows provided by Leap Corporation and uses CUDA parallel architecture to accelerate calculation for real-time rendering of the volume dataset.We can maintain an interactive frame rate with an average of 100 fps.

Figure 6 are the results of using virtual lenses for inspection of different layers of CT datasets in (a) bones, (b) muscle, (c) skin of the brain structure (d) in ROI. Figure 7 are results of using virtual lenses for inspection of different layers of MRI human brain in (a) CSF, (b) the grey matter, (c) the white matter and (d) the image of brain with three tissues.
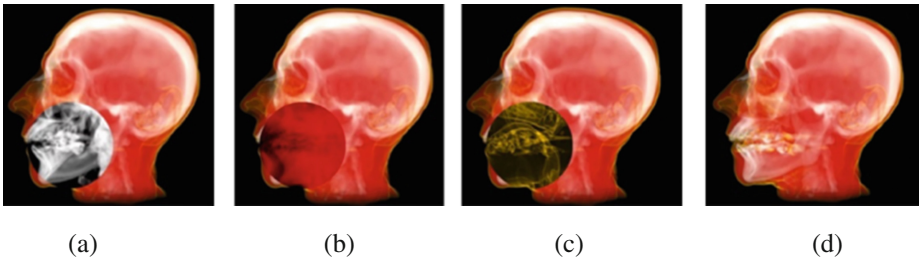
(a)   (b)   (c)   (d)

Fig. 6. Virtual Lenses for inspection of different layers of CT dataset

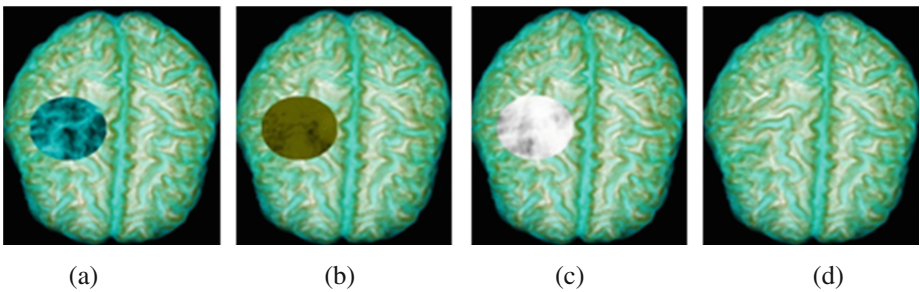(a)   (b)   (c)   (d)

Fig. 7. Virtual Lenses for inspection of different layers of MRI dataset

We use our designed bimanual gestures as shown in Fig. 5 to operate the volume model and the interactive tool simultaneously by Leap Motion sensor, with two different cameras for touch-free motion sensing. The Leap Motion system detects and tracks hands and fingers. The device operates in an intimate proximity with high precision and tracking frame rate. But it cannot identify a specific static gesture directly. As shown in Fig. 8, we used a spot as a metaphor of layer filter in order to explore different structures of volume dataset. A point pose (see Fig. 5(b)), with index finger pointing to the screen, provides a convenience greatly to using the tool. The user can change the position and the size of this

Fig. 8. Bimanual gesture interaction with the MRI human brain volume model

region by moving index finger in the physical space. In addition, our system can achieve different layer exploration as shown in Figs. 6 and 7(a)–(c) by moving index finger in a circular motion, based on our transfer function design described in Sect. 2.2.

4 Conclusions and Future Work

We combine the human MRI brain extraction and the trapezoid functions with non-overlapping region to display its different layers. The segmented MRI dataset and non-segmented CT can be efficiently and easily adapted to CUDA-based volume ray-casting in our framework. The results strongly depend on our trapezoid opacity transfer function design, in which the different trapezoid functions are designed. By intuitive bimanual gesture interaction via virtual lenses, the inner layers are inspected in focus region with semi-transparent out parts by focus+context visualization.

Although the original data we used are not from the same person, our methods and framework can be applied to CT and MRI datasets from the same person. One of our goals in future is to extend to our methods to other MRI and CT volume dataset with more bands. Also we hope to expand to projection based virtual environments toward a more natural interface for cooperative inspection supporting multi-modal interaction and multi-user.

References

1. Gallo, L.: A study on the degrees of freedom in touchless interaction. In: ACM SIGGRAPH Asia Technical Briefs, pp. 1–4 (2013)
2. Gobbetti, E., Marton, F., Iglesias-Guitián, J.A.: A single-pass GPU ray casting framework for interactive out-of-core rendering of massive volumetric datasets. Vis. Comput. **24**(7–9), 797–806 (2008)
3. Rezk-Salama, C., Kolb, A.: Opacity peeling for direct volume rendering. Comput. Graph. Forum **25**(3), 597–606 (2006)
4. Liang, R.H., Li, C., Wu, F.L., et al.: Layered peeling algorithm of direct volume rendering for medical dataset. J. Compu.-Aided Des. Comput. Graph. **21**(10), 1381–1386 (2009). (in Chinese)
5. Wu, F.L., Huang, X.P., Liang, R.H., et al.: Opacity peeling for medical data based on volume membership. J. Comput.-Aided Des. Comput. Graph. **22**(10), 1810–1816 (2010). (in Chinese)
6. Laha, B., Sensharma, K., Schiffbauer, J.D., Bowman, D.A.: Effects of immersion on visual analysis of volume data. IEEE Trans. Vis. Comput. Graph. **18**(4), 597–606 (2012)
7. Hanqi, G., Ningyu, M., Xiaoru, Y.: WYSIWYG (What You See is What You Get) volume visualization. IEEE Trans. Vis. Comput. Graph. **17**(12), 2106–2114 (2011)
8. Luo, Y.L.: Distance-based focus+context models for exploring large volumetric medical datasets. IEEE Comput. Sci. Eng. **14**(5), 63–71 (2012)
9. Wigdor, D., Wixon, D.: Brave NUI World: Designing Natural User Interfaces for Touch and Gesture, pp. 65–72. Elsevier, Amsterdam (2011)