



# A VR-based volumetric medical image segmentation and visualization system with natural human interaction

Yi Gao<sup>1,2,3,4</sup> · Cheng Chang<sup>5</sup> · Xiaxia Yu<sup>1</sup> · Pengjin Pang<sup>1</sup> · Nian Xiong<sup>6,7</sup> · Chuan Huang<sup>8</sup>

Received: 8 September 2020 / Accepted: 31 August 2021 / Published online: 14 September 2021  
© The Author(s) 2021

## Abstract

Volume rendering produces informative two-dimensional (2D) images from a 3-dimensional (3D) volume. It highlights the region of interest and facilitates a good comprehension of the entire data set. However, volume rendering faces a few challenges. First, a high-dimensional transfer function is usually required to differentiate the target from its neighboring objects with subtle variance. Unfortunately, designing such a transfer function is a strenuously trial-and-error process. Second, manipulating/visualizing a 3D volume with a traditional 2D input/output device suffers dimensional limitations. To address all the challenges, we design NUI-VR<sup>2</sup>, a natural user interface-enabled volume rendering system in the virtual reality space. NUI-VR<sup>2</sup> marries volume rendering and interactive image segmentation. It transforms the original volume into a probability map with image segmentation. A simple linear transfer function will highlight the target well in the probability map. More importantly, we set the entire image segmentation and volume rendering pipeline in an immersive virtual reality environment with a natural user interface. NUI-VR<sup>2</sup> eliminates the dimensional limitations in manipulating and perceiving 3D volumes and dramatically improves the user experience.

**Keywords** Virtual reality · Natural user interaction · Medical image segmentation · Volume rendering · Transfer function

## 1 Introduction

Direct volume rendering generates informative two-dimensional (2D) images from three-dimensional (3D) volumes by directly mapping voxel values to optical properties with a transfer function. To highlight a target, users need to design a transfer function so that voxels within the target area carry distinctly different optical properties from those outside.

This process requires tremendous user design efforts, especially when the region of interest is hard to differentiate with neighboring objects. Image segmentation is a highly related field regarding extracting a target. However, utilizing image segmentation techniques to facilitate the volume rendering process has not been well studied. For exploring 3D volumes, there are intrinsic limitations to use a traditional 2D input device, a mouse, or a keyboard, to label and interact

✉ Cheng Chang  
cheng.chang@alumni.stonybrook.edu

Yi Gao  
gaoyi@szu.edu.cn

Xiaxia Yu  
xiaxiayu@szu.edu.cn

Pengjin Pang  
2016222064@email.szu.edu.cn

Nian Xiong  
nianxiong@hust.edu.cn

Chuan Huang  
chuan.huang@stonybrookmedicine.edu

<sup>2</sup> Shenzhen Key Laboratory of Precision Medicine for Hematological Malignancies, Shenzhen 518060, China

<sup>3</sup> Marshall Laboratory of Biomedical Engineering, Shenzhen 518060, China

<sup>4</sup> Pengcheng Laboratory, Shenzhen 518060, China

<sup>5</sup> Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794, USA

<sup>6</sup> Department of Neurology, Union Hospital, Tongji Medical College, Huazhong University of Science and Technology, Wuhan 430074, Hubei, China

<sup>7</sup> Department of Neurology, Wuhan Red Cross Hospital, Wuhan 430015, Hubei, China

<sup>8</sup> Department of Radiology, Stony Brook University, Stony Brook, NY 11794, USA

<sup>1</sup> School of Biomedical Engineering, Health Science Center, Shenzhen University, Shenzhen 518060, China

with 3D volumes. The dimensional discrepancy imposes heavy burdens on users as they need to escalate 2D actions into 3D effects subjectively. Additionally, perceiving the entire 3D dataset on a 2D surface is infeasible. We need a more effective way to observe the data. To address the challenges above, we design and implement NUI-VR<sup>2</sup>, a natural user interface-enabled volume rendering system in the virtual reality space. In NUI-VR<sup>2</sup>, users inspect the 3D volume in a VR environment and specify a few seeds within the target with intuitive gestures and voice commands. With those seeds, image segmentation converts the original volume into a probability volume where voxels in the target yield higher values. A simple linear transfer function will highlight the target well. Users can explore the rendered volume with NUI inside an immersive VR environment. In summary, our main contribution is threefold:

- Propose a generic strategy for integrating image segmentation and volume rendering. Image segmentation and feature selection techniques instead of high-dimensional transfer functions are applied to highlight the target.
- Design and implement a novel end-to-end volume interaction, image segmentation, and volume rendering system in VR.
- Develop an attention-based NUI for the VR environment with unlimited gestures and voice commands.

## 2 Related work

Transfer function design has been the focus of many volume rendering researchers (Pfister et al. 2001; Arens and Domik 2010; Ljung et al. 2016; Mady and Abou El-Seoud 2020). Initially, researchers assign optical properties to voxels based on their intensity values (He et al. 1996; Bajaj et al. 1997; Sabella 1988; König and Gröller 2001). Levoy (1988) first added the local grayscale gradient to the mapping process to isolate objects with similar intensities. Other data features, e.g., curvature (Kindlmann et al. 2003), texture (Caban and Rheingans 2008), and distance (Tappenbeck et al. 2006) information were also included later on. The difficulty in identifying a proper transfer function increases as the dimensionality of the features expands. Tzeng et al. (2005) proposed a smart volume rendering system. They trained a model with the user's inputs and used it to classify the volume to eventually simplify the transfer functions. Topology-based (Takeshima et al. 2005; Weber et al. 2007) and domain-specific (Tiede et al. 1998) segmentations were also applied to divide the original volume into sub-volumes to achieve a similar goal. ImageVis3D (Fogal and Krüger 2010) is a powerful transfer function design software with an intuitive user interface. All of them help design a transfer

function, but none of them frees users from the daunting process.

On the image segmentation side, researchers proposed numerous innovative algorithms (Zhu and Yuille 1996; Gao et al. 2012; Bali and Singh 2015; Kuruvilla et al. 2016) over the years. They usually require some user interactions to guarantee accurate segmentation results. Such user interactions include specifying sample seeds inside the target (Boykov and Jolly 2001; Vezhnevets and Konouchine 2005; Karasev et al. 2013) or boundary masks outside (Mortensen and Barrett 1998). If the sample seeds or boundary masks are not well-defined, which is often the case with a traditional 2D input device, segmentation leakages may occur. Researchers have been trying to improve the user interactions with Microsoft Kinect. Kinect-based interfaces effectively alleviated the dimensional discrepancy between the user space and the 3D data (Wang and Jung 2017; Ju et al. 2018).

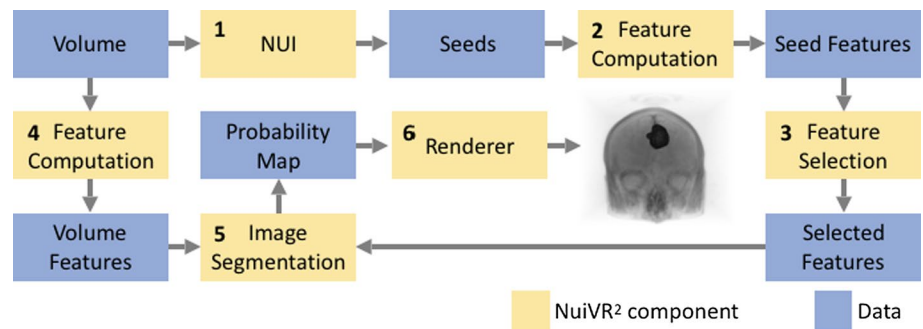
Researchers have been trying to utilize VR as the visualization media to visualize volumetric data. At first, Hänel et al. (2016) used a theatre-like system consisting of a room-sized cube and projectors. However, the high setup requirement prohibits its wide adoption. Recently, portable VR technologies, from simple cardboard inserts for smartphones to sophisticated VR headsets with accurate tracking sensors, became widely available (El Beheiry et al. 2019). The advancement of VR technologies led to flourishing research to exploit portable VR to visualize volumetric data (Cohen et al. 2013; Chan et al. 2013; Faludi et al. 2019), and the results have been promising because of the immersive environment. As VR improves user experience in volume exploration and visualization, we set up our proposed system entirely in VR.

## 3 Methods

NUI-VR<sup>2</sup> is set up in the VR space. With the six-degree-of-freedom positional tracking capability of the current VR headsets, e.g., Oculus Rift, users can virtually interact with the volumes. As reported in Hänel et al. (2016), users are more motivated to explore the data in VR because of the immersive experience. We take advantage of the portable VR headset, Oculus Rift, instead of the enormous theatre-like environment as in Hänel et al. (2016) to make NUI-VR<sup>2</sup> more accessible. The typical rendering process for an  $m \times n \times k$  volume is numbered in Fig. 1 and can be summarized as below:

1. Users browse through the volume with voice commands and gestures. Once users locate the target, they could record a few seeds within the target and generate all kinds of 3D masks outside the target. Those seeds and

**Fig. 1** The NUI-VR<sup>2</sup> workflow overview. Blue boxes represent the data, whereas yellow boxes represent the components of NUI-VR<sup>2</sup>. The numbers indicate the workflow sequence



masks serve as the inputs for various segmentation algorithms.

2. Assume users record  $s$  seeds, and we compute  $f$  predefined features for each seed, we will get an  $s \times f$  seed feature matrix after the feature computation step.
3. The feature selection process selects  $r$  features out of the  $f$  features and reduces the size of the seed feature matrix to  $s \times r$ .
4. We compute the  $r$  features for the entire volume and obtain an  $m \times n \times k \times r$  feature volume.
5. With the seed matrix, the feature volume, and optionally the boundary masks, we can apply a wide range of image segmentation techniques to generate an  $m \times n \times k$  probability volume. The target region will carry larger values than the others in the probability volume.
6. Finally, we render the probability volume with a simple linear transfer function to highlight the target. Figure 1 shows a rendering of a human head with a tumor highlighted as an example.

### 3.1 Natural user interface

One challenge for users to explore 3D volumes, especially in VR, is the lack of an efficient input device. Kinect recognizes voice commands and tracks 3D locations of multiple human joints, including fingertips. We designed a NUI system with Kinect to reduce the dimensional discrepancy. It enables users to generate seeds and boundary masks directly in the 3D space effortlessly.

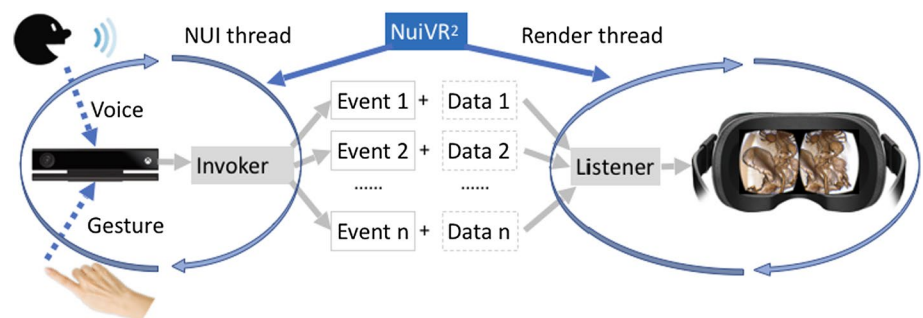
#### 3.1.1 NUI system overview

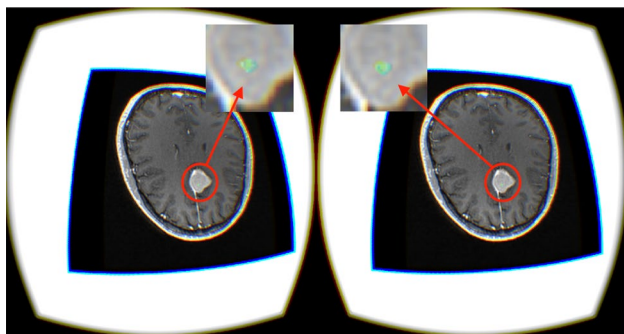
Figure 2 shows an overview of NUI-VR<sup>2</sup> with an emphasis on the NUI system. The NUI system runs on an individual thread separate from the render thread and detects voice commands and gestures. Once an event is detected, the NUI thread sends the event and some optional metadata to the render thread. The render thread then updates the volume rendering as instructed by the event. For example, when users move their left-hand tips, the NUI thread sends the LEFT-HAND-TIP-MOVED event with the position info to the render thread. The render thread then fetches the new position data and repositions the rendering.

#### 3.1.2 3D image browsing

We use the 3D position of the left-hand tip to browse volume slices. To aid further discussions, we define the Kinect origin as  $\mathbf{o} \in \mathbb{R}^3$  and the direction where Kinect is facing as a unit vector  $\mathbf{p}$ .  $\tilde{d}(t)$ , the distance of the left-hand tip to  $\mathbf{o}$  in the  $\mathbf{p}$  direction at time  $t$ , is used to extract a slice from the volume. We assume the volume resides in the center of the Kinect space with the viewing plane perpendicular to  $\mathbf{p}$  and map the volume evenly to a valid range. The first (last, resp.) slice maps to the minimum (maximum, resp.) value within the range. Thus, a slice can be picked with  $\tilde{d}(t)$ . However,  $\tilde{d}(t)$  is noisy and may cause jitterings. To address this issue, we use recursive filtering:

**Fig. 2** NUI-VR<sup>2</sup> system overview





**Fig. 3** Viewing a 2D slice in a 3D MR human head volume in NUI-VR<sup>2</sup>

$$d(t_0) = \tilde{d}(t_0) \quad (1)$$

$$d(t_n) = \alpha \tilde{d}(t_n) + (1 - \alpha)d(t_{n-1}) \quad (2)$$

where  $\alpha$  ( $0 < \alpha < 1$ ) is the smoothing factor.  $d(t_n)$  will be more stable than  $\tilde{d}(t)$ . Figure 3 shows an example of browsing a 3D human brain in NUI-VR<sup>2</sup>. The distance of the user's left-hand tip to  $\mathbf{o}$  in the viewing plane direction controls the positions of the green dots. Users can issue an “insert” voice command to add them to the seed set.

### 3.1.3 3D surface generation

We track and smooth the joints on the arm and hands with Kinect in our NUI. Those joints can form a closed spatial curve. By sweeping and recording the curves over time, we can generate various 3D surfaces:

**3.1.3.1 Surface from polygon** We track a total of  $m$  joints  $\{P_i\}$  from the left-hand tip ( $P_0$ ) to the right-hand tip ( $P_{m-1}$ ), where  $m$  ranges from 3 to 11.  $P_0$  still controls the browsing of the volume, as detailed in the previous section. We use the orthogonal components of  $P_i$  to create polygons on the current viewing plane. Given  $P_0$  and its normal direction  $\mathbf{p}$ , the vector from the Kinect origin  $\mathbf{o}$  to the plane can be denoted as  $\mathbf{v} := \langle P_0, \mathbf{p} \rangle \mathbf{p}$ , where  $\langle \cdot, \cdot \rangle$  indicates the inner product.  $P_i$  is then projected to the plane through  $Q_i := P_i - \langle P_i - \mathbf{v}, \mathbf{p} \rangle \mathbf{p}$ . The loci of  $Q_i, Q_i(t) : i = 1, \dots, m$ , form a continuous 3D surface and controls the shape of the polygon in the slice.

**3.1.3.2 Surface from circles** Instead of polygons, users can also use circles with varying centers and radii to form 3D surfaces. Only two joints, the left-hand tip, and the right-hand tip, are tracked. Since the left-hand tip is always in the viewing plane as designed, we only need to project the right-hand tip to the plane. The line segment in-between

defines the circle's diameter. The sweeping of circles constructs a 3D surface.

**3.1.3.3 Spatial curve from points** We only track the left-hand tip. Its 3D positions construct a continuous spatial curve. While moving the left hand, the viewing slice also follows. The recorded curve helps mark a target for image segmentation algorithms.

### 3.1.4 Voice control

Kinect features a multi-array microphone and recognizes predefined voice commands from the input audio stream with the Microsoft speech application programming interface. We integrated voice commands in the volume rendering pipeline, so that when users explore 3D volumes, they can interact with the rendering process with their voice commands without worrying about physical limitations. Table 1 lists some voice commands in NUI-VR<sup>2</sup>. We can easily add more voice commands to our NUI system.

## 3.2 Interactive image segmentation

Interactive image segmentations enable users to specify seeds and masks that serve as strong hints for extracting the target, e.g., the target shall include all the seeds and shall not leak through the masks. We provide a generic strategy to leverage interactive image segmentation in volume rendering. The image segmentation algorithm converts the original  $m \times n \times k$  volume into an  $m \times n \times k$  probability volume, where voxels in the target area carry larger values as they share similar feature values with the seeds. By allowing users to specify numerous advanced image segmentation algorithms (Gao et al. 2010, 2012; Zhu et al. 2014; Gao et al. 2010; Chang et al. 2018), we provide unlimited optimization opportunities to NUI-VR<sup>2</sup>.

For simplicity, we use kernel density estimation (KDE) with a Gaussian kernel (Terrell and Scott 1992) as an example. KDE is the default image segmentation engine in

**Table 1** A few voice command examples in NUI-VR<sup>2</sup>

Commands	Actions
Scan	Switch to image browsing mode
Insert	Add a voxel to the set of seeds
Circle/Polygon	Construct a 3D surface with circles or polygons
Render	Render the probability volume
Preset i	Use the $i$ th transfer function preset
Rotate X/Y/Z	Rotate the volume along the X/Y/Z-axis
Up/Down/Left/Right	Move the volume up/down/left/right

NUI-VR<sup>2</sup>. The kernel density estimator  $f(\mathbf{v})$  is the summation of  $s$  discrete multivariate Gaussians centered at  $\mathbf{c}_i$ :

$$f(\mathbf{v}) := \sum_{i=1}^s e^{-\|\mathbf{v}-\mathbf{c}_i\|^2/\alpha^2} \quad (3)$$

where  $\mathbf{v} \in \mathbb{R}^r$  is a volume voxel to be estimated,  $r$  is the number of features,  $\mathbf{c}_i \in \mathbb{R}^r$  (for  $i = 1, \dots, s$ ) are the seeds, and  $\alpha$  is the bandwidth of the Gaussian kernel.  $f(\mathbf{v})$  represents the possibility that voxel  $\mathbf{v}$  falls within the same group with  $\mathbf{c}_i$  (for  $i = 1, \dots, s$ ). A linear transfer function will emphasize the target very well in the final probability volume.

### 3.3 Feature selection

NUI-VR<sup>2</sup> provides a feature set that consists of the 3D spatial location, intensity, and texture features (Vimort and McCormick 2017), including energy, entropy, correlation, difference moment (DM), inertia, cluster shade (CS), cluster prominence (CP), Haralick's correlation (HC), short run emphasis (SRE), long run emphasis (LRE), gray level non-uniformity (GLNU), run length non-uniformity (RLNU), low gray level run emphasis (LGLRE), and high gray level run emphasis (HGLRE). One significant advantage of using NUI-VR<sup>2</sup> is that users could supply additional features to expand the default feature set.

However, different targets may share different sets of descriptive features. For example, points on a vertical line in a 2D Cartesian coordinate system bear the same  $x$  coordinate. It will yield the most accurate result to characterize them only with their  $x$  values instead of their 2D coordinates. It is time-consuming and may lead to decreased accuracy if all of the features are used to characterize the target (Caban and Rheingans 2008). Thus, we add a feature selection process in NUI-VR<sup>2</sup>. The default feature selection algorithm is based on the  $\ell_1$ -norm support vector machine (SVM). SVM is a supervised machine learning algorithm to find the optimal hyperplane for the classification problem Zhu et al. (2004). Given a set of  $n$  labeled training data  $\{(\mathbf{x}_i, y_i)\}_1^n$  with  $\mathbf{x}_i \in \mathbb{R}^r$  being the training data and  $y_i \in \{-1, 1\}$  being the label, the  $\ell_1$ -norm SVM tries to solve the following optimization problem:

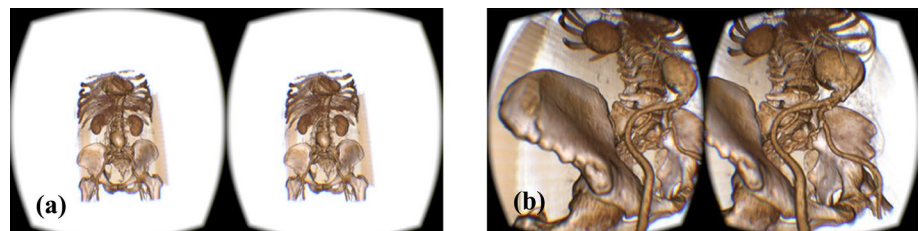
$$\min_{b, \mathbf{w}} \sum_{i=1}^n (1 - y_i(b + \mathbf{w} \cdot \varphi(\mathbf{x}_i))) + \lambda \|\mathbf{w}\|_1 \quad (4)$$

$\varphi: \mathbb{R}^r \rightarrow \mathbb{R}^q$  is the kernel function that maps  $\mathbf{x}$  from the original  $r$ -dimensional space to a  $q$ -dimensional space, where  $\mathbf{x}$  will be easier to be separated by a hyperplane.  $\mathbf{w}$  represents the hyperplane, and  $b$  is the bias.  $\lambda$  is the penalty parameter that controls the sparsity of  $\mathbf{w}$ . All training data carries a label of 1 or  $-1$  depending on whether or not they are within the target. We retain  $\mathbf{x}$  in the original space, i.e.,  $r = q$ , so that  $\mathbf{w}$  corresponds to the feature set. The solution to the above optimization problem leads to a sparse  $\mathbf{w}$ . We only select features with nonzero  $\mathbf{w}$  values as they play a major role in discriminating the two groups of training data. By varying  $\lambda$ , we can control the number of selected features.

### 3.4 Virtual reality

Using VR to explore 3D volumes has big potentials (El Beheiry et al. 2019; Cohen et al. 2013; Chan et al. 2013; Faludi et al. 2019). Coupled with our touch-less NUI, VR will undoubtedly make NUI-VR<sup>2</sup> more innovative, efficient, and enjoyable because of its realistic and immersive nature. There are various kinds of consumer VR devices, including cardboard viewer (e.g., Google Cardboard), mobile device mount (e.g., Samsung Gear VR), standalone VR (e.g., Oculus Quest), and tethered VR (e.g., Oculus Rift). We decided to build our system with Oculus Rift because it offers the best performance as we can connect the VR headset with a powerful computer. We use the Visualization Toolkit (VTK) Schroeder et al. (2004) as the rendering engine. VTK supports Oculus Rift natively and provides a rich feature set related to volume rendering. Figure 4 shows the rendering result of a  $512 \times 512 \times 288$  abdominal CT volume in NUI-VR<sup>2</sup>. A predefined transfer function is applied. Both figures mirror the entire display in Oculus Rift after chromatic aberration and lens distortion corrections. In Fig. 4a, users can have an overall view of the data, whereas in Fig. 4b, users can have a closer look at the internal structure of the volume. The rendering results illustrate that NUI-VR<sup>2</sup> can deliver high-quality images in the VR space just as in a traditional desktop setup.

**Fig. 4** Render an abdominal CT volume with NUI-VR<sup>2</sup> with a predefined transfer function. **a** Render the volume from outside; **b** render from inside the volume



### 3.5 System evaluation

As far as we know, there is no comparable image segmentation and volume rendering system as NUI-VR<sup>2</sup>, so it is hard to perform direct system-to-system comparisons. In Sect. 3.4, we have qualitatively shown the superb rendering quality of NUI-VR<sup>2</sup> in the VR space. In the next section, we will comprehensively evaluate NUI-VR<sup>2</sup> from the other perspectives:

- First, we compare the NUI system with a traditional mouse to show that the NUI system ensures higher image segmentation accuracy than a mouse.
- Next, we illustrate the effectiveness of image segmentation and feature selection in volume rendering. Users can easily adjust the rendering results by changing their parameters.
- Finally, we compare NUI-VR<sup>2</sup> with ImageVis3D, the leading software in transfer function design. The results show that NUI-VR<sup>2</sup> highlights targets better and requires less effort from users than ImageVis3D.

## 4 Results

### 4.1 Evaluation of NUI

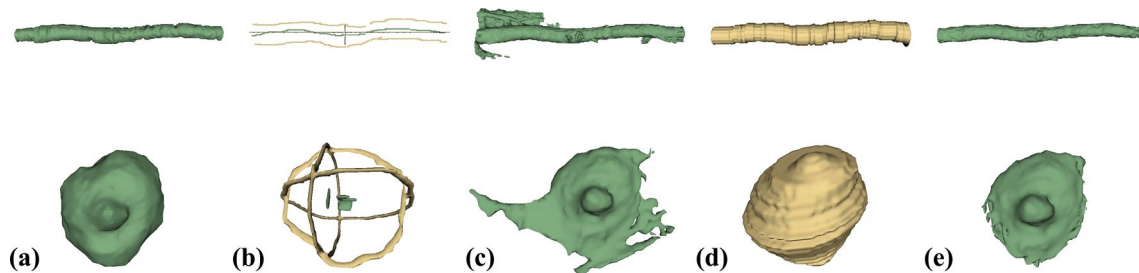
In this section, we qualitatively and quantifiably evaluate the NUI system. In NUI-VR<sup>2</sup>, the interactive image segmentation result plays a vital role in the rendering quality. We select Shortcut (Zhu et al. 2014) as the image

segmentation algorithm that requires a bounding surface outside the target as the initialization. Within the same amount of time, we use a traditional mouse and our NUI to define the bounding surfaces. Figure 5 shows that we can only define some sparse strips outside the targets with a mouse, and segmentation leakages occur. In comparison, we can swap closed 3D surfaces outside the targets with gestures using our NUI. There are no segmentation leakages because of the well-defined surfaces. Table 2 shows the dice coefficients and the Hausdorff distances of the segmentation results. NUI ensures consistently more accurate results than the mouse. Both the qualitative and quantifiable results show the superb efficiency and effectiveness of the NUI system.

### 4.2 Effectiveness of image segmentation and feature selection

We do not rely on transfer functions to adjust the volume rendering effects. Instead, we modify the parameters for the image segmentation and feature selection algorithms. This section will show the effectiveness of our proposed approach and how to change the rendering results in NUI-VR<sup>2</sup>.

We first use the 3D spatial location, the intensity, and eight texture features to highlight the cerebral cortex. To compute those texture features, we calculate them over the entire voxel intensity range of the volume, set a voxel intensity bin at each intensity level, and average their values in all the 13 directions. The only parameter left is the neighborhood radius size  $N$ . We use KDE with the Gaussian kernel as the image segmentation algorithm, and two hundred seeds



**Fig. 5** Image segmentation result comparison between a mouse and our NUI. (top) Vessel segmentation; (bottom) brain tumor segmentation; **a** ground truth of targets; **b** initializations with the mouse; **c**

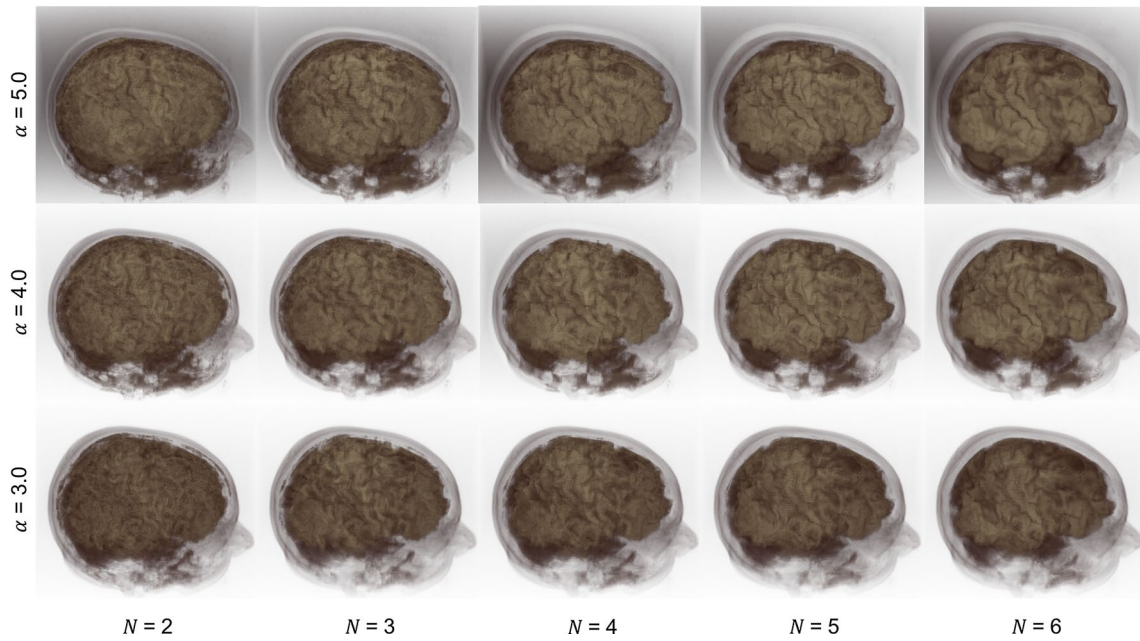
segmentation results with the mouse; **d** initializations with our NUI; **e** segmentation results with our NUI

**Table 2** Dice coefficients (the larger the better) and Hausdorff distances (the smaller the better) result comparison

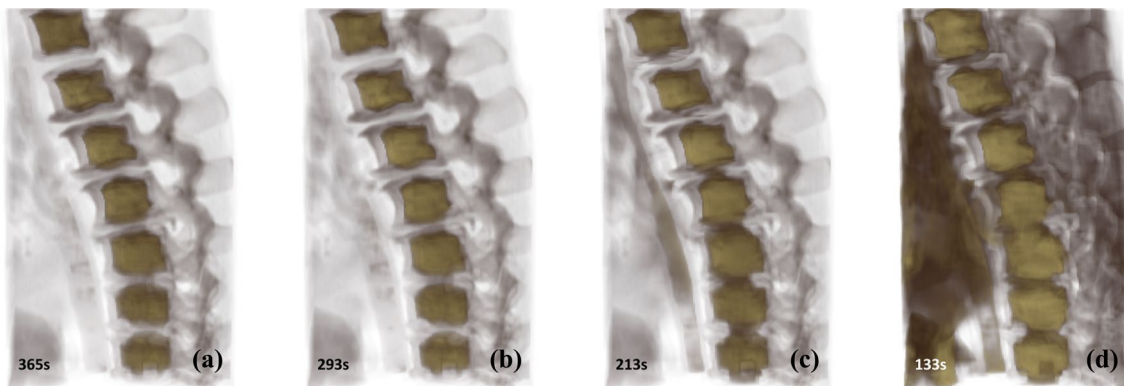
Measurements	Vessel	Tumor 1	Tumor 2	Tumor 3	Tumor 4
Dice coefficient (mouse)	0.807	0.922	0.691	0.853	0.942
Dice coefficient (NUI)	0.88	0.951	0.84	0.934	0.944
Hausdorff distance (mouse)	30.7	28.4	51.9	16.2	5.2
Hausdorff distance (NUI)	7.0	3.46	11.0	12.1	4.7

are selected within the target. The bandwidth of the Gaussian kernel  $\alpha$  dominates the KDE calculation. Figure 6 shows the different rendering results with different  $N$  and  $\alpha$  values.

The cerebral cortex is well highlighted. The rendering effects can be adjusted by simply varying those two parameters. On the contrary, it would be pretty hard to highlight the cerebral



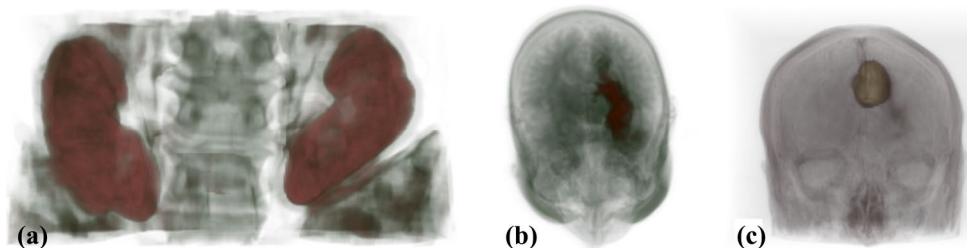
**Fig. 6** The cerebral cortex is well highlighted in all cases. The rendering quality can be adjusted by tuning  $N$  and  $\alpha$

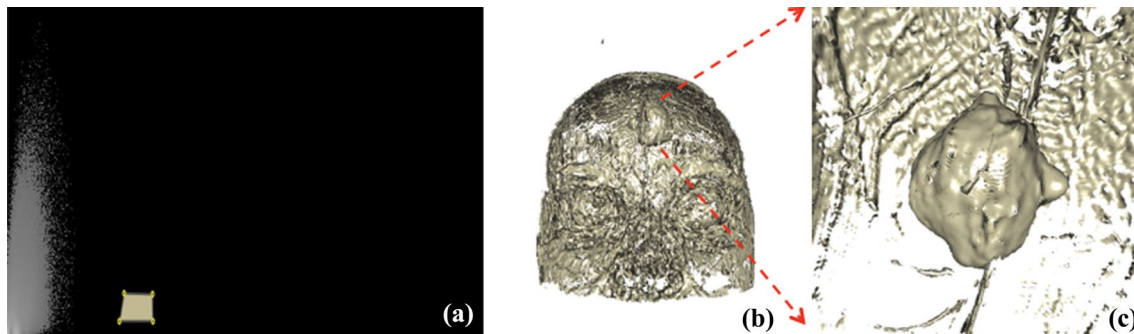


**Fig. 7** Rendering of the spine with different numbers of selected features. **a** With all the 18 features; **b** with correlation, SRE, LRE, and LGLRE removed; **c** with correlation, SRE, LRE, LGLRE, energy,

entropy, IDM, and inertia removed; **d** with only intensity, CP, HC, GLNU, RLNU, and HGLRE

**Fig. 8** Target highlighting in NUI-VR<sup>2</sup>. **a** kidneys; **b** the left hippocampus; **c** a brain tumor





**Fig. 9** Highlight the brain tumor with ImageVis3D. **a** The carefully handcrafted 2D transfer function; **b** the rendering result. The tumor buried inside is barely visible from outside the human brain; **c** The zoom-in view around the tumor area

cortex with a transfer function based on these 12 features, i.e., designing a 12-dimensional transfer function.

Next, we illustrate the effectiveness of feature selection. We use all of the 18 features in the default feature set, and perform a feature selection process as detailed in Sect. 3.3. Figure 7 shows the rendering result of a  $86 \times 142 \times 240$  CT volume with different numbers of selected features. The time to compute the probability volume on a commodity laptop is labeled. As we increase  $\lambda$  in Eq. 4, we increase the penalty for non-sparse  $w$  in the optimization, so the resulting  $w$  becomes more sparse and has more 0 elements. As a result, fewer features get selected. Feature selection effectively filters out redundant features and speeds up the volume rendering process. How many features should be selected depends on the feature set and the data. In NUI-VR<sup>2</sup>, users can adjust  $\lambda$  to find the sweet spot for the optimal rendering quality and speed.

### 4.3 Comparison with transfer function design

In this section, we compare NUI-VR<sup>2</sup> with the traditional transfer function design method. With NUI-VR<sup>2</sup>, users only need to specify some seeds/masks to label the target. Figure 8 shows some example renderings with NUI-VR<sup>2</sup>. The default feature set and image segmentation engine in NUI-VR<sup>2</sup> highlight the targets very well. For example, we only select seeds from the left hippocampus, and NUI-VR<sup>2</sup> only renders the left hippocampus accordingly, even though the right hippocampus shares almost the same contexture as the left one. It is tough to do a similar rendering with the traditional transfer function design method.

The brain tumor is the easiest one to render with a transfer function, and we used ImageVis3D to highlight it. The 2D transfer function editor in ImageVis3D is a histogram of the intensity ( $x$ -axis) and gradient magnitude ( $y$ -axis) of voxels. To highlight a target, we need to place all kinds of geometries in different places in the editor by trials and errors. In comparison, users only need to designate a few

sample points in the VR environment with NUI in NUI-VR<sup>2</sup>. Figure 9 shows the rendering result with ImageVis3D. We cannot render the brain tumor differently from the other objects even with a carefully crafted 2D transfer function. A higher-dimensional transfer function will be more effective in highlighting the buried head tumor, but a steeper learning curve and more design efforts will be required.

## 5 Conclusion

In this paper, we detail the design of NUI-VR<sup>2</sup>: a NUI-enabled volume rendering system in the VR space. NUI-VR<sup>2</sup> marries image segmentation and volume rendering. Numerous general-purpose image segmentation algorithms could fit into our system. Users only need to define some seeds/masks to label the target instead of designing a complicated transfer function. We also design a Kinect-based NUI system based on 3D gestures and voice commands. Users can explore the volume, select seeds, and generate boundary masks directly in the 3D space. All the operations happen in an immersive VR environment. VR naturally fits 3D volume rendering and improves the user experience in perceiving the volume. NUI-VR<sup>2</sup> dramatically simplifies the target-centric volume rendering process and delivers high-quality rendering results.

**Acknowledgements** This work was supported in part by the Department of Education of Guangdong Province under Grant 2017KZDXM072, in part by the National Natural Science Foundation of China under Grant 61601302, in part by the Shenzhen Key Laboratory Foundation ZDSYS20200811143757022, in part by the Shenzhen Peacock Plan under Grant KQTD2016053112051497, and in part by the Faculty Development Grant of Shenzhen University under Grant 2018009. N. Xiong would like to thank the support from the National Key R&D Program of China 2016YFC1306600 and 2018YFC1314700, grant 2016CFB624 from Natural Science Foundation of Hubei Province, Grant 2017050304010278 from The Youth Science and technology morning light program of Wuhan City, 2018 Hubei medical research project WJ2019F030, 2018 Wuhan medical



research project WX18A10, 2018 Wuhan Young and Middle-aged medical Talents Program and 2017 Hubei provincial Party Committee Organization Department the second batch of Hubei youth elite development plan. X. Yu would like to thank the support from the Startup funding for Youth Faculty of Shenzhen University Grant 2018009. We thank the authors of ImageVis3D, supported by the National Institute of General Medical Sciences of the National Institutes of Health under grant number P41 GM103545-18, and the DOE SciDAC Visualization and Analytics Center for Enabling Technologies, DEFC0206ER25781.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Arens S, Domik G (2010) A survey of transfer functions suitable for volume rendering. In: Proceedings of the 8th IEEE/EG international conference on Volume Graphics. Eurographics Association, pp 77–83
- Bajaj CL, Pascucci V, Schikore DR (1997) The contour spectrum. In: Proceedings of the 8th conference on Visualization'97. IEEE Computer Society Press, pp 167–ff
- Bali A, Singh SN (2015) A review on the strategies and techniques of image segmentation. In: 2015 fifth international conference on advanced computing & communication technologies. IEEE, pp 113–120
- Boykov YY, Jolly M-P (2001) Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE international conference on, vol 1. IEEE, pp 105–112
- Caban JJ, Rheingans P (2008) Texture-based transfer functions for direct volume rendering. *IEEE Trans Visual Comput Graphics* 14(6):1364–1371
- Chan S, Conti F, Salisbury K, Blevins NH (2013) Virtual reality simulation in neurosurgery: technologies and evolution. *Neurosurgery* 72 (suppl\1) A154–A164
- Chang C, Huang C, Zhou N, Li SX, Ver Hoef L, Gao Y (2018) The bumps under the hippocampus. *Hum Brain Map* 39(1):472–490
- Cohen AR, Lohani S, Manjila S, Natsupakpong S, Brown N, Cavusoglu MC (2013) Virtual reality simulation: basic concepts and use in endoscopic neurosurgery training. *Childs Nerv Syst* 29(8):1235–1244
- El Beheiry M, Doutreligne S, Caporal C, Ostertag C, Dahan M, Masson J-B (2019) Virtual reality: beyond visualization. *J Mol Biol* 431(7):1315–1321
- Faludi B, Zoller EI, Gerig N, Zam A, Rauter G, Cattin PC (2019) Direct visual and haptic volume rendering of medical data sets for an immersive exploration in virtual reality. In: International conference on medical image computing and computer-assisted intervention. Springer, pp 29–37
- Fogal T, Krüger JH (2010) Tuvok, an architecture for large scale volume rendering. In: Proceedings of the 15th international workshop on vision, modeling, and visualization, pp 139–146. <http://www.sci.utah.edu/~tfogal/academic/tuvok/Fogal-Tuvok.pdf>
- Gao Y, Kikinis R, Bouix S, Shenton M, Tannenbaum A (2012) A 3d interactive multi-object segmentation tool using local robust statistics driven active contours. *Med Image Anal* 16(6):1216–1227
- Gao Y, Tannenbaum A, Kikinis R (2010) Simultaneous multi-object segmentation using local robust statistics and contour interaction. In: International MICCAI workshop on medical computer vision. Springer, pp 195–203
- Gao Y, Gholami B, MacLeod RS, Blauer J, Haddad WM, Tannenbaum AR (2010) Segmentation of the endocardial wall of the left atrium using local region-based active contours and statistical shape learning. Georgia Institute of Technology
- He T, Hong L, Kaufman A, Pfister H (1996) Generation of transfer functions with stochastic search techniques. In: Visualization'96. Proceedings. IEEE, pp 227–234
- Hänel C, Weyers B, Hentschel B, Kuhlen TW (2016) Visual quality adjustment for volume rendering in a head-tracked virtual environment. *IEEE Trans Visual Comput Graphics* 22(4):1472–1481
- Ju M, Choi Y, Seo J, Sa J, Lee S, Chung Y, Park D (2018) A Kinect-based segmentation of touching-pigs for real-time monitoring. *Sensors* 18(6):1746
- Karasev P, Kolesov I, Fritscher K, Vela P, Mitchell P, Tannenbaum A (2013) Interactive medical image segmentation using PDE control of active contours. *IEEE Trans Med Imaging* 32(11):2127–2139
- Kindlmann G, Whitaker R, Tasdizen T, Moller T (2003) Curvature-based transfer functions for direct volume rendering: methods and applications. In: Visualization. VIS 2003. IEEE. IEEE, 2003, pp 513–520
- Kuruvilla J, Sukumaran D, Sankar A, Joy SP (2016) A review on image processing and image segmentation. In: International conference on data mining and advanced computing (SAPIENCE). IEEE, pp 198–203
- König A, Gröller E (2001) Mastering transfer function specification by using VolumePro technology. *Spring Conf Comput Graphics* 17:279–286
- Levoy M (1988) Display of surfaces from volume data. *IEEE Comput Graphics Appl* 8(3):29–37
- Ljung P, Krüger J, Gröller E, Hadwiger M, Hansen CD, Ynnerman A (2016) State of the art in transfer functions for direct volume rendering. In: Computer graphics forum, vol. 35. Wiley Online Library, pp 669–691
- Mady AS, Abou El-Seoud S (2020) An overview of volume rendering techniques for medical imaging. *Int J Online Biomed Eng* 16(6):95–106
- Mortensen EN, Barrett WA (1998) Interactive segmentation with intelligent scissors. *Graphical Models Image Process* 60(5):349–384
- Pfister H, Lorensen B, Bajaj C, Kindlmann G, Schroeder W, Avila LS, Raghu K, Machiraju R, Lee J (2001) The transfer function bake-off. *IEEE Comput Graphics Appl* 21(3):16–22
- Sabella P (1988) A rendering algorithm for visualizing 3d scalar fields. *ACM SIGGRAPH Comput Graphics* 22(4):51–58
- Schroeder WJ, Lorensen B, Martin K (2004) The visualization toolkit: an object-oriented approach to 3D graphics, Kitware
- Takeshima Y, Takahashi S, Fujishiro I, Nielson GM (2005) Introducing topological attributes for objective-based visualization of simulated datasets. In: Fourth international workshop on volume graphics, 2005. IEEE, pp 137–236
- Tappenbeck A, Preim B, Dicken V (2006) Distance-based transfer function design: Specification methods and applications. In: *SimVis*, pp 259–274
- Tiede U, Schiemann T, Hohne KH (1998) High quality rendering of attributed volume data. In: Visualization'98. Proceedings. IEEE, pp 255–262

- Terrell GR, Scott DW (1992) Variable kernel density estimation. *Ann Stat* 20(3):1236–1265
- Tzeng F-Y, Lum EB, Ma K-L (2005) An intelligent system approach to higher-dimensional classification of volume data. *IEEE Trans Visual Comput Graphics* 11(3):273–284
- Vezhnevets V, Konouchine V (2005) Growcut: interactive multi-label nd image segmentation by cellular automata. In: *Proceedings of Graphicon*, vol 1, pp 150–156
- Vimort J, McCormick M, Budin F, Paniagua B (2017) Computing textural feature maps for n-dimensional images. *Insight J* 80014-2
- Wang Y, Jung C (2017) Interaction-free hand segmentation using Kinect camera. In: *2017 IEEE international conference on image processing (ICIP)*. IEEE, pp 4593–4593
- Weber GH, Dillard SE, Carr H, Pascucci V, Hamann B (2007) Topology-controlled volume rendering. *IEEE Trans Visual Comput Graphics* 13(2):330–341
- Zhu SC, Yuille A (1996) Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Trans Pattern Anal Mach Intell* 18(9):884–900
- Zhu L, Kolesov I, Gao Y, Kikinis R, Tannenbaum A (2014) An effective interactive medical image segmentation method using fast growcut. *MICCAI Workshop on Interactive Medical Image Computing*
- Zhu J, Rosset S, Tibshirani R, Hastie TJ (2004) 1-norm support vector machines. *Advances in Neural Information Processing Systems*, pp 49–56

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.