Real-time True-color Volume Visualization of Multi-channel 3D CLSM Images Based on CUDA

Yakang Dai*, Yunhai Zhang, Zhiyong Zhou, Haomin Yang, Xiaojun Xue

Suzhou Institute of Biomedical Engineering and Technology,
Chinese Academy of Sciences, Suzhou 215163, China
*Correspondence: daiyk@sibet.ac.cn

Abstract. Confocal laser scanning microscope (CLSM) can obtain multi-channel 3D images from biological tissues for analysis of tissue structures, where each channel results from a particular fluorescent label. The visualization of the multi-channel 3D confocal microscope images is required for qualitative, interactive, and even quantitative analysis of the structures of the studied tissues. Despite the available volume rendering algorithms for visualization of single-channel 3D images, the algorithms and toolkits specially used for true-color volume visualization of multi-channel 3D confocal microscope images were seldom reported. We have developed a new true-color volume rendering tool for real-time volume visualization of the multi-channel 3D confocal microscope images based on compute unified device architecture (CUDA). Manipulations including rotation, translation, and zooming are integrated with the volume rendering for interactive analysis of the multi-channel 3D images. In this paper, the methodologies of the tool are described and experimental results are illustrated. The tool is free of charge for research purpose.

Keywords: CLSM, multi-channel 3D images, visualization, volume rendering, CUDA.

1 Introduction

Confocal laser scanning microscope (CLSM) can emit and focus a laser beam at a spot on a focal plane within a specimen, and detect scattered and reflected laser light as well as any fluorescent light to image the spot of the specimen. The focal plane is scanned spot by spot, obtaining a 2D image with very high spatial resolution and signal-to-noise ratio [1]. Furthermore, the depth of the focal plane is selective (in z axis direction), leading to 3D imaging of a thick specimen [2]. CLSM is a very effective and necessary scientific instrument to image tiny structures at micron/submicron scale, thus has been widely applied in tissue biology, cell biology, molecular biology, genomics, neurology, embryology, pathology, immunology, epidemiology, oncology, bacteriology and virology etc. [3-6].

The basic concept of confocal microscopy was originally proposed by M. Minsky in 1950s [7]. CLSM instruments were then designed by several investigators, and the first commercial product was released in 1987. After that, confocal technologies have

attracted great attention and interest from researchers, and both research and commercial CLSM systems were largely developed. Currently, studies of CLSM are mainly focused on improvement of the image resolution, development of new functions, and analysis of CLSM images [8-9]. One of the popular topics is multi-channel (may include R, G, and B channels) CLSM, which can image multiple fluorescent labeling tissues simultaneously. We have also developed a new multi-channel CLSM system recently [10].

A multi-channel CLSM can discriminate various fluorescent components [11], and implement complex functional experiments such as fluorescence recovery after photobleaching (FRAP) [12], fluorescence resonance energy transfer (FRET) [13], and fluorescence lifetime imaging microscopy (FLIM) [14]. Obviously, true-color volume visualization of the multi-channel 3D image is very helpful for vivid display of the results from the multi-channel CLSM. However, most existing image analysis tools (such as VTK, ITK, and MITK) can only visualize single-channel 3D images. Voxx [15] and Vaa3D [16] are two tools that are able to render multi-channel 3D images. Voxx implements volume rendering based on GLSL or ARB fragment program, and mainly supports two-photon Bio-Rad PIC, Zeiss LSM, and raw voxel files. Vaa3D implements volume rendering using OpenGL 2D or 3D texture mapping, and supports three-channel TIFF files.

We have developed a new true-color volume rendering tool for real-time volume visualization of the multi-channel 3D CLSM images using the novel compute unified device architecture (CUDA) [17] developed by NVIDIA. Compared to Voxx [15] and Vaa3D [16], the major feature of our tool is that it implements the multi-channel volume rendering based on the C-language-like CUDA architecture which is a parallel computing platform and programming model. The volume rendering speed of our tool is comparable to Voxx and Vaa3D, and the proposed CUDA-based true-color volume rendering algorithm is easy to follow. The following sections describe the methodologies of the CUDA-based true-color volume rendering tool and demonstrate representative experimental results.

2 Methods

2.1 CUDA Architecture

Before the description of the CUDA-based true-color volume rendering algorithm, we firstly present a brief overview of the CUDA architecture [17], which was designed by NVIDIA and is now very popular for parallelization of scientific computation. A graphical processing unit (GPU) on a computer is regarded as a separate parallel processing coprocessor to the computer. An algorithm implemented in CUDA language is called a kernel, which, when invoked by a program on the computer, is executed on the GPU by using a 1-D or 2-D grid which consists of blocks. Each block contains multiple threads. The blocks (1-D, 2-D, or 3-D) are distributed to multiprocessors on the GPU, and the threads within each block execute the kernel in parallel on the corre-

sponding multiprocessor. After all blocks accomplish the execution, the kernel is terminated.

2.2 Rendering environment

We used ray casting to render the volume (i.e., multi-channel 3D image). As shown in Fig. 1, there are five coordinate systems in the rendering environment, including the coordinate system of the grid space *G*, the coordinate system of the volume space *M*, the world coordinate system *W*, the coordinate system of the view space *V*, and the screen coordinate system *S*. *W* is the absolute coordinate system where the volume is placed. The view space defines the visible region in *W*. The coordinate transformation from the grid space to the screen space can be written as:

$$ {}^{S}X = {}^{S}T_{V} \cdot {}^{V}T_{W} \cdot {}^{W}T_{M} \cdot {}^{M}T_{G} \cdot {}^{G}X \quad (1) $$

where ${}^{G}X$ is the position of each voxel (which has R, G, and B channels) in *G*, and ${}^{S}X$ is the transformed coordinate value in *S*. Both of them can be denoted as $[x, y, z, 1]^{T}$. ${}^{M}T_{G}$, ${}^{W}T_{M}$, ${}^{V}T_{W}$ and ${}^{S}T_{V}$ can be written in a unified format ${}^{J}T_{I}$, which is a 4x4 matrix representing the transformation from the coordinate system *I* to the coordinate system *J*. The CUDA-based true-color ray casting algorithm is described in the following section.

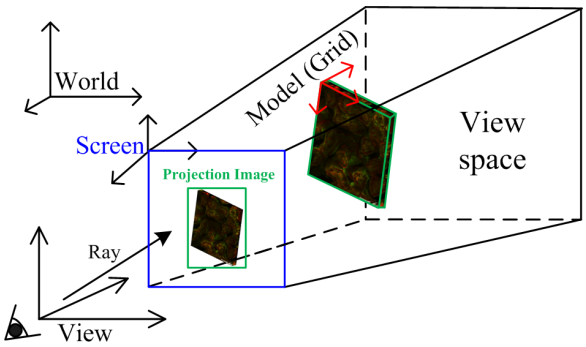

Fig. 1. Rendering environment. There are five coordinate systems. The volume model placed in the view space is projected onto the screen using the CUDA-based true-color ray casting algorithm.

2.3 CUDA-based True-color Ray Casting Algorithm

The algorithm consists of two steps: calculation of the range of the projection image, and parallel ray casting using CUDA.

(1) Calculation of the position and size of the projection image

Firstly, we computed the projection coordinates (*x* and *y* elements of the ${}^{S}X$) of the eight vertices of the volume in the screen by

$$^{S}X = {}^{S}T_{G} \cdot {}^{G}X \tag{2}$$

where ${}^{S}T_{G}$ is the transformation matrix from the grid coordinate system G to the screen coordinate system S, ${}^{G}X$ and ${}^{S}X$ are the coordinates of the vertices in the G and S, respectively. Secondly, with the minimal and maximal x and y elements of the eight coordinates in S, we determined a compact rectangle enclosing the projection image (see Fig. 1). Finally, we performed clamping operations to ensure that the projection image to be obtained by using the CUDA-based parallel ray casting would be within the screen.

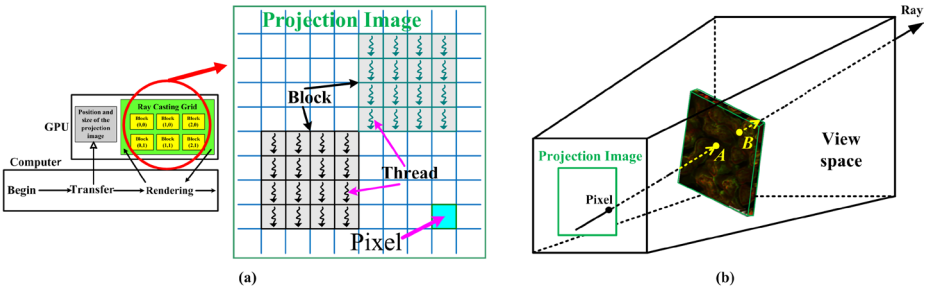

Fig. 2. Parallel ray casting using CUDA. (a) Each pixel in the projection image was assigned a thread based on the CUDA architecture, and the finally accumulated colors of the pixels were calculated in parallel by the threads. (b) Ray casting computation performed by a thread associated with the corresponding pixel.

(2) Parallel ray casting using CUDA

Fig. 2 illustrates the approach we used to obtain the projection image. Specifically, as shown in Fig. 2(a): firstly, the position and size of the projection image were transferred from the host computer to the GPU, and the accumulated color and accumulated opacity of each pixel in the projection image were initialized to zeros on the GPU; secondly, parallel ray casting was performed on the GPU by using the ray casting grid which assigned each pixel a thread to calculate the finally accumulated color of the pixel. The algorithm implemented by each thread associated with a pixel is described as follows:

1) Cast a ray from the origin (see Fig. 1) of the view space V to the pixel, and find the two sequential intersections (A and B, respectively, see Fig. 2(b)) between the ray and the volume in the view space.
2) Traverse from A to B, calculate color and opacity at each sampling position, and compute the final accumulated color and accumulated opacity recursively according to the following equation

$$\begin{cases} c \leftarrow C_S \cdot \alpha_S \cdot (1-\alpha) + c \\ \alpha \leftarrow \alpha_S \cdot (1-\alpha) + \alpha \end{cases} \tag{3}$$

where c (which has R, G, and B channels) and α are respectively the accumulated color and accumulated opacity, C_S (which has R, G, and B channels) and α_s are respectively the color and opacity at the sampling position S. It's worth noting

that C_S was reconstructed from the colors of the neighboring voxels by interpolation, while α_s was defined according to a linear grey-opacity transfer function, where the grey corresponding to α_s was calculated based on the R, G, and B values of C_S by

$$\text{Grey} = \text{R} \times 0.299 + \text{G} \times 0.587 + \text{B} \times 0.114 \quad (4)$$

Finally, the projection image was obtained after all pixels were processed.

2.4 Graphical user interface and human computer interaction

We have developed a software system with graphical user interface (GUI) to integrate our CUDA-based true-color ray casting algorithm. The GUI was designed using Qt (a cross-platform GUI tool) [18], while the proposed algorithm was implemented in CUDA and C/C++ based on the volume rendering framework of our medical imaging toolkit (MITK) [19], where the projection image was rendered using OpenGL [20]. The 3D interaction framework of the MITK was also used for interactive and intuitive manipulation of the volume, including rotation, translation, and zooming of the volume by mouse. Fig. 3 illustrates the main window of the software system.

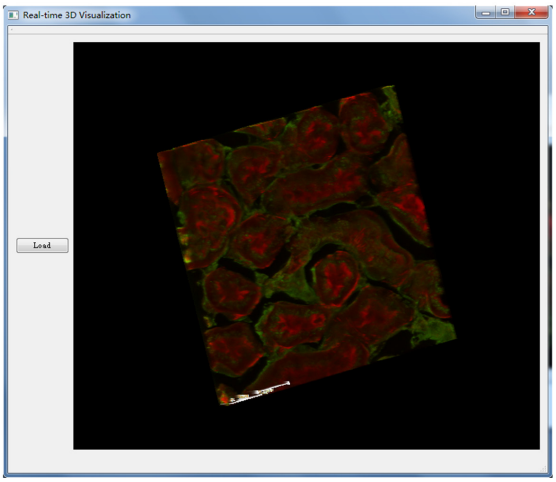

Fig. 3. The main window of the software system for true-color volume visualization of multi-channel 3D CLSM images. The 3D image can be rotated, translated, and zoomed by mouse manipulation in the display window.

3 Experimental results

Experiments were performed on a 32-bit Windows PC (with an Intel Core i5-2400 3.1 GHz processor, 2GB physical memory, and Geforce 605 GPU with 512 MB memory) to demonstrate the validity of our algorithm for real-time volume visualization of

multi-channel 3D CLSM images. Three specimens were scanned using a LEICA CLSM system, including bovine pulmonary artery endothelial (BPAE) cell, mouse kidney section, and muntjac skin fibroblasts, generating three multi-channel (RGB) 3D CLSM volume images. The sizes of the volume images from the BPAE cell, mouse kidney section, and muntjac skin fibroblasts were 512×512×121×3 (90 MB), 1024×1024×50 ×3 (150 MB), and 512×512×99×3 (74 MB), respectively. The rendering efficiencies of our algorithm for the three volume images are shown in Table 1, indicating that real-time visualization can be achieved. Fig. 4, Fig. 5, and Fig. 6 show the volume rendering results for the BPAE cell, mouse kidney section, and muntjac skin fibroblasts, respectively.

Table 1. Rendering efficiencies of the CUDA-based true-color ray casting algorithm for the three multi-channel (RGB) 3D CLSM volume images. It can be seen that real-time rendering rate can be achieved.

Specimen	Frames/second
BPAE cell	43
Mouse kidney section	24
Muntjac skin fibroblasts	36

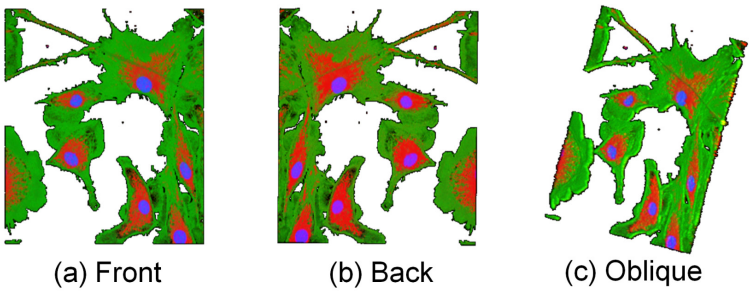

Fig. 4. Volume rendering results for the BPAE cell. (a) The front view. (b) The back view. (c) An oblique view.

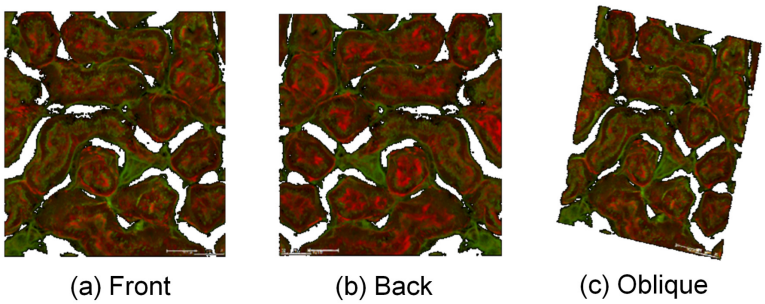

Fig. 5. Volume rendering results for the mouse kidney section. (a) The front view. (b) The back view. (c) An oblique view.

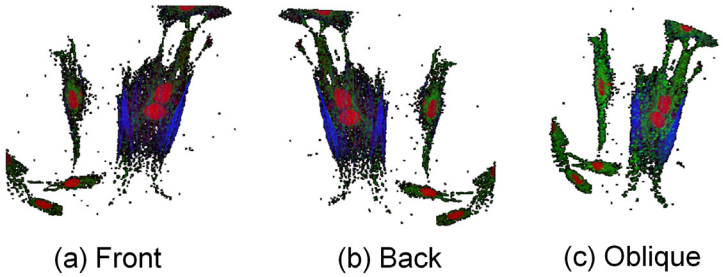

Fig. 6. Volume rendering results for the muntjac skin fibroblasts. (a) The front view. (b) The back view. (c) An oblique view.

4 Conclusion and future work

We have proposed a CUDA-based true-color ray-casting volume rendering algorithm for real-time volume visualization of multi-channel 3D CLSM images, and developed a preliminary software system integrating the algorithm. The experimental results demonstrated the validity of our algorithm and software system. Interested researchers could contact us for the developed software which is free of charge for research purpose. In the future, we will improve the functionality of the tool, such as designing interfaces for adjusting the grey-opacity transfer function interactively, adding an interactive volume clipping module, etc.

Acknowledgement

This work was supported in part by the Hundred Talents Program of CAS, NSFC grants (61301042, 61201117), NSFJ grant (BK2012189), and STPS grant (SYG201324).

References

1. Hunter, J.J., Cookson, C.J., Kisilak M.L., Bueno J.M., Campbell, M.C.W.: Characterizing image quality in a scanning laser ophthalmoscope with differing pinholes and induced scattered light. J Opt Soc Am A Opt Image Sci Vis, Vol. 24, pp. 1284-1295 (2007)
2. Uhl, R., Daum, R., Harz, H., Seebacher, C., Neogy, S.: High Throughput High Content Live Cell Screening Platform. Biophotonics 2007: Optics in Life Science, J. Popp and G. von Bally, eds., Vol. 6633 of Proceedings of SPIE-OSA Biomedical Optics (Optical Society of America, 2007), paper 6633_11 (2007)
3. Matsumoto, B.: Cell Biological Applications of Confocal Microscopy. Methods in Cell Biology, Vol. 70, New York: Academic Press (2002)
4. Hibbs, A.R.: Confocal Microscopy for Biologists. New York: Kluwer Academic (2004).

5. Peterman E.J.G., Sosa, H., Moerner, W.E.: Single-Molecule Fluorescence Spectroscopy and Microscopy of Biomolecular Motors. Ann. Rev. Phys. Chem., Vol. 55, pp. 79-96 (2004)
6. Dobbs J.L., Ding, H., Benveniste, A., Keurer, H., Krishnamurthy, S., Yang, W., Richards-Kortum, R.: Confocal Fluorescence Microscopy for Evaluation of Breast Cancer in Human Breast Tissue. Biomedical Optics, OSA Technical Digest (Optical Society of America, 2012), paper BTu3A.19 (2012)
7. Minsky, M.: Microscopy Apparatus. US Pat. 3,013,467, (1961)
8. Boruah, B.R.: Lateral resolution enhancement in confocal microscopy by vectorial aperture engineering. Appl. Opt., Vol. 49, pp. 701-707 (2010)
9. Bückers, J., Wildanger, D., Vicidomini, G., Kastrup, L., Hell, S.W.: Simultaneous multi-lifetime multi-color STED imaging for colocalization analyses. Opt. Express, Vol. 19, pp. 3130-3143 (2011)
10. Zhang, Y., Hu, B., Dai, Y., Yang, H., Huang, W., Xue, X., Li, F., Zhang, X., Jiang, C., Gao, F., Chang, J.: A New Multichannel Spectral Imaging Laser Scanning Confocal Microscope. Computational and Mathematical Methods in Medicine, Vol. 2013, Article ID 890203, 8 pages (2013)
11. Sinclair, M.B., Haaland, D.M., Timlin, J.A., Jones, H.D.T.: Hyperspectral confocal microscope. Appl. Opt., Vol. 45, pp. 6283-6291 (2006)
12. Mazza, D., Cella, F., Vicidomini, G., Krol, Silke., Diaspro, A.: Role of three-dimensional bleach distribution in confocal and two-photon fluorescence recovery after photobleaching experiments. Appl. Opt., Vol. 46, pp. 7401-7411 (2007)
13. Kim, S., Choi, D., Kim, D.: Single-molecule Detection of Fluorescence Resonance Energy Transfer Using Confocal Microscopy. J. Opt. Soc. Korea, Vol. 12, pp. 107-111 (2008)
14. Auksorius, E., Boruah, B.R., Dunsby, C., Lanigan, P.M.P., Kennedy, G., Neil, M.A.A., French, P.M.W.: Stimulated emission depletion microscopy with a supercontinuum source and fluorescence lifetime imaging. Optical Society of America, Vol.33, pp. 113-115 (2008)
15. Clendenon, J.L., Phillips, C.L., Sandoval, R.M., Fang, S., Dunn, K.W.: Voxx: a PC-based, near real-time volume rendering system for biological microscopy. Am J Physiol Cell Physiol, Vol. 282, pp. C213-218 (2002)
16. Peng, H., Ruan, Z., Long, F., Simpson, J.H., Myers, E.W.: V3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. Nature Biotechnology, Vol. 28, pp. 348-353 (2010)
17. CUDA. NVIDIA, https://developer.nvidia.com.
18. QT. Digia, http://www.trolltech.com.
19. Tian, J., Xue, J., Dai, Y., Chen, J., Zheng, J.: A novel software platform for medical image processing and analyzing. IEEE Transactions on Information Technology in Biomedicine, Vol. 12, pp. 800-812 (2008)
20. OpenGL. SGI, http://www.opengl.org.