Real-time 4D signal processing and visualization using graphics processing unit on a regular nonlinear-k Fourier-domain OCT system

Kang Zhang* and Jin U. Kang

Department of Electrical and Computer Engineering, The Johns Hopkins University, 3400 N. Charles Street, Baltimore 21218, Maryland, USA
**kzhang8@jhu.edu*

Abstract: We realized graphics processing unit (GPU) based real-time 4D (3D + time) signal processing and visualization on a regular Fourier-domain optical coherence tomography (FD-OCT) system with a nonlinear k-space spectrometer. An ultra-high speed linear spline interpolation (LSI) method for λ-to-k spectral re-sampling is implemented in the GPU architecture, which gives average interpolation speeds of >3,000,000 line/s for 1024-pixel OCT (1024-OCT) and >1,400,000 line/s for 2048-pixel OCT (2048-OCT). The complete FD-OCT signal processing including λ-to-k spectral re-sampling, fast Fourier transform (FFT) and post-FFT processing have all been implemented on a GPU. The maximum complete A-scan processing speeds are investigated to be 680,000 line/s for 1024-OCT and 320,000 line/s for 2048-OCT, which correspond to 1GByte processing bandwidth. In our experiment, a 2048-pixel CMOS camera running up to 70 kHz is used as an acquisition device. Therefore the actual imaging speed is camera-limited to 128,000 line/s for 1024-OCT or 70,000 line/s for 2048-OCT. 3D Data sets are continuously acquired in real time at 1024-OCT mode, immediately processed and visualized as high as 10 volumes/second (12,500 A-scans/volume) by either *en face* slice extraction or ray-casting based volume rendering from 3D texture mapped in graphics memory. For standard FD-OCT systems, a GPU is the only additional hardware needed to realize this improvement and no optical modification is needed. This technique is highly cost-effective and can be easily integrated into most ultrahigh speed FD-OCT systems to overcome the 3D data processing and visualization bottlenecks.

OCIS codes: (170.4500) Optical coherence tomography; (170.3890) Medical optics instrumentation; (200.4560) Optical data processing.

References and links

1. B. Potsaid, I. Gorczynska, V. J. Srinivasan, Y. Chen, J. Jiang, A. Cable, and J. G. Fujimoto, "Ultrahigh speed spectral / Fourier domain OCT ophthalmic imaging at 70,000 to 312,500 axial scans per second," Opt. Express **16**(19), 15149–15169 (2008), http://www.opticsinfobase.org/oe/abstract.cfm?uri=oe-16-19-15149.
2. R. Huber, D. C. Adler, and J. G. Fujimoto, "Buffered Fourier domain mode locking: Unidirectional swept laser sources for optical coherence tomography imaging at 370,000 lines/s," Opt. Lett. **31**(20), 2975–2977 (2006).
3. I. Grulkowski, M. Gora, M. Szkulmowski, I. Gorczynska, D. Szlag, S. Marcos, A. Kowalczyk, and M. Wojtkowski, "Anterior segment imaging with Spectral OCT system using a high-speed CMOS camera," Opt. Express **17**(6), 4842–4858 (2009), http://www.opticsinfobase.org/oe/abstract.cfm?uri=oe-17-6-4842.
4. M. Gora, K. Karnowski, M. Szkulmowski, B. J. Kaluzny, R. Huber, A. Kowalczyk, and M. Wojtkowski, "Ultra high-speed swept source OCT imaging of the anterior segment of human eye at 200 kHz with adjustable imaging range," Opt. Express **17**(17), 14880–14894 (2009), http://www.opticsinfobase.org/oe/abstract.cfm?uri=oe-17-17-14880.

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

5. M. Gargesha, M. W. Jenkins, D. L. Wilson, and A. M. Rollins, "High temporal resolution OCT using image-based retrospective gating," Opt. Express **17**(13), 10786–10799 (2009), http://www.opticsinfobase.org/oe/abstract.cfm?uri=oe-17-13-10786.
6. M. Gargesha, M. W. Jenkins, A. M. Rollins, and D. L. Wilson, "Denoising and 4D visualization of OCT images," Opt. Express **16**(16), 12313–12333 (2008), http://www.opticsinfobase.org/oe/abstract.cfm?uri=oe-16-16-12313.
7. M. W. Jenkins, F. Rothenberg, D. Roy, V. P. Nikolski, Z. Hu, M. Watanabe, D. L. Wilson, I. R. Efimov, and A. M. Rollins, "4D embryonic cardiography using gated optical coherence tomography," Opt. Express **14**(2), 736–748 (2006), http://www.opticsinfobase.org/oe/abstract.cfm?URI=OPEX-14-2-736.
8. G. Liu, J. Zhang, L. Yu, T. Xie, and Z. Chen, "Real-time polarization-sensitive optical coherence tomography data processing with parallel computing," Appl. Opt. **48**(32), 6365–6370 (2009).
9. J. Probst, P. Koch, and G. Huttmann, "Real-time 3D rendering of optical coherence tomography volumetric data," Proc. SPIE **7372**, 73720Q (2009).
10. B. R. Biedermann, W. Wieser, C. M. Eigenwillig, G. Palte, D. C. Adler, V. J. Srinivasan, J. G. Fujimoto, and R. Huber, "Real time en face Fourier-domain optical coherence tomography with direct hardware frequency demodulation," Opt. Lett. **33**(21), 2556–2558 (2008).
11. Y. Watanabe, and T. Itagaki, "Real-time display on Fourier domain optical coherence tomography system using a graphics processing unit," J. Biomed. Opt. **14**(6), 060506 (2009).
12. Z. Hu, and A. M. Rollins, "Fourier domain optical coherence tomography with a linear-in-wavenumber spectrometer," Opt. Lett. **32**(24), 3525–3527 (2007).
13. K. Zhang, W. Wang, J. Han, and J. U. Kang, "A surface topology and motion compensation system for microsurgery guidance and intervention based on common-path optical coherence tomography," IEEE Trans. Biomed. Eng. **56**(9), 2318–2321 (2009).
14. U. Sharma, and U. Jin, "Common-path optical coherence tomography with side-viewing bare fiber probe for endoscopic OCT," Rev. Sci. Instrum. **78**, 113102 (2007).
15. K. Zhang, E. Katz, D. H. Kim, J. U. Kang, and I. K. Ilev, "Common-path optical coherence tomography guided fiber probe for spatially precise optical nerve stimulation," Electron. Lett. **46**(2), 118–120 (2010).
16. U. Sharma, N. M. Fried, and J. U. Kang, "All-fiber common optical coherence tomography: sensitivity optimization and system analysis," IEEE J. Sel. Top. Quantum Electron. **11**(4), 799–805 (2005).
17. NVIDIA, "NVIDIA CUDA Compute Unified Device Architecture Programming Guide Version 2.3.1," (2009).
18. NVIDIA, "NVIDIA CUDA CUFFT Library Version 2.3," (2009).
19. J. Kruger, and R. Westermann, "Acceleration techniques for GPU-based volume rendering," in *Proceedings of the 14th IEEE Visualization Conference (VIS'03)* (IEEE Computer Society, Washington, DC, 2003), pp. 287–292.
20. A. Kaufman, and K. Mueller, "Overview of Volume Rendering," in *The Visualization Handbook*, C. Johnson and C. Hansen, ed. (Academic Press, 2005).
21. M. Levoy, "Display of surfaces from volume data," IEEE Comput. Graph. Appl. **8**(3), 29–37 (1988).
22. D. Shreiner, M. Woo, J. Neider, and T. Davis, *OpenGL Programming Guide, Sixth Edition* (Addison-Wesley Professional, 2007), chap. 3.
23. C. Dorrer, N. Belabas, J. Likforman, and M. Joffre, "Spectral resolution and sampling issues in Fourier-transform spectral interferometry," J. Opt. Soc. Am. B **17**(10), 1795–1802 (2000).
24. A. D. Aguirre, P. Hsiung, T. H. Ko, I. Hartl, and J. G. Fujimoto, "High-resolution optical coherence microscopy for high-speed, *in vivo* cellular imaging," Opt. Lett. **28**(21), 2064–2066 (2003).

1. Introduction

The acquisition line (A-scan) speed of Fourier-domain optical coherence tomography (FD-OCT) has been advancing rapidly to >100,000 line/s level in the last few years. For a spectrometer based FD-OCT, an ultrahigh speed CMOS line scan camera based system has achieved up to 312,500 line/s [1]; while for a swept laser type, 370,000 line/s has been realized by using a Fourier domain mode-locking swept laser [2]. Such ultrahigh acquisition speed enables time-resolved volumetric (4D) recording and reconstruction of dynamic processes such as eye blinking, papillary reaction to light stimulus [3,4], and embryonic heart beating [5–7]. However, parallel efforts have not been embarked on data processing and visualization at the matching speed of the acquisition. Therefore, current real-time video-rate display is generally limited to 2D (B-scan) images. The most common way dealing with a huge volumetric data (C-scan) is to "capture and save" and then perform post-processing at a later time. The post-processing of 3D data usually includes two stages, FD-OCT signal processing and volumetric visualization, both of which are heavy-duty computing task due to the huge data size. Therefore the real-time signal processing and volumetric visualization become two bottlenecks for an ultra-high speed FD-OCT system that could be a practical

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

system for clinical applications such as surgical intervention and instrument guidance, which usually requires a real-time 4D imaging capability.

To overcome the signal processing bottleneck, several solutions have recently been proposed and demonstrated: Multi-CPU parallel processing has been implemented and achieved 80,000 line/s processing rate on nonlinear-k system [8] and 207,000 line/s on linear-k system for 1024-OCT [9]; A linear-k Fourier-domain mode-locked laser (FDML) with direct hardware frequency demodulation method enabled real-time *en face* image by yielding the analytic reflectance signal from one depth for each axial scan [10]; More recently, a graphics processing unit (GPU) has been utilized for processing FD-OCT data [11] using linear-k spectrometer. However, the methods in [9–11] are limited to highly-special linear-k FD-OCT systems to avoid interpolation for λ-to-k spectral re-sampling. Therefore they are not applicable to the majority of nonlinear-k FD-OCT systems. Moreover, a linear-k spectrometer is not completely linear over the whole spectrum range [11,12] so re-sampling would still be required for a wide spectrum range which is essential for achieving ultra-high axial resolution.

For the volumetric visualization issue, multiple 2D slice extraction and co-registration is the simplest approach, while volume rendering offers more comprehensive spatial view of the whole 3D data set, which is not immediately available from 2D slices. However, volume rendering such as ray-casting is usually very time-consuming for CPU. So real-time rendering for a large data volume is only available through GPU. Moreover, a complete 3D data set must be ready prior to any volumetric visualization due to the feature of FD-OCT signal processing method [10], which still would require a solution.

In this paper, we realized GPU based real-time 4D signal processing and visualization on a regular FD-OCT system with nonlinear k-space for the first time to the best of our knowledge. An ultra-high speed linear spline interpolation (LSI) method for λ-to-k spectral re-sampling is implemented in GPU architecture. The complete FD-OCT signal processing including interpolation, fast Fourier transform (FFT) and post-FFT processing have all been implemented on a GPU. 3D Data sets are continuously acquired in real time, immediately processed and visualized by either *en face* slice extraction or ray-casting based volume rendering from 3D texture mapped in graphics memory. For standard FD-OCT systems, a GPU is the only additional hardware needed to realize this improvement and no optical modification is needed. This technique is highly cost-effective and can be easily integrated into most ultrahigh speed FD-OCT systems to overcome the 3D data processing and visualization bottlenecks.

2. System configuration and CPU-GPU hybrid architecture

The FD-OCT system used in the test is shown in Fig. 1. A 12-bit CMOS camera (Sprint spL2048-140k, Basler AG, Germany) with 70,000 line/s effective line rate at 2048 pixel mode works as the detector of the OCT spectrometer. A superluminescence diode (SLED) (λ_0 = 825nm, $\Delta\lambda$ = 70nm, Superlum, Ireland) was used as the light source, which gives an axial resolution of approximately 5.5μm in air /4.1μm in water. The beam scanning was implemented by a pair of high speed galvanometer mirrors driven by a dual channel function generator and synchronized with a high speed frame grabber (PCIe-1429, National Instruments, USA). To simplify the alignment issues, our OCT system was configured in a common-path mode, where the reference signal comes from the bottom surface reflection of a glass window placed in between the scanning lens and sample, while the up surface is anti-reflective coated. The common-path structure doesn't require dispersion compensation optics while maintain a high axial resolution [13–16]. The lateral resolution is estimated to be 9.5 μm assuming Gaussian beam. An 8-core Dell T7500 workstation was used to obtain and display images, and a GPU (NVIDIA Quadro FX5800 graphics card) with 240 stream processors (1.3GHz clock rate) and 4GBytes graphics memory was used to perform OCT signal processing and 3D visualization such as *en face* slice extraction or volume rendering.

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

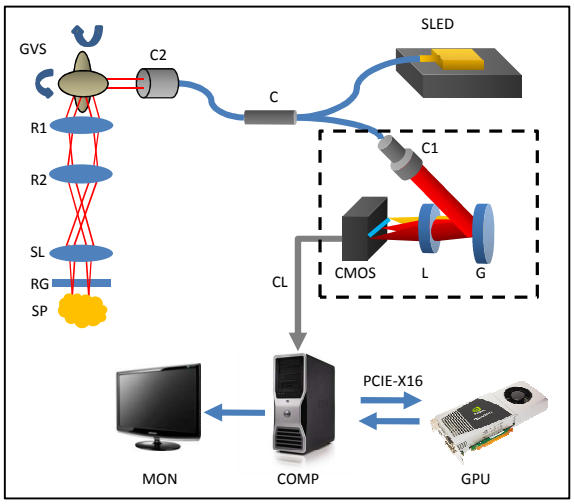

Fig. 1. System configuration; CMOS, CMOS line scan camera; L, spectrometer lens; G, reflective grating; C1, C2, achromatic collimators; C, 50:50 broadband fiber coupler; CL, camera link cable; COMP, host computer; GPU, graphics processing unit; PCIE-X16, PCI Express x16 2.0 interface; MON, Monitor; GVS, galvanometer mirror pairs; R1, R2, relay lens; SL, scanning lens; RG, reference glass; SP, Sample.

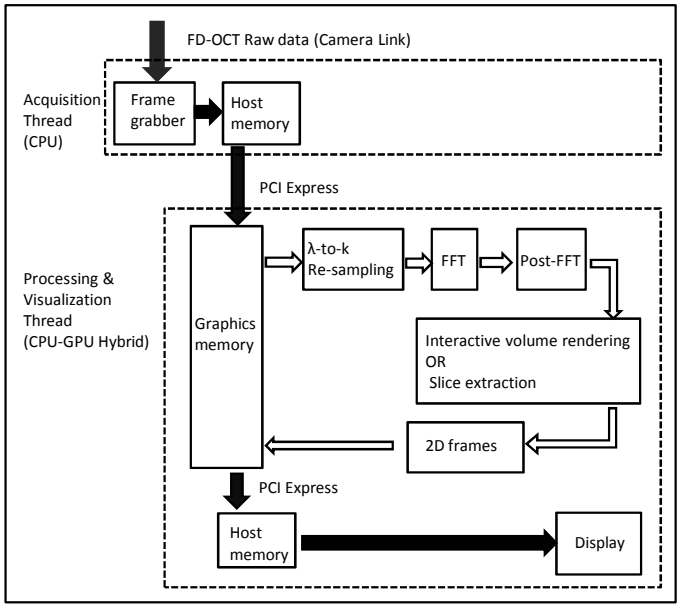

Fig. 2. CPU-GPU hybrid system architecture.

Figure 2 presents the CPU-GPU hybrid system architecture, where two synchronized threads were used for data acquisition, signal processing and visualization, respectively. The solid arrows describe the main data stream and the hollow arrows indicate the internal data flow of the GPU. In the acquisition thread, a certain number of raw OCT interference spectrums were sequentially grabbed from the CMOS camera, transferred into the frame grabber through camera link cable and then routed into the host computer's memory as a whole data block. In the processing and visualization thread, the grabbed data block was transferred into the graphics card memory through the PCI Express x16 2.0 interface, and then operated for each interferogram in parallel by the 240 graphics stream processors to complete

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

the standard processing including λ-to-k spectral remapping, fast Fourier transform, and post-FFT processing. In the post-FFT processing part, a reference volume acquired and saved in the graphics memory prior to imaging any sample was subtracted by the whole volume before logarithm scaling to remove the DC component as well as the noise and artifact caused by irregular reference signal from the reference plane. The processed 3D data set is then sent to the next stage for visualization by either direct *en face* slices extraction or being mapped to 3D texture allocated in the graphics memory to perform volume rendering, which will be illustrated in details in the later section. Finally the processed 2D frame is transferred back to the host memory and displayed in the graphical user interface (GUI). The GPU is programmed through NVIDIA's Compute Unified Device Architecture (CUDA) technology [17]. The FFT operation is implemented by the CUFFT library [18]. Since currently there is no suitable function in CUDA library for λ-to-k spectral re-sampling, here we propose a GPU accelerated linear spline interpolation (LSI) method as following:

Start from the LSI equation:

$$S'[j] = S[i] + \frac{S[i+1] - S[i]}{k[i+1] - k[i]}\left(k'[j] - k[i]\right), \tag{1}$$

where $k[n] = 2\pi / \lambda[n]$ is the nonlinear k-space value series and $\lambda[n]$ is the calibrated wavelength values of the FD-OCT system. $S[n]$ is the spectral intensity series corresponding to $k[n]$. $k'[n]$ is the linear k-space series covering the same frequency range as $k[n]$. Linear spline interpolation requires a proper interval $\left[k[i], k[i+1]\right]$ for each $k'[j]$, that is:

$$k[i] < k'[j] < k[i+1]. \tag{2}$$

Let a series $E[n]$ to present the lower ends for each element of $k[n]$, then Eq. (1) can be written as:

$$S'[j] = S[E[j]] + \frac{S[E[j]+1] - S[E[j]]}{k[E[j]+1] - k[E[j]]}\left(k'[j] - k[E[j]]\right). \tag{3}$$

$E[n]$ can be easily obtained before interpolation by comparing $k[n]$ and $k'[n]$. From Eq. (3), one would notice that $S'[j]$ is independent of other values in the series $S'[n]$, therefore this LSI algorithm is highly suitable for the parallel computation. Figure 3 shows the flowchart of parallelized LSI, where the parallel loops are distributed onto the GPU's 240 stream processors. The values of $E[n]$, $k[n]$ and $k'[n]$ are all stored in graphics global memory prior to interpolation, while the $S[n]$ and $S'[n]$ are allocated in real-timely refreshed memory blocks.

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

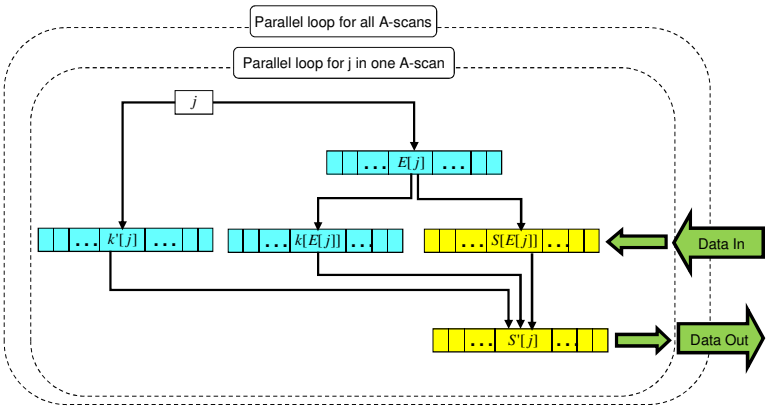

Fig. 3. Flowchart of parallelized LSI. Blue blocks: memory for pre-stored data; yellow blocks: memory for real-timely refreshed data.

3. Interactive volume rendering by ray-casting

Volume rendering is a numeric simulation of the eye's physical vision process in the real world, which provides better presentation of the entire 3D image data than the 2D slice extraction [19–21]. Ray-casting is the simplest and most straightforward method for volume rendering, shown as Fig. 4(a). An imaging plane is defined between the observer's eye and the data volume, and each pixel of the imaging plane is the integration along the specific eye ray through the pixel, which can be presented by the following recursive back-to-front compositing equations [21]:

$$C(\lambda)_{out}(u_j) = C(\lambda)_{in}(u_j) * (1-\alpha(x_i)) + C(\lambda)(x_i) * \alpha(x_i), \tag{4}$$

$$\alpha_{out}(u_j) = \alpha_{in}(u_j) * (1-\alpha(x_i)) + \alpha(x_i), \tag{5}$$

where $C(\lambda)(x_i)$ and $\alpha(x_i)$ stands for the color and opacity values of a single voxel at the spatial position x_i. $C(\lambda)_{out}(u_j)$, $\alpha_{out}(u_j)$, $C(\lambda)_{in}(u_j)$ and $\alpha_{in}(u_j)$ are the color and opacity values on a particular eye ray in and out of this voxel. The eye ray corresponds to a pixel position u_i on the image plane, and voxels along the ray will be taken into account for color and opacity accumulation.

The principle of ray-casting demands heavy computing duty, so in general real-time volume rendering can only be realized by using hardware acceleration devices like GPU. Figure 4(b) illustrates the details of the interactive volume rendering portion for Fig. 2. After post-FFT processing, the 3D data set is mapped into 3D texture, a pre-allocated read-only section on the graphics memory. A certain modelview matrix is obtained through the GUI functions to determine the relative virtual position between data volume and imaging plane [22]. Then the GPU performs ray-casting method to render the 2D frame from the 3D texture according to the modelview matrix. To insure compatibility with the NI-IMAQ Win32 API and simplify the software structure, we have developed and implemented the ray-casting function using the CUDA language and the 2D frames are finally displayed using an NI-IMAQ window.

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

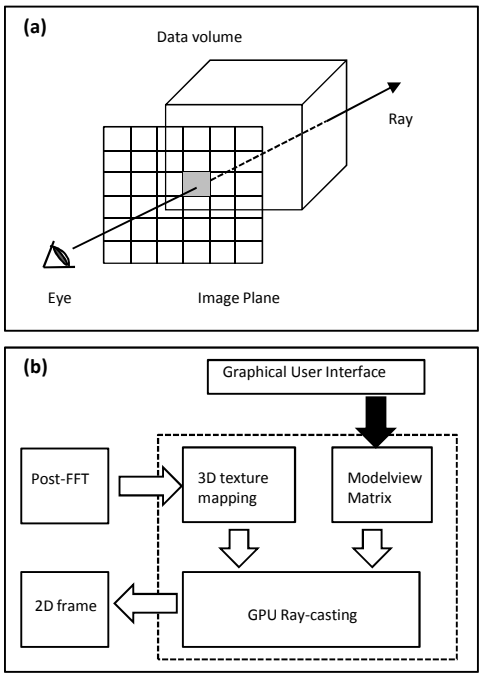

Fig. 4. (a) Schematic of ray-casting CPU-GPU hybrid architecture; (b) flowchart of interactive volume rendering by GPU.

4. OCT data processing capability

To test the GPU's OCT data processing ability, we processed a series of large numbers of A-scan lines in one batch. The complete processing time is recorded in millisecond from the interval between the data transfer-in (host memory to graphics memory) and data transfer-out (graphics memory to host memory), and the time for interpolation is also recorded. Here both 2048-pixel and 1024-pixel OCT modes were tested and the 1024-pixel mode was enabled by the CMOS camera's area-of-interest (AOI) output feature. The processing time versus one-batch line number is shown as Fig. 5(a). The corresponding processing line rate can be easily calculated and shown in Fig. 5(b). The interpolation speed averages at >3,000,000 line/s for 1024-OCT and >1,400,000 line/s for 2048-OCT. The complete processing speed goes up to 320,000 line/s for the 2048-OCT and 680,000 line/s for the 1024-OCT. This is equivalent to approximately 1GBytes/s processing bandwidth at 12 bit/pixel. Since commonly used high-end frame grabbers (i.e. PCIe-1429) has an acquisition bandwidth limit of 680MBytes/s, the GPU processing should be able to process all the OCT data in real-time. As one can see, the processing bandwidth decreases in the case of smaller A-scan batch numbers (1000~10,000) due to the GPU's hardware acceleration feature but it is still above 140,000 line/s for 2048-pixel and above 200,000 line/s for 1024-pixel, which is adequate enough to over-speed the camera and also leaves enough time for volume rendering.

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

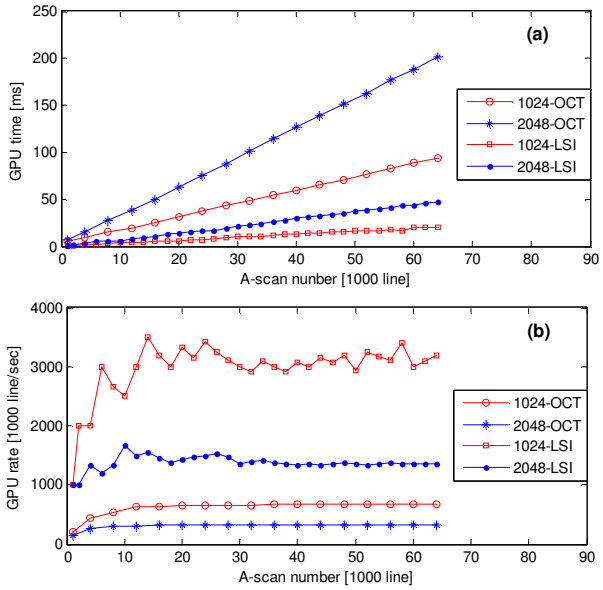

Fig. 5. (a) GPU processing time versus one-batch A-scan number; (b) GPU processing line rate versus one-batch A-scan number.

Figure 6 shows the system sensitivity roll-off at both 1024-OCT and 2048-OCT modes, where the A-scans are processed by GPU based LSI and FFT. As one can see, the background noise increases with imaging depth due to the error of linear interpolation, and this issue can be solved by a more complex zero-filling method [23], which will be implemented on GPU in our future work.

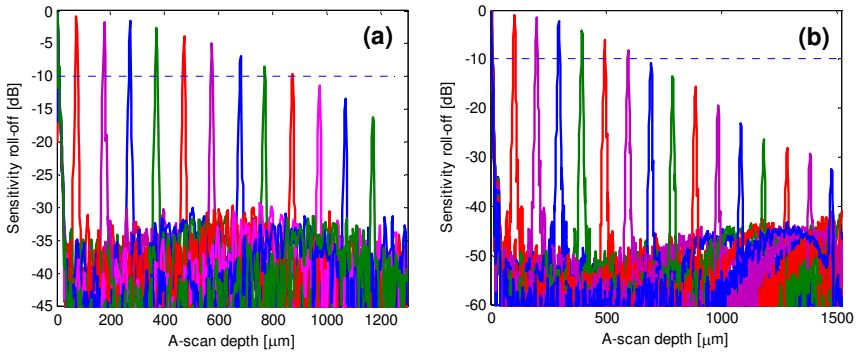

Fig. 6. System sensitivity roll-off: (a) 1024-OCT; (b) 2048-OCT.

Then we tested the actual imaging speed by performing the real-time acquisition and display of 2-D B-scan images. The target used is an infrared sensing card, as in Fig. 7. Each frame consists of 10,000 A-scans and we got 12.8 frame/s for 1024-OCT (minimum line period = 7.8μs) and 7.0 frame/s for 2048-OCT (minimum line period = 14.2 μs), corresponding to 128,000 and 70,000 A-scan/s respectively, which is limited by the CMOS camera's acquisition speed.

To demonstrate the higher acquisition speed case and evaluate the possible bus and memory contention issue, for each frame the raw data transferring-in and processing were repeated for 4 times within each frame period, while achieving the same frame rate for both OCT modes. Therefore the minimum *effective* processing speeds of 512,000 A-scan/s for

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

1024-OCT and 280,000 A-scan/s for 2048-OCT can be expected. These speeds represents more than double the currently highest acquisition speed using a CMOS camera, which is 215,000 A-scan/s for 1024-OCT [9] and 135,000 A-scan/s for 2048-OCT [3].

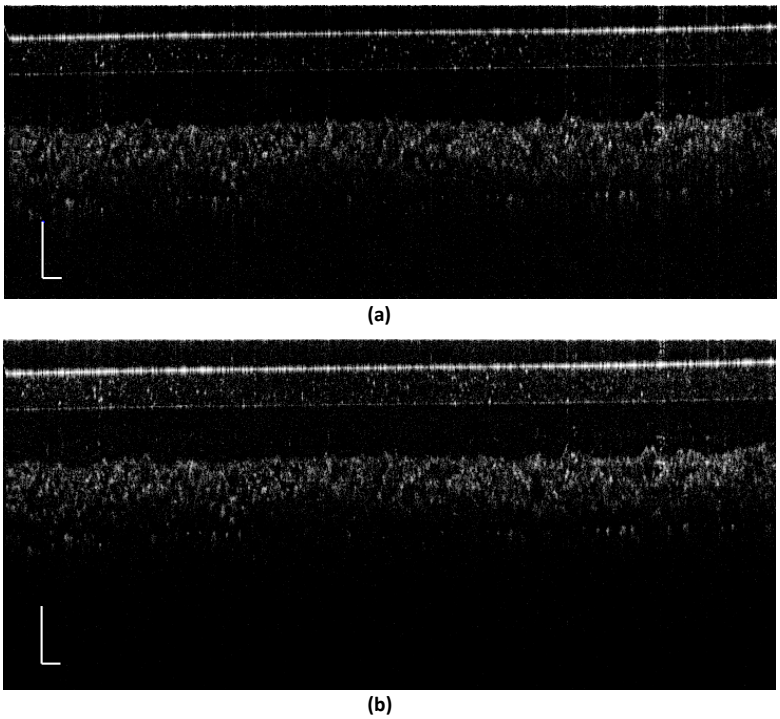

Fig. 7. B-scan images of an infrared sensing card: (a) 1024-OCT, 10,000 A-scan/frame, 12.8 frame/s; (b) 2048-OCT, 10,000 A-scan/frame, 7.0 frame/s. The scale bars represent 250μm in both dimensions.

5. Volumetric visualization by *en face* slicing

We further tested the real-time volumetric data processing and *en face* image reconstruction by running the OCT at 1024-pixel mode. The line scan rate was set to 100,000 line/second for the convenience of the synchronization. A Naval orange juice sac was used as the sample. Three different volume sizes are tested: 250 × 160 × 512 voxels (40,000 A-scans/volume); 250 × 80 × 512 voxels (20,000 A-scans/volume); 125 × 80 × 512 voxels (10,000 A-scans/volume); corresponding to a volume rate of 2.5, 5 and 10 volume/second, respectively. Figure 8 shows the *en face* slices of approximately 1mm × 1mm region in two different depths extracted from the same volumetric data and the depth difference of about 25 μm. All the A-scans of one volume were acquired and processed as one batch and remapped for *en face* slice extraction. More than one *en face* images at different depth can be quickly reconstructed and displayed simultaneously since the complete 3D data is available. As one can see, with decreasing volume size and increasing volume rate, the image quality degenerate but the major details such as cell wall are still clear enough to be visible compared with the largest volume size slices as in Fig. 8(a) and 8(b).

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

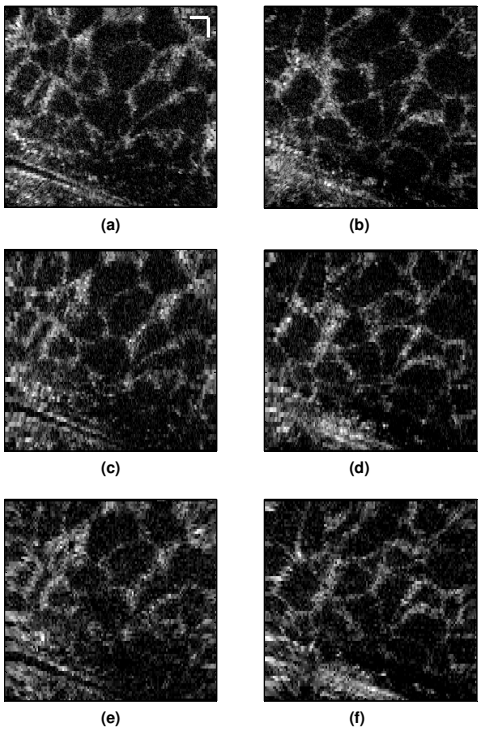

Fig. 8. *En face* slices reconstructed from real-timely acquired and processed volumetric data, the scale bar represents 100μm for all images: (a) 250 × 160 × 512 voxels; (b) from the same volume as (a) but 25 μm deeper; (c) 250 × 80 × 512 voxels; (d) from the same volume as (c) but 25 μm deeper; (e) 125 × 80 × 512 voxels; (f) from the same volume as (e) but 25 μm deeper.

Here it is necessary to compare an *en face* FD-OCT imaging with another *en face* OCT imaging technology—time-domain transverse-scanning OCT/OCM (TD-TS-OCT/OCM) which acquires only one resolution element per A-scan. A typical TD-TS-OCT/OCM system can achieve a large *en face* image size (250,000 pixels) at 4 frame/s [24], giving 1,000,000 transverse points per second. In contrast, *en face* FD-OCT has less transverse scan rate (typically <500,000 A-scan/s) because a whole spectrum has to be acquired for each A-scan. However, *en face* FD-OCT provides a complete 3D data set so multiple *en face* images at different depth of the volume can be extracted simultaneously, which is not available by TD-TS-OCT/OCM.

6. Volumetric visualization by ray-casting

Then we implemented the real-time volume rendering of continuous acquired data volume and realized the 10 volume per second 4D FD-OCT "live" image. The acquisition line rate is set to be 125,000 line/s at 1024-OCT mode. The acquisition volume size is set to be 12,500 A-scans, providing 125(X) × 100(Y) × 512(Z) voxels after the signal processing stage, which takes less than 10 ms and leaves more than 90 ms for each volume interval at the volume rate of 10 volume/s. As noticed from Fig. 6(a), the 1024-OCT has a 10-dB roll-off depth of about 0.8mm, and the background noise also increases with the depth. Therefore the optimum volume for the rendering in the visualization stage is truncated in half from the acquisition volume to be 125(X) × 100(Y) × 256(Z) voxels excluding the DC component and the low SNR portion in each A-scan. Nevertheless, the whole volume rendering is available if a larger image range is required. The image plane is set to 512 × 512 pixels, which means a total number of $512^2 = 262144$ eye rays are used to accumulate though the whole rendering volume

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

for the ray-casting process according to Eq. (4) and Eq. (5). The actual rendering time is recorded during the imaging processing to be ~3ms for half volume and ~6ms for full volume, which is much shorter than the volume interval residual (>90ms). Also for the purpose of demonstrating the higher acquisition speed case, the data transfer-in, the complete FD-OCT processing and the volume rendering of the same frame were repeated 3 times within each volume period, while still maintaining 10 volume/s real-time rendering. Therefore a minimum effective processing and visualization speeds of 375,000 A-scan/s for 1024-OCT can be expected.

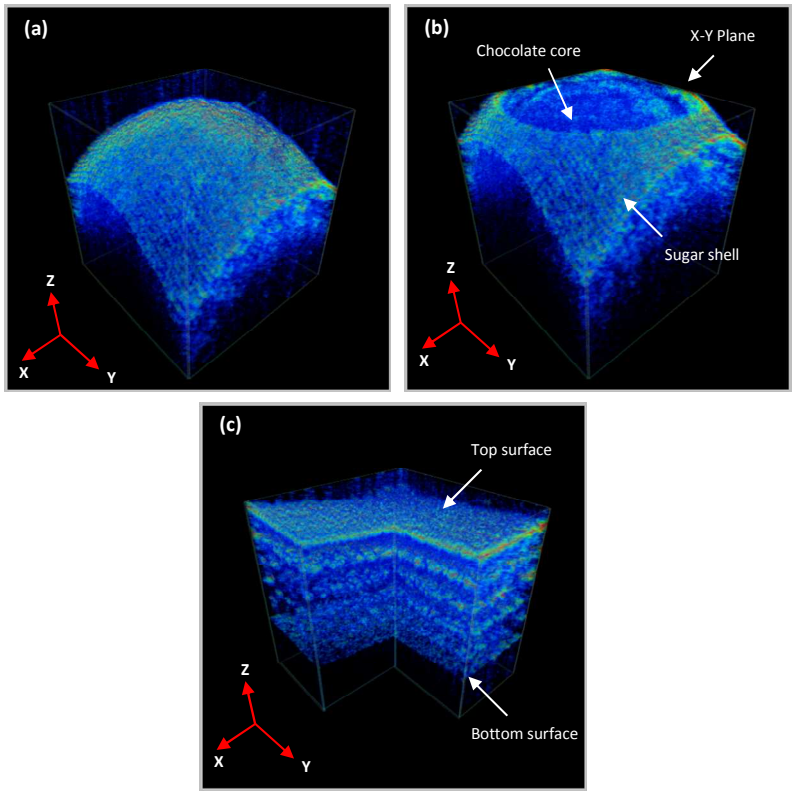

Fig. 9. (a) (Media 1) The dynamic 3D OCT movie of a piece of sugar-shell coated chocolate; (b) sugar-shell top truncated by the X-Y plane, inner structure visible; (c) a five-layer phantom.

First we tested the real-time visualization ability by imaging non-biological samples. Here the half volume rendering is applied and the real volume size is approximately 4mm × 4mm × 0.66mm. The dynamic scenarios are captured by free screen-recording software (BB FlashBack Express). Figure 9(a) presents the top surface of a piece of sugar-shell coated chocolate, which is moving up and down in axial direction with a manual translation stage. Here the perspective projection is used for the eye's viewpoint [19], and the rendering volume frame is indicated by the white lines. As played in Media 1, Fig. 9(b) shows the situation when the target surface is truncated by the rendering volume's boundary, the X-Y plane, where the sugar shell is virtually "peeled" and the inner structures of the chocolate core is clearly recognizable. Figure 9(c) illustrates a five-layer plastic phantom mimicking the retina, where the layers are easily distinguishable. The volume rendering frame in Fig. 9(c) is configured as "L" shape so the tapes are virtually "cut" to reveal the inside layer structures.

Then we implemented the *in vivo* real-time 3D imaging of a human finger tip. Figure 10(a) shows the skin and fingernail connection, the full volume rendering is applied here giving a real size of 4mm × 4mm × 1.32mm considering the large topology range of the nail

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

connection region. The major dermatologic structures such as epidermis (E), dermis (D), nail fold (NF), nail root (NF) and nail body (N) are clearly distinguishable from Fig. 10(a). Media 2 captured the dynamic scenario of the finger's vertical vibration due to artery pulsing when the finger is firmly pressing against the sample stage. The fingerprint is imaged and shown in Media 3 in Fig. 10(b), where the epithelium structures such as sweat duct (SD), stratum corneum (SC) can be clearly identified. Figure 10(c) offers a top-view of the fingerprint region in Media 4, where the surface is virtually peeled by the image frame and the inner sweat duct are clearly visible. The volume size for Fig. 10(b) and Fig. 10(c) is 2mm × 2mm × 0.66mm.

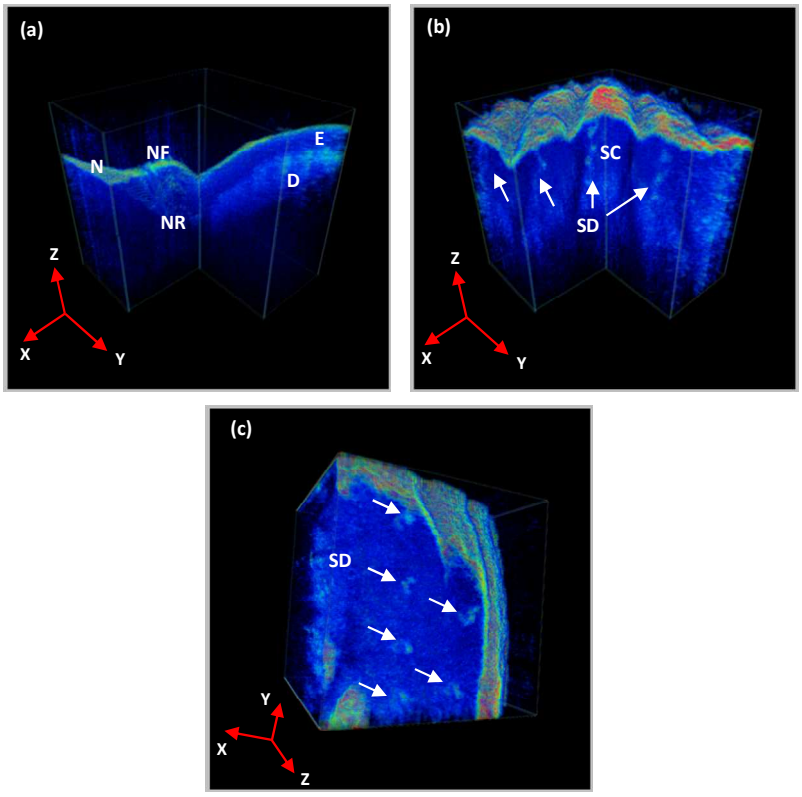

Fig. 10. *In vivo* real-time 3D imaging of a human finger tip. (a) (Media 2) Skin and fingernail connection; (b) (Media 3) Fingerprint, side-view with "L" volume rendering frame; (c) (Media 4) Fingerprint, top-view.

Finally, to make full use of the ultrahigh processing speed and the whole 3D data, we implemented multiple 2D frames real-time rendering from the same 3D data set with different model view matrix in Media 5, including side-view [Figs. 11(a, b, d, e)], top-view [Fig. 11(c)] and bottom-view [Fig. 11(f)], where Fig. 11(a) and Fig. 11(d) are actually using the same model view matrix but the later displayed with the "L" volume rendering frame to give more information of inside. All frames are rendered within the same volume period and displayed simultaneously, thus gives more comprehensive information of the target. The two vertexes with the big red and green dot indicate the same edge for each rendering volume frame.

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010

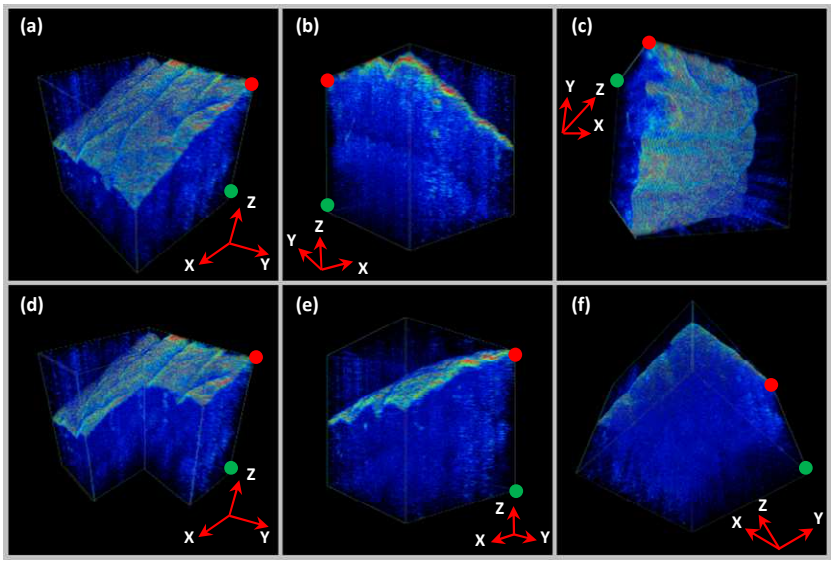

Fig. 11. (Media 5) Multiple 2D frames real-time rendering from the same 3D data set with different model view matrix.

The processing bandwidth showed in Section 4 is much higher than most of the current FD-OCT system's acquisition speed, which indicates a huge potential for improving the image quality and volume speed of real-time 3D FD-OCT by increasing the acquisition bandwidth. The GPU processing speed can be increased even higher by implementing a multiple-GPU architecture using more than one GPU in parallel. Therefore the bottleneck for 3D FD-OCT imaging would now lie in the acquisition speed.

For all the experiments described above, the only additional device required to implement the real-time high speed OCT data processing and display for most cases is a high-end graphics card which cost far less compared to the most optical setup and acquisition devices. The graphics card is a plug-and-play computer hardware without need for any optical modifications. And it is much simpler than adding a prism to build a linear-k spectrometer or developing a linear-k swept laser. The both are complicated to build and will change the overall physical behavior of the OCT system.

7. Conclusion

In conclusion, we realized GPU based real-time 4D signal processing and visualization on a regular FD-OCT system with nonlinear k-space for the first time to the best of our knowledge. An ultra-high speed linear spline interpolation (LSI) method for interpolation for λ-to-k spectral re-sampling is implemented in GPU architecture. The complete FD-OCT signal processing including interpolation for λ-to-k spectral re-sampling, fast Fourier transform (FFT) and post-FFT processing have all been implemented on a GPU. 3D Data sets are continuously acquired in real time, immediately processed and visualized by either *en face* slice extraction or ray-casting based volume rendering from 3D texture mapped in graphics memory. For standard FD-OCT systems, a GPU is the only additional hardware needed to realize this improvement and no optical modification is needed. This technique is highly cost-effective and can be easily integrated into most ultrahigh speed FD-OCT systems to overcome the 3D data processing and visualization bottlenecks.

Acknowledgments

This work was supported by National Institutes of Health (NIH) grant R21 1R21NS063131-01A1.

#125079 - $15.00 USD Received 5 Mar 2010; revised 29 Apr 2010; accepted 18 May 2010; published 19 May 2010