Real-time 3D and 4D Fourier domain Doppler optical coherence tomography based on dual graphics processing units

Yong Huang,* Xuan Liu, and Jin U. Kang

Department of Electrical and Computer Engineering, The Johns Hopkins University, 3400 North Charles Street,
Baltimore, MD 21218, USA
**yhuang60@jhu.edu*

Abstract: We present real-time 3D (2D cross-sectional image plus time) and 4D (3D volume plus time) phase-resolved Doppler OCT (PRDOCT) imaging based on configuration of dual graphics processing units (GPU). A GPU-accelerated phase-resolving processing algorithm was developed and implemented. We combined a structural image intensity-based thresholding mask and average window method to improve the signal-to-noise ratio of the Doppler phase image. A 2D simultaneous display of the structure and Doppler flow images was presented at a frame rate of 70 fps with an image size of 1000 × 1024 (X × Z) pixels. A 3D volume rendering of tissue structure and flow images—each with a size of 512 × 512 pixels—was presented 64.9 milliseconds after every volume scanning cycle with a volume size of 500 × 256 × 512 (X × Y × Z) voxels, with an acquisition time window of only 3.7 seconds. To the best of our knowledge, this is the first time that an online, simultaneous structure and Doppler flow volume visualization has been achieved. Maximum system processing speed was measured to be 249,000 A-scans per second with each A-scan size of 2048 pixels.

OCIS codes: (100.2000) Digital image processing; (100.6890) Three-dimensional image processing; (110.4500) Optical coherence tomography; (170.3890) Medical optics instrumentation.

References and links

1. W. Drexler and J. G. Fujimoto, *Optical Coherence Tomography, Technology and Applications* (Springer, 2008)
2. Z. P. Chen, T. E. Milner, S. Srinivas, X. Wang, A. Malekafzali, M. J. C. van Gemert, and J. S. Nelson, “Noninvasive imaging of *in vivo* blood flow velocity using optical Doppler tomography,” Opt. Lett. **22**(14), 1119–1121 (1997).
3. R. K. Wang, S. L. Jacques, Z. Ma, S. Hurst, S. R. Hanson, and A. Gruber, “Three dimensional optical angiography,” Opt. Express **15**(7), 4083–4097 (2007).
4. A. Mariampillai, B. A. Standish, E. H. Moriyama, M. Khurana, N. R. Munce, M. K. K. Leung, J. Jiang, A. Cable, B. C. Wilson, I. A. Vitkin, and V. X. D. Yang, “Speckle variance detection of microvasculature using swept-source optical coherence tomography,” Opt. Lett. **33**(13), 1530–1532 (2008).
5. V. J. Srinivasan, J. Y. Jiang, M. A. Yaseen, H. Radhakrishnan, W. Wu, S. Barry, A. E. Cable, and D. A. Boas, “Rapid volumetric angiography of cortical microvasculature with optical coherence tomography,” Opt. Lett. **35**(1), 43–45 (2010).
6. G. Liu, L. Chou, W. Jia, W. Qi, B. Choi, and Z. P. Chen, “Intensity-based modified Doppler variance algorithm: application to phase instable and phase stable optical coherence tomography systems,” Opt. Express **19**(12), 11429–11440 (2011).
7. Y. H. Zhao, Z. P. Chen, C. Saxer, Q. Shen, S. Xiang, J. F. de Boer, and J. S. Nelson, “Doppler standard deviation imaging for clinical monitoring of *in vivo* human skin blood flow,” Opt. Lett. **25**(18), 1358–1360 (2000).
8. Y. H. Zhao, Z. P. Chen, Z. Ding, H. Ren, and J. S. Nelson, “Real-time phase-resolved functional optical coherence tomography by use of optical Hilbert transformation,” Opt. Lett. **27**(2), 98–100 (2002).
9. Y. K. Tao, A. M. Davis, and J. A. Izatt, “Single-pass volumetric bidirectional blood flow imaging spectral domain optical coherence tomography using a modified Hilbert transform,” Opt. Express **16**(16), 12350–12361 (2008).

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

10. Z. Yuan, Z. C. Luo, H. G. Ren, C. W. Du, and Y. Pan, "A digital frequency ramping method for enhancing Doppler flow imaging in Fourier-domain optical coherence tomography," Opt. Express **17**(5), 3951–3963 (2009).
11. B. Baumann, B. Potsaid, M. F. Kraus, J. J. Liu, D. Huang, J. Hornegger, A. E. Cable, J. S. Duker, and J. G. Fujimoto, "Total retinal blood flow measurement with ultrahigh speed swept source/Fourier domain OCT," Biomed. Opt. Express **2**(6), 1539–1552 (2011).
12. H. Ren, C. Du, and Y. Pan, "Cerebral blood flow imaged with ultrahigh-resolution optical coherence angiography and Doppler tomography," Opt. Lett. **37**(8), 1388–1390 (2012).
13. S. Yazdanfar, M. D. Kulkarni, and J. A. Izatt, "High resolution imaging of *in vivo* cardiac dynamics using color Doppler optical coherence tomography," Opt. Express **1**(13), 424–431 (1997).
14. R. M. Werkmeister, N. Dragostinoff, M. Pircher, E. Götzinger, C. K. Hitzenberger, R. A. Leitgeb, and L. Schmetterer, "Bidirectional Doppler Fourier-domain optical coherence tomography for measurement of absolute flow velocities in human retinal vessels," Opt. Lett. **33**(24), 2967–2969 (2008).
15. R. K. Wang and L. An, "Doppler optical micro-angiography for volumetric imaging of vascular perfusion *in vivo*," Opt. Express **17**(11), 8926–8940 (2009).
16. B. White, M. Pierce, N. Nassif, B. Cense, B. Park, G. Tearney, B. Bouma, T. Chen, and J. de Boer, "*In vivo* dynamic human retinal blood flow imaging using ultra-high-speed spectral domain optical coherence tomography," Opt. Express **11**(25), 3490–3497 (2003).
17. G. Liu, W. J. Qi, L. F. Yu, and Z. P. Chen, "Real-time bulk-motion-correction free Doppler variance optical coherence tomography for choroidal capillary vasculature imaging," Opt. Express **19**(4), 3657–3666 (2011).
18. 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).
19. K. Zhang and J. U. Kang, "Real-time 4D signal processing and visualization using graphics processing unit on a regular nonlinear-k Fourier-domain OCT system," Opt. Express **18**(11), 11772–11784 (2010).
20. K. Zhang and J. U. Kang, "Real-time numerical dispersion compensation using graphics processing unit for Fourier-domain optical coherence tomography," Electron. Lett. **47**(5), 309–310 (2011).
21. K. Zhang and J. U. Kang, "Real-time intraoperative 4D full-range FD-OCT based on the dual graphics processing units architecture for microsurgery guidance," Biomed. Opt. Express **2**(4), 764–770 (2011).
22. Y. Watanabe, S. Maeno, K. Aoshima, H. Hasegawa, and H. Koseki, "Real-time processing for full-range Fourier-domain optical-coherence tomography with zero-filling interpolation using multiple graphic processing units," Appl. Opt. **49**(25), 4756–4762 (2010).
23. S. Van der Jeught, A. Bradu, and A. G. Podoleanu, "Real-time resampling in Fourier domain optical coherence tomography using a graphics processing unit," J. Biomed. Opt. **15**(3), 030511 (2010).
24. J. Rasakanthan, K. Sugden, and P. H. Tomlins, "Processing and rendering of Fourier domain optical coherence tomography images at a line rate over 524 kHz using a graphics processing unit," J. Biomed. Opt. **16**(2), 020505 (2011).
25. Y. Huang and J. U. Kang, "Real-time reference A-line subtraction and saturation artifact removal using graphics processing unit for high-frame rate Fourier-domain optical coherence tomography video imaging," Opt. Eng. **51**(7), 073203 (2012).
26. K. K. C. Lee, A. Mariampillai, J. X. Z. Yu, D. W. Cadotte, B. C. Wilson, B. A. Standish, and V. X. D. Yang, "Real-time speckle variance swept-source optical coherence tomography using a graphics processing unit," Biomed. Opt. Express **3**(7), 1557–1564 (2012).
27. H. Jeong, N. H. Cho, U. Jung, C. Lee, J.-Y. Kim, and J. Kim, "Ultra-fast displaying spectral domain optical Doppler tomography system using a graphics processing unit," Sensors (Basel Switzerland) **12**(6), 6920–6929 (2012).
28. NVIDIA, "NVIDIA CUDA C Programming Guide Version 4.2," (April 2012).
29. S. Kimel, L. O. Svaasand, M. Hammer-Wilson, M. J. Schell, T. E. Milner, J. S. Nelson, and M. W. Berns, "Differential vascular response to laser photothermolysis," J. Invest. Dermatol. **103**(5), 693–700 (1994).
30. T. Leng, J. M. Miller, K. V. Bilbao, D. V. Palanker, P. Huie, and M. S. Blumenkranz, "The chick chorioallantoic membrane as a model tissue for surgical retinal research and simulation," Retina **24**(3), 427–434 (2004).

1. Introduction

Optical coherence tomography (OCT) is a well-established, noninvasive optical imaging technology that can provide high-speed, high-resolution, three-dimensional images of biological samples. Since its invention in the early 1990s, OCT has been widely used for diagnosis, therapy monitoring, and ranging [1]. *In vivo* noninvasive imaging of both microcirculation and tissue structure is a hot area that has attracted significant amounts of interest since it is an indicator of biological functionality and abnormality of tissues. Pioneering work by Z. P. Chen et al. combining the Doppler principle with OCT has enabled high resolution tissue structure and blood flow imaging [2]. Since then, OCT-based flow imaging techniques have evolved into two different approaches: optical coherence angiography (OCA) to detect microvasculature [3–7] and Doppler tomography (ODT) to quantitatively measure blood flow [8–15]. In spectral domain ODT, the magnitude of Fourier

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

transformation of the spectral interference fringes is used to reconstruct cross-sectional, structural image of the tissue sample, while the phase difference between adjacent A-scans is used to extract the velocity information of the flow within the tissue sample [2,8].

Real-time imaging of tissue structure and flow information is always desirable and is becoming more urgent as fast diagnosis, therapeutic response, and intraoperative OCT image-guided intervention become established medical practices. In addition, a higher imaging speed can effectively reduce motion artifact for *in vivo* imaging and thus significantly improve the quality of ODT images [16,17]. High speed CCD camera or swept source enables OCT or ODT to have higher signal acquisition speed, or higher temporal resolution of blood flow imaging system which allows for better reconstruction of the time course of dynamic processes. However, due to the large amount of raw data generated by an OCT engine during a high-speed imaging process and heavy computation task for computer systems, real-time display is highly challenging. A graphics processing unit (GPU)-accelerated signal-processing method is a logical solution to this problem due to the way OCT data are acquired and due to the fact that they can be processed in parallel. Although researchers have reported a number of studies using GPU to real-time process and display OCT images [18–27], reports of real-time functional OCT imaging based on GPU processing—which is highly demanding and would be of great value for medical and clinical applications—have been uncommon. GPU-based speckle variance swept-source OCT (SS-OCT) [26] and 2D spectral domain Doppler OCT (SD-DOCT) [27] have recently been reported.

In this report we present real-time 3D (2D cross-sectional image plus time) and 4D (3D volume plus time) phase-resolved Doppler OCT (PRDOCT) imaging based on configuration of dual graphics processing units. The dual graphics processing units configuration offers more computation power, dynamic task distribution with more stability, and an increased software-friendly environment when further performance enhancement is required [21]. To achieve real-time PRDOCT, we developed a GPU-based phase-resolving processing algorithm; this was integrated into our current GPU-accelerated processing algorithm, which included cubic wavelength-to-wavenumber domain interpolation, numerical dispersion compensation [20], numerical reference and saturation correction [25], fast Fourier transform, log-rescaling, and soft-thresholding. These processes were performed with the first GPU. Once 4D imaging data were processed, the whole structure volume and flow volume data were transferred to the second dedicated GPU for ray-casting-based volume rendering. The 3D and 4D imaging mode can be switched easily by customized graphics user interface (GUI). For phase-resolved image processing, we combined a structure image-based mask, thresholding and an average window method to improve the signal-to-noise ratio of the Doppler phase image. Flow and structure volume rendering shares the same model view matrix—for the sake of easy visual registration when ray-casting was performed—with two different customized transfer functions. The model view matrix can be modified interactively through the GUI. This flexibility makes the interpretation of volume images easier, more reliable, and complements a single-view perspective. Real-time 2D simultaneous display of structure and flow images were presented at a frame rate of 70 fps with an image size of 1000 × 1024, corresponding to 70K raw spectra per second; To present the 3D image data set, real-time 3D volume rendering of tissue structure and flow images—each with a size of 512 × 512 pixels—were presented 64.9 ms after every volume scanning cycle where the acquired volume size was 500 × 256 × 512 (X × Y × Z). To the best of our knowledge, this is the first time online simultaneous structure and flow volume visualization have ever been reported. The theoretical maximum processing speed was measured to be 249,000 A-scans per second, which was above our current maximum imaging speed of 70,000 A-scans per second limited by the camera speed. Systematic flow phantom and *in vivo* chorioallantoic membrane (CAM) of chicken embryo imaging were performed to characterize and test our high-speed Doppler spectral domain OCT imaging platform.

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

2. Methods

2.1. System configuration

We integrated the GPU-accelerated Fourier domain PRDOCT method into our previously developed GPU-accelerated OCT data processing methods based on an in-house-developed spectral domain OCT. The hardware system configuration is shown in Fig. 1. The A-line trigger signal from the frame grabber was routed to the data acquisition (DAQ) card as the clock source to generate the waveform control signal of the scanning galvanometers. We used a line-scan camera (EM4, e2v, USA) with 12-bit depth, 70 kHz line rate, and 2048 pixels as the spectrometer detector. We used a superluminescent (SLED) light source with an output power of 10 mW and an effective bandwidth of 105 nm centered at 845 nm, which gave an axial resolution of 3.0 µm in air for the experiment. The transversal resolution was approximately 12 µm, assuming a Gaussian beam profile.

We used a quad-core @2.4 GHz Dell Precision T7500 workstation to host a frame grabber (National Instrument, PCIe-1429, PCIE-x4 interface), a DAQ card (National Instrument, PCI 6211, PCI interface) to control the galvanometer mirrors and two NVIDIA (Santa Clara, California) Geforce series GPUs: One is GTX 590 (PCIE-x16 interface, 32-stream multiprocessors, 1024 cores at 1.21 GHz, 3 GB graphics memory); the other is GTS 450 (PCIE-x16 interface, 4-stream multiprocessors, 192 cores at 1.57 GHz, 1 GB graphics memory). GTS 450 was dedicated to perform volume ray-casting and image rendering while GTX 590 was used to process all the necessary pre-volume rendering data sets for GTS 450. All the scanning control, data acquisition, image processing, and rendering were performed on this multi-thread, CPU-GPU heterogeneous computing system. A customized user interface was designed and programmed through C++ (Microsoft Visual Studio, 2008). We used computer unified device architecture (CUDA) version 4.0 from NVIDIA to program the GPU for general purpose computations [28].

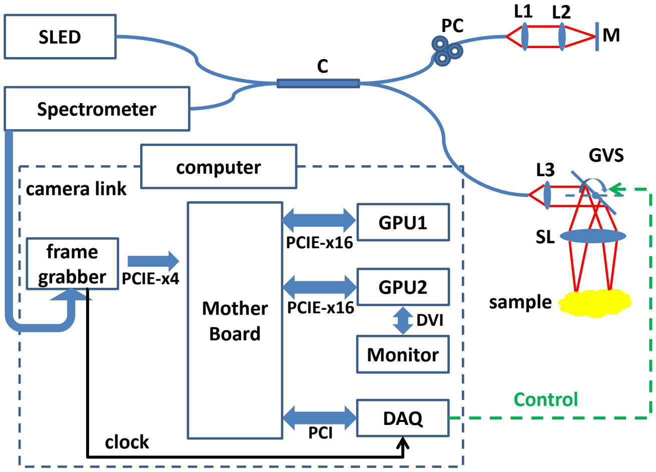

Fig. 1. System configuration: L1,L3, achromatic collimators; L2, achromatic focal lens; SL, scanning lens; C, 50:50 broadband fiber coupler; GVS, galvanometer pairs; PC, polarization controller, M, reference mirror.

2.2. Data processing

Figure 2. shows the data process flowchart of the OCT system. Thread 1 marked by a green box controls the data acquisition from frame grabber to host memory. Once one frame is ready, thread 2 marked by a yellow box copies the B-scan frame buffer to GPU1 frame buffer and controls GPU1 to perform B-frame structure and phase image processing. Once both images are ready, they are transferred to corresponding host buffers for display and to host C-

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

scan buffers for later volume rendering. Thread 2 also controls the DAQ card to generate scanning control signals to galvanometer mirrors using A-line acquisition clocks routed from the frame grabber (not illustrated in Fig. 2). When the host C-scan volume buffers are ready, thread 3 marked by a red box transfers both the structure volume and phase or velocity volume from the host to device, and commands the GPU2 to perform ray-casting-based volume rendering. Details about the implementation of structure image processing and ray-casting-based volume rendering can be found in our previously reported studies [19,21,25]. We made further improvement to the ray-casting algorithm—including a real-time, user-controlled model view matrix—to provide multiple view perspectives and customized different transfer functions to structure volume image and flow volume image. Here synchronization and hand-shake between different threads are realized through a software event-based trigger.

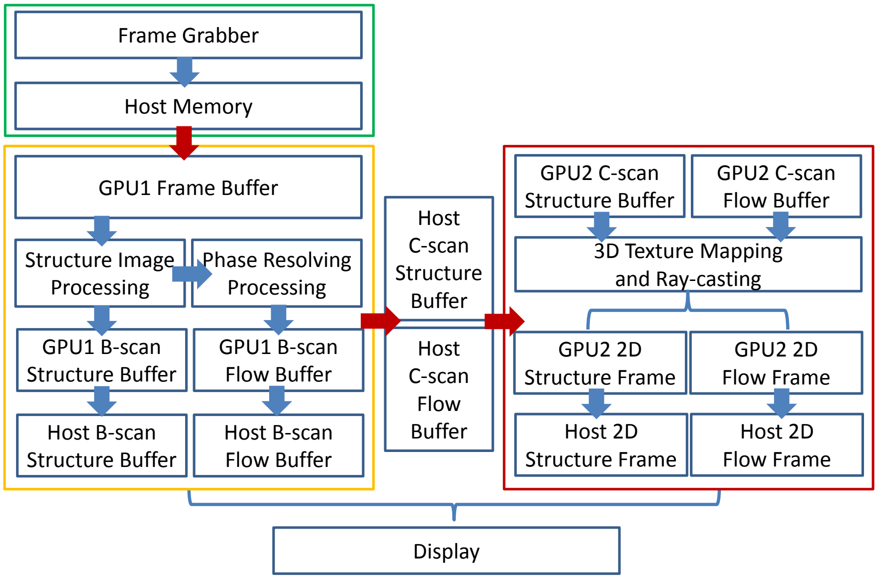

Fig. 2. Data processing flowchart of the OCT system. Solid arrows: data stream, blue indicates internal GPU or Host data flow red indicates GPU-host data flow; here the entire GPU memory buffers were allocated on global memory. Thread 1 boxed by green controls the OCT data acquisition; thread 2 boxed by yellow controls the GPU1 data processing and galvanometer mirrors; thread 3 boxed by red controls the GPU2 volume rendering processing. Synchronization and hand-shake between threads are realized through a software event-based trigger.

After structure image processing, which includes wavelength-to-wavenumber cubic spline interpolation, numerical dispersion compensation, FFT, reference and saturation correction, the complex structure image can be expressed as

$$\tilde{I}(z,x) = A(z,x)\exp[i\varphi(z,x)] \tag{1}$$

where φ(z,x) is the phase of the analytic signal. The phase difference between adjacent A-scans, n and n-1, is calculated:

$$\Delta\varphi(z,x) = \tan^{-1}\left[\frac{\mathrm{Im}[\tilde{I}(z,x_n)\bullet\tilde{I}^*(z,x_{n-1})]}{\mathrm{Re}[\tilde{I}(z,x_n)\bullet\tilde{I}^*(z,x_{n-1})]}\right] \tag{2}$$

Based on the linear relationship between phase difference between adjacent A-lines and velocity, the velocity of flow signal image can be expressed as

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

$$v(z,x)=\frac{\lambda\Delta\varphi(z,x)}{4\pi\cos(\theta)\Delta t} \tag{3}$$

In this study the camera was running at 70 kHz. We measured our system phase noise level to be 0.065 rad by measuring the standard deviation of the phase of a stationary mirror as a target. The velocity of flowing target projected to the parallel direction of the scanning beam thus was $[-14.2, -0.294] \cup [0.294, 14.2]$ mm/s. By varying the camera scanning speed, a different velocity range can be achieved based on Eq. (3).

The phase-resolving processing box in Fig. 2 consists of the following operations:

1. Generate a structure image intensity level-based binary phase-thresholding mask to filter out the background non-signal area. Most OCT images consist of a relatively large background area that carries no information. The signal intensity in the background area is usually low. By thresholding the structure image intensity, a binary mask with the same size of structure image can be generated. The value of each pixel in the mask was assigned to one if the corresponding structure pixel value has intensity level above the threshold value and to zero if the corresponding structure pixel value has intensity level below the threshold value. The threshold value was currently controlled by the user based on visual judgment. Automatic threshold value generation by statistically analyzing the image intensity will be our future modification.

2. Calculate the phase based on Eq. (2) and previously generated binary mask. If the value of a certain position in the mask was zero, we assigned zero phase value to that position instead of performing the phase calculation operation. Otherwise, the phase was calculated according to Eq. (2). This mask operation would reduce the amount of calculation load of the GPU cores.

3. Average the phase images with an averaging window to further improve the signal-to-noise ratio. Here we mapped the phase image to a certain portion of texture memory of the GPU. As the averaging operation used a lot of locality or neighboring values, texture memory would accelerate the data read speed compared to normal global memory of GPU. The window size we used here was 3 × 3, which is a commonly used window size for processing Doppler images.

4. Map the phase value to a color scheme. We used a so-called jet color map during our phase-to-color mapping process, which maps π to deep red and -π to deep blue. In between, the color varies from light red to yellow and green and then light blue. Green color corresponds to zero phase value.

5. Shrink the phase image by half in lateral and axial directions to 500 × 512 pixels to accommodate the display monitor size, which is equivalent to a final 6 × 6 average window over the phase image.

Volume rendering is a set of techniques used to display a 2D projection of a 3D discretely sampled data set, which simulates the physical vision process of the human eye in the real world and provides better visualization of the entire 3D image data than 2D slice extraction. Ray-casting is a simple and straightforward method for volume rendering. 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 [19]. To render a 2D projection of the 3D data set, a model view matrix—which defines the camera position relative to the volume—and an RGBA (red, green, blue, alpha) transfer function—which defines the RGBA value for every possible voxel value—are required. In this study the structure and flow velocity volume rendering shared the same model view matrix controlled by the user for people to easily correlate the structure and flow image. An identical jet color

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

map used when performing the phase value to color mapping with opacity equaling 0.2 was applied as the transfer function for flow velocity volume rendering. Another color map varying from black-red-yellow-green with opacity 1.0 was applied as the transfer function for structure volume rendering. Each volume data set consists of 500 × 256 × 512 voxels. Two 512 × 512 pixel size 2D projection images will be generated after volume rendering.

3. Results and Discussion

Prior to any structure and Doppler imaging, it was necessary to characterize the phase noise properties of our SD-OCT system. We calculated the phase variation by imaging a stationary mirror at 70 kHz A-scan rate without any averaging process. The result is shown in Fig. 3. The standard deviation of the Gaussian fitting curve was 65 mrad. This value incorporates both the internal system and external environmental phase noises.

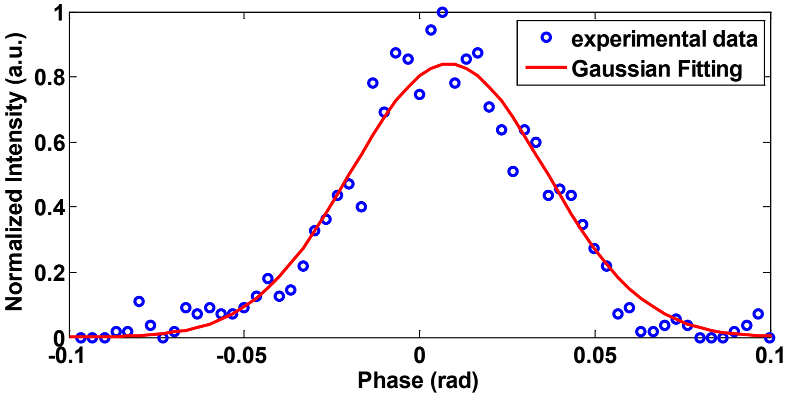

Fig. 3. Normalized phase noise measured from a stationary mirror.

3.1. Phantom experiments

To evaluate the system performance, we first performed a set of experiments using a phantom microchannel having a diameter of 300 µm with bovine milk flowing in it. The microchannel was fabricated by drilling a 300 µm channel on a transparent plastic substrate. The flow speed was controlled by a precision syringe pump. During the experiment we obtained B-scan images, each containing 1000 A-lines covering 0.6 mm .

Figure 4 shows the effect of our adopted phase-resolving process described in the Methods section. The pump speed was set at 45 µl/min with a Doppler angle of 70°, which corresponded to an actual average flow speed of 8.3 mm/s and 2.8 mm/s speed projection on the incident beam. As can be seen in Fig. 4(a), the raw image contains background having a lot of random phase variation. After filtering out the image with an intensity-based mask, Fig. 4(b) becomes much cleaner. Then an averaging window 6 × 6 was convolved with the image to form the final image, Fig. 4(c). We can clearly see the signal-to-noise ratio improvement using these processing techniques. Figure 4(d) is the result using only the averaging process. We can clearly see the advantage of combining intensity-based masking and averaging. It is also worthy pointing out that an image with a clean background or high signal-to-noise ratio is critical to the next volume rendering process, as these random and rapid variations of the phase will accumulate due to the nature of the ray-casting process.

Choosing the ideal intensity threshold value to generate the phase mask is important, as a lower threshold value would have less effect on generating a clean background, and a high threshold value would cause structure information loss—especially in situations such as when the intensity is low due to the shadowing effect of blood vessels while the flow speed is high. In this study, the threshold value was manually selected based on visual perception. Setting the pump speed at 0.8 ml/h, Fig. 5 illustrates the effect of different threshold values. The

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

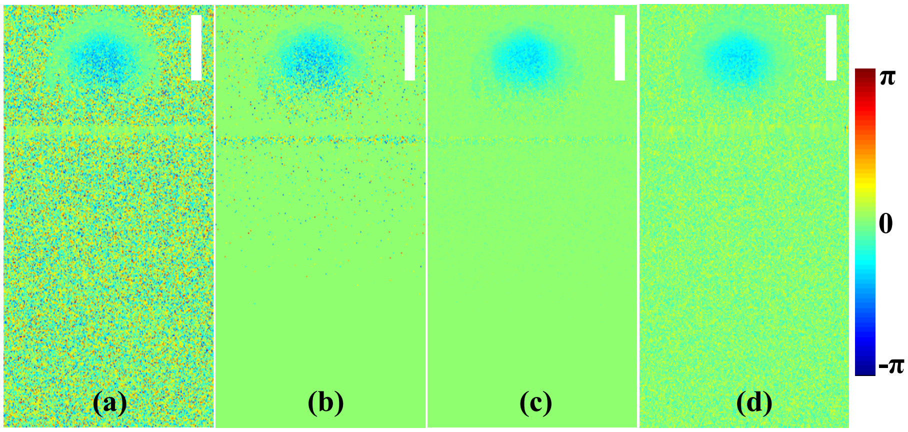

Fig. 4. Illustration for intensity-based mask and averaging of phase images: (a) raw phase image without any processing, (b) phase image after mask thresholding, (c) phase image after mask thresholding and averaging, (d) phase image after only averaging (scale bar: 300 μm).

threshold value was used after the image intensity was transformed into log-scale. As can be seen from Fig. 5(a), when the threshold value increased from 5.0 to 5.8, the background became cleaner, as expected. Figure 5(b) shows the phase profile along the red line marked in Fig. 5(a). We can see the decrease in the noise level of the background when the threshold value was increased while the signal region profile was the same; however, we can also see that the area of signal that indicates that the flow region shrank. To further evaluate the quantitative flow speed measurement of our system, we set the pump at five different speeds: 0 μl/min, 30 μl/min, 60 μl/min, 90 μl/min, and 120 μl/min. The cropped screen-captured structure and phase images to emphasize the flow region are presented in Fig. 6(a). As the

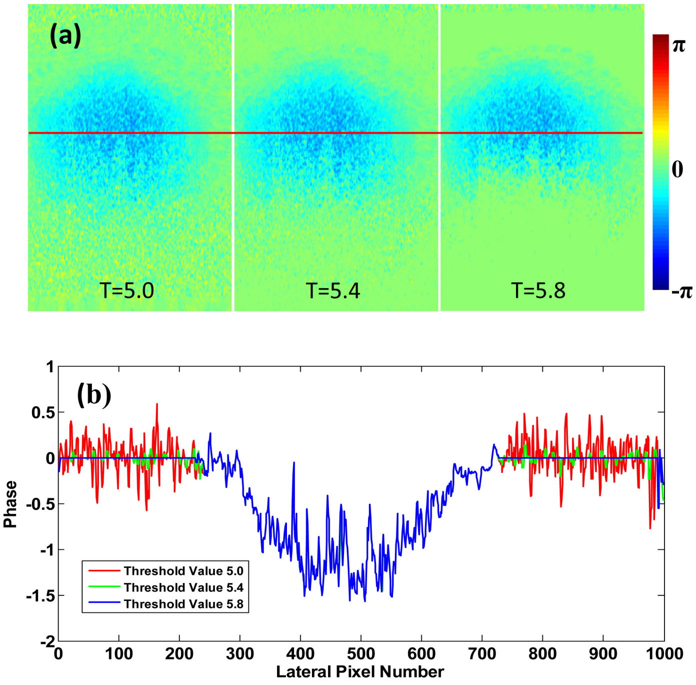

Fig. 5. (a) phantom flow phase images showing the effect of different thresholding values: 5.0, 5.4 and 5.8 (b) phase profile along the red line in (a).

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

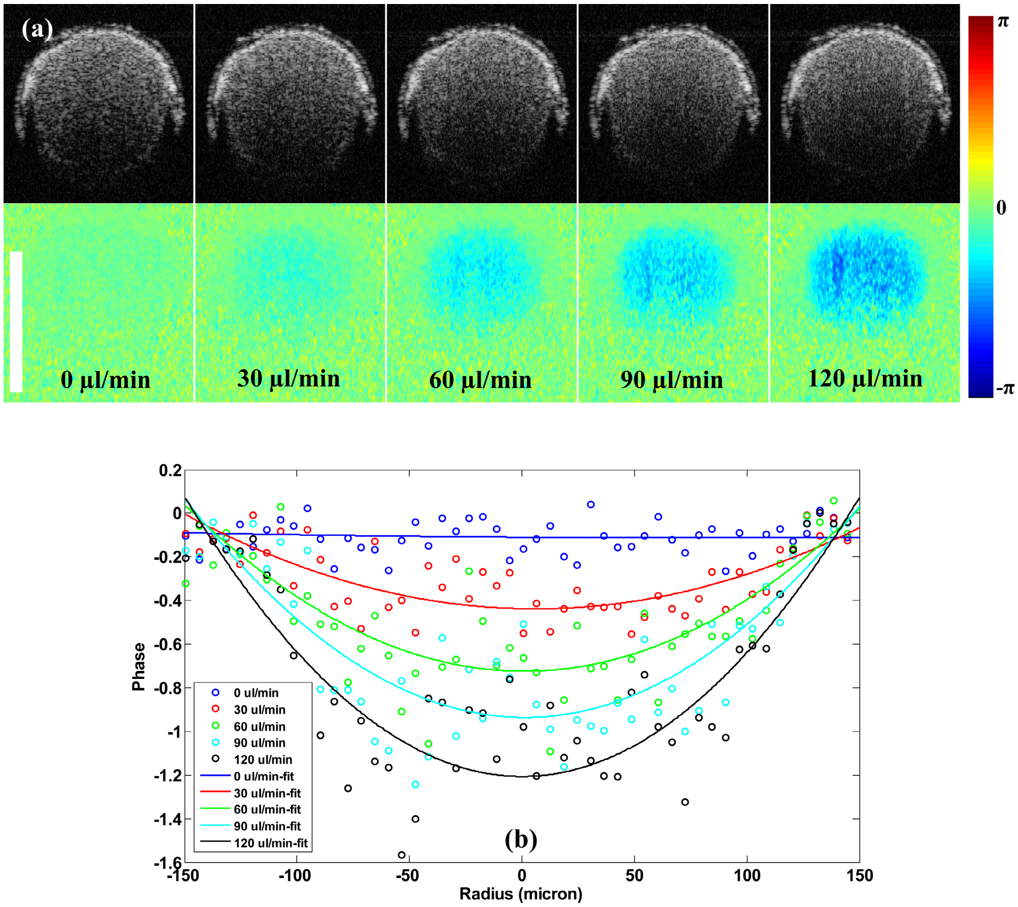

Fig. 6. (a) Zoomed screen-captured B-mode structure and phase images of a 300 µm microchannel with different flow velocities. Doppler angle: 85°. (b) Phase profile along the center of the microchannel with parabolic fitting.

pump rate increased, we can see the color varied from light blue to deep blue. Experimental phase profile along the center of the microchannel and the parabolic fitting curves are shown in Fig. 6(b). Note that at 0 µl/min pump rate, there was still a small amount of flow signal above our system phase noise level and the profile was almost flat. We suspect that might be due to the gravity caused by moving of the scattering particles.

We then performed 4D simultaneous structure and Doppler flow imaging. The camera was operating at 70 kHz A-line rate. Each B-mode image consisted of 1000 A-scans in the lateral fast X scanning direction. The volume consisted of 256 B-mode images in the lateral slow Y scanning direction. The displayed B-mode structure and flow images were 500 × 512 pixels; both were reduced by half in X and Z directions. Thus the volume data size was 500 × 256 × 512 (X × Y × Z) voxels, corresponding to a physical volume size 0.6 × 1.0 × 1.2 (X × Y × Z) mm^3. It takes 3.66s to acquire such volume data. The results are shown in Fig. 7. The red box is a screen-captured image of our customized program display zone. The name of each image was marked out at the bottom of each. To show the flexibility of our volume rendering method, two more screen-capture images—displaying only the volume velocity and structure image region under isotropic and front view—are also displayed. Since the microchannel was fabricated using a diameter 300 um drill bit on a transparent plastic substrate, the microchannel was not perfectly circular; we can clearly see from the velocity volume image that the velocity field distribution along the channel direction is not uniform. This could

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

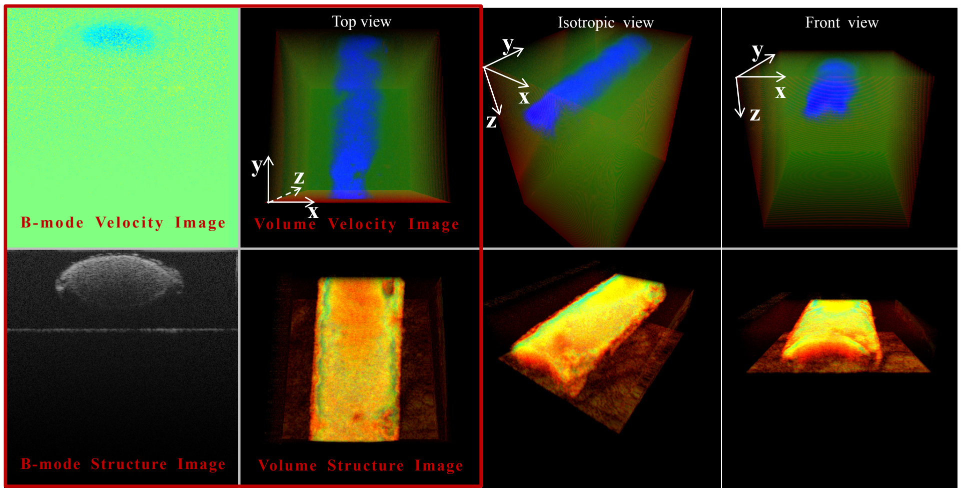

Fig. 7. Phantom volume rendering: red box indicates the screen-captured image of the program display zone and volume rendering images under top, isotropic, and front views.

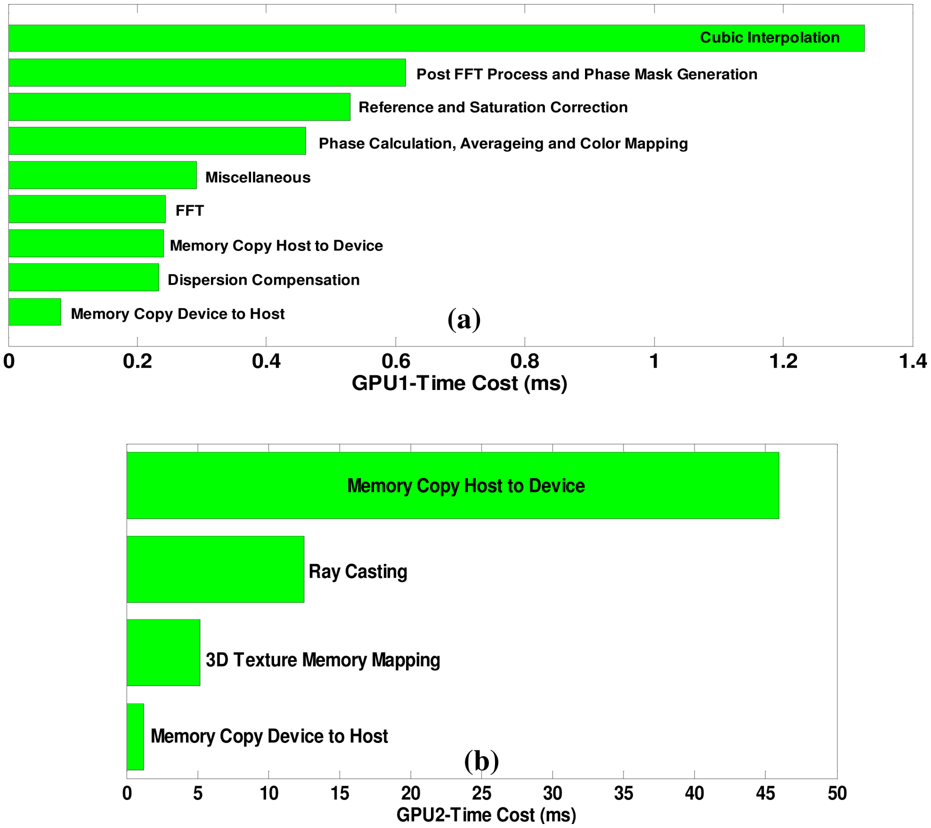

Fig. 8. Processing time measurement of all GPU kernel functions: (a) GPU1 for a B-mode image size of 1000 × 1024 pixels and (b) GPU2 for a C-mode volume size of 500 × 256 × 512 voxels.

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

essentially provide much more information than solely two-dimensional cross-sectional images. By sharing the model view matrix between the flow and structure volume, it was easy to visually correlate these two images.

The time cost of all GPU kernel functions of a previous system data acquisition, processing, and rendering setup is shown in Fig. 8. CUDA profiler 4.0 from CUDA Toolkit 4.0 was used to analyze the time cost of each kernel function of our GPU program. The data shown in Fig. 8 are based on an average value of multiple measurements. As shown in Fig. 8(a), the total time cost for a B-mode image size of 1000 × 1024, corresponding to 1000 × 2048 raw spectrum size, was 4.02 ms. Among them, phase calculation, averaging and color mapping took only 0.46 ms, which was about 11.4% of the GPU1 computation time. We did not see too much host-to-device bandwidth limit here. For the volume rendering task on GPU2, however, copying the volume data of both structure and flow from the host to the device took 45.9 ms. The strategy to reduce this memory copy cost includes future hardware upgrades into a higher speed PCI-x16 3.0 from 2.0 host-to-device interface and a more powerful CPU. Instead of copying all the volume data at one time—which is the case in our current setup—another effective solution would be to divide the copy task into multiple times for example every 20 B-frames while the acquisition was continuing to hide the latency of memory data transfer. Further GPU program optimization using two streams for GPU1 and asynchronous data transfer mode to hide the data transfer latency will be implemented in our future study. For 64bit operating systems that utilizing multiple GPUs from Tesla series can be utilized to implement peer-to-peer memory access function to bypass the host memory transfer [28]. The ray-casting of two volume data sets cost 12.5 ms. Based on the measurement, our system could provide a theoretical maximum imaging speed of 249,000 A-scans per second.

3.2. In vivo chicken embryo imaging

We further tested our system by *in vivo* imaging of chicken embryo to show the potential benefits of our system for noninvasive assessment of microcirculations within tissues. Here we used the chorioallantoic membrane (CAM) of a 15-day-aged chick embryo as a model. The CAM is a well-established model for studying microvasculature and has been used extensively to investigate the effects of vasoactive drugs, optical and thermal processes in blood vessels, as well as retina simulation [29,30]. Shown in Fig. 9 is one video frame showing real-time chicken embryo blood flow with an imaging rate of 70 fps; the video (Media 1) was played back at 30 fps. From the structure image we can clearly see the blood vessel wall, chorion membrane. In the velocity image we can clearly identify two blood vessels; one is flowing with larger speed than the other. It was also evident that blood moved at different speeds within the vessel. The magnitude of the blood flow was maximal at the center and gradually went down to the peripheral wall. From this video we can clearly observe the blood flow speed variation over time. Both vessel blood-flowing speed fields were modulated by the pulsation effect of the blood flow. C-mode imaging was achieved by scanning the focused beam across the sample surface using X-Y scanning mirrors. The physical scanning range was 2.4 × 1.5 × 1.2 (X × Y × Z) mm^3, while all the other parameters were the same as the previous phantom C-mode imaging. It took 3.7 seconds to image a volume; the volume rendering of structural and flow information were displayed right after the volume data set was ready, with a delay of only 64.9 ms, which could be further reduced. To the best of our knowledge, this is the first-time demonstration of online simultaneous volume structure and flow-rendering OCT imaging. Combining volume flow speed with structural volume images could be highly beneficial for intraoperative applications such as microvascular anastomosis and microvascular isolation. The rendering of flow volume would allow the surgeon to evaluate the surgical outcomes.

#172846 - $15.00 USD Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

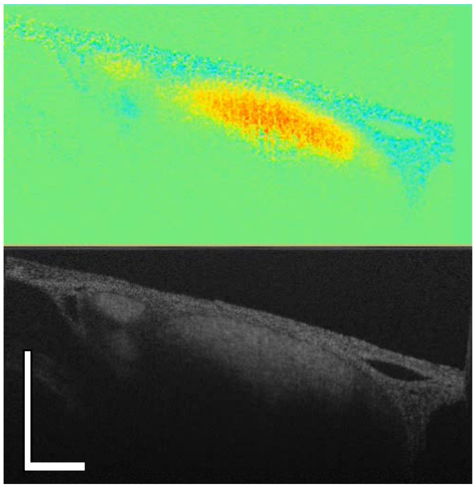

Fig. 9. Real-time video image (Media 1) showing the pulsation of blood flow of one vessel of chicken embryo membrane, imaged at 70 fps and played back at 30 fps (scale bar: 300 µm).

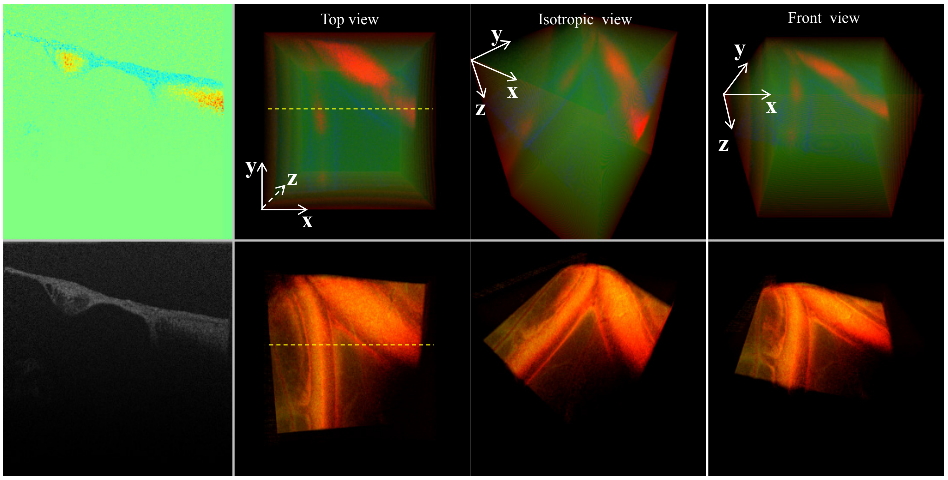

Fig. 10. Screen-captures of simultaneous flow and structure imaging of CAM under different views; B-mode images correspond to position marked by yellow dashed line on the volume image.

To resolve the Doppler phase information, the B-mode image lateral direction needs to be oversampled (see Fig. 10). For example, in our system the lateral transverse resolution was 12 µm—typical for a scanning length of 2.4 mm; the oversampling factor of 5 needs to be applied. This requires 1000 A-scans for each B-scan. In our imaging one volume consists of 256 B-frames and the camera speed was 70,000 A-scans per second; therefore, our volume imaging rate was 0.27 volumes per second, although our system could sustain a volume rendering rate of 15 volumes per second. If a higher-speed camera having 249,000 A-scans per second were used, the volume imaging rate would be 1 volume per second for the same volume size. As camera speed goes up, however, the minimum detectable flow speed will also go up. There is a trade-off between imaging speed and system flow sensitivity. The Doppler en-face preview method proposed in [11] is one possible approach to temporarily increase the volume rate before increasing the sampling area and sampling density, which will be incorporated into our system in future studies.

#172846 - $15.00 USD   Received 19 Jul 2012; revised 15 Aug 2012; accepted 16 Aug 2012; published 20 Aug 2012

4. Conclusion

In conclusion, we have demonstrated a real-time 3D and 4D phase-resolved Doppler optical coherence tomography based on dual GPUs configuration. A phase-resolving technique with structure image intensity-based thresholding mask and average window was implemented and accelerated through a GPU. Simultaneous B-mode structural and Doppler phase imaging at 70 fps with image size of 1000 × 1024 was obtained on both flow phantom and CAM model. The maximum processing speed of 249,000 A-lines per second was limited by our current camera speed. Simultaneous C-mode structural and Doppler phase imaging were demonstrated, with an acquisition time window of only 3.7s and display delay of only 64.9 ms. This technology would have potential applications in real-time fast flow speed imaging and intraoperative guidance for microsurgeries and surgical outcome evaluation.

Acknowledgments

We thank Mingtao Zhao for helpful discussions. This work was supported in part by NIH grants 1R01EY021540-01A1 and R21 1R21NS063131-01A1. Yong Huang is partially supported by the China Scholarship Council (CSC).