

## Ultra-Fast Digital Tomosynthesis Reconstruction Using General-Purpose GPU Programming for Image-Guided Radiation Therapy

www.tcrt.org

The purpose of this work is to demonstrate an ultra-fast reconstruction technique for digital tomosynthesis (DTS) imaging based on the algorithm proposed by Feldkamp, Davis, and Kress (FDK) using standard general-purpose graphics processing unit (GPGPU) programming interface. To this end, the FDK-based DTS algorithm was programmed "in-house" with C language with utilization of 1) GPU and 2) central processing unit (CPU) cards. The GPU card consisted of 480 processing cores ( $2 \times 240$  dual chip) with 1,242 MHz processing clock speed and 1,792 MB memory space. In terms of CPU hardware, we used 2.68 GHz clock speed, 12.0 GB DDR3 RAM, on a 64-bit OS. The performance of proposed algorithm was tested on twenty-five patient cases (5 lung, 5 liver, 10 prostate, and 5 head-and-neck) scanned either with a *full-fan* or *half-fan* mode on our cone-beam computed tomography (CBCT) system. For the *full-fan* scans, the projections from  $157.5^\circ$ - $202.5^\circ$  ( $45^\circ$ -scan) were used to reconstruct coronal DTS slices, whereas for the *half-fan* scans, the projections from both  $157.5^\circ$ - $202.5^\circ$  and  $337.5^\circ$ - $22.5^\circ$  ( $2 \times 45^\circ$ -scan) were used to reconstruct larger FOV coronal DTS slices. For this study, we chose  $45^\circ$ -scan angle that contained  $\sim 80$  projections for the *full-fan* and  $\sim 160$  projections with  $2 \times 45^\circ$ -scan angle for the *half-fan* mode, each with  $1024 \times 768$  pixels with 32-bit precision. Absolute pixel value differences, profiles, and contrast-to-noise ratio (CNR) calculations were performed to compare and evaluate the images reconstructed using GPU- and CPU-based implementations. The time dependence on the reconstruction volume was also tested with  $(512 \times 512) \times 16, 32, 64, 128,$  and  $256$  slices. In the end, the GPU-based implementation achieved, at most, 1.3 and 2.5 seconds to complete full reconstruction of  $512 \times 512 \times 256$  volume, for the *full-fan* and *half-fan* modes, respectively. In turn, this meant that our implementation can process  $> 13$  projections-per-second (pps) and  $> 18$  pps for the *full-fan* and *half-fan* modes, respectively. Since commercial CBCT system nominally acquires 11 pps (with 1 gantry-revolution-per-minute), our GPU-based implementation is sufficient to handle the incoming projections data as they are acquired and reconstruct the entire volume immediately after completing the scan. In addition, on increasing the number of slices (hence volume) to be reconstructed from 16 to 256, only minimal increases in reconstruction time were observed for the GPU-based implementation where from 0.73 to 1.27 seconds and 1.42 to 2.47 seconds increase were observed for the *full-fan* and *half-fan* modes, respectively. This resulted in speed improvement of up to 87 times compared with the CPU-based implementation (for 256 slices case), with visually identical images and small pixel-value discrepancies ( $< 6.3\%$ ), and CNR differences ( $< 2.3\%$ ). With this achievement, we have shown that time allocation for DTS image reconstruction is virtually eliminated and that clinical implementation of this approach has become quite appealing. In addition, with the speed achievement, further image processing and real-time applications that was prohibited prior due to time restrictions can now be tempered with.

Key words: Digital tomosynthesis; GPU programming; FDK; on-board imager; IGRT.

Justin C. Park, M.Sc.<sup>1,5</sup>  
Sung Ho Park, Ph.D.<sup>2</sup>  
Jin Sung Kim, Ph.D.<sup>3</sup>  
Youngyih Han, Ph.D.<sup>3</sup>  
Min Kook Cho, Ph.D.<sup>4</sup>  
Ho Kyung Kim, Ph.D.<sup>4</sup>  
Zhaowei Liu, Ph.D.<sup>5</sup>  
Steve B. Jiang, Ph.D.<sup>1</sup>  
Bongyong Song, Ph.D.<sup>1</sup>  
William Y. Song, Ph.D.<sup>1</sup>

<sup>1</sup>Department of Radiation Oncology, Center for Advanced Radiotherapy Technologies, University of California San Diego, La Jolla, California

<sup>2</sup>Department of Radiation Oncology, Asan Medical Center, College of Medicine, University of Ulsan, Seoul, South Korea

<sup>3</sup>Department of Radiation Oncology, Samsung Medical Center, Sungkyunkwan University School of Medicine, Seoul, South Korea

<sup>4</sup>Department of Mechanical Engineering, Pusan National University, Busan, South Korea

<sup>5</sup>Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, California

Corresponding authors:  
Dr. Jin Sung Kim, Ph.D.  
Dr. William Y Song, Ph.D.  
E-mail: mp.jinsung.kim@samsung.com  
wysong@ucsd.edu

## Introduction

In recent years, the introduction of cone-beam computed tomography (CBCT) system in treatment room settings has enabled implementation of various image guidance and adaptive radiotherapy techniques (1-24). This was possible due to the wealth of information that can be obtained from three-dimensional (3D) CBCT images obtained immediately prior to treatment, including anatomic information for setup (20-24), and CT numbers for dose calculation (25-29) and on-line/off-line re-optimization of plans (5-16). All in all, there is minimal doubt that CBCT has prominent role in current and future radiotherapy practices.

In certain clinical settings, however, CBCT may not be the optimal method for localization because the patient dose is significant (30-32) and acquisition times are long (33-35). In addition, the images may be impossible to acquire for large off-axis patient set-ups, large patients, and/or bulky immobilization devices due to mechanical collisions. This can become quite unacceptable for clinical sites with large inter-fraction variations.

As an alternative to full 3D CBCT imaging, there is growing interest to adopt digital tomosynthesis (DTS) for use in image-guided radiation therapy (IGRT) applications (36-48). This technique takes advantage of the CBCT scanning geometry and results in pseudo 3D images stacked around the imaging isocenter that are reconstructed from projections data scanned at limited projection angles (typically 40-60°). The reconstructed images generally contain enough anatomic information for IGRT applications in clinic (39, 42, 44, 46, 48), but due to its limited scan angles, the resolutions are limited in directions that are not orthogonal to the scanning geometry (36) (*i.e.*, due to pseudo 3D volume).

Current interest in DTS is due to its many advantages over full 3D CBCT, including 1) patient dose reduction ( $> \times$  times) (36, 37), 2) scan time reduction ( $> 4$  times) for faster set-up to beam-on work flow (39, 42, 44-46), 3) faster image reconstruction due to less projections data ( $> 4$  times) (36), and 4) limited scan angle allows more flexibility in scanning geometries and patient set-ups (46). In addition to these advantages, and despite the reduced image quality and resolution, numerous publications have shown that DTS contains enough image quality and anatomic information for many IGRT applications including head-and-neck (37, 39, 42, 43), lung (48), liver (37, 41, 43), prostate (37, 44, 45), and breast (46, 48) radiotherapy. In particular, the faster scan time can be of major benefit for clinical sites (or patients) that demonstrate large intra-fractional motion drifts since the influence of this effect on treatment quality will be minimized.

At current, however, there is no commercial system that allows DTS reconstruction for clinical IGRT applications, although multiple systems are available for 3D CBCT reconstruction (1, 3, 30). The current commercial 3D CBCT reconstructions can be accomplished in ~10-15 seconds after image acquisition. This is possible because the reconstruction starts as soon as the first X-ray projection is acquired. Therefore, it is anticipated that 3D DTS reconstruction would take  $< 10$  seconds at most if it is to be commercially implemented. However, a faster reconstruction is still desirable if more sophisticated image processing (such as noise, scatter reduction, *etc*) and real-time applications (47) are to be realized. To accommodate this need, we have investigated the data-parallelization approach using graphics processing unit (GPU) (49-64). In fact, there has been increasing number of studies utilizing the computational power of GPU to accelerate classically heavy and "parallel" computational tasks in radiation therapy including image reconstruction (49-55), deformable image registration (56, 57), dose calculation (58-61), treatment plan (re-) optimization (62), and most importantly here, DTS reconstruction (63, 64).

General-purpose GPU (GPGPU) computing with the Compute Unified Device Architecture (CUDA, NVIDIA Corporation, Santa Clara, CA) (65) technology is an innovative combination of computing features that enables programmers to use general C language to assign computational tasks to GPU (or *device*) as *kernels* from CPU (or *host*). Kernels are executed through a predetermined number of parallel *threads*, similar to multi-threaded programming on traditional CPUs, that can perform a large number of similar computations at once (depending on the number of processing cores in GPU card). Therefore, in contrast to multi-core CPUs, where only a few threads execute at each time, CUDA technology can process thousands of threads simultaneously enabling much faster capacity of information flow.

In this study, we present ultra-fast GPU-based DTS reconstruction scheme using the recently developed CUDA programming environment (65). To the best of our knowledge, Yan *et al.* (63, 64) has been the only group to investigate the use of GPU hardware for accelerating DTS reconstruction for RT applications. They used OpenGL application programming interface (API) to program and assign computational tasks to the graphics hardware (GPU model not stated, however). With this approach, they were able to achieve ~40 seconds to reconstruct 255 slices (64). In this paper, we will present much faster, non-graphics programming strategy using CUDA, encoded with the general C programming, which takes at maximum 2.5 seconds to complete 256 slices, virtually eliminating the time allocation for reconstruction.

Materials and Methods

FDK-Based DTS Image Reconstruction

There are many reconstruction algorithms for DTS imaging (36, 38). For cone-beam geometry, however, the well-known FDK algorithm proposed by Feldkamp, Davis, and Kress (66), is the algorithm of choice due to its simplicity and efficiency. Briefly, in FDK, the anatomical pixel information at  $(x, z)$  plane, at an arbitrary depth  $y$ , denoted by  $f(x, z/y)$ , can be calculated from the following equation (66):

$$f(x, z / y) = \frac{1}{N_o} \int_{\beta=\min \beta}^{\max \beta} \frac{d^2}{(d-s)^2} \int_{-\infty}^{+\infty} \frac{d}{\sqrt{d^2+p^2+\zeta^2}} \times R(\beta, p, \zeta) h\left(\frac{d \cdot t}{d-s} - p\right) W(p) dp d\beta$$

where  $N_o$  refers to the total number of projections,  $\beta$  refers to the angle of each projection,  $d$  refers to source-to-isocenter distance,  $s$  refers to voxel-to-detector distance,  $p$  and  $\zeta$  are the detector axes perpendicular and parallel to the axis of rotation, respectively,  $R(\beta, p, \zeta)$  corresponds to cone beam projection data,  $h(\cdot)$  refers to convolution filter, and  $W(p)$  refers to “half-fan” weighting function for stitching two opposite projections into a single large one (used in “half-fan” mode only) (67). Figure 1 illustrates the reconstruction geometry. From this equation, it is clear that FDK algorithm is composed of two major calculation-intensive steps, including 1) integral term  $\int_{-\infty}^{+\infty} \dots dp$  corresponds to the *preprocessing* stage that performs convolution filtering of all projections data, and 2) integral term  $\int_{\beta=\min \beta}^{\max \beta} \dots d\beta$  corresponds to the *back projection* of the preprocessed projections data, from all angles of measurement, to build the final 3D DTS image volume. It is this *back projection* step that requires the largest computation time, and as shown in this work, this is where GPU gains significant speed-up compared with CPU-based calculations.

Imaging Device and DTS Image Acquisition

The patient CBCT data evaluated in this study were acquired using the On-Board Imager (OBI v1.4) integrated with the Varian Trilogy unit (Varian Medical Systems, Palo Alto, CA). This system consists of a kV X-ray source (KVS) with a flat panel *a*Si detector (KVD) mounted orthogonal to the gantry axis using a robotic arm (Exact™), sharing approximately the same rotation center with the treatment unit. The flat panel detector consists of  $1024 \times 768$  pixels with pixel size of  $0.388 \times 0.388$  mm. The source-to-detector distance (SID) is approximately 150 cm, with maximum gantry rotation speed of 6 degrees-per-second.

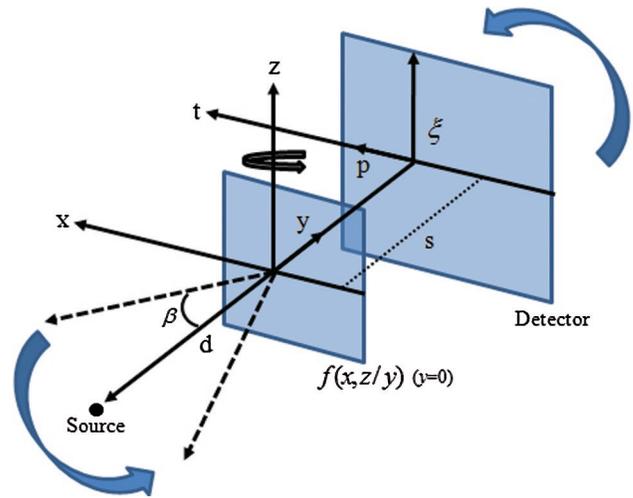


Figure 1: The cone-beam-based DTS reconstruction geometry.

There are two main scanning modes with OBI, namely 1) *full-fan*, and 2) *half-fan*. In the *full-fan* mode, the detector is centrally placed on the beam axis during scan resulting in field-of-view (FOV) of 24-cm diameter. In this mode, only ~200-degree scanning is necessary to reconstruct a full 3D CBCT volume. In the *half-fan* mode, the detector is shifted laterally by 14.8 cm to increase the FOV to a maximum of 50-cm diameter (normally set to 45-cm in clinic). Since the detector is shifted during scan, each projection image requires 180-degree opposite image pair to obtain the larger projection image used for reconstruction (67) (see Figure 2). Therefore, this mode requires ~360-degree scanning to reconstruct a full 3D CBCT volume.

Since there is no “DTS” mode available in the current OBI version (v.1.4), the DTS images were reconstructed using subset of projections data extracted from both the *full-fan*

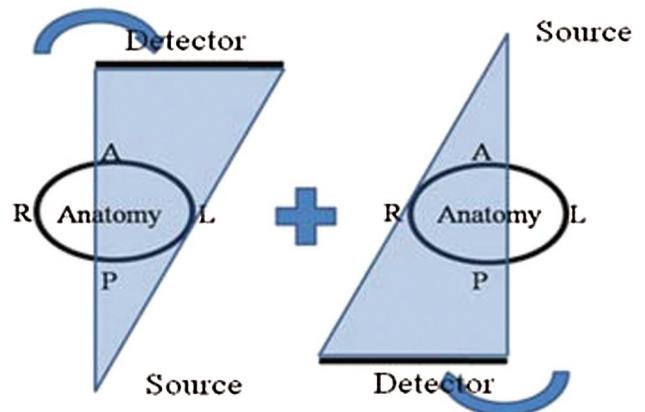


Figure 2: The *half-fan* scanning geometry of the Varian OBI system. The two smaller laterally-shifted projections with 180-degrees apart can be combined to obtain one larger projection to reconstruct a larger FOV images.

and *half-fan* scans obtained in clinic. For the *full-fan* scans, the projections from  $157.5^\circ$ - $202.5^\circ$  ( $45^\circ$ -scan) were used to reconstruct coronal DTS slices, whereas for the *half-fan* scans, the projections from both  $157.5^\circ$ - $202.5^\circ$  and  $337.5^\circ$ - $22.5^\circ$  ( $2 \times 45^\circ$ -scan) were used to reconstruct larger FOV coronal DTS slices. The speed of acquisition is constant at 1 RPM (*i.e.*,  $360^\circ$  revolution takes 60 seconds). This is a hardware limit and cannot be modified in linear accelerators. This means that for a  $45^\circ$  scan, it would take  $\sim 7.5$  seconds ( $= [45^\circ/360^\circ] \times 60$  seconds). Similarly, for a  $90^\circ$  scan, it would take twice as much to  $\sim 15$  seconds. Therefore, the total scan-to-reconstruction times would be, at maximum, the scan times plus our reported reconstruction times (*e.g.*, in Table II). For this study, we chose  $45^\circ$ -scan angle that contained  $\sim 80$  projections for the *full-fan* and  $\sim 160$  projections with  $2 \times 45^\circ$ -scan angle for the *half-fan* mode, each with  $1024 \times 768$  pixels with 32-bit precision. The reconstruction volume was set to  $512 \times 512 \times (16, 32, 64, 128, \text{ or } 256)$  resolution with 1-mm slice thickness, in order to evaluate the reconstruction time dependence on reconstruction volume. In both the CPU and GPU implementations, the “stitching and weighting” of the opposite-angle projections from the *half-fan* scans are performed as part of the *preprocessing* step in the FDK algorithm. Both “stitching” and “weighting” are necessary to merge the opposite half-fan projections and to avoid cupping artifact due to the projections overlap in the middle (67). Therefore, the algorithms are fully automated and only require uploading of raw projections data, whether acquired in the *full-fan* or *half-fan* mode. It also needs to be stated here that, if clinically implemented, the DTS imaging should ideally be acquired in the *full-fan* mode since the *half-fan* mode is 1) more prone to potential motion artifacts during longer scans, unless larger FOV is absolutely necessary, 2) the physical scan time is longer, and 3) the similar mechanical restraints issues exist as in full CBCT scanning.

#### CUDA-enabled GPU Programming

In this study, we have used the NVIDIA GTX 295 GPU card (68) (NVIDIA Corporation, Santa Clara, CA) for speeding-up DTS image reconstructions. This model consists of 480 processing cores ( $2 \times 240$  dual chip) with 1,242 MHz processing clock speed and 1,792 MB memory space, which is more than enough to store a typical DTS volume images ( $\sim 256$  MB). In terms of hardware, we used Intel Core™ i7 CPU with 2.68 GHz clock speed, 12.0 GB DDR3 RAM, on a 64-bit Window 7 OS.

The general work flow of DTS volume reconstruction is illustrated in Figure 3. The program begins by loading all of the X-ray projections data ( $\sim 3$  MB/projection,  $1024 \times 768$  with 32-bit precision) onto the CPU RAM after image acquisition ( $\sim 240$  and  $480$  MB of RAM needed for *full-fan* and *half-fan* modes, respectively), then each projection image is sent to

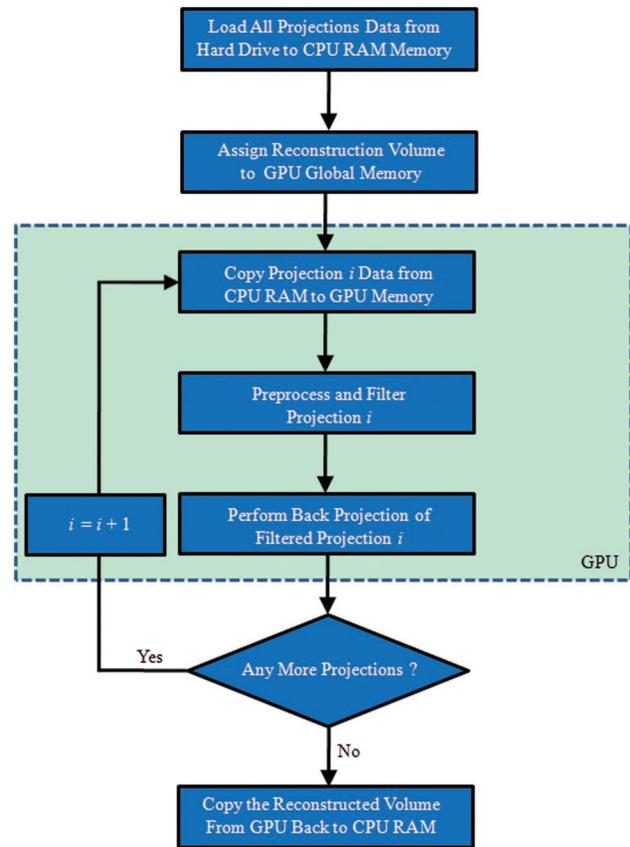
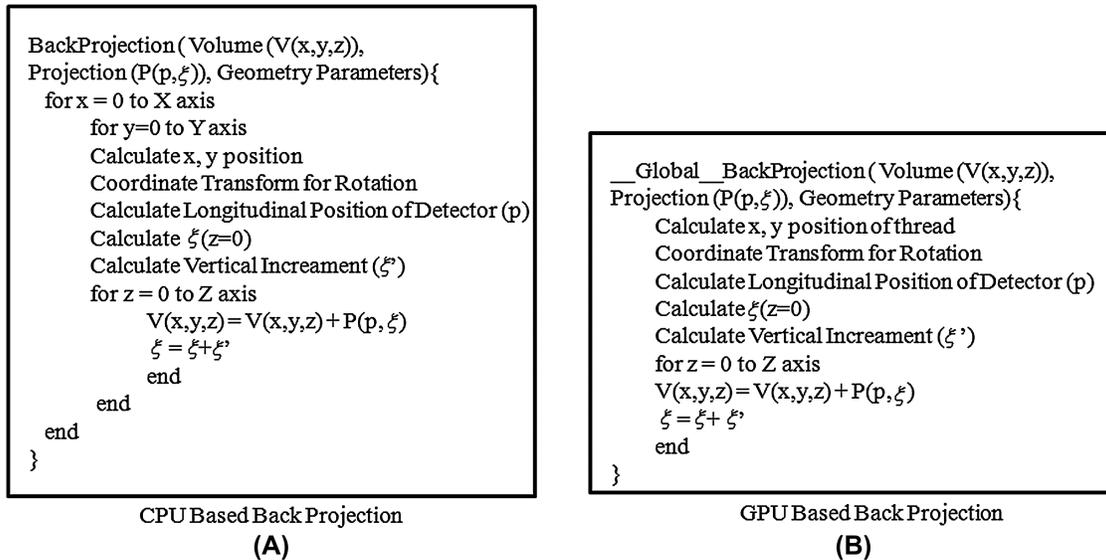


Figure 3: The GPU-based DTS reconstruction work flow.

the GPU for *preprocessing* and *back projection* (shaded in light green) computations, until all projections are used to build the pseudo 3D DTS image volume. This reconstruction volume is stored in the GPU global memory at all times. For convolution filtering of the *preprocessing* step, we have accelerated the Fourier transform calculations by utilizing the CUFFT (or CUDA FFT) library provided through CUDA. For the *back projection* step, which is the most time consuming part for CPU, we have parallelized the tasks as threads in GPU to gain massive speed-up. The *preprocessing* step, for the GPU-based implementation, includes 1) uploading of the projections data from CPU RAM to GPU memory and 2) the convolution filtering.

Figure 4 is a pseudo code that illustrates how this was implemented. In short, the back-projection volume reconstruction per  $512 \times$  number-of-slices (X and Y axis) can be parallelized into a single simultaneous calculation (*i.e.*, no loops) using GPU threads, and therefore  $(512 \times \text{number-of-slices}) \times 512$  volume calculation would be computed as 512 loops (Z axis) of  $512 \times \text{number-of-slices}$  calculations. This means that if number-of-slices = 256, then a single parallel execution of  $512 \times 256 = 131,072$  calculations is performed for each loop, for 512 times, to reconstruct the entire image volume.



**Figure 4:** The pseudo code illustrating how (A) the CPU-based and (B) GPU-based DTS reconstructions were programmed.

The reasons that “ $512 \times \text{number-of-slices}$ ” ( $\times 512$  loops) calculation are parallelized and not “ $512 \times 512$ ” ( $\times \text{number-of-slices}$  loops) instead are two folds. 1) For X and Y axis (Figure 1), coordinate transform are function of gantry angles at which projections are taken, and hence, coordinate transform calculation needs to be performed for each pixel in the X-Y plane, whereas Z axis is independent of this, and hence performing Z axis last would minimize the total calculation time. 2) In addition, since “ $512 \times \text{number-of-slices}$ ” (or X and Y axis) are performed as a single-step massive parallel calculation, increasing the number-of-slices would not increase the calculation time. That is, whether  $512 \times 16$  or  $512 \times 256$  volume calculations are chosen, it would still be a single-step massive parallel calculation.

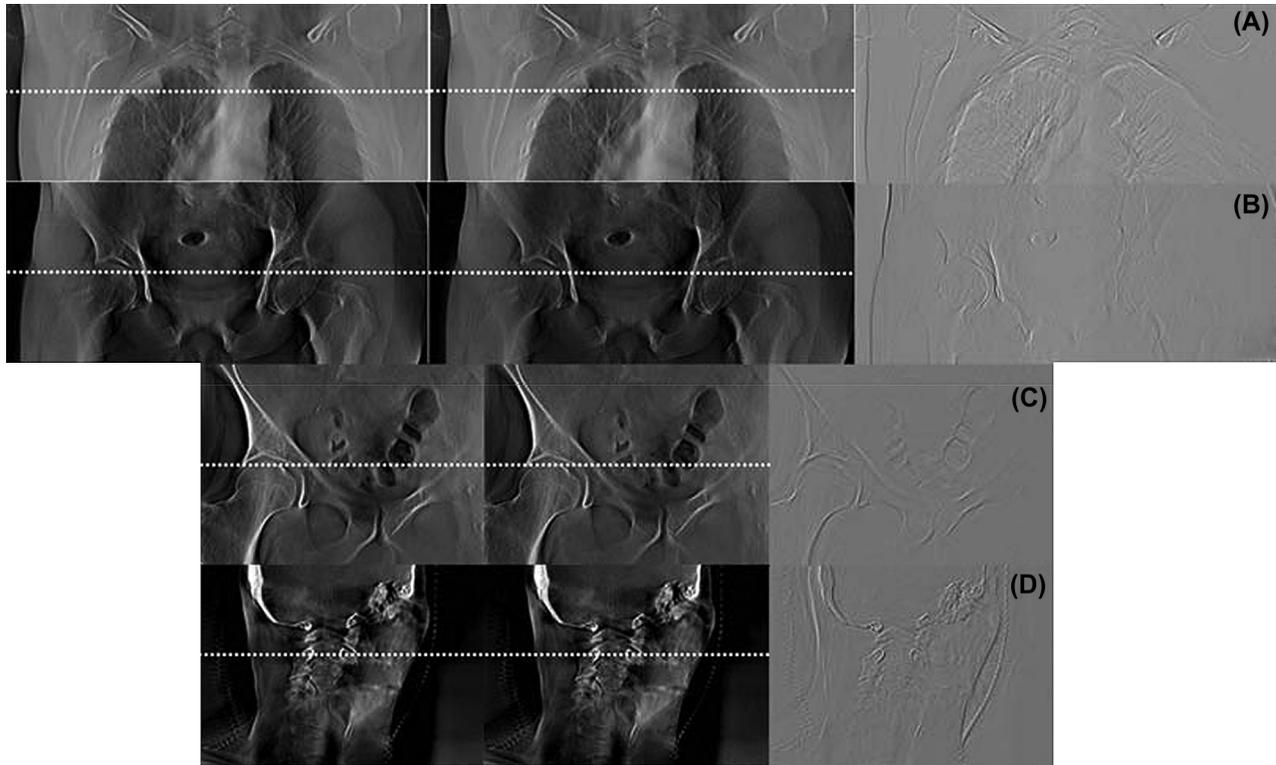
To gain further speed-up, after completing the *preprocessing* of each projections data, before *back projection*, we have re-copied each projections data that was stored in the global memory (not cached, but writable) of GPU to the texture memory (cached, but not writable within each core) of GPU since no further manipulation of the projections data are needed. This way, data-read speed is faster, stable, and most importantly, the interpolation for pixel value calculations on predefined pixel grid size ( $512 \times 512$ ) is automated by the texture memory hardware during the *back projection* stage. If not, a separate bi-linear interpolation has to be performed for each back projection process, which delays the reconstruction (as is done in CPU-based implementation). For the CPU-based method, all computational tasks, as illustrated in Figure 4, were performed using the CPU hardware described above.

#### Patient Cases and Performance Analysis

The performance of the GPU implementation was tested on twenty-five patient cases of various clinical sites (5 lung, 5 liver, 10 prostate, and 5 head-and-neck cases) scanned either with the *full-fan* or *half-fan* vendor-established standard mode on the OBI v.1.4. For lung and liver, the standard “Pelvis” *half-fan* protocol was used. For prostate, both the “Pelvis” *half-fan* and “Pelvis spot light” *full-fan* protocols were used. For head-and-neck cases, all of the head protocols supplied by OBI were used, namely “Low dose head”, “Standard dose head”, and “High quality head” *full-fan* mode. These various protocols are chosen for evaluation due to their frequent use in clinic for various treatment applications, due to their image quality differences, as well as the volume and site being scanned.

For each clinical case reconstructed using the CPU- and GPU-based methods, the overall times taken as well as each stage of reconstruction were recorded for comparison, using the internal clock measurement function provided by both CPU (clock function in C) and GPU (cut-timer function in CUDA). In addition, the times taken for the reconstruction volume of  $512 \times 512 \times (16, 32, 64, 128, \text{ and } 256 \text{ slices})$ , with 1-mm slice thickness, were recorded.

For image similarity/difference measurements, the difference maps between the two image volumes, the pixel absolute difference averages, and percentage of the absolute difference averages to the average pixel values were computed, along with line profiles. In addition, as part of the comprehensive image comparisons, the contrast-to-noise (CNR) ratio was



**Figure 5:** The CPU-based (left), the GPU-based (middle) DTS reconstructions, and the difference map (right) images of (A) lung case scanned with “Pelvis” protocol, (B) prostate case scanned with “Pelvis” protocol, (C) prostate case scanned with “Pelvis spot light” protocol, and (D) head-and-neck case scanned with “Standard dose head” protocol.

also computed and compared for both CPU- and GPU-based techniques, using the formula:

$$\text{CNR} = \frac{2 \times (S_a - S_b)^2}{\sigma_a^2 + \sigma_b^2}$$

where  $S_a$  and  $S_b$  correspond to mean-intensity of regions  $a$  (inside anatomy) and  $b$  (background), and  $\sigma_a$  and  $\sigma_b$  correspond to standard deviations of regions  $a$  and  $b$ .

## Results

### Image Quality Comparison

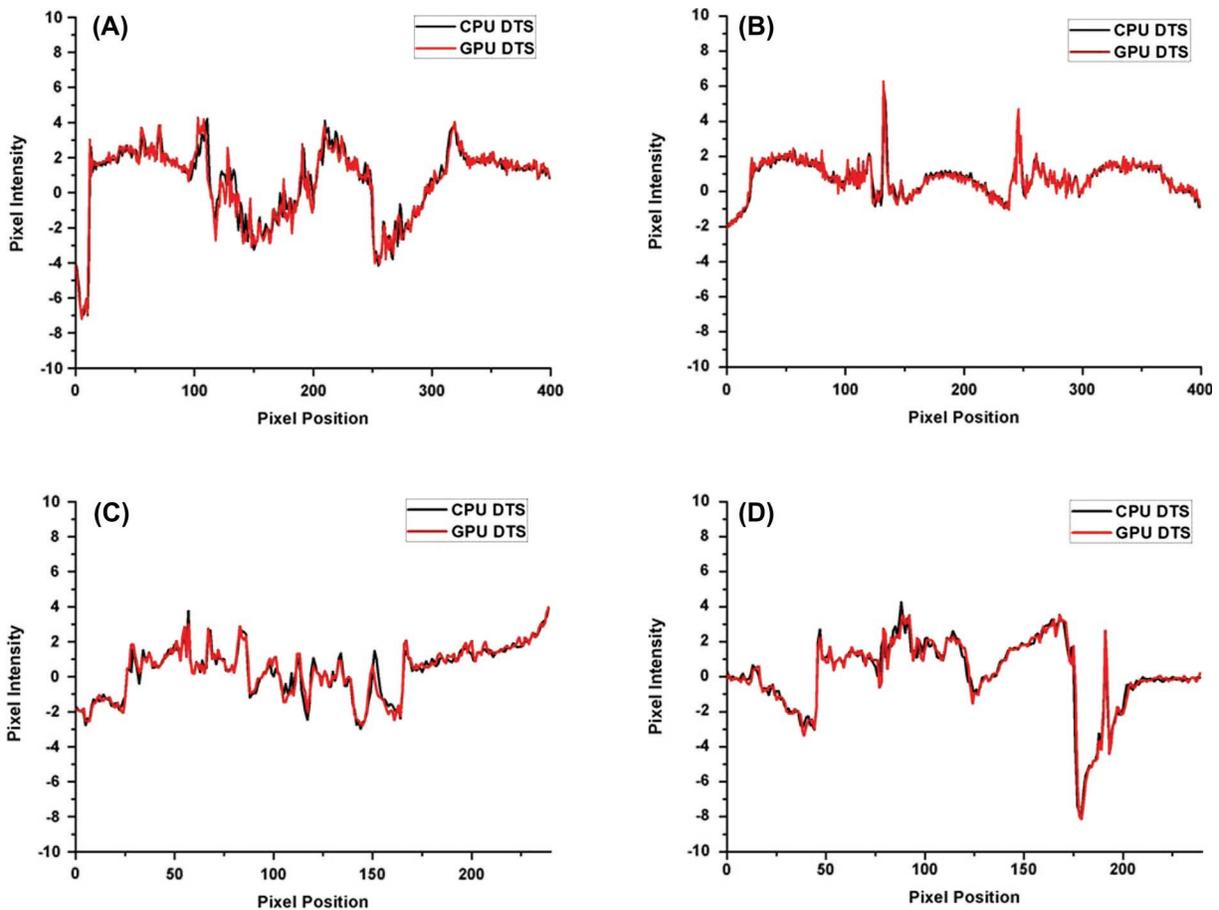
Figure 5 shows the final reconstructed DTS images using the CPU- and GPU-based implementations, as well as the difference maps, for selected clinical cases. The images shown were reconstructed with  $512 \times 512 \times 256$  resolution, with  $1 \times 1 \times 1 \text{ mm}^3$  pixel size. As can be seen, with naked eye, there are virtually no differences between the two types of implementations on image quality. The small differences that are seen is due to the differences in fast-Fourier-transform (FFT) and inverse-fast-Fourier-transform (IFFT) library

implementations in C and CUDA languages, which are beyond our control.

Figure 6 shows the selected 2D profiles comparisons between the CPU- and GPU-based implementations (from white dashed lines in Figure 5). Besides the residual differences due to FFT and IFFT calculations, the profiles are very similar. Table I lists the overall average absolute pixel differences, percentage of the absolute difference averages to the average pixel values, and CNRs and their percentage differences calculated on the 3D volume ( $512 \times 512 \times 256$ ) for all twenty-five patient cases evaluated. As the table shows, the percentage of the absolute difference averages to the average pixel values are relatively small ( $< 6.3\%$ ). In addition, the CNR calculations were  $< 2.3\%$  different.

### Speed-up Performance of GPU

Table II lists the break down of average times taken for each step in the overall DTS reconstruction process. FDK step consists of both the *preprocessing* and *back projection* steps. As can be seen, there's significant speed-up in the FDK process with the GPU-based implementation. There is significant speed-up in the *preprocessing* step ( $> 16$  times in the



**Figure 6:** Two-dimensional profiles comparing the CPU- and GPU-based DTS reconstructions, corresponding to the white dashed lines in Figure 5, for (A) lung case scanned with “Pelvis” protocol, (B) prostate case scanned with “Pelvis” protocol, (C) prostate case scanned with “Pelvis spot light” protocol, and (D) head-and-neck case scanned with “Standard dose head” protocol.

*full-fan*), but the greatest speed-up occurs in the *back projection* step where a factor of greater than 168 times is seen in the *full-fan* mode. Overall, there is 87.0 and 79.4 times speed-up in the *full-fan* and *half-fan* modes, respectively, for FDK reconstruction. As one can also infer from Table II, on average, the GPU implementation can process  $> 13$  projections-per-second (pps) and  $> 18$  pps for the *full-fan* and *half-fan* modes, respectively. It was also found that time variation from reconstruction to reconstruction is very consistent and less than 20 milli-seconds. This variation comes from multiple sources including 1) clock speed variation, 2) electronic noise in GPU hardware, and 3) variation in image display load (remember, GPU was used to display the monitor while computing reconstruction). However, the time variation of  $< 20$  ms is negligible compared with the reconstruction time of 1.5-2.5 seconds ( $< 1.5\%$  variation).

Projection loading step copies the projections data stored in the hard disk to the CPU RAM, and as can be seen in the table, this can take a few seconds also. However, this step

can be completely eliminated at the clinical implementation stage since one can theoretically upload the projections data onto the CPU RAM as the images are acquired in real-time. Therefore, it can be concluded that, as of this study, DTS reconstruction takes at most 2.5 seconds to reconstruct  $512 \times 512 \times 256$  image volume, thereby virtually eliminating the time allocation for reconstruction during DTS-based IGRT.

Figure 7 illustrates the relationship of time taken for DTS volume reconstruction as a function of number of slices, for both the CPU- and GPU-based implementations. On increasing the number of slices from 16 to 256, the time-to-reconstruct increases drastically for the CPU-based implementation where from 19.0 to 110.6 seconds and 26.1 to 195.9 seconds increase were observed for the *full-fan* and *half-fan* modes, respectively. In comparison, only minimal increases are observed for the GPU-based implementation where from 0.73 to 1.27 seconds and 1.42 to 2.47 seconds increase were observed for the *full-fan* and *half-fan* modes,

**Table I**

The average absolute difference in pixel values ( $|GPU-CPU|$ ) and its standard deviation (SD) calculated using  $512 \times 512 \times 256$  reconstruction volume between the CPU- and GPU-based implementations, for each clinical case examined.

Patient	Site	Scan Technique	Scan Mode	Avg. Pixel Value	Avg. Abs. Diff.	% Avg. Abs. Diff.	CPU CNR	GPU CNR	CNR % Diff.
1	Lung	Pelvis	Half-fan	2.77	0.068	2.45%	0.19	0.19	0.63%
2	Lung	Pelvis	Half-fan	2.28	0.069	3.03%	5.33	5.31	0.40%
3	Lung	Pelvis	Half-fan	2.32	0.068	2.93%	0.10	0.10	1.57%
4	Lung	Pelvis	Half-fan	2.59	0.068	2.63%	0.61	0.60	1.55%
5	Lung	Pelvis	Half-fan	2.74	0.069	2.52%	0.24	0.24	0.78%
6	Liver	Pelvis	Half-fan	2.61	0.138	5.29%	9.85	9.76	0.94%
7	Liver	Pelvis	Half-fan	2.52	0.137	5.44%	39.79	40.29	1.26%
8	Liver	Pelvis	Half-fan	2.27	0.092	4.05%	63.92	64.09	0.26%
9	Liver	Pelvis	Half-fan	2.81	0.072	2.56%	60.47	59.70	1.27%
10	Liver	Pelvis	Half-fan	2.92	0.091	3.12%	40.22	39.62	1.51%
11	Prostate	Pelvis	Half-fan	2.31	0.146	6.32%	40.89	40.17	1.76%
12	Prostate	Pelvis	Half-fan	2.56	0.148	5.78%	45.07	44.97	0.22%
13	Prostate	Pelvis	Half-fan	2.72	0.154	5.66%	43.11	42.76	0.80%
14	Prostate	Pelvis	Half-fan	2.71	0.145	5.35%	38.24	39.08	2.21%
15	Prostate	Pelvis	Half-fan	2.54	0.098	3.86%	66.07	65.18	1.33%
16	Prostate	Pelvis Spot Light	Full-fan	2.11	0.045	2.13%	53.33	52.97	0.68%
17	Prostate	Pelvis Spot Light	Full-fan	2.89	0.045	1.56%	56.64	55.74	1.58%
18	Prostate	Pelvis Spot Light	Full-fan	2.99	0.032	1.07%	35.03	34.55	1.36%
19	Prostate	Pelvis Spot Light	Full-fan	2.05	0.039	1.90%	40.89	40.17	1.76%
20	Prostate	Pelvis Spot Light	Full-fan	1.89	0.037	1.96%	41.90	41.97	0.16%
21	Head n Neck	Low Dose Head	Full-fan	3.32	0.077	2.32%	134.58	132.75	1.36%
22	Head n Neck	Low Dose Head	Full-fan	3.69	0.07	1.90%	111.52	110.79	0.65%
23	Head n Neck	Standard Dose Head	Full-fan	3.75	0.058	1.55%	116.99	115.42	1.35%
24	Head n Neck	High Qual. Head	Full-fan	3.75	0.056	1.49%	169.53	167.88	0.97%
25	Head n Neck	High Qual. Head	Full-fan	3.98	0.052	1.31%	141.22	139.41	1.28%

respectively (see Figure 8). The increase in times is due mainly to the fact that during the *back projection* step, more pixel scoring is needed as the slices are increased. As was illustrated in Figure 4, for the CPU-based implementation, this means the Y loop is increased and hence the increase

in time. For the GPU-based implementation, however, this means that only the number of *block* assignments need to be increased (there is no Y loop) for multi-threaded processing on GPU, therefore the minimal impact on calculation time. The differences in time slopes therefore are due mainly to the multi-threaded and non-parallel-processing approaches in the GPU- and CPU-based calculations, respectively. In addition, increasing the number of slices do not affect the *preprocessing* step, in either case, due to the need for convolution filtering of all projections independently before any back projection takes place for image volume reconstruction.

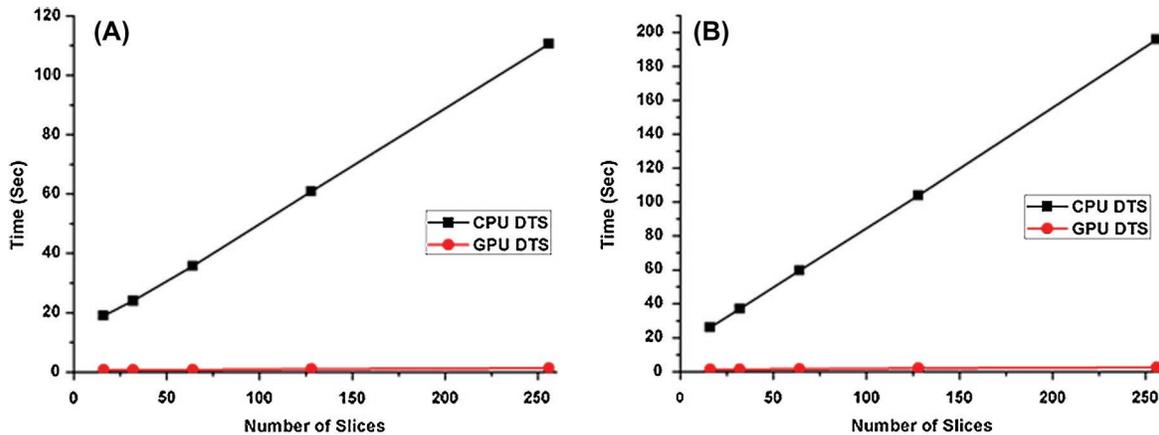
**Table II**

Break down of the average times taken to perform each stage of the FDK-based DTS reconstruction process. The reconstruction to reconstruction time variation was consistently within 20 milli-seconds or less.

Full-fan DTS ( $512 \times 512 \times 256$ )					
	Projection loading from disk to RAM	Preprocessing	Back projection	FDK	Total
CPU (Sec)	4.53	10.92	97.81	108.78	113.27
GPU (Sec)	4.53	0.67	0.58	1.25	5.78
Speed-up ( $\times$ )	1.00	16.3	168.6	87.0	19.6
Half-fan DTS ( $512 \times 512 \times 256$ )					
	Projection loading from disk to RAM	Preprocessing	Back projection	FDK	Total
CPU (Sec)	6.39	15.95	179.92	195.88	202.27
GPU (Sec)	6.39	1.33	1.14	2.47	8.86
Speed-up ( $\times$ )	1.00	12.0	157.8	79.4	22.8

**Discussion**

In this study, we have developed ultra-fast DTS image reconstruction algorithm using CUDA-enabled GPU programming. To the best of our knowledge, CUDA-based DTS programming using OBI projections has never been attempted for RT applications. The times taken for FDK volume reconstruction of  $512 \times 512 \times 256$  were less than 1.3 and 2.5 seconds for the *full-fan* and *half-fan* modes, respectively. Increasing the number of reconstructed slices had negligible impact on the overall time. In addition, these ultra-fast reconstruction times did not have



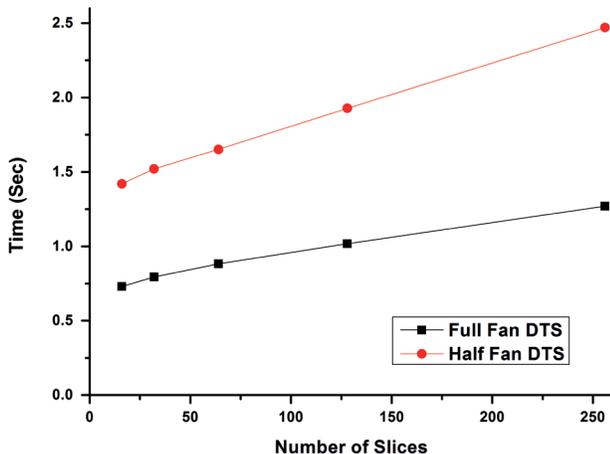
**Figure 7:** Relationship of times taken for the 3D DTS ( $512 \times 512 \times \#$ ) volume reconstruction as a function of number of slices for (A) the *full-fan*, and (B) the *half-fan* acquisitions.

negative impact on the integrity of the reconstructed images as there were negligible visual and pixel-value discrepancies between the CPU- and GPU-based implementations. With the cost of the latest GPU card being minor (around \$500-US (68)) compared to its positive impact on IGRT workflow, as shown in this work and others (49-64), our community should take full advantage of this latest gadget for all heavy computational tasks in RT that are suitable for parallel processing.

As mentioned earlier, Yan *et al.* (63, 64) has been the only group to investigate the speed-up performance of GPU hardware for DTS reconstruction, albeit using the OpenGL API. In their work, they have down-sampled the original projection images (which we didn't do) of size  $1024 \times 768$  to  $512 \times 384$  resulting in total reconstruction time of ~40 seconds. Ontesting the same data size (by down-sampling), we achieved 0.65 seconds to reconstruct  $512 \times 512 \times 256$  volume. This is about 66 times faster than their results. However, this comparison is confounded by the fact that we

used different graphics card (ours is much newer hardware and theirs was not specified in the paper) and hence may not be a fair comparison between OpenGL and CUDA. Therefore, with this work, it is not possible to conclude the superiority of one programming environment over the other.

We have used  $45^\circ$ -scan angle with ~80 projections for the *full-fan* and  $2 \times 45^\circ$ -scan angle with ~160 projections for the *half-fan* mode, each projection with  $1024 \times 768$  pixels and 32-bit precision (3.072 MB/projection), as the set input condition for our DTS reconstructions. However, depending on the clinical situation, one may need to use greater scan angle (or less) for better image quality, practicality, and quantitative information. We have found that the times taken for the GPU-based reconstruction technique proposed here approximately scales linearly with the number of projections (or scan angles) used. According to our tests, for  $90^\circ$ -scan angle with ~160 projections for the *full-fan* and  $2 \times 90^\circ$ -scan angle with ~320 projections for the *half-fan* mode, it took 2.6 and 5.0 seconds, respectively. Therefore, even with the  $90^\circ$ -scan angle, which is considered generally too large for DTS applications in RT (39, 40, 42, 44-46, 48, 63, 64), the reconstruction times are still quite fast and in the realm for on-line use.



**Figure 8:** Expanded graph plotting GPU computation times only, taken from Figure 7.

In our GPU-based FDK implementation, we have chosen to upload each projections data onto the global memory of GPU one-by-one for processing (see Figure 3). The reason for this approach, instead of uploading all projections data at once, was to be able to adapt to the realistic clinical situation where the projections data are acquired in the same one-by-one manner, and hence, our implementation allows the processing of each projection data as soon as they become available, in a streamlined process. As one can infer from Table II, on average, our implementation can process  $> 13$  projections-per-second (pps) and  $> 18$  pps for the *full-fan* and *half-fan* modes, respectively, which includes from loading the projections from hard disk to CPU RAM to completely downloading the reconstructed

volume from GPU global memory to CPU RAM. Since OBI nominally acquires 11 pps (with 1 gantry-revolution-per-minute), our GPU-based implementation is more than sufficient to handle the incoming projections data as they are acquired and reconstruct the entire volume immediately after completing the scan. This, of course, applies irrespective of the scan-angle range, including the full 3D CBCT. Therefore, theoretically at least, the wait time for image reconstruction (whether it is 3D DTS or 3D CBCT) can be completely eliminated using our approach.

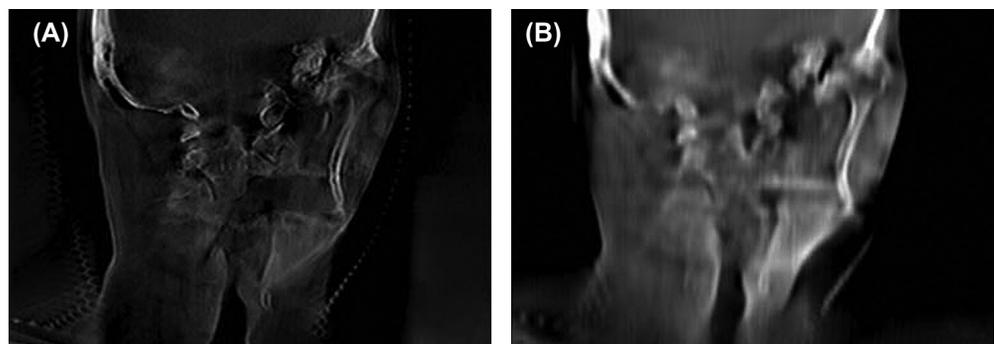
Such a stream-lined approach using GPU, could also be of tremendous help in the stationary-gantry tomosynthesis array for radiotherapy (STAR) (47) system, where multi-source carbon nanotubes (CNT) are attached to the gantry head, immediately below the multileaf collimators (MLC), for stationary DTS imaging for on-line image guidance. Since this system is intended for multiple DTS imaging within a single treatment session, there could be a benefit in utilizing our proposed GPU-based reconstruction strategy.

In this study, we have investigated DTS reconstruction using both the *full-fan* and *half-fan* modes. However, use of the *half-fan* mode should be discouraged as this would 1) increase the scan time, 2) increase the likelihood of patient pose change between opposite angle scans resulting in motion artifacts, 3) eliminate the possibility of single-breath-hold DTS scans, and 4) increase the patient dose. In addition, most IGRT application suffices with viewing/registering the target and its nearby structures (39, 42, 44, 46) making larger FOV (in *half-fan* mode), in general, a non-necessity.

For image registration of the reconstructed DTS volume (called treatment-DTS) for image guidance, one generally requires a separate reference DTS volume (called reference-DTS) to be constructed (40, 42, 45), although the planning CT itself has been used successfully (48). This is because the information presented in pseudo-3D DTS images is different from the full 3D CT images. To reconstruct reference-DTS volume, one needs to first construct digitally reconstructed radiographs (DRR) from the planning CT, in cone-beam

geometry, using ray-tracing technology (63, 69), simulating virtual OBI projections, each with the same gantry angle as the actual projections to be taken. Once the DRR projections are constructed, it is matter of using the same FDK algorithm, used in this study, to reconstruct the reference-DTS volume. Therefore, the image quality of the reference DTS would depend directly on the resolution and image quality of the planning CT. Figure 9 shows an example treatment- and reference-DTS slice for a head-and-neck patient case examined in this study. Planning CT had  $1 \times 1 \times 2.5$ -mm resolution, thus the cause of blurriness in Figure 9B. With the current implementation on GPU, it takes  $\sim 140$  milli-seconds-per-DRR calculation with  $1024 \times 768$  pixel resolution. Therefore, to reconstruct reference-DTS (DRR calculation + FDK) from 80 DRR projections for a *full-fan* mode, it takes about 12.5 seconds. Since time constraints on calculating reference-DTS volume is not as strict as the treatment-DTS, this achievement is reasonable for routine use in clinic. With the reference- and treatment-DTS calculators fully developed, we plan to evaluate the effectiveness of DTS-guided IGRT on various clinical sites and develop adaptive protocols.

The current maximum gantry rotation speed is set to 6 degrees/second (*i.e.*, 1 rpm), as mentioned above. This means that, for a  $45^\circ$ -scan angle, it would take 7.5 seconds to complete image acquisitions for a *full-fan* scan. If the reconstruction is performed immediately after, then it would take  $< 9$  seconds from acquisition to volume reconstruction. With rapid automatic-to-manual image registration (42) ( $< 30$  seconds), corresponding couch shift ( $< 10$  seconds), beam loading ( $< 10$  seconds), and beam delivery using the latest volumetric modulated arc therapy (VMAT) (70) technology, which can deliver a single treatment fraction in generally less than 2 minutes (70, 71), it is quite possible to treat a patient from DTS scan to completion of radiation delivery in under  $\times$  minutes. With intra-fraction motions and drifts being a major concern as the radiation is being delivered (33-35), this new work flow would ensure that patient pose is (more likely) maintained during the entire treatment fraction. In addition, of course, patient throughput will be naturally increased, an advantage that requires no further elaboration.



**Figure 9:** The DTS images reconstructed using (A) the OBI projections (treatment-DTS) and (B) the DRR projections constructed from the planning CT images (reference-DTS).

## Conclusions

In this study, an ultra-fast reconstruction technique for DTS based on the FDK algorithm using CUDA-based GPU programming is proposed. The performance of proposed algorithm was tested on twenty-five patient cases (5 lung, 5 liver, 10 prostate, and 5 head-and-neck) scanned either with a full-fan or half-fan mode on the OBI CBCT system. The GPU-based implementation achieved, at most, 1.3 and 2.5 second to complete full reconstruction of  $512 \times 512 \times 256$  volume, for the full-fan and half-fan modes, respectively. This resulted in speed improvement of  $> 87$  times compared with the central processing unit (CPU)-based implementation, with visually identical images and small pixel-value and CNR differences. With this achievement, we have shown that time allocation for DTS image reconstruction is virtually eliminated and that clinical implementation of this approach has become quite appealing. In addition, with the speed achievement, further image processing and real-time applications (47) that may have been prohibited prior due to time restrictions can now be tempered with.

## Conflicts of Interest

There is no conflict of interest.

## Acknowledgments

JSK & SHP are partially supported by the Nuclear Research & Development Program of National Research Foundation of Korea (NRF) Grant No 2010-0017477 & 2010-0012164, respectively, provided by Ministry of Education, Science & Technology (MEST).

## References

- Jaffray, D. A., Siewerdsen, J. H., Wong, J. Flat-panel cone-beam computed tomography for image-guided radiation therapy. *Int J Radiat Oncol Biol Phys* 53, 1337-1349 (2002).
- Jaffray, D. A. Emergent technologies for 3-dimensional image-guided radiation delivery. *Semin Radiat Oncol* 15, 208-216 (2005).
- Amies, C., Bani-Hashemi, A., Celi, J. C. A multi-platform approach to image guided radiation therapy (IGRT). *Med Dosim* 31, 12-19 (2006).
- Yan, D., Vicini, F., Wong, J. Adaptive radiation therapy. *Phys Med Biol* 42, 123-32 (1997).
- Xing, L., Thorndyke, B., Schreibmann, E. Overview of image-guided radiation therapy. *Med Dosim* 31, 91-112 (2006).
- Song, W. Y., Schaly, B., Bauman, G. Image-guided adaptive radiation therapy (IGART): Radiobiological and dose escalation considerations for localized carcinoma of the prostate. *Med Phys* 32, 2193-2203 (2006).
- Song, W. Y., Wong, E., Bauman, G. Dosimetric evaluation of daily rigid and non rigid geometric correction strategies during on-line image-guided radiation therapy (IGRT) of prostate cancer. *Med Phys* 34, 352-365 (2007).
- Mackie, T. R., Kapatoes, J., Ruchala, K. Image guidance for precise conformal radiotherapy. *Int J Radiat Oncol Biol Phys* 56, 89-105 (2003).
- Mohan, R., Zhang, X. D., Wang H. Use of deformed intensity distributions for on-line modification of image-guided IMRT to account for interfractional anatomic changes. *Int J Radiat Oncol Biol Phys* 61, 1258-1266 (2005).
- Wu, C., Jeraj, R., Olivera, G. H. Re-optimization in adaptive radiation therapy. *Phys Med Biol* 47, 3181-3195 (2002).
- Wu, C., Jeraj, R., Lu, W. Fast treatment plan modification with an over-relaxed Cimmino algorithm. *Med Phys* 31, 191-200 (2004).
- Birkner, M., Yand, D., Alber, D. Adapting inverse planning to patient and organ geometrical variation: algorithm and implementation. *Med Phys* 30, 2822-2831 (2003).
- Wu, Q., Liang, J., Yan, D. Application of dose compensation in image-guided radiotherapy of prostate cancer. *Phys Med Biol* 51, 1405-1419 (2006).
- Wu, Q., Thongphiew, D., Wang, Z. On-Line re-optimization of prostate IMRT plans for adaptive radiation therapy. *Phys Med Biol* 53, 673-691 (2008).
- Wu, W., Chi, Y., Chen, P. Y. Adaptive replanning strategies accounting for head and neck IMRT. *Int J Radiat Oncol Biol Phys* 75, 924-932 (2008).
- de la Zerda, A., Armbruster, B., Xing, L. Formulating adaptive radiation therapy (ART) treatment planning into a closed-loop control framework. *Phys Med Biol* 52, 4137-4153 (2007).
- Paquin, D., Levy, L., Xing, L. Multiscale registration of planning CT and daily cone beam CT images for adaptive radiation therapy. *Med Phys* 36, 4-11 (2009).
- Lee, L., Le, Q. T., Xing, L. Retrospective IMRT dose reconstruction base don cone-beam CT and MLC log-file. *Int J Radiat Oncol Biol Phys* 70, 634-644 (2008).
- Lu, W. G., Chen, M., Chen, Q. Adaptive fractionation therapy: I. Basic concept and strategy. *Phys Med Biol* 53, 5495-5511 (2008).
- Fu, W. H., Yang, Y., Yue, N. J. A cone beam CT-guided online modification technique to correct interfractional anatomic changes for prostate cancer IMRT treatment. *Phys Med Biol* 54, 1691-1703 (2009).
- Bissonnette, J. P., Purdie, T. G., Higgins, J. A. Cone-beam computed tomographic image guidance for lung cancer radiation therapy. *Int J Radiat Oncol Biol Phys* 73, 927-934 (2009).
- Fuller, C. D., Scarbrough, T. J., Sonke, J. J. Method comparison of automated matching software-assisted cone-beam CT and stereoscopic kilovoltage x-ray positional verification image-guided radiation therapy for head and neck cancer: a prospective analysis. *Phys Med Biol* 54, 7401-7415 (2009).
- Sonke, J. J., Rossi, M., Wolthaus, J. Frameless stereotactic body radiotherapy for lung cancer using four-dimensional cone beam CT guidance. *Int J Radiat Oncol Biol Phys* 74, 567-574 (2009).
- Case, R. B., Sonke, J. J., Moseley, D. J. Inter- and intra-fraction variability in liver position in non-breath-hold stereotactic body radiotherapy. *Int J Radiat Oncol Biol Phys* 75, 302-308 (2009).
- Yoo, S., Yin, F. F. Dosimetric feasibility of cone-beam CT-based treatment planning compared to CT-based treatment planning. *Int J Radiat Oncol Biol Phys* 66, 1553-1561 (2006).
- Yang, Y., Schreibmann, E., Li, T. Evaluation of on-board kV cone beam (CBCT)-based dose calculation. *Phys Med Biol* 52, 685-705 (2007).
- Richter, A., Hu, Q., Steglich, D. Investigation of the usability of cone beam CT data sets for dose calculation. *Radiother Oncol* 3, 1-13 (2008).
- Guan, H., Dong, H. Dose calculation accuracy using cone-beam CT (CBCT) for pelvic adaptive radiotherapy. *Phys Med Biol* 54, 6239-6250 (2009).
- Hatton, J., McCurdy, B., Greer, P. B. Cone beam computerized tomography: the effect of calibration of the Hounsfield unit number to electron density on dose calculation accuracy for adaptive radiation therapy. *Phys Med Biol* 54, N329-346 (2009).

30. Song, W. Y., Kamath, S., Ozawa, S. A dose comparison study between XVI and OBI CBCT systems. *Med Phys* 35, 480-486 (2008).
31. Downes, P., Jarvis, R., Radu, E. Monte Carlo simulation and patient dosimetry for a kilovoltage cone-beam CT unit. *Med Phys* 36, 4156-4167 (2009).
32. Ding, G. X., Coffey, C. W. Radiation dose from kilovoltage cone beam computed tomography in an image-guided radiotherapy procedure. *Int J Radiat Oncol Biol Phys* 73, 610-617 (2009).
33. Xu, F., Wang, J., Bai, S. Detection of intrafractional tumour position error in radiotherapy utilizing cone beam computed tomography. *Radiother Oncol* 89, 311-319 (2008).
34. Ghilezan, M. J., Jaffray, D. A., Siewerdsen, J. H. Prostate gland motion assessed with cine-magnetic resonance imaging (cine-MRI). *Int J Radiat Oncol Biol Phys* 62, 406-417 (2005).
35. Kerkhof, E. M., van der Put, R. W., Raaymakers, B. W. Intrafraction motion in patients with cervical cancer: The benefit of soft tissue registration using MRI. *Radiother Oncol* 93, 115-121 (2009).
36. Dobbins, J. T., Godfrey, D. J. Digital x-ray tomosynthesis: current state of the art and clinical potential. *Phys Med Biol* 48, R65-R106 (2003).
37. Godfrey, D. J., Yin, F. F., Oldham, M. Digital tomosynthesis with an on-board kilovoltage imaging device. *Int J Radiat Oncol Biol Phys* 65, 8-15 (2006).
38. Godfrey, D. J., McAdams, H. P., Dobbins, J. T. Optimization of the matrix inversion tomosynthesis (MITS) impulse response and modulation transfer function characteristics for chest imaging. *Med Phys* 33, 655-667 (2006).
39. Wu, Q. J., Godfrey, D. J., Wang, Z. On-board patient positioning for head-and-neck IMRT: comparing digital tomosynthesis to kilovoltage radiography and cone-beam computed tomography. *Int J Radiat Oncol Biol Phys* 69, 598-606 (2007).
40. Godfrey, D. J., Ren, L., Yan, H. Evaluation of three types of reference image data for external beam radiotherapy target localization using digital tomosynthesis (DTS). *Med Phys* 34, 3374-3384 (2007).
41. Maurer, J., Godfrey, D. J., Wang, Z. On-board four-dimensional digital tomosynthesis: first experimental results. *Med Phys* 35, 3574-3583 (2008).
42. Ren, L., Godfrey, D. J., Yan, H. Automatic registration between reference and on-board digital tomosynthesis images for positioning verification. *Med Phys* 35, 664-672 (2008).
43. Ren, L., Wu, Q. J., Godfrey, D. J. A novel digital tomosynthesis (DTS) reconstruction method using a deformation field map. *Med Phys* 35, 3110-3115 (2008).
44. Yoo, W., Wu, Q. J., Godfrey, D. Clinical evaluation of positioning verification using digital tomosynthesis and bony anatomy and soft tissues for prostate image-guided radiotherapy. *Int J Radiat Oncol Biol Phys* 73, 296-305 (2009).
45. Mestrovic, A., Nichol, A., Clark, B. G. Integration of on-line imaging, plan adaptation and radiation delivery: proof of concept using digital tomosynthesis. *Phys Med Biol* 54, 3803-3819 (2009).
46. Zhang, J., Wu, Q. J., Godfrey, D. Comparing digital tomosynthesis to cone-beam CT for position verification in patients undergoing partial breast irradiation. *Int J Radiat Oncol Biol Phys* 73, 952-957 (2009).
47. Maltz, J. S., Sprenger, F., Fuerst, J. Fixed gantry tomosynthesis system for radiation therapy image guidance based on a multiple source x-ray tube with carbon nanotube cathodes. *Med Phys* 36, 1624-1636 (2009).
48. Winey, B., Cormack, R., Zygmanski, P. Registration of On-Board Digital Tomosynthesis and Planning CT for Partial Breast Irradiation Patient Setup Verification with Surgical Clips. *Med Phys* 35, 2913 (2008).
49. Xu, F., Mueller, K. Real-time 3D computed tomographic reconstruction using commodity graphics hardware. *Phys Med Biol* 52, 3405-3419 (2007).
50. Sharp, G. C., Kandasamy, N., Singh, H. GPU-based streaming architectures for fast cone-beam CT reconstruction and demons deformable registration. *Phys Med Biol* 52, 5771-5783 (2007).
51. Riabkov, D., Xue, X., Tubbs, D. Accelerated cone-beam backprojection using GPU-CPU hardware. in: Proc. Ninth Int'l Meeting Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine. 68-71 (2007).
52. Noël, P. B., Walczak, A. M., Hoffmann, K. R. Clinical evaluation of GPU-based cone beam computed tomography. in: Proc. High-Performance Medical Image Computing and Computer Aided Intervention (HP-MICCAI) (2008).
53. Okitsu, Y., Ino, F., Hagihara, K. Fast cone beam reconstruction using the CUDA-enabled GPU. in: Proc. 15th Int'l Conf. High Performance Computing (HiPC). 108-119 (2008).
54. Xu, F., Mueller, K. Accelerating popular tomographic reconstruction algorithms on commodity PC graphics hardware. *IEEE Transactions on Nuclear Science*. 52, 654-663 (2008).
55. Okitsu, Y., Ino, F., Hagihara, K. High-performance cone beam reconstruction using CUDA compatible GPUs. *Parallel Computing* 36, 129-141 (2010).
56. Samant, S. S., Xia, J. Y., Muyan-Ozcelik, P. High performance computing for deformable image registration: Towards a new paradigm in adaptive radiotherapy. *Med Phys* 35, 3546-3553 (2008).
57. Gu, X., Pan, H., Liang, Y. Implementation and evaluation of various demons deformable image registration algorithms on a GPU. *Phys Med Biol* 55, 207-219 (2010).
58. Hissoiny, S., Ozell, B., Despres, P. Fast convolution-superposition dose calculation on graphics hardware. *Med Phys* 36, 1998-2005 (2009).
59. de Greef, M., Crezee, J., Van Eijk, J. C. Accelerated ray tracing for radiotherapy dose calculations on a GPU. *Med Phys* 36, 4095-4102 (2009).
60. Gu, X., Choi, D., Men, C. GPU-based ultra-fast dose calculation using a finite size pencil beam model. *Phys Med Biol* 54, 6287-6297 (2009).
61. Jia, X., Gu, X., Sempau, J. Development of a GPU-based Monte Carlo dose calculation code for coupled electron-photon transport. *Phys Med Biol* 55, 3077-3086 (2010).
62. Men, C., Gu, X., Choi, D. GPU-based ultra fast IMRT plan optimization. *Phys Med Biol* 54, 6565-6573 (2009).
63. Yan, H., Ren, L., Godfrey, D. J. Accelerating reconstruction of reference digital tomosynthesis using graphics hardware. *Med Phys* 34, 3768-3776 (2007).
64. Yan, H., Godfrey, D. J., Yin, F. F. Fast reconstruction of digital tomosynthesis using on-board images. *Med Phys* 35, 2162-2169 (2008).
65. NVIDIA, CUDA Programming Guide v.2.3 (2009), <http://www.nvidia.com>.
66. Feldkamp, L. A., Davis, L. C., Kress, J.W. Practical cone beam algorithm. *J Opt Soc Am* 1, 612-619 (1984).
67. Cho, P. S., Johnson, R. H., Griffin, T. W. Cone-beam CT for radiotherapy applications. *Phys Med Biol* 40, 1863-1883 (1995).
68. NVIDIA, GeForce GTX 295 dual chip graphics card (2009), <http://www.nvidia.com>.
69. Siddon, R. L. Fast calculation of the exact radiological path for a three-dimensional CT array. *Med Phys* 12, 252-255 (1985).
70. Otto, K. Volumetric modulated arc therapy. *Med Phys* 35, 310-317 (2008).
71. Palma, D., Vollans, E., James, K. Volumetric modulated arc therapy for delivery of prostate radiotherapy: comparison with intensity-modulated radiotherapy and three-dimensional conformal radiotherapy. *Int J Radiat Oncol Biol Phys* 72, 996-1001 (2008).

Received: January 4, 2011; Revised: April 15, 2011;

Accepted: May 5, 2011