Efficiently GPU-Accelerating Long Kernel Convolutions in 3-D DIRECT TOF PET Reconstruction via a Kernel Decomposition Scheme

Sungsoo Ha, Zhiyuan Zhang, Mikayel Ispiryan, Samuel Matej, *Senior Member, IEEE,*

and Klaus Mueller, *Senior Member, IEEE*

Abstract– **The DIRECT approach for 3-D Time-of-Flight (TOF) PET reconstruction performs all iterative predictor-corrector operations directly in image space. A computational bottleneck here is the convolution with the long TOF (resolution) kernels. Accelerating this convolution operation using GPUs is very important especially for spatially variant resolution kernels, which cannot be efficiently implemented in the Fourier domain. The main challenge here is the memory cache performance at non-axis aligned directions. We devised a scheme that first re-samples the image into an axis-aligned orientation offering good memory coherence for the convolution operations. In order to maintain good accuracy, we carefully design the resampling and new convolution kernels to combine into the original TOF kernel. This paper demonstrates the validity, accuracy, and high speed-performance of our scheme for a comprehensive set of orientation angles. Future work will apply these cascaded kernels within a GPU-accelerated version of DIRECT.**

I. INTRODUCTION

THE DIRECT (Direct Image Reconstruction for TOF) approach [1] has been recently proposed as a more efficient alternative to the binned and list-mode TOF PET reconstruction approaches [2]. In the binned approaches, the events are binned by their LOR (Line of Response) and arrival time to form a set of *histo-projections*, one for each angular view. On the other hand, in the DIRECT approach the events are first sorted into a (sub)set of angular views and then deposited for each view into a dedicated *histo-image*, each having the same lattice configuration and the same resolution as the reconstructed image. In DIRECT each corrective update involves simple 3D convolutions using the system response (SR) kernel, which can be performed efficiently in Fourier space when the SR kernel is spatially invariant. However, in practical applications the SR kernel is not spatially invariant – its width increases about 40% towards the edge of the scanner. This prohibits the use of efficient Fourier-space methods to accelerate the convolution operations. Since the SR kernel is typically quite large (45×5×5 voxels) a spatial convolution within a 144×144×62 matrix and 120 views can be

prohibitively expensive for clinical application. We seek to overcome this challenge by GPU-acceleration [3][4], using their massively parallel computations to meet this challenge.

In general, mapping a CPU-based algorithm to the GPU and achieving 1-2 orders of speedup is typically not straightforward. One needs to carefully consider both GPU architecture and programming model and break down the algorithm into appropriate steps, reordering and decomposing them if needed. An especially critical component in GPUs is the memory, which is organized into a hierarchy and requires coherent access patterns to minimize latencies. In DIRECT, convolving an image with a long kernel in non-grid aligned image directions causes incoherent memory access patterns. We therefore propose a two-stage kernel decomposition scheme that shifts the irregular memory access patterns to stage in which a small kernel resamples the image into a standardized memory-friendly orientation and then applies the long convolution kernel there.

We note that shifting more involved computations into a standardized compute-friendly configuration is a practice often used in high-performance computing. For example, in [5] the authors performed cone-beam spiral backprojection at a standard orientation (the same for each projection angle) which they followed by a resampling (and accumulation) of the slice into the angularly-correct orientation.

Next we outline our approach, discuss its theoretical aspects, and present results for time and quality performance.

II. OVERALL APPROACH

Figure 1 illustrates the fundamental idea of our approach for a convolution (forward-projection) angle of 30°. It replaces an (elliptical Gaussian) convolution at an arbitrary angle α by a rotation of that image by -α followed by a convolution along the y-direction (note that the data is always stored in slices along the direction indicated by the arrow on the right – the dark encasing region is only shown for illustration). This rotation is performed by resampling the source image using an isotropic Gaussian interpolation filter. The overall savings come from the fact that poor memory access patterns are now restricted to the resampling phase which has only a small filter footprint. Conversely, the subsequent elliptical Gaussian convolution now has a very regular access pattern along the y-direction, utilizing fast coalesced memory reads. Once the image has been convolved we rotate it back (the last rotation can be avoided in certain reconstruction schemes - dotted line). The back-projection operation (transpose of the forward-
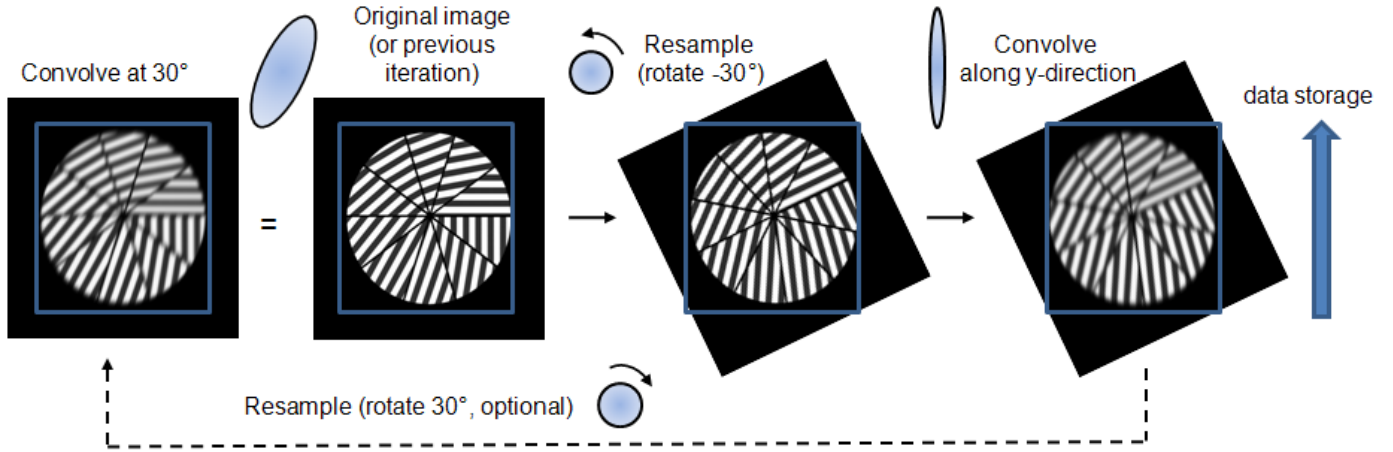
Figure 1: Two-stage convolution (forward-projection) pipeline.

projection) is a sequence of the same operations performed in the opposite order.

III. THEORY

One can decompose a Gaussian kernel C into two Gaussian kernels, S and C' where $S \otimes C' = C$, or $C = S \cdot C'$ in the frequency domain (note, while Figure 1 shows C oriented at $30°$ this derivation can assume C to be in standard position). Since the following Fourier pair exists for the Gaussian:

$$e^{-\frac{x^2}{2\sigma^2}} \Leftrightarrow \frac{1}{\sqrt{1/\sigma^2}} e^{-\frac{\omega^2 \sigma^2}{2}} \qquad (1)$$

we can obtain the following condition: $\sigma_C^2 = \sigma_S^2 + \sigma_{C'}^2$. We seek to minimize the extent of S (that is, $\sigma_{C'}$) to limit uncoalesced memory access patterns in the resampling phase. On the other hand, we may also choose $\sigma_S = \sigma_c$ within the 2D cross-section of C and then use just a 1D convolution along the y-direction for C'. In any event, we must set $\sigma_{C'}^2 = \sigma_C^2 - \sigma_S^2$ in the long axis of the convolution filter.

IV. RESULTS AND CONCLUSIONS

We tested our algorithm on a NVIDIA Tesla C870 (which has the G80 chip also used in the 8800 GTX consumer-grade GPU). Convolving 120 3D-images of size $144 \times 144 \times 62$ with the 2-stage equivalent of a $45 \times 5 \times 5$ filter took about 4s. The GPU-accelerated original 1-stage method required 40s, while the FFT-based approach on a 2 GHz Mac G5 single processor [1] consumed about 1 minute. It is noteworthy here that the latter assumed a spatially invariant kernel, while our GPU implementation widened the kernel width towards the edges of the detector and hence was spatially variant which prohibits the log(n) acceleration facilitated by FFTs.

Figure 2 estimates the error, comparing the original 1-stage convolution algorithm with our 2-stage scheme. Using the phantom shown in Figure 1, we observed an RMS error of about 0.5% of the maximum image value. Note that this error also includes the error incurred by the second interpolation needed to align the 1- and 2-stage images for comparison.

We are currently working on a faster Gaussian filtering scheme that better exploits the GPU fixed function pipeline and also on incorporating the 2-stage filter scheme into the DIRECT reconstruction framework.

REFERENCES

[1] S. Matej, S. Surti, S. Jayanthi. M. Daube-Witherspoon. R. Lewitt. J. Karp, "Efficient 3-D TOF PET reconstruction using view-grouped histo-images: DIRECT-direct image reconstruction for TOF," *IEEE Trans Med Imaging*, 28(5):739-51, 2009.

[2] L. Popescu, S. Matej, R. Lewitt, " Iterative image reconstruction using geometrically ordered subsets with list-mode data," *IEEE Medical Imaging Conference*, M9-211, pp. 3536–3540, 2004.

[3] F. Xu, K. Mueller, "Accelerating popular tomographic reconstruction algorithms on commodity PC graphics hardware," *IEEE Trans. on Nuclear Science*, 52(3):654-663, 2005.

[4] F. Xu, K. Mueller, "Real-Time 3D Computed Tomographic Reconstruction Using Commodity Graphics Hardware," *Physics in Medicine and Biology*, 52:3405–3419, 2007.

[5] S. Steckmann, M. Knaup, M. Kachelrieß, "High performance cone-beam spiral backprojection with voxel-specific weighting," *Physics in Medicine and Biology*, 54:3691–3708, 2009.

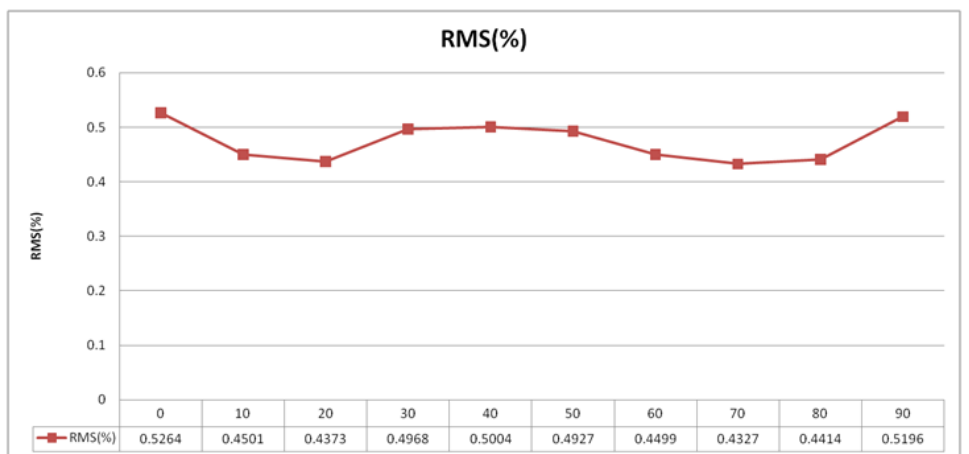|	0	10	20	30	40	50	60	70	80	90
RMS(%)	0.5264	0.4501	0.4373	0.4968	0.5004	0.4927	0.4499	0.4327	0.4414	0.5196

Figure 2: RMS error in % of the maximum image value (this error includes the error incurred by the second interpolation needed to align the 1- and 2-stage images for comparison).