Iterative Helical Cone-Beam CT Reconstruction using Fast Hierarchical Backprojection/Reprojection

Jeffrey Brokish*, Hongbin Guo*, Paul Sack*, Daniel B. Keesing* and Yoram Bresler†

*InstaRecon, Inc., Champaign, IL USA 61820
†Dept. of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign

Abstract—This is the first report on a new fast iterative CT reconstruction algorithm for helical cone beam (HCB) scans, accelerated by InstaRecon’s fast \(O(N^3\log N)\) hierarchical cone beam backprojection and reprojection algorithms. We report on the results of image quality evaluations for dose reduction, demonstrating that the iterative algorithm introduced here can provide image quality indistinguishable from an iterative algorithm using conventional BP/RP operators. We further show the iterative algorithm provides image quality comparable to a high-dose FBP reconstruction, but with only 25% of the dose. Finally, run-time statistics are reported for a version of the algorithm using GPU implementations of the hierarchical operators. The combined algorithmic and hardware acceleration provides a reconstruction engine with sufficient throughput to be viable as the default modality for typical clinical workflow, but with very modest hardware requirements.

I. INTRODUCTION

Incorporating statistical and physical models of the data acquisition, iterative CT reconstruction algorithms reduce noise and artifacts in images, which is a key enabler for reduced dose imaging. However, iterative algorithms are computationally expensive, as each iteration requires a forward and backward projection operation. These backprojection/reprojection (BP/RP) operators, which scale as \(O(N^4)\) for 3D cone-beam reconstruction of an \(N\times N\times N\) voxel volume, are the computational bottleneck of iterative reconstruction, just as the BP operation is the bottleneck of standard filtered backprojection (FBP). Hierarchical algorithms reduce the complexity of these operators to \(O(N^3\log N)\), offering the potential to greatly accelerate the reconstruction throughput. In this paper, we report on the performance of an iterative algorithm for the 3D helical cone-beam case using fast hierarchical algorithms [1].

II. ITERATIVE ALGORITHM FRAMEWORK

A. Cost Function

The iterative reconstruction framework plays a key role in determining image quality, dose reduction, and computational requirements. There are various methods for implementing an iterative reconstruction scheme: the choice of a cost function, regularization, and optimization strategy. For this work we chose a penalized weighted least squares cost function [2] of the form

\[
J(f) = \|y - Rf\|_W^2 + \beta C(f)
\]

(1)

In this cost function, \(f\) is the estimate of the image, \(y\) is the measured projection data, \(R\) computes the forward projection of its argument, and \(C(f)\), with relative weight \(\beta\), is a non-quadratic regularization term that favors smooth regions in \(f\) while preserving sharp transitions. The \(W\) indicates a weighted norm of the difference between the measured values \(y\) and estimated projections \(Rf\). The form of this cost function lends itself to conjugate gradient (CG)-based minimization strategies, which are the most amenable to acceleration by InstaRecon fast hierarchical operators.

B. System Model

There are many options for the implementation of the conventional reprojector \(R\) that depend on the representation of the image \(f\). The choice of the image representation (basis function) has a significant influence on the performance of the iterative algorithm, both in terms of runtime and resulting image quality. Based on the ideas presented in [4] we implemented a higher order separable footprint for each voxel. For computational efficiency, the footprint integration is calculated not on the detector panel, but rather detector boundaries are brought to image ‘slabs’ and integration is performed in the image domain. The choice between \(x-z\) or \(y-z\) image slabs is based on which direction is most orthogonal to the line between the source and center of rotation. Such a voxel representation is useful in reducing aliasing artifacts in the reprojection step, which may not be suppressed by an iterative algorithm. Additionally, reprojectors with a large amount of aliased energy are more challenging for the hierarchical algorithms to handle accurately.

III. FAST HIERARCHICAL OPERATORS

The fast 3D conebeam BP/RP algorithms used in this work reduce the computational complexity from \(O(N^4)\) to \(O(N^3\log N)\) by a hierarchical decomposition of the image volume. The hierarchical reprojection algorithm is briefly described here. It is based on two main concepts. The first concept is divide and conquer, in which the volume is successively divided into smaller non-overlapping volumes, and the reprojection operation is applied to these subvolumes. An example of this decomposition used for 3D volumes is shown in Figure 1. By itself this does not provide any reduction in computational cost.

The second concept invokes sampling conditions, where the number of views required to accurately sample the projections of a bandlimited subvolume at the center of the source of rotation is proportional to the size of the subvolume. For the
projections of a half-size subvolume, the projection data set can initially be reprojected at a sparser set of P/2 projections and then angularly interpolated by a factor of 2, yielding projection data with comparable quality.

These two concepts are combined to form the hierarchical algorithm. Subvolumes are approximately ‘centered’ through shifts in the projection data to position the center of the subvolume’s shadow at the detector isocenter. This sufficiently reduces the angular sampling requirements of the subregion that it can be reprojected at a coarser set of views, then interpolated to a higher angular sampling rate. Projection data for groups of neighboring subvolumes are then added together to form data for larger subvolumes, which is again interpolated to a higher number of views. This process continues until the entire volume has been reconstituted.

An example demonstrating one level of this process for 2D reprojection is shown in Figure 2. The image is broken down into small subregions, which are reprojected (R) at a sparse set of view angles. These projection sets are shifted and interpolated (S/I) to a larger projection count and merged together to form projection sets of larger subregions.

The fast hierarchical backprojector is constructed as the adjoint of this operation, with the appropriate flow reversal of each step in the algorithm. It is analogous in form to the fast hierarchical backprojection algorithms [3] that have been developed for filtered backprojection reconstruction (FBP). An FBP reconstruction is typically used to form the initial guess in the iterative algorithm, so this step can also be accelerated by hierarchical methods.

IV. GPU IMPLEMENTATION

The primary processing element of an Nvidia GPU is organized as an array of simple processing units (SP) grouped together into multiprocessor units (MP). A GPU consists of several such multiprocessors operating in parallel. Typical GPUs have a combined SP count in the hundreds. Each multiprocessor contains a small amount of local shared memory and a set of registers that are shared among all of the simple processing units in that multiprocessor. Additionally, each multiprocessor has a dedicated texture unit, which can perform array lookup and interpolation operations.

While GPUs offer tremendous computational resources, care must still be taken in programming the computation kernels. Extracting high performance requires taking into account architectural features of the GPU. A sufficient number of parallel threads of execution must be available to keep all the SP units busy and tolerant of any delays in computation or memory accesses. The resources (registers and shared memory) required by each thread must be minimized to enable a large number of threads to run on an MP.

Conventional filtered backprojection (FBP) has a very obvious parallel structure, where each voxel can be reconstructed independently. The GPU is particularly amenable to an FBP implementation, as a key component of the algorithm is linear interpolation of projection data. The texture unit serves as a data cache that provides linear interpolation in hardware, offloading a good portion of the work required by the FBP kernel. Implementation on the GPU of a reprojector, on the other hand, is substantially more complicated and does not map as elegantly to the GPU, especially for system models that incorporate some form of basis representation along with detector integration.

Our approach to implementation of hierarchical operators on the GPU platform involved looking at the decomposition in ‘stages’ (i.e. kernel launches). The first stage is comprised of taking the reconstruction volume and dividing it into the set of smallest subvolumes (after all recursive decompositions have been applied), performing the reprojection operation on each subregion using the conventional reprojector at a projection set with sparse angular sampling. Each successive stage then involves interpolation of each dataset to a higher angular sampling rate, followed by a shifting operation (the I/S operation from Figure 2). Finally, the projection data for groups of four subvolumes are added together to form the projection dataset of the ‘parent’ subvolume.

It turns out that this process does not map to the GPU architecture as elegantly (or simply) as the traditional FBP. Different subvolumes will have different sized decimated projection datasets (dependent on the rays that intersect the subvolume) leading to challenges in packing data efficiently and effectively assigning worker threads to the low-rate projection detectors in a regular manner. We developed a scheduling mechanism to prevent the creation of idle threads in a thread block due to this irregularity. Other GPU hardware features are also leveraged to improve efficiency, including storing intermediate calculations in Shared Memory, and using the texture units as a memory caching mechanism for accessing the decimated projection data.

One other challenging issue was the data set sizes encoun-
tered in helical geometries, where a scan may consist of many revolutions. It becomes very easy to exhaust the available memory on the GPU cards we were using (around 1GB of on-board memory). We devised a set of strategies, which partition the problem into a sequence of manageable subproblems that can fit on the GPU. We leverage the asynchronous operational mode of the GPU to overlap transfers and computation as much as possible.

V. RESULTS
   A. Equivalence of Hierarchical and Conventional Operators

   Figure 3 demonstrates reconstructions for simulated data for a 64 row detector panel with helical pitch 1. The iterative algorithm is able to reduce noise and suppress artifacts in the image (i.e. streaks emanating from the air cells in the “temporal bone”). The low contrast features are very apparent while the sharpness of the image (air cells, bone) is maintained. Most importantly, the good agreement between hierarchical and conventional operator based iterative algorithms is confirmed.

   B. Low Dose Imaging

   Figure 4 shows a comparison of reconstruction methods at different dose levels. Noise was added to the clinical dataset to simulate a reduction of dose to 25% of the full dose level. The following reconstructions were performed: FBP using high dose data, FBP using low dose data, and Iterative using low dose data. The iterative algorithm is able to bring back the image quality to a level comparable to that of the high-dose FBP reconstruction.

   C. Iterative Reconstruction Throughput

   The CPU platform used here is a quad-core Core i7 processor running at 2.66 GHz with 24 GB of RAM. The GPU used is the nVidia GTX 470, a modest gaming-class graphics card, with 448 cores running at 1215 MHz and 1280 MB of on-board memory. The hierarchical BP/RP operators are run on the GPU, the remaining aspects of the iterative framework (regularization term, etc) are run on the CPU. The CPU code is written in C++ and uses thread parallelism to leverage all available processing cores and when possible extracts data level parallelism using the SSE vector instruction set.

   Table I lists results from two clinical datasets with different helical pitches and region of interest (ROI) sizes. In both of these cases we achieve a reconstruction rate of 2 seconds per slice or better, which is a rate viable for a typical clinical workflow (6 patients per hour, 300 slices per scan). This throughput is also competitive with other commercially available iterative reconstruction packages employing the model-based iterative reconstruction (MBIR) techniques used here, but does so on vastly lower powered hardware.

   Further gains in reconstruction rates could be had by porting the remainder of the iterative framework to the GPU. While such aspects are fairly minor overhead for an iterative algorithm using conventional operators, after the BP/RP operators have been accelerated by both the hierarchical algorithm and the GPU, the extra computation is a non-trivial component of the total runtime. With a modest 4x acceleration of these components by running them on the GPU, we estimate potential improvement of 15-30% in system throughput. The newer 500 series of graphics cards can also provide a significant performance boost over the GTX 470 used in this evaluation.

VI. CONCLUSION

   In this paper, we reported on results in evaluating an iterative reconstruction framework for dose reduction, accelerated by the combination of the algorithmic techniques of InstaRecon hierarchical backprojection/reprojection operators along with hardware acceleration on the GPU platform. This is the first time that an algorithmically-accelerated iterative reconstruction engine for 3D HCB geometries has been demonstrated.

   The iterative reconstruction engine delivers significant decrease in dose (a factor of 4 reduction) at a throughput that enables iterative reconstruction as the default reconstruction mode without impacting workflow. Further, the modest hardware requirements enable the possibility of including low-dose scanning in the value segment of the CT market. In addition, the delivered acceleration enables to incorporate, with additional development, additional enhanced physical modeling at reduced cost, to facilitate even greater dose reductions in the future.

ACKNOWLEDGMENT

   The project described was supported by a Phase II SBIR Award Number R44EB005576 from the National Institute of Biomedical Imaging and Bioengineering to InstaRecon, Inc. The content of this publication is solely the responsibility of the authors and does not necessarily represent the official views of the National Institute of Biomedical Imaging and Bioengineering or the National Institutes of Health.

REFERENCES

Fig. 3: Slices through reconstruction of simulated phantom dataset comparing (a) FBP with (b) Iterative using conventional operators and (c) Iterative using hierarchical operators

Fig. 4: Comparison of reconstructions of clinical dataset from FBP with high dose data (a), FBP with low dose data (b), and iterative with low dose data (c)