Fast Temporally Constrained Reconstruction on a GPU Cluster

Jordan P Hulet[1,2], Ganesh Adluru[3], Julio C Facelli[1], Edward DiBella[3], and Dennis L Parker[4,5]
[1]*Biomedical Informatics, University of Utah, Salt Lake City, Utah, United States,* [2]*Utah Center for Advanced Imaging Research, Salt Lake City, Utah, United States,* [3]*University of Utah, Salt Lake City, Utah,* [4]*University of Utah, Salt Lake City, utah,* [5]*Utah Center for Advanced Imaging Research, University of Utah, Salt Lake City, Utah, United States*

Introduction

Advances in under-sampled MR image reconstruction have allowed for significantly reduced acquisition times, and advanced techniques such as constrained reconstruction or compressed sensing have been shown to allow for very high reduction factors. One drawback of these methods is the lengthy image reconstruction time due to their high computational demand; computing a sufficient number of iterations to converge on an optimal image estimate can take minutes or hours, depending on the dimensionality of the data. As a result, reconstructed images cannot be obtained during an exam, making it impractical to assess image quality and the need for any repeat scans, which severely limits their clinical utility. Graphics processing units (GPUs) have been used to accelerate computationally demanding MRI image reconstructions. However, using a single GPU does not work well with very large data sets due to limited on-card memory and thread count restrictions. In this work we attempted to speed up temporally constrained reconstruction (TCR) of a large under-sampled cardiac perfusion data set by using 12 GPUs distributed across 6 compute nodes. We hypothesized that this configuration could perform TCR image reconstruction of a large data set in less than 1 minute, which would be a reasonable reconstruction time during an exam.

Methods

With informed consent and IRB approval, an under-sampled, radial cardiac perfusion dataset was obtained using an ungated radial saturation recovery turboFLASH sequence with the following parameters: TR=2.2 ms, TE=1.3 ms, pixel size=1.8 mm2, radial views/repetition= 24, repetitions=250, slices=4. Prior to image reconstruction, principal component analysis was used to reduce the number of individual receiver channels from 32 down to 5 virtual coils corresponding to the top 5 principal components. [ref MRM 2007] The following cost functional was used in the TCR algorithm with the following definitions, W: under-sampling mask, $\mathcal{F}$: 2D FFT, S: coil sensitivities, $\tilde{m}$: image estimate, d: measurement data, λ: temporal constraint weight, and ∇_t: temporal gradient.

$$C(\tilde{m}) = \|W\mathcal{F}S\tilde{m} - d\|_2^2 + \lambda\|\nabla_t\tilde{m}\|_1^1$$

100 iterations were used because it was determined that this was a sufficient number for convergence of the image estimate. The TCR GPU code was implemented using the NVIDIA CUDA architecture as well as on a threaded multi-CPU architecture using posix threads. The data was divided among different compute nodes by splitting it into several shorter temporally overlapping data sets (an overlap of 3 frames on each side). This overlap was necessary to deal with boundaries created by the data sub-division. Each sub-set was reconstructed in parallel, and later put together. Message Passing Interface (MPI) was used to distribute work between nodes, each having two NVIDIA Tesla M2090 GPUs. The TCR reconstruction time was measured for 3 different scenarios: 1) using 16 CPU threads on a single node, 2) using 1 NVIDIA Tesla M2090 GPU on a single node, and 3) using 12 Tesla M2090 GPUs distributed across 6 different nodes.

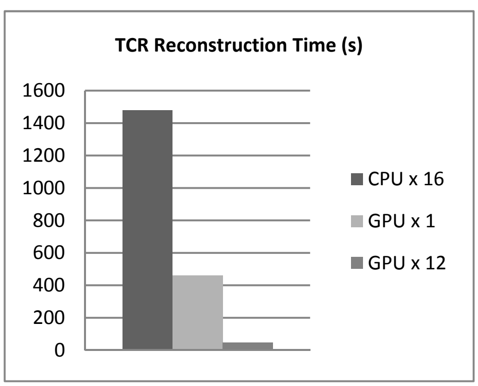

Fig 1: Total TCR reconstruction times for the data set acquired. Times shown are for 16 cores on a multi-core CPU node, 1 GPU, and 12 GPUs distributed across multiple nodes. 12 GPUs showed more than a 30x speedup over 16 CPU cores.

Results

The reconstruction times are compared in Fig 1. Using 16 CPU cores on a single node the reconstruction time was 1480 seconds, when a single GPU was used the time was reduced to 456 seconds, and when all 12 GPUs were used the total reconstruction time was only 48 seconds. It should be noted that reconstruction times included preprocessing operations such as data re-gridding and coil sensitivity estimation, which have not yet been parallelized.

Discussion

This work has shown that multiple GPUs (12 in this case) can be used simultaneously to drastically reduce the reconstruction time required for TCR. With the data set collected reconstruction time was reduced from 1480 seconds with a multi-core CPU system to only 48 seconds using multiple GPUs. The reduction time achieved makes image reconstruction of similarly sized, large data sets reasonable during the exam. In conclusion, the use of multi-GPU systems for performing complex MRI reconstructions may allow for exam-time image reconstruction, allowing the quality of these fast acquisition techniques to be assessed while the patient is still available for any repeat scans.

Acknowledgements

This research is supported in part by the Ben B. and Iris M. Margolis Foundation, NIH grant HL 57990, A Clinical Merit Review Grant from the Veterans Administration Health Care System, and NLM training grant LM007124.

References

[3]Huang, Magn Resn Imaging, 2007.