Fast Reconstruction Technique for Medical Images Using Graphics Processing Unit

Mohammad Nazmul Haque[1], Mohammad Shorif Uddin[2], M. Abdullah-Al-Wadud[3], and Yoojin Chung[4]

[1] Department of CSE, Daffodil International University, Dhaka, Bangladesh
nazmul@daffodilvarsity.edu.bd

[2] Department of CSE, Jahangirnagar University, Savar, Dhaka, Bangladesh
shorifuddin@gmail.com

[3] Department of IME, Hankuk University of Foreign Studies, Yongin, South Korea
wadud@hufs.ac.kr

[4] Department of CS, Hankuk University of Foreign Studies, Yongin, South Korea
chungyj@hufs.ac.kr

Abstract. In many medical imaging modalities, the Fast Fourier Transform (FFT) is being used for the reconstruction of images from acquired raw data. The objective of the paper is to develop FFT and Inverse FT algorithms to run under GPU for performing in much faster way. The GPU based FFT implementation provides much faster reconstruction of Medical images than CPU based implementation. The GPU based algorithm is developed in MATLAB environment. GPUMat is used to running CUFFT library code in MATLAB. This work exercises the acceleration of MRI reconstruction algorithm on NVIDIA's GPU and Intel's Core2 Duo based CPU. The reconstruction technique shows that GPU based MRI reconstruction achieved significant speedup compared to the CPUs for medical applications at a cheaper cost.

Keywords: MRI reconstruction, image reconstruction, medical imaging, GPU based image processing, image processing using CUDA.

1 Introduction

The Fourier Transform (FT) is a mathematical operation used widely in many fields. In medical imaging it is used for many applications such as image filtering, image reconstruction and image analysis. When Magnetic Resonance Imaging (MRI) machines take an image of the human body, the output is in the form of raw data. The FT is used to reconstruct the image from this raw data. In medical imaging devices specialized hardware or CPUs are used to reconstruct images from acquired raw data. When the raw data size is relatively small, it takes moderate time to reconstruct an image. But, as the raw data size continues increasing, the time for processing the reconstruction increases as well. Image reconstruction has reached a bottleneck where further speed improvement from the algorithmic perspective is difficult. But requirement for faster FT has not comes in an end. This paper's aim is to increase the performance of FT based medical image reconstruction.

T.-h. Kim et al. (Eds.): SIP 2011, CCIS 260, pp. 300–309, 2011.

2 Literature Review

General-purpose computing on graphics processing units (GPU) supports a broad range of scientific and engineering applications, including physical simulation, signal and image processing, database management, and data mining [1]. There are several excellent reviews of image reconstruction by many other authors. These include: Calvetti, Reichel & Zhang (1999) on iterative methods; Molina et al. (2001) and Starck, Pantin & Murtagh (2002) on image reconstruction in astronomy.

Medical imaging was one of the first GPU computing applications, with computed tomography (CT) reconstruction achieving a speedup of two orders of magnitude on the SGI Reality Engine in 1994 [2]. A wide variety of CT reconstruction algorithms have since been accelerated on graphics processors [2],[3],[4],[5] and the Cell Broadband Engine [1]. In [3] the GPU is used to accelerate Simultaneous Algebraic Reconstruction Technique (SART), an algorithm that increases the quality of image reconstruction relative to the conventional filtered back-projection algorithm under certain conditions. MRI reconstruction on the GPU has not been studied extensively. Research in this area has focused on accelerating the FFT.

3 Magnetic Resonance Imaging Basics

Magnetic resonance imaging (MRI) is an imaging technique used primarily in medical settings to produce high quality images of the inside of the human body. MRI is based on the principles of tomographic imagine technique [7],[8],[9]. It is used primarily in medical fields to produce images of the internal structure of human body. It does not involve the use of ionizing radiation hence, free from associated harmful effects known with other imaging techniques [10].

MRI imaging equation can be express as a two dimensional entity given as

$$S\left(k_x, k_y\right) = f\left[I\left(x, y\right)\right] \tag{1}$$

where f represent spatial information encoding scheme [7]. If f is invertible, a data consistent I can be obtained from the inverse transformation such that

$$I\left(x, y\right) = f^{-1} S\left(k_x, k_y\right) \tag{2}$$

The desired image intensity function $I(x, y)$ is a function of: Relaxation times, $T1$, $T2$ and $T2^*$ spin density ρ, Diffusion coefficients$D$ and so on [11]. Using function notation, this can be written as

$$I = f\left[T1, T2, T2, \rho, D\right] \tag{3}$$

Generally, $T1$ and $T2$ are two independent processes and happen simultaneously. $T1$ is called spin lattice relaxation, because the energy from this process is released to the surrounding tissue (lattice) [7],[8],[11]. $T1$ happens along the z-component axis and its value is always greater than the spin-spin relaxation $T2$. The relationship between protons and their immediate surroundings (molecules) is described by the spin-spin relaxation $T2$ and it happens along x-y plane.

3.1 Fourier Transform Based MR-Imaging Principle

FT base tomographic imaging is the most commonly used MRI method utilized today. It uses a type of magnetic field gradient, called a phase encoding gradient, in addition to the slice selection and frequency encoding gradients. The MRI recording process is repeated 128 to 256 times, varying the magnitude of the phase encoding gradient each time, in order to obtain sufficient data, free induction decays or signals, for creating an image [12].

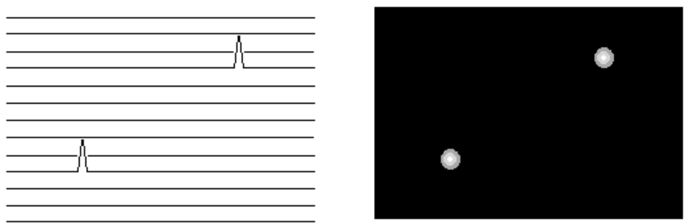


Fig. 1. Tomographic image created from Fourier transformed data

Before actually creating the image, however, the signals must be Fourier transformed. This is done first in the direction in which the spins are located to extract the frequency domain information. Then it is done in the phase encoding direction to obtain information about the spin locations in that direction. The FT data finally becomes an image when the intensities of the data peaks are converted into intensities of pixels. This creates a tomographic image, shown in Fig.1.

3.2 MRI Data Acquisition

MRI data acquisition involves three major steps shown in Fig. 2, namely:

- Gz, Slice selection by the use of Gz gradient: This select axial slice in the object to be imaged.
- Gy, Phase encoding using the Gy gradient: This creates rows with different phases.
- Gx, Frequency encoding using the Gx gradient: This will create columns with different frequencies.

The data that are captured during MR imaging are called k-space data or, simply, raw data. Typically, the data are collected by using quadrature detection, which provides both real and imaginary k-space data. K-Space data include useful information but can be interpreted only after they are translated into images with the Fourier transform method which is illustrated in Fig 3. The K-space data contains all the necessary information required to reconstruct an image. Also, it gives a comprehensive way for classification and understanding of the imaging properties and method of reconstruction [7],[13], [14].

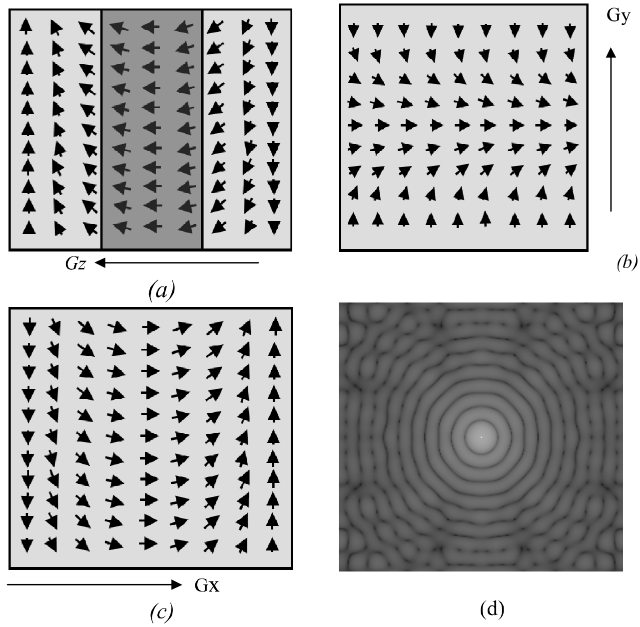


Fig. 2. K-Space Data formation: a)Gz Gradient data Filling, b) Gy Gradient Data Filling, c) Gx Gradient Data Filling and d) K-Space Data

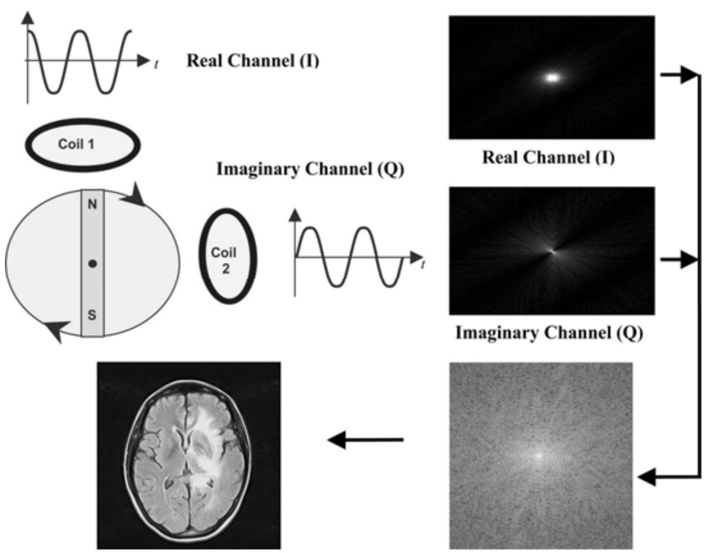


Fig. 3. Schematic of the collection and combination of imaginary MR signal data to produce a complex map of k-space

4 MR Image Reconstruction

There exist various methods of converting the acquired K-space data to real images in spatial domain. These include: the use of DFT, Radon Transform, Parametric technique, artificial neural network based reconstruction technique and so on. DFT technique involves the application of Fourier series on the linearly or radially sampled k-space data. This work deals with only DTF technique using FFT.

4.1 Application of Inverse FFT to K-Space Data

MRI reconstruction using FFT/IFFT is done in two steps; firstly the 2D- IFFT of the data is computed then data are shifted to center for display the image. In practical MRI reconstruction, there are some other pre-processing activities that must be accomplished before the application of IFFT [15],[16],[17]. The reconstruction algorithm is depicted here.

Algorithm : MRI Reconstruction

Input: MRI Raw Spectral Domain Data
Output: Reconstructed Spatial Domain Image

```
Step 1: Read data header information:
        Load RAW data contain information about the MRI data file.
        It is a text file containing Information about Offset,
        DATA size, Kx Co-ordinate, Ky Co-ordinate etc.
Step 2: Read in the K-space DATA.
Step 3: IFFT in K(x,y) Direction
Step 4: FFT shift
Step 5: Image Display
```

K-space is the rawest form of data obtained at MR imaging. An acquisition with a 256 x256 matrix contains 256 lines of data, and each of those lines contains 256 data points. The distance between neighboring points in k-space determines the field of view of the object imaged, and the extent of k-space determines the resolution of the image [18],[19].

5 Experimental Setup

The experiment is divided into two sections. First is simulation of performance for FFT-IFT on different sized images. After that the experiment is done on MR data. All experiments are done using both CPU and GPU. The Hardware configurations for them are listed in Table 1 and Table 2 . The experiment requires some software and tools for programming and documenting purpose. Following Table 3 lists up all used software and tools.

Table 1. CUDA Device Configuration at Experiment

Features	Specification
Name:	GeForce G 103M
CUDA Driver Version:	3.2
Total Global memory:	521601024 bytes
Multiprocessors x Cores:	1 x 8
Clock rate:	1.60 GHz

Table 2. Host Machine Configuration at Experiment

Features	Specification
System Model:	Compaq Presario CQ40 Notebook PC
Operating System:	Windows 7 Ultimate 32-bit
Processor:	Intel(R) Core(TM)2 Duo CPU
Clock Speed:	2.00GHz
Memory:	2048MB RAM

Table 3. Required Software and Tools

Software	Version	Purpose
NVIDIA GPU Computing SDK	3.2	SDK for NVIDIA's GPU
MATLAB R2010a	7.10	Simulation and Programming
CUFFT	2.3	CUDA capable FFT library
GPUmat	0.27	Wrapper for MATLAB to run CUDA Program

For comparing the runntime of FFT based image Reconstruction using GPU we have used three images shown in Fig 4(a). Images are chosen different resolution for figure out the performance on CPU and GPU based implementation. MRI raw data in Fig 4(b) has been collected from Duke Virtual Imaging Laboratory[6].

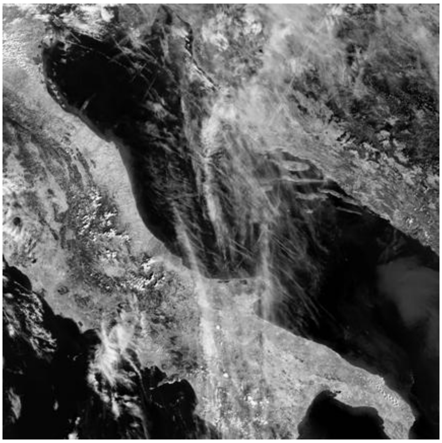
6 Simulation Results

GPU and CPU performance results are obtained by computing FFT of experimental images in MATLAB and measuring the execution time using commands tic and toc. All CPU iterations are measured and averaged over 100 iterations.

Spatial resolution of lena image is 512x512. Both CPU based and GPU based reconstruction is done 100 times for measuring runtime and plotted in Fig 5(a). The simulation shows an average speed up of GPU vs. CPU by a factor of 3.1 times. The spatial resolution of airplane image is 1024x1024. Both CPU based and GPU based reconstruction is done 100 times for measuring runtime. The simulation output in Fig 5(b) shows an average speed up of GPU by a factor of 4.1 time speedy than CPU.We have used earth2k image having resolution of 2048x2048. For this image, both CPU based and GPU based reconstruction is done 100. The simulation result in Fig 5(c) shows an average speed up of 4.349 times faster in GPU than CPU.

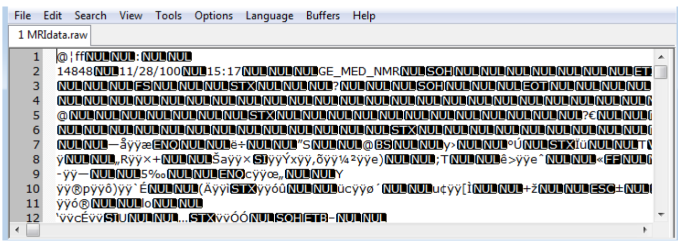
a) lena [512 × 512]

b) airplane [1024 × 1024]



c) earth2k [2048 × 2048]

(a) Experimental Images



(b) MRI Raw data file

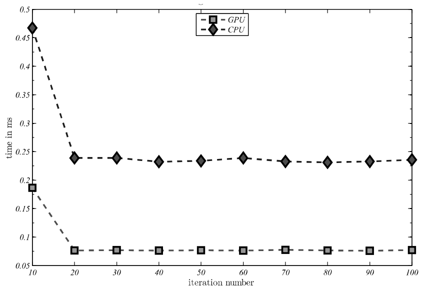
Fig. 4. Experimental (a) Images and (b) Data

FFT and IFT time for 100 iterations are measured and plotted total time requirement in Fig 5(d). From the performance measures of different image size has been observed. The result in Fig 5(d) the reconstruction performance of GPU over CPU was augmented gradually while the image size increased.

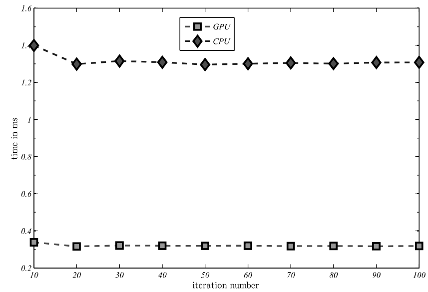
7 GPU Based MR-Image Reconstruction

Raw data file is opened and the proposed system separates header info and k-space data. Then raw k-space data is separated into real image and imaginary image data space. After that spatial frequency data is created from both real and imaginary data. Spatial frequency data is resized by reducing the dimension data and matrix transpose operation. The frequency image is shown as in Fig 6. After applying the IFT, the reconstructed MR Phantom Image is acquired. The visual property of CPU and GPU based reconstructions are similar.

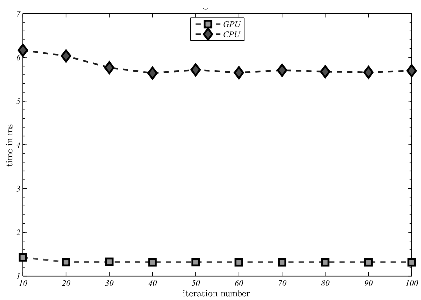
The GPU Based FFT improves the performance of the reconstruction algorithm by a factor of 8.2 which is showed in Fig 7. The resulting image quality is



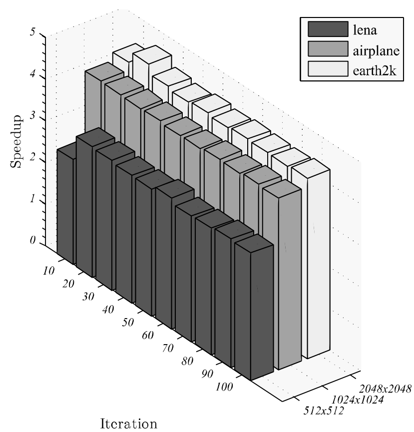
(a) lena image



(b) airplane image

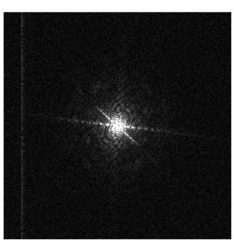


(c) earth2k image

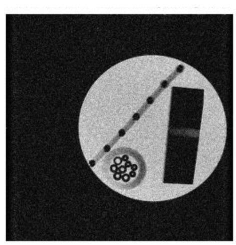


(d) Average Performance

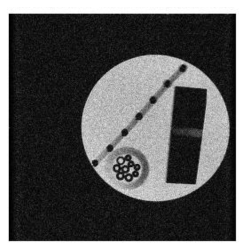
Fig. 5. GPU vs. CPU Performance of Reconstruction for (a) Lena Image (b) Airplane Image (c) Earth2k Image and (d) Speedup of Image Reconstruction by Image Size



a) Raw Data



b) MRI Reconstruction by CPU



c) MRI Reconstruction by GPU

Fig. 6. MRI Reconstruction Outputs in both CPU and GPU

visually same for both CPU-based and GPU-based implementations. In future, the real-time GPU-based MR image reconstruction will be possible.

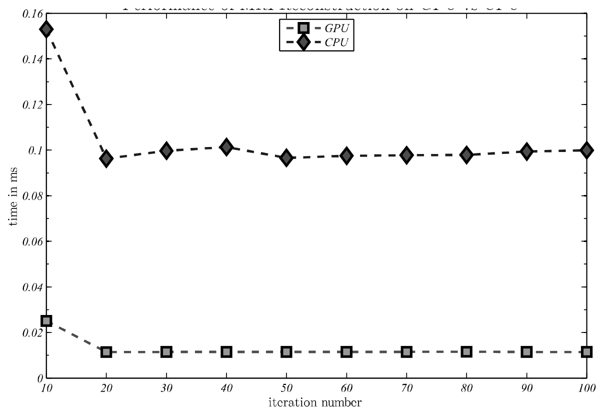


Fig. 7. MR Image Reconstruction Performance of GPU over CPU

8 Limitations and Future Work

Currently the work can perform only on data that resides in GPU memory. External memory algorithms based on the hierarchical algorithm can be designed to handle larger data. Computation can also be performed on multiple GPUs. It can also possible to do other processing, such as de-noising and de-blurring on MRI raw data before output the reconstructed image.The work can be ported to real-time visualization of MR image that is very important for envision situation of sensitive internal organs while operating a patient.

9 Conclusion

Besides the performance advantage of using a GPU over a CPU for MR-image reconstruction, there are other advantages as well. In MRI device, the CPU can be preoccupied with time-critical tasks such as controlling the data acquisition hardware. In this case, it is beneficial to use the GPU for image reconstruction, leaving the CPU to do data acquisition. Moreover, because of the GPU is free of interrupts from the operating system, it results better performance than interrupt driven CPU. The rate of increase in performance of GPUs is expected to outshine that of CPUs in the next few years, increasing the demand of the GPU as the processor of choice for medical applications.

Acknowledgment. This work was supported by Hankuk University of Foreign Studies Research Fund of 2011.

References

1. Bockenbach, O., Knaup, M., Kachelrie, M.: Implementation of a cone-beam backprojection algorithm on the Cell Broadband Engine processor. In: SPIE Medical Imaging 2007: Physics of Medical Imaging (2007)

2. Mueller, K., Xu, F., Neophytou, N.: Why do commodity graphics hardware boards (GPUs) work so well for acceleration of computed tomography? In: SPIE Electronic Imaging 2007 (2007)
3. Mueller, K., Yagel, R.: Rapid 3-D cone-beam reconstruction with the simultaneous algebraic reconstruction technique (SART) using 2-D texture mapping hardware. IEEE Transactions on Medical Imaging 19, 1227–1237 (2000)
4. Chidlow, K., Möller, T.: Rapid emission tomography reconstruction. In: Int'l Workshop on Volume Graphics (2003)
5. Xue, X., Cheryauka, A., Tubbs, D.: Acceleration of uro-CT reconstruction for a mobile C-Arm on GPU and FPGA hardware: A simulation study. In: SPIE Medical Imaging (2006)
6. Virtual Imaging Laboratory, Department of Biomedical Engineering, Duke University, http://dukemil.bme.duke.edu/MRI/index.html
7. Liang, Z.P., Lauterbur, P.C.: Principles of Magnetic Resonance Imaging. A signal processing perspective. IEEE Press (2000)
8. Haacle, E.M., Liang, Z.P.: Challenges of Imaging Stucture and Function with MRI. IEEE Transactions on Medicine and Biology 19, 55–62 (2000)
9. Nishimura, D.G.: Principles of Magnetic Resonance Imaging (April 1996)
10. MRI Basics, http://www.cis.rit.edu/htbooks/mri/inside.htm
11. Haacle, E.M., Liang, Z.P.: Challenges of Imaging Stucture and Function with MRI. IEEE Transactions on Medicine and Biology 19, 55–62 (2000)
12. Uzun, I.S., Bouridane, A.A.A.: FPGA Implementations of Fast Fourier Transforms for Real-Time Signal and Image Processing. In: Proceedings. 2003 IEEE International Conference on Field-Programmable Technology, FPT (2003)
13. Zhuo, J., Gullapalli, R.P.: MR Artifacts, Safety, and Quality. Physics Tutorial for Residents, pp. 275–279 (2006)
14. Aibinu, A.M., Salami, M.J.E., Shafie, A.A., Najeeb, A.R.: MRI Reconstruction Using Discrete Fourier Transform: A tutorial. World Academy of Science, Engineering and Technology (2008)
15. Moratal, D., Valls-Luch, A., Mart-Bonmat, L., Brummer, M.: k-Space tutorial: an MRI educational tool for a better understanding of k-space. Biomedical Imaging and Intervention Journal (2008)
16. Chaari, L., Pesquet, J.-C., Benazza-Benyahia, A., Ciuciu, P.: Autocalibrated Parallel MRI Reconstruction in the Wavelet Domain. In: IEEE International Symposium on Biomedical Imaging, Paris, France, May 14-17, pp. 756–759 (2008)
17. Schultz, G., et al.: K-Space Based Image Reconstruction of MRI Data Encoded with Ambiguous Gradient Fields. In: Proc. of International Society for Magnetic Resonance in Medicine (2011)
18. Stone, S.S., et al.: Accelerating Advanced MRI Reconstructions on GPUs. In: Proceedings of the 5th International Conference on Computing Frontiers (2008)
19. Kleut, D., Jovanovi, M., Reljin, P.D.B.: 3D Visualisation of MRI images using MATLAB. Journal of Automatic Control 16(1-3) (2006)