AGILE: An open source library for image reconstruction using graphics card hardware acceleration

F. Knoll[1], M. Freiberger[1], K. Bredies[2], and R. Stollberger[1]

[1]Institute of Medical Engineering, Graz University of Technology, Graz, Austria, [2]Institute for Mathematics and Scientific Computing, University of Graz, Graz, Austria

Introduction: Iterative image reconstruction of undersampled data from multiple coils has shown a huge potential for a wide range of applications during the last years [1,2]. While pronounced improvements in imaging speed were achieved with the proposed methods, the reconstruction times of most implementations were too long for online image reconstruction, which is necessary in order to facilitate application in clinical routine. Parallelized implementations on graphics hardware were recently established as a feasible method to significantly speedup image reconstruction [3]. While programming graphics hardware was simplified significantly with the introduction of dedicated libraries for general purpose computing like CUDA [4] or OpenCL [5], efficient implementation, especially concerning memory management, is still a challenging task. However, the mathematical operations needed for image reconstruction are often very similar, and can be re-used in future implementations. These include basic matrix-vector operations, algorithms to solve linear systems of equations, (non-uniform)-FFTs and solvers for constrained optimization problems. The goal of this work is to introduce the open source library *AGILE (Environment for Linear and non-linear Image reconstruction using Gpu Acceleration)*, designed for image reconstruction on GPUs in biomedical sciences. It is based on highly efficient implementations of numerical methods, but also includes code for iterative MR image reconstruction as well as a framework for finite element calculations and applications for Fluorescence Tomography [6].

Materials and Methods: **Library design:** The *AGILE* library is based on highly efficient implementations of basic matrix-vector and vector-vector operations [7]. A collection of solvers for linear equation systems has been implemented into the library. Currently these include the minimum residual method (MINRES), the conjugate gradient method (CG), the preconditioned CG (PCG), the complex orthogonal CG (COCG) and the generalized minimal residual method (GMRES). The code of the library is available as open source under the GNU General Public License version 3 (GPLv3) at www.imt.tugraz.at. *AGILE* builds on Nvidia's CUDA library for the communication with the graphics card. The library itself is written in C++ and makes heavy use of the object-oriented design pattern and template classes both of which facilitate the rapid development of applications. Additionally, the class design follows established standards like the standard template library (STL) [8].

MR image reconstruction experiment: An example for TGV [9] constrained iterative image reconstruction of undersampled radial data from multiple coils with *AGILE* is presented. A fully sampled T2-weighted turbo spin echo scan of the brain of a healthy volunteer was performed (3T scanner, 4-channel head coil, TR=5000ms, TE=99ms, turbo factor 10, matrix size (x,y)=(256,256), slice thickness=4mm, in-plane resolution=0.86mm×0.86mm). Written informed consent was obtained prior to the examination. Raw data was exported from the scanner and then subsampled, to simulate an accelerated acquisition. Images were reconstructed from 96, 48, 32 and 24 radial projections. Each projection consisted of 256 sample points. Calculations were performed on a conventional desktop PC (Intel Core 2 DUO E6600 2.40GHz, 2GB RAM), using Matlab R2009b (64bit). While this version of Matlab already offers the possibility to perform calculations in parallel on multiple core systems, the actual parallelization performance cannot be determined exactly. To provide a clear reference implementation, it was therefore ensured that only a single CPU core was used for the CPU calculations. The CUDA implementation was compiled with release 3.0 of the NVCC compiler and was tested on a NVIDIA GTX480 GPU with 1536MB memory.

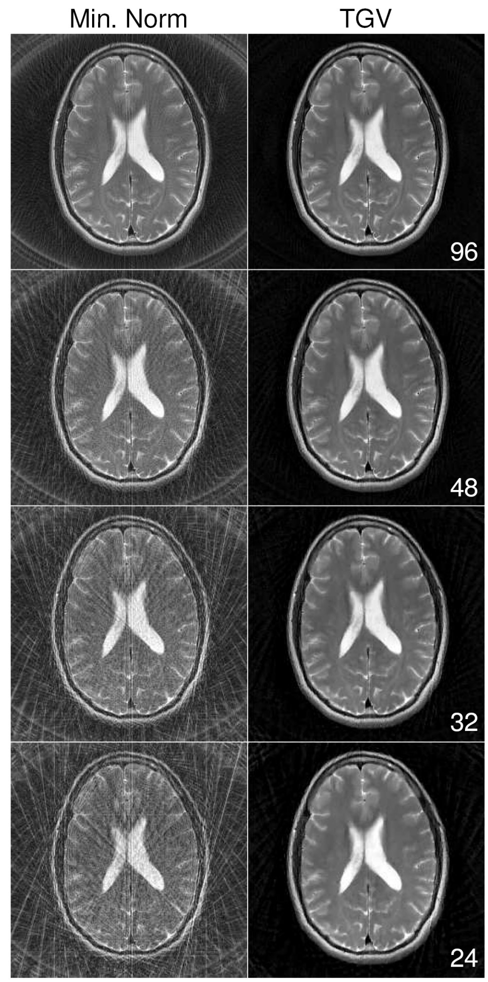

Figure 1: Image reconstruction from undersampled radial data with 96, 48, 32 and 24 projections. Conventional minimum norm (left) and CUDA TGV constrained reconstructions (right) are shown.

Results: Reconstruction results of conventional minimum solutions (regridding reconstruction with density compensation of each individual receiver channel, followed by a sum of squares combination), as well as reconstructions with a TGV constraint are shown in **Figure 1**. Conventional reconstructions suffer from pronounced streaking artifacts which get increasingly worse as acceleration is increased. These artifacts are eliminated efficiently with the TGV based method. The corresponding reconstruction times are given in **Table 1**. Speedups up to 40 were achieved with the AGILE implementation. While a fixed number of iterations (500) was used for both implementations, it should be noted that the reconstruction time of both implementations could be decreased by optimizing the number of iterations. In data sets with less corruption (e.g. the 96 projections data set), no visual changes of the reconstructed image occur after the first 150 iterations while more iterations are needed for data with higher corruption. However, the goal of this experiment was the comparison of the GPU and the CPU implementation. A reduction of iterations affects both implementations in the same way. In order to keep the experimental setup as simple as possible, the number of iterations was fixed to a rather high value which ensures that all artifacts are eliminated in all data sets.

Data set	Matlab reconstruction time	*AGILE* reconstruction time	Speedup
96 projections	253.57s	15.90s	15.88
48 projections	227.47s	8.76s	25.97
32 projections	220.33s	6.36s	34.64
24 projections	217.12s	5.11s	42.49

Table 1: Comparison of the reconstruction times of Matlab CPU and *AGILE* GPU implementations of iterative radial image reconstruction with a TGV constraint. Computational results are shown for data with 96, 48, 32 and 24 radial projections.

Discussion: The results from this work illustrate the pronounced computational speedup with the GPU implementation of the *AGILE* library for iterative image reconstruction. As the toolbox is an open-source project, all algorithms can be used either as a black box for the reconstruction of new data, or as a basis for own implementations for similar problems. The latter case is facilitated by the object-oriented and templated design which provides a well defined structure for extensions and allows for maximum code reusability. The intention behind the release of this open-source library is to alleviate the usage of graphics hardware in medical image reconstruction, which has the potential to bring a wide range of interesting methods from research labs to daily clinical practice.

References: [1] Block et al., MRM 57: 1086-1098 (2007), [2] Lustig et al., MRM 64: 457-471 (2010), [3] Hansen et al., MRM 59: 463-468 (2008), [4] NVIDIA Corporation, NVIDIA CUDA – Programming Guide v3.2 (October 2010), [5] Khronos Group OpenCL 1.1 Specification (revision 36, September 30, 2010), [6] S. R. Arridge, Inverse Problems, 15:41–93 (1999), [7] M. Liebmann, "Efficient pde solvers on modern hardware with applications in medical and technical sciences," Ph.D. thesis, University of Graz, 2009. [8] M. H. Austern, Addison-WesleyLongman Publishing Co., Inc. (1998), [9] Knoll et al., Proc. Intl. Soc. Mag. Reson. Med. 18:4855 (2010).