# Memory-Saving Iterative Reconstruction on Overlapping Blocks of K-Space

Martin Uecker[1] and Michael Lustig[1]
[1]Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, California, United States

**Target audience:** implementation of iterative reconstruction algorithms, parallel MRI, graphical processing units

**Introduction:** The purpose of this work to develop memory-efficient implementations of iterative reconstruction algorithms, such as SENSE[1], nonlinear inversion[2], and ESPIRiT[3]. Although iterative reconstruction provides faster imaging and improved image quality, high computational demand limits its practical application for large 3D data sets. Highly parallel graphical processing units (GPUs) can reduce computation time, but a 3D reconstruction with $256^3$ voxels and 32 channels uses 4 GiB of memory just to store the coil sensitivities (!) which exceeds the memory of a typical GPU. For example, a GTX 580 (Nvidia, San Jose, CA) has 1.5 GiB on board memory. While progress has been made by decoupling the reconstruction into a series of 2D reconstructions[4], true 3D or even 4D (3D+time) reconstructions are preferable, because they can exploit redundancy of the image in all dimensions.

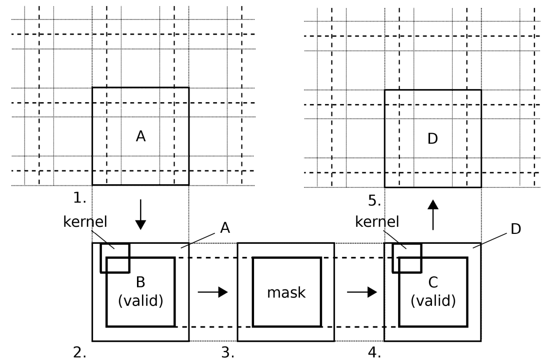


***Figure 1**: Proposed blockwise processing:*
*1.) Extraction of an overlapping block A*
*2.) Convolution with two applications of an FFT. The result is valid in region B (overlap-and-save).*
*3.) Multiplication of B with the sampling mask (in-place) and setting of the invalid part to zero.*
*4.) Adjoint convolution*
*5.) Accumulation of the extended result D into the output (overlap-and-add).*

**Theory:** Iterative reconstruction with multiple receivers typically consists of multiplying the image with the sensitivities, Fourier transforming each channel, applying a sampling mask (for data consistency), and reversing these operations to get an update for the image. This requires the sensitivities to be stored at full resolution. Multiplication in the image domain is equivalent to a convolution in k-space, which could make use of a much smaller representation of the sensitivities, but uses much more operations. Using the overlap-and-add (OaA) and overlap-and-save (OaS) algorithms operation count can be reduced by processing overlapping blocks and combining the results[5]. By making use of the typical structure of iterative algorithms, it is possible to combine OaA and OaS to process all k-space blocks independently (Fig. 1).

**Methods:** ESPIRiT (SENSE with autocalibrated Eigen-Vector based maps) has been implemented using the proposed algorithm and in the conventional way[1]. The implementation used the C language, OpenMP 3.1, and the FFTW 3 library. Data from a 3D IR-FLASH sequence (TR/TE = 12.5/5.2 ms, TI = 450 ms, FA = 20º, matrix size: 256x180x230) was acquired at 1.5 T (Signa Excite, GE, Milwaukee, WI) with an eight-channel head coil and was retrospectively undersampled. A block size of 32x30x46 was used for OaA and OaS and the sensitivities were stored as 17x17x17 Fourier coefficients (convolution kernel). Zero-padding to 272x196x246 avoided special treatment of the boundary.

| | Data | Image | Sensitivities | Temp. Mem. | Working Set |
|---|---|---|---|---|---|
| Conventional Implementation | 646.9 MiB 256x180x230x8 | 80.9 MiB 256x180x230 | 646.9 MiB 256x180x230x8 | 2 x 646.9 MiB 256x180x230x8 | 1698.3 MiB 5x Img + Tmp |
| Proposed Algorithm | 646.9 MiB 256x180x230x8 | 100.1 MiB 272x196x246 | 0.3 MiB 17x17x17x8 | 28.8 MiB (various) | 528.9 MiB 5x Img + Tmp |

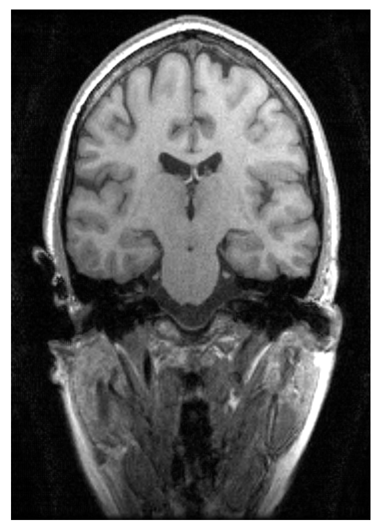

***Figure 2**: Section from a 3D volume reconstructed with the proposed technique (accl. 2x2).*

**Results and Discussion:** The table shows dimensions and size of different arrays used in both algorithms. The proposed method dramatically reduced the amount of memory for storing the sensitivities resulting in memory use (Working Set) reduction of 3-fold requiring less than 1 GiB. This is small enough for an implementation on a GPU. Apart from achieving this goal, the technique might also be useful on CPUs: Because it operates on small independent blocks, it allows to exploit caches and makes new parallelization strategies possible.

**Conclusion:** The presented technique allows significant savings in memory needed to store coil sensitivities for iterative algorithms. This is a prerequisite for the reconstruction of 3D data sets on GPUs. In addition, it opens up new opportunities for further optimization and parallelization.

**References: 1.** Pruessmann KP, Weiger M, Börnert P, Boesiger P. Advances in sensitivity encoding with arbitrary *k*-space trajectories. Magn Reson Med, 2001;46:638–651. **2.** Uecker M, Hohage T, Block KT, Frahm J. Image Reconstruction by Regularized Nonlinear Inversion - Joint Estimation of Coil Sensitivities and Image Content. Magn Reson Med. 2008;60:674-682. **3.** Uecker M, Peng L, Murphy JM, Virtue P, Elad M, Pauly JM, Vasanawala SS, Lustig M. ESPIRiT – An Eigenvalue Approach to Autocalibrating Parallel MRI: Where SENSE meets GRAPPA. Magn Reson Med 2012; submitted.
**4.** Murphy M, Alley M, Demmel J, Keutzer K, Vasanawala S, Lustig M. Fast l1-SPIRiT compressed sensing parallel imaging MRI: Scalable parallel implementation and clinically feasible runtime. IEEE Trans Med Imaging 2012; 31:1250–1262. **5.** Oppenheim AV, and Schafer RW. Discrete-Time Signal Processing. Prentice Hall. 3rd edition 2010.