# Motion Corrections in BART

Moritz Blumenthal, Martin Uecker

September 3, 2024

Institute of Biomedical Imaging, TU Graz

# Declaration of Financial Interests or Relationships

Speaker Name: Moritz Blumenthal

I have no financial interests or relationships to disclose with regard to the subject matter of this presentation.

# BART[1]: Software Toolbox for Computational MRI

## Purposes

- Rapid prototyping
- Reproducible research
- Clinical translation

## Command line tools for MRI reconstruction:

- Estimation of coil sensitivities
- Parallel Imaging (PI) + Compressed Sensing (CS)
- Non-linear (model-based) recos: NLINV, ENLIVE and MOBA

## Numeric Backend

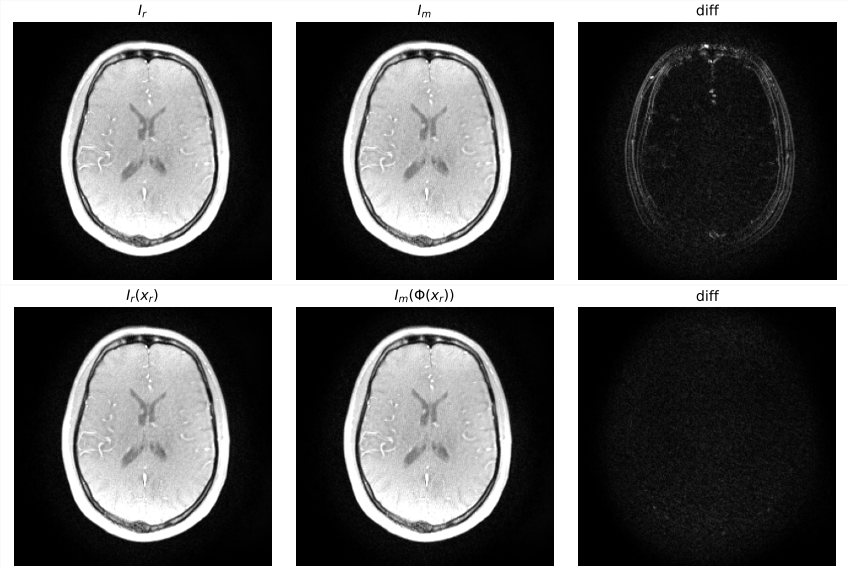
- Written in C, few external dependencies
- Operations on multi-dimensional arrays
- Accelerated by GPU, OpenMP and MPI


[1] M. Uecker et al. "Software toolbox and programming library for compressed sensing and parallel imaging". ISMRM Workshop on Data Sampling and Image Reconstruction. Sedona, 2013

# Motion correction, what's new?

- Motion estimation in image space
  - `$ bart affinereg <reference> <moved> <affine>`
  - `$ bart estmotion 7 <reference> <moved> <motion> [<imotion>]`
- Interpolation in image space
  - `$ bart interpolate -D 7 <moved> <motion> <interpolated>`
- Correction of rigid motion in k-space
  - `$ affine_kspace.sh <ksp_raw> <trj_raw> <affine> <ksp_cor> <trj_cor>`
- Motion fields into PICS reconstruction
  - `$ bart pics ... --motion-field=<imotion> <ksp> <col> <img>`

# Part 1: Rigid Motion Correction



# Part 2: iMoCo[1]-like Reconstruction Pipeline in BART

1. Prepare data from RAGA[2] measurement
2. Self-gating with SSA-FARY[3] (ssa)
3. Sensitivity estimation with NLINV[4] (ncalib)
4. Motion-resolved GRASP[5] reconstruction (pics)
5. Estimation of motion fields (estmotion)
6. Motion-compensated reconstruction (pics)

---

[2] N. Scholand et al. "Rational Approximation of Golden Angles: Accelerated Reconstructions for Radial MRI". Magn. Reson. Med. 2024; accepted

[3] S. Rosenzweig et al. "Cardiac and Respiratory Self-Gating in Radial MRI Using an Adapted Singular Spectrum Analysis (SSA-FARY)". IEEE TMI. 2020; 39:3029–3041

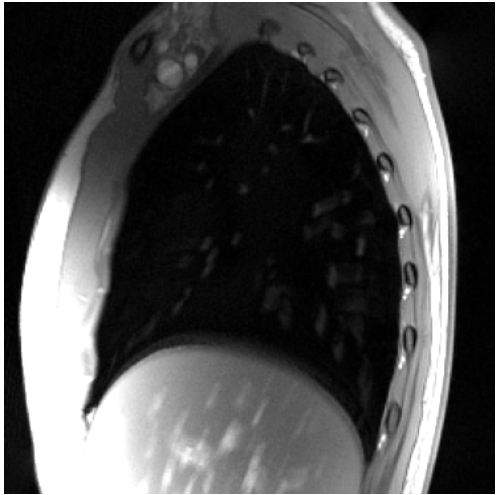
[4] M. Uecker et al. "Image reconstruction by regularized nonlinear inversion—Joint estimation of coil sensitivities and image content". Magn. Reson. Med. 2008; 60:674–682

[5] L. Feng et al. "Golden-angle radial sparse parallel MRI: Combination of compressed sensing, parallel imaging, and golden-angle radial sampling for fast and flexible dynamic volumetric MRI". Magn. Reson. Med. 2014; 72:707–717

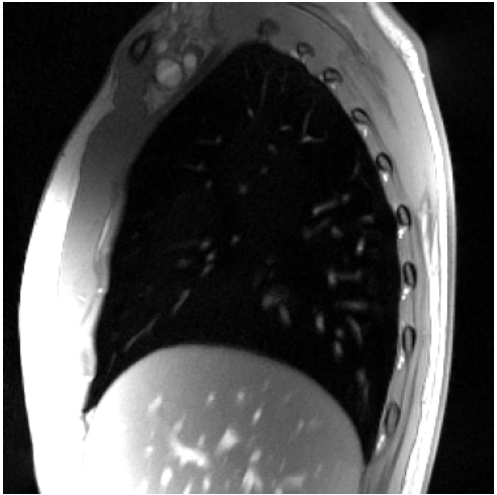
[1] X. Zhu et al. "Iterative motion-compensation reconstruction ultra-short TE (iMoCo UTE) for high-resolution free-breathing pulmonary MRI". Magn. Reson. Med. 2020; 83.4:1208–1221

# Part 2: iMoCo-like Reconstruction in BART

Motion Resolved



iMoCo-TV





demo_motion.ipynb

demo_motion.ipynb > Motion Correction with BART

Code | Markdown | Run All | Restart | Clear All Outputs | Variables | Outline | base (Python 3.12.2)

# Motion Correction with BART

This tutorial uses the BART command-line interface (CLI) and presents how to perform motion corrected reconstructions with BART.

## Outline

1. Bart Setup
2. Conventions in BART
3. Rigid body motion correction
4. Non-rigid motion fields for PICS

**Author**: Moritz Blumenthal

**Presenter**: Moritz Blumenthal

**Institution**: Institute of Biomedical Imaging, Graz University of Technology, Austria

## 1 Setup BART

This notebook is designed to run on a local system and on Google Colab. It uses the python kernel, however, almost all commands use the `%%bash` cell magic to be executed in a `bash` subshell.

### 1.1 Google Colab

To run BART on Google Colab, this notebook automatically installs dependencies and sets up the GPUs if the environment variable `COLAB=1` is set. If you run this notebook on your local system, you might not want this setup. Please set `COLAB=0` in this case. For a detailed explanation, see the How to Run BART on Google Colaboratory.

```
[1]  %env COLAB=0
     ✓ 0.0s                                                    Python
...  env: COLAB=0
```

Not all GPUs on Google Colab support CUDA 11, we downgrade CUDA if necessary:

```
     %%bash
     [ $COLAB -ne 1 ] && echo "Skipp cell (not on Colab)" && exit 0
```

Cell 1 of 67