PULSAR Version 1.00

Parallel Imaging Utilizing Localized Surface-coil Acquisition and Reconstruction

USER MANUAL

Jim Ji, Swati Rane,
Dan Spence and Jong Bum Son

Magnetic Resonance Systems Lab
Department of Electrical Engineering
Texas A&M University

COPYRIGHT

Copyright 2005, Department of Electrical Engineering, Texas A&M University.
All rights reserved. See the file COPYRIGHT for more information.

You are granted a license to use the Software for academic, research and internal business purposes only, without a fee. You may distribute the software to third parties provided that the copyright notice and this statement appear on all copies and that no charge is associated with such copies. If you make derivative works and distribute it, you will (1) notify PULSAR developers regarding its distribution of the derivative work, and (2) clearly notify users that such derivative work is a modified version and not the original PULSAR distributed by Texas A&M University.

We MAKE NO REPRESENTATIONS ABOUT THE SUITABILITY OF THIS SOFTWARE FOR ANY PURPOSE. IT IS PROVIDED "AS IS" WITHOUT EXPRESSED OR IMPLIED WARRANTY. WE SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY THE USERS OF THIS SOFTWARE.

Any Licensee wishing to make commercial use of the Software should contact:

Jim Ji
Department of Electrical Engineering
Texas A&M University
College Station, TX 77843-3128
jimji@tamu.edu

INDEX

USER MANUAL

Jim X. Ji, Swati D. Rane
Department of Electrical Engineering, Texas A&M University

1. Introduction

This manual is written with the first-time user in mind, to explain how to use this software tool. This software package allows the user to simulate different parallel magnetic resonance imaging techniques such as

1. **Sens**itivity **E**ncoding (SENSE),
2. **P**arallel **I**maging with **L**ocalized **S**ensitivities (PILS),
3. **S**i**M**ultaneous **A**cquisition of **S**patial **H**armonics (SMASH),
4. **G**ene**R**alized **A**utocalibrating **P**artially **P**arallel **A**cquisition (GRAPPA) and
5. **S**ensitivity **P**rofiles from an **A**rray of **C**oils for **E**ncoding and **R**econstruction **I**n **P**arallel (SPACE RIP).

The software tool goes through the entire process of data input, sensitivity estimation and image reconstruction.

The reconstructed image can be then be used to analyze the performance of the reconstruction technique by calculating the following parameters:
1. Signal to Noise Ratio (SNR)
2. Artifact Power
3. 'g' factor
4. Contrast/ contrast to Noise Ratio (CNR)
5. Resolution Phantom

The entire procedure is broken down into three steps. Each step can be executed individually to analyze the process of parallel magnetic resonance imaging.
Individual functions are defined for every process within the step. The user can replace a function with his/her own function to test the code.

The structure and details of the program execution is provided. A Quick-Start guide is also available.

2. Quick Start

A few example files are available in the '***workdir/***examples' folder.
Select one of the example files:

1. `SENSE1_Inputs_List.m`: SENSE from simulated data
2. `SENSE2_Inputs_List.m`: SENSE for real data (example dataset `data1.mat`)
3. `SENSE3_Inputs_List.m`: SENSE from simulated data with a non-linear coil array for a coronal slice using reference maps.
4. `PILS_Inputs_List.m`: PILS for real data (example dataset `data2.mat`)

5. `SMASH_Inputs_List.m`: SMASH for simulated data
6. `GRAPPA_Inputs_List.m`: GRAPPA for simulated data
7. `GRAPPA2_Inputs_List.m`: GRAPPA for real data (Example dataset `data1.mat`) with MCMLI reconstruction
8. `SPACERIP_Inputs_List.m`: SPACERIP for simulated data.

Then, rename and save the file as 'Inputs_list.m' in the folder '***workdir*/**source/common'. Executing the `Run_file.m` in '***workdir*/**source/common' will produce an image. The steps in the image reconstruction process can also be viewed in the command window.

For `SENSE3_Inputs_List.m,` you will be prompted to enter the number of coils in each direction.

E.g.

```
How many coils in the array along Y axis?
How many coils in the array along X axis?
```

X is the direction of subsampling. The values should multiply to equal the number of receiver coils.

For `GRAPPA2_Inputs_List.m,` you will be prompted to enter the number of columns to be considered.

E.g.,

```
How many columns to consider for MCMLI reconstruction for GRAPPA?
```

You can modify example files to try different configurations.

Otherwise follow the following procedure:

2.1 Alternative Start-up

1. Open the `Inputs_list.m` file.
2. Select a value for `data_type` from the following options:
   1: Complete simulation - image will be selected from the software package itself
   2: Use file containing image as input
   3: Use full scan kspace data entered as a file
   4: Use reduced scan kspace data as a file.
3. Enter file containing the data. See Section 3.1 for further details about data input.
4. Set all flags to 1.
5. If `data_type = 1` or `data_type = 2`, enter all the parameters for the structure `Receiver` or leave them as is. If `data_type = 3` or `data_type = 4`, let `Receiver.sensitvity_estimation=2`. Enter a value for `Receiver.num_coils`.
6. If `data_type = 4,` enter the image's dimension in `Image_size`, otherwise leave as is.
7. Enter all parameters for the structure `Sampling`. Values can be kept unchanged but adjust `Sampling.center_locs` suitably.
8. Select a value for `option` to specify a method for image reconstruction.
9. Leave structure `Misc` unchanged.
10. Select a value for per_list. See Section 3.3 for further details about parameter options.
11. Save Inputs_list.m.

12. Open and run `Run_file.m`.
13. In the case of an error, a message will be displayed in the command window.
    E.g. `per_list=7`
    Message in command window:
    ```
    Error: Variable per_list can take values from 1 to 6. Please
    correct errors first!
    ```
14. If there is no error, the command window shows:
    ```
    Loading files and generating kspace data...
    Estimating sensitivity...
    Reconstructing image...
    ```
15. Depending on the choice of performance analysis method, user may be asked to enter file names for multiple acquisitions or select regions to make SNR calculations. See Section 3.3 for further details about performance evaluation or Section 3.2 for more information about the image generated.

2.2 Using the GUI

Move the files `PMRI_Toolbox.m` and `PMRI_Toolbox.fig` from **workdir**/gui to **workdir**/source/common. Type `PMRI_Toolbox` into the command line of the command window to open the interface.

E.g.
```
>>PMRI_Toolbox
>>
```

Select the appropriate options and fill in all required data in the order specified by the numbers in the top-right corner of the boxes.

1. Data Input
Click desired option for data input. If using an image from the toolbox, continue to step 3.

2. File Input
Enter the file names in the appropriate fields. See Section 3.1 for more information about the format of the files.

3. Reconstruction
Click desired method for image reconstruction. Unless an image with reduced kspace data is being used, continue to step 5.

4. Image Details
Enter the total number of phase encodings and frequency encodings in the appropriate fields.

5. Sampling
Enter the desired subsampling factor. This should be an integer less than the number of coils (receiver elements) to be entered in step 6. Choose the subsampling direction. Enter the number of extra center lines for sensitivity estimation. Enter the locations of the first and last extra lines.

If reduced kspace data was selected in step 1, choose the type of data sampling. If you wish to enter your own sampling locations, select method 3 (Enter separately) and enter a vector containing the locations.

Sample input would be in the format 1:16 17:4:57 58:72 73:4:112 113:128

If the SPACERIP method was selected in step 3, choose the sampling type. Usually options 2 and 3 result in a higher condition number of the reconstruction matrix and hence poorer reconstructions.

6. Additional inputs for simulation
If using simulated data, enter the number of receiver elements (coils), coil localization. Select the type of array. Enter the value for the reference SNR. The number of receiver elements should be an integer greater than the subsampling factor entered in step 5. Coil localization should be an integer value between 1 and 20. If 1, then the coil is highly localized. If 20, then the coil is not very localized. If a non-linear array is selected, values for the number of coils along the X and Y axis, as well as the depth, will be prompted for in the MATLAB command window.

Enter the background phase frequency.

If SMASH was selected in step 3 and tight harmonic fitting is desired, check the appropriate box. Enter an integer between 1 and 100 to declare the percentage of space occupied by the object in the FOV. I.e. enter 75 for 75% of the FOV to be occupied by the object.

If GRAPPA was selected in step 3, enter the number of blocks and number of reconstructions.

Select the method for sensitivity estimation. To divide by reference, coil data must be available either as reference scans or full FOV scans in the file `ref_data.mat`. The SOS method should be used only if a few lines are collected in the center of the kspace. The sensitivity estimation is then done by obtaining low resolution coil images and dividing them by the sum of squares. For the SVD method, coil images are obtained using principal component analysis for better background noise suppression.

Select a filtering method. Currently, only the no filtering and the polynomial filtering options are available.

7. Performance analysis
Click the image analysis desired.

For SNR (2 acquisitions), two acquisitions are required.
For Pixelwise SNR, at least two acquisitions are required. For approximate results, 7 to 10 acquisitions are necessary. Otherwise, at least 50 reconstructions are required. In both cases, user will be prompted in the MATLAB command window.

For Artifact Power calculation, a reference image must be available.

Resolution analysis is not included. It is only available as a simulation result using a resolution phantom.

Selecting ‘No evaluation’ merely returns the reconstructed images without further analysis.

8. Click ‘RUN’ to perform image reconstruction and process evaluation.

3. Organization and Structure of the Tools

The software can be structured as shown below:

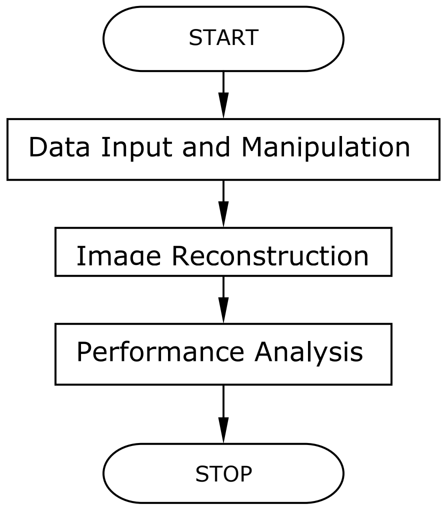

Fig.1: Broad overview of the software structure

3.1. Data Input and Manipulation

File format:
- Kspace data is entered in the standard MATLAB format as a `filename.mat` file.

- Each `.mat` file contains one variable.
  The variable name must be the same as the file name.
  E.g.
  File `kspace.mat` contains variable `kspace` ONLY.

- 2D image data can be input in the following formats:
  `filename.mat` (where the file name is the same as the variable name),
  `filename.gif`, `filename.bmp`, or `filename.jpg`.

Data format:
- 2D image data can be entered.

- 3D Kspace data can be input. If `PEs`, `FEs` and `num_coils` are the number of phase encodings, frequency encodings and number of coils, then data can be entered as
  `PEs X FEs X num_coils` or
  `FEs X PEs X num_coils`.

- Extra data, data from reference scans, etc. should be entered as mentioned previously.

- Data files are added in the folder '***workdir*/**examples/data'**.**

- If reduced data and calibration data are collected separately, then field of view (FOV) must be accurately known for correct location of the calibration data.
  E.g.
    Actual FOV = 10 cm
    Data size = 256 X 256
    10cm ≈ 256 lines
    1cm ≈ 25 lines (25.6 exact)
    10/256 cm ≈ 1 line

  If calibration data is collected from the 5^{th} cm then the line location will be found as 5 cm -> 25.6 X 5 = line no. 128.
  If the next line is at 5.04 cm, then the line no. is 129 and so on. These locations are stored in a vector and entered for the variable `center_locs (Sampling).`

3.2. Image Reconstruction Process

- The image reconstruction process is executed depending on the type of data input in the toolbox.

- The reconstructed image `I_rec` is available in the file `parameters.mat` (located in ***workdir*/**source/parameters.mat) and the parameters in the structure `Perform_parameters`.

To view this image, enter the following command in the command window.

```
>>imagesc(abs(I_rec));
```

To view the image in black and white or to view the color scale, enter the following:

```
>>colormap('gray');
>>colorbar;
```

3.3. Performance Evaluation

The method of reconstruction is analyzed on the basis of a number of parameters selected during the program.

-For SNR (2 acquisitions) – Method 2 calculation, two acquisitions are necessary. User will be prompted to enter data twice.

-For Pixelwise SNR Calculation - Method 3, at least two acquisitions are required. For approximate results, 7 to 10 acquisitions are necessary. Otherwise, at least 50 reconstructions are required. Each acquisition is saved separately.

-If SNR analysis was selected for performance analysis, user will be prompted to select the region of interest and region of noise for SNR measurement. The mouse turns into a cross-hair on the image. Decide on the region and click on the diagonal points (top-left and then bottom-right corners, in that order) of the region.

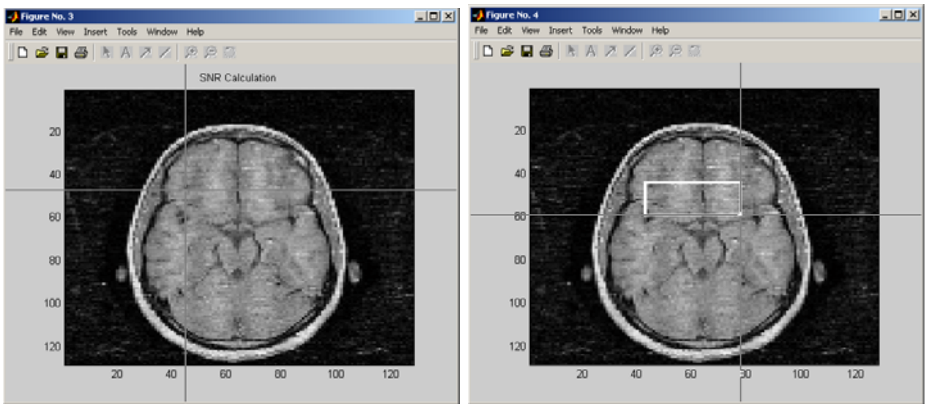

Fig.2: Selecting a region

A rectangle appears showing the region selected, as seen in Figure 1. Repeat procedure for selecting a region of noise.

- For the Artifact Power Calculation, a reference image must be available.

- For checking resolution, the resolution phantom must be selected. Hence, it is available only as a simulation result.
  Some resolution phantoms are provided. A ray phantom and a square phantom are generated by `rays_image.m` and `res_image.m`, respectively.

- If `SNRbox`, `Artifact_Pwr`, `box_snr` are used as a stand-alone command (i.e. not called through variable `per_list`), the return value is stored in a temporary variable `ans`.

E.g.

1. `>>SNRbox(I_rec)`
   SNR value is stored in ans
2. `>>snr_value=SNRbox(I_rec)`
   SNR value is stored in `snr_value`
3. If `per_list` is set to 1 in `Input_list.m`, the value will be displayed in the command window. Otherwise, the value can be recalled by the following commands:
   ```
   >>load parameters
   >>Perform_parameters
   ```

The performance analysis parameter is saved in the structure `Perform_parameters.` The structure and the reconstructed image are stored in the file `parameters.mat`.

---

4. Control Variables

The various inputs that can be fed to the program are listed alphabetically in the following table. Not all are required for a particular simulation. Parenthesis indicate the structure including the variable. Most of the inputs are assigned to a structure. The structures are:

Files
Image_size
Receiver

Sampling
Flags
Misc

E.g.
auto_smash_flag (Flags): referred as Flags.auto_smash_flags

4.1. Variables

Variable / Inputs	Type	Range	Description
auto_smash_flag(Flags)	Integer	0-1	0: Perform Regular SMASH Reconstruction 1: Perform AUTO-SMASH Reconstruction
backgroundphase_freq (Receiver)	Float		For phase variations in the sensitivity estimation
blocks (Misc)	Integer		Choose no. of blocks for GRAPPA reconstruction
calibration (Files)	Text		File containing central data for sensitivity estimation
center_lines(Sampling)	Integer		
center_locs (Sampling)	Vector of Integers		Vector with locations of the center lines for self calibration.
coils	2 X 1 matrix		Number of coils in the array in the Z and X direction
depth	float		depth
data_type	Integer	1 – 4	1: Complete simulation, Image will be selected from the software package itself. 2: Use file containing image as input 3: Use full scan kspace data entered as a file 4: Use reduced scan kspace data as a file
direction (Sampling)	Text	X, Y	Direction of phase encoding/sub-sampling X: Sub-sampling along columns Y: Sub-sampling along rows
FEs (Image_size)	Integer		Total frequency encodings
fit_img (Flags)	Integer	0-1	0: Do not fit harmonics tightly in SMASH 1: Fit harmonics tightly in SMASH
lin_flag (Flags)	Integer	0-1	0: Non-linear array 1: Linear array
max_sensitivity (Receiver)	Integer		Maximum sensitivity for the simulated receivers

num_coils(Receiver)	Integer		Number of coils
option	Integer	1-5	1: SENSE 2: PILS 3: SMASH 4: GRAPPA 5: SPACE RIP
PEs (Image_size)	Integer		Total phase encodings
per	Float	1-100	Percentage of FOV the object occupies
per_list	Integer	1-6	1: SNR Method 1 2: SNR 2 acquisition 3: Pixel wise SNR 4: Artifact Power 5: Resolution 6: No evaluation
poly_tick (Flags)	Integer	0-1	0:Do not perform polynomial filtering 1: Performa polynomial filtering
psi	Matrix		Noise correlation matrix
recons (Misc)	Integer		No. of reconstructions in GRAPPA
reference_scan (Files)	Text		File containing reference scans
reference_image (Files)	Text		File containing reference body coil image
regularization_flag (Flags)	Integer	0-1	0: No regularization for SENSE 1: Regularization for SENSE
samp (Misc)	Integer	1-4	1: Uniform lines 2: All lines in the center 3: Alternate lines in center 4: Non-uniform sampling 5: Use extra lines
sampling_location (Sampling)	Vector of integers		Vector containing the positions of the acquired lines
sensitivity_estimation (Receiver)	Integer	1-3	1: Use reference scans and reference image 2: Self calibration using SOS method 3: Self calibration using singular value decomposition.
sos_flag (Flags)	Integer	0-1	0: No iterative SOS Reconstruction 1: Iterative SOS Reconstruction
SNR (Receiver)	Float		Reference SNR for adding noise in a simulation
subsampling_factor(Sampling)	Integer		Reduction factor
variance_factor (Sampling)	Integer	1-5	Choose localization of the

			coil 1: Very localized 2: Not localized

The following table lists the variables required for different types of data input.

Variable / Inputs	Simulation	Image	Full kspace data	Reduced kspace Data
auto_smash_flag(Flags)	✓	✓	✓	✓
backgroundphase_freq (Receiver)	✓	✓		
blocks (Misc)	✓	✓	✓	✓
calibration (Files)			✓	✓
center_lines(Sampling)	✓	✓	✓	✓
center_locs (Sampling)	✓	✓	✓	✓
coils	✓	✓		
data_type	1	2	3	4
depth	✓	✓		
direction (Sampling)	✓	✓	✓	✓
FEs (Image_size)	✓	✓	✓	✓
fit_img (Flags)	✓	✓	✓	✓
lin_flag (Flags)	✓	✓	✓	✓
max_sensitivity (Receiver)	✓	✓		
mcmli_flag (Flags)	✓	✓	✓	✓
mcmli	✓	✓	✓	✓
num_coils(Receiver)	✓	✓		
option	✓	✓	✓	✓
PEs (Image_size)				✓
perc	✓	✓	✓	✓
per_list	✓	✓	✓	✓
poly_tick (Flags)	✓	✓	✓	✓
psi	✓	✓	✓	✓
recons (Misc)	✓	✓	✓	✓
reference_scan (Files)			✓	✓
reference_image (Files)			✓	✓
regularization_flag (Flags)	✓	✓	✓	✓
samp (Misc)	✓	✓	✓	✓
sampling_location (Sampling)				✓
sensitivity_estimation (Receiver)	✓	✓	✓	✓
SNR (Receiver)	✓	✓		
subsampling_factor(Sampling)	✓	✓	✓	✓
variance_factor (Sampling)	✓	✓		

5. Individual Subroutines

Individual programs, their inputs and outputs are described in the following table:

Functions / Subroutine	Description	Input	Output
arbitrarywire.m	Finds the corners of the coil loop to find magnetic field component along X ,Y, Z	Point on the grid where magnetic field has to be evaluated(x,y,z co-ordinates), end points of the wire	Magnetic field B at a point
addnoise.m	Adds noise in the simulated kspace data	Reference SNR SNR, ideal kspace data full_kspace_data	Noise kspace data
Artifact_Pwr.m	Calculates Artifact Power	Reference image, Reconstructed image I_rec	Artifact Power
auto_smash.m	AUTO_SMASH Reconstruction	data_type, full/reduced kspace data*, sampling_location, PEs, FEs	Reconstructed Image, I_rec
box_snr.m	Calculates SNR with 2 acquisition	Reconstructed images of 2 acquisitions	SNR
Bxarbwire.m Byarbwire.m Bzarbwire.m	magnetic field by analytical solution for magnetic field component along X-axis, along Y-axis, along Z-axis	Point on the grid where magnetic field has to be evaluated(x,y,z co-ordinates), length, direction and center of wire	X,Y and Z component of the B field
Error_chk.m	Checks if any parameters is entered incorrectly	All parameters	Error flag set to 1 if error present
estimate_maps.m	Estimates sensitivity profiles using SOS method	Full kspace data*, extra calibration data, sampling locations, data_type	Sensitivity maps, Regularization image**
evaluate_method .m	Main listing of performance analysis	data_type, per_list, flag for multiple/two acquisitions (two_acq), reconstructed image and files containing original data (data entered)	Value of performance parameters selected
field_map.m	Calculates the field map due to coil array	Aperature,FOV, Pix, Plane, offset, array_offset, theta, coils	Field map sensitivity
fitcurve.m, fitcurve_gauss.m	Gaussian fitting for SMASH and PILS	Sensitivity maps	Gaussian profiles, centers

			of coils
gencol_gen.m	Collected the necessary lines to regenerate a specific missed line	ACS, reduced kspace data with acquired lines at their proper locations (size= actual image size X num_coils), PES, FEs, start of the ACS lines-1, coil number (under reconstruction), num_coils, column number, blocks blocks, reconstructions recons, coils considered for generating coefficients consider_coils.	Correct column of acquired lines to generate the missed line
genmat_gen.m	Generated matrices for obtaining coefficients for GRAPPA reconstruction	Same as gencol_gen.m	Matrix of rearranged ACS lines for calculating coefficients
get_contour.m	Generates mask to extract object for polynomial filtering	Coil profile, threshold	Mask for profile
get_kspace_data.m	Generates reduced kspace data, computes sampling location	data_type, Files containing data, subsampling factor, number of receivers, localization factor, linear array	
grappa.m	GRAPPA reconstruction	reduced kspace data with acquired lines at their proper locations (size= actual image size X num_coils), number of ACS lines, location to start, sampling_location, consider_coils, blocks, recons	I_rec
idft.m	Computes inverse DFT	Fourier data, Size	IDFT of data entered
Inputs_list.m	List of user definable parameters -to be modified as per need		
kspacedata_sim.m	Simulates k-space data for a simulation	Image image, coil sensitivities, subsampling_factor	Full kspace data, reduced kspace data, sampling_location

make_profile, make_gauss_profile	fitting for SMASH and PILS, call fitcurve.m, fitcurve_gauss.m	Sensitivity maps	Gaussian profile. Centers of coils
myfun.m	Contains Gaussian function used for approximation		
non_linear_array.m	Generate sensitivities for a non-linear array	image, num_coils, variance_factor	Sensitivity profiles
pils_gauss.m	PILS reconstruction	rough_map, reduced_kspace_data, lin_flag, subsampling_factor	Reconstructed image I_rec
pixel_snr.m	calculates pixel-wise SNR	array containing reconstructed images from all acquisitions, loop_cnt	SNR map
PMRI_Toolbox.m	GUI to use software		
prepare_grappa2.m	Collects ACS lines and aligns them for GRAPPA reconstruction		
profile_1pd.m	Generates sensitivity profiles for the 1D Gaussian linear array	Coil localization, PEs, FEs, Coil centers background phase frequency	Coil profiles
rays_image.m	Generates the ray phantom for resolution check	-	image
sample_kd.m	Decimates full kspace data	full_kspace_data, subsampling_factor	reduced_kspace_data, sampling_location
sense.m	SENSE reconstruction	reduced_kspace_data, rough_map, psi	I_rec
sense_tikhnov.m	Regularized SENSE reconstruction	reduced_kspace_data, rough_map, regularization_image, psi	I_rec
sensitivity_estimation_process.m	Main program to call the selected the correct sensitivity estimation process	data_type, option, full_kspace_data, reduced_kspace_data, Receiver, Misc, Sampling, Flags, Image_size, per_list	rough_map, regularization_image
sensitivity_polyfilt.m	Polynomial filtering for sensitivity estimation	Noisy sensitivities, Order of the filter, and Neighborhood	Filtered sensitivities
smash.m	SMASH reconstruction	reduced_kspace_data, profiles, subsampling_factor,	I_rec

		sampling_location, fit_img(Flags), perc (Percentage FOV occupied by object)	
smash_fit_harmonic.m	Generates coefficients for harmonics	profiles, subsampling_factor, sampling_location, fit_img(Flags), perc (Percentage FOV occupied by object)	Smooth harmonics (sine and cosine parts), SMASH coefficients
SNRbox	Calculates SNR by method 1	I_rec	SNR
sos_recon.m	Iterative SOS reconstruction	I_rec, center_data	
spacerip.m	SPACE RIP reconstruction	reduced_kspace_data, rough_map, sampling_location	I_rec
square_loop_B.m	Calculates the corners of the wire loop to find the effect of each wire individually at a point	Length and width of each coil, centers of each coil, grid of points to evaluate B	Magnetic field map B
subsamp_gen.m	Non-uniform sampling for SPACE RIP	data_type, full_kspace_data, samp, total number of lines acquired, subsampling_factor	reduced_kspace_data, sampling_location
walsh_sensitivity.m	Sensitivity estimation using SVD	full_kspace_data, center_data, center_lines, subsampling_factor, num_coils, PEs, FEs, data_type	Sensitivity maps, Regularization image**

* If only reduced kspace data is available, full kspace data is set to 0.
** Regularization image for Regularized SENSE otherwise = 0.

6. Structures

To use the software tool as a whole, variables are passed in the form of 'structures' between functions. For running or using individual programs, the same programs are available in the folder ***individual codes***. The programs are the same and have the same name but variables are passed individually.

E.g.
In the entire tool box, the reduction factor is called subsampling_factor in the structure Sampling. This structure will also contain other variables. To use the variable, it must be referred to as Sampling.subsampling_factor. It is passed between functions within the structure Sampling, not as an individual variable. In independent programs, the function will pass the variable as subsampling_factor only. The exact variable needed for the execution of the function can be easily

found in the individual codes. Otherwise, the entire structure is passed and the variables from the structure that will actually be used cannot be clearly understood.