

Sep 03, 2022

Version 2

Processing of Radial K-space DW-MRI Data V.2

DOI

<https://dx.doi.org/10.17504/protocols.io.j8nlkkwj1l5r/v2>

Miguelrj¹, Hee Kwon Song¹, Stephen Pickup¹, Mamtaaryagupta¹, Rong Zhou¹

¹University of Pennsylvania, Department of Radiology



Miguel Romanello Joaquim

Create & collaborate more with a free account

Edit and publish protocols, collaborate in communities, share insights through comments, and track progress with run records.

Create free account

OPEN  ACCESS



DOI: <https://dx.doi.org/10.17504/protocols.io.j8nlkkwj1l5r/v2>

Protocol Citation: Miguelrj , Hee Kwon Song, Stephen Pickup, Mamtaaryagupta , Rong Zhou 2022. Processing of Radial K-space DW-MRI Data. protocols.io <https://dx.doi.org/10.17504/protocols.io.j8nlkkwj1l5r/v2> Version created by **Miguel Romanello Joaquim**

License: This is an open access protocol distributed under the terms of the **Creative Commons Attribution License**, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited

Protocol status: Working

We use this protocol and it's working



Created: September 02, 2022

Last Modified: September 03, 2022

Protocol Integer ID: 69520

Keywords: mri data, mri data the protocol, reconstruction of diffusion, kurtosis index, diffusion, metrics of adc, space data

Abstract

The protocol includes reconstruction of diffusion weighted images from radial k-space data and using a 3-parameter fit to derive metrics of ADC and kurtosis index.

1 Image reconstruction

Radially acquired diffusion-weighted images (DWIs) are reconstructed using the following steps:

1. Apply zero-order phase correction to each radial spoke using the average phase offset at the center slice of the lowest b-value image
2. Zerofill k-space by a factor of 2 to double field of view
3. Multiply signal of each point by its respective area on a Voronoi diagram of the points (including added zerofill points) in k-space
4. Re-grid each radially defined point to its nearest Cartesian coordinate using its Kaiser-Bessel index
5. Apply Fourier transform to now Cartesian-defined k-space

This process can be easily done using the **DWI Processing resources [link]**.

To use this, place both **DWI_shell.ipynb** and **DWI_processing_functions.py** in the same directory (your working directory). The **readme.txt** file outlines specific file formats, function inputs, and outputs.

Quickly, once all files are formatted according to **readme.txt**, the first cell defines all image and reconstruction parameters:

```
##### Parameters to change #####
input_file = 'fid' # k-space data
# Acquisition data
xres_ro = 128 # total readout points stored
views = 403 # number of radial views
# angl = (math.sqrt(5)-1)/2 * PI # golden angle in radians
angl = 2*PI/views # used by Steve Pickup
# Image data
xres = 96 # actual number of points collected
yres = 96
slices = 16
bvalues = 5
b_array = np.array([10,535,1070,1479,2141]) # b-values
```

Each cell should then be run consecutively.

