

# Visual Display Interface (VDI): A MATLAB Software Library For Simulating and Processing In-Vivo Magnetic Resonance Spectroscopy and Spectroscopic Imaging Data

Yiling Liu<sup>1</sup>, Georg Oeltzschner<sup>2</sup>, Itamar Ronen<sup>3</sup>, Rita Schmidt<sup>4</sup>, Amir Seginer<sup>4</sup>, Ivan Kirov<sup>5</sup>, Dan Wu<sup>6</sup>, Zhiyong Zhang<sup>1</sup>, Assaf Tal<sup>7</sup>

<sup>1</sup>Department of Biomedical Engineering, Shanghai Jiao Tong University, Shanghai, China

<sup>2</sup>Russel H. Morgan Department of Radiology and Radiological Science, The Johns Hopkins University School of Medicine, Baltimore, MD, USA

<sup>3</sup>Clinical Imaging Sciences Center, Brighton and Sussex Medical School, Brighton, UK

<sup>4</sup>Department of Neurobiology, Weizmann Institute of Science, Rehovot, Israel

<sup>5</sup>Department of Radiology, New York University School of Medicine, NY, USA

<sup>6</sup>Department of Biomedical Engineering, Zhejiang University, Hangzhou, China

<sup>7</sup>Department of Chemical and Biological Physics, Weizmann Institute of Science, Rehovot, Israel

**Running Head:** Visual Display Interface (VDI)

**Corresponding Author:**

Assaf Tal, PhD

Department of Chemical and Biological Physics

Weizmann Institute of Science

234 Herzl St.

Rehovot 7610001, Israel

Telephone: 972-(0)8-934-4951

FAX: 972-(0)8-934-4123

E-mail: [assaf.tal@weizmann.ac.il](mailto:assaf.tal@weizmann.ac.il)

**Keywords:** MRS, Magnetic Resonance Spectroscopy, Spin Simulations, Spectral fitting, linear regression, Spectroscopic Imaging

**Sponsors, Grants & Funding:** Assaf Tal acknowledges the support of the Monroy-Marks Career Development Fund and the historic generosity of the Harold Perlman Family. Assaf Tal acknowledges the support of the Israel Science Foundation (Personal Grant 416/20). Yiling Liu and Zhiyong Zhang acknowledge the support of the National Natural Science Foundation of China (No. 62001290)

# Abstract

We describe a new comprehensive and open-source MATLAB library, called the Visual Display Interface (VDI), designed to process, model, quantify and visualize magnetic resonance spectroscopy (MRS) and spectroscopic imaging (MRSI) data. The library focuses on three major strengths: promoting reproducible research by creating comprehensive data logging and reporting and identifying outlier datasets; seamlessly combining spectral and spatial processing of both single and multivoxel data; and offering a modern, object-oriented design. VDI handles a wide range of common tasks, including spectral and spatial transforms, built-in spectral fitting, absolute quantification, and running density-matrix simulations for coupled and uncoupled spin systems to generate appropriate basis sets and aid in sequence design. VDI interfaces with the Statistical Parametric Mapping (SPM) toolbox to carry out tissue segmentation, calculate tissue fractions within voxels, derive mean metabolite values from regions of interest defined by anatomical or functional atlases, and perform linear regression for global white matter and gray matter metabolite concentrations. The library's workings are demonstrated for two tasks: (1) Pre-processing, fitting and analysis of single-voxel proton MRS data from healthy volunteers; and (2) Extracting region-specific metabolite concentrations from spectroscopic imaging data based on an existing cortical atlas in MNI space, and calculating average gray and white matter global concentrations using linear regression.

# Introduction

The analysis of Magnetic Resonance Spectroscopy (MRS) and spectroscopic imaging (MRSI) data involves the manipulation of spatial, temporal and spectral degrees of freedom. Such analyses might include registering averages, carrying out phase corrections, fitting spectra, quantifying metabolite concentrations, calculating the mean value of a metabolite inside an anatomical region of interest, calculating fractions of gray matter and white matter tissue within a spectroscopic voxel, producing metabolic maps that are properly aligned with high resolution structural images. These require unique and highly-specialized computational tools. Over the past decades, several packages have been proposed for handling some of these aspects, including FID-A<sup>1</sup>, FiTAID<sup>2</sup>, FSL-MRS<sup>3</sup>, Gannet<sup>4</sup>, INSPECTOR<sup>5</sup>, jMRUI<sup>6</sup>, LCMoel<sup>7,8</sup>, MRSCloud<sup>9</sup>, Osprey<sup>10</sup>, SPANT<sup>11</sup>, TARQUIN<sup>12</sup> and VESPA<sup>13</sup>. Some of these freely available packages can handle spectral fitting and quantification, others preprocessing. A few are equally geared towards single and multivoxel data. Some, but not all, provide spin simulation capabilities.

Here, we describe a new set of libraries, written in MATLAB (The Mathworks, Natick, MA, USA) and compatible with releases 2018b onward, called the Visual Display Interface (VDI). VDI was specifically written to enhance the reproducibility of MRS research by focusing on three key quality-of-life areas: Provide users a modern, object-oriented interface for handling MRS data which promotes writing self-documenting code; Seamlessly combine the handling of multiresolution MRS and MRSI data, by providing a single coherent framework to oversee all spectral and spatial data processing; And provide in-depth HTML reporting and analysis report-generating capabilities for detecting outliers and summarizing data statistics which is specifically suited for large datasets with multiple subjects. More specifically, VDI is:

- (1) Freely and widely accessible: VDI is open-source and freely available for download (via [www.vdisoftware.net](http://www.vdisoftware.net)), and is implemented in MATLAB, one of the most widely-used scientific computing packages in academia.
- (2) Extensively documented: The VDI libraries are extensively documented. They were written to anticipate a wide range of user inputs. An extensive set of examples is provided in a separate subdirectory titled “examples”.

- (3) Report-Generating: Preprocessing, fitting, and simulations all produce HTML reports which provide numerical and graphical feedback, as well as try to identify corrupted or outlier data. This is done both at the single-subject and group levels. All figures which appear herein were pasted directly from the generated reports, with no or minimal modifications.
- (4) Accessible to beginners: By adopting an object-oriented architecture, VDI provides users with clear, robust and accessible command-line tools that produce self-documenting and succinct code. Most parameters are automatically determined by the routines without the need for any user intervention.
- (5) Extendible by experts: The object-oriented structure of VDI provides a rich and elegant framework for producing highly complex scripts capable of handling diverse datasets.
- (6) Integrated with SPM: The Statistical Parametric Mapping package (<https://www.fil.ion.ucl.ac.uk/spm/>) offers powerful MATLAB-based image segmentation and processing capabilities. VDI can call several SPM routines directly to handle a variety of tasks, from segmenting T<sub>1</sub>-weighted images into gray matter, white matter and cerebrospinal fluid maps, to the calculation of metabolite statistics within atlas-based predefined regions of interest.
- (7) Capable of handling multimodal, multiresolution data: VDI has been written to handle multidimensional (2D and 3D) spectroscopic imaging data, and can operate efficiently on such data. Some of its capabilities include masking; calculating tissue fractions within voxels; tissue-type regression; and calculation of descriptive statistics within regions of interest. Users can provide their own masks in DICOM or NIfTI format, or employ SPM to generate such masks.
- (8) Interactive and visual: The library offers interactive tools for displaying and interacting with both spectral and spatial data dimensions in real-time. Furthermore, the library produces and summarizes several of its data structures in table format, and offers powerful tools for plotting and exploring such tabular data using a variety of plot types.

VDI is already in use by several groups and underlies the processing of data in multiple publications over the past several years<sup>14–20</sup>. To facilitate its widespread dissemination, the current manuscript describes the general philosophy behind the framework, and its main classes and the way they interact among themselves. Two examples of usage of the library are presented in the Methods section, and highlight its versatility: In the first, SemiLASER single-voxel MRS data are loaded, preprocessed, fitted, quantified (in terms of absolute concentrations) and visualized. In the second, global linear regression is carried out on a 32×32 2D MRSI supra-ventricular slice from a healthy volunteer to obtain average white matter and gray matter tissue metabolite concentrations<sup>21–23</sup>.

## Theory

### Data Structures for Single and Multivoxel Data

VDI employs an object oriented approach to handling spatial data (Fig. 1), where data and methods are grouped into classes. Inheritance, in the object-oriented sense, is implemented wherever possible – that is, some classes retain and extend the capabilities of others – and indicated by black arrows. At its root is the class `VDIVolume`, which describes a general 3D oriented volume in space, with position, size and orientation metadata. These properties are given relative to some common reference frame, which for most applications coincides with that of the magnet's frame of reference, with its center at the center of the gradient system. `VDIVolumeMatrix` inherits from `VDIVolume` and enriches it by adding the ability to segment `VDIVolume` into voxels. However, `VDIVolumeMatrix` is rarely used itself. Rather, its children, which can also associate data with the voxels, are much more useful: `VDIImage`, which is used to store 3D data, such as structural MRI images or metabolite maps; and `VDIImageND`, which is used to store n-dimensional data (in the data property), such as spectroscopic imaging or BOLD-fMRI data. Thus, `VDIImageND` is the most general class for storing and manipulating spectral-spatial-temporal data, such as functional MRS<sup>16,17,24–26</sup>, hyperpolarized <sup>13</sup>C-MRS<sup>27,28</sup>, deuterium metabolic imaging data<sup>29–31</sup>, spectroscopic fingerprinting<sup>18,32,33</sup>. However, `VDIImage` offers greater flexibility, a smaller footprint and specialized methods specific to 3D imaging data. A third important class that derives from `VDIVolume` is `VDIShape`, which

is used to describe three-dimensional boxes and ellipsoids, which can be used, e.g., to define regions of interest for processing, or the position of a voxel.

The use of object oriented design and operator overloading (namely, the redefinition of default operators such as addition or element-wise multiplication) allows for the seamless manipulation of spatial-spectral datasets, even with different orientations, positions and spatial resolutions. Consider, for example, the following code, which zeros out voxels in a low-resolution VDIImage metabolic map of n-acetyl-aspartate (NAA) outside the posterior cingulate gyrus (PCG), using a high resolution binary mask:

```
imgNAA = VDIIIO.LoadNII('NAA.nii');
mask = VDIIIO.LoadNII('maskPCG.nii');
imgNAA = imgNAA.*mask;
```

This code uses VDIIIO to load the NAA metabolic map and PCG mask – both of which are VDIImage objects – and uses the fact that the .\* operator is overloaded for the VDIImage class. It interpolates the high resolution mask of the PCG onto the spatial grid of the NAA image before carrying out the multiplication. All interpolations within VDI are nearest-neighbors: the value of each voxel in the interpolated image equals that of the voxel in the high-resolution image closest to it (center-to-center). VDI classes have overloaded several common MATLAB operators, including element-wise multiplication, addition, subtraction and element-wise division.

Spectroscopic data, whether single or multivoxel, are stored as VDIImageND objects. These objects offer extensive capabilities for preprocessing, visualizing and manipulating such data (as shown in the Methods section below). VDIImageND objects are initialized with an n-dimensional array of data and appropriate labeling of each dimension type, space, unit and axis. For example, 6-dimensional spatial-spectral array data from multiple coils with dimensions (x, y, z, time, averages, coil) can be initialized as follows:

```
metab = VDIImageND(data, 'dimType', ["x", "y", "z", "spec", "avg", "coil"], ...
    'dimSpace', ["space", "space", "space", "time", "time", "undefined"], ...
    'dimUnit', ["mm", "mm", "mm", "sec", "sec", ""], ...
    'axis', {xAxis, yAxis, zAxis, timeAxis, avgAxis, 1:32}, ...
    'B0', 3);
```

xAxis, yAxis, zAxis and timeAxis are all vectors (defined elsewhere), indicating the relevant axes for the data along each of its dimensions (e.g., yAxis is in millimeters, and has the same number of elements as data along its second dimension).

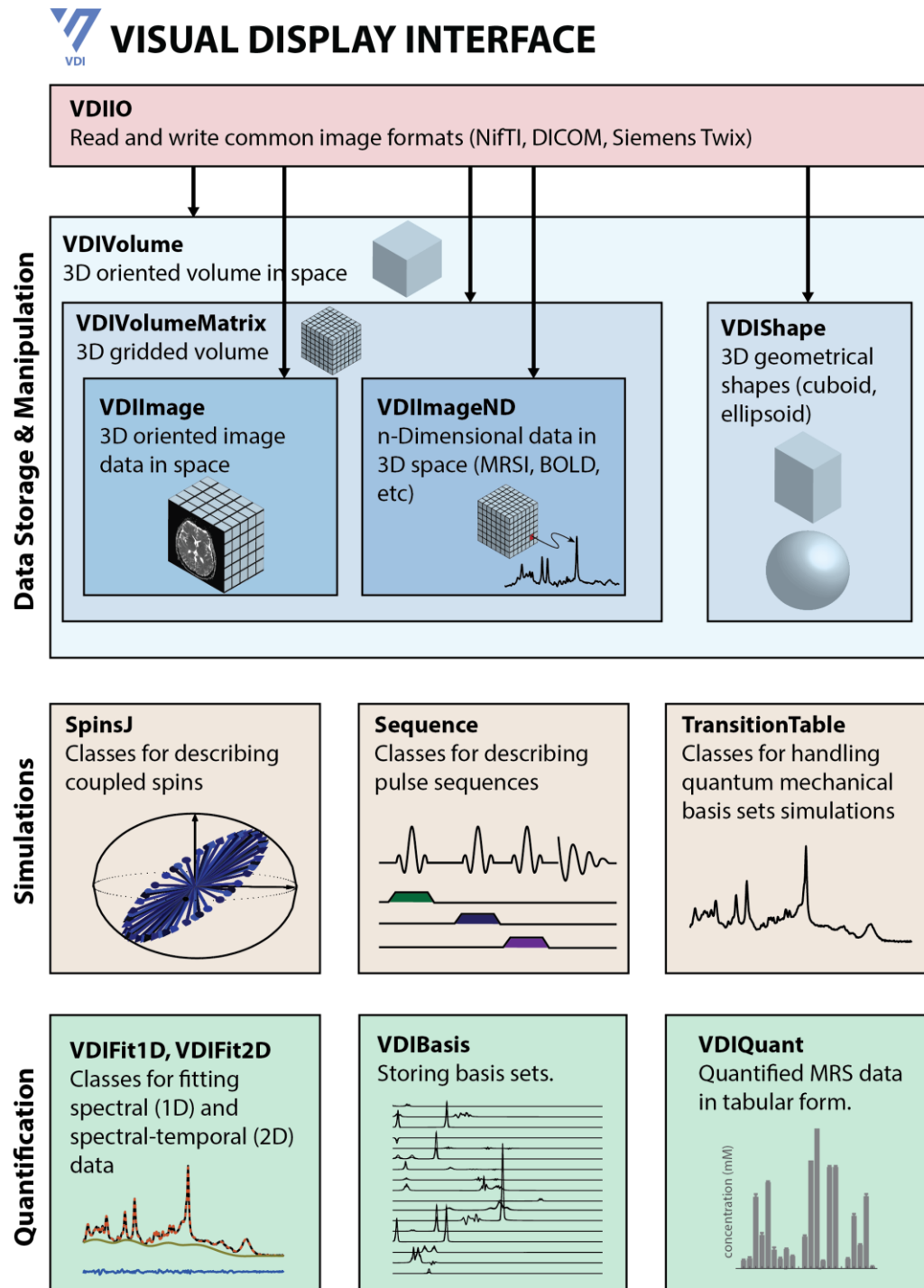


Figure 1. The major classes in the VDI framework, and their various relationships. Arrows indicate class inheritance, in the object-oriented sense. The main classes for storing and manipulating MRS and MRSI data are shown in the top (blue) row. Classes for carrying out quantum mechanical spin simulations are shown in the second row (yellow). The final row details major classes involved in fitting and modeling spectral data.



## Preprocessing of Data

Preprocessing refers to several standard steps in which spectral and spatial data is modified prior to spectral fitting. Methods for carrying out common preprocessing steps are implemented for VDIImageND objects. For example, coil addition, spectral registration, and automated phasing can all be carried out with a few commands:

```
metab.AddCoils;
metab.RegisterFIDs;
metab.Average;
metab.FT;
metab.Phase('order', "zerofirst");
```

These methods obtain most required metadata from the VDIImageND object itself and therefore require little additional input, but can also be tailored with additional pairs of variable names and values. For example, phasing only corrects zero order distortions. However, in the above example, both zero and first order phasing are requested by setting the input parameter order in the Phase method to zerofirst. Some of the more commonly used preprocessing methods implemented for VDIImageND are described in Table 1, alongside the algorithms they implement.

Action	Method	Description
Coil Combination	AddCoils	Singular-Value decomposition coil addition <sup>34,35</sup> , modeling the data as a linear combination of signals $s_j$ which differ up to a complex constant $a_j$ : $s(t) = \sum_j s_j a_j$ . By default, addition coefficients are proportional to the signal intensity. If a reference noise dataset is provided, coefficients become proportional to the SNR.
Spectral Registration	RegisterFIDs	An spectral-alignment and zero-order phasing algorithm following Mikkelsen et. al. <sup>36</sup> , in which FIDs are sorted according to similarity and then iteratively weighted according to similarity, phased and aligned.
Averaging	Average	Allows for simple or complex averaging patterns. For example: Average('dim', "set", 'pattern', [1 2], 'coeffs', [1 -1]) subtracts consecutive averages, as done in MEGAPRESS. More complex phase cycling reconstruction schemes can easily be implemented this way, by setting coeffs accordingly.
Apodization	Apodize	Gaussian (default) or Lorentzian apodization. Can be equally applied along spectral and spatial dimensions.
Zero-Filling	ZF	Zero-filling, either one-sided (default) or symmetrical.
Automatic Phasing	Phase	Minimum-entropy based automatic phasing for either zero and/or first order phase distortions <sup>37</sup> .
Manual Phasing	N/A	Manual zero-order phasing can be applied by using the overloaded .* operator. For example, typing: myData.*exp(1i*pi)
Fourier Transform	FT	Fourier Transforms are handled automatically based on the dimension type (spectroscopic, spatial), and space (freq, time for the spectral dimension; space or kspace for spatial dimensions).

**Table 1.** Several common VDIImageND methods used to preprocess data.



A second powerful set of methods allows the user to extract quality assurance metrics for their spectroscopic data, such as linewidth, SNR, and FWHM. Default behaviors, which can be modified via optional parameter inputs.

Method	Description
EstimateFWHM	Estimate water full-width at half maximum (can also be used for any single, well-resolved peak).
EstimateLipidContamination	Estimate spectral lipid contamination, defined as the ratio of the maximal intensity of the signal in the [0.3, 1.7] ppm range to the maximum of the NAA signal (in the [1.85, 2.15] ppm range).
EstimateSNR	Estimate the signal-to-noise ratio of the NAA peak to the real part of the noise. The noise is either estimated from a reference noise dataset, or the [7, 9] ppm range in the spectral dataset. Noise is linearly detrended by default.
DetectOutlierSpectra	Detects outlier spectra <i>across subjects</i> , defined via their $L_2$ distance from the median spectrum. Only works on the arrays of VDIImageND objects. By default, this method operates voxel-by-voxel; normalizes all spectra by their individual $L_2$ -norms prior to commencing outlier detection; takes the absolute value of the spectrum; operates on the [0.2, 4.2] ppm range. Outliers are defined as spectra with $L_2$ -norms more than three scaled absolute median deviations from the median.

**Table 2.** VDIImageND methods for quality assurance of MRS and MRSI data.

Almost all VDIImageND methods offer the ability to produce detailed HTML reports of their actions, by accepting a VDILog object, as demonstrated by the following snippet:

```
metabLog = VDILog; % Creates 'log' subdirectory for scripts
metabLog.Delete; % Remove previous log entries (if desired)

metabProc.RegisterFIDs('log', metabLog);
metabProc.FT("spec", metabLog);
metabProc.Phase('log', metabLog);

% Append lipid contamination quality assurance metrics to the log file.
metabProc.EstimateLipidContamination('log', metabLog);
```

## Spin Simulations

VDI is provided with an independent sub-library called SpinTool, hosted on a separate path, which contains code for simulating the behavior of both uncoupled and coupled spin systems (SpinsJ class) in response to pulse sequences (Sequence class). It does so by solving the full quantum-mechanical Liouville equations of motion, without relaxation. Spins are initialized in a straightforward manner. For example, creating a spin ensemble with NAA, creatine (Cr) and phosphocreatine (PCr) can be done with two lines:

```
spins = SpinsJ('B0', 3);
spins.AddMetab({'NAA', 'Cr', 'PCr'}, 'position', posMatrix);
```

posMatrix is a N×3 matrix, with each row corresponding to an [x,y,z] position at which to place a metabolite instance, allowing for arbitrary 3D distributions (e.g., posMatrix=[0,0,0] will place a single copy of each metabolite at the center of the magnet system).

Pulse sequences can be defined easily using a user-friendly syntax. For example, a spin-echo sequence using idealized hard pulses can be coded within a single line, and applied to the previously-defined spin structure with an additional line of code. A matrix containing the time-domain simulated basis function for each metabolite is created in a third line:

```
TE=40;
seq = Sequence({'hard', 90, 270}, {'delay', TE/2}, {'hard', 90, 0}, ...
               {'delay', TE/2}, {'acquire'}));
[spins, TT] = seq.Apply(spins);
FIDMatrix = TT.GetSpec('domain', 'time', 'FWHM', 0.001, ...
                       'lineShape', 'Lorentzian', 'numAcqPts', 4096, 'SW', 2);
```

More complex sequences, containing arbitrary pulse waveforms and phase cycling can be created just as easily. This sequence can be easily visualized using the Plot method in the Sequence class (i.e. by typing seq.Plot). The Apply method returns the spins following the sequence. Acquired spectral data TT is stored in the TransitionTable class; each TransitionTable object described the spectral pattern associated with a particular metabolite, in terms of the frequencies, amplitudes and phases associated with each resonance. These can be convolved with appropriate lineshape functions – e.g. Lorentzians or Gaussians – to generate either free induction decays in the time domain, or spectra in the frequency domain. This is done using the GetSpec method, which accepts as inputs multiple parameters, such as the full-width-at-half-maximum (FWHM) and spectral width (SW) for shaping the spectral acquisition parameters.

## Fitting, Spectral Modeling and Absolute Quantification

The Sequence and SpinsJ classes offer the necessary machinery for generating basis sets, which are essential for spectral fitting. Basis sets are handled via the VDIbasis class. Once a TransitionTable object has been obtained via simulations, it can be

converted into a basis set, and exported to a format consistent with the popular LCModel fitting package<sup>7,8</sup>, within a few lines of code:

```
myBasis = VDIbasis(TT, 'B0', 3, 'numAcqPts', 4096, 'dwellTime', 1e-3);
myBasis.ExportBasisToLCModel('fit.basis', 'TE', 40);
```

LCModel has recently been released to the public domain (<https://github.com/schorschinho/lcmodel>) and offers a free, robust, widely validated fitting package; its precompiled binaries need only be added to the MATLAB path. Spectroscopic data in VDIImageND objects – whether single or multivoxel – can then be directly fit using the FitLCModel method.

VDI contains two additional classes, distinct from LCModel, which implement least squares fitting of spectroscopic data, appropriately named VDIFit1D and VDIFit2D. The first can be directly used as a replacement for LCModel for fitting one-dimensional spectroscopic data. VDIFit2D has been specifically created to fit spectral-temporal two-dimensional datasets, where temporal correlations exist between successive transients. These approaches have been suggested (and subsequently also implemented in FSL-MRS<sup>38</sup>) to substantially improve fitting precision<sup>39</sup> by providing additional temporal models (i.e. constraints) for the time evolution of metabolite amplitudes, frequencies, phases or lineshape parameters<sup>2,3</sup>.

All spectral fitting routines generate tabular data which is stored in objects belonging to the VDIQuant class. The metabTable property of a VDIQuant object contains a table in which each metabolite from each region and volunteer occupies an independent line. Table columns describe sequence and tissue related parameters, such as tissue fractions, TR, TE, FA, the measured signal, and tissue-type dependent relaxation constants. Default parameter values can be added using the SetDefaultValues methods. Absolute quantification<sup>40–43</sup> is carried out via the AbsQuant method, using water as a reference and handling tissue heterogeneity and relaxation corrections.

## Auxiliary Classes and Visualization Tools

VDI offers several auxiliary classes, designed to facilitate loading, processing and display of data. Examples below will illustrate the use of these classes in real-life applications; here, we highlight the major ones.

VDIIO is used to read and write common data formats, including NifTI (.nii), DICOM, Bruker Paravision 5 and 360, and Siemens propriety (.rda) formats. It consists mostly of static methods – that is, methods which do not require a specific instantiation of a VDIIO object. Some of the most useful ones are LoadNII and WriteNII (reading and writing NifTI files into/from VDIImage objects), LoadDICOMImageSeries (reading a DICOM image series) and LoadTwix (reading Siemens raw data as VDIImageND objects).

VDIPlot contains multiple routines for efficiently partitioning tabular data and producing bar, line and correlation plots, such as PlotTableBars, Plot. This can be used to plot the results of absolute quantification. For example, PlotTableBars(myTable, 'conc', {'metabName', 'ROI'}) partitions the data in the MATLAB table myTable according to the different values in the variable columns metabName and ROI (region-of-interest), calculates the mean and standard-error-of-the-mean of the data stored in the variable conc within each such sub-partition, and produces a bar plot of these quantities.

The ability to interactively view, modify and process MRI and MRSI data is an essential component of scientific discovery. VDI includes a class, VDIImageBrowser, specifically designed for the visualization of data in VDIImage, VDIShape and VDIImageND objects. VDIImageBrowser allows for exploring n-dimensional data by reslicing it (using real-time nearest-neighbors interpolation) and overlaying different datasets, binary masks, and shapes in a common space.

## Methods

### Preprocessing and Fitting SemiLASER Data

Our first example script loads a collection of Siemens raw data files (.dat) containing SemiLASER data from the dorsal anterior cingulate of ten healthy volunteers (TR/TE=7000/42 ms), and performing coil combination, trimming of pre-echo points in the FID, registration of each transient via alignment and phasing, averaging, Fourier transformation and automated phasing. The script then creates a basis set by an idealized SemiLASER sequence with idealized hard pulses, and proceeds to fit it using the LCModel package. Default fitting parameters are used, except the dkntmn parameter which controls the baseline tortuosity (almost all LCModel control file parameter can be

passed to the FitLCModel method). Finally, it carries out absolute quantification. Data were acquired on a 7T Siemens Terra MRI scanner, using the default single-channel transmit/24 channel receive Nova coil; the data were been previously published as part of a comparative study between edited and unedited quantification at ultrahigh fields<sup>44</sup>.

```
% Load data from three subjects
[metab, water, noise] = VDIIIO.LoadTwix(dir('Example Data\*.dat'), ...
    'isICEChop', true, 'name', "%filename%", 'TRDimType', "set", ...
    'isVerbose', true);

% Create and initialize logging object; delete all previously created logs.
myLog = VDILog;
myLog.Delete;

% Preprocessing: Coil addition, spectral registration, FT, phasing
metabProc = metab.AddCoils('refData', water, 'log', myLog);
waterProc = water.AddCoils;
metabProc.ChopPts('direction', "post", 'type', 'numRemainingPoints', 'numPts', 4096);
waterProc.ChopPts('direction', "post", 'type', 'numRemainingPoints', 'numPts', 4096);
metabProc.RegisterFIDs('log', myLog);
metabProc.Average('dim', "time");
waterProc.Average('dim', "time");
metabProc.FT;
waterProc.FT;
metabProc.Phase;
waterProc.Phase;

% Append data quality assurance metrics to the log file.
metabProc.EstimateLipidContamination('log', metabLog);
metabProc.EstimateSNR('log', metabLog, 'noiseLims', [11 15]);
waterProc.EstimateFWHM('log', metabLog);

% Simulating and exporting a basis set
spins = SpinsJ('B0', metabProc(1).B0);
metNames = {'Glu', 'NAA', 'NAAG', 'PCh', 'PCr', 'Cr', 'GPC', 'Gln', 'Ch', ...
    'mI', 'GABA', 'Ala', 'GSH', 'Lac', 'Tau', 'Glc', 'Asp'};
spins.AddMetab(metNames);
TE = 42;
seq = Sequence.GetIdealSequence('SemiLASER', 'TE', TE);
[~, TT] = seq.Apply(spins); % Apply the sequence
myBasis = VDIBasis(TT, 'B0', metabProc(1).B0, 'numAcqPts', metabProc(1).numSpecPts,
    'dwellTime', metabProc(1).dwellTime);
myBasis.ExportBasisToLCModel("fit.basis", 'TE', TE);

% Fitting and absolute quantification
quant = metabProc.FitLCModel("fit.basis", 'waterData', waterProc, ...
    'log', metabLog, 'isVerbose', true, 'dkntmn', 5);
quant.SetDefaultValues('dACCDefaults.json');
quant.AbsQuant('log', metabLog);
```

The static method LoadTwix in the VDIIIO class is used to read all raw Siemens data (\*.dat) in the Example Data directory. Data was acquired using a widely available SemiLASER implementation by the CMRR spectroscopy group (<https://www.cmrr.umn.edu/spectro/>); Setting the isICEChop option to true removes the pre-echo points acquired by the sequences (as per information stored in the ICE Siemens header), and the name variable tells the function how to populate the name property of each loaded dataset. Almost all methods accept an isVerbose parameter, providing real time processing feedback to the command line as the methods are executed. LoadTwix returns arrays of VDIImageND objects containing metabolite, reference water and noise data from each of the ten volunteers scanned: metab(k) is the VDIImageND object corresponding to the metabolite data from the k<sup>th</sup> volunteer. The subsequent lines perform preprocessing of both the water and metabolite data. Nearly all methods accept a VDILog object via the log input parameter, and output some form of feedback to the HTML report. The VDILog object is initialized before preprocessing takes place, and deletes the previous report file.

Basis simulation follows using the SpinsJ and Sequence classes. A coupled spin structure is defined and populated with fifteen metabolites, starting out at thermal equilibrium. An idealized SemiLASER sequence using hard pulses is initialized, and applied to the spins. The resulting TransitionTable object, TT, is fed into a VDIIBasis object with the appropriate spectral parameters, which is used to create a basis file in a format readable by LCModel.

Once a basis set is generated, fitting can take place using the publicly-available LCModel package via the FitLCModel method. This method only requires the name of an appropriate basis file; it obtains all required metadata from the basis file or from the VDIImageND object which calls it. It operates sequentially on all loaded single-voxel datasets in the 1×10 array metabProc. Additional LCModel parameters can be passed via additional input switches to FitLCModel; here, the baseline is constrained to remain linear by setting the LCModel dkntmn parameter to five. FitLCModel not only saves all processed control and output files generated by LCModel, but also reads them and returns a VDIQuant object, quant, which is subsequently used for absolute

quantification<sup>42,43,45–47</sup>. quant is populated with metadata from the VDIImageND objects; however, a few parameters which are not stored in these objects – such as the region of interest, tissue fractions, transmit inhomogeneity and tissue relaxation values – default to values stored in the file dACCDefaults.json. JavaScript Object Notation (JSON) is an open standard file format, widely adopted by is which is a short text file containing default values for parameters not found elsewhere:

```
{
  "ROI": "dACC",
  "frac": [1.0, 0.0, 0.0],
  "compNames": ["GM", "WM", "CSF"],
  "relaxationTable": "Relaxation Data.json",
  "TR": "inf",
  "B0": 7,
  "nameRef": "water",
  "seq": "SemiLASER",
  "FA": 90,
  "B1": 1.0,
  "V": 1.0
}
```

Most importantly, this file points towards another a second JSON file, Relaxation Data.json, containing relaxation data for metabolites and reference water from multiple ROIs, summarized from the literature. Because of the paucity of relaxometry studies at the dACC at 7T, the  $T_1$  and  $T_2$  values used herein originate from other regions with similar GM and WM composition.

## Global Metabolite Concentrations Using Linear Regression

Global linear regression extracts tissue-specific metabolite concentrations over the entire MRSI VOI. Such an approach is superior to ROI-based approaches for assessing diffuse disease activity<sup>48</sup>, because it circumvents two major problems with MRSI: The poor SNR in each voxel, and the partial volume due to a mixture of both GM, WM and CSF tissue in each voxel. Briefly, linear regression models the signal in each voxel as a linear combination arising from each of the voxels, assuming a global, single concentration for each tissue type (Fig. 2):  $S_k = f_k^{GM} S^{GM} + f_k^{WM} S^{WM} + f_k^{CSF} S^{CSF}$ , where  $k$  is the voxel index, and  $f_k^{WM}, f_k^{GM}, f_k^{CSF}$  are the tissue fractions ( $f_k^{WM} + f_k^{GM} + f_k^{CSF} = 1$ ). Since no detectable metabolite signals are observed in the CSF,  $S^{CSF} = 0$ . This approach is discussed in depth in the literature<sup>21</sup>, and has been used extensively to study healthy



physiology<sup>49–51</sup>, diffuse axonal injury in mild traumatic brain injury<sup>23,48,52–57</sup>, and normal appearing white matter in multiple sclerosis<sup>22,58</sup>. Recently, linear regression was also adapted to sodium MRI<sup>59</sup>.

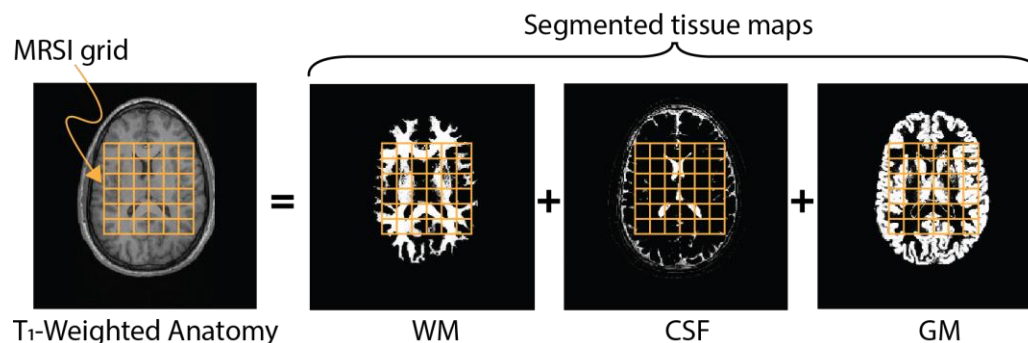


Figure 2. In linear tissue regression, the MRSI metabolite's signal in each voxel is modeled as a linear combination of the contributions of each tissue type (GM, WM, CSF), proportional to the amount of each tissue in each voxel.

The application of linear regression to MRSI data becomes straightforward with VDI, which interfaces with SPM to generate tissue maps from high resolution T<sub>1</sub>-weighted anatomical images. The required VDI code is:

```
% Load anatomical MPRAGE data, and use SPM to segment.
imgMPR = VDIIIO.LoadNII('ExampleData\MPRAGEAnatomy.nii');
[imgCSF, imgWM, imgGM] = imgMPR.SPMSegment;
% Load raw spectroscopic imaging data, fit, and run absolute quantification
quant = imgSpec.FitLCModel('fit.basis', 'isVerbose', true);
quant.SetDefaultValues('AverageBrainDefaults.txt');
quant.AbsQuant;
% Extract NAA metabolic map and run linear tissue regression
imgNAA = quant.GetVDIImage('groupVarNames', ["name"], ...
    'constraints', "name==tNAA", 'ref', imgSpec);
metabLog = VDILog;
[stats, imgRecon] = imgNAA.TissueLinearRegression(imgWM, imgGM, 'log', metabLog);
```

The first two lines load an anatomical T<sub>1</sub>-weighted MPRAGE image and uses SPM12 (added to the MATLAB path) to segment it into three VDIImage high resolution tissue maps. A previously loaded VDIImageND object, imgSpec, containing MRSI data from a 2D slice with a 32×32 in-plane matrix is fit on a voxel-by-voxel basis using LCModel; this step is time consuming, since each voxel takes about 10 seconds to fit, requiring a total processing time of approximately 32×32×10 seconds≈3-4 hours. As with the single-voxel example, it returns a VDIQuant object (quant), which contains fitted metabolite signals from each and every voxel. This undergoes fitting using an analogous pipeline to the single-voxel case. The GetVDIImage method of the VDIQuant class is then used to create

imgNAA, a metabolic map of total NAA of class VDIImage, containing the concentration of NAA, and using the spatial orientation and positioning of the original imgSpec. Finally, linear tissue regression is run via the final line, and its results are both returned as variables and appended to an existing log file.

## Metabolite Concentrations from Atlas-Defined ROIs

When working with an anatomical binary mask (a VDIImage object) and an MRSI dataset (a VDIImageND object), VDI offers a convenient method called CalcStats to calculate metabolite averages, medians, and other statistics. Automation is possible when an anatomical high-resolution T<sub>1</sub>-weighted image is available. In such cases, the image can be registered to a common space like MNI, and the inverse transformation can be applied alongside a predefined atlas to generate anatomical masks in the subject's original space. VDI simplifies the handling of these transformations through its SPMGetAtlasStructure method, which automates the relevant SPM routines. Atlases are comprised of two files: a NifTI image containing imaging data in a common space (e.g., MNI), and an XML file that encodes various atlas labels, such as those found in the Automated Anatomical Labeling (AAL) atlas<sup>60</sup>. To use these atlases with VDI, the corresponding files should be placed in the VDI atlases subdirectory. This enables VDI to not only detect their presence but also provide auto-completion for all available label names.

```
% Retrieve parietal lobe mask
maskParietal = imgMPR.SPMGetAtlasStructure("MNI", "Parietal Lobe");

% Calculate statistics over parietal lobe
maxNAA = imgNAA.CalcStats("mask", maskParietal, "log", metabLog, "stat", "max",
"anat", imgMPR);
```

## Results

### Preprocessing and Fitting SemiLASER Data

Fig. 3 shows the output of several of the preprocessing steps, as well as fitting and quantification. All plots were taken directly from the generated log file. The spectral registration (Fig. 3A) plots the mean spectrum, taken over all ten volunteers (dark blue), and the spectra from each individual volunteer (light blue), before and after spectral

registration. The relative stability of the magnet is visually reflected by the similarity between the spectra before and after registration. A more quantitative view can be had by looking at the zero order phases and spectral shifts of each FID, shown in the next two plots, for all subjects. As expected, greater shifts are required for later FIDs, as the magnet drifts slightly over time. Automatic phasing is shown in Fig. 3B for all subjects before and after phasing, as well as for a sample subject. A histogram showing the distribution of zero order phases is provided, showing the mean zero-order phase was  $36^\circ$ , with some inter-subject variability (standard deviation:  $16^\circ$ ). The third line (Fig. 3C) shows quality assurance plots: lipid contamination, water FWHM (in ppm) and SNR taken over all volunteers, as well as an image showing the spectroscopic signal from the region designated as noise. This image provides visual reassurance that the region does indeed contain normally distributed noise; a small, linear bias can be seen, likely due to the slight imperfections in the phasing of the residual water baseline. Finally, the mean spectrum, its standard deviation, the mean fit and standard deviation, residual and baseline, over all volunteers, are shown in Fig. 3D, alongside a bar plot showing the quantified concentrations from the voxel.

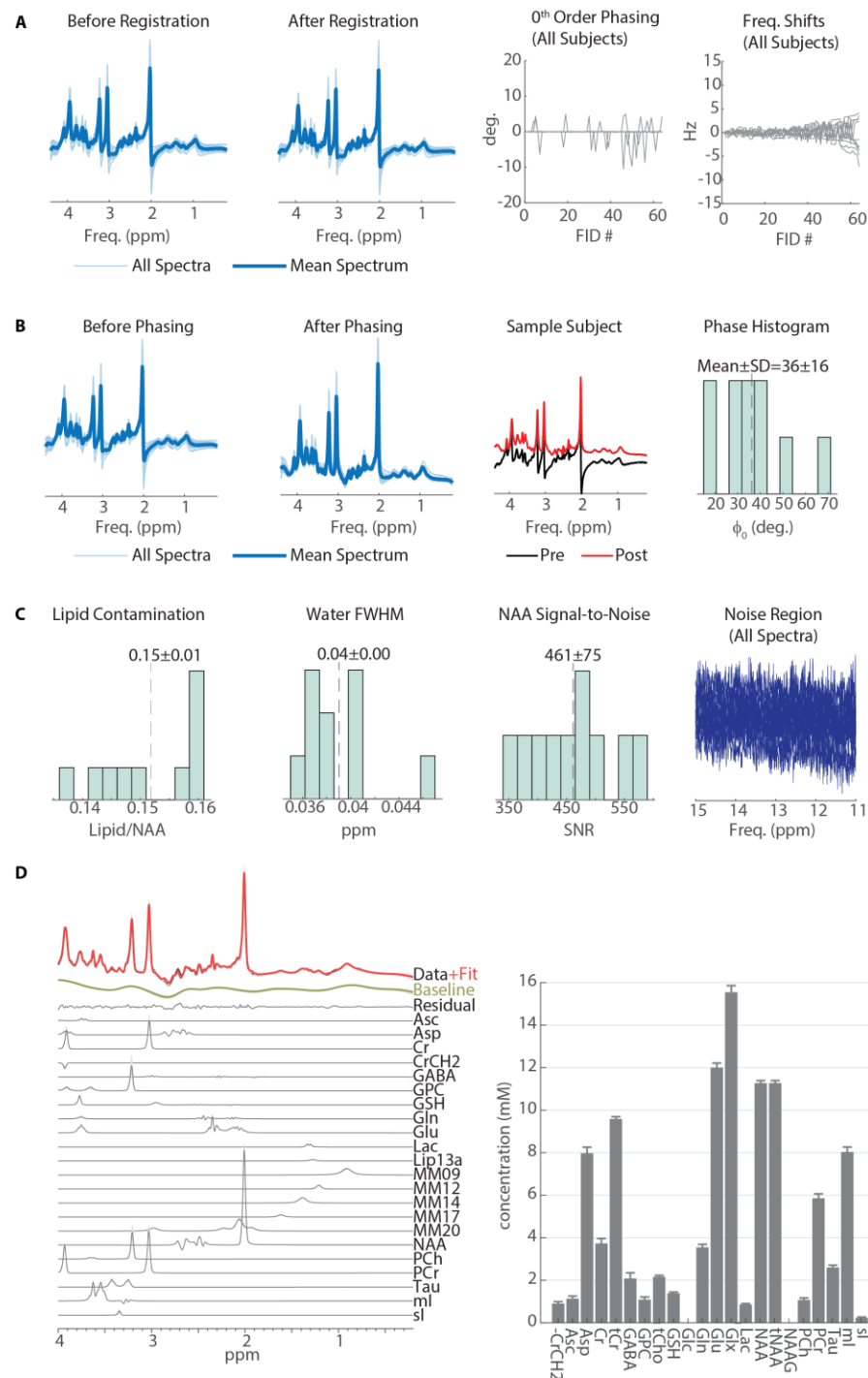


Figure 3. Spectral preprocessing and fitting of SemiLASER data acquired at 7T from 10 volunteers. All figures were taken with minor modifications from the logging files produced by the MATLAB scripts using VDI (via the VDIlog library). (A) Spectral registration. (B) Zero order automated minimum entropy phasing. (C) Quality assurance metrics, including lipid contamination (relative to NAA), water full-width-at-half-maximum, NAA SNR, and a visualization of the region used to estimate noise. (D) Results of spectral fitting and absolute quantification.

## Tissue and Region Specific Metabolite Concentrations

Fig. 4 showcases the results of the linear regression for obtaining mean tissue-type concentrations. As with the single-voxel analysis, all plots and details shown are taken directly from the output log file. A total of 248 voxels out of the full  $32 \times 32 = 1024$  in-plane 2D matrix were used; these were chosen as those voxels for which the combined WM and GM fractions exceeded 80%. The mean tissue concentrations were 11.50 mM (95% confidence interval: [11.08, 11.93] mM) for GM, and 10.54 mM (95% confidence interval: [10.01, 11.06] mM) for WM, with  $R^2=0.156$  (Fig. 4A). The variance inflation factor of GM and WM fractions vectors was 0.35, indicating the two regressors do not exhibit substantial co-linearity. Histograms of GM, WM fractions for the 248 voxels are shown in Fig. 4B, and the signal vs. the WM fraction is plotted in Fig. 4C, indicating voxel tNAA concentrations decrease with increasing WM content, as expected from the literature<sup>21</sup>. Fig. 4D shows the residuals obtained by subtracting the fitted linear model values from the quantified concentrations in each voxel. Figs. 4E-G show the map of tNAA, the spatial mask of the voxels used in the regression analysis, and the linear reconstructed tNAA map. Finally, Fig. 4H shows the map of residuals, the values of which are shown in the histogram in Fig. 4D.

Fig. 5 shows the parietal lobe that was automatically generated from the MNI atlas overlaid on top of both the metabolic tNAA map and the reference MPRAGE image, as well as the histogram of tNAA concentrations from the mask voxels. A total of 47 voxels fall within the mask, with a mean  $\pm$  standard deviation concentration of  $8.83 \pm 2.02$  mM and a range of [2.31, 10.95] mM.

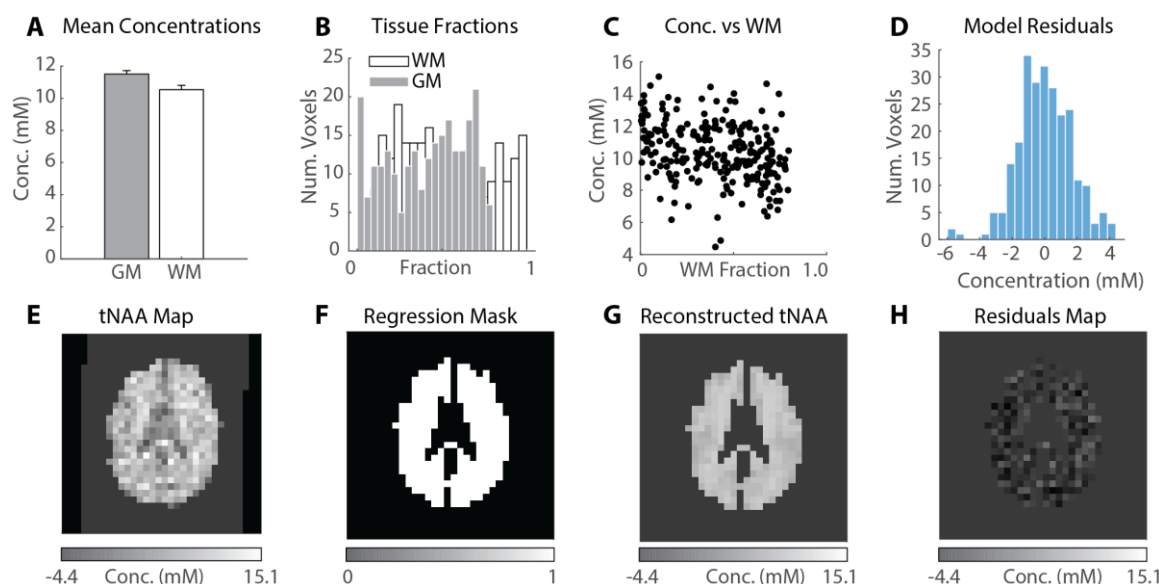


Figure 4. Linear tissue regression. (A) Mean±standard error of the global NAA concentration in each tissue type. (B) Histograms of gray and white matter fractions in the spectroscopic voxels used in the regression. (C) NAA concentrations as a function of white matter fractions (each point is a separate spectroscopic voxel). (D) Histogram of fitting residuals, i.e. the difference between data and fitted values in each voxel. Note the distribution is normal, as should be. (E) Metabolic map of NAA (stored in the `imgNAA` variable). (F) Binary mask over which regression is carried out, defined by the brain segmentation and the requirement  $WM+GM \geq 0.8$ . (G) Reconstructed NAA map obtained from linear regression. (H) Spatial map of residuals (same numbers used in (D)). All metabolic maps employ the same intensity scale for consistency, but only the residuals exhibit negative values.

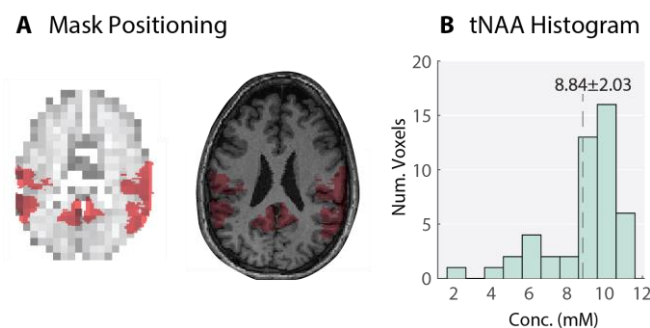


Figure 5. ROI tNAA concentrations from a predefined mask (parietal lobe). (A) Mask localization on both the metabolic map and anatomical MPRAGE image. (B) Histogram of tNAA concentrations from the mask.

## Discussion

The increasing complexity and size of neuroimaging datasets requires tools that are not merely efficient and accurate, but also easy to use, self-documenting and equipped with the ability to alert the user about potential pitfalls. For example, the recent focus on harmonization techniques of large-scale multisite MRS data is one such indication of this trend<sup>15,61,62</sup>. VDI was constructed to address such a need; It focuses on providing its users

with a modern, extensively-documented object-oriented framework that promotes writing self-documenting code and minimizes tedious tasks. Its methods were written to promote batch processing of data, as shown in the first example where several single-voxel datasets were processed simultaneously.

A second major focus of VDI is on its automated reporting capabilities. Succinct code often runs the risk of hiding assumptions, and complex processing pipelines could fail in surprising ways while deceptively outputting seemingly-meaningful numbers. Both challenges can be overcome by detailed reporting capabilities. VDI implements this philosophy by making it possible to generate HTML log files. These provide detailed reports of the parameters used, even those not directly input into each method, as well as visual feedback, quickly indicating whether a particular processing step has failed. All figures generated are saved by default both as bitmaps and vector graphics, making them eminently usable in scientific publications, as has been done herein (Figs. 3, 4). In addition, each processing step attempts to automatically identify failures and outliers and alert the users to their presence at the appropriate place in the report. For example, when phasing multiple datasets simultaneously, individual datasets which significantly differ in their zero order phase from the group median by more than three standard deviations are listed in red in the report.

Unlike many other MRS packages, VDI does not offer a graphical user interface, except when visualizing raw MRS data. While graphical interfaces appeal to newcomers, they are prone to manual user errors which are difficult to identify and correct. The underlying philosophy of VDI instead focuses on writing reproducible, self-documenting scripts, and instead relegates visual feedback to the reporting stage, after all processing has ended.

MATLAB remains a widely used, popular platform for scientific computing and data analysis, available in many major universities and academic institutions. Although MATLAB itself is a commercial product, VDI does not require any additional toolboxes to operate, and is free and open source, made available through a modern version control framework (GitHub). It also makes use of standardized file formats for storing and using metadata, such as JSON and XML formats for relaxation data and atlas label definitions, in an attempt to increase standardization among the MRS community. Such efforts are



becoming increasingly common, as evident by a range of consensus papers on its methodology, analysis and reporting standards<sup>42,63–68</sup>.

## **Software Availability**

VDI can be downloaded from the corresponding Author's website:  
<https://www.vdisoftware.net>

## References

1. Simpson R, Devenyi GA, Jezzard P, Hennessy TJ, Near J. Advanced processing and simulation of MRS data using the FID appliance (FID-A)—An open source, MATLAB-based toolkit. *Magn Reson Med*. 2017;77(1):23-33. doi:10.1002/mrm.26091
2. Chong DGQ, Kreis R, Bolliger CS, Boesch C, Slotboom J. Two-dimensional linear-combination model fitting of magnetic resonance spectra to define the macromolecule baseline using FiTAID, a Fitting Tool for Arrays of Interrelated Datasets. *Magn Reson Mater Physics, Biol Med*. 2011;24(3):147-164. doi:10.1007/s10334-011-0246-y
3. Clarke WT, Stagg CJ, Jbabdi S. FSL-MRS: An end-to-end spectroscopy analysis package. *Magn Reson Med*. 2021;85(6):2950-2964. doi:10.1002/mrm.28630
4. Edden RAE, Puts NAJ, Harris AD, Barker PB, Evans CJ. Gannet: A batch-processing tool for the quantitative analysis of gamma-aminobutyric acid–edited MR spectroscopy spectra. *J Magn Reson Imaging*. 2014;40(6):1445-1452. doi:10.1002/jmri.24478
5. Gajdošík M, Landheer K, Swanberg KM, Juchem C. INSPECTOR: free software for magnetic resonance spectroscopy data inspection, processing, simulation and analysis. *Sci Rep*. 2021;11(1):2094. doi:10.1038/s41598-021-81193-9
6. Stefan D, Di Cesare F, Andrasescu A, et al. Quantitation of magnetic resonance spectroscopy signals: the jMRUI software package. *Meas Sci Technol*. 2009;20(10). doi:Artn 104035 10.1088/0957-0233/20/10/104035
7. Provencher SW. Estimation of metabolite concentrations from localized in vivo proton NMR spectra. *Magn Reson Med*. 1993;30(6):672-679. doi:10.1002/mrm.1910300604
8. Provencher SW. Automatic quantitation of localized in vivo <sup>1</sup>H spectra with LCModel. *NMR Biomed*. 2001;14(4):260-264. doi:10.1002/nbm.698
9. Hui SCN, Saleh MG, Zöllner HJ, et al. MRSCloud: A cloud-based MRS tool for basis set simulation. *Magn Reson Med*. 2022;88(5):1994-2004. doi:10.1002/MRM.29370

10. Oeltzschner G, Zöllner HJ, Hui SCN, et al. Osprey: Open-source processing, reconstruction & estimation of magnetic resonance spectroscopy data. *J Neurosci Methods*. 2020;343:108827. doi:10.1016/j.jneumeth.2020.108827
11. Wilson M. spant: An R package for magnetic resonance spectroscopy analysis. doi:10.21105/joss.03646
12. Wilson M, Reynolds G, Kauppinen RA, Arvanitis TN, Peet AC. A constrained least-squares approach to the automated quantitation of in vivo <sup>1</sup>H magnetic resonance spectroscopy data. *Magn Reson Med*. 2011;65(1):1-12. doi:10.1002/mrm.22579
13. Young K, Soher BJ, Maudsley AA. Automated spectral analysis II: Application of wavelet shrinkage for characterization of non-parameterized signals. *Magn Reson Med*. 1998;40(6):816-821. doi:10.1002/mrm.1910400606
14. Chen AM, Gerhalter T, Dehkharghani S, et al. Replicability of proton MR spectroscopic imaging findings in mild traumatic brain injury: Implications for clinical applications. *NeuroImage Clin*. 2023;37:103325. doi:10.1016/J.NICL.2023.103325
15. Harris AD, Amiri H, Bento M, et al. Harmonization of multi-scanner in vivo magnetic resonance spectroscopy: ENIGMA consortium task group considerations. *Front Neurol*. 2023;13:2888. doi:10.3389/FNEUR.2022.1045678/BIBTEX
16. Volovyk O, Tal A. Increased Glutamate concentrations during prolonged motor activation as measured using functional Magnetic Resonance Spectroscopy at 3T. *Neuroimage*. 2020;223. doi:10.1016/j.neuroimage.2020.117338
17. Bezalel V, Paz R, Tal A. Inhibitory and excitatory mechanisms in the human cingulate-cortex support reinforcement learning: A functional Proton Magnetic Resonance Spectroscopy study. *Neuroimage*. 2019;184:25-35. doi:10.1016/j.neuroimage.2018.09.016
18. Kulpanovich A, Tal A. The application of magnetic resonance fingerprinting to single voxel proton spectroscopy. *NMR Biomed*. 2018;31(11):e4001. doi:10.1002/nbm.4001
19. Wolf T, Eden-Kossoy A, Frydman L. Indirectly detected satellite-transition quadrupolar NMR via progressive saturation of the proton reservoir. *Solid State*

- Nucl Magn Reson.* 2023;125:101862. doi:10.1016/J.SSNMR.2023.101862
20. Wolf T, Jaroszewicz MJ, Frydman L. Quadrupolar Isotope-Correlation Spectroscopy in Solid-State NMR. *J Phys Chem C.* 2022;126(22):9386-9395. doi:10.1021/ACS.JPCC.2C00578/ASSET/IMAGES/LARGE/JP2C00578\_0007.JPG
  21. Tal A, Kirov II, Grossman RI, Gonen O. The role of gray and white matter segmentation in quantitative proton MR spectroscopic imaging. *NMR Biomed.* 2012;25(12):1392-1400. doi:10.1002/nbm.2812
  22. Kirov II, Tal A, Babb JS, Herbert J, Gonen O. Serial proton MR spectroscopy of gray and white matter in relapsing-remitting MS. *Neurology.* 2013;80(1):39-46. doi:10.1212/WNL.0b013e31827b1a8c
  23. Kirov II, Tal A, Babb JS, et al. Proton MR spectroscopy correlates diffuse axonal abnormalities with post-concussive symptoms in mild traumatic brain injury. *J Neurotrauma.* 2013;30(13):1200-1204. doi:10.1089/neu.2012.2696
  24. Pasanta D, He JL, Ford T, Oeltzschner G, Lythgoe DJ, Puts NA. Functional MRS studies of GABA and glutamate/Glx – A systematic review and meta-analysis. *Neurosci Biobehav Rev.* 2023;144:104940. doi:10.1016/J.NEUBIOREV.2022.104940
  25. Kolasinski J, Hinson EL, Divanbeighi Zand AP, Rizov A, Emir UE, Stagg CJ. The dynamics of cortical GABA in human motor learning. *J Physiol.* 2019. doi:10.1113/JP276626
  26. Bednařík P, Tkáč I, Giove F, et al. Neurochemical and BOLD responses during neuronal activation measured in the human visual cortex at 7 Tesla. *J Cereb Blood Flow Metab.* 2015;35(4):601-610. doi:10.1038/jcbfm.2014.233
  27. Guglielmetti C, Najac C, Didonna A, der Linden A, Ronen SM, Chaumeil MM. Hyperpolarized (13)C MR metabolic imaging can detect neuroinflammation in vivo in a multiple sclerosis murine model. *Proc Natl Acad Sci U S A.* 2017;114(33):E6982-E6991. doi:10.1073/pnas.1613345114
  28. Larson PE, Bok R, Kerr AB, et al. Investigation of tumor hyperpolarized [1-13C]-

- pyruvate dynamics using time-resolved multiband RF excitation echo-planar MRSI. *Magn Reson Med*. 2010;63(3):582-591. doi:10.1002/mrm.22264
29. Feyter HM De, Behar KL, Corbin ZA, et al. Deuterium metabolic imaging (DMI) for MRI-based 3D mapping of metabolism in vivo. *Sci Adv*. 2018;4(8):eaat7314. doi:10.1126/sciadv.aat7314
  30. Kaggie JD, Khan AS, Matys T, et al. Deuterium metabolic imaging and hyperpolarized <sup>13</sup>C-MRI of the normal human brain at clinical field strength reveals differential cerebral metabolism. *Neuroimage*. 2022;257:119284. doi:10.1016/J.NEUROIMAGE.2022.119284
  31. Straathof M, Meerwaldt AE, De Feyter HM, de Graaf RA, Dijkhuizen RM. Deuterium Metabolic Imaging of the Healthy and Diseased Brain. *Neuroscience*. 2021;474:94-99. doi:10.1016/J.NEUROSCIENCE.2021.01.023
  32. Kulpanovich A, Tal A. What is the optimal schedule for multiparametric MRS? A magnetic resonance fingerprinting perspective. *NMR Biomed*. 2021;34(5):e4196. doi:10.1002/nbm.4196
  33. Kirov II, Tal A. Potential clinical impact of multiparametric quantitative MR spectroscopy in neurological disorders: A review and analysis. *Magn Reson Med*. 2020;83(1):22-44. doi:10.1002/mrm.27912
  34. Sandgren N, Stoica P, Frigo FJ, Selén Y. Spectral analysis of multichannel MRS data. *J Magn Reson*. 2005;175(1):79-91. doi:10.1016/J.JMR.2005.03.019
  35. Rodgers CT, Robson MD. Receive array magnetic resonance spectroscopy: Whiten singular value decomposition (WSVD) gives optimal Bayesian solution. *Magn Reson Med*. 2010;63(4):881-891. doi:10.1002/MRM.22230
  36. Mikkelsen M, Tapper S, Near J, Mostofsky SH, Puts NAJ, Edden RAE. Correcting frequency and phase offsets in MRS data using robust spectral registration. *NMR Biomed*. 2020;33(10). doi:10.1002/nbm.4368
  37. Chen L, Weng Z, Goh LY, Garland M. An efficient algorithm for automatic phase correction of NMR spectra based on entropy minimization. *J Magn Reson*. 2002;158(1-2):164-168. doi:10.1016/S1090-7807(02)00069-1

38. Clarke WT, Ligneul C, Cottaar M, Ip IB, Jbabdi S. Universal Dynamic Fitting of Magnetic Resonance Spectroscopy. *bioRxiv*. June 2023:2023.06.15.544935. doi:10.1101/2023.06.15.544935
39. Tal A. *The Future Is 2D: Spectral-Temporal Fitting of Dynamic MRS Data Provides Exponential Gains in Precision over Conventional Approaches*. John Wiley and Sons Inc; 2022:1-10. doi:10.1002/mrm.29456
40. Gussew A, Erdtel M, Hiepe P, Rzanny R, Reichenbach JR. Absolute quantitation of brain metabolites with respect to heterogeneous tissue compositions in (1)H-MR spectroscopic volumes. *MAGMA*. 2012;25(5):321-333. doi:10.1007/s10334-012-0305-z
41. Jansen JF, Backes WH, Nicolay K, Kooi ME. 1H MR spectroscopy of the brain: absolute quantification of metabolites. *Radiology*. 2006;240:318-332. doi:10.1148/radiol.2402050314
42. Near J, Harris AD, Juchem C, et al. Preprocessing, analysis and quantification in single-voxel magnetic resonance spectroscopy: experts' consensus recommendations. *NMR Biomed*. 2021;34(5). doi:10.1002/nbm.4257
43. Gasparovic C, Song T, Devier D, et al. Use of tissue water as a concentration reference for proton spectroscopic imaging. *Magn Reson Med*. 2006;55(6):1219-1226. doi:10.1002/mrm.20901
44. Finkelman T, Furman-Haran E, Paz R, Tal A. Quantifying the excitatory-inhibitory balance: A comparison of SemiLASER and MEGA-SemiLASER for simultaneously measuring GABA and glutamate at 7T. *Neuroimage*. 2022;247:118810. doi:10.1016/j.neuroimage.2021.118810
45. Christiansen P, Henriksen O, Stubgaard M, Gideon P, Larsson HBW. In vivo quantification of brain metabolites by 1H-MRS using water as an internal standard. *Magn Reson Imaging*. 1993;11(1):107-118. doi:10.1016/0730-725X(93)90418-D
46. Hoch SE, Kirov II, Tal A. When are metabolic ratios superior to absolute quantification? A statistical analysis. *NMR Biomed*. 2017;30(7). doi:10.1002/nbm.3710

47. Malucelli E, Manners DN, Testa C, et al. Pitfalls and advantages of different strategies for the absolute quantification of N-acetyl aspartate, creatine and choline in white and grey matter by 1H-MRS. *NMR Biomed.* 2009;22(10):1003-1013. doi:10.1002/nbm.1402
48. Davitz MS, Gonen O, Tal A, Babb JS, Lui YW, Kirov II. Quantitative multivoxel proton MR spectroscopy for the identification of white matter abnormalities in mild traumatic brain injury: Comparison between regional and global analysis. *J Magn Reson Imaging.* 2019;50(5):1424-1432. doi:10.1002/jmri.26718
49. Maudsley AA, Domenig C, Sheriff S. Reproducibility of serial whole-brain MR spectroscopic imaging. *NMR Biomed.* 2010;23(3):251-256. doi:10.1002/nbm.1445
50. Posse S, Otazo R, Caprihan A, et al. Proton echo-planar spectroscopic imaging of J-coupled resonances in human brain at 3 and 4 Tesla. *Magn Reson Med.* 2007;58(2):236-244. doi:10.1002/mrm.21287
51. Maudsley AA, Govind V, Arheart KL. Associations of age, gender and body mass with 1H MR-observed brain metabolites and tissue distributions. *NMR Biomed.* 2012;25(4):580-593. doi:10.1002/NBM.1775
52. Kirov II, Tal A, Babb JS, Lui YW, Grossman RI, Gonen O. Diffuse axonal injury in mild traumatic brain injury: a 3D multivoxel proton MR spectroscopy study. *J Neurol.* 2013;260(1):242-252. doi:10.1007/s00415-012-6626-z
53. Govind V, Gold S, Kaliannan K, et al. Whole-brain proton MR spectroscopic imaging of mild-to-moderate traumatic brain injury and correlation with neuropsychological deficits. *J Neurotrauma.* 2010;27(3):483-496. doi:10.1089/neu.2009.1159
54. Gasparovic C, Yeo R, Mannell M, et al. Neurometabolite concentrations in gray and white matter in mild traumatic brain injury: an 1H-magnetic resonance spectroscopy study. *J Neurotrauma.* 2009;26(10):1635-1643. doi:10.1089/neu.2009-0896 10.1089/neu.2009.0896
55. Yeo RA, Gasparovic C, Merideth F, Ruhl D, Doezenia D, Mayer AR. A longitudinal proton magnetic resonance spectroscopy study of mild traumatic brain injury. *J Neurotrauma.* 2011;28(1):1-11. doi:10.1089/neu.2010.1578



56. Mayer AR, Ling JM, Dodd AB, Gasparovic C, Klimaj SD, Meier TB. A Longitudinal Assessment of Structural and Chemical Alterations in Mixed Martial Arts Fighters. *J Neurotrauma*. 2015;32(22):1759-1767. doi:10.1089/NEU.2014.3833/ASSET/IMAGES/LARGE/FIGURE4.JPEG
57. Dennis EL, Babikian T, Alger J, et al. Magnetic resonance spectroscopy of fiber tracts in children with traumatic brain injury: A combined MRS – Diffusion MRI study. *Hum Brain Mapp*. 2018;39(9):3759-3768. doi:10.1002/HBM.24209
58. Azevedo CJ, Kornak J, Chu P, et al. In vivo evidence of glutamate toxicity in multiple sclerosis. *Ann Neurol*. 2014;76(2):269-278. doi:10.1002/ANA.24202
59. Gerhalter T, Chen AM, Dehkharghani S, et al. Global decrease in brain sodium concentration after mild traumatic brain injury. *Brain Commun*. 2021;3(2). doi:10.1093/BRAINCOMMS/FCAB051
60. Rolls ET, Huang CC, Lin CP, Feng J, Joliot M. Automated anatomical labelling atlas 3. *Neuroimage*. 2020;206:116189. doi:10.1016/J.NEUROIMAGE.2019.116189
61. Bell TK, Godfrey KJ, Ware AL, Yeates KO, Harris AD. Harmonization of multi-site MRS data with ComBat. *Neuroimage*. 2022;257:119330. doi:10.1016/J.NEUROIMAGE.2022.119330
62. Dewey BE, Zhao C, Reinhold JC, et al. DeepHarmony: A deep learning approach to contrast harmonization across scanner changes. *Magn Reson Imaging*. 2019;64:160-170. doi:10.1016/j.mri.2019.05.041
63. Ligneul C, Najac C, Döring A, et al. Diffusion-weighted MR spectroscopy: consensus, recommendations and resources from acquisition to modelling. May 2023. <https://arxiv.org/abs/2305.10829v1>. Accessed August 3, 2023.
64. Wilson M, Andronesi O, Barker PB, et al. Methodological consensus on clinical proton MRS of the brain: Review and recommendations. *Magn Reson Med*. 2019;82(2):527-550. doi:10.1002/MRM.27742
65. Öz G, Deelchand DK, Wijnen JP, et al. Advanced single voxel 1H magnetic resonance spectroscopy techniques in humans: Experts' consensus recommendations. *NMR Biomed*. 2021;34(5). doi:10.1002/NBM.4236

66. Andronesi OC, Bhattacharyya PK, Bogner W, et al. Motion correction methods for MRS: experts' consensus recommendations. *NMR Biomed.* 2021;34(5):e4364. doi:10.1002/NBM.4364
67. Tkáč I, Deelchand D, Dreher W, et al. Water and lipid suppression techniques for advanced 1H MRS and MRSI of the human brain: Experts' consensus recommendations. *NMR Biomed.* 2021;34(5):e4459. doi:10.1002/NBM.4459
68. Maudsley AA, Andronesi OC, Barker PB, et al. Advanced magnetic resonance spectroscopic neuroimaging: Experts' consensus recommendations. *NMR Biomed.* 2021;34(5). doi:10.1002/NBM.4309