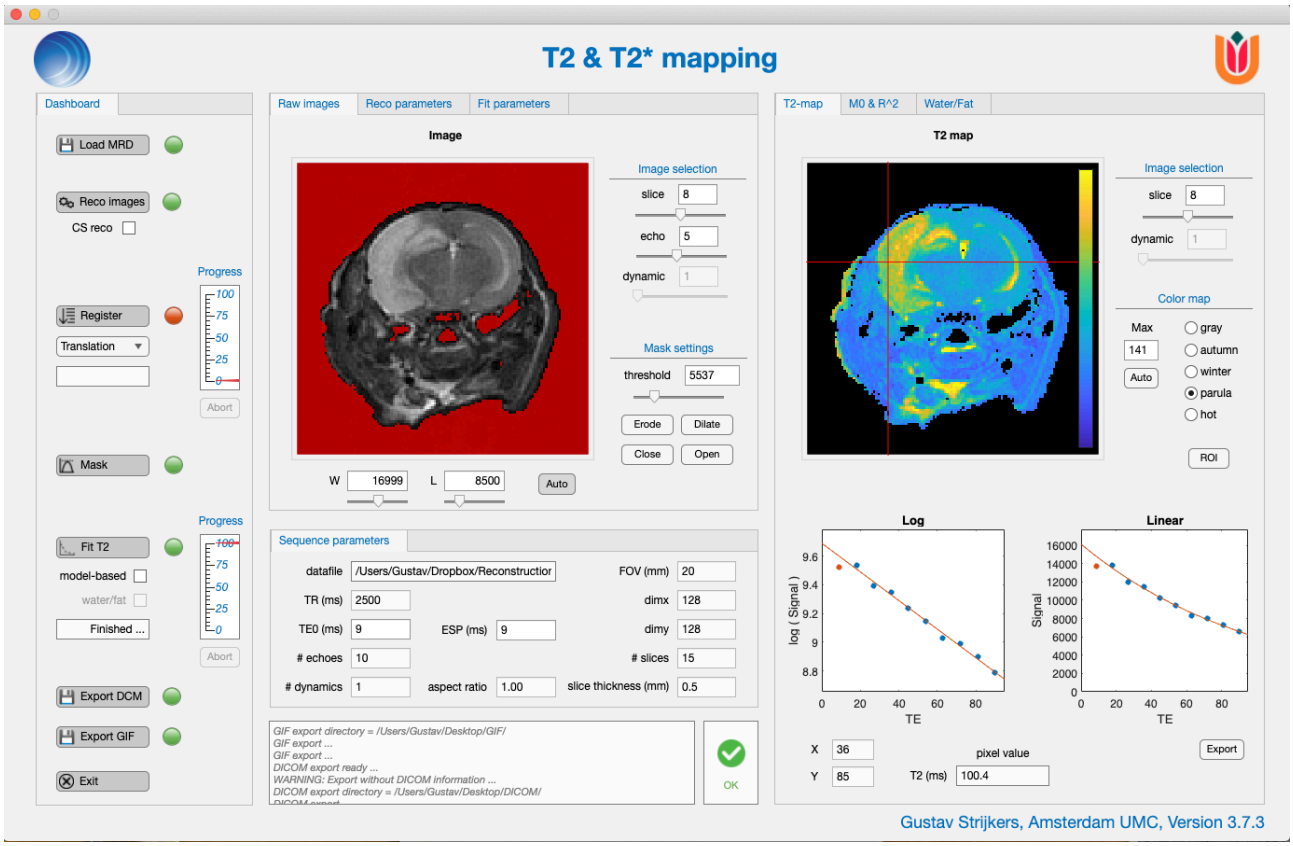


Table of contents

1. Overview

T2mapp is a software utility and graphical user interface to reconstruct T2 maps from multi-echo spin-echo and T2* maps from multi-gradient-echo MRI acquired with an MR Solutions preclinical MRI system.

2. Contact

T2mapp and this manual are written by:

Gustav Strijkers
Amsterdam UMC
Biomedical Engineering and Physics
Amsterdam, the Netherlands
g.j.strijkers@amsterdamumc.nl

3. Disclaimer

The software has been extensively tested using mouse and rat data acquired with an MR Solutions preclinical 7.0T/24cm system.

However, this does not warrant the functions contained in the program will meet your requirements or that the operation of the program will be uninterrupted or error-free.

In case of questions or issues, please contact Gustav Strijkers.

4. Installation notes

Software download

Matlab source code and a Windows standalone version (using the free Matlab runtime engine) can be downloaded from github:

`https://github.com/Moby1971?tab=repositories`

Installation of the Windows standalone version

`MyAppInstaller_web.exe`

Will install the Matlab runtime engine and the T2mapp program.

Bart toolbox download

The advanced reconstruction options in T2mapp require the Bart reconstruction toolbox, which can be downloaded from:

`https://mrirecon.github.io/bart/`

Bart toolbox installation in OSX with MacPorts

(1) Install Xcode from the Mac App Store

```
$ xcode-select --install
```

(2) Install MacPorts (`http://www.macports.org/`)
It is recommended to install a newer version of gcc from MacPorts

(3) Install packages

```
$ sudo port install fftw-3-single
$ sudo port install gcc10
$ sudo port install libpng
$ sudo port install openblas
```

(4) Make

```
$ make
$ sudo make PREFIX=/usr/local install
```

Bart toolbox installation in OSX with HomeBrew (for Mac with Apple Silicon)

(1) Install Xcode from the Mac App Store

```
$ xcode-select --install
```

(2) Install HomeBrew

```
$ /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/
Homebrew/install/HEAD/install.sh)"
```

(3) Install packages

```
$ brew install --cask gcc-arm-embedded
$ brew install libpng
$ brew install fftw
$ brew install openblas
```

(3) Make

```
- Use the provided makefile for Apple Silicon
$ make
$ sudo make PREFIX=/usr/local install
```

Bart toolbox installation in Windows 10

(1) Install the Windows subsystem for Linux

Start a windows powershell as administrator and run the following command:

```
Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Windows-Subsystem-Linux
```

A system restart will be needed.

For more information, see:

```
https://docs.microsoft.com/en-us/windows/wsl/install-win10
```

(2) Download Ubuntu Linux

Open PowerShell

To see a list of available Linux distributions for download, enter:

```
wsl -l -o
```

Install a Linux distribution:

```
wsl --install -d <distribution name>
```

You will be asked to create a user account the first time you start a Linux command prompt.

(3) Upgrade the Linux distribution

```
$ sudo apt-get update
$ sudo apt-get dist-upgrade
```

(4) Install Bart prerequisites

```
$ sudo apt-get install make gcc gcc-10 libfftw3-dev liblapacke-dev
  libpng-dev libopenblas-dev gfortran
```

(5) Download Bart

```
$ wget https://github.com/mrirecon/bart/archive/v0.8.00.tar.gz
```

For WSL1 version 0.4.02 is recommended. Newer versions seem to compile but some functions which are required produce errors.

Recommended: Upgrade to WSL2 (see e.g. https://pureinfotech.com/install-windows-subsystem-linux-2-windows-10/ on how to upgrade).

(6) Build Bart

```
$ tar xvfz v0.8.00.tar.gz
$ cd bart-0.8.00
$ make
$ sudo make PREFIX=/usr/local install
```

5. Running the software

Running in Matlab 2023a

The T2mapp software can be started from its root directory from the command line.

```
>> T2mapp
```

Additional licenses may be required.

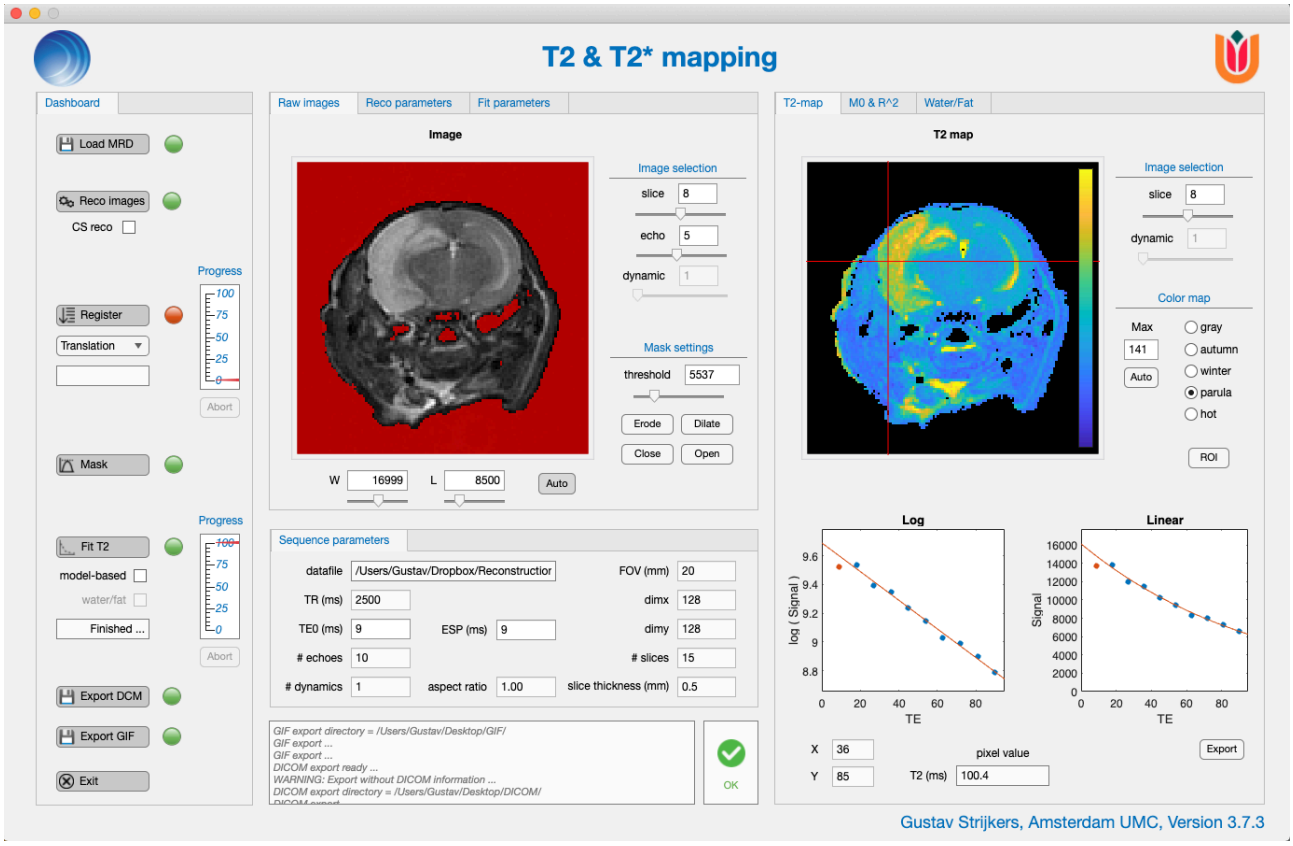
```
>> license 'inuse'
   distrib_computing_toolbox
   image_toolbox
   matlab
   phased_array_system_toolbox
   signal_blocks
   signal_toolbox
```

Running the Windows standalone

The Windows standalone version can be run from the start menu or the desktop icon.

6. Basic operation

The T2mapp program operates from a single window with 5 panels.



Panel 1: Dashboard

This panel contains the task buttons and parameters that control the reconstruction process. The dashboard tasks need to be completed from top to bottom. A green light next to the task indicates that the task has been completed. Red indicates not completed yet.

Panel 2: Raw images - Reco parameters - Fit parameters

This panel has 3 tabs for raw images, reconstruction parameters and fit parameters.

Panel 3: Scan parameters

This panel displays the relevant acquisition parameters.

Panel 4: T2-map - M0 & R^2 - Water/Fat

During and after fitting is completed this panel shows the fitted T2-maps. M0-maps and R^2 goodness of fit maps are displayed in the second tab. The 3rd tab is for water/fat imaging.

Panel 5: Messages

Displays program status and messages.

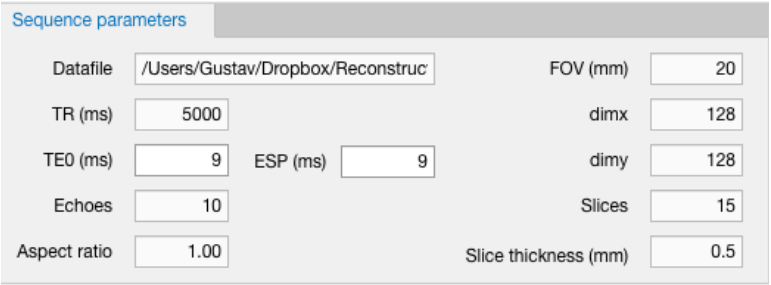
Step 1: Loading data

Press Load MRD to import load the MRD raw data file.

T2mapp in needs multi-echo spin-echo data or multi-gradient-echo (FLASH) data.

For multi receiver coil data, select 1 of the MRD raw data files.

Relevant acquisition parameters will be shown in panel 3.



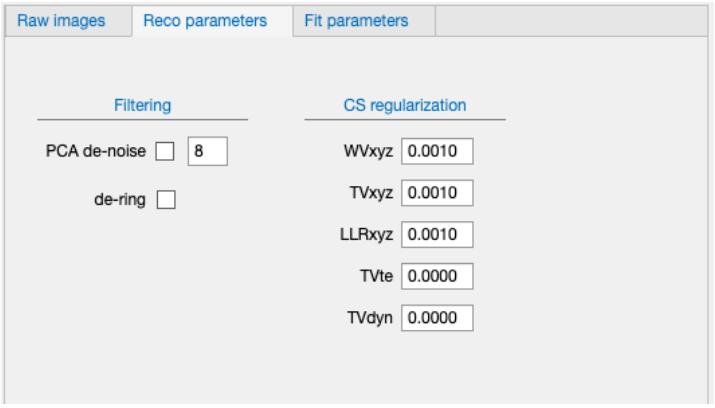
Step 2: Image reconstruction

Upon initial loading of the data, an automatic FFT image reconstruction will be performed.

For k-space undersampled data or to improve image quality by iterative regularization, a compressed sensing reconstruction of the data can be performed.

Select the CS reco check-box CS reco ☑ to enable compressed sensing reconstruction.

Compressed sensing reconstruction regularization parameters can be accessed via the reconstruction parameters panel.



Five compressed sensing reconstruction regularization parameters are available: wavelet in spatial domain (WVxyz), total variation in spatial domain (TVxyz), locally-low rank in spatial domain (LLRxyz), total variation for echo-time (TVte), and total variation in the dynamic domain (TVdyn).

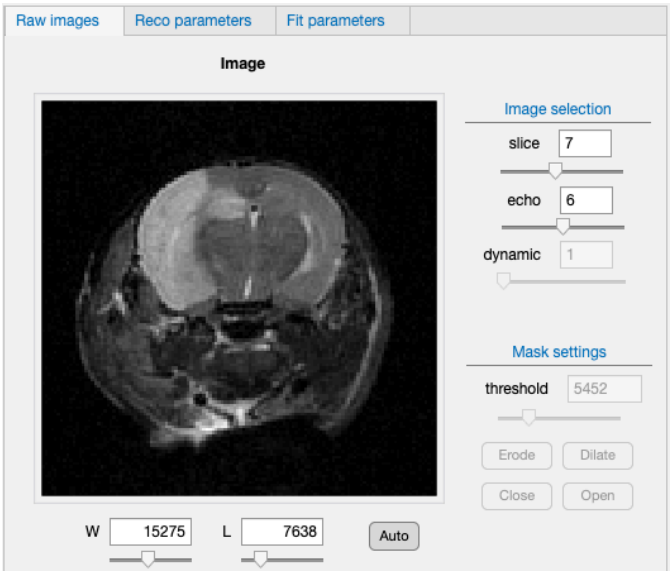
Additionally a principal-component analysis denoising and a Gibbs ringing suppression can be engaged.

After setting the parameters, press Reco images to reconstruct the images.

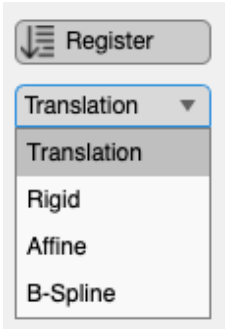
Step 3: Image registration

NOTE: this step usually can be omitted.

Use the sliders and/or edit-fields to inspect the slices and images as function of echo-time (TE) or dynamic/repetition. The image scale can be adjusted with the window (W) and Level (L) values and sliders.



In case the images as function of TE are not perfectly registered, press Register to perform an image registration.



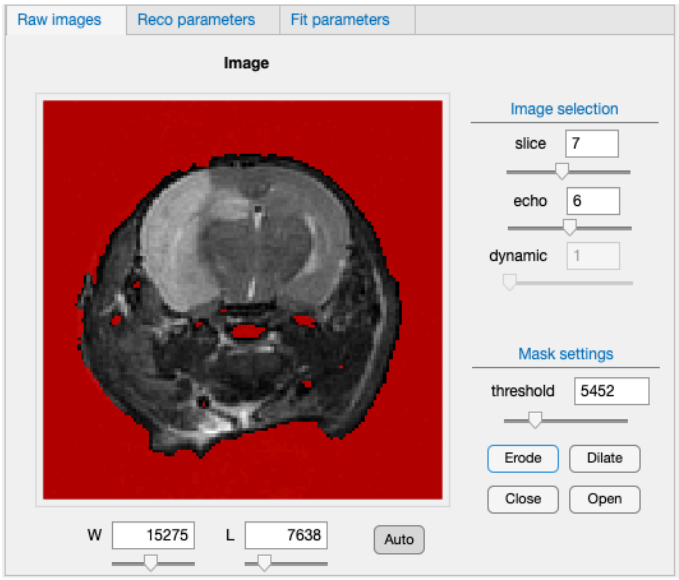
There are 4 possible registration methods translation, rigid, affine, and b-spline, which can be selected via the drop-down menu below the register button.

Step 4: Masking the background in the images

To prevent unnecessary fitting of the background pixels, the images need to be masked first.

Press Mask to perform an automatic masking via thresholding.

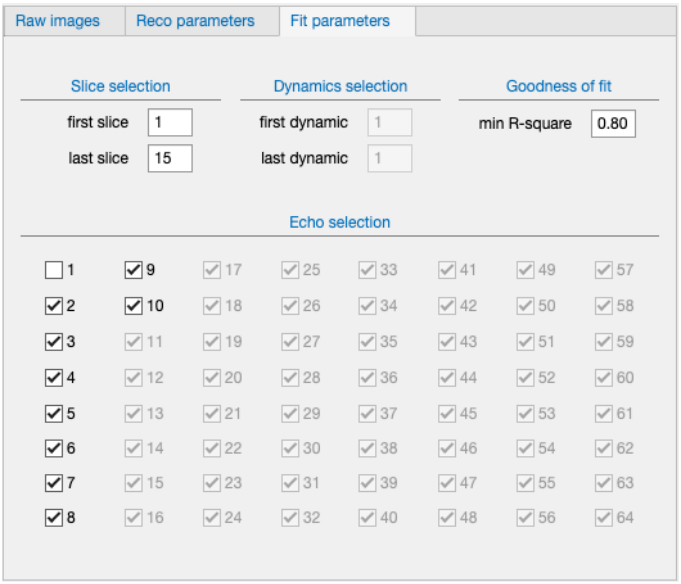
The result will look something like the figure below in which the red pixels indicate the regions that will be skipped during T2 fitting.



In case automatic thresholding is not satisfying, the threshold can be manually adjusted per slice. The mask can be optimized by erode, dilate, close, and open morphological operations.

Step 5: Fitting T2 maps

Press the Fit parameters tabs to inspect some fit settings.



The first and last slice that will be fitted can be selected.
The first and last dynamic that will be fitted can be selected.

Min R-square is the minimally accepted R^2 value that is accepted after fitting. If the result of a T2 fit in a pixel results in an R^2 value less that Min R-square, then the T2 and M0 values in this pixel will be omitted and set to zero.

The bottom part of this tab shows a list of all the echo numbers. The echos that are selected will be included in the fitting process. This option can be used to exclude certain TEs, for example the first TE, the last TEs below noise level, or in some cases you may only want to fit the even or odd echo numbers.

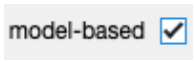
Press 

to start the T2 fitting.

Fitting will be done using a least-squares algorithm according to the equation

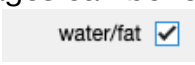
$$Signal = M_0 \exp(-TE/T_2)$$

Model-based fitting

Thick the model-based checkbox 

to perform a model-based reconstruction of the data using the Bart toolbox.

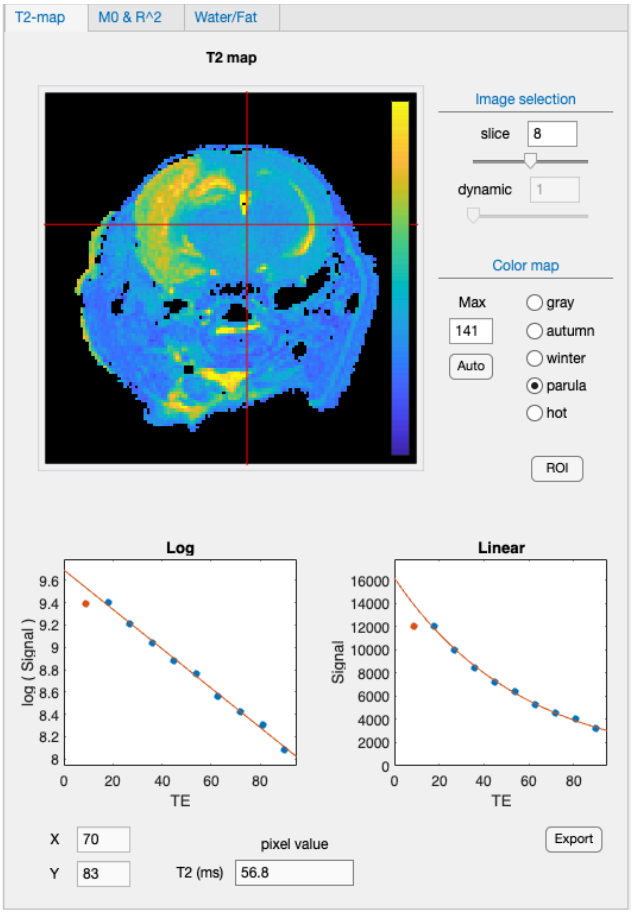
Water/fat separation

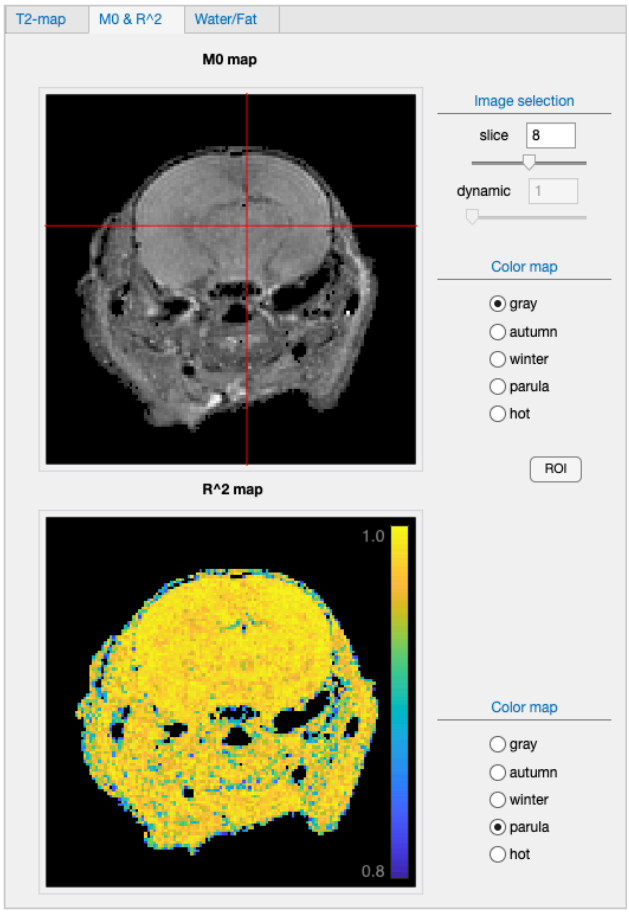
DIXON-based water/fat images can be reconstructed for multi-gradient echo data by checking the water/fat box 

.

Step 6: Fit results

The T2 maps will be shown in the T2-map tab. An M0 map and R^2 map are shown in the second tab. A different color scheme can be chosen for the different maps.

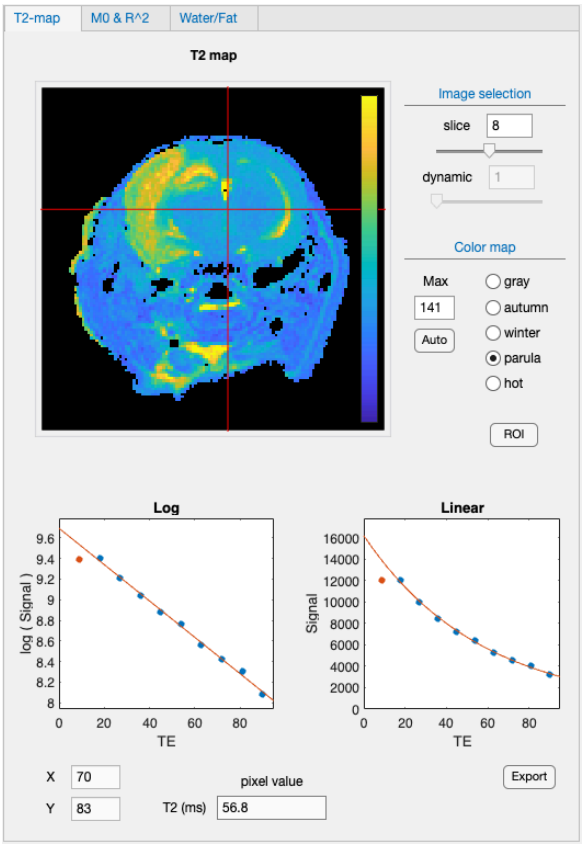


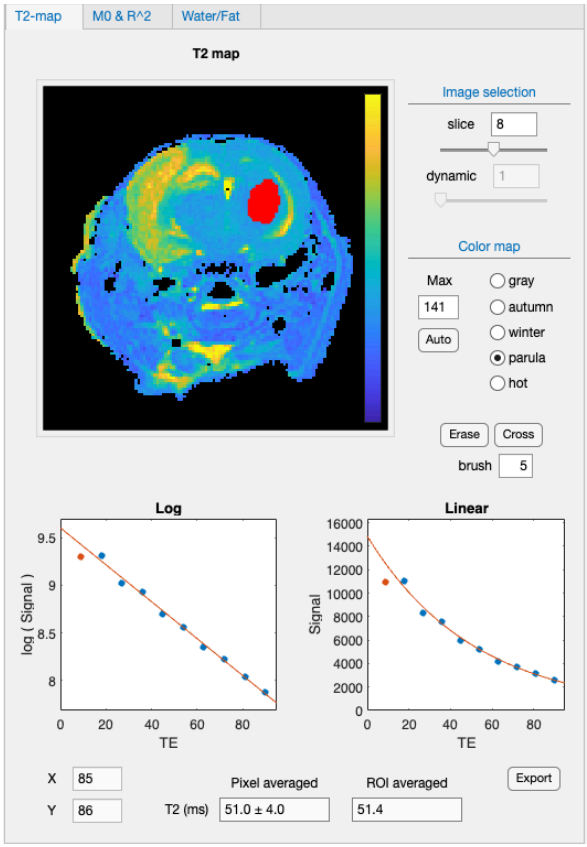


Step 7: Extracting T2 and T2* values from pixels and ROIs

When you click on the T2 map with the mouse cursor a cross-hair will be shown. The log(Signal) and Signal as function of TE (symbols) and the fit result (red lines) of the cross-hair center pixel, as well as the fitted T2(*) value of that pixel will be shown.

A red symbol indicates that this TE was omitted in the fitting process, according to the list in the Fit parameters tab (see step 4).





You can toggle between the cross-hair and region-of-interest (ROI) values by pressing the ROI button. A single ROI can be 'painted' on the T2 map in one or more slices by mouse clicking on the image.

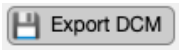
For the ROI, two T2(*) values are calculated:

- Pixel averaged: The average of the T2(*) values in the pixels.
  $T2_{ROI} = 1/Sum(R2_{pixel})$
- ROI averaged: The T2(*) equation is fitted to the average signal intensities in the ROI, producing a single T2(*) value for that ROI.

The ROI can be removed with Erase .

Step 8: Exporting the maps

There are two ways to export the maps.

(1) **Export Dicom** 

Exports the data in Dicom format for further processing in 3rd party software. The program searches for the Dicom information. If this information is not found, tags will be generated by the program itself. In the latter case the correct image position and orientation information are lost.

To prevent discretization into integer values in de Dicom files, the values for the T2 and R^2 maps are multiplied by 100. (Thus actual T2 = $T2_{dicom\ value}$ / 100)

(2) **Export Gif** 

Exports the data in GIF format.

Step 8: Exit

Press 

to shut down the program.