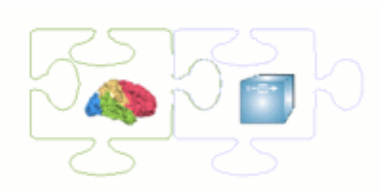
nifti converter plugin manual
for
brainvoyager qx



Hester Breman and Rainer Goebel

Maastricht, April 2010

Contents

1 Introduction **4**
- 1.1 General information on using the converter . . . 4
  - 1.1.1 General procedure . . . 4
  - 1.1.2 Naming of converted files . . . 4
  - 1.1.3 Location of converted files . . . 4
  - 1.1.4 Design . . . 4
  - 1.1.5 The NIfTI-1 data format . . . 4
  - 1.1.6 Background information . . . 5
  - 1.1.7 Surface files . . . 5
  - 1.1.8 General limitations of the converter . . . 5
- 1.2 Future . . . 6
  - 1.2.1 Planned updates in the plugin . . . 6
- 1.3 History . . . 7
  - 1.3.1 Changes in the plugin . . . 7
  - 1.3.2 Changes in the documentation . . . 9
- 1.4 Installation . . . 10
- 1.5 Main menu . . . 11
- 1.6 Acknowledgements . . . 12

2 Conversion functions **13**
- 2.1 Conversion menu . . . 13
- 2.2 Converting one file . . . 13
- 2.3 Explanation of options . . . 14
  - 2.3.1 Transform . . . 14
  - 2.3.2 Flip . . . 14
  - 2.3.3 Open in BrainVoyager QX . . . 14
  - 2.3.4 Zeropad . . . 14
  - 2.3.5 Adapt intensity for VMR import . . . 14
  - 2.3.6 Adapt interpolation for import of VOI to VMR . . . 14
- 2.4 Converting a list of files . . . 15
  - 2.4.1 Creating a text file with the list of files to be converted . . . 15
  - 2.4.2 Run the conversions . . . 16
- 2.5 Converting coregistered files . . . 18

3 Converting anatomical data **19**
- 3.1 Illustration of the process . . . 19
  - 3.1.1 Import to *.vmr . . . 19
  - 3.1.2 Export from *.vmr . . . 22
- 3.2 Possibilities and limitations . . . 23
- 3.3 Anatomical file formats in BrainVoyager . . . 24

4 Converting functional data **25**
4.1 Illustration of the process . . . 25
4.1.1 FMR files . . . 25
4.1.2 VTC files . . . 29
4.2 Possibilities and limitations . . . 31
4.2.1 FMR files . . . 31
4.2.2 VTC files . . . 31
4.3 Functional file formats in BrainVoyager . . . 32

5 Converting diffusion-weighted data **33**
5.1 Illustration of the process . . . 33
5.1.1 Importing a *.nii file to *.dmr/*.dwi . . . 33
5.1.2 Exporting a *.dmr project to NIfTI-1 . . . 35
5.2 Possibilities and limitations . . . 36
5.3 Diffusion-weighted file formats in BrainVoyager . . . 37

6 Converting statistical data **38**
6.1 Illustration of the process . . . 38
6.1.1 Import to *.vmp . . . 38
6.1.2 Export from *.vmp . . . 40
6.2 Possibilities and limitations . . . 41
6.2.1 Recognized map types . . . 41
6.2.2 Parameters . . . 41
6.2.3 Tables . . . 42

7 Converting sparse data (VOI) **43**
7.1 Illustration of the process . . . 43
7.1.1 Import to *.voi . . . 43
7.1.2 Export from *.voi . . . 44
7.2 Possibilities and limitations . . . 46
7.3 Sparse file formats in BrainVoyager . . . 47

8 Conversion cases **48**
8.1 Preprocessing in AFNI, rest in BrainVoyager . . . 48
8.2 Segmentation in FSL, rest in BrainVoyager . . . 49
8.3 Import a batch of eddy current corrected files from FSL into BrainVoyager . . . 50
8.4 Importing maps from GingerALE . . . 51
8.5 Import data from NeuroLens via FreeSurfer into BrainVoyager . . . 53
8.6 Processing in BrainVoyager, surfaces in SUMA or Caret . . . 54
8.7 Image registration in SPM, rest in BrainVoyager . . . 55
8.8 EPI distortion correction in SPM, rest in BrainVoyager . . . 58

9 Problem solving **59**
9.1 Solutions to problems . . . 59
9.1.1 My functional data files (*.hdr/*.img) are not 4D . . . 59
9.1.2 The brightness of the data sets looks different . . . 59
9.1.3 A problem occurred during conversion . . . 59
9.1.4 How do I know what is left and right? . . . 60
9.1.5 The positioning information in my *.fmr/*.dmr file is wrong . . . 61
9.1.6 The orientation of the image is wrong . . . 63

9.1.7 I don't see an image, only stripes . . . 66
9.1.8 My data are in MNI space . . . 66
9.2 My problem is not solved after reading the first part of this chapter . 67

10 Obtaining information about an image 68
10.1 Introduction . . . 68
10.1.1 Header information menu . . . 68
10.2 Print image information to BrainVoyager QX Log tab . . . 70
10.3 Print image information to a text file . . . 70
10.4 Print image information to BrainVoyager QX Log tab and text file . . 70

11 Positioning functions 71
11.1 Represent positioning (*.pos) file in matrix . . . 71
11.1.1 Purpose . . . 71
11.1.2 Using the function . . . 72
11.2 Create positioning (*.pos) file for NIfTI-1 image . . . 73
11.2.1 Purpose . . . 73
11.2.2 Using the function . . . 73
11.3 On positioning terminology . . . 74
11.3.1 Orientation of the image . . . 75
11.3.2 Position of the image . . . 76
11.3.3 Orientation and positioning information in a matrix . . . 77

12 Transformation functions 78
12.1 Transformations menu . . . 78
12.2 Notes on transformations . . . 79
12.2.1 Introduction . . . 79
12.2.2 Storage of transformation information . . . 79
12.2.3 Differences in storage of transformations . . . 81

A Spatial transformations in BrainVoyager 82
A.1 Theoretical background and implementation in BrainVoyager . . . 82
A.1.1 Introduction . . . 82
A.1.2 BrainVoyagers axes conventions . . . 83
A.1.3 3D affine transformation matrices . . . 86
A.1.4 Composing 3D affine transformations . . . 88
A.1.5 The order of rotations in BrainVoyager . . . 88
A.1.6 The TRF file format for spatial transformations . . . 90
A.1.7 Summary of coordinate systems . . . 92

B Transformations 93
B.1 Contents of transformation matrices . . . 94
B.2 Transformation matrices used during conversion . . . 95
B.3 The quaternion in the NIfTI-1 header . . . 97

C BrainVoyager file formats 99

D Creation of the MNI 305 template 101

Chapter 1

Introduction

1.1 General information on using the converter

1.1.1 General procedure

First, read or load the file that should be converted or for which one desires to obtain information. In case of BrainVoyager files, they need to be currently opened in BrainVoyager QX because the BrainVoyager QX plugin access functions are used to get access to image information and data.
Then, if is chosen for conversion, the output format can be selected.

1.1.2 Naming of converted files

The names for produced files are generally adopted from the original files. In case the files are not converted images, an infix might be added which indicates the type of provided information in the file (for example a header dump has the extension `<image name>_info.txt`).

1.1.3 Location of converted files

Also, the produced files can be found in the same directory as the original files.

1.1.4 Design

The design of the converter is that it is basically capable of converting any incorporated image format to another. This means that the file-specific information is not transferred to the converted image file. However, this information can be printed to a text file, an option that is offered afterwards or directly in case one does not select the conversion option.
Currently, only the conversions of original purpose have been tested, which is between the BrainVoyager QX formats and the NIfTI-1 format.
More information about the design of the NIfTI-1 converter can be found in the NIfTI-1 converter design document, also available on the BrainVoyager wiki:
`http://wiki.brainvoyager.com/BVQX_plugins/`.

1.1.5 The NIfTI-1 data format

This is a manual for the nifti-converter for BrainVoyager QX [4]. NIfTI-1 is the data standard defined by the Data Format Working Group (DFWG) as successor of the Analyze 7.5 format.

For more information about the data format, see http://nifti.nimh.nih.gov and [2]. It is also downloadable in *.chm and FlashHelp format from the BrainVoyager wiki at http://wiki.brainvoyager.com/BVQX_Plugins. For libraries, see http://niftilib.sourceforge.net. The NIfTI converter is based on the nifti-c library that was kindly made available by R.W. Cox and R. Reynolds.

1.1.6 Background information

When importing files to one of the BrainVoyager formats, the anatomical image volume(s) mostly needs to be reoriented so that it will be in sagittal orientation. Also, the voxel sizes often need to be equal (for VTC and VMP).
When exporting to NIfTI-1 format, it can be convenient to have the image volume(s) reoriented so that its axes correspond with the NIfTI-1 coordinate system. Therefore the images are transformed in most cases using code from P. Thevenaz [6] with kind permission. The selected interpolation degree is quadratic. In later versions one will be able to select between a range of nearest neighbor and septic (quadratic is a fine accuracy, somewhere in between).
In the other cases for exporting files, a standard transformation matrix is written in the NIfTI-1 header file which indicates its orientation with respect to the NIfTI-1 coordinate system. The software package SPM5/SPM8 will take the orientation matrix into account when displaying the image. In FSL 3.2 the image that is not transformed but has an orientation matrix is shown in its original position.
The transformations are based on a general conversion between the BrainVoyager QX internal and the NIfTI-1 coordinate systems and currently does not take individual orientation information into account.

1.1.7 Surface files

Please note that the conversion of surface files (SRF, MTC, SMP) is implemented in a separate plugin: the GIFTI converter. This plugin is available with BrainVoyager QX 2.0. For other conversions, please see the BVQXTools for Matlab by Jochen Weber on the BrainVoyager QX wiki
at http://wiki.brainvoyager.com/. The NIfTI and GIFTI plugins can be downloaded from
http://support.brainvoyager.com/.

1.1.8 General limitations of the converter

Table 1.1: Cases which the converter currently does not accept or have not yet been tested

property	*Import to BVQX*	*Export to NIfTI-1*
extensions (extra stored information)	no	no
positioning info	beta	beta
transformation info	to native space	beta
datatype	complex	-
system byte order	big endian	big endian

1.2 Future

1.2.1 Planned updates in the plugin

- Zeropad VMP files for export
- Apply any transformation to an image
- Import of MAP files
- Import of AMR files

1.3 History

1.3.1 Changes in the plugin

Version 1.07.260310
A graphical user interface has been added for BrainVoyager QX 2.1. When importing to VMR, FMR or DMR, it is possible to automatically open the project in BrainVoyager after conversion. The export from VMR is using the good-old transformation matrix from before summer 2009.

Version 1.07.050210
DMR import fix (slices and volumes reversed).

Version 1.06.260809
The export to VTC function is updated to BrainVoyager QX 2.0. Fixed a bug when not scaling intensity values when importing to VMR.

Version 1.06.080709
This version imports one pair or multiple coregistered anatomical and functional data (see section **??**). Also, the situation that the number of volumes in an imported NIfTI file is set to 0 (should be 1 in case of a single volume!) is now handled more gracefully.

Version 1.06.170609
This version is for BrainVoyager QX 2.0. It writes FMR's STC files in float format, it asks whether to reorient images and rescale intensity values rather than just doing it, the positioning information was broken (again) and fixed (again), for importing anatomical volumes also *.v16 images are written and if the anatomical volume is reoriented during conversion, this is saved in the "past transformations" header.

Version 1.06.121108
This version reads and writes positioning information for BrainVoyager project files (*.vmr/*.fmr/*.dmr) in native space.

Version 1.05.210808
Importing statistical maps to *.vmp. Zero-padding to 256 x 256 x 256 for importing to *.vmr files.

Version 1.05.070808
Swapping for big endian NIfTI files, statistical codes for *.vmp files that are exported to NIfTI.

Version 1.05.0000
The nifti-converter has been completely refurbished because of portability limitations and code maintenance reasons. Version 1.04 was only for Windows. The new version (from 1.05) will be cross-platform. Also the source code will be available.
Furthermore, the BrainVoyager QX plugin access functions have been used to obtain access to the files instead of reading and writing directly from and to disk.
Finally, the implementation is based on C++ instead of C to a much larger extent, using inheritance and templates. It is however still under construction so not all

features shown in the menus are available yet.
For more information about the design, please consult the document
`niftiplugin_design_v05.pdf`.

1.3.2 Changes in the documentation

Version 1.2
Updated most of the information and screenshots to the NIfTI-converter for BrainVoyager QX 2.1.

Version 0.9
Added description for importing coregistered files.

Version 0.8
Changed the documentation for the plugin for BrainVoyager QX 2.0.

Version 0.7
Added a case for importing data from NeuroLens (no pretty pictures, though). Added a section on sparse data (VOIs).

Version 0.6
Added importing *.vmp files in statistical section, added a case for importing data from GingerALE.

Version 0.5
Added logo, positioning computation script and modified the sections about *.dmr files. Added information about GIfTI export plugin. Added mapping of statistical codes for BrainVoyager to NIfTI (section 6.2.1).
Applicable to BrainVoyager QX version: 1.9 and 1.10
Applicable to plugin version: 1.05.070808

Version 0.4
Added information about the positioning and transformation functions, and enriched the document with some text.

Version 0.3
Added import of *.dmr files, a description for converting a batch of files and the case for importing a batch of eddy current corrected files.

Version 0.2
Added import- and export cases and menus.

1.4 Installation

The plugin should be activated after performing the following steps:

1. First, download the nifticonverter from the BrainVoyager wiki at: `http://support.brainvoyager.com/` → Available Tools → Available Plugins.
2. Unpack the *.zip file and place the *.dll, *.dylib or *.so in the folder:

   Windows: `/My Documents/BVQXExtensions/Plugins_32or64/`

   Mac OS X: `/~/Documents/BVQXExtensions/Plugins_32or64/` (see figure 1.1)

   Linux: `/~/Documents/BVQXExtensions/Plugins_32or64/`
3. Start BrainVoyager QX 2.1.

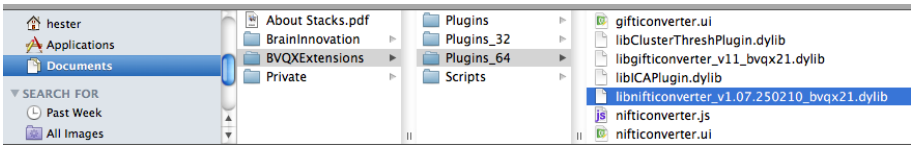


Figure 1.1: Location of plugin on Mac OS 10.6

1.5 Main menu

Menu options

The main menu is depicted in figure 1.2.

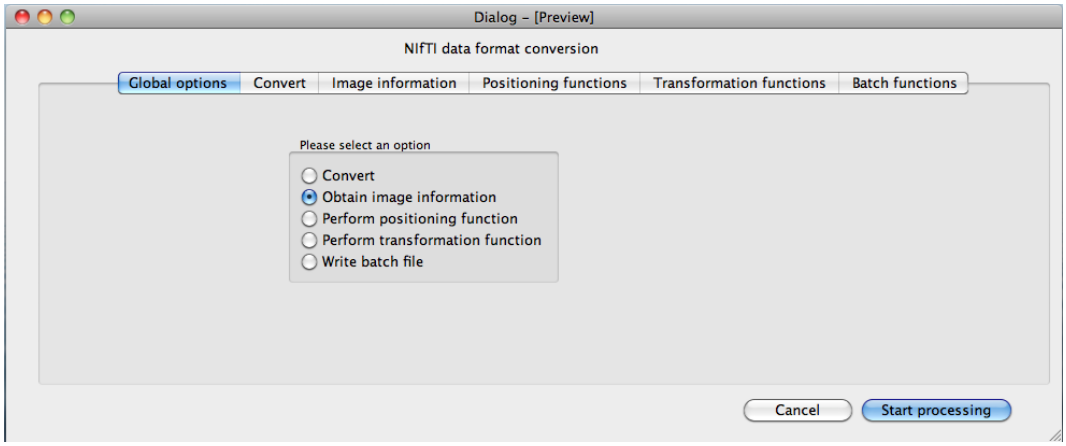


Figure 1.2: Main menu

The options that are available in the main menu are described below.

Print header: This menu option is included to provide the information about a file that is saved in its header. For a description of the available functions, see chapter 10.2.

Convert: Converts one or several files from one file format to another. For a description of the available functions, see chapter 2.

Positioning: The 'Positioning' menu option takes care of providing information about the position of the image in a coordinate system. For a description of the available functions, see chapter 11.

Transformations: Provide information about transformations that have been applied to an image file. In future, this might contain functions to apply a transformation to an arbitrary image file. For a description of the available functions, see chapter B.

Batch functions: Contains a function to create a text file with NIfTI files for batch conversion.

1.6 Acknowledgements

Thanks to Emma Duerden and Jean-Michel Hupé who were most helpful and kind. To Joost Mulders for advice on the positioning information. To Philippe Thevenaz for use of his affine transformation algorithm [6]. To Rainer Goebel for adapting the BrainVoyager QX plugin access functions for retrieving and updating positioning and transformation information.

Chapter 2

Conversion functions

2.1 Conversion menu

Converts one or several files from one file format to another (see figure 2.1).

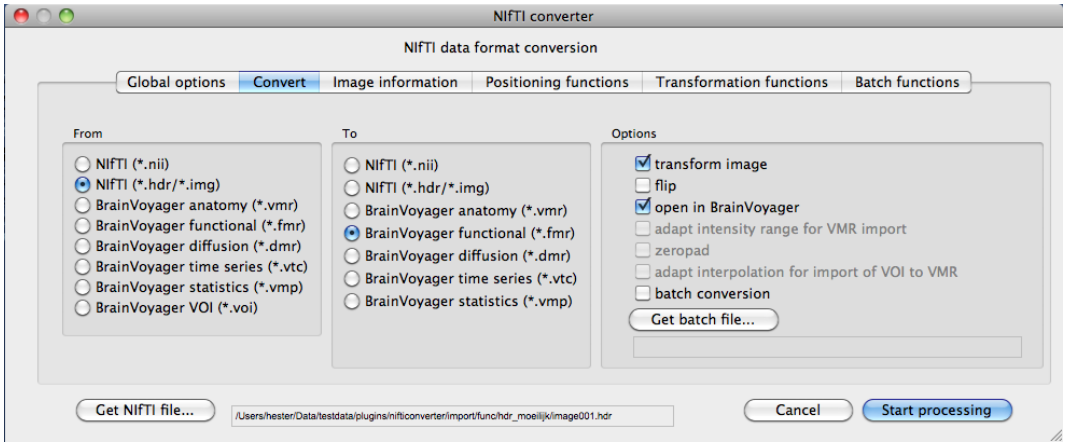


Figure 2.1: Conversion menu

2.2 Converting one file

When converting a file, the image is reoriented to its new coordinate system, except when the image matrix is not square (for example 96 x 104). The transformation is also saved in the header of the destination format, in case there is a field for the history of transformations. This is the case for BrainVoyager anatomical files (*.vmr) and the NIfTI-1.1 *sform* header fields.

For converting anatomical images, see chapter 3.
A description for converting functional images can be found in chapter 4.
The conversion of diffusion-weighted data is described in chapter 5.
For converting statistical data, please consult chapter 6.

Examples for converting from different software packages can be found in the "Conversion cases" chapter 8.

2.3 Explanation of options

2.3.1 Transform

If the option "Transform" is checked, the image will be oriented using the most suitable of the transformation matrices (see section B.2) so that the x, y and z-axes of the image are the same as of the coordinate system. However, if the image happens to be in one of the less likely orientations so that the transform option does not help, one option is to try the conversion without this option checked, another is to transform using a self-specified matrix (see section B) or to use the transformation options on the 3D Volume Tools (only available for VMR and VMP).

2.3.2 Flip

The option "Flip" makes it possible to transform to a mirror image via a reflection in the x, y or z-axis. This option can help as solution for the left-right or radiological vs. neurological convention issue. It is also possible to use the flip option by specifying a matrix that flips the image on the transform tab (see section B). Dependent of the type of image, flipping will have the effect of interchanging left and right, superior (top) and inferior (bottom) or anterior (front) and posterior (back).

2.3.3 Open in BrainVoyager QX

If the option "Open in BVQX" is checked, the image will be shown in BrainVoyager as soon as it has been converted from NIfTI format. This option is available for anatomical (VMR), functional (FMR) and diffusion images (DMR).

2.3.4 Zeropad

If the option "Zeropad" is checked, the space outside its boundaries but within $256 \times 256 \times 256$ will be filled with zeroes, which is black. This option is available for statistical volume maps (VMP). Because VMP files are saved as a subvolume of VMR files, the "zeropad" option is a solution to make them fit on the anatomical file after export to NIfTI format.

2.3.5 Adapt intensity for VMR import

The option "Adapt intensity for VMP import" will linearly transform the range of intensity values to 0-255, which is the maximum range that fits in the `char` datatype, which is used to save VMR data. Please note that this option is not required anymore when the VMR data format changes to using 16 or 32-bits integer or 32-bits float.

2.3.6 Adapt interpolation for import of VOI to VMR

If the option "Adapt interpolation for import of VOI to VMR" is checked, a nearest neighbour interpolation will be applied so that similar values stay the same.

2.4 Converting a list of files

2.4.1 Creating a text file with the list of files to be converted

Manually

First, create a plain text document (*.txt). In this document, the number of files that need to be converted is mentioned first. Then, each file is listed with on the next line the destination format for that file. The name of the file should be complete, including directories and subdirectories. On the Mac, the name will start with `/Users/<home>/...` or the 'home' sign `~/`. On Windows, the name will start with the name of the disk drive and the (sub)directories follow, for example `C:/Hester/data/dti/pp1.hdr`. For reasons of convenience, it is better to write the file separator as a forward slash '/' than a backslash \. But in theory both ways of writing should work.
This "destination format" is the extension of the future file format. In figure 2.2, an example is shown where two Analyze 7.5 *.hdr/*.img pairs are converted to the BrainVoyager QX diffusion weighed format *.dmr/*.dwi. For the destination format is written `dmr` (without dot).
The format of the conversion batch files is the following:

```
<number of files n>
<file 1>
<destination format 1>
<file 2>
<destination format 2>
...
<file n>
<destination format n>
```

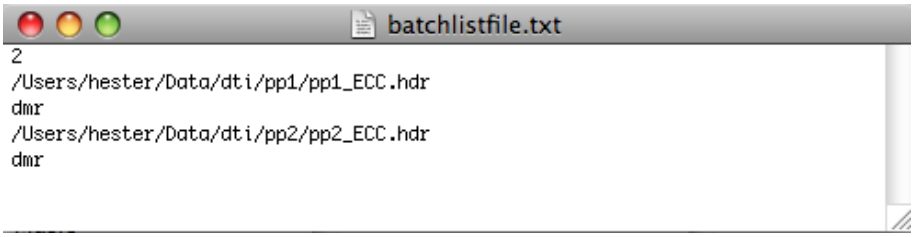


Figure 2.2: An example batch file for importing two Analyze 7.5 images to the BrainVoyager *.dmr format

Other destination files could be `fmr` or `vmr`. Currently, the batch processing option only works for files that do not need to be open in BrainVoyager QX. In future, the type of files that can be batch-processed will be extended.

Automated

To create a batch file via the NIfTI-1 converter plugin, select option "Batch functions" from the main menu (see section 1.5). This will result in the presentation of a file dialog, so that files can be selected until "Cancel" has been pressed.
In case the batch file is used for import of coregistered files (see section 2.5), please note that it is not possible to use the same anatomical file twice in the list, for technical reasons. (This file name can be added to the text file manually afterwards, if it needs to be re-used).

2.4.2 Run the conversions

Start the NIfTI-converter from the BrainVoyager QX 'Plugins' menu. Select the option 'Convert'. From the 'Convert' tab, select the 'Convert batch of files' checkbox.

For the batch conversion, click the "Get batch file..." button to select the plain text file containing the names of the files that should be converted (see figure 2.3). This text file can be stored at any location.

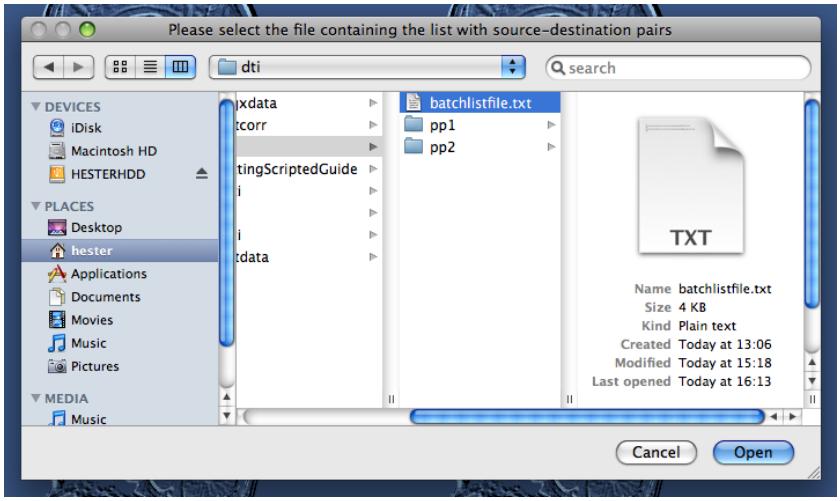


Figure 2.3: Selecting the text file containing the file list

First the batch file is read (see figure 2.4).

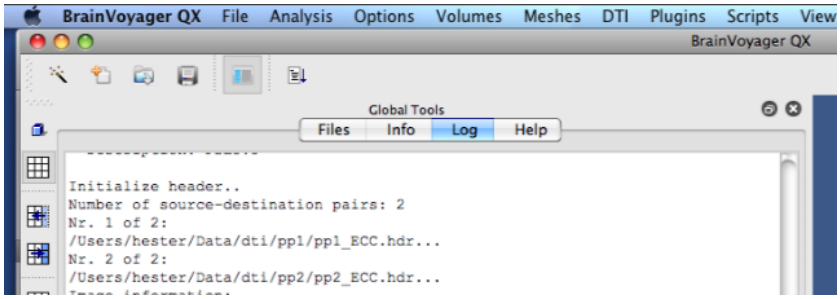


Figure 2.4: Logging the reading of the batch file

After the text file has been read, the files will be converted. The process can be followed via the BrainVoyager QX Log tab (see figure 2.5).

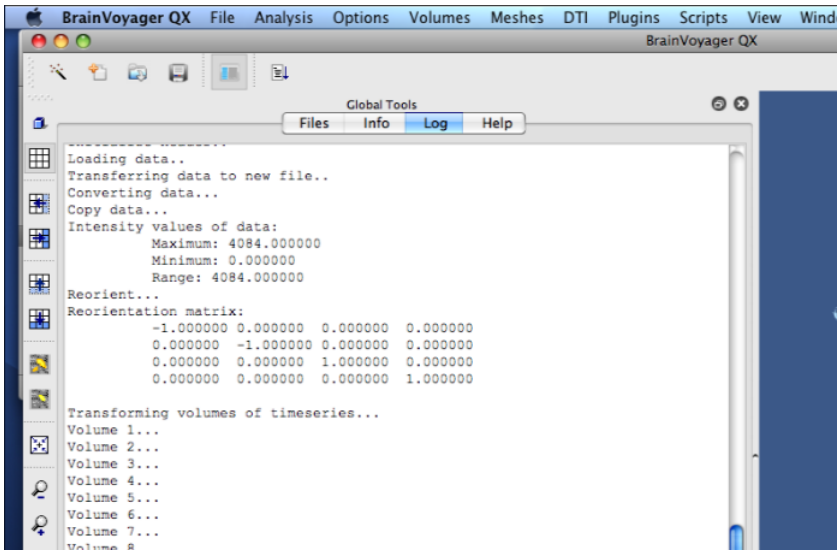


Figure 2.5: Following the conversion process via the BrainVoyager QX Log tab

2.5 Converting coregistered files

The functions to import coregistered anatomical and functional data need to be updated and are therefore not included in this documentation. The procedure to create a text file with a list of file names has been described above (see section 2.4). In the first version(s) of the NIfTI converter for BrainVoyager QX 2.1 and higher this function are not yet available, since the automatic script generation needs to be updated for this feature.

Chapter 3

Converting anatomical data

3.1 Illustration of the process

3.1.1 Import to *.vmr

The import of anatomical data to BrainVoyager *.vmr files has been shown in figures 3.1 and 3.2. Please note that it is also possible to import coregistered anatomical and functional data simultaneously since NIfTI converter version 1.06.080709. This has been described in section **??**.

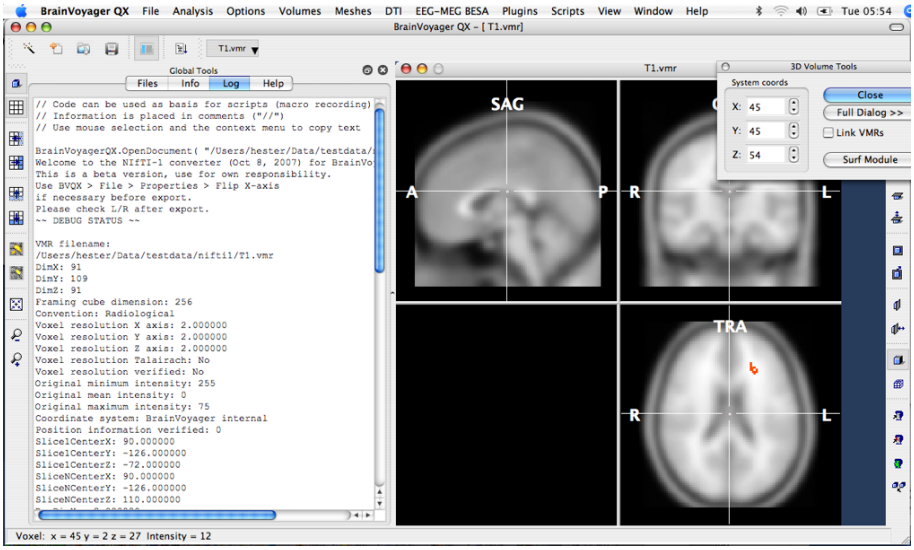


Figure 3.1: Importing an anatomical NIfTI-1 image to a VMR file

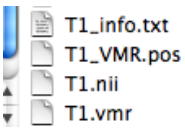


Figure 3.2: Resulting files when importing an anatomical NIfTI-1 image to a VMR file

Image orientation

If the image is imported from NIfTI-1 to BrainVoyager anatomical format (*.vmr), no image transformations are applied when the image is in sagittal orientation (see also section 9.1.6). When the image's x-axis is left $\rightarrow$ right, the image is transformed to sagittal. The information is stored in the positioning header. The image transformation matrix (actually it's inverse) is stored in the transformation header, so that is clear what has happened to the image.
When the image appears to be in neurological convention, or left-handed, indicated by the "qfac" being -1, it can be flipped using the "Flip" option. See also section 9.1.4.

Image positioning

If there is positioning information present in the NIfTI-1 image[1], this will be converted to BrainVoyager positioning information. The positioning information in a *.vmr file can be found via the 'POS info...' button on the VMR Properties dialog of the BrainVoyager File menu (see figure 3.3).

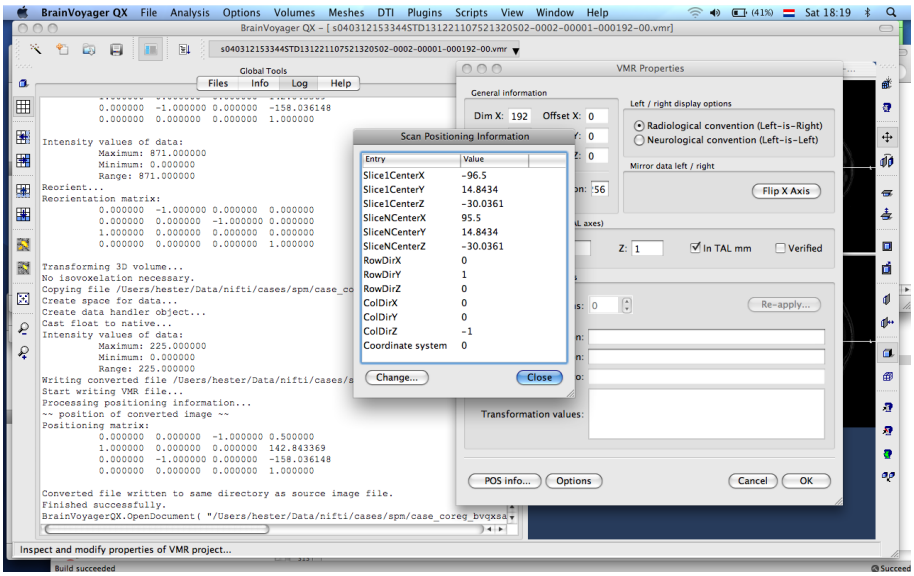


Figure 3.3: Importing an anatomical NIfTI-1 image to a VMR file: positioning information

[1]Although NIfTI-1.1 images also can have the *.hdr/*.img file extension, their predecessor *.hdr/*.img forms of Analyze 7.5 do not contain positioning information. The positioning information can be checked via another option in the NIfTI converter plugin, see section 10.2)

Scaling

Because the data of a standard anatomical file in BrainVoyager (*.vmr) are saved in 1 byte per voxel, and only the first 225 values are used for gray scale intensity values, the values in the NIfTI file are scaled via a linear transformation as described in [5, Ch.6] according to the following procedure:

```
float range = max + fabs(min);
float newrange = newmax + fabs(newmin); // should be 225.0 in case of vmr's
float p = (newrange/range);
float q = newmin - p * min;
```

where `range` is the intensity value range in the NIfTI data, `newmin` is 0 and `newmax` is 225[2].
The scaling parameters in the NIfTI header are currently not yet taken into account.

[2]The maximum is not 255 because the value range $226 - 255$ in the *.vmr file are used for colors.

3.1.2 Export from *.vmr

When an anatomical BrainVoyager file (*.vmr) is exported, the converter will (currently) assume that the image is in sagittal orientation. It will apply a default transformation so that the x-axis of the image is in left-right direction, the y-axis is along the anterior-posterior line and the z-axis in the inferior-superior direction. This is performed so that the image will be shown in the default display mode in FSL 3.2.

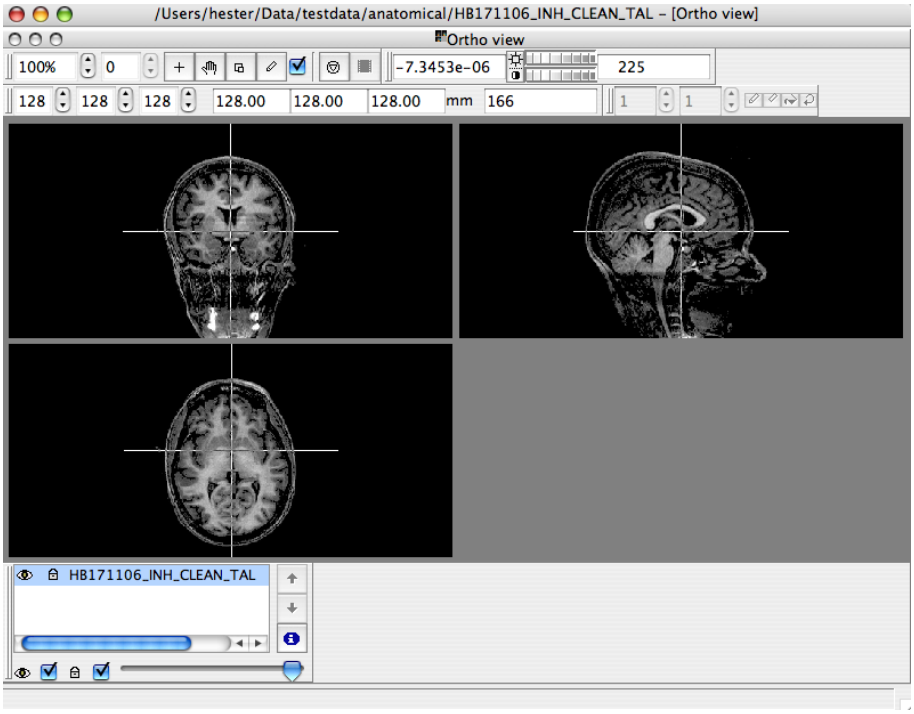


Figure 3.4: Exporting a VMR file to a NIfTI-1 anatomical image

3.2 Possibilities and limitations

Table 3.1: VMR: cases accepted by the converter

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size	any	should be any
read/write	yes	yes
orientation	any	any (mostly sagittal)
positioning info	yes	not yet
transformation info	yes	not yet
voxel resolution	any	any
datatype	any to char	exports to float
file version	to VMR v.3	any
coordinate system	NIfTI	BV internal, system or Talairach
range of intensity values	any to 0-255	
byte order	little endian	little endian

Table 3.2: VMR: cases which the converter does not accept or have not yet been tested

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size	512 x 512 x 28	
read/write		
orientation		
positioning info		
transformation info		
voxel resolution	slice gap	
datatype		
file version		
coordinate system		
range of intensity values		
byte order		

3.3 Anatomical file formats in BrainVoyager

The file formats to save anatomical data in BrainVoyager QX are:

VMR: Voxel-based anatomical data (T1).

SRF: Triangulated mesh.
(This is convertible via the GIfTI converter, see `http://support.brainvoyager.com/`.)

See also figure C.1.

Chapter 4

Converting functional data

In BrainVoyager, there are two file types containing functional data.
The FMR project consists of one text file (*.fmr) containing the image information, with the image data in one or several binary files (*.stc). An *.fmr file can be opened in BrainVoyager directly.
Normalized functional data are saved in the volume time course (*.vtc) file format[1]. This filetype can only be opened when an anatomical project (VMR project) is open in BrainVoyager.

4.1 Illustration of the process

4.1.1 FMR files

Import to *.fmr

If the image is imported from NIfTI-1 to BrainVoyager functional format (*.fmr), a small image transformation is applied when the image is in coronal orientation (see also section 9.1.6). The information is stored in the positioning header.
When the image appears to be in neurological convention, or left-handed, it can be flipped without recording it in the position matrix (see sections 2.3.2 and 9.1.4.
Some fields in the header file might not be filled properly (check after import for example the slice gap in the *.fmr file in a text editor, and correct this when necessary). The data should be converted properly, however (see figure 4.1).

If there is positioning information present in the NIfTI-1 image[2], this will be converted to BrainVoyager positioning information. The positioning information in a *.fmr file can be found via the 'POS info...' button on the FMR Properties dialog of the BrainVoyager File menu (see figure 4.2), or by opening the *.fmr file in a text editor (see figure 4.3). If the positioning information is not correct, it can be changed in these ways.
To compute the positioning information, see section 9.1.5.

[1] In SPM, these data would be saved in a resliced functional file (rf*.nii)

[2] Although NIfTI-1.1 images also can have the *.hdr/*.img file extension, their predecessor *.hdr/*.img forms of Analyze 7.5 do not contain positioning information. The positioning information can be checked via another option in the NIfTI converter plugin, see section 10.2)

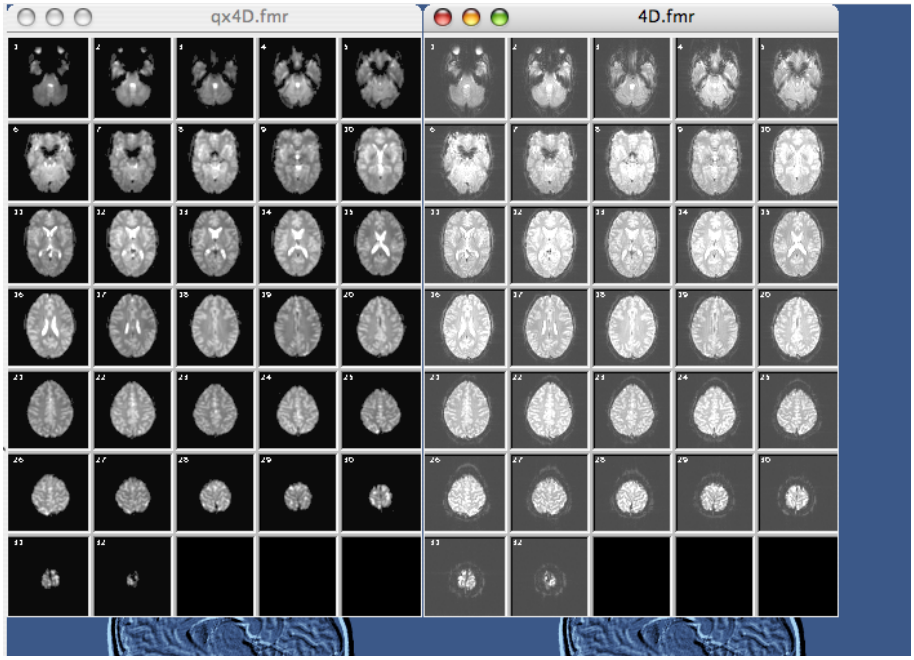


Figure 4.1: Direct DICOM import in BrainVoyager QX (left) and importing a NIfTI-1 functional image to an FMR file (right)

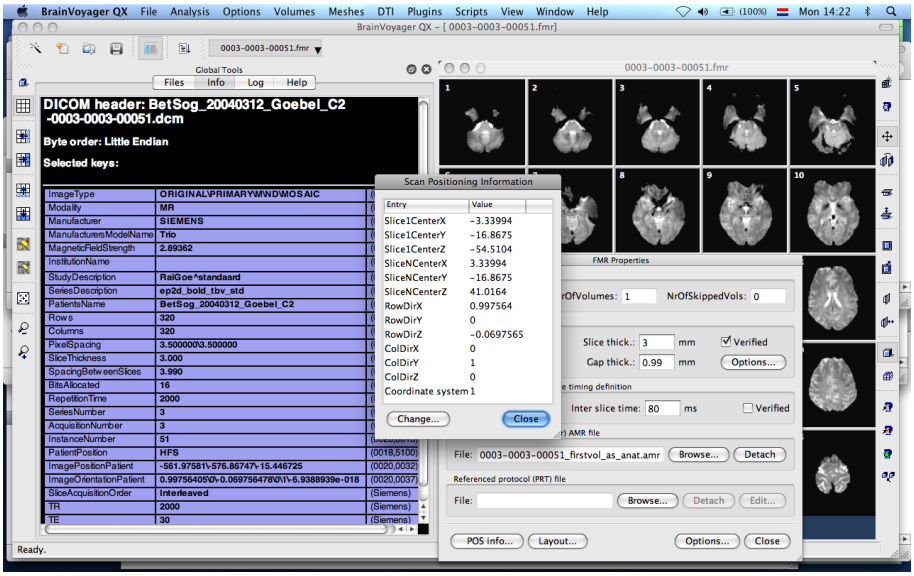


Figure 4.2: Positioning information of the third functional file in the BVQX Sample Data as generated by BrainVoyager QX

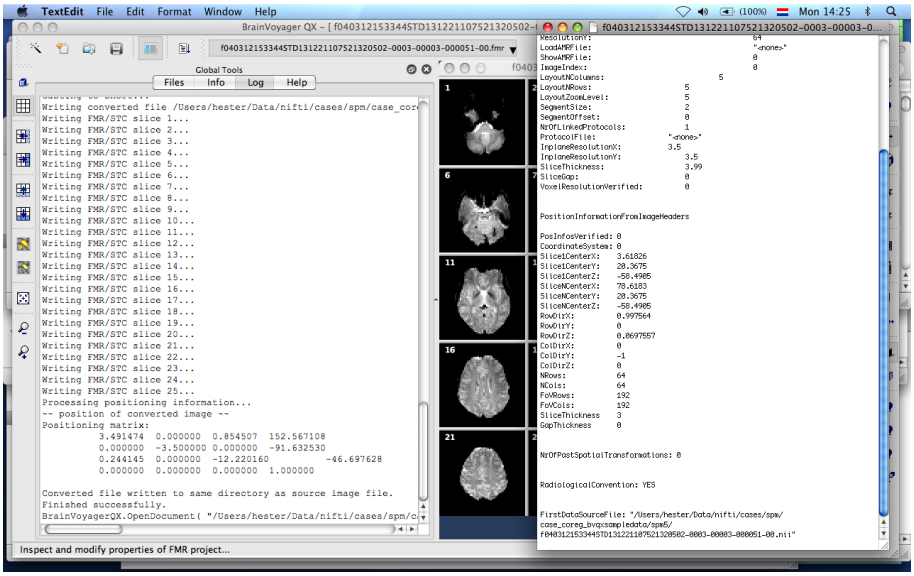


Figure 4.3: Positioning information of the third functional file in the BVQX Sample Data as generated by the NIfTI-1 converter.

Export from *.fmr

When exporting an FMR project, the header information in the *.fmr file and the data in one or several *.stc files are converted to one NIfTI-1 file (*.nii) in the same directory as where the original files are located. This file can now be opened in AFNI, SPM (see figure 4.4), FSL or another NIfTI compliant software package (for example MRIcron).

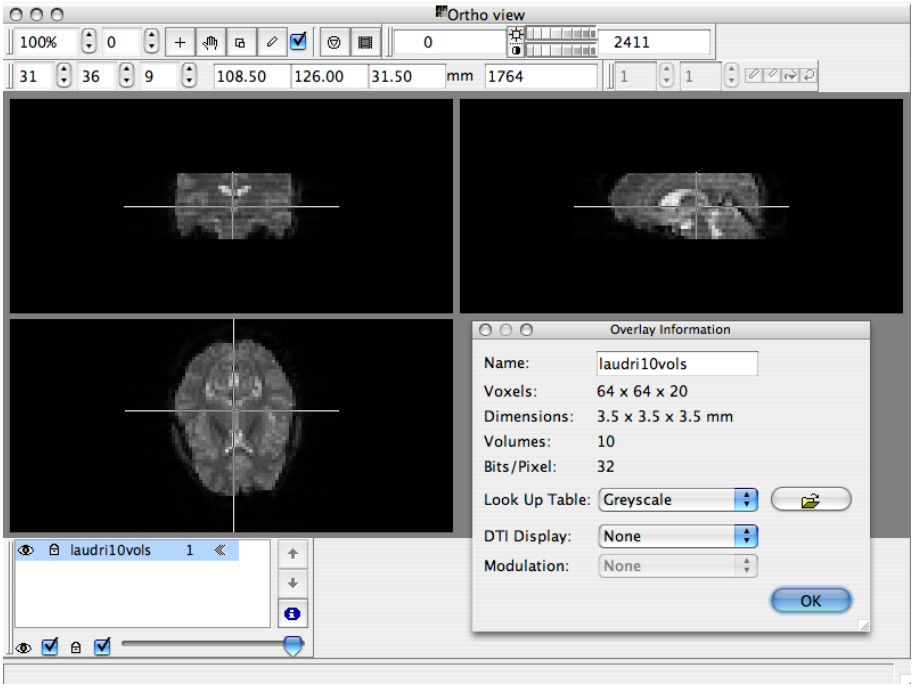


Figure 4.4: Exporting an FMR file to a NIfTI-1 functional image

4.1.2 VTC files

Import to *.vtc

When importing functional data to a *.vtc file, an anatomical file (*.vmr) needs to opened in BrainVoyager at the moment of conversion.
Before saving the *.vtc file to disk, the sequence of volumes is permuted to a sequence of voxel time courses, so that one does not find first all voxels of the first volume and then the second etc, but first all timepoints of the first voxel, then all timepoints of the second, etc.
The image is currently stored in the center of the *.vmr image.
The image can be visualized (see figure 4.6) using the "Show VTC volume" function on the "Spatial Transformations" tab of the 3D Volume Tools dialog.

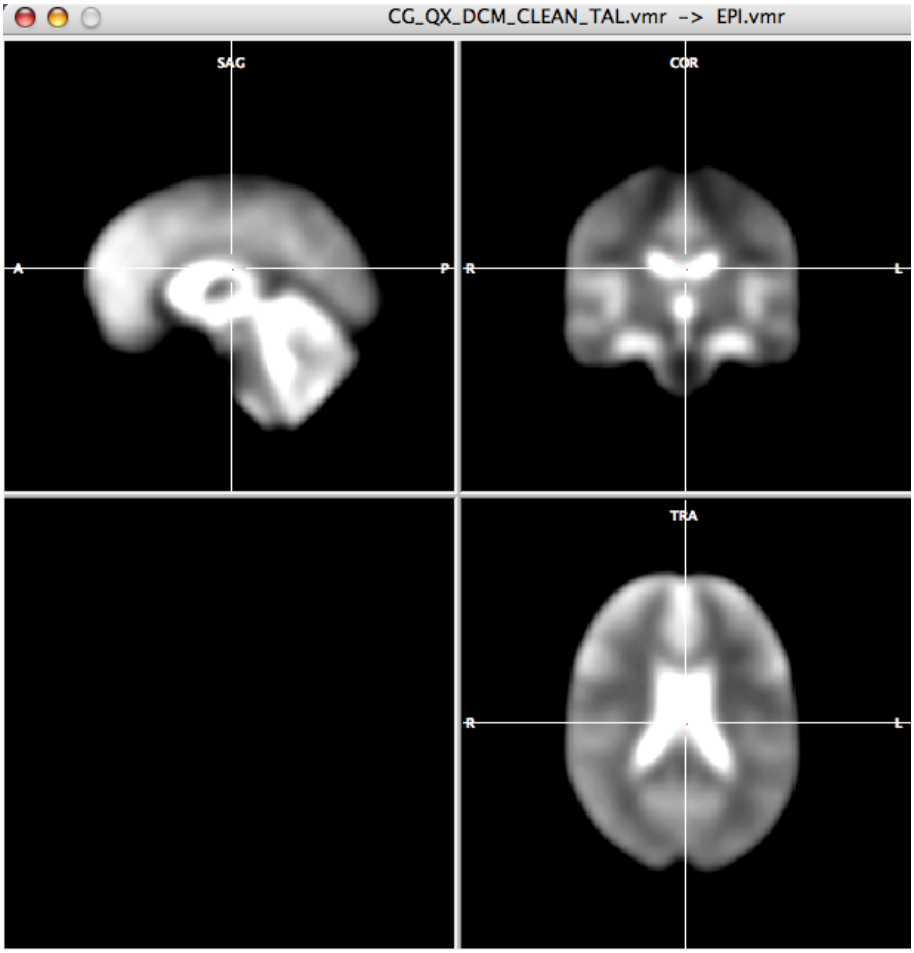


Figure 4.5: Importing a a NIfTI-1 functional image to a VTC file

If there is positioning information present in the NIfTI-1 image[3], this will be used to compute the bounding box parameters in a *.vtc file.

Export from *.vtc When one would like to export a *.vtc file to the NIfTI-1 file format, this *.vtc file should be open in BrainVoyager. This can be performed by

[3]Although NIfTI-1.1 images also can have the *.hdr/*.img file extension, their predecessor *.hdr/*.img forms of Analyze 7.5 do not contain positioning information. The positioning information can be checked via another option in the NIfTI converter plugin, see section 10.2)

opening an anatomical file (*.vmr) and linking the *.vtc via the "Link 3D Time Course (VTC) File" function of the BrainVoyager "Analysis" menu.
This file can now be opened in AFNI, SPM, FSL (see figure 4.6), or another NIfTI compliant software package (for example MRIcron).

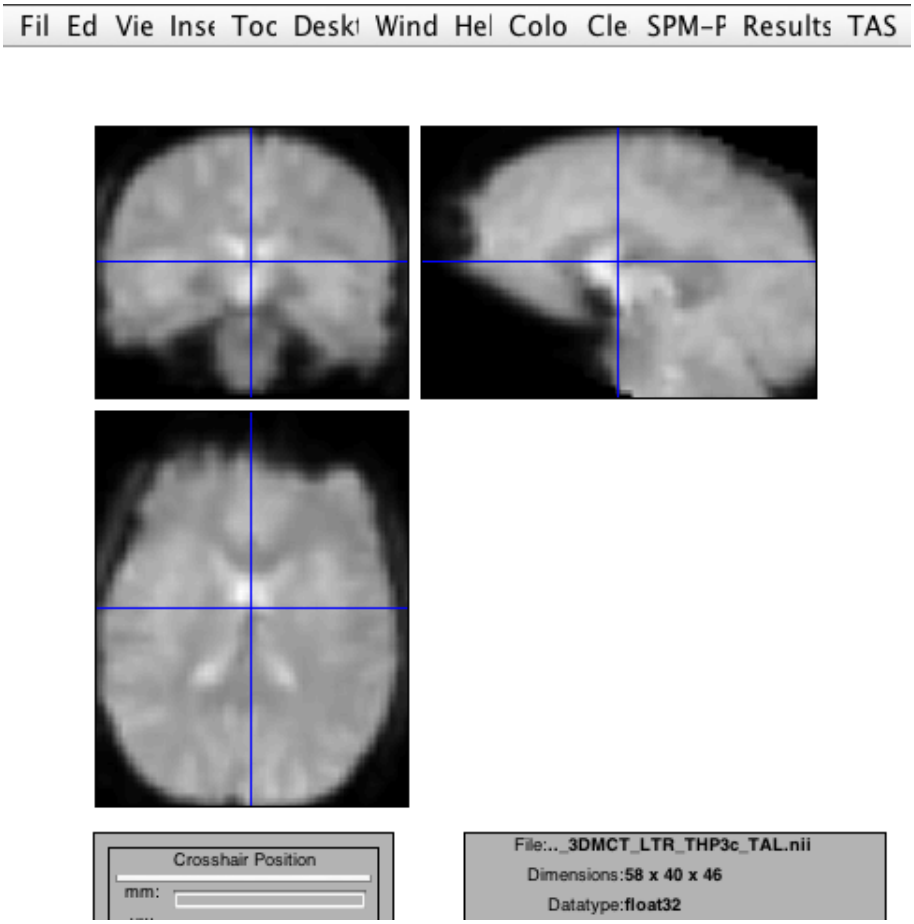


Figure 4.6: Exporting a VTC file to a NIfTI-1 functional image

4.2 Possibilities and limitations

4.2.1 FMR files

Table 4.1: FMR: possibilities and limitations

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size		any
read/write	yes	yes
positioning info	yes	not yet
transformation info		not yet
voxel resolution	any	any
permutations in space-time	expects volumes	expects stacks of slices
datatype	to 16-bit integer (QX 1.10), to float (QX 2.0)	integer or float to float
file version	version 6	any
coordinate system		
range of intensity values	any (can be scaled)	any
byte order	little and big endian	little endian

4.2.2 VTC files

Table 4.2: VTC: possibilities and limitations

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size		any
read/write	yes	yes
positioning info	no	no
transformation info	no	no
voxel resolution		any
permutations in space-time	expects volumes	expects timecourses
datatype	any	short
file version		any
coordinate system		
range of intensity values		
byte swap	no	no

4.3 Functional file formats in BrainVoyager

The file formats to save anatomical data in BrainVoyager QX are:

FMR: Description of the functional data.

STC: Functional data (slice time courses). In previous versions, the files were written per slice time course, from FMR version 5 onwards they are saved in one *.stc file.

VTC: Normalized functional data (volume time courses)

MTC: Functional data on vertices of mesh (mesh time courses)(use GIFTI plugin in BrainVoyager QX 2.0 or higher)

See also figure C.1.

Chapter 5

Converting diffusion-weighted data

5.1 Illustration of the process

5.1.1 Importing a *.nii file to *.dmr/*.dwi

For importing NIfTI-1 *.nii files, select as source file the *.nii file. The destination file format should then be *.dmr.*Note: due to file system limitations, big files (larger than about 79 volumes x 85 slices) might not be convertible on Windows XP (no problems on Mac OS X).*

In the import function for *.dmr files, it is possible to use the option "Open in BrainVoyager" (see section 2.3.3) to let the converter open the converted file (see figure 5.1). Check the positioning information in the *.dmr file (this is a text file). If necessary, see section 9.1.5 for computing the positioning information.

See also section 8.2.

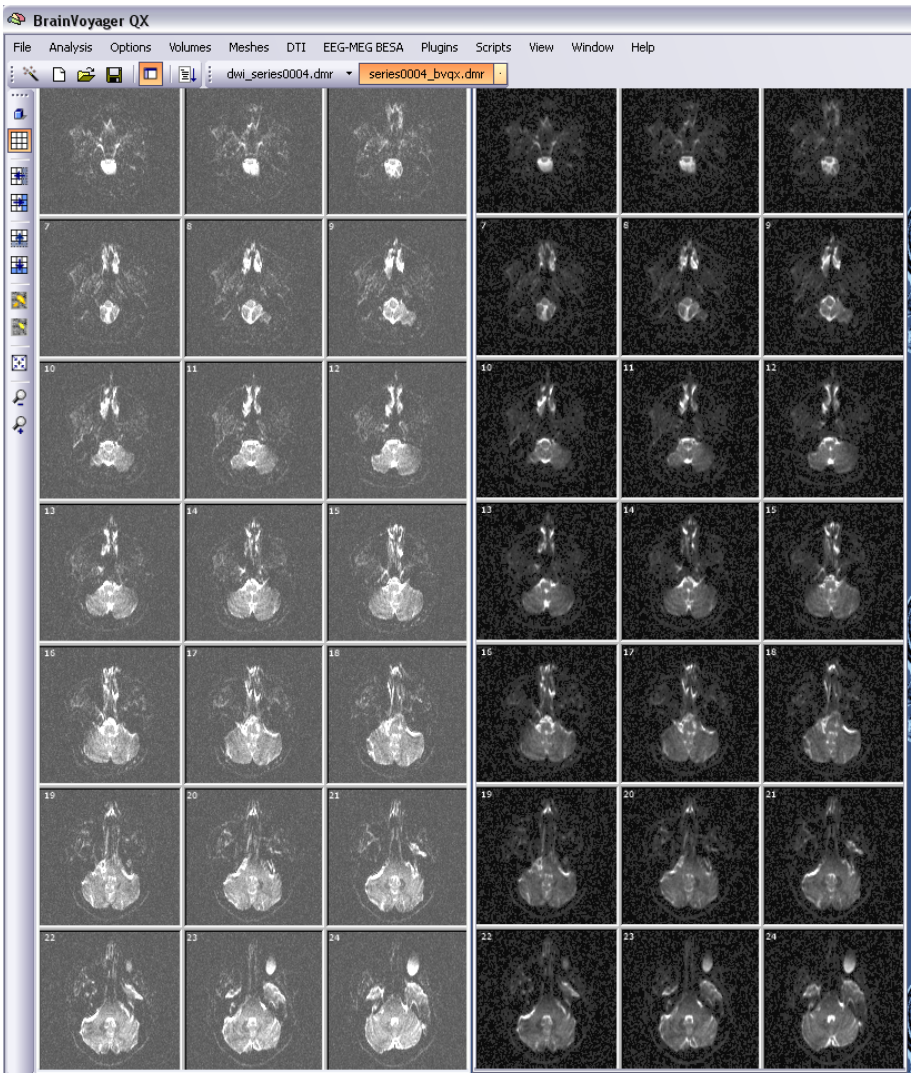


Figure 5.1: Importing a NIfTI-1 image to DMR project

5.1.2 Exporting a *.dmr project to NIfTI-1

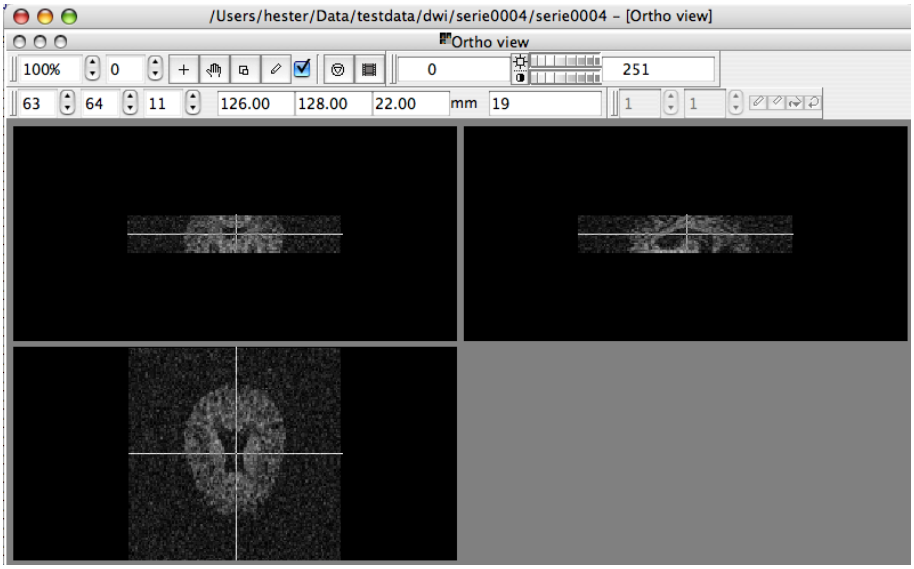


Figure 5.2: Exporting a DMR file to a NIfTI-1 image

5.2 Possibilities and limitations

Table 5.1: DMR: cases which the converter accepts (info needs to be updated)

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size		128 x 128 x 23
read/write		yes
orientation		
positioning info		
transformation info		
voxel resolution		2 x 2 x 2
datatype		short
file version		1
coordinate system		
range of intensity values		
byte order		LE

Table 5.2: DMR: cases which the converter currently does not accept or have not yet been tested

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size		
read/write		
orientation		
positioning info		
transformation info		
voxel resolution		
datatype		
file version		
coordinate system		
range of intensity values		
byte order		

5.3 Diffusion-weighted file formats in BrainVoyager

The file formats to save diffusion-weighted data in BrainVoyager QX are:

DMR: Description of diffusion-weighed data.

DWI: Diffusion-weighted data (diffusion weighted images).

VDW: Normalized diffusion-weighted data (volume diffusion weighted)

See also figure C.1.

Chapter 6

Converting statistical data

6.1 Illustration of the process

6.1.1 Import to *.vmp

When importing to statistical map (*.vmp) for anatomical volumes (*.vmr), the file is imported as NR-VMP, so that also the new map types, like Z-scores and beta values, can be used.

Map orientation
The map will be reoriented to sagittal orientation, for display on anatomical images (*.vmr).

Map positioning
The map will be zeropadded to 256 x 256 x 256, and interpolated to 1 x 1 x 1mm, like anatomical images. After zero-padding, the image will be translated to the center. Importing maps will require a bit of manual labor afterwards to get your map in the proper place[1]. A description of repositioning your images can be found in section 8.4.

Procedure
When importing, select '*.nii' as source and as destination the '*.vmp' type. The converter will automatically isovoxel, center and reorient the image. If the map type is recognized, it is automatically set in the *.vmp header. For a list of recognized map types, please consult the table below.
If the `NIFTI_INTENT` code is not convertible to a BrainVoyager statistical map type, a dialog is presented which enables the user to select a map type (see figure 6.1).
The map is transformed to the center of the image (see figure 6.2). In this case an imported map of z-scores from GingerALE is shown.

[1] As the import of maps gained priority over fixing the positioning information

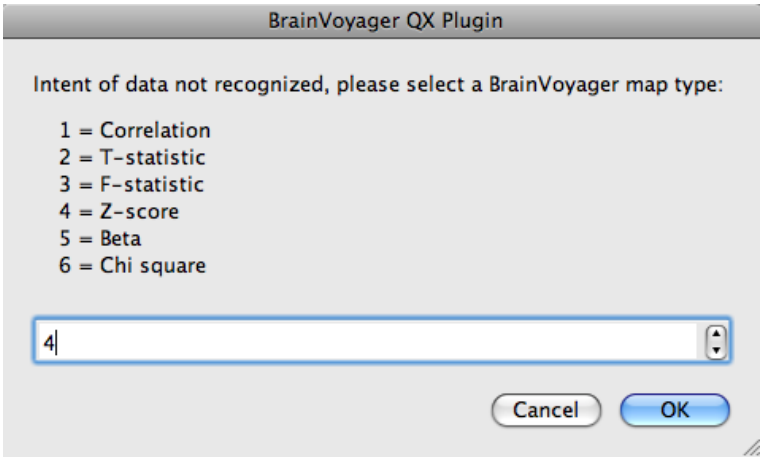


Figure 6.1: Dialog for selecting a map type

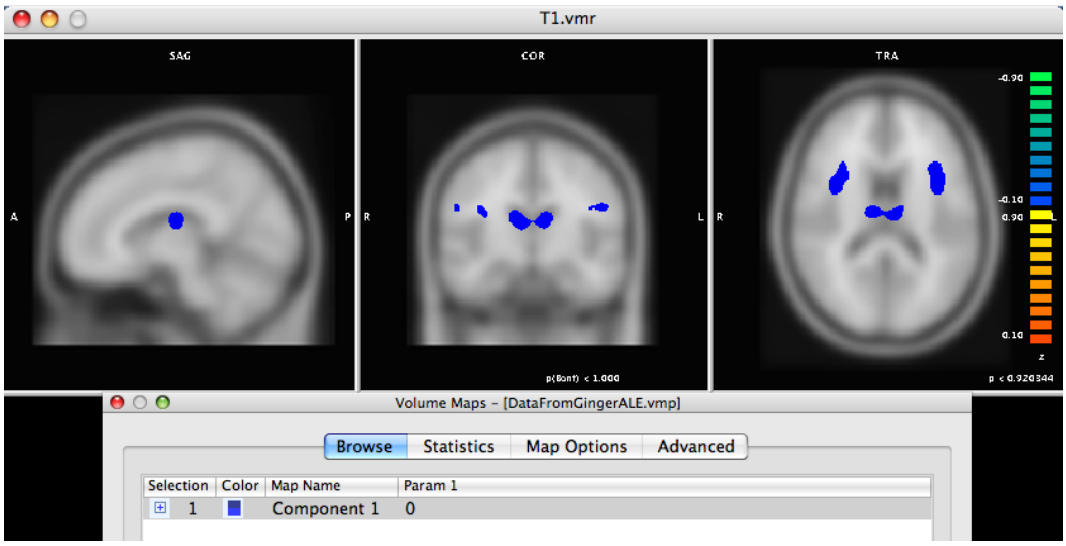


Figure 6.2: Imported statistical map (NR-VMP)

6.1.2 Export from *.vmp

The exported format is AR-VMP. For an illustration of an exported image, please see the figure below. The image is not transformed, rather its orientation is provided in the NIfTI-1 header. This will cause some of the packages to display the image according to its orientation, while others will display the image like it is stored on disk.

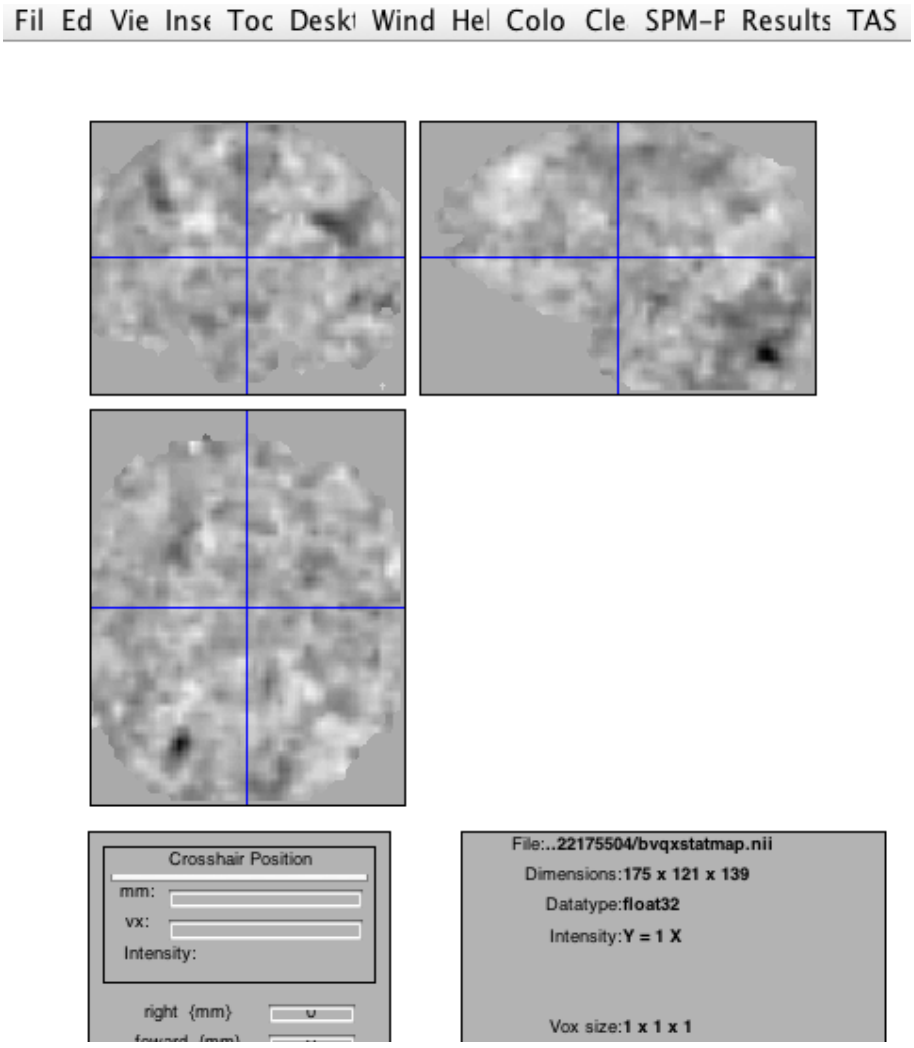


Figure 6.3: Exporting a VMP-AR file to a NIfTI-1 image

6.2 Possibilities and limitations

6.2.1 Recognized map types

The map types that are recognized for import and/or export are displayed in the table below. When exporting to NIfTI, the map type is stored in the `NIFTI_INTENT` header field.

Table 6.1: Recognized statistical maps

Description	*Code in BrainVoyager*	*Description in NIfTI*	*Code in NIfTI*	*Parameters*
T-test	1	`NIFTI_INTENT_TTEST`	3	1
correlation	2	`NIFTI_INTENT_CORRELATION`	2	1
cross-correlation	3	`NIFTI_INTENT_CORRELATION`	2	1
F-test	4	`NIFTI_INTENT_FTEST`	4	2
z-score	5	`NIFTI_INTENT_ZSCORE`	5	0
beta	15	`NIFTI_INTENT_BETA`	7	2
chi-square	14	`NIFTI_INTENT_CHISQ`	6	1

For more information about statistical distributions in NIfTI, see the document "NIfTI-1 Statistical Distributions: Descriptions and Sample C Functions" by Robert W. Cox. This document can be downloaded from `http://nifti.nimh.nih.gov/`.

6.2.2 Parameters

If the particular distribution has parameters like degrees of freedom ν, location/mean μ or first moment (see figure 6.4), scale/standard deviation σ or a noncentrality parameter λ, these will be saved in the intent parameter fields. Any first intent parameter is transferred to $df1$ in a BrainVoyager statistical map, and the second intent parameter to $df2$.

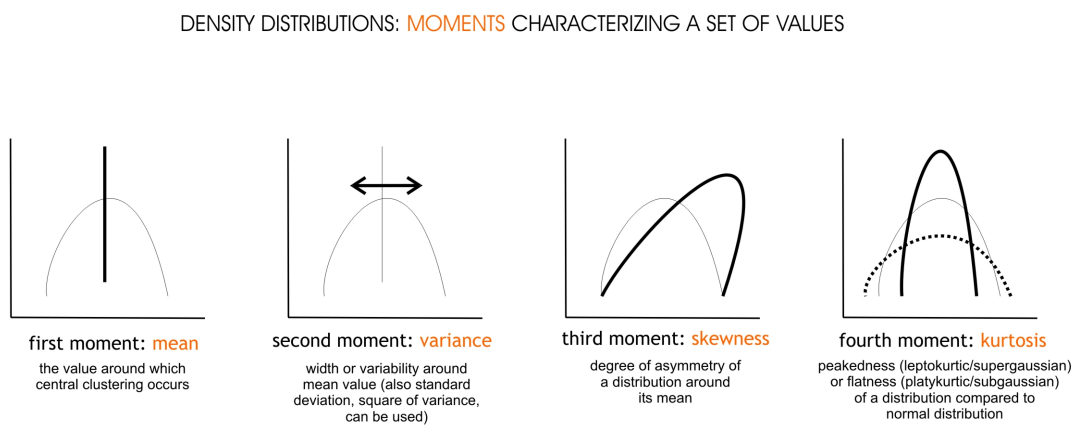


Figure 6.4: Moments of a probability distribution

Table 6.2: VMP: possibilities and limitations

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size	any	any
orientation	standard NIfTI	conforming to VMR (sagittal)
positioning info	no	no
transformation info	no	n/a
voxel resolution	any	1 x 1 x 1
datatype	any	float
file version	-	BVQX 1.9.1-10
coordinate system	NIfTI	-
range of intensity values	any	any
byte order	little endian	little endian
number of maps	1	-

6.2.3 Tables

Statistical file formats in BrainVoyager
The file formats to save statistical data in BrainVoyager QX are:

MAP: Statistical values to be displayed on FMR

VMP-NR: Statistical values to be displayed on VMR (native resolution, f.e. 3x3x3)

VMP-AR: Statistical values to be displayed on VMR (anatomical resolution, 1x1x1)

SMP: Statistical values on vertices of mesh

See also figure C.1.

Chapter 7

Converting sparse data (VOI)

Sparse data are files that only contain coordinates of points that are not empty. In BrainVoyager, this is the case for the region of interest text files. These are region-of-interest (*.roi) files for FMR projects, volume-of-intest (*.voi) files for VMR projects and patch-of-interest (*.poi) files for surface files SRF.
Because the NIfTI data format is designed for volumetric data, the converting *.voi files is discussed in this chapter. *.poi files will in future be convertible via the GIFTI plugin, which is created for converting surface data.

7.1 Illustration of the process

7.1.1 Import to *.voi

This will be implemented in two phases:

1. Import as anatomical volume with special flag to color the areas
2. Import directly to VOI.

7.1.2 Export from *.voi

The VOI file should be loaded in BrainVoyager QX.Each coordinate that is present in the *.voi, will be saved with the same voxel value in the exported NIfTI file. The voxels in the first VOI will get number “1”, the second VOI label “2”, etc. All other voxels will be saved with voxel value “0”.
The resulting file will have the float (32-bit) datatype, like all other exported files. The transformation used is from BrainVoyager system coordinates to NIfTI, see B.2.

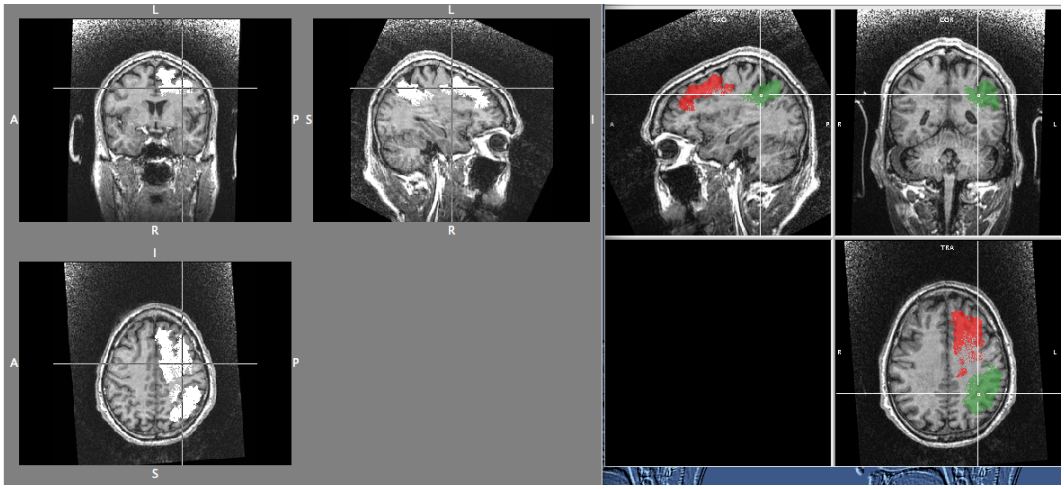


Figure 7.1: An exported VOI in FSLview (left) and the original VOI in BrainVoyager QX 1.10.4 (right)

Labeling

Each VOI will be labeled with an index. The labeling offset is 100. For each next VOI, this number is incremented with 100. The indices are saved in a text file (see figure 7.2).

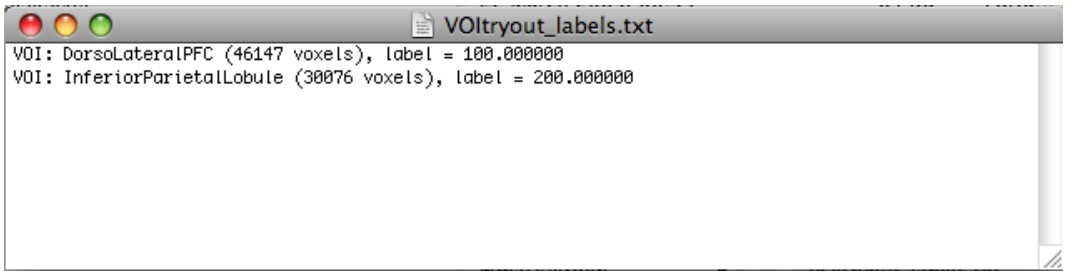


Figure 7.2: The indices for each VOI are saved in a text file

7.2 Possibilities and limitations

Table 7.1: VOI: cases accepted and not accepted by the converter

property	*Import to BVQX*	*Export to NIfTI-1*
matrix size	-	should be any (only one case tested)
read/write	-	yes
orientation	-	should be any (only one case tested)
positioning info	-	not yet
transformation info	-	not yet
voxel resolution	-	similar to VMR resolution
datatype	-	will become float
file version	-	should be any (only one case tested)
coordinate system	-	tested for ACPC and Talairach to NIfTI
range of intensity values	-	n/a
byte order	-	n/a (ascii)

7.3 Sparse file formats in BrainVoyager

The file formats to save anatomical regions in BrainVoyager QX are:

ROI: Slice-based region-of-interest.

VOI: Volume-based region-of-interest.

POI: Patch-based region-of-interest (will be convertible to GIFTI with GIFTI converter)

See also figure C.1.

Chapter 8

Conversion cases

Please find below a number of cases that are included to show how to deal with particular import- and export situations.

8.1 Preprocessing in AFNI, rest in BrainVoyager

From `afni02_to3d.pdf`:

> To save the dataset as a NIfTI (*.nii) file, include the .nii suffix in the prefix name, e.g., `to3d -prefix anat.nii *.dcm`. You can also do this via the user interface.

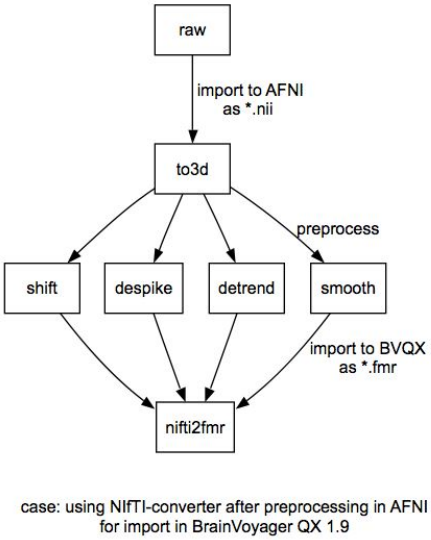


Figure 8.1: Workflow for AFNI preprocessing and importing in BrainVoyager QX

[finish]

8.2 Segmentation in FSL, rest in BrainVoyager

FSL does not import DICOM files, so the NIfTI file can be an export file from the BrainVoyager QX plugin or created with some other software package (Chris Rorden has made a command-line tool for converting DICOM to NIfTI (*.nii), SPM5 has a DICOM import tool).

1. Export the image to NIfTI [insert image here]
2. Apply the segmentation in FSL-FAST (see figure 8.2).

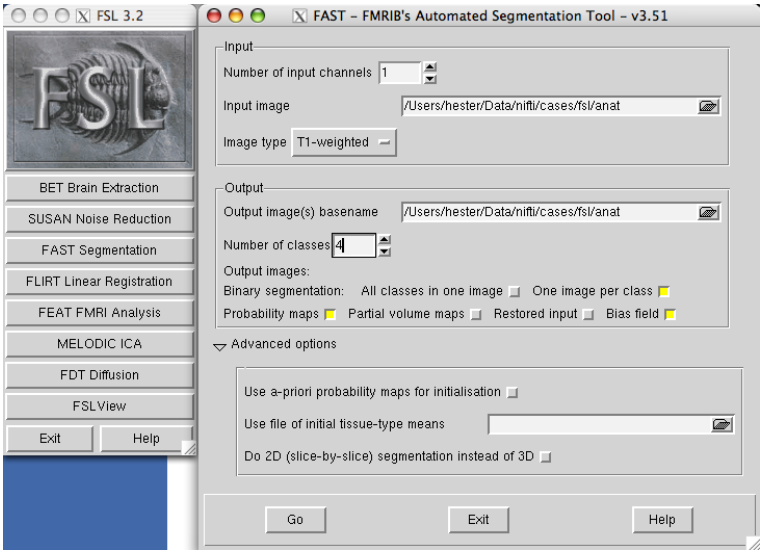


Figure 8.2: The FAST user interface

On Mac OS X, the process is visible via the shell; i.e. this provides the command to run segmentation another time via the command line (see figure 8.3).

```
/Applications/fsl-3.2-macosx/fsl//bin/fast -t1 -c 4 -n -os -op -ob -od /Users/hester/Data/nifti/cases/fsl/anat /Users/hester/Data/nifti/cases/fsl/anat

FAST - FMRIB's Automated Segmentation Tool             Version 3.51

Image: /tmp/fsl_6LYh2P_anat

T1-weighted image
Imagesize : 256 x 256 x 176
Pixelsize : 1 x 1 x 1

Initial K-means segmentation...
8 main iterations ...

Segmentation done successfully!

Calculation time 7160 seconds!

Write bias field image /tmp/fsl_6LYh2P.fastout/QQQQQ_bias.hdr Successfully!
Write segmentation image of tissue 0 Successfully!
Write segmentation image of tissue 1 Successfully!
Write segmentation image of tissue 2 Successfully!
Write segmentation image of tissue 3 Successfully!
Write probability map of tissue 0 Successfully!
Write probability map of tissue 1 Successfully!
Write probability map of tissue 2 Successfully!
Write probability map of tissue 3 Successfully!
Finished
```

Figure 8.3: The FAST user interface

3. Import the image in BrainVoyager QX [insert image here]

8.3 Import a batch of eddy current corrected files from FSL into BrainVoyager

To import a batch of diffusion weighted images that have corrected for eddy currents in FSL, see section 2.4. This case has been used as an example batch processing.
Ask Pim Pullens of the BrainVoyager support team, he wrote a document "Finding inverse coregistration of DTI and T1 using FSL 4.04".

8.4 Importing maps from GingerALE

In this section is described a way to reposition statistical maps; in this example meta-analysis results from GingerALE [7] are used.
First, import your image to *.vmp format according to the procedure described in paragraph 6.1.1.
In its current status, the converter centers the map, which means normalization and registration is not recognized. Because *.vmr and *.vmp files will be zeropadded to 256 x 256 x 256, the following can be done to solve this situation:

1. Import an anatomical image. Open it.
2. Perform VMR → VMR registration to a BrainVoyager sample image in Talairach space (see figure 8.4. This results in a text file with a transformation matrix (*.trf).

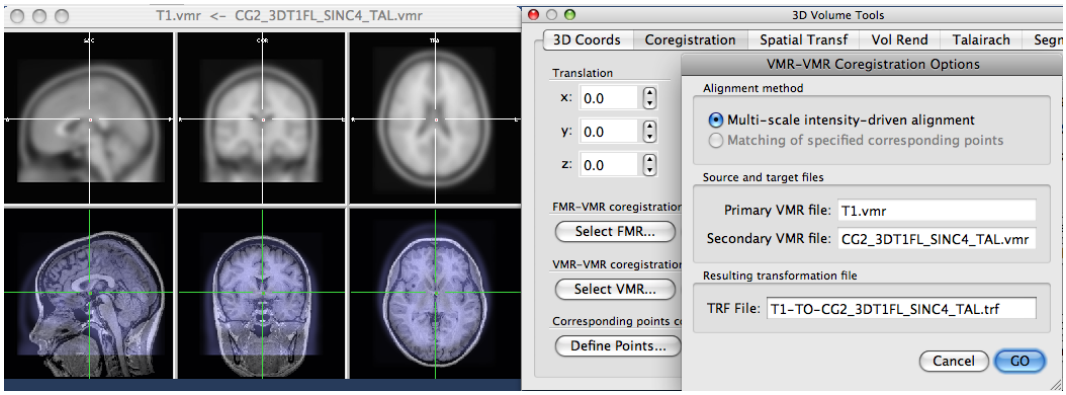


Figure 8.4: Register the imported anatomical file to a BrainVoyager sample file

3. Close the files and open the imported VMR.
4. Via the 'Spatial Transf' tab of the 3D Volume Tools, load the transformation file (*.trf) that was a result of the VMR → VMR registration (see figure 8.5).

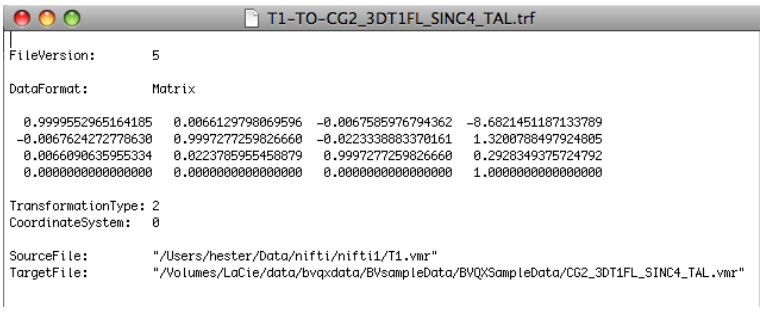


Figure 8.5: BrainVoyager transformation file (*.trf)

5. Via the 'Spatial Transf' tab of the 3D Volume Tools, apply the transformation to the *.vmr.
6. Close the original imported *.vmr file.

7. Load the imported *.vmp and go to the 'Spatial Transf' tab on the 3D Volume Tools dialog.
8. Via the 'Spatial Transf' tab of the 3D Volume Tools, load the transformation file (*.trf) that was a result of the VMR → VMR registration.
9. Via the 'Spatial Transf' tab of the 3D Volume Tools, apply the transformation to the *.vmp.

After the transformation the map will be an anatomical resolution map (AR-VMP). The transformed map is shown in figure 8.6 on the imported anatomical file and the BrainVoyager sample file.

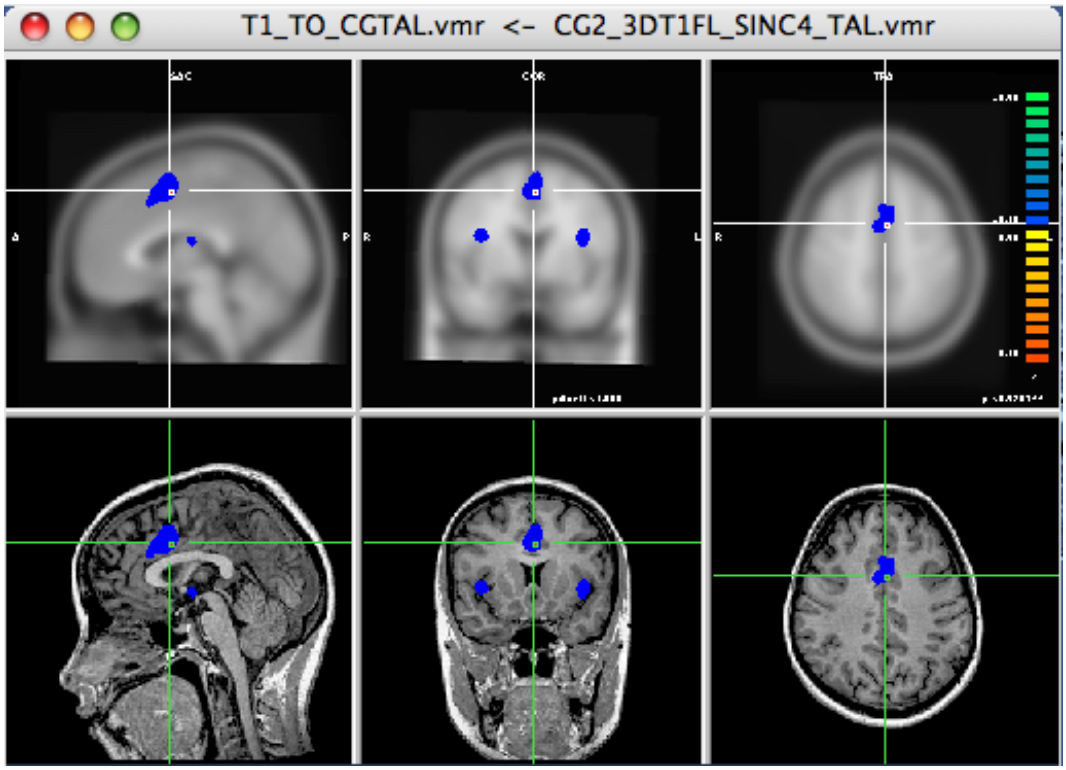


Figure 8.6: The transformed map shown on the imported anatomical file and the BrainVoyager sample file

8.5 Import data from NeuroLens via FreeSurfer into BrainVoyager

Data from NeuroLens are in the 'MINC' dataformat from the Montréal Neurological Institute (MNI). These files have the "*.mnc" extension and are based on the advanced HDF5 dataformat. To import these data, one can use FreeSurfer.
The data might not contain position information. To check which fields are in the header, use option 2 of the nifticonverter: "2 = Obtain image information". The header information can be printed to the BrainVoyager QX Log tab and/or to a text file.

Thanks to Federico de Martino

8.6 Processing in BrainVoyager, surfaces in SUMA or Caret

For surface files (models (*.srf), time series (*.mtc) and statistical data (*.smp), the exchange format GIfTI will be implemented ('G' stands for 'geometry'). GIfTI is defined by a subcommittee of the NIfTI Data Format Working Group. The data are in XML (eXtended Markup Language) format, which is nice, because it is flexible, extendible and (when in ASCII) human readable. The format can be recognized by its extension "*.gii".
A plugin to export BrainVoyager surfaces can be downloaded from the wiki soon: `http://wiki.brainvoyager.com/NIfTI#Tools_2`. and from the GIfTI project page on NITRC. For more information on the GIfTI data format, see `http://www.nitrc.org/projects/gifti`.

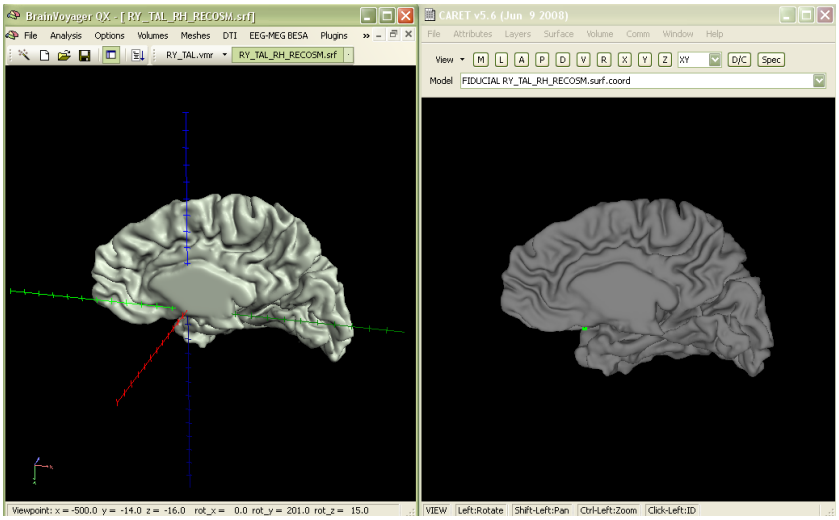


Figure 8.7: Same surface in BrainVoyager and Caret via GIfTI

8.7 Image registration in SPM, rest in BrainVoyager

If the image registration has been performed in SPM5, your functional image(s) has been transformed to the space of a template image, probably the MNI image. In the `spm_YearMonthDay.ps` file, the transformation is displayed in three separate functions for the new coordinates of X1, Y1 and Z1. On each line of the function 4 numbers can be found, which are the *coefficients* or *weights*, for example $X1 = coefficient1 * X - coefficient2 * Y + coefficient3 * Z + coefficient4$. These coefficients can also be presented in a coefficient matrix, which can be found in the BrainVoyager transformation files (*.trf)(see figure 8.8).

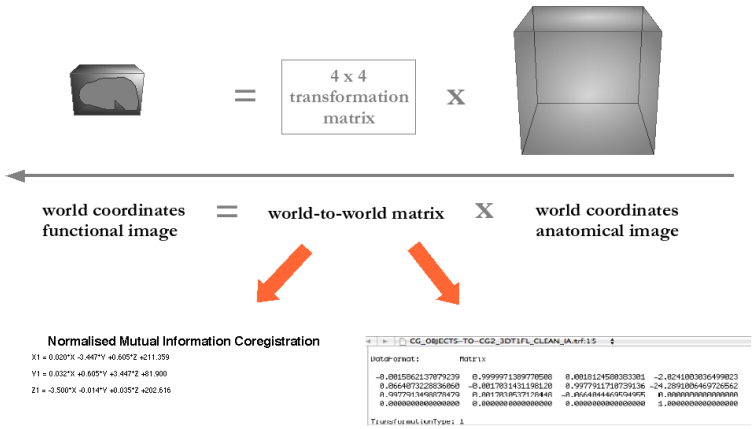


Figure 8.8: Transformation coefficients in BrainVoyager QX transformation file (*.trf) and spm5*.ps

Image registration in general consists of two parts. The *estimation* part estimates the parameters to transform the source image to the space of the target c.q. reference or template image. These 12 parameters indicate how much the source image needs to be rotated, translated and scaled to match the target image (see figure 8.9). These are the kind of parameters that can be found in the BrainVoyager QX initial and fine alignment transformation (*.trf) files.

Normalised Mutual Information Coregistration

X1 = 0.020*X -3.447*Y +0.605*Z +211.359

Y1 = 0.032*X +0.605*Y +3.447*Z +81.900

Z1 = -3.500*X -0.014*Y +0.035*Z +202.616

Figure 8.9: Transformation equations as result from the estimation process

The second part, the *reslicing*, transforms the source image using the estimated parameters.

When the functional data have been registered to the anatomical data, the functional data can be imported as VTC file. If the parameters have only been estimated, it is advisable first to reslice (resample) as well; a resliced file can be recognized from its 'r' prefix.
Because of the introduction of the NIfTI-1 file format in SPM5, the produced files are different. In SPM5, the parameters are saved in the header of a NIfTI-1 file

(*.hdr/*.img or *.nii). In SPM2, the data are saved in an Analyze 7.5 file and the parameters are saved in a *.mat file.

1. Use the Tools → 3D to 4D function to concatenate the functional images to one file with the name `4D.nii` (see figure 8.10). The positioning values of the separate volumes will be written to `4D.mat` file.

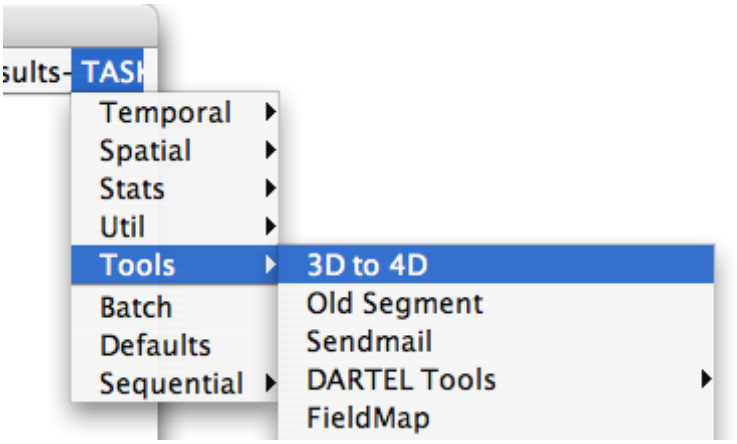


Figure 8.10: The 3D to 4D tool in SPM5

2. Open the VMR in BrainVoyager (or import the anatomical file).
3. Import the functional data in BrainVoyager QX. First, start the plugin (see figure 8.11).

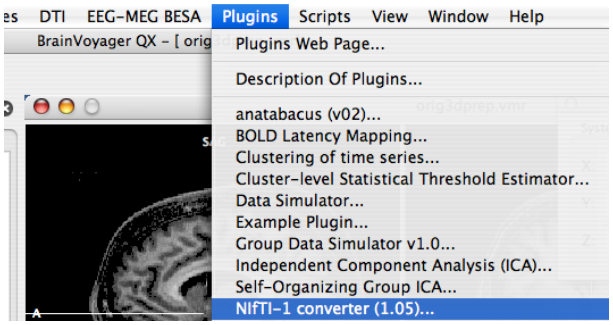


Figure 8.11: Select the NIfTI-1 converter from the BrainVoyager QX 'Plugins' menu

Then, select the *.nii option (see figure 8.12).

Choose a BrainVoyager file format for converting the functional data (see figure 8.13).

4. Adapt positioning [insert image here]

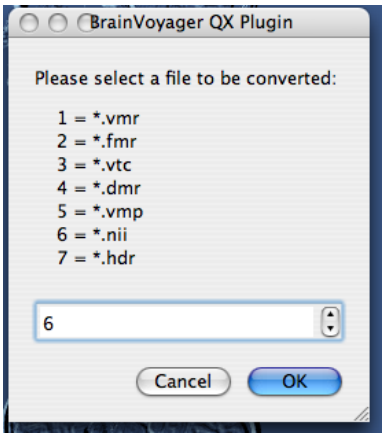


Figure 8.12: Select the NIfTI-1 file format (*.nii or *.hdr/*.img)

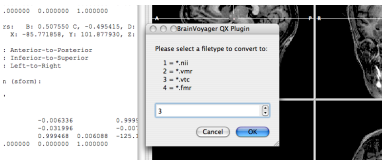


Figure 8.13: Select the volume time course (*.vtc) format

8.8 EPI distortion correction in SPM, rest in BrainVoyager

For a description of performing EPI distortion correction with the FieldMap toolbox, see
`ftp://ftp.fil.ion.ucl.ac.uk/spm/toolbox/FieldMap2/FieldMap_Manual_280207.pdf`.
The procedure is also briefly described in the
`DistortionInformation_v0x.pdf` document (by HB).

1. After unwarping of the EPI image(s), use the Tools $\rightarrow$ 3D to 4D function to concatenate the unwarped functional images (uf*.nii) to one file.
2. Import the image in BrainVoyager QX Import the unwarped functional images as *.fmr file. In case you like to visualize the magnetic field values, also import the fpm*.nii file, for example as *.vmr.
3. Adapt positioning

Chapter 9

Problem solving

9.1 Solutions to problems

9.1.1 My functional data files (*.hdr/*.img) are not 4D

To concatenate the files to one 4D file, use the "3D to 4D" function in the "Tools" section of SPM (see figure 8.10).

9.1.2 The brightness of the data sets looks different

9.1.3 A problem occurred during conversion

If you get the message "Conversion did not succeed. Please consult the converter manual for advice." check the following items:

1. Are the dimension fields in the NIfTI data correct?
2. Are the data in big endian byte order? If so, the magnificent `BVQXtools` by Jochen Weber can be used to swap your data to little endian.

9.1.4 How do I know what is left and right?

The handedness of an image can be checked by running the "Obtain image information" function, see chapter 10.2.
In the BrainVoyager anatomical data format, a header field `RadiologicalConvention` is available, which indicates that the image is in right-handed position (see figure 9.1).

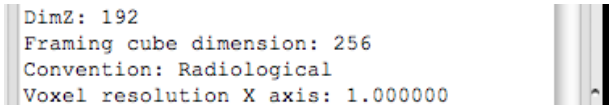


Figure 9.1: The handedness of a *.vmr file printed to the BrainVoyager QX Log tab

If necessary, change the handedness by flipping the X-axis. This can be performed via the BrainVoyager QX function "Flip X-axis" on the "VMR Properties" dialog (see figure 9.2).

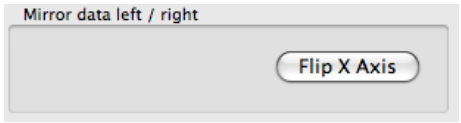


Figure 9.2: Change the handedness of a *.vmr file by flipping the X-axis

In NIfTI-1 images, the information is stored in the "qfac" parameter, in `pixdim[0]`. If this parameter is -1, this indicates that the image is in neurological convention (see figure 9.3).

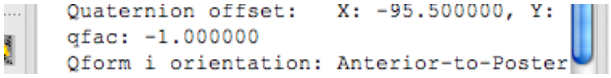


Figure 9.3: The handedness of a *.nii/*.hdr file printed to the BrainVoyager QX Log tab

9.1.5 The positioning information in my *.fmr/*.dmr file is wrong

In beta versions of the NIfTI-1 converter v1.05, the positioning information in the *.fmr or *.dmr file can be wrong. With the Matlab script below the proper positioning can be computed. The usage is indicated in the header. The script will produce a text file with the SliceCenter1X/Y/Z, SliceCenterNX/Y/Z, RowDirX/Y/Z and ColDirX/Y/Z values (see figure 9.4).

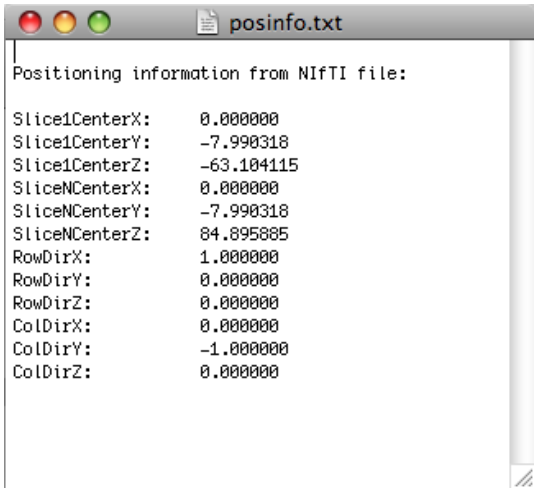


Figure 9.4: Positioning information for *.dmr file computed by script

```
%%%
```

9.1.6 The orientation of the image is wrong

The converter currently assumes that the orientation of the image is similar to the coordinate system, so that the positioning matrix has values on the diagonal:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Table 9.1: Identity matrix

where the 1s could also be other values, like 2 or 3.5; these numbers indicate the voxel sizes in each dimension (see also B.1).
If these numbers are not on the diagonal, the image is oriented differently (see figure 9.5).

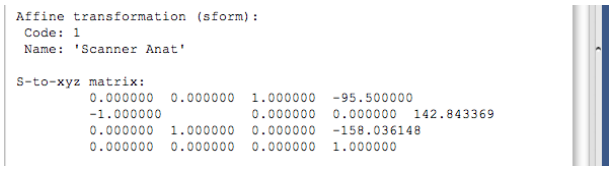


Figure 9.5: Sagittal orientation in a NIfTI file header printed to the BrainVoyager QX Log tab

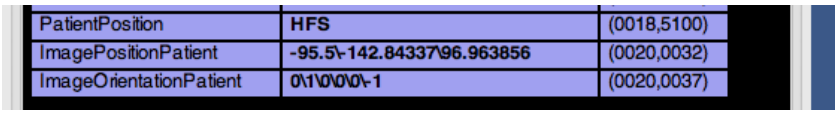
The orientation of the image that is captured in this *positioning* or *voxel-to-world matrix* can be found in the NIfTI-1 header fields *qform* and *sform*. If there are both qform and sform values, then the sform indicates the current position of the image in the coordinate system, while qform stores its original position.

0	0	n/a	-95.5
1	0	n/a	-142.84
0	-1	n/a	96.96
0	0	0	1

Table 9.2: The original orientation of the image w.r.t. the DICOM coordinate system

The header fields can be read via option "2 = Obtain image information" of the nifticonverter (see section 10.1.1). If the original DICOM files are available, the orientation can be found in the "ImageOrientationPatient" tag (0020, 0037) (see figure 9.6; the vectors are printed in the form `0\1\0\0\0\-1`). These 6 values make two orientation vectors, for example $(0, 1, 0)$ for the x-axis of the image and $(0, 0, -1)$ for the y-axis. The vectors are also known as "direction cosines". As shown in table 9.1.6, the vector for the x-axis $(0, 1, 0)$ is placed in the first column of a positioning matrix. The second vector $(0, 0, -1)$ is placed in the second column of the matrix. The third vector can be computed by computing the cross-product of the first two vectors.

From the first vector (0,1,0), one can conclude that the x-axis of the image is oriented in the direction of the y-axis of the coordinate system. The second vector (0,0,-1) indicates that the y-axis is oriented along the z-axis of the coordinate



PatientPosition	HFS	(0018,5100)
ImagePositionPatient	-95.5\-142.84337\96.963856	(0020,0032)
ImageOrientationPatient	0\1\0\0\0\-1	(0020,0037)

Figure 9.6: The orientation can be found in the "ImageOrientationPatient" tag (0020, 0037) in the form of two direction vectors (0,1,0) and (0,0,-1)

system. In the BrainVoyager positioning file (*.pos), these vectors are indicated with "RowDir" and "ColDir" 9.7. If the first vector (RowDir) for your *.vmr image is (0,1,0), you know that the image is along the anterior → posterior axis, c.q. sagittally oriented.

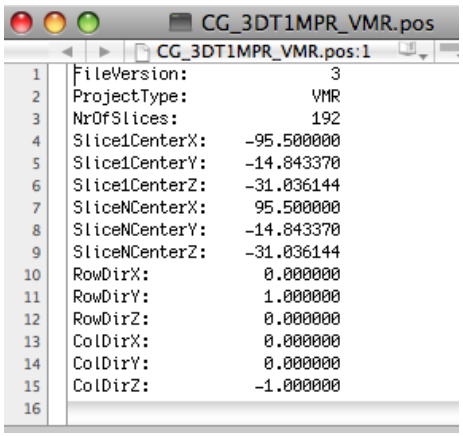


Figure 9.7: The RowDir and ColDir values in a BrainVoyager *.pos file reflect the direction vectors of the x-axis and y-axis of the image in DICOM space

If the plugin detects that the file is sagittal, it will not reorient the anatomical image. When the image is coronal, it will be reoriented. For all exceptions, use the "To Sag..." function on the 'Spatial Transf' tab of the 3D Volume Tools in BrainVoyager QX (see figure 9.8).

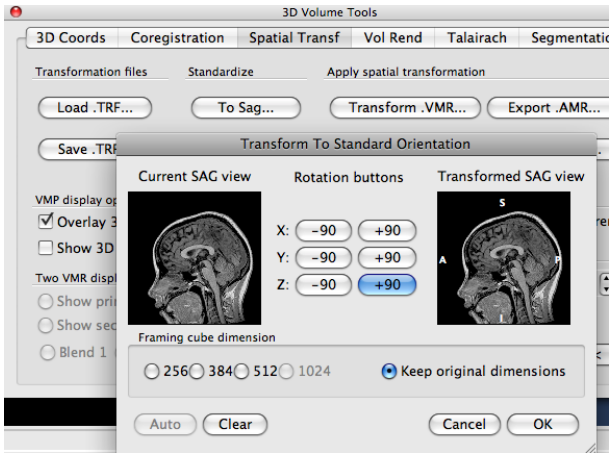


Figure 9.8: The function to transform to sagittal orientation on the 3D Volume Tools of BrainVoyager QX

9.1.7 I don't see an image, only stripes

In this case something might have gone wrong with the assumption of the storage order. When importing a 4D image, the converter assumes that the image is stored in blocks of volumes. If the image has however another order, the order of the voxels in the image will turn out wrong. In this case one would for example have chosen the 'Slices x time' option (see figure 9.9) on the advanced part of the BrainVoyager 'Create Project' dialog.

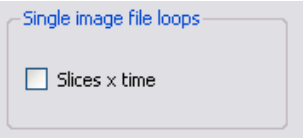


Figure 9.9: The slices x time option on the 'Create Project' dialog of BrainVoyager indicates a different voxel storage order

9.1.8 My data are in MNI space

The MNI templates (MNI305, MNI251) from the Montreal Neurological Institute in Canada are produced by transforming many brains to Talairach space, and then matching other brain scans to this template. MNI space is slighly different from Talairach space (for a review, see Brett et al, 2002, [1]). On his website `http://www.mrc-cbu.cam.ac.uk/Imaging/Common/mnispace.shtml`, Matthew Brett suggests the following solutions to convert from MNI to Talairach coordinates.

Redo the affine transform: This is a global affine transform as the ones used in BrainVoyager, that are applied to each voxel in the image. This was, according to Matthew Brett, suggested by Andreas Meyer-Lindenberg.

Non-linear transform (linear subvolume transformation): This method is based on performing another affine transformation to the data above the anterior commissure (AC) than below the AC.

Consequences of the transformation

For both solutions is the disadvantage that there is no brain scan of the Talairach brain, so that a perfect match will not be achieved. For the first solution, the affine transform, the drawback is that because of different brain shapes, the transformation will deform other parts of the brain. The second solution results in complex equations relating the two sets of coordinates.

The transformations

The proposed transformation by Andreas Meyer-Lindenberg is the following.

$$X' = 0.88X - 0.8Y' = 0.97Y - 3.32Z' = 0.05Y + 0.88Z - 0.44 \tag{9.1}$$

Matthew Brett proposed the following transformations for points above AC (where coordinates on the z-axis are larger or equal to 0).

$$X' = 0.9900XY' = 0.9688Y + 0.0460ZZ' = -0.0485Y + 0.9189Z \tag{9.2}$$

For below AC (where coordinates on the z-axis are smaller than 0) the transformation is printed below.

$$X' = 0.9900XY' = 0.9688Y + 0.0420ZZ' = -0.0485Y + 0.8390Z \tag{9.3}$$

Run one of these transformations to convert the coordinates from MNI to Talairach space; this could for example be performed by pasting the transformation matrix in a BrainVoyager transformation file (*trf) and using the "Load .TRF" and "Apply .TRF" functions on the 3D Volume Tools dialog. An automated solution will be available in one of the next versions of BrainVoyager QX.

9.2 My problem is not solved after reading the first part of this chapter

If the problem is not solved after trying to find a solution with the first part of this chapter, there are three solutions:

1. Upload the data to the brainvoyager.net ftp server and send a message to the developer of the nifti converter plugin
2. Use another converter
3. Use another file format

Chapter 10

Obtaining information about an image

10.1 Introduction

Sometimes one just would like to obtain more information about an image, for example whether it contains positioning information, which can be relevant for coregistering the image. This is possible via the Header information menu. It is possible to print the information about the image, which is contained in the header of the image, to the BrainVoyager QX Log tab, save the information in a text file, or both.

10.1.1 Header information menu

This menu option is included to provide the information about a file that is saved in its header.

Print file header to BrainVoyager QX Log tab: Print the information contained in an arbitrary image file to the BrainVoyager QX Log tab.

Save file header in text file: In this case the header information is saved in a text file; it resides in the same directory as the original image.

Print header to BVQX log tab and text file: This is a combination of the functions described above.

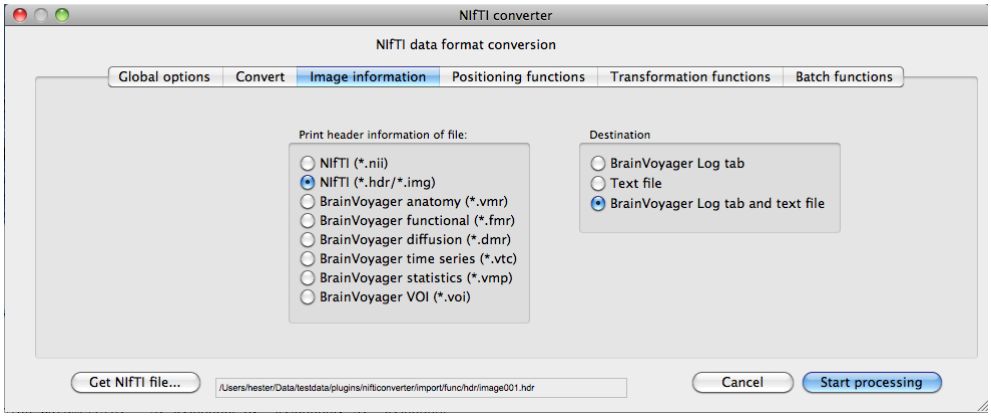


Figure 10.1: Print header menu

10.2 Print image information to BrainVoyager QX Log tab

Via this option the header information of the image that was selected earlier in the process will be printed to the BrainVoyager QX Log tab. If the Log tab is not the currently shown tab (the other tabs are 'File', 'Info' and 'Help'), or the Sidebar is not shown at all, the Sidebar will be opened and/or the Log tab will be shown.

10.3 Print image information to a text file

Via this option the header information of the image that was selected earlier in the process will be printed to a text file. This text file will have the same name as the image with the extension `*_info.txt`. The text file will be written to the same directory as the image.

10.4 Print image information to BrainVoyager QX Log tab and text file

When this option is selected, the information about an image that is contained in the header of an image is printed to the BrainVoyager QX Log tab (see section 10.2) and to a text file (see section 10.3).

Chapter 11

Positioning functions

The 'Positioning' functions provide information about the position of the image in a coordinate system. The positioning menu dialog is depicted in figure 11.1.

Represent positioning (*.pos) file in matrix: In BrainVoyager QX, the positioning information is contained in a positioning (*.pos) file. Via this submenu option the information in the positioning file is represented in a $4x4$ positioning matrix.

Create positioning (*.pos) file from matrix: Write a BrainVoyager QX a positioning (*.pos) file from a $4x4$ positioning matrix.

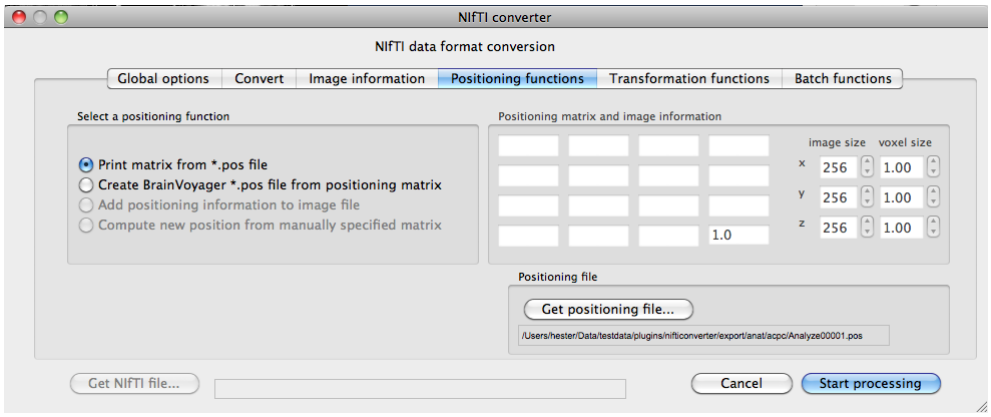


Figure 11.1: Positioning menu

For more information on positioning and transformations in MRI images in BrainVoyager and in general, please see the NIfTI-1 converter design document.

11.1 Represent positioning (*.pos) file in matrix

11.1.1 Purpose

In BrainVoyager QX, the positioning information is contained in a positioning (*.pos) file; the same fields can also be found in the headers of the anatomical and functional projects (*.vmr resp. *.fmr). Via this submenu option the information in

an arbitrary positioning file (*.pos) is represented in a $4x4$ positioning matrix. This makes it easier to compare the positioning information of BrainVoyager files with positioning information in the *qform* and *sform* fields of the NIfTI-1 file header. However, of course the information in the matrix refers to the position of the image in the BrainVoyager internal coordinate system in voxel coordinates. This means that in order to compare this to the *qform* and *sform* fields of the NIfTI-1 file header a voxel-to-world transformation to the NIfTI-1 coordinate system is required. For this functionality, please consult the section B on transformation functions in this plugin.

11.1.2 Using the function

To use this function, use the button on the tab to select a BrainVoyager positioning file (*.pos) file (see figure 11.2).

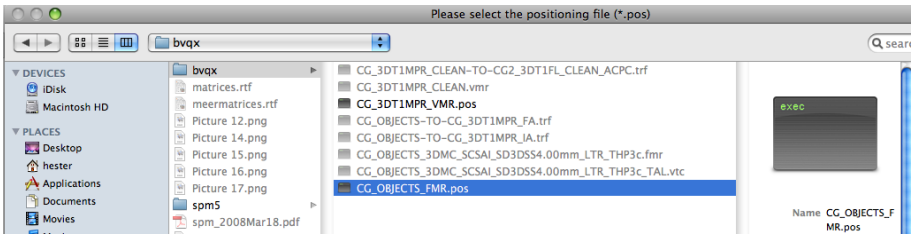


Figure 11.2: Selecting the positioning file

The positioning file will be read. The positioning information will be printed to the BrainVoyager QX Log tab in matrix representation (see figure 11.3).

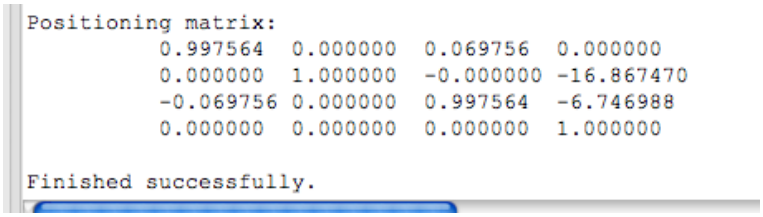


Figure 11.3: The resulting positioning matrix

11.2 Create positioning (*.pos) file for NIfTI-1 image

11.2.1 Purpose

Write a BrainVoyager QX a positioning (*.pos) file from a $4x4$ positioning matrix. These matrices can be found in NIfTI-1.1 files. The NIfTI-1 file extension can be *.nii or *.hdr/*.img. In the latter case, please verify that the file is not an Analyze 7.5 file (see section 10.2) or NIfTI-1.0, because these files do not contain positioning information, in which case it is impossible to generate a BrainVoyager (*.pos) file.

11.2.2 Using the function

Enter the matrix and the image and voxel sizes of the image on the "Positioning" tab. After clicking "Start", the plugin will provide a dialog to enter a name for the new positioning file (*.pos). The positioning information file (*.pos) can now be opened in a text editor in order to be inspected. Also, this file can be used to add positioning information to a BrainVoyager QX *.vmr or *.fmr file (see section **??**).

11.3 On positioning terminology

This section has been added for people with an interest to learn more about the positioning terminology in image processing and how positioning information is represented in the different coordinate systems of BrainVoyager, DICOM and NIfTI-1. It is not necessary to read this section in order to be able to use the converter.
To indicate the position of an image in a coordinate system, one can use orientation and position vectors. The *orientation vectors* indicate how much an image is rotated with respect to its coordinate system, and the *position vector* indicates the location of the image with respect to the origin.
The information provided in this section has been summarized via image 11.4.

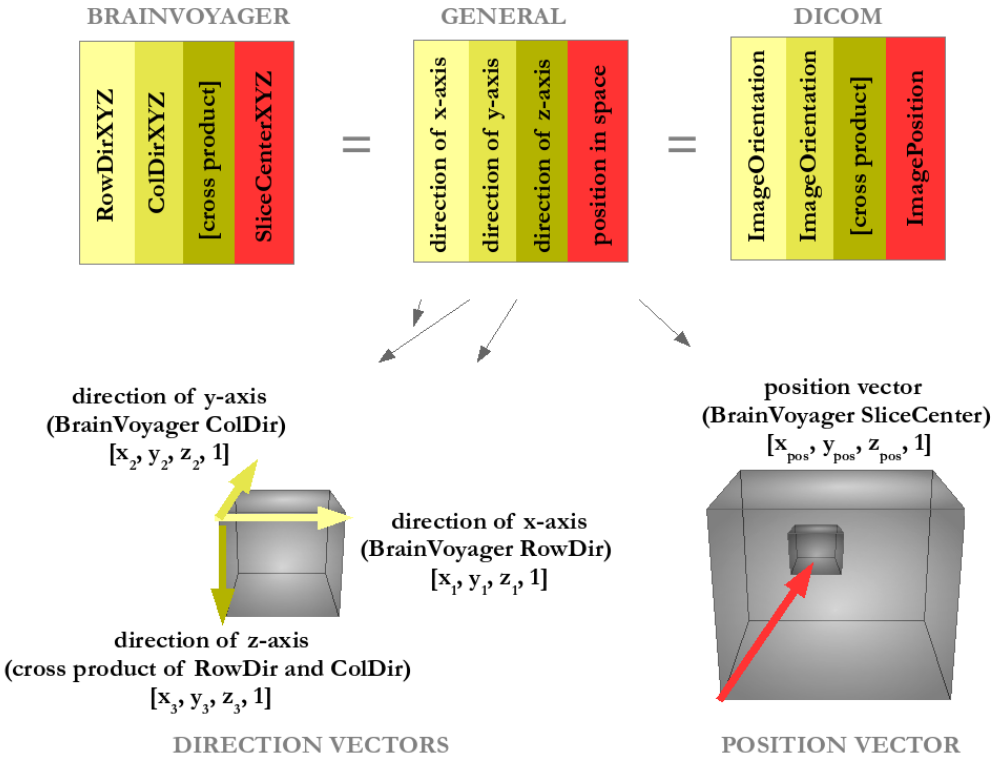


Figure 11.4: Positioning terminology and depiction of the orientation and position vectors

11.3.1 Orientation of the image

There are three orientation vectors (see left side of figure 11.4). The rotation of the x-axis of the image with respect to the coordinate system is one vector. The second vector is the rotation of the y-axis of the image with respect to the y-axis of the coordinate system. The third vector is the rotation of the z-axis of the image with respect to its coordinate system. The rotations are indicated in radians. These vectors that indicate the orientation of the image can also be called *direction vectors*. In BrainVoyager, these direction vectors can be found under the name "RowDir" for the x-axis vector and "ColDir" for the vector that indicates the orientation of the y-axis (see figure 11.5). The vector for the z-axis can be computed by taking the cross product of the first two vectors. This results in a normal vector.

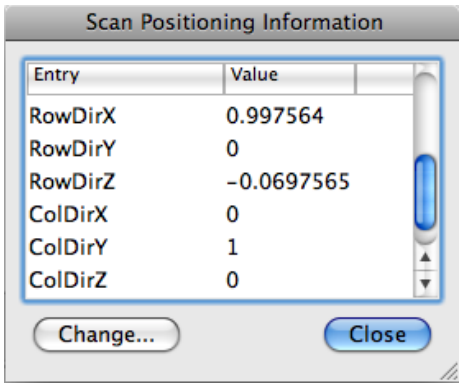


Figure 11.5: Orientation vectors in the FMR header

In DICOM, these orientation vectors can be found in the tag ImageOrientation-Patient (0020, 0037). The first three values represent the vector for the x-axis; in figure 11.6 these are the values [0.99756405, 0, -0.069756478]. The fourth, fifth and sixth values indicate the vector for the orientation of the y-axis; in figure 11.6 these values are [0, 1, -6.9388939e-018], values which can be rounded to [0, 1, 0].



Figure 11.6: Orientation vectors in the DICOM header tag (0020, 0037)

When taking the cross product, the vector for the z-axis can be computed. The cross product of [0.99756405, 0, -0.069756478] and [0,1,0] turns out to be [0.0698, 0, 0.9976]. This is the vector that indicates the orientation of the z-axis of the image and can be placed in the third column of the positioning matrix (see upper part of figure 11.4).

11.3.2 Position of the image

There is one position vector (see right side of figure 11.4). This vector contains the vector from the origin (0,0,0) to the first voxel of the image. In BrainVoyager, this vector can be found with the name SliceCenter1X/Y/Z and SliceCenterNX/Y/Z (see figure 11.7). In the DICOM header, this can be found in tag (0020, 0032). When creating a project in BrainVoyager, this information is also depicted in the "Info" tab (see figure 11.8).

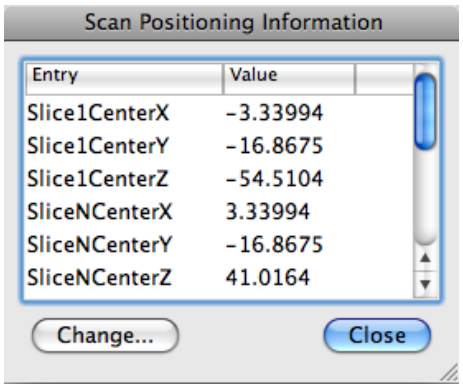


Figure 11.7: Position vector in the FMR header

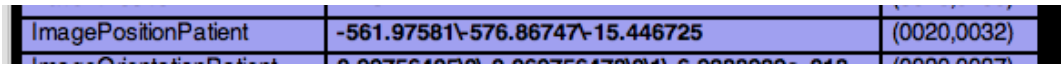


Figure 11.8: Position vector in the DICOM header (tag 0020, 0032) of functional data. In the Siemens version of the DICOM format, all images of one volume are often placed in one file (mosaic). The positioning information is now computed using concatenated image matrices, as if the images of the volume were lying in one plane. The plane is usually square, so an image volume of 36 slices would be placed in a plane with $6x6$ images.

11.3.3 Orientation and positioning information in a matrix

The positioning information in BrainVoyager anatomical (*vmr) and functional (*.fmr) files and the positioning file (*.pos) can be placed in a positioning matrix in the following way:

$$world2voxel = \begin{pmatrix} RowDir_x & ColDir_x & normal_x & SliceCenter_x \\ RowDir_y & ColDir_y & normal_y & SliceCenter_y \\ RowDir_z & ColDir_z & normal_z & SliceCenter_z \\ 0 & 0 & 0 & 1 \end{pmatrix}. \quad (11.1)$$

This matrix represents the BrainVoyager world-to-voxel transformation, which converts positions of voxels in the DICOM coordinate system (x, y, z) to BrainVoyager voxel indices (i, j, k). To compute the coordinates of a voxel, multiply the inverse of this "world-to-voxel" matrix with the voxel indices (i, j, k) of a point in the image:

$$voxelposition = world2voxel^{-1} * [i, j, k, 1]^T \quad (11.2)$$

This computation transforms the set of voxels back to a metric space (coordinate system).

In NIfTI, the positioning matrix can be created from the *qform* and *sform* header fields (see figure 11.9). Also in this case the first three columns indicate the orientation of the image, while the fourth column indicates the position with respect to the origin (0,0,0).

```
Rigid body transformation (qform):
 Code: 1
 Name: 'Scanner Anat'

Q-to-xyz matrix:
         -3.491474           -0.000000           -0.278325          115.067116
         0.000000  3.500000  -0.000000           -91.632530
         -0.244145           0.000000  3.980281  -46.697628
         0.000000  0.000000  0.000000  1.000000
```

Figure 11.9: Orientation and position vectors in the NIfTI-1 header

Chapter 12

Transformation functions

12.1 Transformations menu

In this preview version of the plugin v1.07.010410, the transformation functions have not yet been enabled.

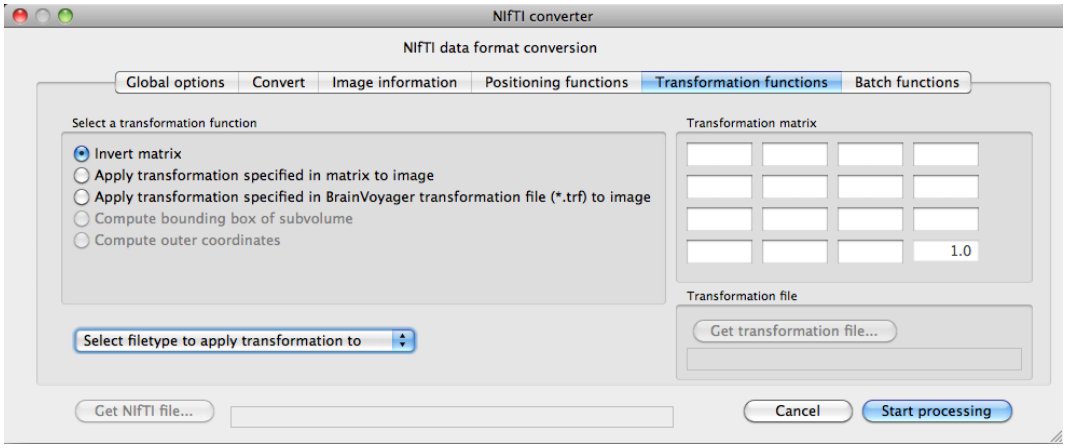


Figure 12.1: Transformation menu

12.2 Notes on transformations

12.2.1 Introduction

This section has been added for people with an interest to learn more about the transformations terminology in image processing and how transformation information is represented in the different coordinate systems of BrainVoyager, DICOM and NIfTI-1. It is not necessary to read this section in order to be able to use the converter.

Coordinate changes for a 3-dimensional object in vector space can be classified as *global* or *local*. When the coordinates of an object are changed, this can be represented in a matrix. In global transformations, the same matrix can be applied to each voxel.
For a 3-dimensional brain image, a $4x4$ matrix can be used for a global coordinate change. This involves a coordinate change along all axes or some of the axes. Changes can be translations (shifts to a certain direction), rotations, scaling (change of size, sometimes referred to as zooms), shears (skewness towards one direction) and perspective transformations (vanishing horizon in a single point) (see figure 12.2).

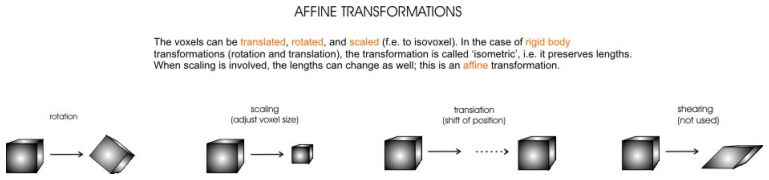


Figure 12.2: There are four types of actions in affine transformations

In local, or non-linear, coordinate changes, different transformations are applied to certain voxels. This is the case for the Talairach transformation in BrainVoyager, where different matrices are applied for different sub-volumes. This is sometimes called 'piecewise affine', since the matrix is still a $4x4$ matrix.
An example of an affine transformation is the initial alignment in BrainVoyager. In this transformation, the position of the functional image (*.fmr) is transformed to the position of the anatomical image (*.vmr). To achieve this, the inverse of the positioning matrix of the anatomical image is multiplied with the positioning matrix of the functional image (see figure 12.3).

The resulting matrix is stored in a transformation file (`*_IA.trf`). In figure 12.4 is shown that the shift or translation of the image can be found in the fourth column of the transformation matrix. The other parameters are shown in appendix B.

12.2.2 Storage of transformation information

While the original position of the image is stored in the image in the cases of BrainVoyager image data (anatomical (*.vmr), functional (*.fmr), diffusion weighed (*.dmr)), each transformation being applied to the image is also stored in some way.
The storage for volumetric data in BrainVoyager is summarized in table 12.2.2.

The storage for volumetric data in NIfTI is summarized in table 12.2.2.

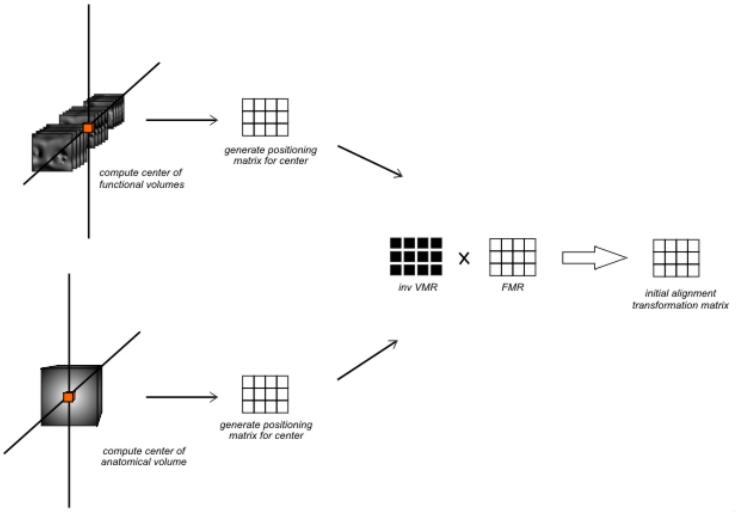


Figure 12.3: Transformation from FMR to VMR (please note that the drawing of the center on the FMR volume should be on the first volume only)

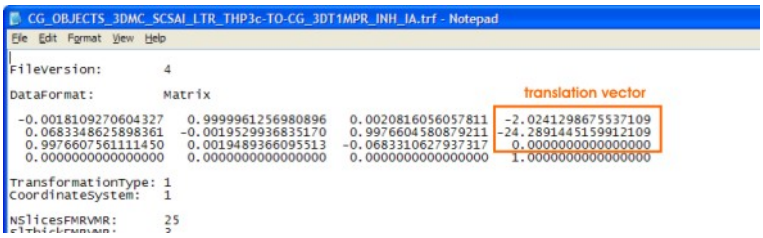


Figure 12.4: The translation vector can be found in the fourth column of a transformation matrix

type	internal storage	external storage
position (scanner)	VMR Properties	*.pos
rigid body transformation	VMR Properties	*.trf
affine transformation (IA, FA)	VMR Properties	*.trf
Talairach transformation	VMR Properties	*.tal

Table 12.1: Storage of position and transformation information in BrainVoyager files

type	internal storage	external storage
position	qform	n/a
rigid body transformation	qform	n/a
affine transformation	sform	n/a
Talairach transformation	no	n/a

Table 12.2: Storage of position and transformation information in BrainVoyager files

12.2.3 Differences in storage of transformations

There is a difference between how the transformations are stored in NIfTI-1 images and how the transformations are stored in BrainVoyager image data.
In BrainVoyager image files, the original position of the image in a BrainVoyager flavour of DICOM coordinates is preserved (see the figures 11.7 and 11.5). All subsequent transformations are also preserved (see figure 12.6). The current position of the image has to be calculated by multiplying these transformations. The fact that the original position in the scanner and the history of transformations is preserved in BrainVoyager image data, provides the user with a high degree of flexibility and a nice logging functionality.

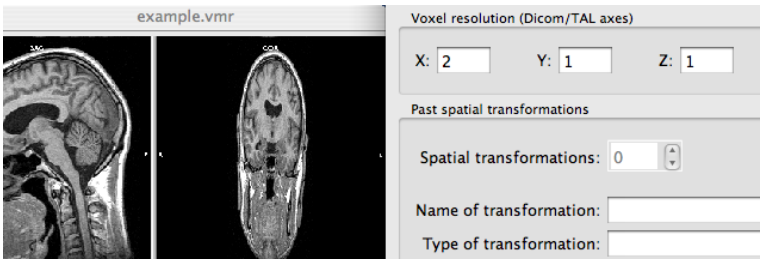


Figure 12.5: Before the transformation: anisotropic voxel sizes in the VMR Properties

The image properties, for example voxel sizes, can be inspected via the 'VMR Properties' of the BrainVoyager 'File' menu (see figure 12.5). As in shown in figure 12.6, transformations are also saved in the 'VMR Properties'.

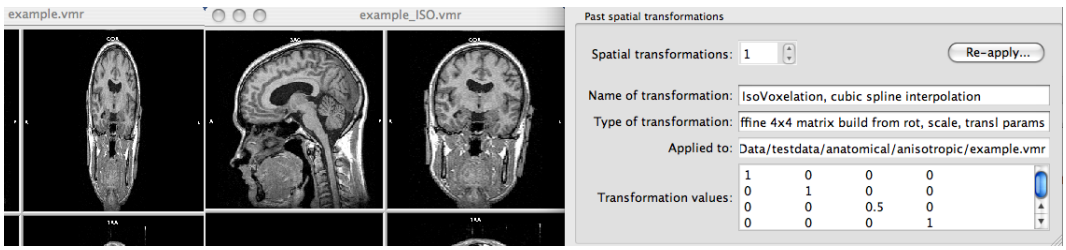


Figure 12.6: After the transformation: storage in the VMR Properties

The matrix in figure 12.6 makes clear that the transformation is applied to BrainVoyager internal coordinates, because the anisotropic voxel size being reported in the VMR Properties is X, while in the matrix the size for Z is set to 0.5. This indicates that the x-axis in BrainVoyager system coordinates is z-axis in the BrainVoyager internal coordinate system (see appendix A).
In NIfTI images, the positioning information is always updated to the newest transformation. Since there is only place for storing two transformations, in the *qform* and *sform* fields, this might not make it possible to retrieve the native position as the image was acquired.

Appendix A

Spatial transformations in BrainVoyager

A.1 Theoretical background and implementation in BrainVoyager

Rainer Goebel

A.1.1 Introduction

This technical document aims to provide detailed knowledge about spatial transformations in general and how they are implemented in BrainVoyager. The document focuses on the axes systems used in BrainVoyager and the assumed order of axes rotations. In addition, it is described how rotation, translation and scaling transformations are properly combined to create a $4x4$ affine transformation matrix as well as how such a matrix is properly decomposed into elementary transformations. The presented information is aimed towards advanced users who want to a) simply understand these issues better or b) want to use transformation results from other software in BrainVoyager or c) want to use transformation results produced by BrainVoyager for other (custom) software.

It is a necessity that successive rotations about coordinate axes are treated consistently in all volume- and surface-level coordinate transformation routines of BrainVoyager. This is particularly important since successive axis rotations (in contrast to successive translations) do not commute, in that the composed transformation depends on the order in which individual rotations are applied. BrainVoyager saves spatial transformations in a .TRF file, which contains, among other parameters, three values for rotations around the three coordinate axes. The order of applying these angles must be consistent across different modules of the program. It is, for example, possible to load a TRF file within the surface module in order to apply the same transformation on a surface which had been previously applied to a 3D VMR data set, or vice versa. Besides ensuring a consistent explicit specification of rotation angles across modules, all automatic rigid body coregistration routines (3D motion correction, 3D-3D coregistration etc.) also have to result in rotation angles, which are consistent with the implied order of axes rotations.

A.1.2 BrainVoyagers axes conventions

BrainVoyager uses several different coordinate systems: the internal axes, the standard Dicom and Talairach axes and the OpenGL axes. To the user, normally only the Dicom/Talairach axes system is presented. The internal axes system of BrainVoyager was defined initially for sagittal 3D volumes. The dimensions of the sagittal images defined the X and Y axes with values ranging from 0 to 255 (X: anterior to posterior, Y: superior to inferior) and the dimension across the slices defined the Z axis (right-to-left) with values from 0 to 255 or less. This original decision (which was at the end of the year 1995) is still the basis of the internal axes system, which is depicted in figure A.1.

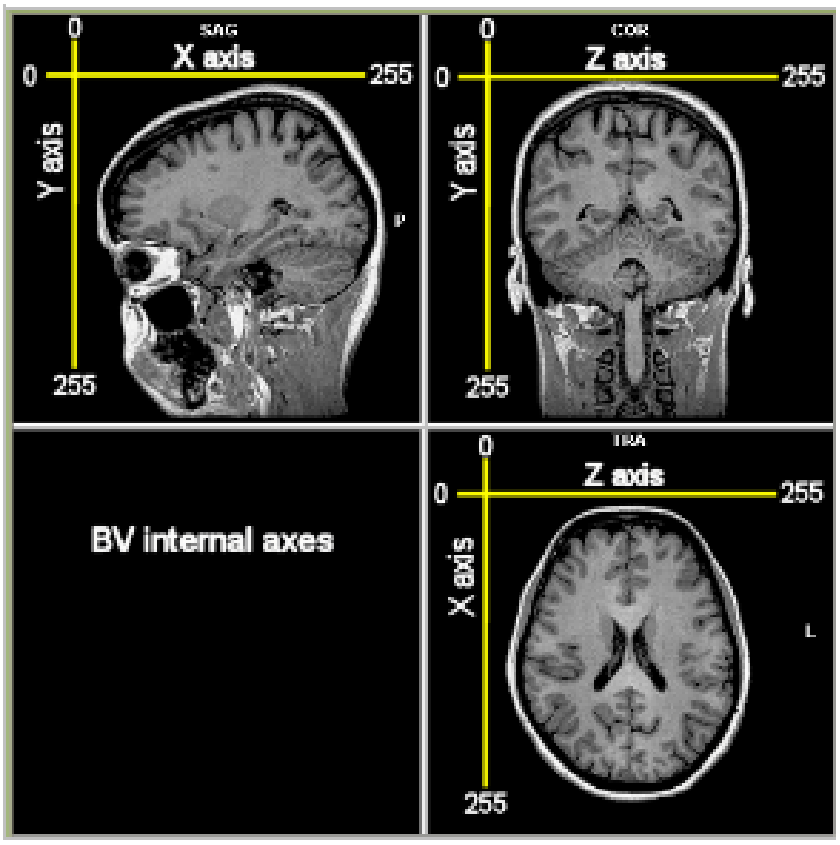


Figure A.1: The BrainVoyager internal coordinate system

To the user, the axes are presented according to the Talairach/Dicom naming standard, i.e. the X axis in Talairach space corresponds to the Z axis in BVs internal definition, the Y axis in Talairach space corresponds to the X axis in BVs internal definition, and the Z axis in Talairach space corresponds to the Y axis in BVs interal definition. Note that so far, this only changes the labeling of the axes, the values are still from 0 to 255 along these relabeled axes in the original direction. Since these relabeled axes are still internal definitions, they are shown to the user as "system coordinates", for example in the System coords field of the 3D Volume Tools dialog (see below). In the following these relabeled axes will be called X_{SYS}, Y_{SYS} and Z_{SYS} (SYS for "System"). The original (internal) axes are referred to as X_{BV}, Y_{BV} and Z_{BV} (BV for "BrainVoyager internal").

Besides the internal (BV) and system (SYS) coordinates, BrainVoyager also sup-

ports "real" Talairach coordinates, if appropriate. In the Talairach coordinate system, the origin and axes values are defined with respect to landmarks of the brain. Most importantly, the origin of the coordinate system is specified to be the anterior commissure (AC) of the brain. Together with the posterior commissure (PC) and additional landmarks specifying the border of the brain, the values along the X, Y, and Z axis are defined. These Talairach coordinates are shown in the Talairach coords field (see figure above). To enable Talairach coordinates, the Use Tal ref points option has to be checked in the Talairach tab of the 3D Volume Tools dialog. The figure below shows that the directions of the Talairach axes are oppositely defined as compared to the internal/system axes (compare 0 to 255 with $-$ to $+$ directions). The Talairach axes will be referred to as X_{TAL}, Y_{TAL} and Z_{TAL}. In BrainVoyager, a brain is transformed into Talairach space in two steps, 1) ACPC transformation and 2) Talairach scaling based on the proportional grid system. The first step is a normal rigid body transformation (represented with a standard TRF file) while the latter requires a special step based on a "TAL" file. A TAL file contains x,y,z specifications of the AC and PC points and the cerebrum borders defined on the ACPC brain. The landmarks are used for Talairach piecewise scaling of the ACPC brain according to the proportional grid (Talairach & Tournaux, 1988) resulting in a normalized brain.

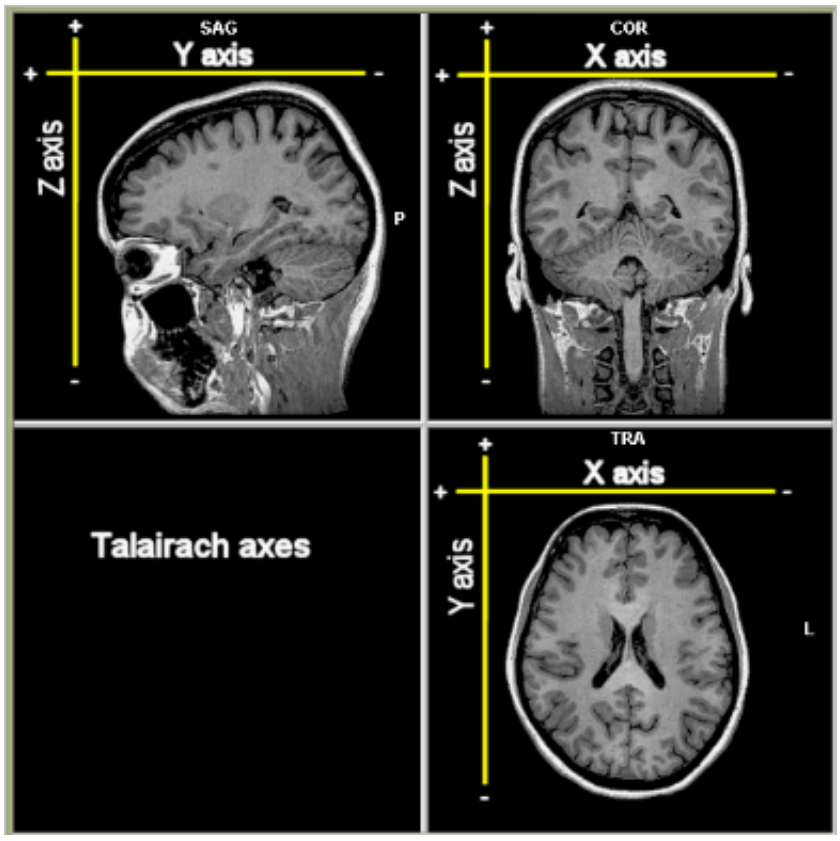


Figure A.2: The Talairach coordinate system

Note that the definitions of the system coordinates assume that the brain is in BrainVoyagers "standard" orientation, i.e. that 3D data is represented as a series of sagittal planes. If this is not the case for a raw data set, the program provides the "To SAG" function to exchange the axes accordingly. BrainVoyager QX tries

to perform this step automatically based on header information and an analysis of the symmetry properties of the data set. BrainVoyagers standard orientation also assumes that the data set is in radiological convention ("Left-Is-Right"). This is normally the case when reading native scanner data (manufacturers DICOM or proprietary file formats such as Siemens IMA (Numaris versions prior to 4) or GE "I" or GE "MR" files). If you are sure that your data is not in radiological but in neurological convention (Left-Is-Left), you have to specify this in the *Transform to Standard SAG* dialog. Data in neurological convention may be encountered if you read data not directly from the scanner but from files exported by another program.

The surface module visualizes reconstructed meshes and optionally displays two coordinate frames, the OpenGL and the Talairach coordinate system. The OpenGL coordinate axes (see figure below) are shown in the lower left corner and correspond directly to the system coordinates ($X_{OGL} = X_{BV}$ etc.). The OpenGL axes are identified with letters as well as with a color code denoting the X axis with red, the Y axis with green and the Z axis with blue. In addition to the OpenGL axes, the Talairach coordinate system is also shown (see figure below). The axes can be identified by color, i.e. the X_{TAL} axis is drawn in red, the Y_{TAL} axis in green and the Z_{TAL} axis in blue. If enabled, the Talairach axes are shown always in the same way even if a displayed mesh is not normalized into Talairach space. The mesh shown in figure A.3 is drawn 1-to-1 from voxel coordinates of the corresponding 3D (VMR) data set. Since the original VMR data sets are normally in radiological convention, the mesh is shown in a left-right mirrored way.

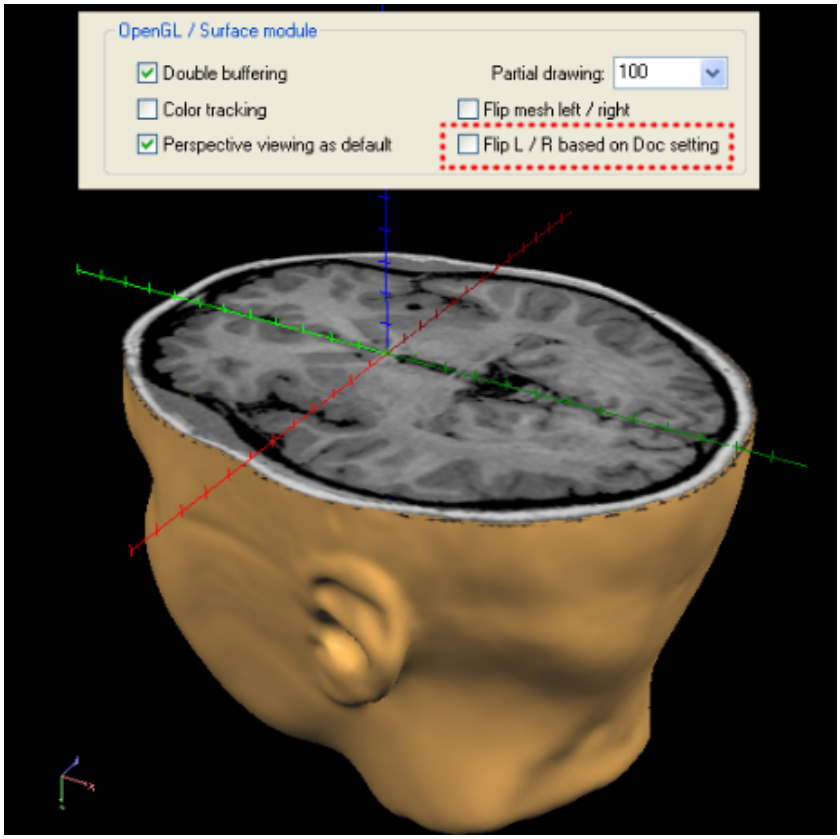


Figure A.3: The OpenGL coordinate system

As default, the surface module does not display the meshes as shown above but attempts always to display them in its natural space: The program tries to assure that the seen left side of a mesh always corresponds to the true left side of the data set independent of neurological or radiological convention. To accomplish this, a general flag `Flip L / R based on Doc` setting is defined, which flips the values of the left-right coordinate axis in case that the data set is in radiological convention. This flag can be found in the Global Options dialog, which can be invoked by clicking the File → Global options menu item (see figures above and below). This flag is checked as default and it should be always turned on to ensure correct display of a mesh as shown in the figure below.

A.1.3 3D affine transformation matrices

Any combination of translation, rotations, scalings/reflections and shears can be combined in a single 4 by 4 affine transformation matrix:

$$M = \begin{pmatrix} M_{11} & M_{12} & M_{13} & M_{14} \\ M_{21} & M_{22} & M_{23} & M_{24} \\ M_{31} & M_{32} & M_{33} & M_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.1}$$

The 4 by 4 matrix M corresponds to a affine transformation T() that transforms point v to point u. In other words, the transformation of point u is found by multiplying v by M:

$$u = Mv \tag{A.2}$$

The 4 by 4 transformation matrix uses homogeneous coordinates, which allow to distinguish between points and vectors. Vectors have a direction and magnitude whereas points are at certain coordinates with respect to the origin and the three base vectors i, j and k. Points and vectors are both represented as mathematical column vectors (column-matrix representation scheme, see note below) in homogeneous coordinates with the difference that points have a 1 in the fourth position whereas vectors have a zero at this position. The transformation of the point v to point u is thus written as:

$$\begin{pmatrix} x' \\ y' \\ z' \\ 0 \end{pmatrix} = M \begin{pmatrix} x \\ y \\ z \\ 0 \end{pmatrix} \tag{A.3}$$

We now consider the nature of elementary 3D transformations individually and then compose them into general affine transformations. Note that for an affine transformation matrix, the final row of the matrix is always 0001 and we have to understand the role of the elements in the upper 3 by 4 matrix.

Translation

For a pure translation, the matrix M has the simple form:

$$\begin{pmatrix} 1 & 0 & 0 & M_{14} \\ 0 & 1 & 0 & M_{24} \\ 0 & 0 & 1 & M_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.4}$$

Applying this matrix to point v reveals that $u = Mv$ is simply a shift in v by the vector $t = (t_x = m_{14}, t_y = m_{24}, t_z = m_{34})$.

$$M = \begin{pmatrix} 1 & 0 & 0 & x+t_x \\ 0 & 1 & 0 & y+t_y \\ 0 & 0 & 1 & z+t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.5}$$

Scaling

A scaling operation along the three axes is represented by the following matrix:

$$\begin{pmatrix} m_{11} & 0 & 0 & 0 \\ 0 & m_{22} & 0 & 0 \\ 0 & 0 & m_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.6}$$

Applying this matrix to point v results in (with $sx = m_{11}$, $sy = m_{22}$, $sz = m_{33}$):

$$M = \begin{pmatrix} x' \\ y' \\ z' \\ 0 \end{pmatrix} = \begin{pmatrix} m_{11} & 0 & 0 & 0 \\ 0 & m_{22} & 0 & 0 \\ 0 & 0 & m_{33} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} x & v_x \\ y & v_y \\ z & v_z \\ 0 & 0 \end{pmatrix} \tag{A.7}$$

Shearing

Shearing operations belong to affine transformations and are achieved by non-zero off-diagonal elements in the upper 3 by 3 submatrix. Shears are, however, not used in BrainVoyagers standard spatial transformation, which corresponds to pure rigid body transformations (rotations and translations) plus scaling for matching different voxel sizes between data sets.

Rotations

Rotations represent the last elementary 3D transformation, which are the most important ones in the present context. It is common to specify arbitrary rotations with a sequence of simpler ones each along one of three coordinate axes. In each case, the rotation is through an angle, about the given axis. The following three matrices R_x, R_y and R_z and represent transformations that rotate points through the angle b in radians about the coordinate origin:

$$R_x(b) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & cos(b) & -sin(b) & 0 \\ 0 & sin(b) & cos(b) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.8}$$

$$R_y(b) = \begin{pmatrix} cos(b) & 0 & sin(b) & 0 \\ 0 & 1 & 0 & 0 \\ sin(b) & 0 & cos(b) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.9}$$

$$R_z(b) \begin{pmatrix} cos(b) & -sin(b) & & 0 \\ sin(b) & cos(b) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{A.10}$$

It must be further defined whether positive angles perform a clockwise (CW) or counterclockwise (CCW) rotation around an axis with respect to a specification of the orientation of the axis. In BrainVoyager QX, positive rotation angles cause

a counterclockwise rotation about an axis as one looks inward from a point on the positive axis toward the origin. This is commonly the case for right-handed coordinate systems as used in BrainVoyager.

A.1.4 Composing 3D affine transformations

An important property of affine transformations is that they can be composed, and the result is another 3D affine transformation. A single matrix an be set up for any sequence of transformations as a composite transformation matrix. Forming products of transformation matrices is often referred to as a concatenation, or composition of matrices. For column-matrix representation of coordinate positions, we form composite transformations by multiplying matrices in order from right to left. That is, each successive transformation matrix premultiplies the product of the preceding transformation matrices. The matrix that represents the overall transformation is the product of the individual matrices M_1 and M_2 that perform the two transformations, with M_2 premultiplying M_1:

$$M = M_2 M_1 \tag{A.11}$$

Any number of affine transformations can be composed in this way, and a single matrix results that represents the overall transformation. This composite matrix can then be applied to any point (column vector) as usual, i.e. $u = Mv$.

NOTE 1 Matrix multiplication is associative. For any three matrices, A, B, and C, the matrix product ABC can be performed by first multiplying A and B or by first multiplying B and C : $ABC = (AB)C = A(BC)$. Therefore we can evaluate matrix products using either a left-to-right or a right-to-left associative grouping. The important point is that matrix multiplication is not commutative in general: The matrix product AB ("A premultiplies B") is generally not the equal to BA ("B premultiplies A"). This means that if a sequence of translations, rotations and scalings is applied, the order in which the elementary transformation matrices appear is critical to determine the overall transformation. Only for some special cases, such as a sequence of transformations all of the same kind (i.e. two translations or two rotations around the same axis), the multiplications of transformation matrices is commutative.

NOTE 2 The "right-to-left" order of transformation matrices holds for column-matrix representations as used in this text. In this representation, points such as u and v are represented as column vectors. Another convention being used in the literature is row-matrix representation in which points are represented as row vectors. A conversion between these conventions is easy by exploiting a property of matrix transposition: The transposition of a matrix product is equivalent to the product of the transposition of each matrix, with the order of multiplication reversed: $(AB)^T = B^T A^T$. Thus, the transformation of vector **v** in columnar-matrix representation $u = M_2 M_1 v$ equals $u = v^T M_1^T M_2^T$ in row-matrix representation.

A.1.5 The order of rotations in BrainVoyager

Since translations commute, the order of applying displacements along the three axes does not matter. The order of rotations about the three coordinate axes, however, is critical since rotations are not commutative. The default order of rotations in BrainVoyager is:

1. Rotation around Y_{SYS} axis (X_{BV} axis)

2. Rotation around Z_{SYS} axis (Y_{BV} axis)

3. Rotation around X_{SYS} axis (Z_{BV} axis)

If three non-zero angles are supplied, BrainVoyager performs first the rotation about the Y_{SYS} axis (X_{BV} axis), then about the Z_{SYS} axis (Y_{BV}) and finally about the X_{SYS} axis (Z_{BV}). This order was defined in a "natural order" (X_{BV}, Y_{BV}, Z_{BV}) with respect to the internal axes definition, but appears arbitrary with respect to the system coordinates. In BrainVoyager QX and BrainVoyager 5.x, the order of axes rotation can now be specified in the new TRF file format (see below). Because BrainVoyager was developed initially in the context of data from Siemens scanners, the rotation about the coordinate axes does also appear in Siemens terminology in the user interface, especially in the Angles field of the Reslicing tab of the 3D Volume Tools dialog (see red rectangle in the figure below).



Figure A.4: Rotations in BrainVoyager 2000

The $Tra \rightarrow Cor$ angle corresponds to rotation about the X axis, the $Tra \rightarrow Sag$ angle corresponds to rotation about the Y axis, and the $Sag \rightarrow Cor$' angle corresponds to rotation about the Z axis. In BrainVoyager QX, this Siemens terminology is no longer used and is replaced by standard transformation labels (see figure below). The rotation axes are now denoted as "x", "y", "z", corresponding to the X_{SYS} (Z_{BV}), Y_{SYS} (X_{BV}) and Z_{SYS} (Y_{BV}) axes:

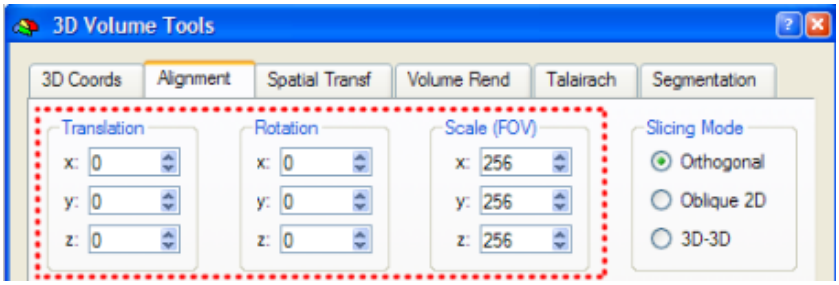


Figure A.5: Rotations in BrainVoyager QX

The scaling parameter can be specified now either as Field-Of-View units (millimeter) or as standard scaling parameters. A FOV value of 256 corresponds to a scale value of "1.0".
For a complete specification of a rotation, we must specify a rotation angle and the position of the *rotation point* (or *pivot point*) about which the data set is to be

rotated. The default coordinates for the rotation point is the center of the 3D data set, i.e. D/2 with D equal to the number of voxels in the respective dimension. In a 256 by 256 by 256 data set, the rotation point would be thus 128, 128, 128. If translation parameters are specified, the rotation point changes accordingly because the translation is performed prior to the rotation. In BrainVoyager, the translation/rotation point is defined as the coordinates of the current position of the red cross. In BrainVoyager QX, the position of the red cross and the x, y, z translation values are separated (see Translation fields in figure A.5). This separation of the translation parameters from the position of the red cross in BrainVoyager QX has the advantage that a spatial transformation can be specified while it is still possible to "browse" the data set.

Note. The default rotation point is not the exact center of the data set, which would be for the x axis: $XC = (DX - 1.0)/2.0$. With a dimension of 256 voxels, the center would be $XC = 127.5$. Since this would, however, put the rotation point at a non-integral (non-visible, intermediate) point, the $D/2$ definition is used for the default rotation point. For scaling operations, however, the default fixed point (the point which remains unchanged) is the true center of the data set, $(D - 1.0)/2.0$. Scaling is used to match the voxel resolution of different data sets, i.e. during FMR-VMR coregistration.

Decomposition of a rotation matrix into Euler angles

As described above, a complex affine transformation can be constructed by composing a number of elementary ones. We can also ask the opposite question and ask, what elementary operations "reside in" a given affine transformation matrix? Unfortunately, this problem has not a unique solution since a matrix M may be factored into a product of elementary matrices in various ways. There are, for example, many ways to combine basic rotations to achieve the same composite rotation. In the following, we assume that we have a matrix representing only translation, rotation and scaling transformations.

The three translation values are easy to extract, they are simply the three upper values of the fourth column

$Tx = m_{14}$
$Ty = m_{24}$
$Tz = m_{34}$

The scaling factors are then extracted as:

Finally the rotations are extracted as follows:

$Ry = asin((-row[0].z)$
if $(cos(y)! = 0)$ **then**
  $Rx = atan2(row[1].z, row[2].z)$
  $Rz = atan2(row[0].y, row[0].x)$
else
  $Rx = atan2((row[1].x, row[1].y)$
  $Rz = 0$
end if

A.1.6 The TRF file format for spatial transformations

Spatial transformations are saved in "TRF" files in BrainVoyager. These plain text files do not contain a 4x4 matrix but save translation, rotation and scale values separately for each axis. This choice has been made solely to allow for easy readability. If a TRF file is applied, a respective 4x4 matrix is internally constructed from the individual values. Transformation matrices from multiple TRF files are

also internally multiplied as detailed above. This happens, for example, during VTC creation combining two TRF files, one for FMR-VMR and one for VMR-VMR (ACPC) transformation. BrainVoyager QX will also support the explicit combination of multiple TRF files as well as the composition and decomposition of homogeneous $4x4$ matrices.
Version 3 is the latest version of this file format introduced with BrainVoyager QX and also supported in BrainVoyager 5.x. The new format allows to explicitly specify the order of rotation while the old format supported only the implicit order: Y_{SYS}, Z_{SYS}, X_{SYS}. A typical TRF file used to look like this:

```
FileVersion:        3

xTranslation:       0
yTranslation:       8
zTranslation:       14

xRotation:          -14
yRotation:          1
zRotation:          -1

xScaleAsFoV:        256
yScaleAsFoV:        256
zScaleAsFoV:        256

OrderOfRotations:   XYZ

TransformationType: 2
CoordinateSystem:   1
```

while in the newer BrainVoyager QX versions the parameters are provided in the form of a transformation matrix in the TRF file:

```
FileVersion:        5

DataFormat:         Matrix

  0.0000010660081671   0.9786220788955688  -0.2056666463613510   4.3583703041076660
 -0.0019511014688760   0.2056662589311600   0.9786202311515808  -9.4430999755859375
  0.9999980926513672   0.0004002332862001   0.0019096103496850   1.4527800083160400
  0.0000000000000000   0.0000000000000000   0.0000000000000000   1.0000000000000000

TransformationType: 1
CoordinateSystem:   1

NSlicesFMRVMR:      20
SlThickFMRVMR:      3.5
SlGapFMRVMR:        0
CreateFMR3DMethod:  3
AlignmentStep:      1

ExtraVMRTransf:     0

SourceFile:         "C:/Data//fmr/series-0005.fmr"
TargetFile:         "C:/Data/vmr/series-0003.vmr"
```

The file shown below is an example of an initial alignment transformation file (`*_IA.trf`), that registers a functional file (*.fmr) to an anatomical file (*.vmr).

A.1.7 Summary of coordinate systems

Summary of axes systems in BrainVoyager QX:

1. Internal coordinates. Origin at voxel (0, 0, 0).
   X_{BV}: anterior $\rightarrow$ posterior
   Y_{BV}: superior $\rightarrow$ inferior
   Z_{BV}: right $\rightarrow$ left

2. System coordinates. Origin, directions/values are defined the same as the internal coordinate system but axes names follow Talairach standard:
   X_{SYS}: right $\rightarrow$ left
   Y_{SYS}: anterior $\rightarrow$ posterior
   Z_{SYS}: superior $\rightarrow$ left

3. Talairach coordinates. Axes names like in system coordinates but opposite directions, origin in AC (128,128,128), values defined according to 8 landmarks (AC, PC, LP, RP, SP, IP, AP, PP).
   X_{TAL}: left $\rightarrow$ right
   Y_{TAL}: posterior $\rightarrow$ anterior
   Z_{TAL}: superior $\rightarrow$ left

4. OpenGL coordinates. Like internal (but also shown as system coordinates to the user, except small axes cross in left lower corner of OpenGL (surface) window.

Rotations CCW when looking along positive (OpenGL) axis to origin IN OPENGL. With respect to real Tal axes, the opposite holds. Rot X and Z change sign in VMR.

Fiber coordinates are supported as "BV" or "TAL".

Appendix B

Transformations

B.1 Contents of transformation matrices

Each parameter of an affine transformation has a fixed location in the transformation matrix (see figure B.1).

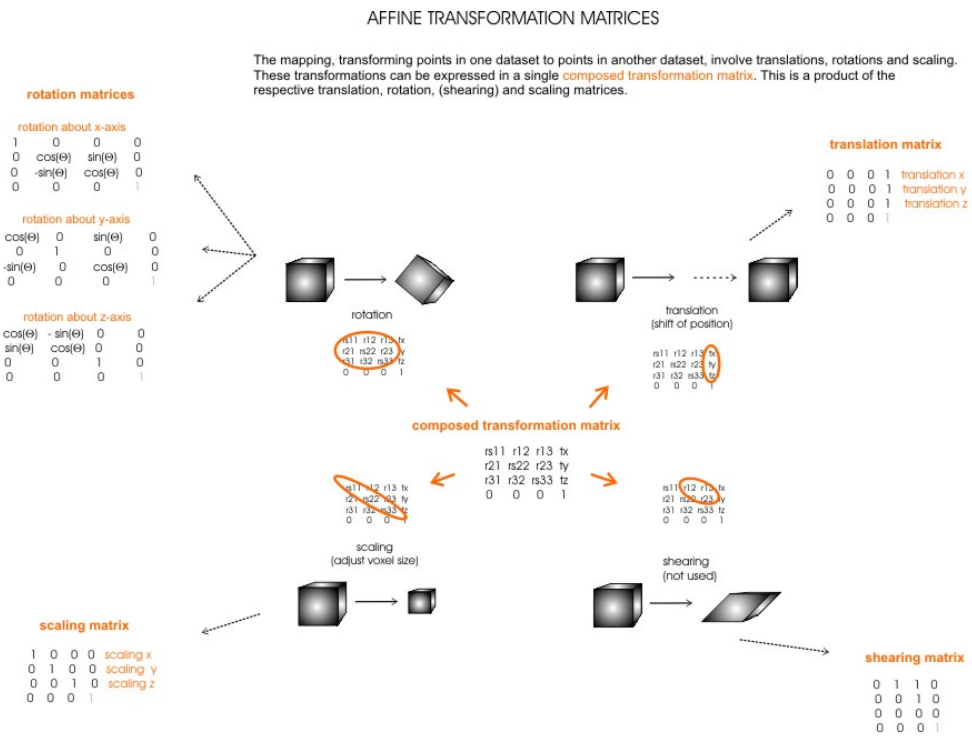


Figure B.1: Each parameter of an affine transformation has a fixed location in the transformation matrix

Useful chapters on spatial transformations can be found in Woods (2000, [8]) and Foley et al (1996, [3]).

B.2 Transformation matrices used during conversion

The following transformations are used in the NIfTI converter. These matrices are fed to the affine transformation module by prof. Thevenaz [6].

0	-1	0	0
0	0	-1	0
1	0	0	0
0	0	0	1

Table B.1: NIfTI → BrainVoyager internal

NIfTI → BrainVoyager internal

0	1	0	0
0	0	-1	0
-1	0	0	0
0	0	0	1

Table B.2: NIfTI → DICOM-BrainVoyager

NIfTI → DICOM-BrainVoyager

NIfTI → BrainVoyager axial This applies to FMR and DMR projects, which are in axial orientation.

1	0	0	0
0	-1	0	0
0	0	1	0
0	0	0	1

Table B.3: NIfTI → BrainVoyager axial

Rest will follow.

B.3 The quaternion in the NIfTI-1 header

Rotations can be stored in a quaternion (see figure B.2). For an explanation, please see the annotated `nifti1.h` file by R.W. Cox (NIH).

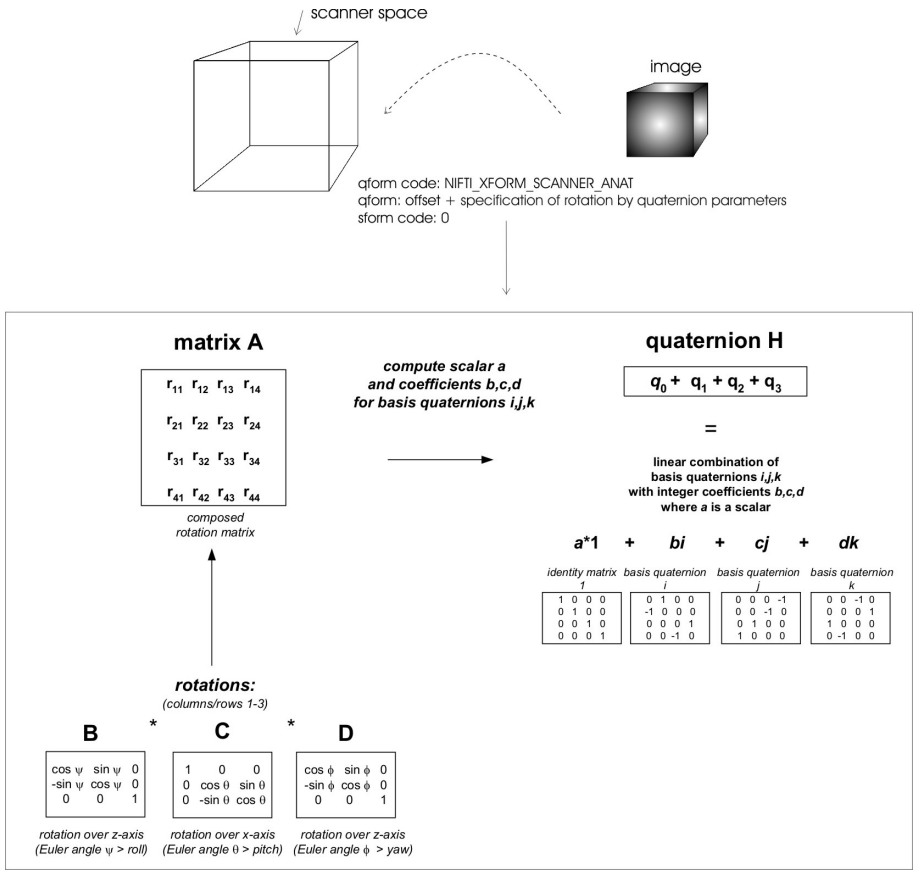


Figure B.2: The qform field in the NIfTI header can store rotations in a quaternion

Appendix C

BrainVoyager file formats

Figure C.1 depicts the relation between the different file formats of BrainVoyager QX. Note: some file formats have been added since creation of this diagram.

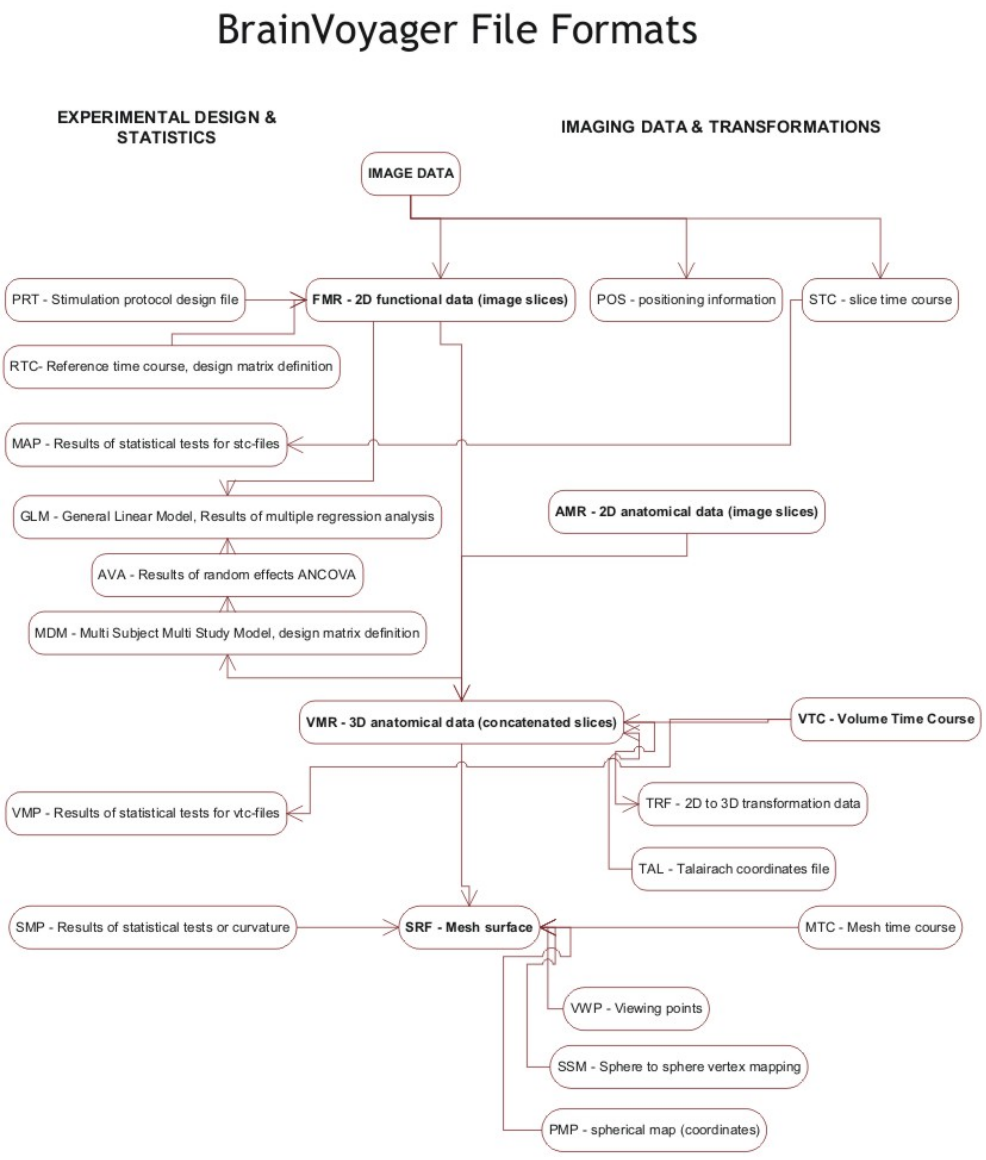


Figure C.1: Entity relationship diagram

Appendix D

Creation of the MNI 305 template

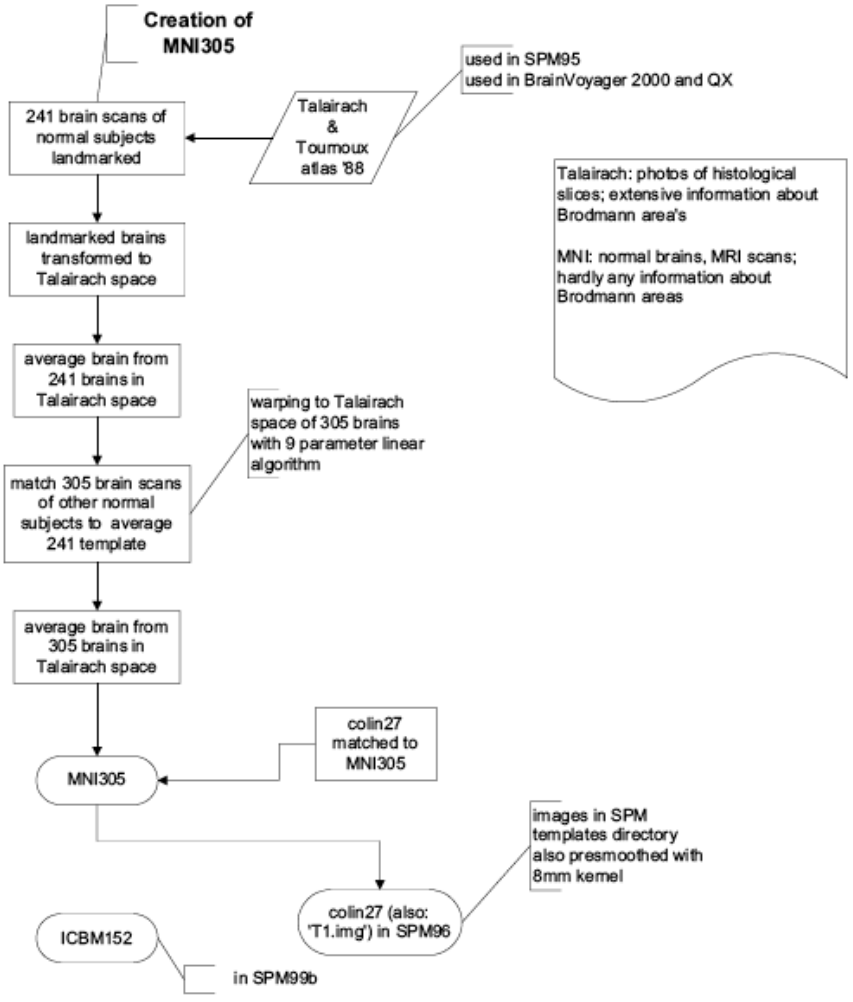


Figure D.1: Construction process of the MNI 305 template

Bibliography

[1] M. Brett, I.S. Johnsrude, and A.M. Owen. The problem of functional localization in the human brain. *Nature Neuroscience*, 3:243–249, 2002.

[2] R.W. Cox et al. A (sort of) new image data format standard: Nifti-1. *Human Brain Mapping*, 25(x):Sx–Sx, 2004.

[3] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes. *Computer Graphics; principles and practice*. Addison-Wesley Publishing Company, Inc., 1996.

[4] R. Goebel, F. Esposito, and E. Formisano. Analysis of fiac data with brainvoyager qx: From single-subject to cortically aligned group glm analysis and self-organizing group ica. *Human Brain Mapping*, 27(5):392–401, 2006.

[5] B. Nijdam and H. van Buuren. *Statistiek voor de sociale wetenschappen*, volume Deel I en II. Alphen aan den Rijn/Zaventem: Samsom Bedrijfsinformatie, 1994.

[6] P. Thevenaz, T. Blu, and M. Unser. Interpolation revisited. *IEEE Transactions on Medical Imaging*, 19(7):739–758, 2000.

[7] P.E. Turkeltaub, G.F. Eden, K.M. Jones, and T.A. Zeffiro. Meta-analysis of the functional neuroanatomy of single-word reading: Method and validation. *NeuroImage*, 16:765–780, 2002.

[8] R.P. Woods. *Spatial Transformation Models*. Academic Press Series in Biomedical Engineering. Academic Press, 2000.