

# **JDTI – Diffusion tensor imaging in Java**

## **What is JDTI?**

JDTI (pronounced “Jay Ditty”) is a Java software tool designed to analyze diffusion tensor imaging (DTI). JDTI is implemented as a plugin to ImageJ, an open source image analysis program ([rsb.info.nih.gov/ij/](http://rsb.info.nih.gov/ij/)).

## **What kinds of images are produced by JDTI?**

JDTI produces three different diffusion-related images, and also two masks

- (1) ADC is the apparent diffusion coefficient map, a measurement of the general tendency of water protons to diffuse in the underlying voxel
- (2) FA is the fractional anisotropy map, a measurement of whether water proton diffusion is equal in all directions (low anisotropy) or whether diffusion proceeds preferentially in certain directions (high anisotropy).
- (3) Directional map is a color map of the primary direction of diffusion (red is right to left, green is top to bottom, and blue is perpendicular (normal) to the plane of the slice.

In addition the brain mask and fractional anisotropy masks used to make these images are produced.

## **What is needed to run JDTI?**

### *Hardware*

JDTI is designed to run on hardware running an operating system that supports ImageJ, a public domain open source Java-based image processing program authored and maintained by Wayne Rasband at the National Institute of Mental Health ([rsb.info.nih.gov/ij/](http://rsb.info.nih.gov/ij/)). Supported operating systems include Mac, Windows, UNIX and Linux platforms. Be aware that the processing of DTI is a processor and memory intensive task. The resources needed to process these images vary depending on the size of the data sets involved. For large data sets, it helps to set the memory requirements of ImageJ correctly (see Installation paragraph below) and install additional RAM if needed.

### *Software*

JDTI runs as a plugin to ImageJ. ImageJ ([rsb.info.nih.gov/ij/](http://rsb.info.nih.gov/ij/)) must be installed before the JDTI plugins. DTI runs synergistically with other plugins available at the ImageJ website (we will note occasions where the use of these plugins is helpful). No other plugins are required. JDTI uses JAMA, a Java matrix package (<http://math.nist.gov/javanumerics/jama/>) to perform matrix calculations. The appropriate version of JAMA is included in the JDTI zip archive.

### *MR imaging data*

JDTI uses the source images from diffusion tensor imaging sequences as input. For typical applications of JDTI, these source images are DICOM part 10 format images (hereafter called “DICOM images”), although other formats supported by ImageJ can be used.

Tensor image sets are assumed to be composed of at least one subset of images obtained without diffusion gradients, followed by multiple subsets of images, each obtained with application of diffusion gradients in a different direction. For calculation of the full diffusion tensor at least 7 subsets of images (six diffusion directions subsets + a subset with no diffusion gradient) must be obtained.

For JDTI, it is assumed that the DTI image data is in *xyzd* order, where *d* is the diffusion direction. This means that each group of images covers the same area of anatomy, and each subset of images either has no diffusion gradient, or has diffusion gradients ordered in specific directions. If your DTI sequence has *zdim* image locations, and one group has been obtained with diffusion weighting and *ngrad* groups have been obtained with diffusion gradients, the entire DTI sequence you import should contain  $zdim * (ngrad + 1)$  images.

HINT: If it is helpful to you, install and use the ImportDicomSequence (<http://rsb.info.nih.gov/ij/plugins/import-dicom.html>) or ImageJ’s File...Import...Image Sequence function to import your images into ImageJ.

HINT: If your imported image stack is in *xydz* rather than *xyzd* order, use Michael Abramoff’s handy Hypervolume Shuffler ([rsb.info.nih.gov/ij/plugins/hypervolume-shuffler.html](http://rsb.info.nih.gov/ij/plugins/hypervolume-shuffler.html)) to place them in *xyzd* order.

### *Information on the direction and magnitude of the diffusion gradients.*

In order to correctly calculate maps derived from DTI data, it is essential to tell JDTI about the direction of the diffusion gradients for each image subset, in order. A text file called TensorDirections.txt is used to specify the diffusion directions. This file needs to be placed in the ImageJ directory. This information unfortunately is often not available on the DICOM header.

### **How can one download JDTI?**

Downloading JDTI is a two step process:

*Step 1. Obtain a user name and password from the lab website.*

If you have not already done so, register to our website, then wait to receive an email confirming your account has been activated.

*Step 2. Request JDTI*

Log in to our website with your username/password, then click on the "Contact Us" link.

In the comments field that appears, type in the keywords “Download JDTI.” Also in the comments field, please give us the following information:

- The institution or company you are affiliated with
- The street address
- City, State and postal code
- Country

You will soon receive email allowing you to download JDTI. Please follow the instructions in the email.

Please read and abide by the software disclaimer. In any publication using JDTI software, please refer to this web address: [dblab.duhs.duke.edu](http://dblab.duhs.duke.edu)

### **How can one install JDTI?**

#### *Step 1. Install ImageJ*

If you have not already installed ImageJ, download and install this free program by following the instructions at [rsb.info.nih.gov/ij/download.html](http://rsb.info.nih.gov/ij/download.html).

#### *Step 2. Adjust memory*

In order to handle large sets of images, maximize the RAM available to ImageJ. To do this, change the memory allocation (-mx option) in the command line launching ImageJ. It often helps to increase the memory allocation option to approximately 2/3 of available RAM (e.g. -mx170m on a 256MB machine). Larger increases may lead to memory thrashing. For more recent versions of ImageJ, you adjust memory from the menus Edit/Options/Memory.

#### *Step 3: Install the JDTI plugins*

Download the source file JDTI\_1.1.zip to your computer. Unzip the contents of the source file into the plugins directory of your ImageJ installation. Make sure to check the boxes “All files/folders in archive” and “Use folder names.” Then restart ImageJ. JDTI should appear as a choice in the Plugins menu.

### **How does one operate the plugin?**

- (1) Open or import your DTI image data and have it as the active stack in ImageJ.
- (2) Place a rectangular region of interest on this stack (Control-A is useful to select the entire image, if needed). JDTI crops the stack to the region of interest (diminishing the area for which maps are made can be helpful to save time, if needed).
- (3) Create a TensorDirections.txt file and place in the ImageJ main directory (see below).
- (4) Launch JDTI from the Plugins menu.

The user will need to provide :

- (1) The b value of the diffusion gradients applied
- (2) Thresholds used to create mask images for the tissue of interest (map values are not calculated for locations where the pixel intensity on B0 images is lower than the threshold value) and for lower limits of fractional anisotropy to be displayed in the tensor directions map. These clean up the visual appearance of the images, but can be set to zero if no thresholding is desired.

After providing the input, click OK, then be patient as the plugin analyzes the DTI data.

### **How should I create the TensorDirections.txt files, and what is its format?**

In JD<sub>DTI</sub>, this is done by opening a text window called TensorDirections.txt (creating this text window can be done easily via Plugins...New (choose the macro function). This text file contains as many lines as you have subsets of images (in other words, if your DTI image data stack has *nsl* slices and *zdim* image positions, this text files should have exactly *nsl/zdim* lines.

In each line, specify the direction the diffusion gradients were applied for the corresponding group of diffusion images. This direction has three components, *dir1*, *dir2* and *dir3*, each of which should be separated by a space. These components represent the magnitude of vector in the x direction (left to right on the images), y direction (top to bottom on the images) and z direction (top to bottom of the stack). These three components together should make a unit vector of length one (in other words,  $dir1^2 + dir2^2 + dir3^2 = 1$ ). If the group of diffusion images has no diffusion gradients applied (this group is called the *B0* images), enter “0 0 0” on the corresponding line.

**Note:** Be sure to save the TensorDirections.txt file to the main ImageJ directory before operating the plugin!

Figure 1 shows an example TensorDirections.txt file from a 25-direction diffusion tensor imaging sequence. In the illustrated case, the B0 image is the first set of images.

### **How do I find out what diffusion directions are used in my DTI pulse sequence?**

As a rule, this information is not present on the DICOM header (unfortunately). Your MR manufacturer technical support should be able to give you this information.

Please be aware that lists of gradient direction components provided by manufacturers may not be in the order needed for use in JD<sub>DTI</sub>. For JD<sub>DTI</sub>, these should be in x (row), y (column) and z (slices) order. Manufacturers may list the components in the phase, frequency and slice direction.

HINT: to find out whether the phase direction is in the x (row) or y (column) direction, check the DICOM header information (Control-I) at field 0018,1312.

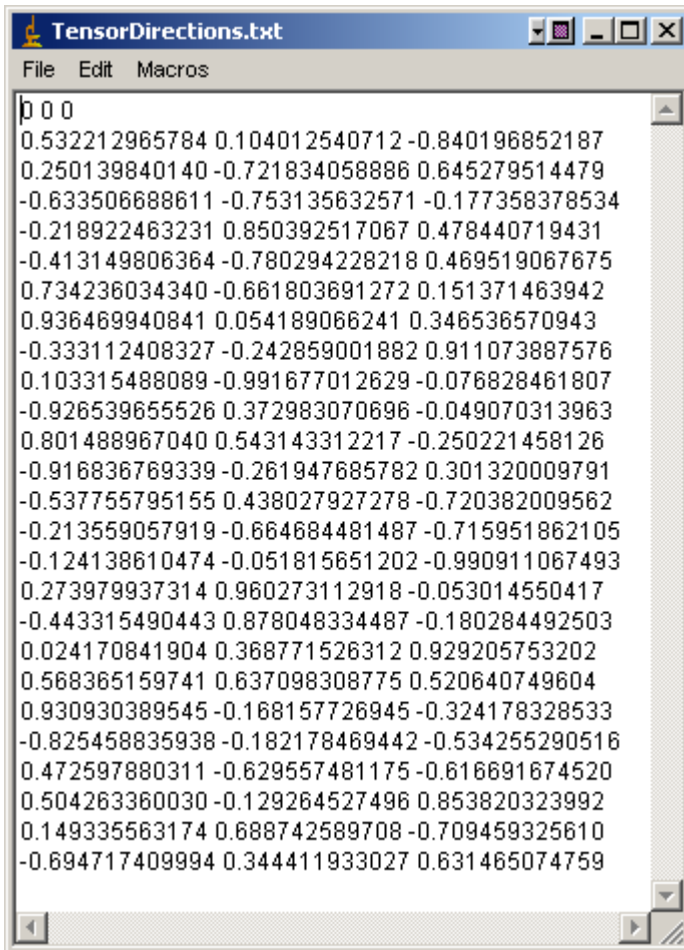


Figure 1: An example of a TensorDirections.txt file window.

### Is the JD TI plugin macro-able?

Yes.

### How is the ADC map generated?

For background, please see LeBihan D et al; JMRI 13:534-5546 (2001) [<http://noodle.med.yale.edu/~mjack/papers/bihan2001.pdf>].

In brief, for each voxel a diffusion tensor matrix is derived from the imaging data

$$\mathbf{D} = \begin{bmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{yx} & D_{yy} & D_{yz} \\ D_{zx} & D_{zy} & D_{zz} \end{bmatrix}$$

This tensor matrix is diagonalized to derive three eigenvalues  $\lambda_1, \lambda_2$ , and  $\lambda_3$  and three eigenvectors  $\varepsilon_1, \varepsilon_2$ , and  $\varepsilon_3$ :

$$\mathbf{D}' = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \bullet \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix}$$

The ADC is taken as the mean diffusivity  $MD$ :

$$MD = \frac{\lambda_1 + \lambda_2 + \lambda_3}{3}$$

The units of ADC depend on the units of b-value provided. We expect values of about 1000 sec/mm<sup>2</sup>. In this case, ADCs are given in mm<sup>2</sup>/sec. If you use values in min/cm<sup>2</sup>, then ADC will be in cm<sup>2</sup>/min. Note: please do not use a b-value less than one!

### How is the FA map generated?

The FA map is generated using the equation:

$$FA = \frac{\sqrt{(\lambda_1 - MD)^2 + (\lambda_2 - MD)^2 + (\lambda_3 - MD)^2}}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}$$

The pixel intensity values are between zero (isotropy) and one (extreme anisotropy).

### How is the directional map calculated?

We use a color scheme of red for left to right, green for top to bottom of image and blue for top to bottom of slices. Although many color schemes have been proposed, this color scheme appears to be the most common type applied to axial images (for example, see <http://www.dtiatlas.org/>).

The color components are created from the eigenvector associated with the largest eigenvalue (we will designate this eigenvector  $\varepsilon_1$ ).  $\varepsilon_1$  has components in the x-, y- and z-directions, which we will designate  $\varepsilon_{1x}$ ,  $\varepsilon_{1y}$ , and  $\varepsilon_{1z}$  respectively. We create red, green

and blue color images using the following equations, then combine the three images to create a single RGB image:

The red image is related to:  $\left| \arctan\left(\frac{\epsilon_{1x}}{\epsilon_{1y}}\right) \right|$

The green image is related to:  $\left| \arctan\left(\frac{\epsilon_{1y}}{\epsilon_{1x}}\right) \right|$

The blue image is related to:  $\left| \arctan\left(\frac{\epsilon_{1z}}{\sqrt{\epsilon_{1x}^2 + \epsilon_{1y}^2}}\right) \right|$

**Does this plugin perform fiber tract mapping?**

No.

**Does this plugin accept DTI images with multiple b-values?**

No.

**What are the advantages of using JDIT compared to other alternatives? (And what are the other alternatives?)**

There are a reasonably large number of software packages available that can analyze DTI images, each with advantages and disadvantages. The advantage of JDIT lies in a combination of:

- (1) Multiplatform availability
- (2) Ease of installation
- (3) Macro ability
- (4) Integration into the ImageJ platform, where ROIs can be generated and saved, and a variety of image analysis techniques can be used.

Other alternative projects that have reasonably well developed or planned tensor analysis packages include:

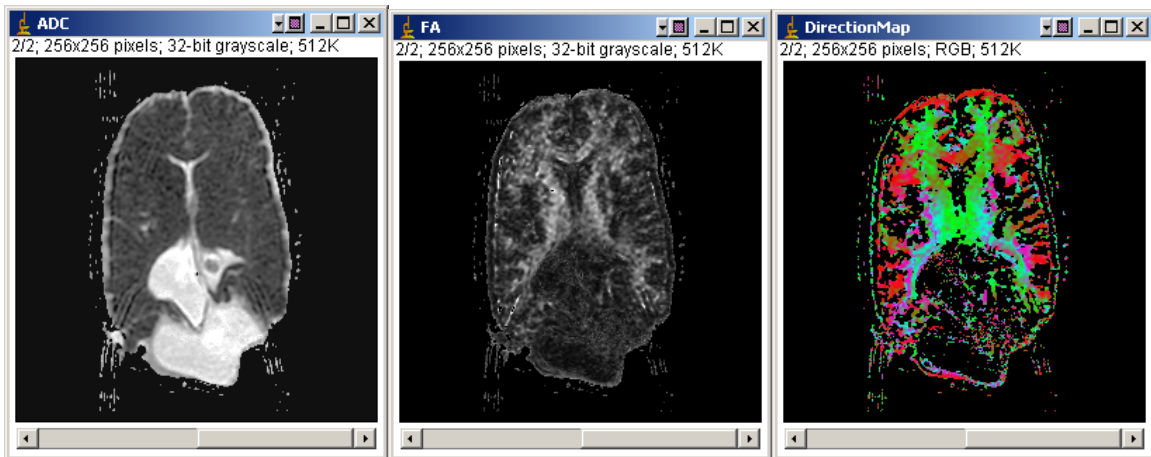
- <http://cmrm.med.jhmi.edu/DTIuser/DTIuser.asp>
- <http://www.ia.unc.edu/dev/download/>
- <http://software.sci.utah.edu/powerapps.html>
- <http://neuroimage.yonsei.ac.kr/dodti/>
- <http://www.ut-radiology.umin.jp/people/masutani/dTV.htm>
- <http://people.csail.mit.edu/lauren/dtmri.html>

### How can I learn more about diffusion tensor imaging?

Here are some helpful links:

- <http://www.dtiatlas.org/>
- [http://splweb.bwh.harvard.edu:8000/pages/papers/martha/DTI\\_Tech354.pdf](http://splweb.bwh.harvard.edu:8000/pages/papers/martha/DTI_Tech354.pdf)
- <http://www.aans.org/education/journal/neurosurgical/july03/15-1-4.pdf>
- <http://noodle.med.yale.edu/~mjack/papers/bihan2001.pdf>
- <http://sirl.stanford.edu/dwi/maj/>
- <http://128.104.229.200/fmri/dti.html>

### Can you show some sample images?



### Acknowledgements:

Special thanks go to James MacFall, Ph.D. Duke University Medical Center  
Thanks also to Bryan Mock, Jim Colsher, Jerry Dahlke and Anila Lingamneni, GE Medical Systems