# Diffusional Kurtosis Estimator (DKE)

# User's Guide

Version 2.6.0

Release date: February 2015

# Contents

Diffusional Kurtosis Estimator (DKE)

User's Guide

Version 2.6.0

Release date: February 2015

Diffusional Kurtosis Estimator (DKE) is a software tool for post-processing diffusional kurtosis imaging (DKI) datasets that includes a suite of command-line programs along with a graphical user interface (GUI). DKE currently supports 32- and 64-bit Windows platforms. DKE generates a set of kurtosis (axial, mean, radial) parametric maps with a given set of diffusion weighted images acquired from a valid DKI protocol. Diffusivity (axial, mean, radial) and fractional anisotropy maps using either DKI or diffusion tensor imaging signal models are also calculated in the processing. In the latest version of DKE, two extra parametric maps (KFA and mean kurtosis tensor) were added. DKE features include: DICOM, NIfTI and Bruker format support, interactive (GUI) as well as batch mode (command-line) processing, and rigid-body motion correction. DKE implements the methods described in the following paper:

Tabesh A, Jensen JH, Ardekani BA, and Helpern JA. Estimation of tensors and tensor-derived measures in diffusional kurtosis imaging. Mag Reson Med. 2011 Mar; 65(3):823-36.
http://www.ncbi.nlm.nih.gov/pubmed/21337412

If you use DKE in a publication, please cite this paper in addition to standard DKI references.

For questions or remarks please contact us at dki@musc.edu .

# Getting Started

## 1. Installation

The software is readily available at our website: http://academicdepartments.musc.edu/cbi//dki/DKE/dke_download.htm. After registration you will receive an email with a link where the software can be downloaded. Note that DKE requires the installation of the MATLAB Compiler Runtime 2012a (MCR version 7.17) available at: http://www.mathworks.com/products/compiler/mcr/. After following the instructions of installation, a folder called DKE will be created in your "Program files" directory.

# Running DKE

There are 2 ways of running DKE: (1) Through the GUI (Graphical User Interface), by double clicking on DKEGUI.exe (DICOM and NIfTI) or (2) using the command prompt (DICOM, NIfTI and Bruker).

## 1. Running DKE using the GUI

The [video tutorial](#) gives a brief overview of the basic steps for processing DKI data with DKE.

### 1.1 File menu

**Load all DICOMs in a directory**

Load a DKI dataset in DICOM format. All images must be stored in a single directory (no subdirectories). DICOM data may consist of multiple series. DKI series must consist of one b0 (i.e., b=0) image and diffusion-weighted images for nonzero b-value 1, b-value 2, b-value 3...etc. to b-value n in the dataset. All DKI series must use identical sets of b-values and gradient directions. An optional series DKIB0 must exclusively consist of extra b0 images. For examples of supported DICOM protocols, see [recommended protocols](#) for various scanners.

**Load a 4D NIfTI image**

Load a DKI dataset as a single 4D NIfTI image. The 4D image must consist of a single b = 0 image followed by subsequent diffusion-weighted images (B0, all B1000's, all B2000's). When different DKI series need to be included in the calculations of the parametric maps you will have to create your own series average before running DKE.

**Save parameters**

Save the current parameter settings in the GUI to a text file.

### 1.2 Help menu

**DKI Website**

Link to DKI website.

**View Help**

Displays Help file

**About DKE**

Displays version and copyright information.

**Exit**

Exit the program.

## 1.3 Processing

Processing is initiated by loading in 4D NIfTi file or a set of DICOMs. If DICOM images are loaded, the DKI image series must be selected manually. This step helps DKE identify the name of relevant DICOM series for processing. After the DICOMs are loaded, proceed to click the button 'Next' in the bottom left corner of the main window. A window with three different fields will pop up.  The first column contains a list of all DICOM series in the directory. Each DICOM series is identified with its series description field. The diffusion weighted images will for example be designated as DKI1, DKI2... DKI$N$ and DKIB0. In the top field on the right side (DICOM Series Description) is where the user will specify which DICOMs will be used for estimating the parametric maps. In the right bottom field (B0 Image Series Name), a series of extra B0 images may be added. Series in the left column can be selected by using the 'Add Series' buttons. After selecting all relevant image series click 'Accept'.

Processing is initiated with the current GUI parameter settings. A copy of the settings is saved in the data directory in a file call DKEParameters.dat.
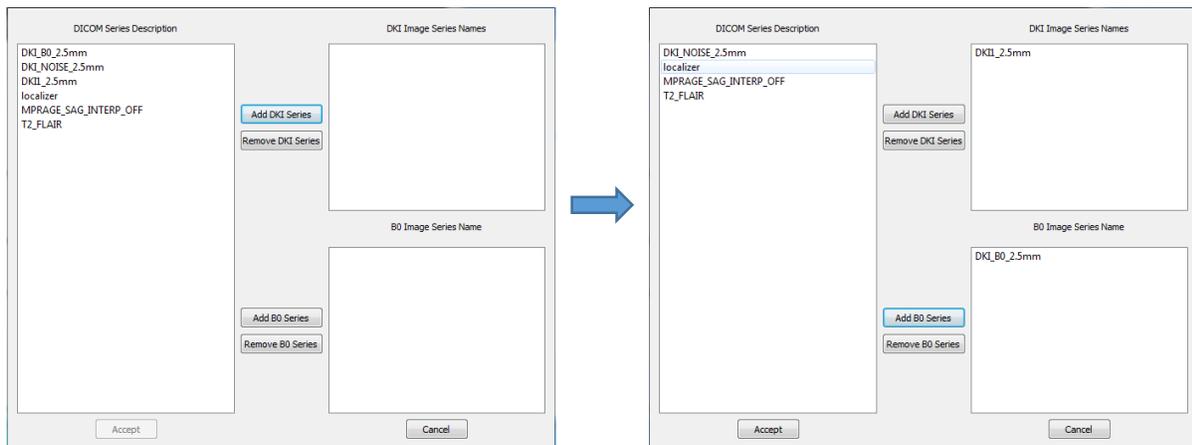


## 1.4 Basic Settings

**B = 0 Threshold**

Background threshold. Voxels with b = 0 intensity values above this threshold (after an optional connected component analysis) are processed. See batch processing on how to turn off the connected component analysis.

**B-values**

Diffusion weightings (b-values) in s/mm$^2$. The first element of the vector must always be 0.

**Gradient Vectors**

Select the gradient table from a list of available tables or specify a custom table. A custom table can be a '.dat' or '.txt' file. Each row corresponds to a gradient vector [$G_x$, $G_y$, $Gz$]. When specifying a custom table, remove gradients that could correspond to the B0 image [0, 0, 0]. When using a 4D NIfTi, make sure that the 4D order corresponds to the order of the gradient table. If a DICOM set is used, DKE will order the images according to the information stored in the DICOM tag-(0018, 0024) 'SequenceName'.

**DWI Spatial Smoothing**

Check this box to apply spatial smoothing to diffusion-weighted images prior to estimation of parametric maps (default: checked). This is accomplished with a linear filter with a Gaussian kernel.

**FWHM**

Full width at half maximum in mm for the Gaussian smoothing filter. An isotropic FWHM of roughly 1.25 times the voxel size is recommended.

**Median Filtering**

Selective median filtering that is applied to parametric maps (default: strong filtering). An outlier removal median filter with a 3x3x3 voxel window is applied to voxels that violate the minimum directional kurtosis constraints. Strong filtering indicates that filtering will be applied to voxels with any constraint violation. Weak filtering indicates that median filtering will be applied to voxels with less than 15 unviolated constraints. No filtering will disable the median filter. When it is suspected that the images are of lower quality it is recommended to set the filtering option to weak.

## 1.5 Advanced Settings

### 1.5.1 DKI Processing

Check this box to enable DKI processing (default: checked).

**DKI Fitting Method**

Select the DKI model fitting method. Select directional to apply directional signal fitting or tensor to apply tensor fitting (default: tensor).

Directional fitting methods are the linear unweighted and linear weighted schemes (default: weighted). Weighting is based on the diffusion signal magnitude.

Tensor fitting methods are the constrained linear weighted, unconstrained linear unweighted, and unconstrained nonlinear schemes (default: constrained).

Check the robust fitting box to apply a RESTORE-type algorithm (Chang et al. (2005)) (outlier detection and removal followed by tensor refitting) for tensor fitting. Tolerance to outliers is controlled with the user-supplied noise tolerance parameter.

## 1.5.2 DTI Processing

Check this box to enable DTI processing (default: unchecked).

**DTI Fitting Method**

Select the DTI model fitting method. Select directional to apply directional signal fitting or tensor to apply tensor fitting (default: tensor).

Directional fitting methods are the linear unweighted and linear weighted schemes (default: weighted). Weighting is based on the diffusion signal magnitude.

Tensor fitting methods are the linear weighted and linear unweighted schemes (default: weighted).

Check the robust fitting box to apply a RESTORE-type algorithm (Chang et al. (2005)) (outlier detection and removal followed by tensor refitting) for tensor fitting. Tolerance to outliers is controlled with the user-supplied noise tolerance parameter.

Specify the nonzero b-value for DTI fitting in the B-value box.

**Co-register Scans**

Check this box to enable rigid-body co-registration of DKI series. This only applies to DICOM images.

**Interpolate**

Check this box to apply linear interpolation to the parametric maps (default: checked). See batch processing for details on how to change the interpolation method (default: trilinear).

## 2. <u>Running DKE in the command window / Batch Processing</u>

DKE allows for batch processing of DKI data using a file that uses the MATLAB syntax. The batch file format is described below. In your program files you can find an example file called dke_parameters.dat.

## 2.1 Command Line

Launch a command window and start processing by using the command (with user specific pathways):

> C:/Users/Program Files/DKE/dke    C:/Users/MyDocuments/Projects/dke_parameters.dat

## 2.2 Batch file

## 2.2 1 Paths

**studydir**

Root folder for data from all study subjects

**subject_list**

Cell array of subject sub-folders within the root folder studydir

## 2.2.2 Preprocessing parameters

**preprocess_options.format**

Input image format ('dicom', 'nifti' or 'Bruker').

**preprocess_options.navg**

Number of DKI series. Each DKI series must contain a b = 0 image followed by diffusion-weighted images. All series must use identical acquisition protocols (i.e., same diffusion weightings and gradient directions). This setting is only used when preprocess_options.format = 'dicom' or 'bruker'.

**preprocess_options.extra_b0**

Whether (1) or not (0) there is an additional b = 0 series. This setting is only used when preprocess_options.format = 'dicom'.

**preprocess_options.coreg_flag**

Whether (1) or not (0) to perform 6-parameter rigid-body co-registration between diffusion-weighted images (default: 1). This setting is only used when preprocess_options.format = 'dicom' or 'Bruker'.

**preprocess_options.series_description**

List of series descriptions from the DICOM image headers. This is used only when preprocess_options.format = 'dicom'.

**preprocess_options.fn_nii**

4D nifti image file name. This setting is only used if preprocess_options.format = 'nifti'. Co-registration is not performed between the diffusion-weighted images.

## 2.2.3 Imaging diffusion weightings and directions

**bval**

Diffusion weightings (b-values) in s/mm² units. The first element of vector must always be 0.

**ndir**

Number of gradient directions. If a scalar value is specified, the number of gradient directions for all b values is set to the scalar. Otherwise, a 1-by-(nbval-1) vector should be specified, with each element of the vector corresponding to the number of gradient directions for a nonzero b-value.

**fn_gradients**

Text file containing the gradient vectors. Each row corresponds to a gradient vector. If a different gradient set was used for each b-value, a 1-by-(nbval-1) cell array of file names should be specified, with each cell specifying the gradient file name for the corresponding nonzero b-value.

**idx_gradients**

Indices of gradient directions to be used for DKI map estimation. Unacceptable diffusion-weighted images (e.g., those affected by motion- or patient table vibration-induced diffusion signal loss) can be excluded here. There must be as many cells as there are nonzero b-values, with each cell corresponding to a nonzero b-value.

**idx_1st_img**

Index of the first diffusion-weighted nifti image for each b-value (typically 0 or 1) (default: 1). This is used when preprocess_options.format = 'dicom' and is rarely changed.

### 2.2.4 Constraints on directional kurtoses

**Kmin**

Constraint on minimum directional kurtosis (cf. Tabesh et al. (2011)) (default: 0). This is rarely changed.

**NKmax**

Parameter defining the constraint on maximum directional kurtosis (cf. Eq. 6 in Tabesh et al. (2011)). This is rarely changed.

### 2.2.5 Thresholds on output kurtosis maps

**Kmin_final, Kmax_final**

Lower (Kmin_final) and upper (Kmax_final) thresholds applied to output (mean, axial, and radial) kurtosis maps. The default values are 0 and 3, respectively.

### 2.2.6 Tissue/background segmentation

**T**

Background threshold. Voxels with b = 0 intensity values above this threshold (after an optional connected component analysis) are processed.

**find_brain_mask_flag**

Whether (1) or not (0) to apply connected component analysis to refine the binary brain mask obtained by applying threshold T to b = 0 image (default: 1).

### 2.2.7 DKI fitting method and parameters

**dki_method.no_tensor**

Whether (1) or not (0) to estimate parametric maps using directional signal fits instead of the default tensor fit of Tabesh et al. (2011) (default: 0). Directional fits only allow estimation of mean diffusivity and mean kurtosis. Directional fit are not recommended unless the number of acquired gradient directions is less than 15.

**dki_method.linear_weighting**

Unweighted (0) or weighted (1) linear least-squares (default: 1).

**dki_method.linear_constrained**

Unconstrained (0) or constrained (1) linear least-squares (cf. Tabesh et al. (2011)) (default: 1).

**dki_method.nonlinear**

Unconstrained nonlinear least-squares (default: 0).

**dki_method.linear_violations**

Whether (1) or not (0) to generate maps of constraint violations (default: 0). Intensity of each voxel in the violation maps represents the proportion of constraints on directional diffusivities ('d_viol' map) and kurtoses ('kmin_viol' and 'kmax_viol' maps) violated by the unconstrained linear least-squares solution. The unconstrained solution used to obtain the violation maps will depend on dki_method.linear_weighting and dki_method.robust_option parameters.

**dki_method.robust_option**

Robust fitting option (default: 0): (0) do not use robust fitting; (1) RESTORE-type algorithm (outlier detection and removal followed by tensor refitting) with a user-supplied 'noise tolerance level' (dki_method.noise_tolerance) expressed as a fraction of the diffusion signal magnitude.

**dki_method.noise_tolerance**

Threshold for outlier detection (used when dki_method.robust_option = 1); the diffusion signal for a gradient direction is declared an outlier if abs(log(diffusion signal) - log(predicted diffusion signal)) > dki_method.noise_tolerance.

## 2.2.8 DTI fitting method and parameters

**dti_method.dti_flag**

Whether (1) or not (0) to estimate DTI parametric maps based on the DTI (in addition to DKI) signal model (default: 0).

**dti_method.dti_only**

Whether (1) or not (0) to only estimate DTI parametric maps (no DKI map estimation) (default: 0).

**dti_method.no_tensor**

Whether (1) or not (0) to estimate parametric maps using directional signal fits instead of the default diffusion tensor fit (default: 0). Directional fits only allow estimation of mean diffusivity. Directional fits are not recommended unless the number of acquired gradient directions is less than 6.

**dti_method.linear_weighting**

Unweighted (0) or weighted (1) linear least-squares (default: 1); weighting is based on the diffusion signal magnitude.

**dti_method.b_value**

Nonzero b-values used for DTI map estimation.

**dti_method.directions**

Indices of gradient directions to be used for DTI map estimation. Unacceptable diffusion-weighted images (e.g., those affected by with motion- or patient table vibration-induced diffusion signal loss) can be excluded here. There must be as many cells as there are nonzero b-values, with each cell corresponding to a nonzero b-value. Indices are relative to the indices specified in idx_gradients.

**dti_method.robust_option**

Robust fitting option (default: 0): (0) do not use robust fitting; (1) RESTORE-type algorithm (outlier detection and removal followed by tensor refitting) with a user-supplied 'noise tolerance level' (dti_method.noise_tolerance) expressed as a fraction of the diffusion signal level.

**dti_method.noise_tolerance**

Threshold for outlier detection (used when dti_method.robust_option = 1). The diffusion signal for a gradient direction is declared an outlier if abs(log(diffusion signal) - log(predicted diffusion signal)) > dti_method.noise_tolerance.

## 2.2.9 Diffusion-weighted image filtering

**fwhm_img**

Full width at half maximum (FWHM) (in mm) of Gaussian kernel for smoothing diffusion-weighted images (default: 1.25 * voxel size). A zero value indicates no smoothing.

### 2.2.10 Rician noise subtraction

**fn_noise**

User-supplied map of Rician noise level (default: '' (i.e., none)). Noise subtraction is based on the method of McGibney and Smith (1993). Estimated signal s_hat at each voxel is obtained as s_hat = (s^2 - n^2) ^0.5, where s is the measured signal and n is the noise level at that voxel.

**fwhm_noise**

FWHM (in mm) of Gaussian kernel for smoothing noise image. A zero value indicates no smoothing (default: 0).

### 2.2.11 Parametric map filtering

**median_filter_method**

Selective median filtering applied to voxels that violate the minimum directional kurtosis constraints (default: 2): (0) no filtering; (1) weak filtering (voxels with less than 15 unviolated constraints will be filtered); (2) strong filtering (voxels with any constraint violation will be filtered).

### 2.2.12 Parametric map interpolation

**map_interpolation_method.flag**

Whether (1) or not (0) to interpolate the parametric maps (default: 1).

**map_interpolation_method.order**

Interpolation polynomial order: (0) nearest neighbor; (1) trilinear; (2 and up) higher order (default: 1).

**map_interpolation_method.resolution**

Target resolution (isotropic) in mm for the interpolated maps (default: 1 mm).