Tutorial For The FMREyemove Plug-In

Ingo R. Keck
ingokeck@ingokeck.de

September 24, 2012

Abstract

FMREyemove is a easy to use SPM 5 plugin that allows to directly estimate the subject's point of gaze from the functional magnetic resonance imaging (fMRI) data - using nothing more than the eye movement artefacts contained in the actual fMRI data. No extra eye-tracking device is necessary, however the temporal resolution is restricted to the temporal resolution of the fMRI data.

1 Installation

Simply download the plugin from it's webpage at sourceforge [1] and copy it into the *toolbox*-folder of SPM. You can then call it from the toolbox menu within SPM, or start it directly from the matlab prompt.

FMREyemove uses file-loading and the orthview routines from SPM and will not work without an SPM install. It also uses the JADE algorithm from Jean-Francois Cardoso, published on `http://www.tsi.enst.fr/~cardoso/Algo/Jade/jadeR.m`. In theory every ICA algorithm can be used, however, the author would strongly recommend JADE as a very stable and reliable choice.

2 How it works

FMREyemove works by calculating a set of eye-movement artefact filters form the fMRI data of a subject. These filters are then used to estimate the point of gaze from other fMRI data sets of the same subject.

The application of the plug-in falls into two parts:

[1] `http://fmreyetrack.sourceforge.net/`

1. Eye movement filter detection

2. Eye movement estimation

The filter detection works best if data from a special eye movement normalisation session is available (e.g. a short experiment where the subject has to look at the centres of all four quadrants of the screen). The time course of horizontal and vertical eye movement from this session can be used for a automated detection (see next section).

3 Usage

The initial window (figure 1) already shows all the buttons of the program, however most of them remain inactive if the plug-in does not already have the necessary information for them to function. You can let yourself be guided by the active buttons in the window.

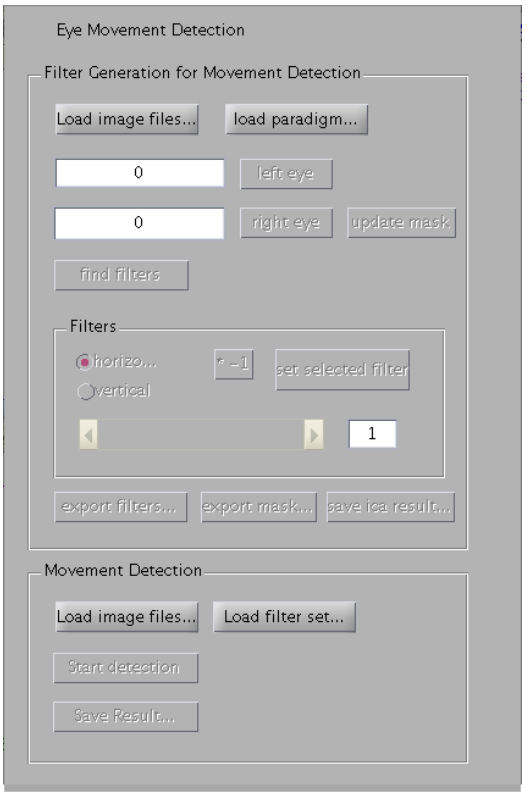
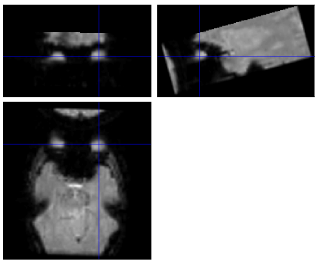


Figure 1: The initial state of the plug-in window.

Now lets start our step-by-step tutorial:

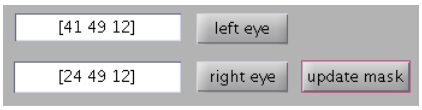
1. Start the plug-in from within SPM

2. Click on the button *Load image files...* A SPM file-select window will open where you can select the fMRI images for the filter detection part just as you would in SPM. The plug-in then loads the first image and displays it in the SPM figure window:

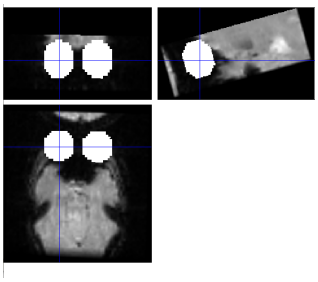


3. If you have the time courses of eye movement from the filter session, load them clicking on the *load paradigm...*-button. A matlab file-select window will open that allows you select the .mat file that contains the eye position as $n \times 2$ matrix, where n is the number of scans of the filter session. Doing this allows the plug-in to automatically detect the correct filters for horizontal and vertical eye movement in the filter detection step. If you do not have this information, for example because it is an old data set, you will have to manually select the best filters.

4. Now you can select the eye position in the orthview window. Move the blue cross to the center of the right eye, then press *right eye* on the plug-in. The plug-in will extract the eye position from the orthview window and display it as co-ordinates. Repeat this with the left eye and the *left eye*-button.

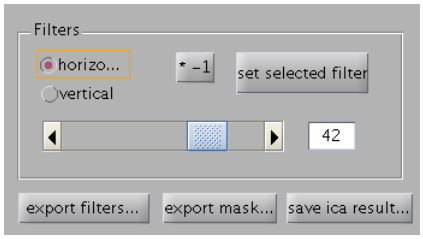


After you have finished, press *update mask* to show the selected mask in the orthview window. You should see something like this:

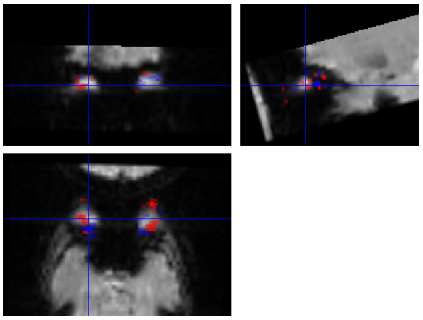


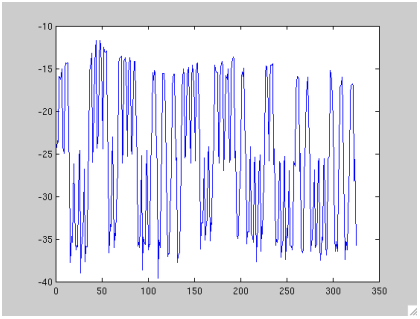
If you are not satisfied with the position of the mask simply select a better position for the centres of the eyes and click again on *update mask*.

5. After the selection of the eye region of interest mask click on the *find filters*-button to start the filter detection. The plug-in now loads all images into the memory and starts the Independent Component Analysis (ICA) to find the eye movement filters. After it has finished it will show the results in the *Filters*-frame:



6. In the *Filters*-frame you now can use the radio buttons *horizontal* and *vertical* to switch between the filter sets that the plug-in has detected for horizontal and vertical eye movement.
   Use the slider to look at the other filters that have been detected. The button $\star - 1$ will inverse the actual selected filter.
   If you want to define a new filter as *horizontal* or *vertical*, simply select the filter and click on the *set selected filter*-button. This allows the manual filter selection without a paradigm .mat file.
   The orthview window will show the spatial distribution of the selected filter, where blue blobs represent negative values and red blobs represent positive values of the filter. A figure window will plot at the same time the extracted time course of movement from the data. Choose filters that represent the actual eye movement in the data.

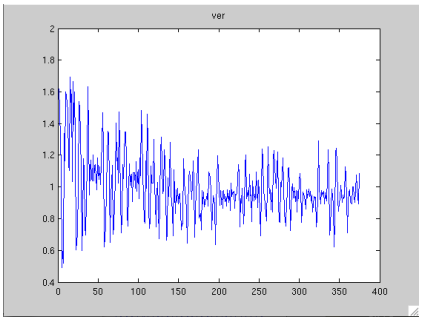


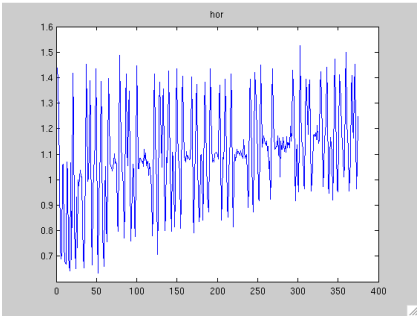


7. If you are satisfied with the selected filters, save them for late analysis using the *export filters...*-button.

8. After the filter selection you can immediately start with the eye movement estimation in your image data of the same subject. If you already have the eye movement filters for this subject from an earlier analysis, load them using the *Load filter set...*-button. To load the image data, use the *Load image files...*-button.

9. After the selection of the image data, click on the *Start detection*-button to start the eye movement estimation. The plug-in will then load the image data and apply the selected filter set. The result will be displayed as two graph windows:





10. Save the movement estimation result as .mat file using the *Save Result...*-button.

4 Comments, Bugs, Etc...

Feel free to send comments or bug-reports to the authors email *ingokeck@ingokeck.de* or use the bug-reporting facilities and forums on `http://fmreyetrack.sourceforge.net/`.