RETIF - Real-Time fMRI

RETIF is a Matlab toolbox that will allow you to run rea-time fMRI experiments.

Some features of the toolbox depend on other (freely available) code, developed by others.

WindowAPI: (<https://se.mathworks.com/matlabcentral/fileexchange/31437-windowapi>) A copy is included in the RETIF installation, so you don’t need to download it separately.

SPM:

Here, a real-time fMRI scan run into two phases, pre-feedback and feedback. In the prefeedback phase there is no neurofeedback information sent to the subject. This division is simply for structuring purposes and in practice the prefeedback phase can be of zero length. However, usually some time is needed for e.g. estimating a signal baseline.

# An experiment in short

An experiment file is opened, defining where to look for new image files and what functions to call once a new volume is available etc. Clicking *Start* starts checking for new files. As soon as a new volume is available, it is read and then processed according to your specifications. The processing usually includes motion correction, detrending, normalization etc. After the processing, usually some value(s) are computed and used for updating a visual stimuli or sending a command to a device etc.

# User defined files

There are (at least) four files that need to be created by the user: the *experiment file,* the *startup function,* the *preparation function,* the *prefeedback function* and the *feedback function.*

## experiment file

This is a mat-file with variables that defines what files should be called, what classifier to use, how many volumes the prefeedback and feedback phases consist of etc.

## startup function

This function is called directly after the experiment file has been opened. Here you can prepare the GUI and pre-allocate variables etc. Basically, do all things here that can be done in advance so you don't waste precious time during scanning.

## prefeedback function

This function is called once a new image volume is available during the prefeedback phase. Note that an exception is the very last volume in the prefeedback phase.

## preparation function

This function is called for the last volume in the prefeedback phase. The reason for a separate function is that the prefeedback data that has been collected until now should be analyzed before the feedback phase starts.

## feedback function

This function is called once a new image volume is available during the feedback phase. Here the new data is used for updating e.g. a stimuli.

# log file

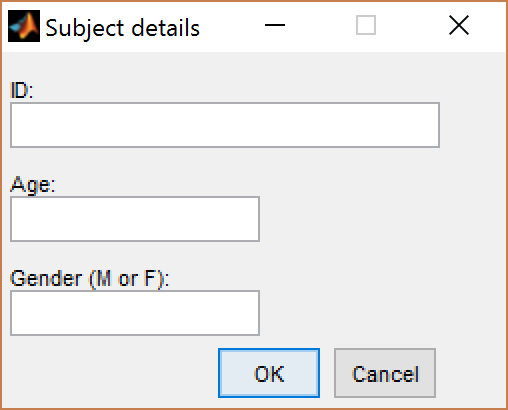
When you start RETIF a GUI is opened, see Figure 1. There is room for subject ID, age and gender.

Figure 1. GUI for filling in subject details.

A log-file with the name RETIFlog\_*ID*.log is created and is used for logging events and timings during the experiment.

You don't have to fill in anything, but a log-file is still created.

# RETIF gui

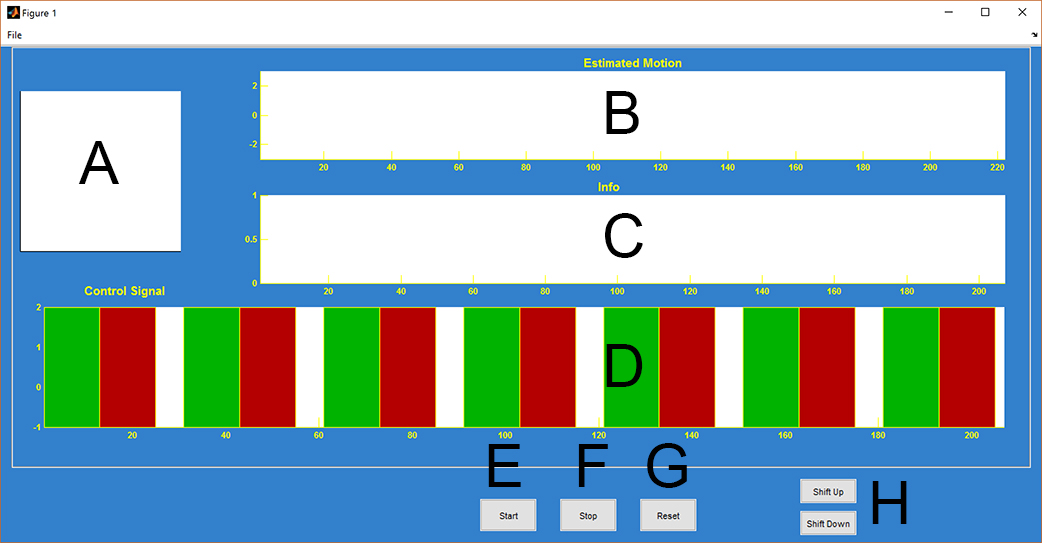
Figure 2 shows how the RETIF interface can look after you have opened an experiment file.

Figure 2. The RETIF user interface.

**A**: Axis for 3D rendering of brain and ROIs.

**B**: Plots showing the estimated motion.

**C**: Nothing, unless specified by the user.

**D**: Plots of control signals. Different colours represent blocks of different tasks.

**E**: Starts the timer, and looks for new image volumes.

**F**: Stops the timer. Also closes the log-file.

**G**: Nothing at the moment....

**H**: Adjusts the baseline up or down. Preferably, they will not be needed.

❶: Startup function❷: Prefeedback function❸: Preparation function❹: Feedback function

**Example 1**

LOCALIZER

PRE-FEEDBACK

FEEDBACK

❶ ❷ ❸ ❹

The localizer data is collected in a separate run, or maybe even in a separate session.

The *Startup function* might e.g. contain:

* Reading in premade ROIs.
* Reading in pre-processed localizer data.
* Training the classifier.

The *Preparation function* might e.g. contain:

* Updating the mean and variance values for normalization.

**Example 2**

LOCALIZER/PRE-FEEDBACK

FEEDBACK

❶ ❷ ❸ ❹

The localizer data is collected during the pre-feedback part.

The *Startup function* might e.g. contain:

* Preparing plots and graphics.

The *Prefeedback function* might e.g. contain:

* Online statistical analysis, such as an incremental GLM analysis.

The *Preparation function* might e.g. contain:

* Computing the mean and variance values for normalization.
* Creating ROIs from the statistical maps.
* Training the classifier.

# Folders

## experiment files

Contains examples of user defined files for experiments. Please note, most of these might be outdated!

## RT\_Data

Class for holding the MR data.

**Data = RT\_Data(size\_data, size\_t [, selector] [,mask])**

**size\_data** = The matrix size of one image volume.

**size\_t** = The number of image volumes you will collect.

**selector** = Vector of length size\_t stating what volumes should be included in sdv and mean estimations. Time points with a non-zero value in selector are included.

**mask** = Binary mask with dimensions size\_data, selecting the data voxels included in the analysis.

Methods for RT\_Data:

**add\_data(Data,newdata,nr)**

Adds newdata at timepoint nr in Data.

## ReadWriteData

## Classifiers

## Collectors

## Connectors

## Display

## Localizers

## Processing

## Applications