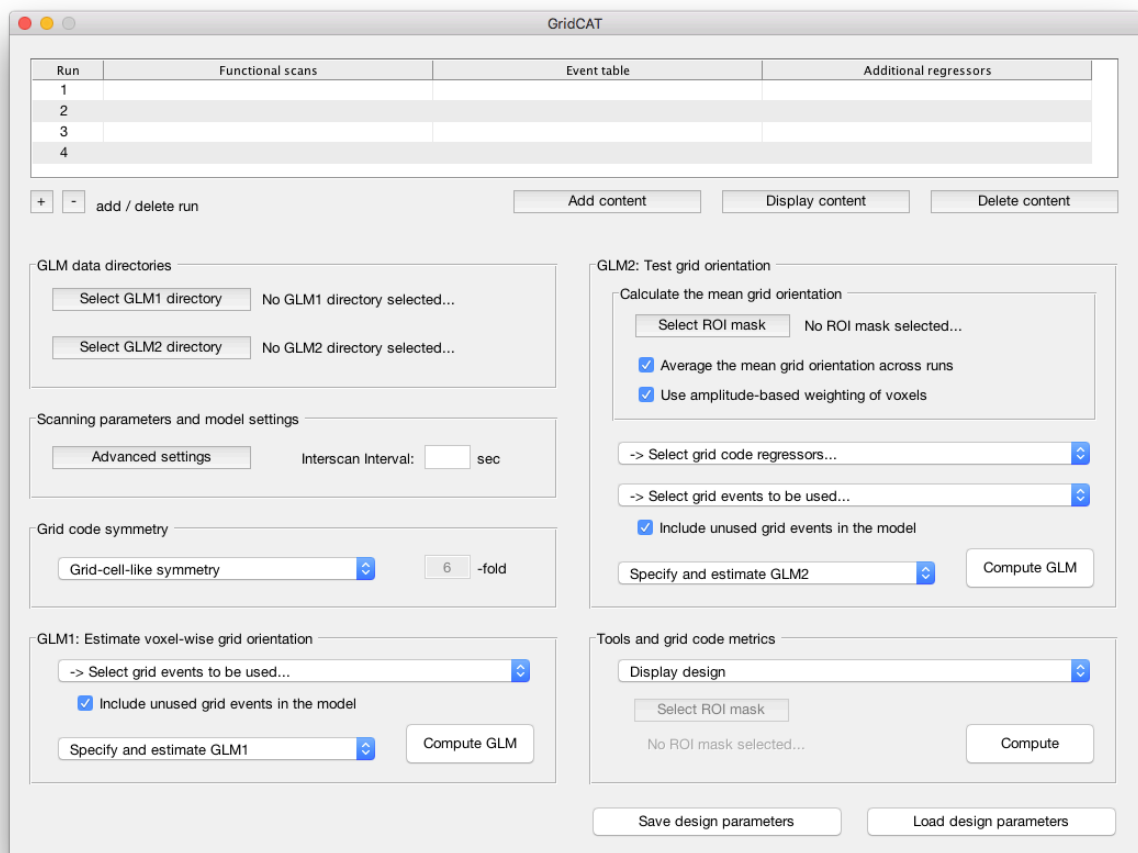


# The GridCAT Manual



We kindly ask you to cite the GridCAT in your publication, if you have used its graphical user interface or any part of the open-source code for your analysis.

## Reference:

M. Stangl\*, J. Shine\* & T. Wolbers (2017). The GridCAT: A toolbox for automated analysis of human grid cell codes in fMRI. bioRxiv. <https://doi.org/10.1101/106096> (\* equal author contributions)

## **Table of Contents**

System Requirements.....	3
Install and start the GridCAT.....	3
Analyze an example dataset.....	3
Terminology .....	4
The raw data table .....	5
Specify GLM data directories.....	6
Specify scanning parameters and model settings .....	7
Specify grid code symmetry.....	7
Specify and compute GLM1.....	7
Specify and compute GLM2.....	9
Tools and grid code metrics.....	12
Save and load design parameters .....	17
Using the example script of the GridCAT.....	17
Appendix A: The GridCAT Selection Dialog .....	18
Appendix B: The event-table.....	19
Appendix C: Advanced scanning parameters and model settings .....	21

## **System Requirements**

- The GridCAT runs on Windows, Mac and Linux Systems.
- The GridCAT requires Matlab version 2014b or above.
- To be correctly displayed, the GridCAT requires a minimum screen resolution of 1050 pixels width and 760 pixels height.
- To run the GridCAT analyses, SPM12 must be installed (i.e., added to your Matlab path). You can check this by typing “spm fmri” into the Matlab command window. If SPM does not start, please install SPM12 following the instructions in the SPM manual.

## **Install and start the GridCAT**

1. Download the GridCAT and unzip the downloaded file.
2. Move the GridCAT directory to a permanent location on your local hard drive.
3. Start Matlab and change to the GridCAT directory within Matlab.
4. Type ‘install\_GridCAT’ into the Matlab command window (this will permanently add the GridCAT to your Matlab path).
5. To start the GridCAT from now on, open Matlab and type 'gridcat' into the command window.

## **Analyze an example dataset**

In order to test the functionality of the GridCAT, you can run a complete grid code analysis and try out different analysis methods on an example dataset that we have provided.

In this manual, we highlight in blue italic text all steps that are necessary to analyze the example dataset and reproduce the results presented in the paper.

*Before you can start analyzing the example dataset, you must download it, unzip the downloaded file and save the unzipped data to your hard drive. The saving location on your hard drive will be henceforth referred to as*

*[exampleDatasetDirectory]*

## **Terminology**

- **GLM1, GLM2:**

The GridCAT analyses are carried out using a two-step approach. In the first step, grid orientations are estimated, and in the second step the estimated grid orientations are tested. We use the term 'GLM1' for the first general linear model that is carried out to estimate voxel-wise grid orientations, and the term 'GLM2' is used for the second general linear model that tests the estimated grid orientations.

- **Event:**

The term 'event' is used for anything happening during the fMRI scanning runs that you want to include in your GLM to explain changes in the blood oxygenated level dependent (BOLD) response. Every single event is specified by a name, an onset, and duration. For example, a typical event during an fMRI run could be the presentation of a fixation cross, beginning at a certain time after the start of the scanning run (onset) and staying on the screen for a specific period of time (duration).

- **Grid event:**

We use the term 'grid event' for events that have also an event-angle (i.e. directional information that is used to determine the grid orientation, quantify the magnitude of grid code response, etc.). For example, in a paradigm requiring subjects to navigate a virtual environment (e.g., like a 3D computer game) while they are in the scanner, a grid event might comprise a movement within the virtual environment that begins at a particular onset, lasts for a certain duration, and the movement is made in a specific direction (event-angle) in the virtual world.

- **Grid event type:**

During an experiment, there will usually be multiple repetitions of the same event type (e.g., multiple 'translation' events that have different onsets, durations, and event-angles). However, a researcher may also want to compare different types of grid events, for example 'active translation' and 'passive translation', denoting trials in which the participant moved through the virtual environment using a joystick versus those where they viewed a video of the movement. To investigate grid code metrics for each 'grid event type' separately, the user can specify multiple grid event types (e.g., active and passive), and label each grid event accordingly.

## The raw data table

Run	Functional scans	Event table	Additional regressors
1	(640 files) functionalScan_run1_0001.nii	eventTable_run1.txt	realignmentParameters_run1.txt
2	(640 files) functionalScan_run2_0001.nii	eventTable_run2.txt	realignmentParameters_run2.txt

add / delete run

1. Specify the number of separate fMRI scanning runs by pressing the +/- buttons in the bottom left corner of the table.

*In the example dataset, we have 2 separate runs.*

2. For each run, specify functional scans, an event-table, and (optionally) an additional regressor file, by assigning data to the table's cells (these steps are described in detail below). In order to assign data to one or multiple specific cell(s), select the cell(s) you want and then press the "Add content" button. To select data for all empty cells of the table, just click on the "Add content" button when no cell is selected.

Data selection works with the "GridCAT Selection Dialog" (this is mostly self-explanatory, but its functionality is explained in more detail in Appendix A of this manual).

If you want to double-check the data you have assigned to one or multiple cell(s), select the cell(s) you want and click on the "Display content" button. Also, you can delete cell content by selecting cells and clicking on the "Delete content" button.

- a. Functional scans are a set of 3D images for each scanning run. We recommend using files in the .nii format. Select all files of a particular run. After selection, the number of selected files and the name of the first image is shown in the table.

*For the example dataset, in order to specify functional scans for run 1, select all functional scans*

```
functionalScan_run1_0001.nii,
functionalScan_run1_0002.nii,
```

...

*from the folder*

```
[exampleDatasetDirectory]/FunctionalScans/run1/
```

*and do the same for run 2. After you have selected the files, the table should now show the number of selected functional scans and the name of the first scan. The example dataset consists of 640 files for each run.*

- b. The event-table is the textfile (.txt) that contains all the information about the individual events within an fMRI scanning run. Detailed information about the specific format of an event-table as well as additional options and examples can be found in Appendix B of this manual. Briefly, the event-table must be a textfile (\*.txt) in which each line defines one event. Each event comprises multiple columns, which can be delimited either by a semicolon, a comma, or a tabstop. For each event, column 1 defines the event-name, column 2 defines the event-onset in seconds (relative to the start of the scanning run), and column 3 defines the event duration in seconds. In addition, grid events have a 4th column in which the event-

angle is defined (0 – 359 degrees).

*To specify the event-table for run 1 or 2 of the example dataset, select the appropriate textfile*

*eventTable\_run1.txt, or  
eventTable\_run2.txt*

*from the folder*

*[exampleDatasetDirectory]/EventTables/*

- c. Optionally, you can load a file for each run containing additional regressors that you want to add to the GLM. This file can be a textfile (.txt), in which the number of lines corresponds to the number of functional scans, and each column containing a different regressor. For example, SPM automatically creates a textfile containing the realignment parameters resulting from the realignment of your functional images. These textfiles can be used directly as additional regressor files. Alternatively, you can specify a Matlab file (.mat) containing a matrix named R, which contains the corresponding values per line and column.

*For the example dataset, we provide realignment parameters that were generated automatically by SPM during data preprocessing. To specify additional regressors for run 1 or 2 of the example dataset, select the appropriate textfile*

*realignmentParameters\_run1.txt, or  
realignmentParameters\_run2.txt*

*from the folder*

*[exampleDatasetDirectory]/AdditionalRegressors/*

*They each contain 6 columns corresponding to the 6 movement dimensions (X/Y/Z direction and yaw/pitch/roll).*

## Specify GLM data directories



Two different directories for GLM1 and GLM2 need to be selected, which will store all the data produced by the GridCAT. We recommend that you create new directories for GLM1 and GLM2, which do not contain any other data.

After specifying a data directory, you can double-check its full path by clicking on the textbox next to the button.

*For the example dataset, you can create a directory*

*[exampleDatasetDirectory]/GridCAT\_data/GLM1/  
and select it as GLM1 data directory, and repeat this for GLM2.*

## Specify scanning parameters and model settings

1. Specify the Interscan-Interval (TR) in seconds, which is the time between acquiring a plane of one volume and the same plane in the next volume.  
*For the example dataset, the data was acquired with an interscan-interval of 1.5 seconds.*
2. You can also change advanced scanning parameters and model settings (microtime-resolution, microtime-onset, high-pass-filter, and the addition of time and dispersion HRF derivatives), used by SPM during the fitting of the GLM. In many cases, it won't be necessary to change the default settings. However, if you want to change these parameters, you can find more information about them in Appendix C of this manual.  
*In order to reproduce the presented results for the example dataset, do not change the advanced scanning parameters and model settings.*

## Specify grid code symmetry

For standard analyses, specify that you want to check for “Grid-cell-like symmetry” (6-fold), since grid-cell-like representations are expected to show a 6-fold symmetrical pattern. However, other symmetry values can be used to carry out control analyses in order to show that effects are strongest for the 6-fold but not other (e.g. 4,5,7,8)-fold symmetries.

*The results presented for the example dataset are generated by running separate analyses with the GridCAT checking for 4/5/6/7/8-fold symmetry, respectively.*

## Specify and compute GLM1

1. Here, you specify which events should be used to calculate GLM1. It is necessary to partition your dataset into two: one to estimate the voxel-wise grid orientation (GLM1) and a separate one to test the estimated orientations (GLM2). Thus, if you select the first half of grid events per run to estimate grid orientations in

GLM1, you would usually select the second half of grid events per run for use in GLM2. When choosing the events for GLM1, the GridCAT will automatically set the unused grid events in GLM1 as the events to be used for testing the estimated grid orientation in GLM2. However, you can change this later when you specify settings for GLM2.

*In order to reproduce the presented results for the example dataset, select to use the first half of grid events per run for GLM1.*

2. Specify whether you want to keep unused grid events in the model or not. For example, if you have selected only the first half of grid events per run for GLM1, the second half of grid events per run can either be completely excluded from GLM1, or they can be included without their corresponding event-angles (meaning that these events will not contribute to the estimation of grid orientations).

*In order to reproduce the presented results for the example dataset, select to keep unused events in the model.*

3. Specify and estimate GLM1. During GLM1 specification, the GridCAT generates and displays a design matrix and saves all model specifications in the GLM1 data directory. During the estimation process, the parameter estimates (beta images) are generated for every regressor, including the two grid event regressors ( $\beta_{\sin}$  and  $\beta_{\cos}$ ) that parametrically model grid events using  $\sin(\alpha_t \cdot 6)$  and  $\cos(\alpha_t \cdot 6)$  with respect to their grid event angle  $\alpha_t$ . Furthermore, images containing the estimated voxel-wise grid orientations are generated. You can select to carry out both specification and estimation together, or separately.
4. Press “Compute GLM” to start the calculation process for GLM1. The GridCAT starts with all the GLM1 calculations, which will take a while. Parameter estimates for each regressor are calculated and voxel-wise grid orientations are estimated. Moreover, an indicator for the strength of the grid-like activation (which might be interpreted as the “firing-amplitude”) is also calculated by  $\sqrt{\beta_{\sin}^2 + \beta_{\cos}^2}$  using the parameter estimates of the two grid event regressors ( $\beta_{\sin}$  and  $\beta_{\cos}$ ). After all the GLM1 calculations have finished, the newly generated data (e.g. beta-images for all regressors, voxel-wise grid orientation images, etc.) are stored in the GLM1 data directory.

*To estimate voxel-wise grid orientations in the example dataset, compute GLM1 (specification & estimation) now.*



## Specify and compute GLM2

1. After estimating voxel-wise grid orientations using GLM1, you may want to test the estimated orientations in GLM2. For this purpose, you have to define a region of interest (ROI) within the brain that you want to use to calculate a mean grid orientation across multiple voxels. For example, if you want to calculate the mean grid orientation within an anatomical region like the entorhinal cortex, you need to specify an ROI mask, which contains the information about where the entorhinal cortex is located in this particular subject. It is necessary, therefore, to specify a ROI mask image, which is aligned/coregistered with your functional scans, and contains binary information about your ROI (i.e., all voxels containing “1” belong to your ROI, all voxels containing “0” do not belong to your ROI).

After specifying a ROI mask image, you can double-check its full path by clicking on the textbox next to the button.

*With the example dataset, we provide ROI mask images for the entorhinal cortex (both right and left hemisphere) of our example participant. These ROI masks have been created by manually delineating the entorhinal cortex on an anatomical image using ITK-SNAP (<http://www.itksnap.org>). Select one of these ROI masks*

*`ROImask_entorhinalCortex_LH.nii`, or*

*`ROImask_entorhinalCortex_RH.nii`*

*from the directory*

*`[exampleDatasetDirectory]/ROI_masks/`*

*to reproduce the presented results.*

2. In GLM1, the voxel-wise grid orientation is estimated for each run individually. However, when testing the estimated grid orientation in GLM2, you can use either individual mean grid orientations per run, or a mean grid orientation averaged across all runs. In experiments where you would expect different grid orientations for individual runs, we recommend not averaging the grid orientation across runs. *In the example dataset, we have no reason to assume that the mean grid orientation would change between runs, so we can choose to average the mean grid orientation across runs.*
3. When calculating the mean grid orientation across all voxels within the ROI, it is possible to use the indicator of “firing-amplitude” (i.e., a measure of grid-like activation that has been calculated during GLM1 computations) per voxel. Selecting this option, the GridCAT assigns different weights to individual voxels, depending on the voxel’s “firing-amplitude”. If you do not select this option, all

voxels are weighted equally.

*In order to reproduce the presented results for the example dataset, select amplitude-based weighting of voxels.*

4. There are several ways to include regressors for grid events in GLM2.  
*In order to reproduce the presented results for the example dataset, select either a parametric modulation regressor or aligned / misaligned regressors (1 each). Including multiple aligned / misaligned regressors does not seem to be reasonable here, as the number of grid events in our example dataset is quite small for this kind of analysis. This analysis would be more appropriate with a larger number of grid events or if results are averaged across multiple participants.*
  - a. Parametric modulation regressor: Grid events can be modelled using parametric modulation that serves as a measure of alignment between the mean grid orientation and the event-angle of an individual event. One grid event regressor and the corresponding parametric modulation value (ranging from -1 to 1) will be included in the GLM2.
  - b. Aligned/misaligned regressors (1 each): Grid events can be assigned either to a regressor for “aligned” or for “misaligned” events. An event is assigned to the “aligned” regressor if its event-angle lies within +/- 15 degrees of the mean grid orientation (or a 60 degree multiple of this value), or otherwise to the “misaligned” regressor.
  - c. Aligned/misaligned regressors (multiple): Based on our model’s assumptions of grid-cell-like symmetry, we expect higher peaks in signal amplitude for an event if the offset between its event-angle and the mean grid orientation is 0 or a multiple of 60 degrees (i.e., 0/60/120/180/240/300 degrees), or lower peaks for an offset of 30 degrees plus a multiple of 60 degrees (i.e., 30/90/150/210/270/330 degrees). Analogous peaks would be expected when testing for other symmetry values (e.g. for 4-fold symmetry, higher peaks are expected with 0/90/180/270 offset, and lower peaks are expected with 45/135/225/315 degrees).  
When using this option, multiple regressors, one for each orientation for which a signal peak (high or low) is expected, will be added to the model, and each grid event will be assigned to the appropriate regressor.
5. Specify, which events should be used to calculate GLM2. Usually, you would use the other half of the data that has not been used in GLM1 to estimate voxel-wise grid orientations. For example, if you have used the first half of grid events per run to estimate grid orientations in GLM1, you would now select to use the second half of grid events per run for GLM2.  
*In order to reproduce the presented results for the example dataset, select the second half of grid events per run.*
6. Specify whether you want to keep unused grid events in the model or not.  
*In order to reproduce the presented results for the example dataset, select to keep unused events in the model.*

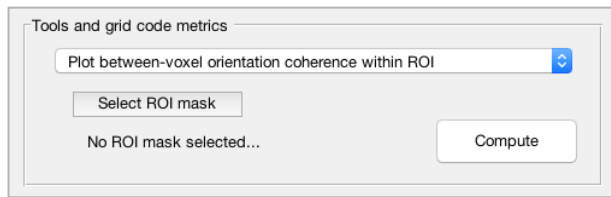
7. Specify and/or estimate GLM2 (this will generate and visualize the design matrix, save all model specifications, generate parameter estimates for each regressor, etc.).

*To test the estimated grid orientations in the example dataset, compute GLM2 (specification & estimation) now.*

The GridCAT automatically creates a default set of contrasts between regressors, which can be used to derive grid code metrics such as the magnitude of grid code response. How these contrasts are created mathematically depends on how you have selected to model grid events in GLM2 (using a parametric modulation, or one aligned/misaligned regressor each, or multiple aligned/misaligned regressors):

- a. If grid events are modelled using parametric modulation:  
t-contrasts for single runs are created by multiplying the parameter estimates for a grid event type's parametric modulation regressor by 1, whereas all other regressors' parameter estimates are multiplied by 0. Furthermore, if the same grid event type is present in multiple runs, a combined t-contrast across all of these runs is created by multiplying the grid event type's parameter estimates of each run by 1, whereas all other regressors' parameter estimates are multiplied by 0.
- b. If grid events are modelled using one regressor each for "aligned" and "misaligned" events:  
t-contrasts for single runs are created by multiplying the parameter estimates for the "aligned" grid event regressor by 1, whereas the parameter estimates for the "misaligned" grid event regressor is multiplied by -1 and all other regressors' parameter estimates are multiplied by 0. Furthermore, if the same grid event type is present in multiple runs, combined t-contrasts across all of these runs are created by multiplying the grid event type's parameter estimates for the "aligned" regressors of each run by 1, whereas the "misaligned" regressors' parameter estimates are multiplied with -1; all other regressors' parameter estimates are multiplied by 0.
- c. If grid events are modelled using multiple "aligned" and "misaligned" regressors:  
A separate t-contrast is created for each individual orientation at which a signal peak is expected. For example, based on our assumption of grid-cell-like (6-fold) symmetry, higher signal peaks are expected for 0/60/120/180/240/300 degrees and lower peaks for 30/90/150/210/270/330 degree offset between an event-angle and the mean grid orientation. Each t-contrast is created by multiplying the parameter estimates for a specific peak-orientation by 1, whereas all other regressors' parameter estimates are multiplied by 0. Again, if the same grid event type is present in multiple runs, a combined t-contrast across all of these runs is created for each peak-orientation by multiplying a peak-orientation's parameter estimates of each run by 1, whereas all other regressors' parameter estimates are multiplied by 0.

## Tools and grid code metrics



8. The GridCAT offers a set of tools and grid code metrics that generate and help visualize data and results. As these tools access data from the GLM data directories, it is necessary to have specified data directories for GLM1 and GLM2, respectively. Moreover, some tools require you select one or multiple ROI mask(s) for which output is generated or visualized. After specifying one or multiple ROI mask(s), you can double-check their full path by clicking on the textbox next to the button.
  - a. Display design
 

This tool accesses data from the GLM1 and GLM2 data directories, and allows you to review a model design by showing the design matrix generated by SPM when GLM1 or GLM2 is specified.
  - b. Plot between-voxel orientation coherence within ROI
 

This tool allows you to visualize the coherence of voxel-wise grid orientations within an ROI. It therefore uses voxel-wise grid orientation data from the GLM1 data directory. In the course of GLM1 calculations, the GridCAT has created images containing voxel-wise grid orientations for each grid event type and run, respectively, and saved them in the GLM1 data directory (named 'voxelwiseOri\_eventName\_runNr\_deg.nii'). In order to plot the coherence of voxel-wise grid orientations within an ROI for a specific grid event type and scanning run, specify your ROI mask image, and select the corresponding voxel-wise grid orientation image you are interested in. Selecting multiple ROI mask and/or multiple voxel-wise grid orientation images at the same time is possible, and will lead to separate independent plots for each image and ROI, respectively. The resulting polar histogram plots display the coherence of the grid orientation between voxels in the selected ROIs. The length of each bar indicates the number of voxels that share a similar grid orientation, and the numbers next to the rings indicate the number of voxels represented by each ring of the polar plot. Moreover, the plot displays the number of voxels within the ROI as well as the number of NaN ("not-a-number") voxels, for which a grid orientation could not be estimated. By changing the plot settings and pressing the "Redraw" button, a number of additional features can be added: The mean grid orientation across all voxels can be visualized by arrows (with and/or without using amplitude-based weighting of individual voxels). Furthermore, Rayleigh's test for non-uniformity of the voxel-wise grid orientations within the ROI can be computed, which serves as a measure of between-voxel orientation coherence. The Rayleigh test is carried out using the CircStat 2012a toolbox (<https://philippberens.wordpress.com/code/circstats/>). Voxel-wise grid orientations within the ROI can be exported, to enable the user to carry out further analyses and/or statistical tests on these data, depending on the

specific research question of interest (such as testing for systematic changes of grid orientations between conditions, or applying corrections for smoothness of the data, etc.).

*In order to plot between-voxel orientation coherence in the left and right entorhinal cortex for the example dataset, select one or both of these ROI masks*

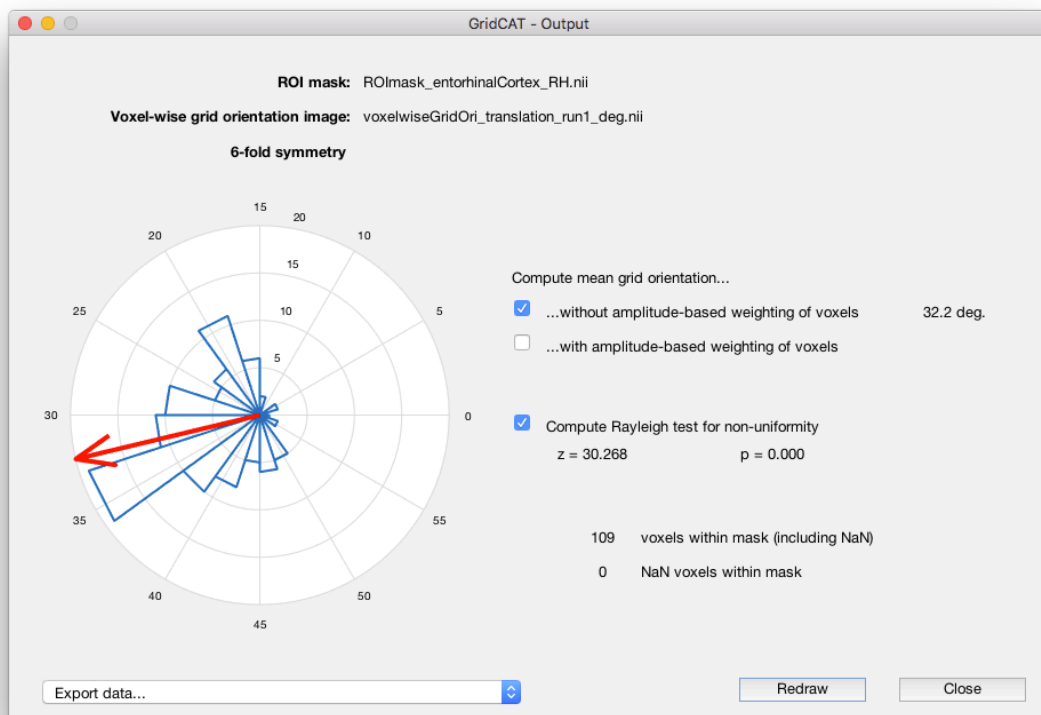
*ROImask\_entorhinalCortex\_LH.nii, and/or  
ROImask\_entorhinalCortex\_RH.nii*

*from the directory*

*[exampleDatasetDirectory]/ROI\_masks/*

*and select one or multiple of the images containing voxel-wise grid orientations in the GLM1 data directory, e.g.:*

*voxelwiseGridOri\_translation\_run1\_deg.nii*



c. Plot within-voxel orientation coherence within ROI

This tool allows you to visualize the voxel-wise coherence of grid orientations between two different conditions within an ROI. For example, different conditions could be two different runs, or two different grid event types. This tool accesses voxel-wise grid orientation data from the GLM1 data directory. Specify your ROI masks, and select two images containing voxel-wise grid orientations that you want to compare (i.e., select one or multiple images for different grid event types and runs, respectively). The tool will visualize and calculate the proportion of stable voxels between the two images within all voxels of the ROI. The two black rings in each plot represent the two different conditions, and each voxel's grid orientation is indicated with a circular marker; a line connects the orientations of each voxel. Whether a voxel's orientation is deemed 'stable' depends on whether its estimated orientation differs by more than a given threshold, which is initially set to  $\pm 15$  degrees for 6-fold symmetry analysis (analogous for other symmetry values). This threshold,

however, as well as the visual appearance of the plot can be changed manually by adapting the settings in the plot and pressing the “Redraw” button.

Independent calculations are performed for each ROI mask image that you have selected.

*In order to plot within-voxel orientation coherence in the left and/or right entorhinal cortex between run 1 and run 2 for the example participant, select one or both of these ROI masks*

*ROImask\_entorhinalCortex\_LH.nii, and/or*

*ROImask\_entorhinalCortex\_RH.nii*

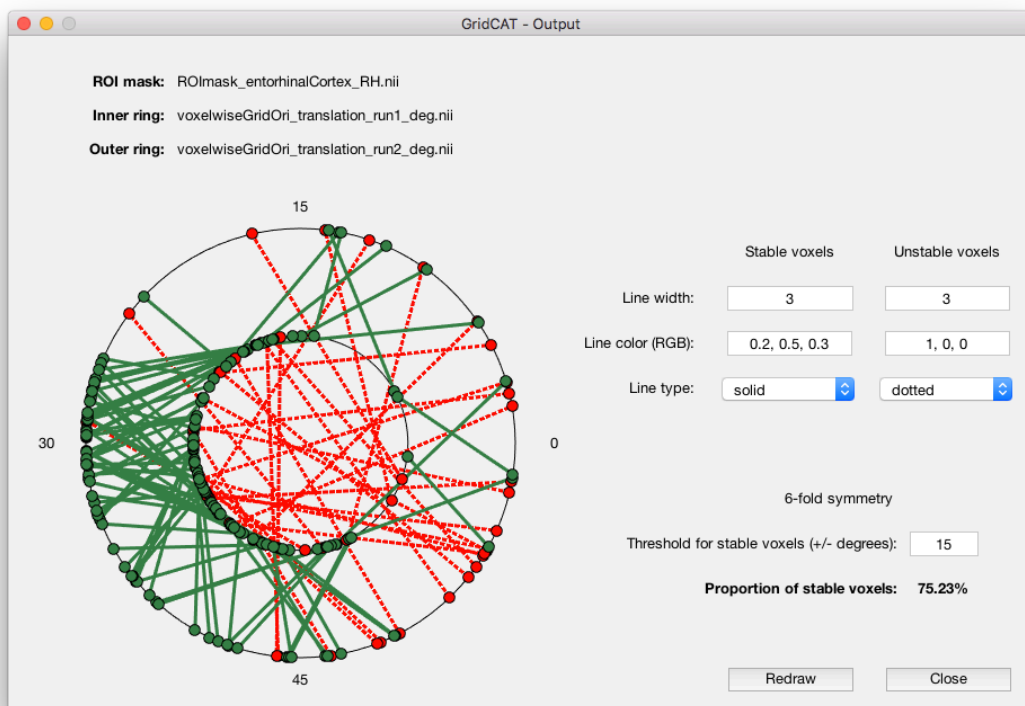
*from the directory*

*[exampleDatasetDirectory]/ROI\_masks/*

*and select the following images containing voxel-wise grid orientations in the GLM1 data directory:*

*voxelwiseGridOri\_translation\_run1\_deg.nii, and*

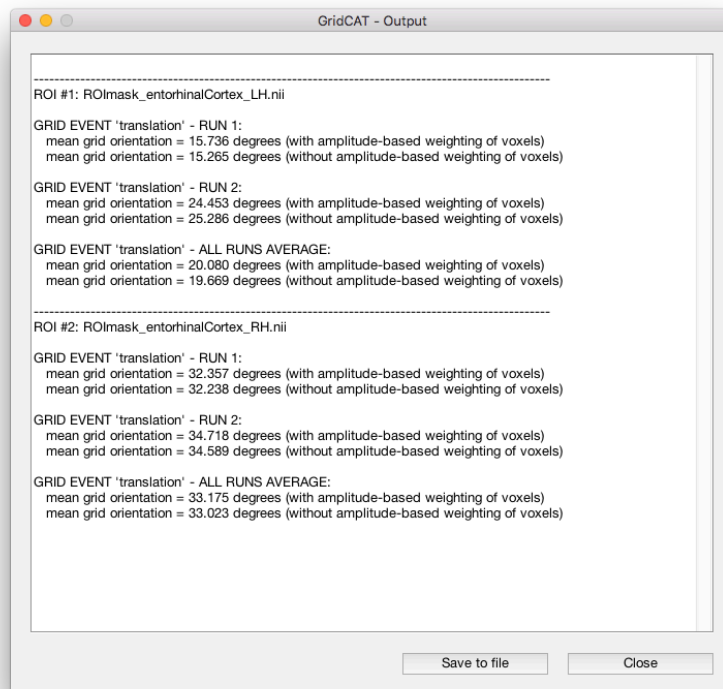
*voxelwiseGridOri\_translation\_run2\_deg.nii*



d. Output mean grid orientation within ROI

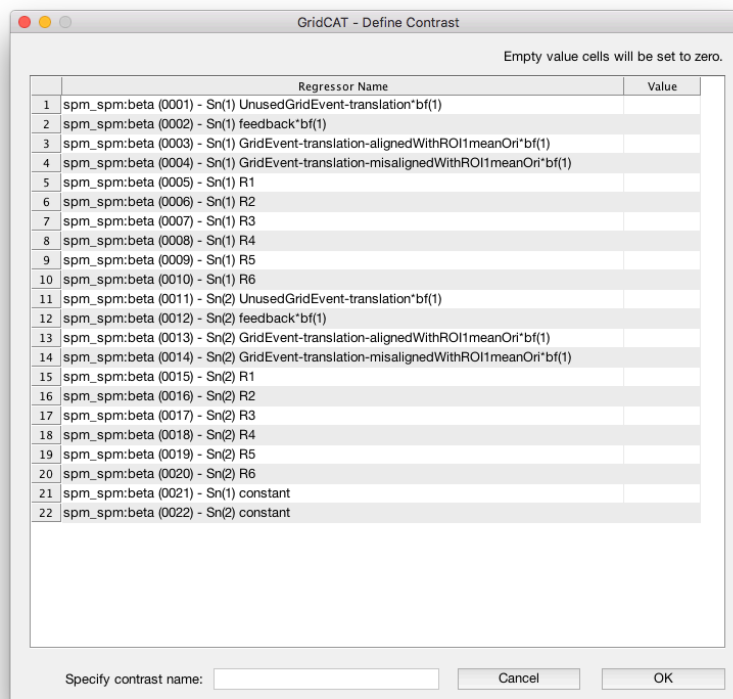
Using this tool, you can output the mean grid orientation within an ROI. It therefore accesses voxel-wise grid orientation data from the specified GLM1 data directory and calculates the mean grid orientation across all voxels within the specified ROI mask(s). As the mean grid orientation can be calculated either with or without using different weightings for individual voxels (i.e., ‘amplitude-based’ weighting), this tool outputs the mean grid orientation for both options. Independent calculations are performed for each ROI mask image that you have selected.



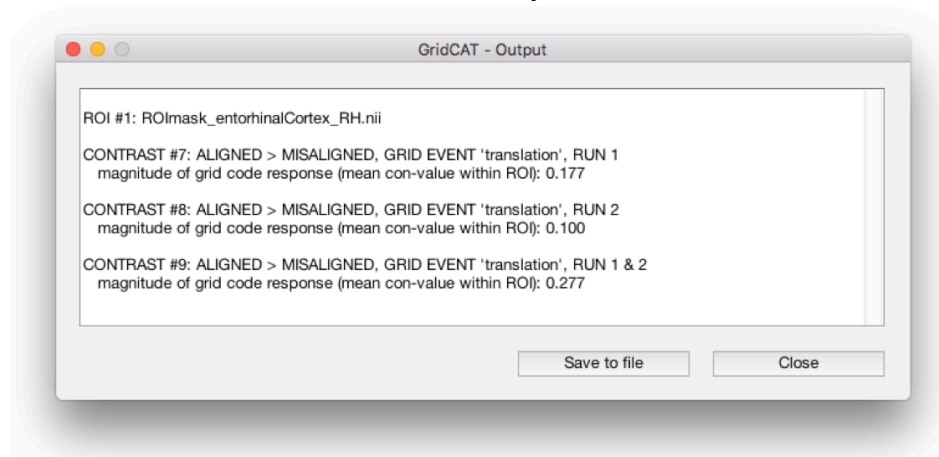


e. Create custom contrasts between GLM2 regressors

If you want to generate results for contrasts that are not specified in the default set of contrasts created by the GridCAT, you can specify additional t-contrasts with this tool by combining individual regressors. This tool accesses data from the GLM2 data directory. In order to create a new t-contrast, just assign a value to any GLM2 regressor you want (which will be used as the multiplication factor for this regressor's parameter estimates), specify a contrast name and press the "OK" button. Parameter estimates of regressors, for which you do not specify a multiplication value, will be multiplied with 0.



- f. Output magnitude of grid code response within ROI
- This tool allows you to calculate the magnitude of grid code response in one or multiple ROIs that you have specified by pressing the “Select ROI mask” button. The magnitude of the grid code response can be shown for contrasts that have been specified automatically by the GridCAT (the default set of contrasts) or for additional contrasts between GLM2 regressors that you have created. Here, you need to select the contrasts you want to interrogate and the GridCAT will display the average magnitude of grid code response within each ROI separately for each selected contrast. To generate this grid code metric, this tool accesses data from the GLM2 data directory.



- g. Export all grid code metrics
- This tool allows you to export a comprehensive list of all grid code metrics that can be calculated based on a given dataset, for example to allow statistical analysis across participants. Therefore, it accesses the specified GLM1 and GLM2 data directories and also requires you select specific ROI masks in which the grid code metrics are calculated.

*In order to export all grid code metrics for the example dataset, select both ROI masks*

*ROI mask\_entorhinalCortex\_LH.nii, and  
ROI mask\_entorhinalCortex\_RH.nii*

*from the directory*

*[exampleDatasetDirectory]/ROI\_masks/*

*and specify a location on your local hard drive where all the output will be saved.*

All output is saved to a textfile, which is formatted in delimited text using a semicolon delimiter. Consequently, the output data can be used with a program of your choice, for example in order to carry out further analysis and statistical tests on the calculated grid metrics, depending on your specific research question. For that purpose, import the textfile into a program that is able to convert (semicolon-)delimited text to separate cells of data, such as Microsoft Excel, SPSS, or others.

The output file includes the following grid code metrics for all specified ROI masks and separately for every grid event type:

- **Magnitude of grid code response:** This grid code metric is calculated for every contrast specified in the GLM2 data directory (i.e., the default set of contrasts as well as those additionally created by the user).



- Between-voxel orientation coherence within ROI: This grid code metric is calculated for each run individually as well as for the average across all runs. Rayleigh z- and p-values are reported.
- Within-voxel orientation coherence within ROI: This grid code metric is calculated and the proportion of stable voxels is reported between every possible pair of voxel-wise grid orientation images (i.e. all pairs of runs and grid event types, respectively).
- Mean grid orientation within ROI: This grid code metric is calculated for all specified ROI images, for all runs and grid event types separately. The resulting mean grid orientation is reported for both calculation methods (i.e., with and without amplitude-based weighting of voxels).

## **Save and load design parameters**



The GridCAT enables you to save all design parameters (i.e., all changes and settings you have made using the GridCAT's graphical user interface), in order to load them again later. All information is stored in a Matlab-file (\*.mat) on your local hard drive. Simply press the "Save design parameters" button and select a location to store the file.

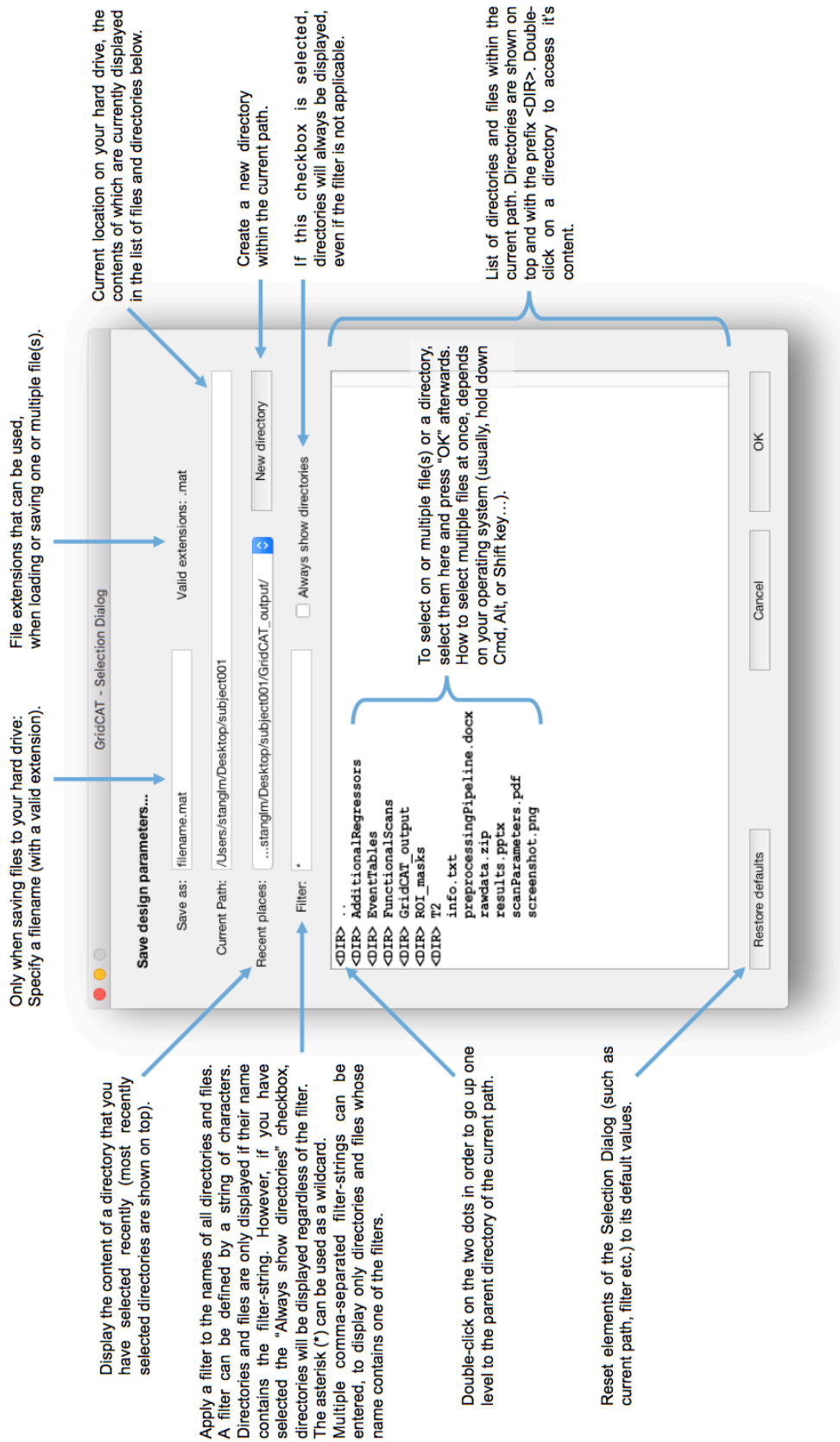
Furthermore, whenever you compute a GLM1 or GLM2, the design parameters are automatically stored in the GLM1/GLM2 data directory (named 'GridCAT\_GLM1.mat', with the same naming convention for GLM2). This option allows you to restore and inspect the settings you have used when computing a GLM.

In order to restore design parameters, press the "Load design parameters" button and select the mat-File containing the relevant design parameters.

## **Using the example script of the GridCAT**

For those who prefer to use and modify the open-source Matlab code of the GridCAT, rather than the GUI, we provide an example script, which includes all steps of a common grid code analysis pipeline. Detailed comments in the script will guide the user through necessary settings and functions in the code. The example script can be found in: `DEMO_GridCAT_Script.m`

## Appendix A: The GridCAT Selection Dialog



## Appendix B: The event-table

An event-table is a textfile (\*.txt) for each fMRI scanning run, which contains all information about single events during this run. Each line in the event-table defines a single event using the following format:

name	onset	duration
------	-------	----------

In addition to the three columns per event as shown above, a grid event also has a 4th column defining its event-angle:

name	onset	duration	angle
------	-------	----------	-------

The following example shows an event-table from a navigation paradigm, in which the subject navigated a virtual environment (like in a 3D computer game). The experimental paradigm allowed the user to perform translational and rotational movements within the environment, as well as to stand still on the spot. In this event-table, translational movements are used as grid events.

stand	0	3.5	
translate	3.5	4	225
rotate	7.5	2.25	
translate	9.75	3.2	192.3
rotate	12.95	2	
translate	14.95	7.05	4.5
stand	22	3.73	
.	.	.	.
.	.	.	.
.	.	.	.

In GLM1 and GLM2, the GridCAT puts all events that are within an event-table and have the same name into one regressor. This is true for normal events as well as for grid events. Consequently, if you want to distinguish between different grid event types within a run, simply use different names for these events. The following example shows an event-table similar to the one above, but now the paradigm includes both active translations ('activeTransl') and passive translations ('passiveTransl'), which should be used as two different types of grid event (i.e. entered as different regressors) in GLM1 and GLM2.

stand	0	2.2	
rotate	2.2	3	
activeTransl	5.2	4.5	359
stand	9.7	3	
activeTransl	12.7	1.3	359
rotate	14	2.5	
passiveTransl	16.5	5	7.5
rotate	21.5	4.5	
passiveTransl	26	2.3	24
activeTransl	28.3	10.2	24
.	.	.	.
.	.	.	.
.	.	.	.

The GridCAT offers several different ways to partition your grid events for GLM1 and GLM2. However, you may wish to specify yourself which events to include in GLM1 and/or GLM2. To do this, you can add two more columns to an event-table that specify whether a grid event is used in GLM1 (column 5) and/or GLM2 (column 6). If column 5 is set to '1' then this event will be used as a grid event in GLM1. Set column 5 to '0' if you do not want to use this event as a grid event in GLM1; the same method is used for GLM2 in column 6.

The following example shows the content of an event-table in which there are only 'translation' and 'rotation' events. Translations are used as grid events (i.e., they also have an event-angle) and the event-table also specifies which grid events to use in GLM1 and GLM2.

stand	0	2			
translation	4.3	5.3	22	1	0
stand	9.9	3			
translation	16	2.1	173	1	0
translation	22.4	5	249.4	0	1
translation	28	2	156	0	1
stand	33.91	2.2			
translation	38	9	97	1	0
translation	52.3	0.7	359	1	0
stand	55.2	3.7			
translation	59.1	8	16.34	0	0
stand	72	2			
translation	74.2	1.3	1.7	0	0
translation	75.5	5	1.7	1	1
translation	86	3.2	39	1	1
stand	91.5	8			
translation	107.3	0.72	100	0	1
translation	109	4.6	66.74	1	0
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.

The GridCAT accepts tabstopps, semicolons, and commas as delimiters between columns. Consequently, the three following examples of event-tables all have an appropriate format and an identical content:

stand

0

3.5

translate

3.5

4

225

rotate

7.5

2.2

translate

9.7

3.2

192

rotate

12.9

2

translate

14.9

7.1

4

stand

22

3

(tabstopp-delimited)

=

stand;0;3.5

translate;3.5;4;225

rotate;7.5;2.2

translate;9.7;3.2;192

rotate;12.9;2

translate;14.9;7.1;4

stand;22;3

(semicolon-delimited)

=

stand,0,3.5

translate,3.5,4,225

rotate,7.5,2.2

translate,9.7,3.2,192

rotate,12.9,2

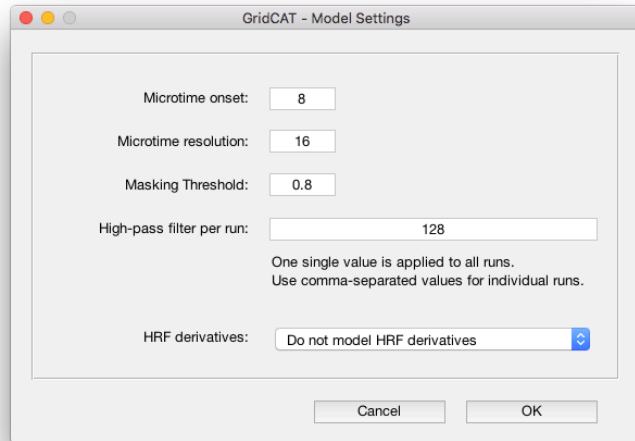
translate,14.9,7.1,4

stand,22,3

(comma-delimited)

## Appendix C: Advanced scanning parameters and model settings

The following information provides only an overview of the advanced model settings that are used by SPM12 and can be adapted in the GridCAT. For more detailed information about these settings, please refer to the SPM manual.



1. Microtime onset is the reference time-bin at which the regressors are resampled to coincide with data acquisition. If you have performed slice-timing correction, you must change this parameter to match the reference slice specified there. Otherwise, you might still want to change this if you have non-interleaved acquisition and you wish to sample the regressors so that they are appropriate for a slice in a particular part of the brain.
2. Microtime resolution is the number of time-bins per scan used when building regressors. If you have performed slice-timing correction, change this parameter to match the number of slices specified there; otherwise, you would typically not need to change this.
3. Masking threshold: By default, SPM uses a proportional threshold to identify these voxels that seem to carry a decent signal in them. If a voxel fails to exceed this threshold (e.g., voxels outside the brain), it will not contribute to the output of a GLM. For example, a masking threshold of 0.8 will only estimate the GLM in voxels whose mean value is at least 80% of the global signal.
4. High-pass filter: Slow signal drifts within a period longer than the value specified here will be removed. Either input one single value, which will be applied for all runs, or input one value per run, separated by commas.
5. HRF derivatives: The canonical HRF combined with time and dispersion derivatives comprise an 'informed' basis set, as the shape of the canonical response conforms to the hemodynamic response that is commonly observed. The incorporation of the derivate terms allow for variations in subject-to-subject and voxel-to-voxel responses. The time derivative allows the peak response to vary by plus or minus a second and the dispersion derivative allows the width of the response to vary.