Stimulation

ContStim – CONTinousSTUMulation

ContStim is used to generate visual stimuli (e.g., drifting bars, drifting/rotating square grating) on a monitor screen, to generate auditory stimuli via a sound card or to produce TTL sequences to control other auxiliary devices. Also, it generates synchronization signals, which are transmitted via DIO or UDP channels to the acquisition computer and recorded by **ContImage**.

Figure 1. Components of ContStim.

Components of ContStim

1   Menu bar. Function: selection of modalities, and stimuli.
2   Main tool bar. Function: starting/stopping stimulation and re-initialization.
3   Log window. Function: error (red) and warning (blue) logging. Other types of logged info are colored black for normal events, green for debugging messages. The latter ones can be ignored.

Selection of the necessary modality and stimulus is made trough the Menu. Modifications of the particular stimulus parameters are made by editing the initialization file. The initialization file can be placed anywhere in the system, its full path should be referenced in the master initialization file: `C:\WINNT\ContStimMaster.ini` or `C:\Windows\ContStimMaster.ini`. The simplest version of `ContStimMaster.ini` has a reference to the actual initialization file only:

```
[contstim]
IniFile=C:\OpticalImaging\contstim.ini
```

The `ContStimMaster.ini` may contain default initialization values of stimuli parameters as well. These values are used if the reference "`IniFile`" file is not specified or cannot be found. It is recommended to have a few initialization file prepared for different modalities, e.g. "`ContStimVisual.ini`", "`ContStimAuditory.ini`", and so on, and/or different stimuli and then modify the link "`IniFile`" in the master file `C:\WINNT\ContStimMaster.ini` only.

The initialization file may contain information about many kinds of stimuli. The particular stimulus to be generated is selected through the Menu (Figure 2). Some parameters may be repeated a few times in the initialization sections of different stimuli. The user should comment out the values that do not belong to the desired stimulus. It is safer to have individual initialization files prepared for each stimulus and to change the link in the master initialization file only.
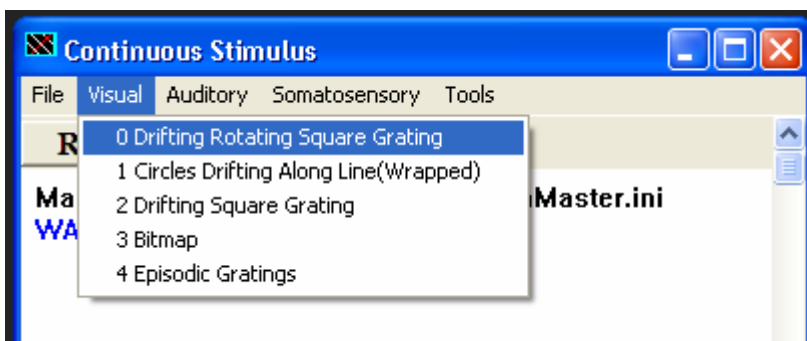

Figure 2. Selection of the visual stimulus.

Pressing Main tool bar "Run" button or the "Space" key starts stimulus generation. Keyboard keystrokes control **ContStim** while it is generating visual stimuli (the program window is not seen on the screen). Pressing "Escape" or "a" (Abort) stops stimulation and pressing "p" (Pause) pauses stimulation. Pressing "r" (Run) continues stimulation.

Visual Stimuli

Useful visual stimuli provided with **ContStim** are:
1 Drifting rotating square grating (0).
2 Circles drifting along a line (1).
3 Drifting square grating (2).
4 Bitmap (3).
5 Episodic gratings (4).

Description of the initialization file: contstim.ini

contstim.ini has a structure of the Windows initialization files (.ini files). Section names are surrounded with square brackets, e.g. `[contstim]`. Other names are variables and values they are set to, e.g. `monitorwidthcm="40.0"`. **contstim.ini** has 3 sections: `[Sync]` for sunchronization options, `[contstim]` for visual stimuli, `[audio]` for auditory stimuli, and `[TTL]` for other types of stimulation, e.g. control of shutters. The variable names are not cASe-Sensitive. Any special character (e.g. ";") inserted before the variable comments out the whole line.

Stimulus independent parameters

Synchronization section

The synchronization section `[Synch]` defines what channels are to be used for synchronization. There are two types of synchronization channels: DIO (direct wire connect between computers) and UDP (connection via Ethernet IP/TCP). The following is a typical `[Sync]` section

```
[Sync]

SyncDIOYes="0"                  // Do not use DIO channel
SyncUDPYes="1"                  // Use UDP channel

SyncUDPHost="grabber"          // The host to receive UDP synch
SyncUDPPort="8936"             // The port on the host to receive the synch
```

Note that the acquisition computer "listens" to preset port "8936"; this port value should not be modified. The `[sync]` section is rarely edited.

Visual stimuli section

The visual section [`contstim`] starts from description of the monitor parameters and settings. These are stimulus independent parameters.

```
[contstim]
MonitorResolutionWidth="800"
MonitorResolutionHeight="600"
refreshratehz="120.0"
monitorwidthcm="40.0"
monitorheightcm="30.0"
screendistancecm="40.0"
```

`MonitorResolutionWidth` and `MonitorResolutionHeight` – resolution of the monitor XVGA mode in pixels. Standard values are 640x480, 800x600, 1024x768, 1280x1024, and higher. It is not recommended to choose high-resolution modes since it increases the CPU and graphic processor load and as a result some frames may be skipped leading to a jerky motion of the stimuli. A reasonable choice is 800x600.

`RefreshRateHz` – refresh rate of the current mode measured in Hertz. **ContStim** does not set the refresh rate. The program uses this value to set proper timing for frame display. The user through the ControlPanel should set the actual refresh rate. Standard values depend on the monitor make and model. They usually vary between 60 and 120Hz or even higher. It is not recommended to use very low refresh rates such as 60Hz. It will lead to flickering. It has been reported that some neurons in the cat visual system can follow this flickering. The values from 80Hz and up are reasonable. Using 120Hz insures stability of the image on the screen and is recommended. Combination of high-resolution with high refresh rate can be very CPU time consuming. If motion of the stimuli is not uniform or jerky either resolution or refresh rate or both should be lowered.

`MonitorWidthCm` and `MonitorHeightCm` – physical dimensions of the monitor screen in centimeters. They should be measured with a ruler for a monitor in use, written to the initialization file and never modified. A typical 22`` monitor has dimensions 40x30cm. Nonetheless the actual dimensions of the visible part should always be measured manually.

`ScreenDistanceCm` – physical distance from the screen to the subject (subject's nose) in centimeters. This parameter is usually set to 40-50cm for ferrets and cats, and to 25cm for mice.

Next group of parameters describes the size and position of the bounding box. Stimuli consisting of bars will be displayed inside this rectangular box only.

```
BoundBoxCenterXDeg="10"
BoundBoxCenterYDeg="10"
BoundBoxCenterWidthDeg="20.0"
BoundBoxCenterHeightDeg="20.0"
```

`BoundBoxCenterXDeg` and `BoundBoxCenterYDeg` – specify position of the center of the bounding box on the screen measured in degrees. The reference point is located in the center of the screen. The example above puts the center of the box 10 degrees to the right and 10 degrees up from the center of the screen.
`BoundBoxCenterWidthDeg` and `BoundBoxCenterHeightDeg` – specify the size of the box in degrees. The example above makes a box of 20x20 degrees. The stimulation will be limited to right-top quadrant of the screen. Setting value of either of these variables to –1 will unbind stimulation in the corresponding dimension. If both are set to –1 the whole screen is used.

The final stimulus independent group of parameters defines foreground and background measured in percent.

```
contrastpct="50"
backgroundpct="50"
```

The formulae used to calculate actual foreground and background gray-scale color are

$$Foreground=(backgroundpct+contrastpct/2)*2.55$$
$$Background=(backgroundpct-contrastpct/2)*2.55$$

The example above sets the foreground to 191 and the background to 64, where 255 corresponds to lightest screen and 0 to darkest. The positive values of give light stimulus elements on a dark screen (ON-OFF type). Use of negative `contrastpct` is permitted; it gives bright background and dark foreground (OFF-ON type). Setting `contrastpct` to 100 and `backgroundpct` to 50 gives highest contrast.

Stimulus dependent parameters

The first two variables of every stimulus define number of cycles this stimulus will be displayed for:

```
iniangle="-1.0"
finangle="11.0"
```

Originally the program was written to generate rotating square gratings only, that is why the variables are called `iniangle` (INItial ANGLE) and `finangle` (FINal ANGLE). The number of cycles is the difference between `finangle` and `iniangle`. The number of cycles to display will be set to 12 by the example above. The display time depends on the number of cycles and on the length of one cycle, which is by another (stimulus dependent) variable.

Parameters for drifting-rotating square grating

This stimulus is used to evaluate orientation and direction selectivity. The stimulus is the full field square grating with two degrees of freedom: (1) slow rotation of the grating and (2) fast drift of the grating. The following parameters describe the rotation:

```
anglecycle_min="1.0"
clockwise="-1"
```

AngleCycle_Min – specifies rotation speed in cycles per minute. The example above sets it to 1 cycle per minute. This variable should take only positive values. Direction of rotation is controlled by another variable `clockwise`. Setting `clockwise` to 1 will rotate the grating clockwise and to –1 will rotate it counterclockwise. The example above will generate a grating rotating counterclockwise.

The pattern and the fast drift are described by the following three parameters:

```
tfcycle_sec="2.0"
sfcycle_deg="0.2"
striperatio="0.5"
```

`tfcycle_sec` – temporal frequency measured in cycles per second. Typically used value for cats and ferrets is 2.0.

`sfcycle_deg` – spatial frequency measured in cycles per second. Typical value for cats and ferrets is 0.2.

`striperatio` – this parameter specifies the duty cycles. It is a ratio of the width of the bright bar to the whole spatial cycle. The examples sets `striperatio` to 0.5, which means that duty cycle is 50% or that the bright and dark bars have equal width. This is the value used in cat experiments.

Parameters for drifting square grating

This stimulus is used to evaluate topography of the visual cortex (retinotopy). The stimulus is a simplified version of the previous stimulus (drifting-rotating grating); it does not have the rotation parameters. Parameters specific to this stimulus are

```
stripesizedeg="2.0"
direction_deg="90.0"
```

`StripeSizeDeg` – carries function similar to `striperatio`. It defines width of the bars in degrees. Both can be used interchangeably. `StripeSizeDeg` is more convenient since it explicitly defines the width.

`Direction_Deg` – specifies direction of the drift in degrees. 0 – vertical bars moving to the right, 90 – horizontal bars moving upward, 180 – vertical bars moving to the left, and 270 – horizontal bars moving downward. All other values (oblique) are acceptable as well.

Typical values for temporal and spatial frequencies are

```
tfcycle_sec="0.125"
sfcycle_sec="0.01"
```

These values are used for mouse imaging.

Parameters for other stimuli

Parameters for "Circles drifting along a line" are similar to those for "Drifting square grating".

Stimulus "Bitmap" is used for presentation of an arbitrary sequence of images prepared by the user.

Stimulus "Episodic gratings" is used to generate drifting square gratings of various orientations in the manner similar to the standard

Appendix A

Example of complete **contstim.ini** file. Note that all entries for the topography stimulus are preceded with a semi-colon, which turns them into a comment. This way the user can keep all stimuli in one file and uncomment the entries for the required stimulus only.

```
[contstim]
MonitorResolutionWidth="800"
MonitorResolutionHeight="600"
refreshratehz="120.0"
monitorwidthcm="40.0"
monitorheightcm="30.0"
screendistancecm="40.0"

BoundBoxCenterXDeg="10"
BoundBoxCenterYDeg="10"
BoundBoxCenterWidthDeg="-1.0"
BoundBoxCenterHeightDeg="-1.0"

contrastpct="50"
backgroundpct="50"

;;Orientation stimulus
iniangle="-1.0"
finangle="11.0"
anglecycle_min="1.0"
clockwise="-1"
tfcycle_sec="2.0"
sfcycle_deg="0.2"
striperatio="0.5"

;;Topography stimulus
;iniangle="0.0"
;finangle="90.0"
;tfcycle_sec="0.125"
;sfcycle_sec="0.01"
;stripesizedeg="2.0"
;direction_deg="90.0"
```