TemporalBarcode Stimulus Specification: Pilot Experiment

Last updated 5/1/23 PR

The Pilot Script is meant to run approximately 2 hours and test a number of stimulus conditions in order to determine which conditions are most effective at driving responses across visual areas in mouse.

Stimulus Categories

1. Full Field Flicker: a spatially uniform field which updates its luminance every video frame, according to a sequence that is passed in as a vector of floating-point values ranging from -1 (black) to +1 (white). *Implementation notes: could be implemented as a sinusoidal grading with spatial frequency of 0?*

2. Standing Grating Flicker: a standing sinusoidal grating of a specified spatial frequency, phase, and orientation, which updates its contrast every video frame, according to a sequence that is passed in as a vector of floating-point values ranging from -1 (100% contrast) to +1 (100% contrast phase reversed, such that stripes that are black at -1 become white at +1).

3. Randomly Drifting Grating: a standing square-wave grating (black and white stripes with sharp edges) with a specified spatial frequency and orientation, which starts with a phase of 0 and then drifts in the direction orthogonal to the stripes updating the drift rate (edge speed) every video frame according to a sequence that is passed in as a vector of floating-point values ranging from -1 to 1, where for vertical stripes -1 is the maximum speed in leftward direction and +1 is maximum speed in rightward direction. The maximum speed is a scalar parameter. *Note: I am the least clear on exactly what this will look like, so I may need to tweak it once I see it. See end of document for more info (PR).*

Implementation notes:

- *A square grating could be implemented as a sinusoidal grading with contrast >>>1*
- *One way to specify the scalar for the maximum drift rate is the degrees of a cycle the phase should shift in one video frame when the input value is +1 (e.g. a*t 30fps a drift rate of 12 degrees/frame corresponds to *1Hz temporal frequency if the input vector is all +1's).*
- *We have implemented to accept any value from -1 to 1 for generality, we currently expect to use values of -1 and +1 exclusively.*

The Temporal Sequences

The Pilot Script uses four temporal sequences for stimuli:
Unique1= binary white noise 3600 values (120s at 30fps)
Repeat1= binary white noise 240 values (8s at 30fps)
Unique2= exponentially distributed white noise 3600 values (120s at 30fps)
Repeat2 = exponentially distributed white noise 240 values (8s at 30fps)
*The second pair are only for the full field flicker stimulus.

Structure of Session

The pilot will test one stimulus category at a time as follows. For 120min session N=32.

- Full Field Flicker: Unique1(x1) Repeat1(xN) Unique1(x1) = 8.3 minutes
- Full Field Flicker: Unique2(x1) Repeat2(xN) Unique2(x1) = 8.3 minutes
- Standing Grating Flicker: loop over 3 spatial frequencies, 2 orientations, and 2 phases; and for each combination play Repeat1(xN). Total 3*2*2*N*8sec = 51.2 minutes
- Randomly Drifting Grating: loop over 3 spatial frequencies, 2 orientations, and 2 drift speeds; and for each combination play Repeat1(xN). Total 3*2*2*N*8sec = 51.2 minutes

GRAND TOTAL 119 minutes

Testing Modes and debugging tools

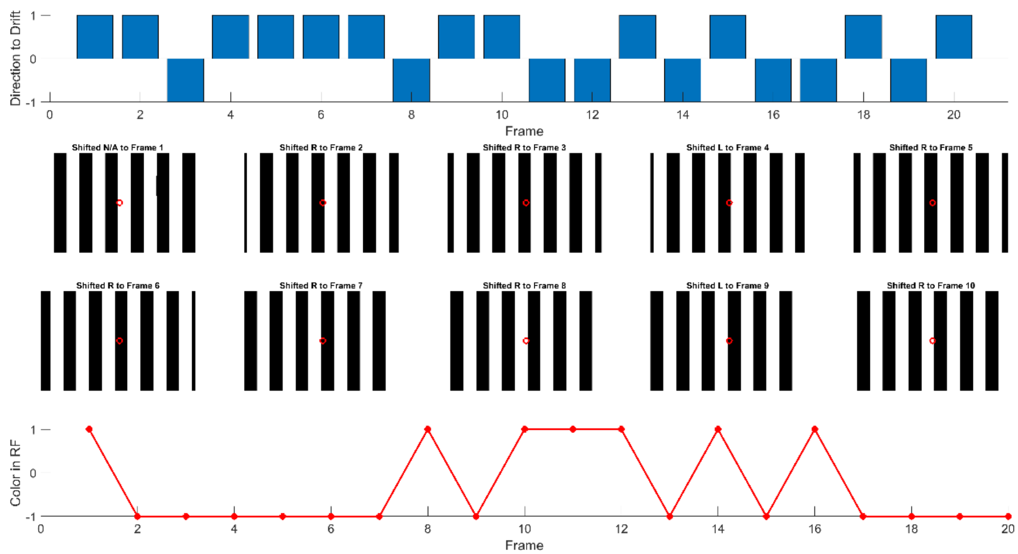
1. The testmode flag runs the fastest possible test sweeping all conditions by setting N to 1. The FFF reduce to 4.133min each and the grating conditions reduce to 1.6min each for a total of 11.5 minutes.
2. The SessionDuration variable is meant to compute the N necessary to achieve approximately the requested duration, for any longer duration. The calculations used are specific to the above session structure however. *No guarantees that it will deal properly with changing the number of parameter values in the grating conditions!*
3. The TimeDilation variable is meant to slow the stimuli down so it is easily to visually inspect that they are doing what we intended. This simply stretches the input stimulus vectors in time by duplicating each frame value the specified number of times. Only positive integer values are supported.
4. The FPS variable specifies the nominal framerate. This is only needed for testing purposes because we developed on machines with much higher framerates. The script has been designed assuming the rig is 30FPS and <u>this should simply be set to 30FPS</u>. Do not set to a more precisely measured exact frame rate; we are specifying the stimuli per frame, not per time interval.

How the production script will differ from this script

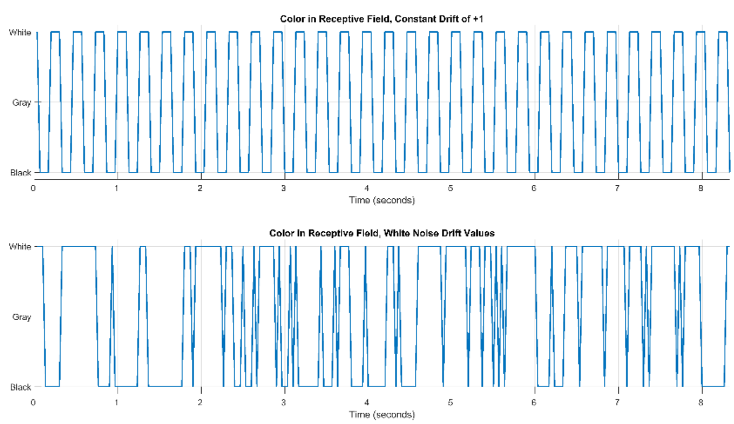
The production script will use only one or two of the three stimulus categories, depending on which one(s) work best in terms of exciting mouse visual neurons. If we use Full Field Flicker, the N for repeats will be much larger than this, but we will use only one of the two temporal stimulus pairs. If we use a grating stimulus, the N for repeats will be on the same order as the pilot script, but we will loop over more distinct values of the parameters. <u>We intend to keep the SessionDuration variable in the production script</u> so you can run short sessions for acclimation purposes. Other test modes and debugging tools may not be included, unless you want them.

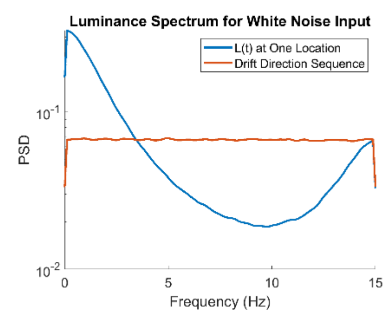
What the drifting grating is intended to look like

This illustration below used a drift constant of 45, meaning that a value of +1 corresponds to drifting 45° to the right (1/8 of grating cycle) in one frame (1/30 sec), and -1 corresponds to drifting 45° to the left. At top I show a short sequence of binary white noise (representing the first 20 values of a stimulus file). In the middle I show the visual stimulus, with a reference location on the screen circled in red (receptive field). The stripes shift left or right each frame according to the sequence at top, causing the reference location's luminance to switch from black to white or back when it crosses over an edge (bottom, red).



If the stimulus sequence was simply +1 every frame for a longer time, the color in the receptive field over time would look like the top graph below. If the stimulus sequence was binary white noise for a longer time, the color in the receptive field over time would look like the bottom graph below.





Note that when the input sequence is white noise, as in the bottom trace, the luminance as a function of time has a strange spectrum due to the interaction of the spatial frequency, drift rate constant, and sharp edges.