

Instructions to Run the RSVP pRF Program

2/28/22

Jeff Kravitz

Curtis Lab

New York University

1. Set monitor to 60 fps & Set primary screen

- This program is designed to run at 60 fps; be sure to change your display settings to ensure that the programs functions properly.
- On Mac, you should set the primary monitor to the monitor that the subject will see. To do this, open System Preferences > Displays > Arrangement, and then drag the white bar at the top of the blue display to the correct display. When you do this, you will not need to use the 'Use Second Monitor' option in the program GUI. Make sure to drag your terminal & relevant windows over to your second window before you run the program.

2. Check the parameters

- Before running the program, check the rsvp_params.txt file to make sure that the parameters meet your needs.

3. Activate Python environment

- Before running the experiment, you must first activate the Python environment so that the program will have access to the correct version of Python and its required packages.
- To activate the environment, open your terminal (on Mac) or your Anaconda Prompt (on Windows) and enter:
conda activate clayspace_psychopy_env

4. Launch the program

- Navigate to the RSVP_pRF/ directory
- To launch the program, enter the following command in your terminal or Anaconda Prompt:
python rsvp_sweep.py

5. Enter the session information into the GUI prompt

- Enter subject initials. Because pylink limits the amount of characters that can be used in an EDF file name and will cause an error if the name length exceeds this limit, you should only enter initials and not a whole name to ensure that EDF files save properly when eye-tracking.
- Enter the session information into the remaining fields.
- On Mac, it is recommended that you switch your primary monitor in System Preferences (see more above) instead of using the “Use Second Monitor” option in the GUI.
- Currently there is a bug on the Windows version in which using the ‘fullscreen’ setting in the GUI will prevent the eyetracker calibration dots from displaying. To avoid this, ensure that the screen specs in the parameters file are accurate, and do not select the ‘fullscreen’ option. The program will still display in fullscreen, and the calibration dots will display.

6. Start eyetracking & Begin run

- If eyetracking is used, a grey screen with instructions will appear. You can press C to calibrate and A to automate. If the calibration looks good, press V to validate and A to automate. If the validation looks good, click ‘Accept’ on the eyetracking computer, and then ‘Output/Record’.
 - o You can change the type of calibration by changing the calib_type parameter in the rsvp_params.txt file
 - o Common reasons why the eyetracker may fail to initialize include:
 - The IP address of the eyetracker is incorrectly set in rsvp_params.txt
 - The Ethernet cable between the experiment computer and the Eyelink computer is not connected
 - The Python environment is not set up properly. Most often this is caused by pylink installation issues
- If the calibrate_targ parameter is set to true, the subject will respond with a button press (‘1’ or ‘2’ with MRI, ‘spacebar’ with no MRI) to indicate that they have seen the target image in the center of the screen and four periphery locations.
- If the extended_start parameter is set to true and the MRI option is used, the program will wait for the experimenter to press the spacebar. Then, you should start the MRI sequence, which will trigger the program and cause it to wait 9.5s and then start the sweeps.

7. After the final run, collect & organize the data

- The summary csv and stimuli log csv files are named with the date & time of the session, ensuring that no csv files will be overwritten. However, due to

the character limit of pylink EDF file names, this is not possible and as a result file names are more basic. *If you re-use the same initials and run number, the EDF files will overwrite, so be careful.*

- To easily collect and organize all of the data files, run the pack_data.py file. This will create a directory with up to three sub-directories, depending on the data collected and options selected (Summary, Log, and EDF). The main directory will be created in RSVP_pRF/Data/ and will be named subject_initials_date_RSVP_pRF. If this folder already exists, the time will be added to the end of the directory name to ensure that no data is overwritten. The pack_data.py script will only work if the data files are all in their original
- To run the pack_data.py script, enter the following command in your terminal (on Mac) or your Anaconda Prompt (on Windows):
python pack_data.py

8. When fully done, deactivate Python environment

- To deactivate the Python environment, enter the following command in your terminal (on Mac) or your Anaconda Prompt (on Windows):
conda deactivate

Other important information about the program:

- *When running this program, you cannot use your mouse or keyboard on any other screen.* If you use your mouse and click off of the PsychoPy window, the program will not be able to detect any feedback from the participant. If you do click off of the window, you can easily fix it by clicking back on the window again and leaving the mouse alone.
- To quit the experiment, press the 'esc' key during a sweep. The program cannot be quit in this way before the sweeps begin (i.e., during instructions or target calibration).
- Instruct the subject that they will only have a certain period of time to respond to the stimuli (set in the variable response_period in rsvp_params.txt), and if they correctly respond, the fixation will turn green for the remainder of the response period.
- This program is computationally demanding, as it requires loading & displaying six jpeg images in rapid succession. As such, it is important to maximize program performance, especially if using the program in a MRI scanner. In order to maximize performance, the PsychoPy developers recommend:
 - shutting down all non-essential programs and background processes
 - running in fullscreen mode

- using a computer with a good graphics card
 - use a GPU that supports at least OpenGL 2.0
 - avoid integrated graphics cards, especially Intel
 - Nvidia and AMD graphics cards recommended
- To test if you are dropping frames while running the program, set the 'Testing' parameter in the rsvp_params.txt file to 'True'
- Mac OSx (Mavericks and after) no longer supports fullscreen applications on a second monitor. The program will warn you if you try to do this. Work-arounds include:
 - (Recommended:) Set your second monitor as your first monitor. To do this, open System Preferences > Displays > Arrangement, and drag the white bar at the top of your primary monitor onto the second monitor. Make sure to drag your terminal & relevant windows over to your second window before you run the program.
 - Set your second monitor to mirror your first monitor, and run in fullscreen mode on your first monitor. You will not be able to view the console while the program is running with these settings, but program performance will not be reduced.
 - Run the program without fullscreen mode. This will reduce program performance but will allow you to keep control of your console while the program runs.
- This program can be used with any set of stimuli, and is preset to work with grey, black, or white backgrounds, though any color can be used.
 - To use another set of stimuli, place a folder of png images in RSVP_pRF/Accessory/ , name it Unproc_Stimuli_pngs/ , and run rsvp_stim_prep.py. Edit rsvp_stim_prep.py file as needed.
- To get a .csv file detailing the timing of the program for MRI experiments, run the file RSVP_pRF/Accessory/get_timing.py
 - This will create a .csv file using the current rsvp_params.txt file