# Useful functions in this project

1. **Useful tricks in coding and debugging**
-ctrl+c: interrupt the running program in MATLAB
-command+0 (Mac os): switch to command window
-sca: exit PTB and close all textures and buffers
-clear all; close all; clc
-image: draw an image matrix, useful for examining your matrix variable outside PTB.
-Show help of a certain function xxxx: “help xxxx” or “doc xxxx”
2. **PTB Screen function**
-Type Screen(‘xxxx?’) to show help for sub-function xxxx
-Screen(‘CloseAll’) / sca: exit PTB and close all textures and buffers
-Screen(‘Close’): close specific texture or buffer
-Screen('Preference', 'SkipSyncTests', 1): skip screen synchronization check, useful when you test your script on laptop.
-Screen('OpenWindow'): initialize PTB
-Screen(‘OpenOffscreenWindow’): Open an offscreen buffer or canvas, with which you can quickly display stimulus later.
-Screen('SelectStereoDrawBuffer'): when you want to draw stereo stimulus
-Screen('Flip'): clears anything on the current screen and replaces it with images in the offscreen buffer.
3. **Draw stimulus (on offscreen)**
-Screen('FillOval'): draw an oval
-Screen('DrawText'): present texts on screen
-[x,y]=meshgrid(1:100,1:100): create a Cartesian gird
-imrotate: rotate an image matrix.
- Screen('MakeTexture'): Convert the image matrix into an OpenGL texture, so that you could draw them onto your screen
-Screen('DrawTexture’): draw textures onto screen/window
4. **“Rect” position**-"rect" is in screen coordinates to define the position of your stimulus in many cases. It is a 4-element vector where each number indicates [left, top, right, bottom] respectively in pixels and zero corresponds to the top left corner.
-Type help PsychRects to get a list of other useful rect-related functions-RectCenter: return the center x and y position of a rect
-CenterRectOnPoint: offsets a rect to center it around an x and y position
5. **Randomization**
-rng('shuffle'): Shuffle seeds for random number generator, otherwise the random number sequence would be the same every time when you restart MATLAB.
-rng also outputs the current state of the random number generator, useful if you want to exactly replicate your experiment conditions.
-randperm: randomly orders the numbers from 1:n and is invaluable for balanced randomization designs
-rand: uniformly distributed pseudorandom numbers between (0,1)
6. **Key press recording**
-KbName('UnifyKeyNames'): Unify key code for different operating system
-input: prompt for user input and save to variable
-KbWait: wait until key press
-KbName: map between key codes and key names.
7. **Time control**
-WaitSecs
-pause
-tic; toc
-GetSecs
8. **Save data into m file**-save

## Program procedures

## Wait for key press to start next trial

##

1. Show fixation for 1200 ms



1. Show stimulus until subject’s responses (key press)



1. Wait for 700 ms



1. Repeat 1-4
2. Exit program until key press



**Stimulus**



28 possible locations for target

30 bars

22 bars

We replicate the experiment 2 search task in paper (Zhaoping, L. 2008)

Task: find orientation singleton and report its location as in the left or right of the display as soon as possible.

Conditions in this session:

1. M (monocular): All bars are presented in the same eye only (either left or right eye).
2. B (binocular): All bars are presented identically to both eyes.
3. DC (dichoptic congruity): Target is both orientation and eye-of-origin singleton

Find other parameters (e.g. stimulus size in visual angle) in paper.

# Useful functions for data analysis

**Analysis**-load: load saved data in log file
-strcmp: test whether two strings are identical
-mean
-std: to compute standard error: ste = std / sqrt(sample size)
-squeeze: remove singleton dimensions
-ttest: one-sample or paired-sample t-test

**Plot**
-subplot: draw multiple figures in the same window
-bar, errorbar, plot, boxplot
-set(gca,'XTickLabel'): Add labels on x axis