PsychoPy Lecture 7 SR Research Pylink

Dr. Jibo He

Tsinghua University

hejibolaboratory@pku.org.cn

install the psychopy

```
pip install psychopy -i https://pypi.tuna.tsinghua.edu.cn/simple

pip install psychopy --no-deps

pip install pyyaml
```

在mac电脑下，使用下面的方式安装：

```
brew install
On a MacOS machine, brew can be used to install PsychoPy®:

brew install --cask psychopy
```

additional packages to run example project

```
pip install wx # this package is only for windows OS
```

Import necessary packages

```
import time

from psychopy import core
from psychopy import monitors
from psychopy import visual

import eyelinker
```

Introduce eyelinker package

GitHub link for eyelinker

Define the stimuli drawing canvas

```
monitor = monitors.Monitor('test_monitor', width=53, distance=70)
monitor.setSizePix([1920, 1080])

win = visual.Window([800, 600], units="pix", color=[0, 0, 0], monitor=monitor)
```

Create a text stimuli using visual.TextStim

```
text_stim = visual.TextStim(win, 'Beginning EyeLinker test...')
text_stim.draw()
win.flip()
```

Will attempt to default to MockEyeLinker if no tracker connected

```
tracker = eyelinker.EyeLinker(win, 'test.edf', 'BOTH')
```

initialize connections to the SR Research Eye tracker

```
tracker.initialize_graphics()
tracker.open_edf()
tracker.initialize_tracker()
tracker.send_tracking_settings()
print('Initalization tests passed...')
time.sleep(1)
win.flip()
```

most core functionality

```
tracker.display_eyetracking_instructions()
tracker.setup_tracker()  # forced setup
tracker.calibrate()  # choice given
tracker.send_status('Recording...')
# this send_message function is import for time of interest in later DataViewer analysis
tracker.send_message('TRIALID 1')
tracker.start_recording()
time.sleep(2)
tracker.stop_recording()
print('Basic functionality tests passed...')
time.sleep(1)
```

this send_message function is import for time of interest in later DataViewer analysis

```
tracker.send_message('TRIALID 1')
```

Get real time eye movement data

```
left_eye_gaze, right_eye_gaze = tracker.gaze_data
print(left_eye_gaze)
print(right_eye_gaze)

left_eye_pupil, right_eye_pupil = tracker.pupil_size
print(left_eye_pupil)
print(right_eye_pupil)
```

continuous real time data at 100Hz

```
real_time_data = []
tracker.start_recording()

print('Continuous data start time:')
start_time = core.getTime()
print(start_time)

while core.getTime() < start_time + 1:  # seconds
    real_time_data.append(tracker.gaze_data)
    core.wait(0.01)  # Get a sample every 10 ms
```

test drift correct

```
tracker.drift_correct()
print('Drift correct tests passed...')
time.sleep(1)
```

ending, save Eye Tracking .EDF file

```
# clean up
tracker.close_edf()
tracker.transfer_edf()
tracker.close_connection()
print('\nClean up tests passed...')
```