ERC-2020-SyG 951631 – XSCAPE Project

Internal document

Pupil Core Protocol

versión 1.0.0, revisión 1 15 de abril de 2025 · 1:30

María Silva Gago, Arturo Valiño Pérez

Contenido

Pupil core instalation

For the instalation of pupil core just download the installer from

https://pupil-labs.com/products/core/

Stimulus code

To install the stimulus code we need to follow a series of steps

Install Python

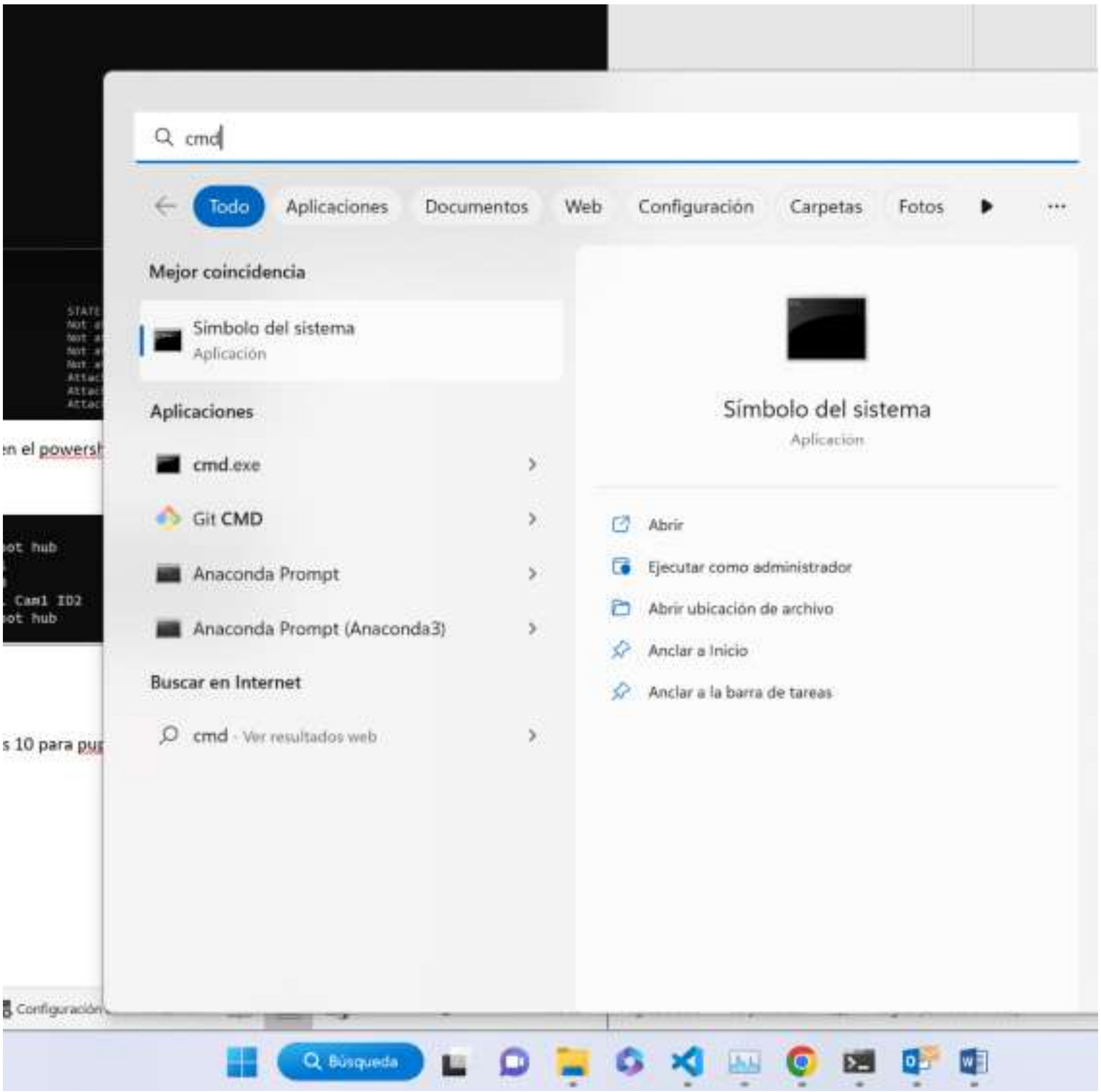
The first thing is to proof if there is a python instalation.

For this we nedd to type:

Python –version

```
Z:\>Python --version
Python 3.10.10
```

To access the console just type "cmd" in windows and press enter. If there is no version installed you will see an error.



Now go to python.org and download the installer for 64 bits in windows for **python 3.10.**

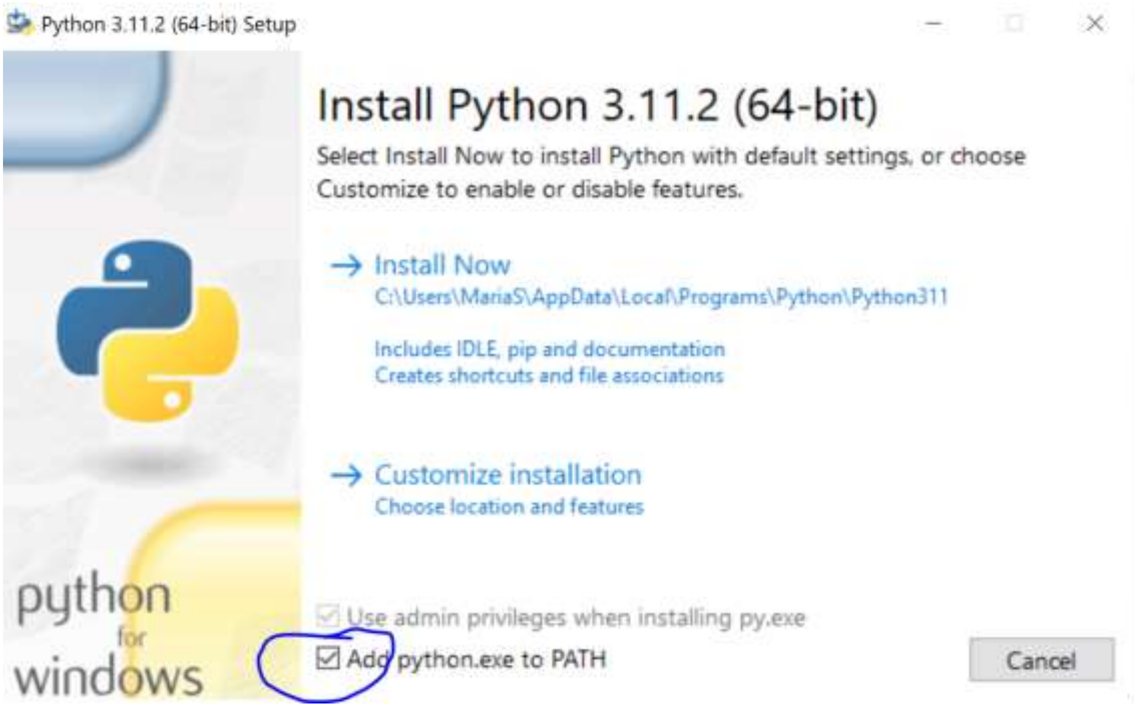
Files

Version	Operating System	Description	MD5 Sum	File Size	GPG	Sigstore	
Gzipped source tarball	Source release		[illegible]	[illegible]	SIG	CRT	SIG
XZ compressed source tarball	Source release		[illegible]	[illegible]	SIG	CRT	SIG
macOS 64-bit universal2 installer	macOS	for macOS 10.9 and later	[illegible]	[illegible]	SIG	CRT	SIG
Windows embeddable package (32-bit)	Windows		[illegible]	[illegible]	SIG	CRT	SIG
Windows embeddable package (64-bit)	Windows		[illegible]	[illegible]	SIG	CRT	SIG
Windows embeddable package (ARM64)	Windows		[illegible]	[illegible]	SIG	CRT	SIG
Windows installer (32-bit)	Windows		[illegible]	[illegible]	SIG	CRT	SIG
Windows installer (64-bit)	Windows	Recommended	[illegible]	[illegible]	SIG	CRT	SIG
Windows installer (ARM64)	Windows	Experimental	[illegible]	[illegible]	SIG	CRT	SIG

Follows the installation instructions and **select disable path limit.**

Important:

Add python path.exe and click install now.



Now check instalation typing again:

Python --version

If everything is correct you should get something like this.

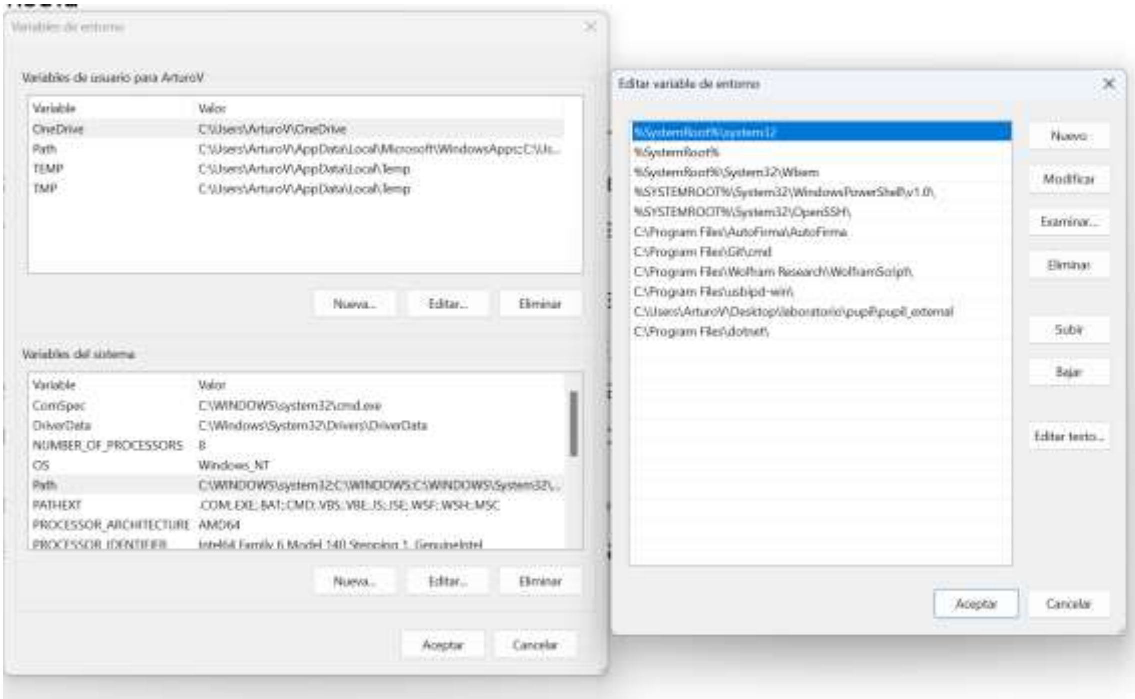
```
Z:\>Python --version
Python 3.10.10
```

If something went wrong it is because the instalation path is not corectly set up. If the happens follow the next step Set up Path Variable, if everythin is o just skip it.

Set up Path Variable

The most important when installing Python is to make sure that the installation path is added in the environment variables. What is the installation path? ***B***asically the directory where the executable (.exe) that was generated when installing the program is located. When the Python command is executed, this executable is searched in the Path environment variable.

To change the environment ***variables,*** the easiest way is to click on the icon windows>write environment variables> click on edit environment variables of the system or of this account (depending on how the installation was done).



Once there we click on new and add the directory where Python is installed. If the installation has been done globally for all users there is no need to do this.

Miniconda instalation

We can install anaconda but to make the installation more lightweight we can install miniconda.

https://docs.conda.io/en/latest/miniconda.html

Windows installers

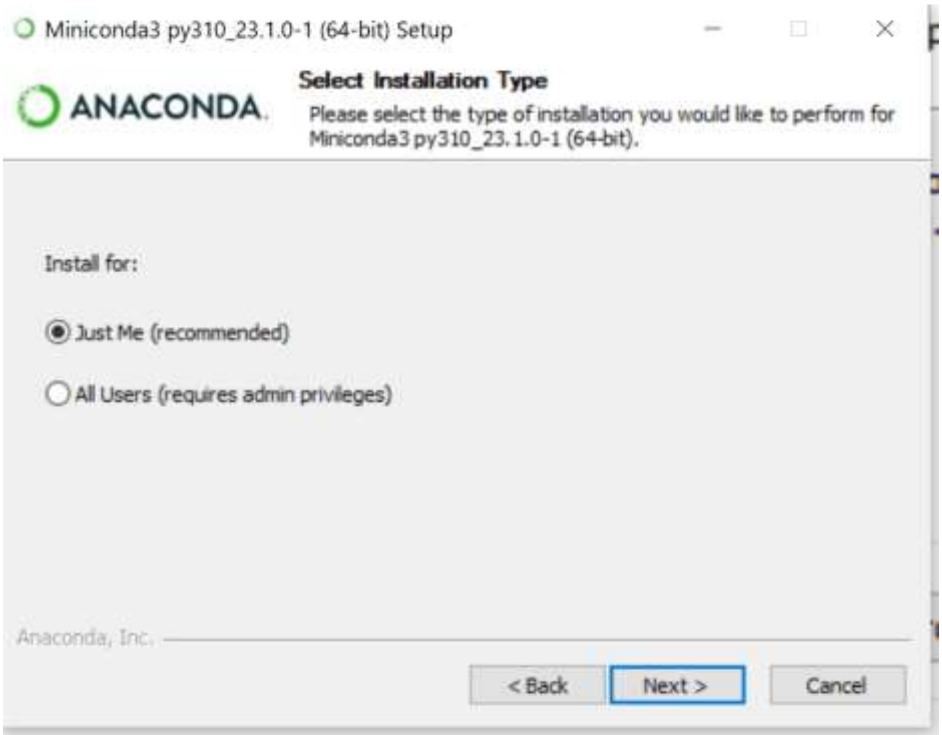
Windows

Python version	Name	Size	SHA256 hash
Python 3.10	Miniconda3 Windows 64-bit	53.2 MiB	[illegible]
Python 3.9	Miniconda3 Windows 64-bit	52.9 MiB	[illegible]
Python 3.8	Miniconda3 Windows 64-bit	52.4 MiB	[illegible]
Python 3.7	Miniconda3 Windows 64-bit	50.6 MiB	[illegible]
Python 3.9	Miniconda3 Windows 32-bit	67.8 MiB	[illegible]
Python 3.8	Miniconda3 Windows 32-bit	66.8 MiB	[illegible]
Python 3.7	Miniconda3 Windows 32-bit	65.5 MiB	[illegible]

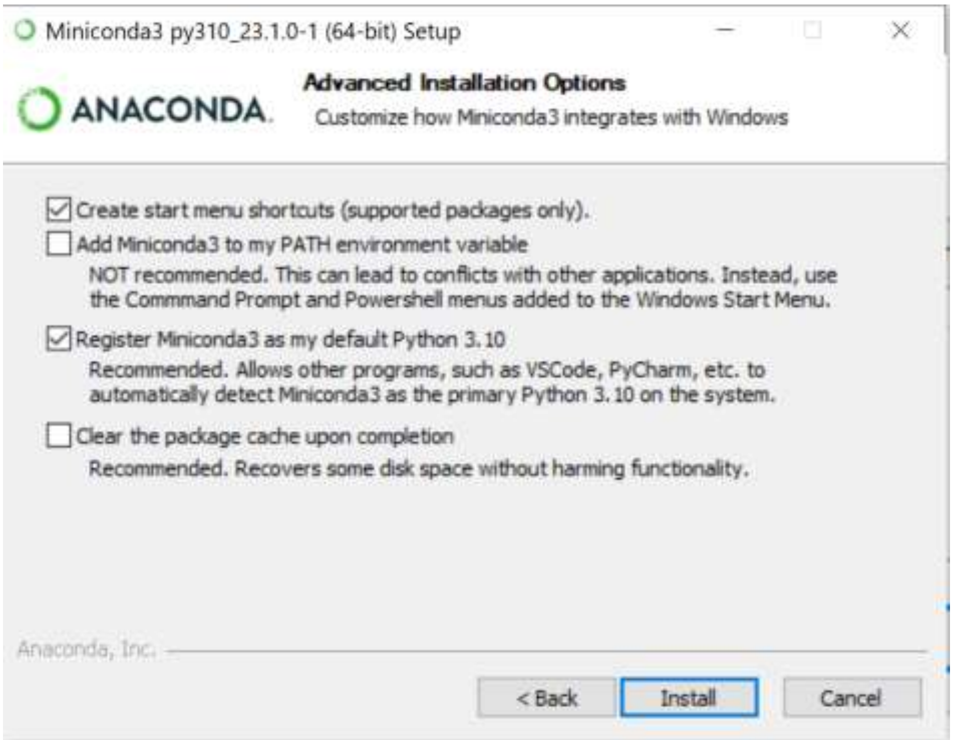
Download the installer for python 10 and check and follow the insttucions.

Important:

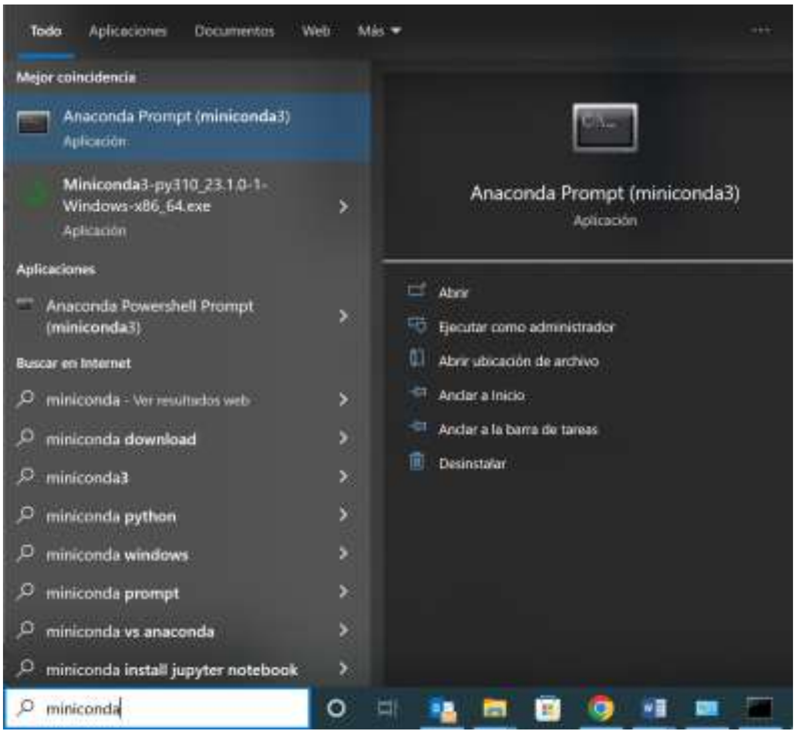
Install miniconda locally.



- Mark these fields



Check that installation is correct, typing “miniconda” in windows search bar. A console named “Anaconda Prompt)” should appear.



In the future:

How to use the program during stimulation

Download de code

Install de code from github (complete this in the future):

See if git is installed

Do a git clone (complete this in the future)

For the moment:

Copy and paste the code from the project folder in Xscape to the place where you want the code to be in the computer. Copy the folder called “screen_stimulus_presentation”.

Directory:

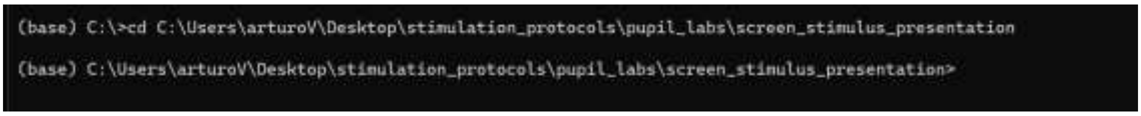
P:\Proyectos\2020\PY200016 - XSCAPE\Trabajo\02 DRAFT WORK\04 CODE FOR EXPERIMENTS

Paste in Deskopt, Local Disk C or whatever you want.

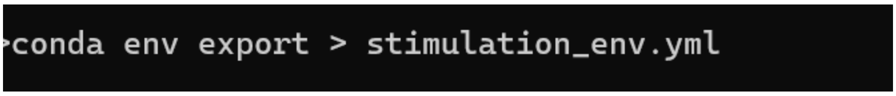
Install de environment

Now we need to install for the code to work

1) Open miniconda console
2) Type in the console:
    a. cd <directory where the code is> (<> means you need to write something there, without the symbols)



Now type:
conda env create -f stimulation_env.yaml

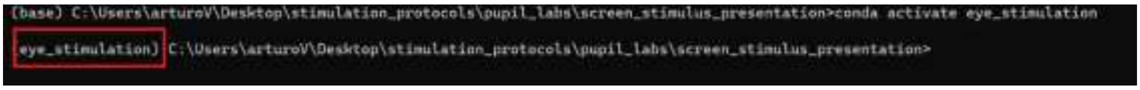


Wait until the installation is finished

Run the experiment

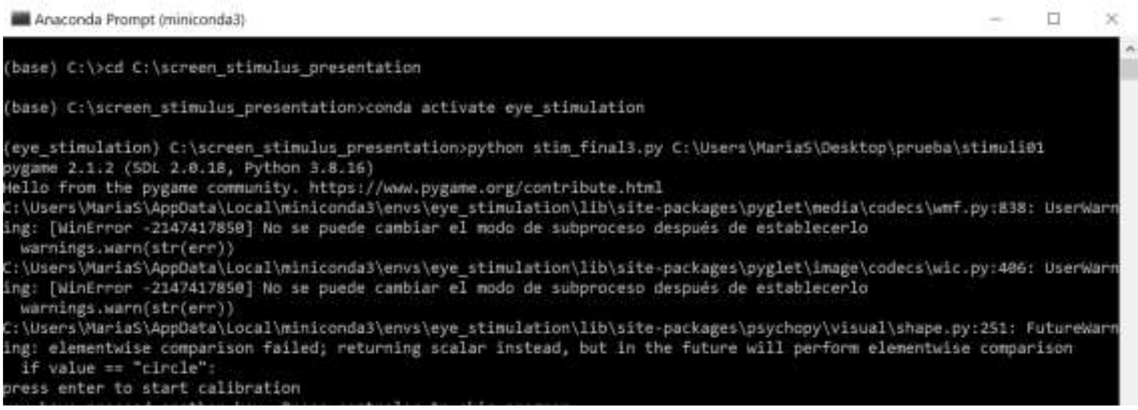
1) Change the main screen in configuration. The main screen must be the displaying screen in order to run the calibration in that screen.
2) Connect and run pupil capture
3) Open the miniconda console.
4) Type in the console:
    a. cd <directory where the code is>
    b. activate the environment. Type:
        conda activate eye_stimulation

As you can see now in the left the environment is activated



5) type:
    a. python stim_final3.py <directory where the image stimulations will be saved>

Before the experiment the script should be like this:



(base) C:\>cd C:\screen_stimulus_presentation

(base) C:\screen_stimulus_presentation>conda activate eye_stimulation

(eye_stimulation) C:\screen_stimulus_presentation>python stim_final3.py C:\Users\MariaS\Desktop\prueba\stimuli01

pygame 2.1.2 (SDL 2.0.18, Python 3.8.16)

Hello from the pygame community. https://www.pygame.org/contribute.html

C:\Users\MariaS\AppData\Local\miniconda3\envs\eye_stimulation\lib\site-packages\pyglet\media\codecs\wmf.py:838: UserWarning: [WinError -2147417850] No se puede cambiar el modo de subproceso después de establecerlo

warnings.warn(str(err))

C:\Users\MariaS\AppData\Local\miniconda3\envs\eye_stimulation\lib\site-packages\pyglet\image\codecs\wic.py:406: UserWarning: [WinError -2147417850] No se puede cambiar el modo de subproceso después de establecerlo

warnings.warn(str(err))

C:\Users\MariaS\AppData\Local\miniconda3\envs\eye_stimulation\lib\site-packages\psychopy\visual\shape.py:251: FutureWarning: elementwise comparison failed; returning scalar instead, but in the future will perform elementwise comparison

if value == "circle":

press enter to start calibration

Control + C is used to start again the console.

Then basically follow the instructions in the console to run the experiment.

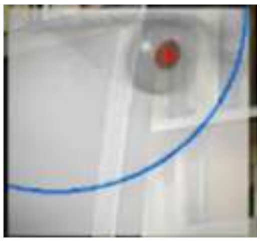
Pupil Capture

Pupil Core headsets have two types of cameras attached. One camera records the subject's field of vision (world camera). Additionally, there are one or more cameras recording the participant's eye movements (eye cameras). The data collected during the calibration period is used afterwards to correlate the world camera with the eye cameras. See https://docs.pupil-labs.com/core/hardware/ for further details and explanation videos about how to adjust the glasses.

Requirements to participants before the experiment

- Do not wear glasses (normal contact lenses are allowed but not false pupil shape lenses)
- Do not wear make up or false eyelashed
- To be comfortable on the seat

1. First, read the manual of Pupil Core: https://docs.pupil-labs.com/core/
2. Make sure that the participant is comfortable on the seat, placed at a distance of half a meter from the screen where the stimuli will be displayed. Then, placing the glasses on the participant. During the testing, avoid that the participant look excessively downward, closing the eyes and not sliding down on the seat. In this sense, inform the subject that if they need to look down they should turn their head instead of their eyes.



Example of a bad eye position

3. Open Pupil Capture.
4. Adjust lateral cameras to each eye. The eye should be as centred as possible in its camera view (see Fig. 1.). Try to avoid the eyes being close to the limits of the eye's image. Eye 0 refers to left eye, and Eye 1 is the right eye. If necessary, use orange extenders to adjust. Adjust upper camera (world camera) to the experiment space (screen, table...).
5. Ask the participant to slowly look around while keeping the head stable for a couple of seconds to sample different gaze angles. This will allow the eye model to stabilize at the start.
6. Click in R – Start recording (turns into red).

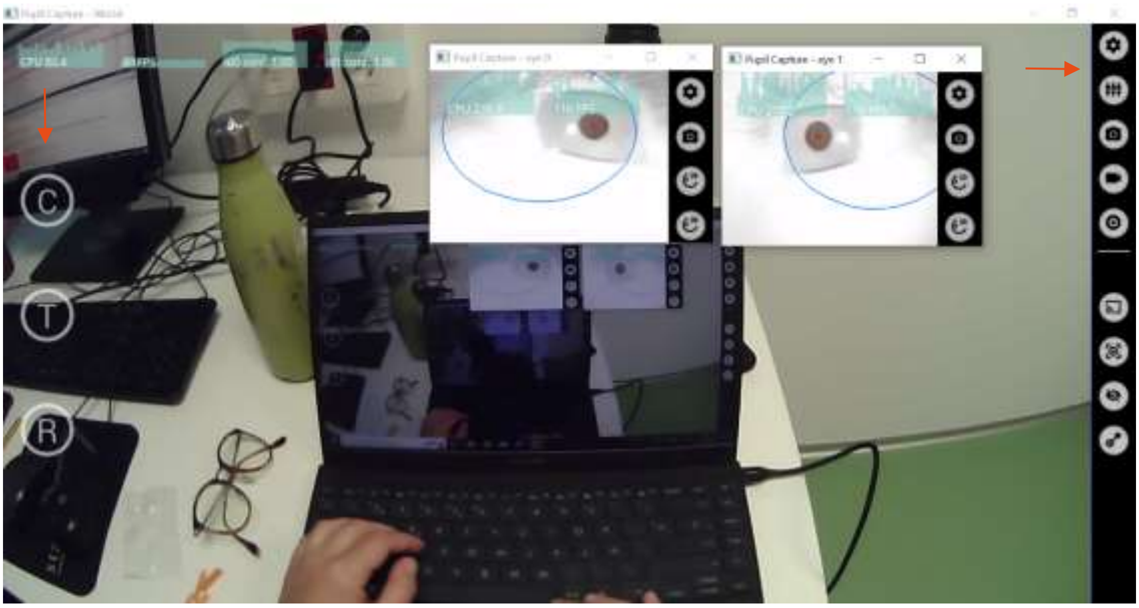


Fig. 1. Screenshot of the software surface before starting the recording. On the left side, there are the buttons of calibration (C), validation (V) and recording (R). On the right, there is the menu bar when different features can be activated called plugins (i.e., fixation plugin).

7. Click in C – Start calibration. Calibration consists of keeping the gaze fixed on the targets that appear sequentially on the screen.

8. Select monitor to display the calibration in the calibration button. Usually it is Monitor 2 if you want to display the calibration and the stimuli in another screen. In this case, make sure that you are using “duplicate screen” during calibration, and that the participant only sees one of them.
9. Start recording/experiment session
10. During calibration, the participant’s head must be immobile, only the eyes can be moved. The calibration error must be less than 3 degrees of angular accuracy and 1 degrees of angular precision. This information can be seen after the calibration in the screen of Pupil Capture, in the script or in the accuracy visualizer plugin (right menu). Whenever possible, calibration points should be displayed at a fixed distance to the participant. If the distance of your stimuli varies, place calibration points at varying distances. Calibration points must cover the outermost bounds of your gaze.

Fig. 2. Calibration point

11. Click in T - Calibration validation. The calibration is not changed; it simply indicates the error and check the accuracy. If the validation reports a high accuracy error, you need to perform another calibration. Make sure to use different points for validation than you used for calibration. The orientation to where the calibration error occurs is also displayed.
12. In experiments using a screen for displaying stimuli through PowerPoint, after calibration and validation, change the mode to “expand screen”. Display the ppt in the expanded screen (the same using for calibration) and minimize the ppt in moderator view to be able to view Pupil Capture
13. When the experiment is finished, perform another validation to estimate of how much slippage error is accumulated during the experiment.
14. Click on R - Stop Recording

15. In case you will carry out the experiment using a screen, it is recommended to use markers (https://docs.pupil-labs.com/core/software/pupil-capture/#surface-tracking) to define an area of interest involving the whole screen (i.e., one marker per screen corner). Be sure that the markers are in order (ID1, ID2, ID3…) and the correct position and orientation.
16. Before closing Pupil Capture, it is recommended to copy the script to save the calibration error data (angular precision and accuracy).

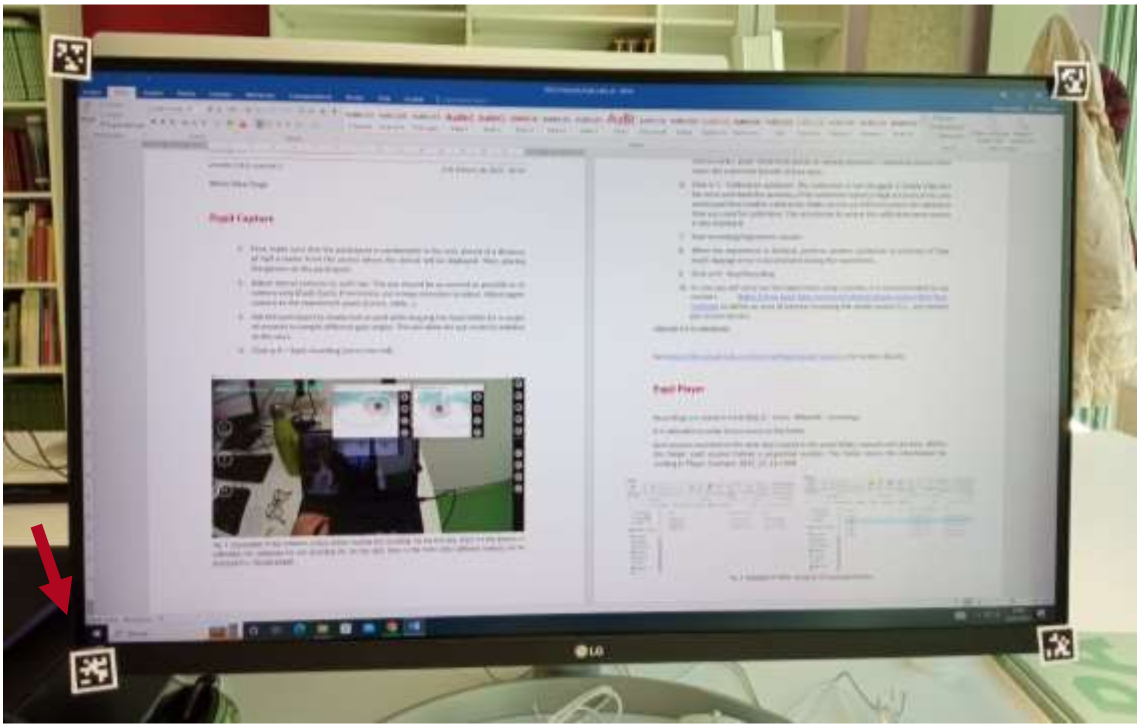


Fig. 3. Example of markers

See https://docs.pupil-labs.com/core/software/pupil-capture/ for further details.

Pupil Player

Recordings are saved in Local Disk :C - Users - #Name# - recordings.

It is advisable to make direct access to this folder.

Each session recorded on the same day is saved in the same folder, named with the date. Within this folder, each session follows a sequential number. This folder stores the information for reading in Player. Example: 2022_12_14 > 000

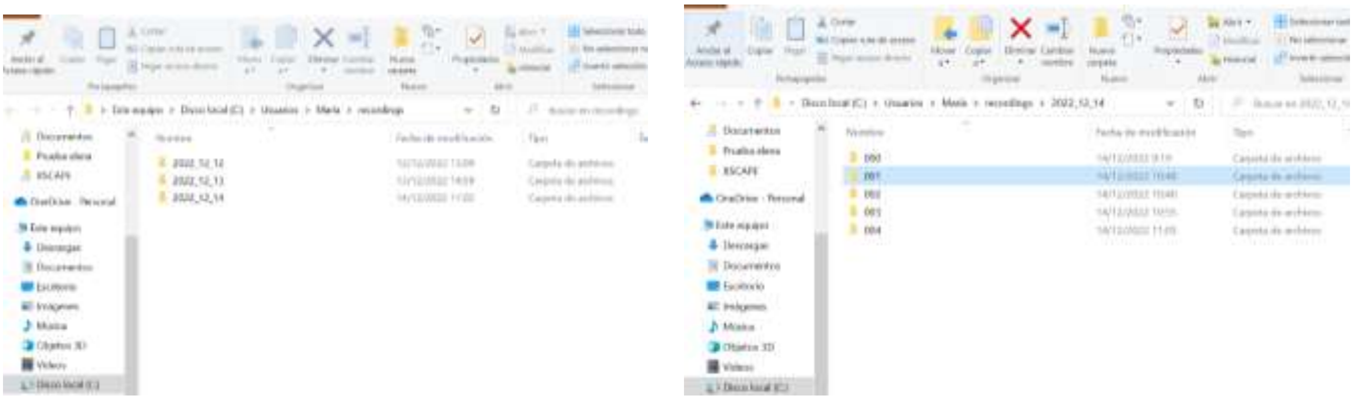


Fig. 3. Examples of folder structure of recording sessions.

1. Open Pupil Player.
2. Drag folder #000 to Player (grey screen)
3. Recording display.
4. Now, the video recording can be seen. You can pause and adjust some features according to the purpose.

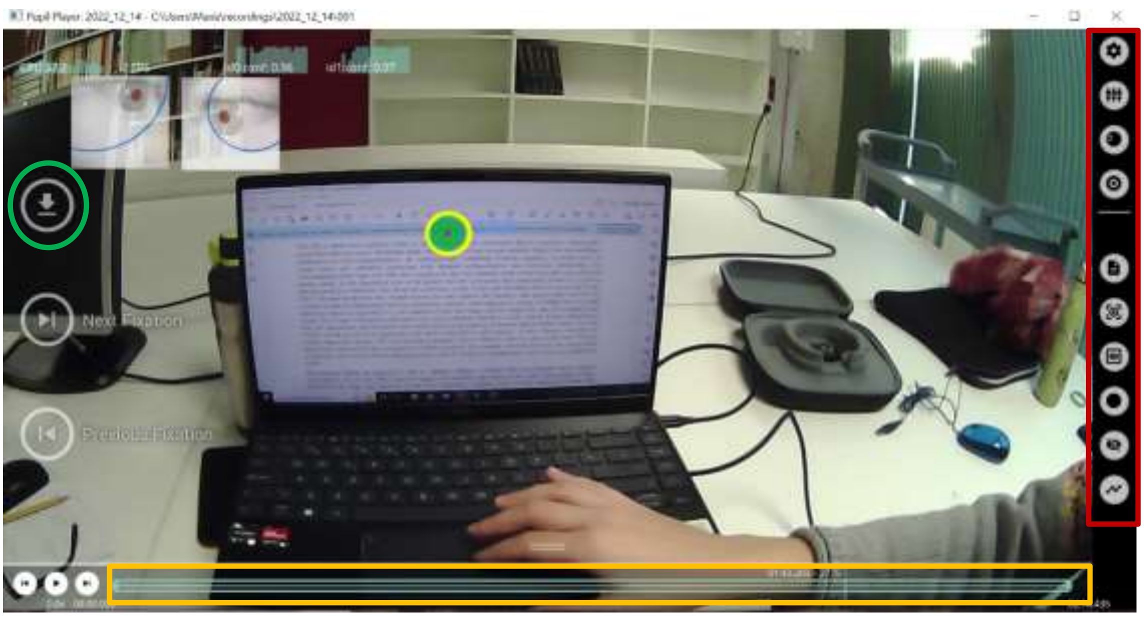


Fig. 4. Example of video recording. Right bar is the menu (red). Each icon refers to a different window. On the left (top) we find download icon (green). At the bottom of the screen is the video bar when you can see the timeline (orange) and adjust the speed

5. Activate necessary plugins in the Plugin Manager (second icon in the right bar). E.g.: blinks, fixations, raw data. Make sure you load raw data exporter plugin to get all the data.
6. Select a range frame using the timeline (green rectangle on the bottom) to focus the analysis on that period of time.

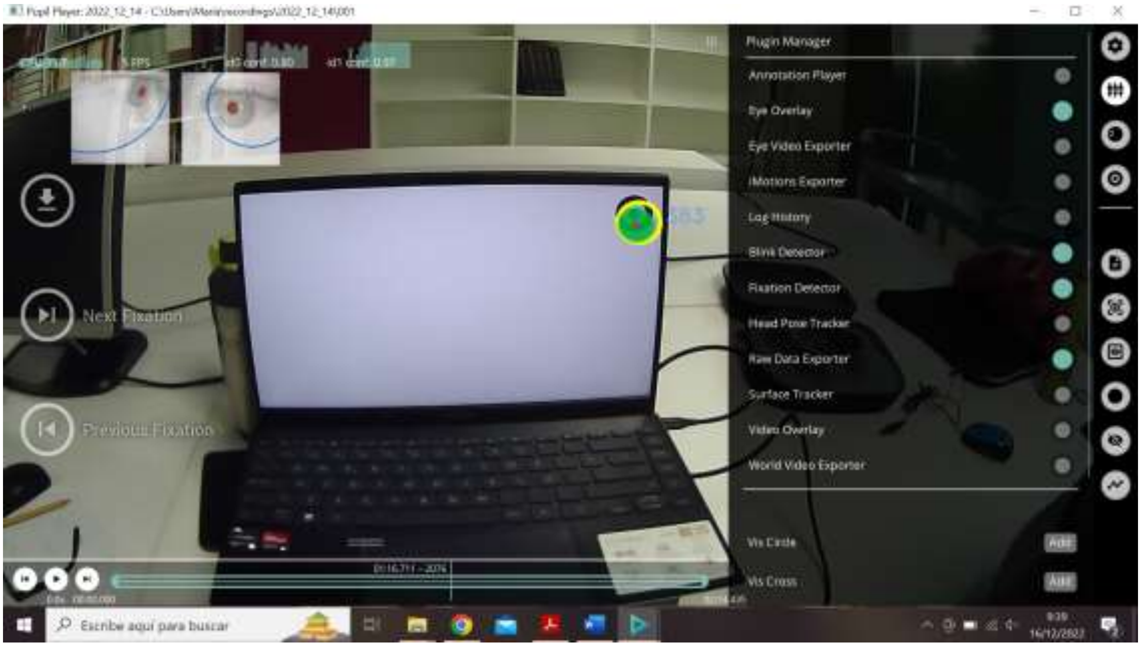


Fig. 5. Example of video recording during calibration and Plugin Manager.

7. Click on download. A new folder "exports" in #000 is created automatically.
8. Raw data includes several files: "gaze positions" (coordinates of the gaze, general and for each eye), "pupil positions" (coordinates of each eye in their respective cameras/views), "fixations" (if plugin activated).
9. Add the surface tracker plugin in the plugin manager menu (second button on the right). Select a frame where all the markers are identified and define the area of interest. When exported the files, a new folder will be created with the information about the surfaces defined, i.e. fixations or gaze positions in that surface.

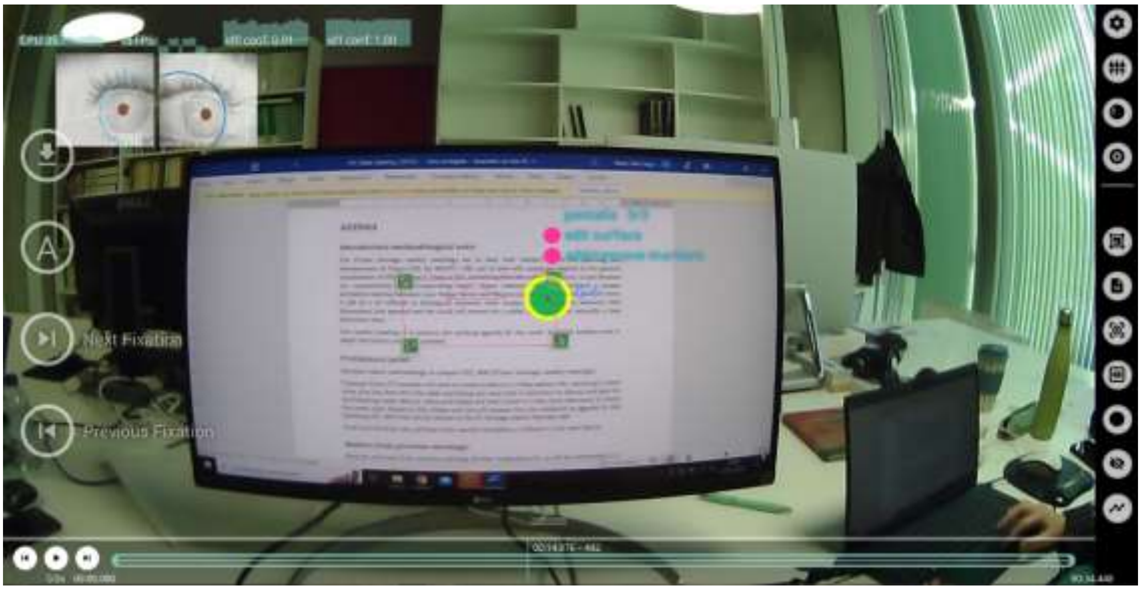


Fig. 6. Area of interest defined with markers.

a. Add surface - click on the Add surface button when the markers you want to user are visible or just click the circular A button in the left hand side of the screen.

b. Surface name and size - In the Surface Tracker menu, define the surface name and real world size. Note - defining size is important as it will affect how heatmaps are rendered.

c. Set trim marks - optional, but if you want to export data for a specific range, then you should set the trim marks.

d. Recalculate gaze distributions - click the (Re)calculate gaze distributions button after specifying surface sizes. You should now see heatmaps in the Player window (if gaze positions were within your defined surfaces).

e. Export gaze and surface data - click e and all surface metrics reports will be exported and saved for your trim section within your export folder

Further information at https://docs.pupil-labs.com/core/software/pupil-player