
Head Tracking Documentation

Release 0.0.3

Andy Lustig

May 28, 2020

Contents

1	Installation and Setup	3
1.1	Install the Python 3.7 version of Anaconda	3
1.2	Create a new virtual environment	3
1.3	Switch into your new environment	3
1.4	Install OpenCV 4.2 in your environment	3
2	Processing Video Files	5
2.1	Prepare video processing directory	5
2.2	Execute tracking script from the command line	5
2.3	Resulting Output	5

This is the documentation for a [Python](#) script used for tracking head mounted LEDs. Primarily using the [OpenCV](#) library, the script takes in multiple .mpg video files and outputs a single annotated .mp4 video file as well as a .csv file containing the locations of the tracked points within each frame.

1.1 Install the Python 3.7 version of Anaconda

[Download Here](#)

1.2 Create a new virtual environment

Open a terminal window and enter the following:

```
conda create -n trackingEnv python=3.8 numpy pandas
```

This will create a Python 3.8 environment called “trackingEnv” with the numpy and pandas libraries installed.

1.3 Switch into your new environment

```
conda activate trackingEnv
```

(trackingEnv) should now be shown at the start of the command line.

1.4 Install OpenCV 4.2 in your environment

```
conda install -c conda-forge opencv=4.2.0
```


2.1 Prepare video processing directory

Place `trackHead.py` into a folder that contains the `timestamps.csv` and `.mpg` files that you want processed

2.2 Execute tracking script from the command line

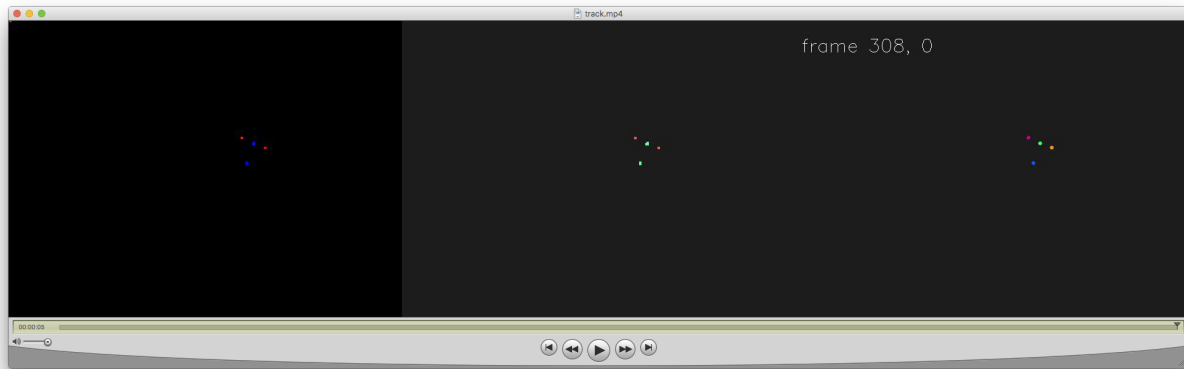
1. Open a terminal window
2. *Switch into your `trackingEnv` if you are not already in it*
3. `cd` into your video processing directory
4. Run the `trackHead.py` script by entering the following:

```
python trackHead.py
```

2.3 Resulting Output

2.3.1 Annotated footage

All of the separate `.mpg` files will be combined into a single `.mp4` that contains 3 frames stiched together.



The left frame is the original video footage. The center frame shows the pixels that remain after filtering for red and blue. The right frame places circular marks at the centers of the filtered pixel clusters.

2.3.2 Tracked coordinates data

A .csv file that combines the timestamp.csv data with x,y coordinates of the tracked LEDs.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	0	83	417	1	0	417	413.833	419.167	401.833	445.167	407.5	429.5	385.5	425.5
2	1	83	433	1	0	433	413.833	417.833	401.833	445.167	408	429.316	385.5	425.5
3	2	83	449	1	0	449	415.167	417.833	403.167	443.833	409.5	429.5	385.643	425.714
4	3	83	467	1	0	467	415.5	417.5	403.5	443.5	409.5	429.5	387.5	425.5
5	4	83	483	1	0	483	415.5	417.5	403.833	443.167	410.6	429	387.833	425.833
6	5	83	499	1	0	499	415.833	417.167	405.167	443.167	411.5	429.5	389.5	425.5
7	6	83	516	1	0	516	416.5	417.5	405.5	443.5	411.5	429.5	389.5	425.5
8	7	83	532	1	0	532	416.778	417.556	405.5	443.5	412.5	429	391.5	425.5
9	8	83	548	1	0	548	417.167	417.833	405.5	443.5	413.5	429.5	391.5	425.5
10	9	83	566	1	0	566	417.167	417.833	405.5	443.5	413.5	429.5	391.5	426.5
11	10	83	582	1	0	582	417.5	418.5	405.5	443.5	413.5	429.5	391.5	426.5
12	11	83	598	1	0	598	417.5	418.5	405.5	444.5	413.833	429.833	392.5	427.5
13	12	83	616	1	0	616	417.5	419.5	405.5	444.5	413.5	431.5	393.167	427.167
14	13	83	632	1	0	632	417.5	419.5	405.5	444.5	413.833	431.167	393.5	427.5
15	14	83	666	1	0	666	417.833	419.167	405.833	443.833	413.833	431.167	393.5	427.5
16	15	83	682	1	0	682	417.833	419.167	406.5	443.5	414.529	430.118	393.833	427.167
17	16	83	716	1	0	716	417.833	419.167	406.5	443.5	415.5	429.5	394.5	427.5
18	17	83	732	1	0	732	417.5	418.5	406.5	443.5	415.5	429.5	394.5	427.5