

<sup>1</sup> ***pi\_tailtrack: A compact, inexpensive, and open-source behaviour-tracking system for head-restrained zebrafish***

<sup>4</sup> **Owen Randlett<sup>1</sup>**

<sup>\*Correspondence:</sup>

<sup>5</sup> [owen.randlett@univ-lyon1.fr](mailto:owen.randlett@univ-lyon1.fr) <sup>6</sup> (OR) <sup>1</sup> Laboratoire MeLiS, UCBL - CNRS UMR5284 - Inserm U1314, Institut NeuroMyoGène, Faculté de Médecine et de Pharmacie, 8 avenue Rockefeller, 69008, Lyon, France

<sup>7</sup> **Abstract**

<sup>8</sup> Quantifying animal behavior during microscopy is crucial to associate optically recorded neural activity with  
<sup>9</sup> behavioural outputs and states. Here I describe an imaging and tracking system for head-restrained larval  
<sup>10</sup> zebrafish compatible with functional microscopy. This system is based on the Raspberry Pi computer, Pi NoIR  
<sup>11</sup> camera, and open-source software for the real-time tail segmentation and skeletonization of the zebrafish tail at  
<sup>12</sup> over 100hz. This allows for precise and long-term analyses of swimming behaviour, that can be related to  
<sup>13</sup> functional signals recorded in individual neurons. This system offers a simple but performant solution for  
<sup>14</sup> quantifying the behavior of head-restrained larval zebrafish, which can be built for 340€.  
<sup>15</sup>

<sup>16</sup>

<sup>17</sup> **Introduction**

<sup>18</sup> A chief application of the larval zebrafish for neuroscience is to image the activity of neurons in the intact and  
<sup>19</sup> behaving animal using microscopy. This is facilitated by its translucent and small brain, measuring approximately  
<sup>20</sup> 0.1 mm<sup>3</sup>. By expressing genetically encoded indicators, such as the GCaMP Ca<sup>2+</sup> sensors (*Akerboom et al., 2012*;  
<sup>21</sup> *Chen et al., 2013*), signals related to the activity of practically any or all neurons can be recorded from the larval  
<sup>22</sup> zebrafish brain (*Ahrens et al., 2012; Portugues et al., 2014*).

<sup>23</sup> Ca<sup>2+</sup> imaging can be performed with standard microscopes, but such systems are not equipped for monitoring  
<sup>24</sup> the behaviour of the animal. Therefore, any analyses directly relating neural activity to behaviour will require the  
<sup>25</sup> integration of a behavioural recording apparatus. Behavioural recording is typically done in the context of custom-  
<sup>26</sup> built microscopes, which can be designed explicitly with this behaviour-monitoring goal in mind. However, many  
<sup>27</sup> groups (including my own) have neither the financial nor technical means to implement such a complete system.  
<sup>28</sup> We rely on microscope equipment in a shared core facility. Such microscopes generally cannot be substantially or  
<sup>29</sup> permanently modified, and often present physical and optical constraints that make installing a behaviour imaging  
<sup>30</sup> system challenging.

<sup>31</sup> Here I present a solution for this problem based on the Raspberry Pi computer, that I call *pi\_tailtrack*. The system  
<sup>32</sup> includes illumination, camera, computer and software, yielding a complete setup that is compact, inexpensive, and  
<sup>33</sup> self-contained. The *pi\_tailtrack* system can reliably track larval zebrafish behaviour at over 100hz while performing  
<sup>34</sup> functional imaging experiments.

## 35 Results and Discussion

### 36 Design goals

37 I wanted to track the swimming behaviour of head-restrained larval zebrafish while performing  $\text{Ca}^{2+}$  imaging. There  
38 are many ways that this might be accomplished, but I wanted a system that was:

- 39 1. Able to identify and characterize individual swimming events while we are imaging the brain using 2-photon  
40 microscopy.
- 41 2. Compact and self contained, so that it can be easily and rapidly installed and removed for our imaging sessions  
42 on a shared microscope.
- 43 3. Made using low-cost and open source hardware and software to facilitate re-use in other contexts, and be-  
44 cause I am a ~~cheap~~ financially responsible researcher.

### 45 Using a Raspberry Pi camera to image the larval zebrafish tail

46 The Raspberry Pi is a very inexpensive, credit-card-sized computer that plugs into a standard monitor, keyboard, and  
47 mouse. The Raspberry Pi's open-source nature and large user community, and its ability to control and interface  
48 with a variety of devices and sensors make it a powerful and accessible platform for developing and sharing custom  
49 neuroscience and behavioural research tools. Indeed many such systems have been developed in recent years  
50 based around the Raspberry Pi and the Pi Camera, and especially the IR-sensitive Pi NoIR camera, as an acquisition  
51 device (*Geissmann et al., 2017; Maia Chagas et al., 2017; Saunders et al., 2019; Tadres and Louis, 2020; Broussard  
et al., 2022*).

52 However, obtaining sufficient resolution and contrast to resolve the larval zebrafish tail is challenging since the  
53 tail is very narrow ( $\approx 0.25\text{mm}$  diameter), and nearly transparent. This is especially true in de-pigmented animals that  
54 are generally used for brain imaging due to their lack of melanin pigment over the brain (e.g. *mitfa*/Nacre mutants, or  
55 larvae treated with N-Phenylthiourea). This also removes melanin pigment from the tail, increasing its transparency  
56 and making it harder to image and track. Thus, it was not clear if the 26€ Pi NoIR Camera would be up to this task.

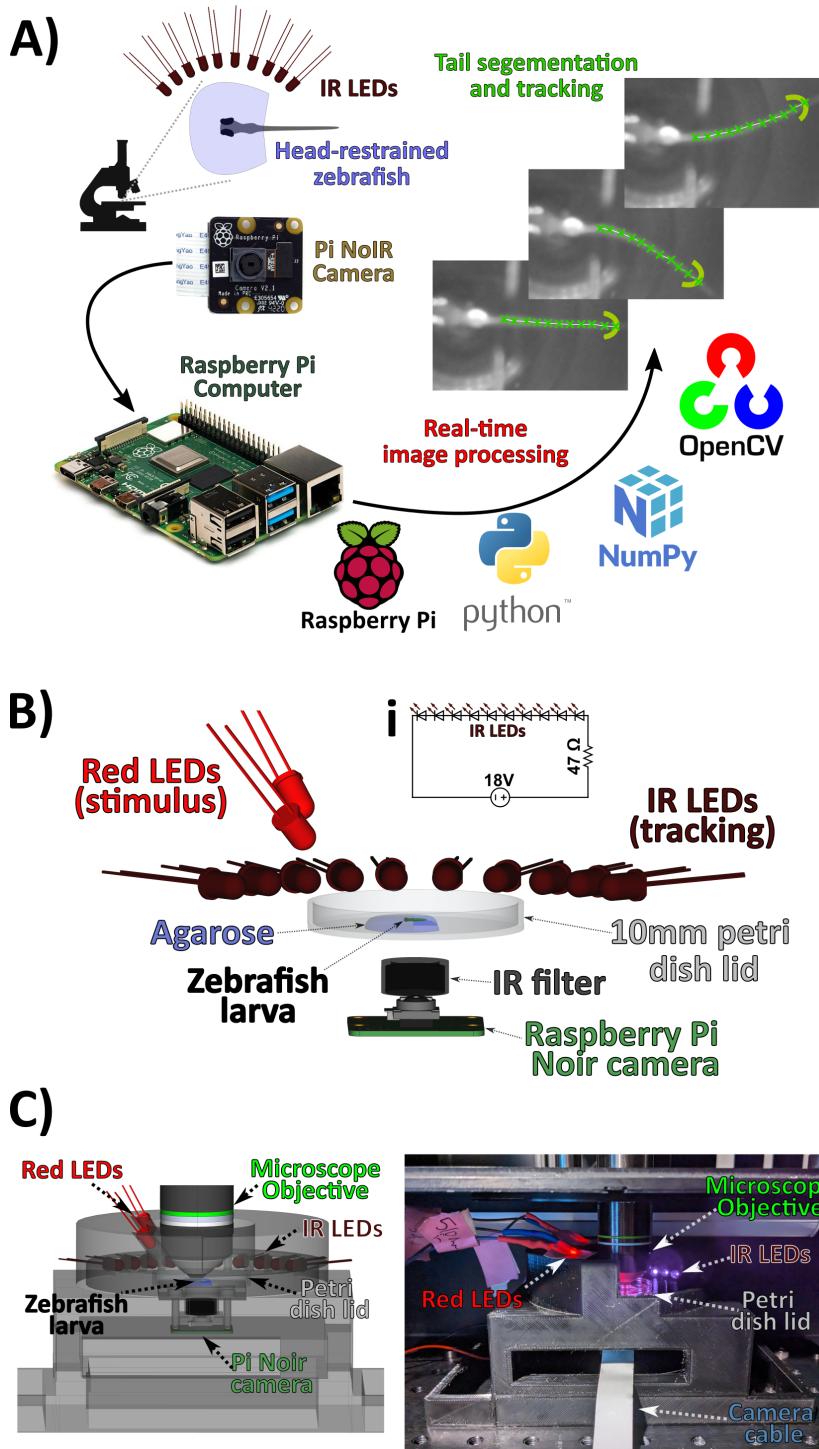
57 The stock lens configuration on the Pi Camera is also not designed for macro photography, and has a minimum  
58 focus distance of 50cm. But, extension tubes are a well-known macro-photography hack that work by increasing the  
59 distance between the lens and the camera (*Wikipedia contributors, 2022*). Increasing this distance acts to decrease  
60 the focus distance of the optical system, increasing the maximal magnification. By unscrewing the lens of the Pi  
61 NoIR camera until just before it falls off, it is possible to focus on objects at a 2 cm distance, allowing for sufficient  
62 magnification to observe the and track the tail of *mitfa* mutant zebrafish (*Figure 1, Figure 2*).

63 A second challenge is that larval zebrafish move their tails very rapidly, with a tail-beat frequency of between 20-  
64 40 hz for normal swimming, which can increase to 100 hz during burst/escape swimming (*Budick and O'Malley, 2000;  
65 Muller, 2004; Severi et al., 2014*). The V2.1 camera documentation indicates maximum frame rate of 30hz, which is  
66 insufficient for imaging tail dynamics. However, by adopting a cropped sensor configuration, and by omitting the  
67 JPG compression step in image processing, the camera can be pushed to image at up to 1000hz (*Elder, 2019*). I  
68 adopted a configuration where I imaged with a fixed gain/ISO of 800 in auto-exposure mode, and with a cropped  
69 sensor of 128x128 pixels covering 3.5x3.5 mm field of view. This gives sufficient spatial resolution to observe and  
70 track the tail of the fish ( $27 \mu\text{m}/\text{px}$ ), and most importantly, minimal CPU load. This frees the limited CPU resources  
71 on the Raspberry Pi to be used for real-time image processing and tail tracking.

### 73 Hardware Setup

74 The short 2 cm focal distance between the animal and the camera allowed for a compact and direct imaging setup,  
75 where the camera is mounted directly below the larva (*Figure 1B*). This avoids the need for any mirrors, and frees  
76 the space above the animal for the microscope objective, and any stimulus apparatus necessary. In our case we  
77 use red LEDs to provide visual stimuli to the larvae (*Lamiré et al., 2022*).

78 To illuminate the larvae and visualize the tail, I used 890nm IR LEDs. Using the IR LEDs as an oblique illumina-  
79 tion source generated a nicely resolved image of the *mitfa* mutant zebrafish tail that was sufficient for reliable  
80 identification and tracking (*Figure 2*). IR LEDs were wired in a simple circuit, with 10 LEDs in a series, powered by  
81 a 18V DC power supply and a 47ohm current limiting resistor (*Figure 1Bi*). Using these exact Voltage/Resistance



**Figure 1. *pi\_tailtrack* apparatus.**

**A)** The zebrafish larvae being imaged under the microscope is illuminated with infra-red (IR) LEDs, and imaged with the IR-sensitive Raspberry Pi NoIR Camera. Image acquisition and processing is done with a Raspberry Pi Computer and open-source Python packages. The zebrafish tail is identified and segmented in real-time as a sequence of 10 tail segments (green X's).

**B)** Rendering of the main components of the apparatus. IR leds illuminate the zebrafish larvae that is head-restrained in agarose in a 35mm diameter petri dish lid. An IR filter blocks the visible stimulus lights (Red LEDs), and the microscope laser from reaching the Raspberry Pi NoIR camera suspended below the fish. **(i)** Wiring diagram for powering the IR LEDs.

**C)** Rendering including the the 3D printed mount and microscope objective.

**D)** Annotated photograph of the *pi\_tailtrack* apparatus.

82 configurations is not important, provided a relevant power supply and resistor are chosen to match the LED char-  
83 acteristics (forward voltage =1.4V, current = 100mA, for our 890nm LEDs: see for example [amplifiedparts.com](#): LED  
84 Parallel/Series Calculator).

85 We used an 880 nm bandpass filter in front of the Raspberry Pi NoIR camera module to selectively pass the IR  
86 LED light. This filter is essential to block the intense microscope laser light, which will obscure the image of the  
87 fish by saturating (and likely damaging) the camera sensor. Notably, this filter it is the most expensive part in the  
88 setup, costing more than the computer and camera, combined ([Table 1](#)). With our typical 2-photon GFP/GCaMP  
89 imaging settings and the laser tuned to 930nm, laser light is not visible in the camera image. Using such a bandpass  
90 filter in the 880 nm range should allow this system to be compatible with many other imaging modalities (confo-  
91 cal, epifluorescence, brightfield, etc), provided that the excitation wavelengths are not in the ≈870-900nm range,  
92 and the microscope system effectively filters out the 890nm light from the LEDs. If necessary, these wavelength  
93 characteristics can be adapted using different LED and filter components.

94 To house the system components I used a 3D printed mount ([Figure 1C,D](#)). This was designed using FreeCAD  
95 ([freecad.org](#), [FreeCAD file](#)), and 3D printed in black PETG using and Creality Ender 3 Pro 3D printer. It consists of the  
96 main body shape that holds the the camera, IR filter, red stimulus LEDs above the fish, and IR LEDs in the oblique  
97 illumination configuration ([Main Shape](#)). An insert is placed into the depression above the IR filter, forming the plat-  
98 form onto which the zebrafish dish is placed ([Depression Insert](#)). The final 3D printed component is a semicircular  
99 shape that completes the encirclement of the objective, and helps minimize light scattering ([Figure 2Bi](#), [Semicircle](#)  
100 [STL file](#)).

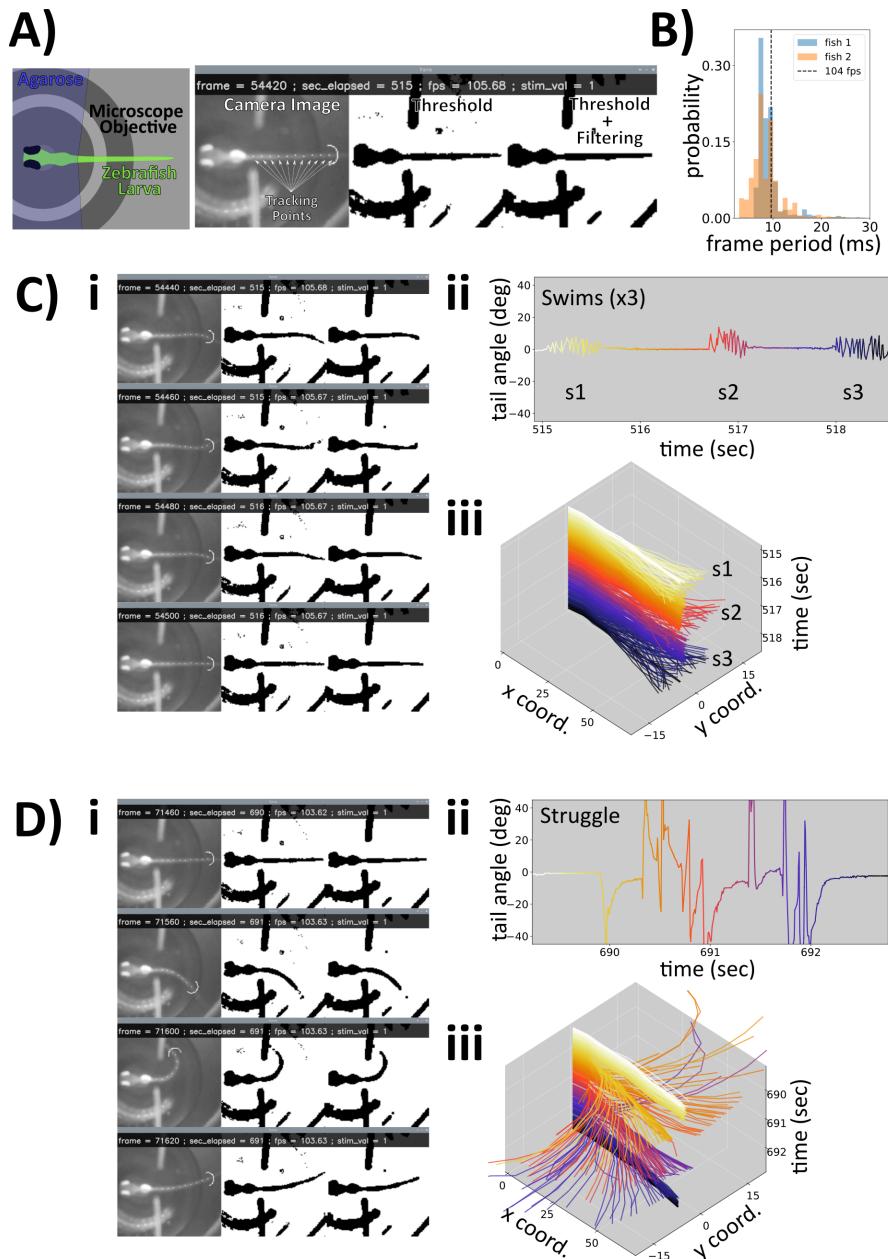
101 I would note that I built up the size of the platform of the mount to match with the relatively spacious config-  
102 uration of the microscope I was using ([Figure 1D](#)). A much more compact configuration is possible, since we only  
103 require ≈26 mm of clearance from the fish to the bottom of the ≈6 mm thick camera. The base design could be  
104 easily adapted to match different microscope stage configurations. For example the entire system could be inverted  
105 to accommodate an inverted microscope to image ventral structures during behaviour, such as the the lateral line  
106 ganglia and the heart. Or, if stimuli need to be bottom-projected, a small 45-degree hot mirror could be used to  
107 divert the image to the camera and free the space directly beneath the animal for stimuli.

## 108 Software and tail tracking strategy

109 Tracking objects in images and videos has undergone a revolution with deep learning and neural network frame-  
110 works, where the tracking and reconstruction of complex animal postures is possible after training networks on only  
111 a few example images ([Mathis et al., 2018; Pereira et al., 2022](#)). However, such approaches are computationally  
112 intensive and generally require dedicated and GPU hardware beyond the capabilities of the standard Raspberry Pi,  
113 making them incompatible with our project goals. In contexts where the image background is predictable and stable,  
114 classical computer vision methods like background subtraction, filtering and thresholding may still be preferable to  
115 network-based object identification, especially when speed or computational resources are priorities ([Mirat et al.,  
116 2013; Štih et al., 2019; Zhu et al., 2023](#)). Here I have used the *Numpy* ([Harris et al., 2020](#)) and *OpenCV* ([Bradski, 2000](#))  
117 libraries to handle the image data and computer vision tasks ([Figure 1](#)).

118 Image frames are acquired directly from the camera buffer as an 8-bit Numpy array, and thresholded using  
119 Adaptive Thresholding (`cv2.adaptiveThreshold`) to identify bright objects in the image ([Figure 2A](#), "Threshold"). This  
120 binary image is then filtered using a morphological Opening and Closing operation (`cv2.morphologyEx`). This combi-  
121 nation generally results in a nicely segmented fish blob in the final binary image ([Figure 2A](#), "Threshold + Filtering").  
122 However, this method identifies all large bright objects in the image, including borders of the agarose block and  
123 reflections on the microscope objective, and therefore we need a method to identify the fish object among these  
124 various segmented blobs.

125 The fish object is identified with a pre-defined coordinate that acts as the first tracking point of the fish. The fish  
126 object is then skeletonized into up to 10 tail segements ([Figure 2A](#), 'Tracking Pts'), which can be used to reconstruct  
127 the posture of the tail to identify swimming events ([Figure 2C,D](#)). To do this skeletonization, the tracking points are  
128 iteratively identified based on the intersection of a semicircle and the fish object, offset 7 pixels (0.19mm) from the  
129 previous tracking point, and oriented in the direction of the previous segment (similar to [Štih et al. \(2019\); Randlett  
130 et al. \(2019\)](#)). For the first search, this direction is toward the right of the image. Therefore, this strategy relies on  
131 the zebrafish larvae being oriented with its tail pointed towards the right, and being placed in the same position



**Figure 2. Larval zebrafish trail tracking examples.**

**A)** Screenshot of a single frame of a tracking image, showing the image from the camera ("Camera Image") with the resultant tracking points overlaid as white dots. The final tracking point is shown as a white semicircle, which is used in the coordinate search algorithm. "Threshold" shows the result of the Adaptive Thresholding operation, and "Threshold + Filtering" the result of the morphological Opening and Closing operations. Displayed along the top are the: frame (current frame number of the experiment), sec\_elapsed (number of seconds elapsed in the experiment), fps (current frame rate, frames per second), stim\_val (the current value read on the stimulus recording pin: GPIO Pin 4). A schematic of the image field, depicting the agarose mounting medium, the position of the zebrafish, and the microscope objective visible in the background is shown in the left panel.

**B)** Probability density distribution of individual frame periods from two representative experiments.

**C)** i) Example frames during a swimming event. ii) Tail angle deflections during 3 distinct swim events. iii) 3D plot of tail coordinates through the same time period as (ii), drawn in the same time color code.

**D)** Same as (C), but for a period in which the larvae executes a struggle/escape maneuver and associated high amplitude tail deflections.

**Figure 2—video 1.** Screen recording of the tail tracking example, [download](#)

132 such that the exit point of the tail from the agarose block intersects with the first tracking point. It also requires that  
133 no other bright objects intersect with the fish object after binarization. Therefore, it is critical to avoid distracting  
134 objects in the imaging scene, such as scratches in the dish or stray pieces of agarose.

135 This computationally lean segmentation and skeletonization strategy takes less than 10 ms on the Raspberry  
136 Pi CPU. The imaging frame rate when using the *picamera* python package will adjust based on the throughput of  
137 the analysis system, which can change with the complexity of the binary images that are processed or external  
138 CPU demands, but runs at approximately 104 fps (*Figure 2B*). This is sufficient to clearly distinguish different types  
139 of movement events, such as "swims" from "struggles" *Figure 2C* vs. D), and where individual tail beats during  
140 swimming events are resolvable. However, this will not be true during rapid/burst swimming, in which tail-beat  
141 frequency will exceed our frame rate. If such temporal resolution is required our setup will be insufficient, and we  
142 will only reliably track tail half-beat frequencies of  $\leq 50\text{hz}$ . Therefore, this system is not capable of comprehensive  
143 behavioural characterization, but can be used to identify different types of swim events.

144 During the experiment the software provides a visual display, as is shown in the screenshots in (*Figure 2*), and  
145 screen capture video (*Figure 2-video 1*). Results of the thresholding, filtering, and skeleton tracking are visible  
146 and updated in real-time. This can be used to optimize the position of the zebrafish, the Adaptive Thresholding  
147 parameters (neighborhood, threshold) using the 'w/a/s/d' keys, and the position of the first tracking point using the  
148 arrow keys.

#### 149 **Output data format**

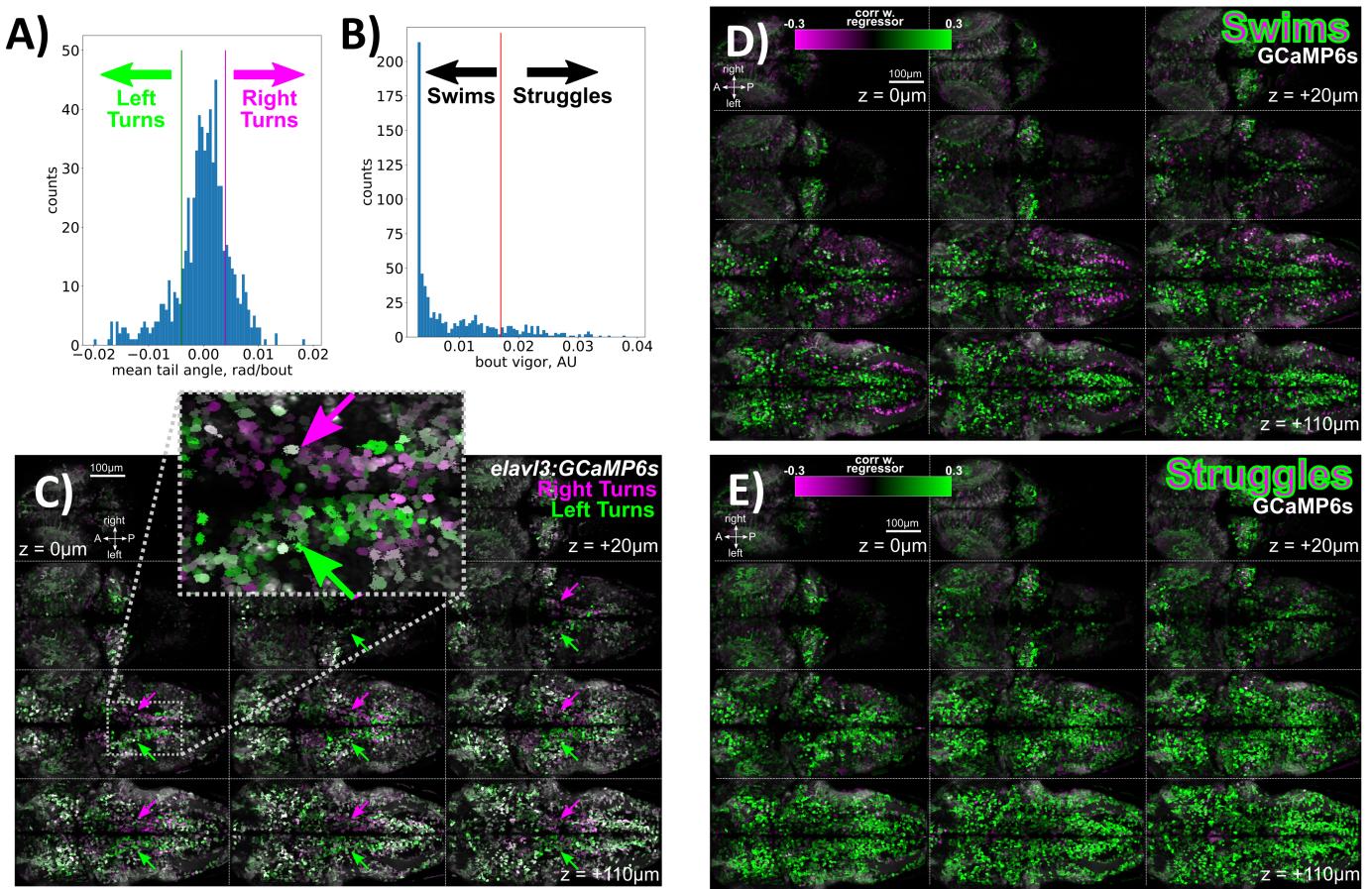
150 The tail tracking data are saved in a comma-separated text file '*\*\_coords.txt*', the 10 pairs of "X" and "Y" coordinates  
151 for each tail point are saved as rows, and thus there are two rows with 10 columns for every tracked frame. 'NaN'  
152 values represent instances where a tail point is not identified.

153 The timing of the data is saved in a separate text file '*\*\_tstamps.txt*', which also has two rows for each frame. The  
154 first value is the "timestamp" reflecting the time elapsed since the beginning of the tracking experiment. This is  
155 used to relate the tail posture and behavioural events to specific points in time. This is important for experiments  
156 in which precise timing of behavioural events is important, because the frame rate is not fixed and can fluctuate  
157 during the experiment (see above). However, it is important to note that the timestamp recorded is based on the  
158 time at which the frame is received from the camera buffer, which may lag from the time at which it was actually  
159 acquired by the camera. This could be problematic if millisecond-level precision on behavioural timing is critical,  
160 for example if differentiating between Short- and Long-Latency acoustic stimulus responses (*Burgess and Granato,*  
161 *2007*).

162 The second value in the '*\*\_tstamps.txt*' file is the value recorded on one of the GPIO pin 4 of the Raspberry Pi.  
163 This value will read either "low"=0 for a voltage less than 1.8V, or "high"=1 for 1.8-3.3V. I use these recordings to  
164 synchronize the behavioural recordings with the frames recorded on the microscope. In our typical setup we are  
165 using an analog output pin from the DAQ board on the microscope to control the red stimulus LEDs (*Figure 1B*), and  
166 we also connect this output of the DAQ board to GPIO pin 4 on the microscope. In this way, we can synchronize the  
167 stimuli, microscope imaging frames, and the behavioural recordings.

168 These datasets can be read into python for analysis using, for example:

```
169     1 import numpy as np
170
171     2
172     3 # load tracking coordinates
173     4 data = np.loadtxt('*_coords.txt', delimiter=',')
174
175     5
176     6 # separate 'x' and 'y' coordinates of tracking points
177     7 x_coords = data[:, :]
178     8 y_coords = data[2::2, :]
179
180     9
181    10 # load timing data
182    11 t_stamps = np.loadtxt('*_tstamps.txt', delimiter=',')
183
184    12
185    13 # separate 'timestamps' and 'stimulus state' recordings
186    14 time = t_stamps[:, 0]
187    15 stim = t_stamps[:, 1]
```



**Figure 3. Identification of behaviour-associated neurons in a larval zebrafish brain via 2-photon  $\text{Ca}^{2+}$  imaging.**

**A)** Histogram for the mean tail angle during individual movement bouts for a single larva over an 80 minute imaging session. Bouts are classified as left or right turns based on a threshold value of 0.004 radians/bout.

**B)** Histogram for the bout vigor, quantified using a rolling standard deviation of absolute tail angle. Movements are classified as "swims" or "struggles" based on a threshold value of 0.017 (AU: arbitrary units).

**C)** Tuning of  $\text{Ca}^{2+}$  traces in ROIs to turns to the left (green) or right (magenta), as classified in (A). Images are the Pearson correlation coefficient to each behavioral regressor (left or right turns), scaled from 0.0 to 0.3. *Tg2(elavl3:GCaMP6s)* expression pattern is shown in grey. Arrows highlight the Anterior Rhombencephalic Turning Region (ARTR): with ipsilateral tuning to turning direction. A = Anterior, P = Posterior

**D, E)** Tuning of neurons to swims (**D**), and struggles (**E**), as classified in (B).

#### 184 Behavioural analysis of $\text{Ca}^{2+}$ imaging data

185 To test the performance of the *pi\_tailtrack* system, I analyzed  $\text{Ca}^{2+}$  imaging data from an 80 minute-long volumetric  
 186 recording covering a large proportion of the brain (as in *Lamiré et al. (2022)*). To identify neurons tuned to be-  
 187 havioural parameters I used "regressors" derived from the *pi\_tailtrack* recordings reflecting different motor states  
 188 convolved with the GCaMP response kernel (as in *Miri et al. (2011)*). Zebrafish swim bouts can be classified as either  
 189 forward swims or turns, and an area within the anterior hindbrain is associated with turning direction. This area is  
 190 known as the Anterior Rhombencephalic Turning Region (ARTR: *Dunn et al. (2016)*, also called the HBO: *Ahrens et al.*  
 191 *(2013); Wolf et al. (2017)*), and shows a conspicuous activity pattern with stripes of neurons tuned to the ipsilateral  
 192 turning direction. By looking at correlations to regressors reflecting right and left turns, I identified these stripes of  
 193 neurons in the ARTR-region, indicating that I can successfully identify the ARTR using *pi\_tailtrack* (**Figure 3A,C**). A simi-  
 194 lar analysis looking at "swims" vs "struggles", with "struggles" reflecting high-amplitude tail flicking events (**Figure 2D**,  
 195 **Figure 3B**), identified differential neuronal activation in the context of these two movement categories (**Figure 3D,E**),  
 196 with the presence of lateral hindbrain populations of neurons that were negatively correlated with "swims", and a  
 197 broader and more positively correlated population with "struggles".

198 **Future developments**

199 Here I have used the *pi\_tailtrack* system to simply record the behaviour of the animal independent of the microscopy  
200 or any stimulus delivery. Therefore, the timing of microscope image acquisition is controlled by the microscope  
201 computer and is independent of *pi\_tailtrack*. These separate experimental clocks (microscope frames vs Pi Camera  
202 frames) must be synchronized, and in our case I have used the GPIO input pin on the Raspberry Pi to record the  
203 timing of the stimuli delivered by the microscope relative to the Pi Camera frames. An alternative solution would  
204 be to use the Raspberry Pi to deliver the stimuli, perhaps by integrating a video projector system to allow for the  
205 delivery of arbitrary and complex visual stimuli. This would also open up possibilities for performing "virtual reality"  
206 experiments, where the behaviour of the animal dictates the stimulus in closed-loop. In some microscope systems  
207 it should also be possible to use the Raspberry Pi GPIO to trigger microscope acquisitions. This may be preferable  
208 if the synchronization between imaging and behaviour frames is critical.

209 It is also important to note that hardware in this micro-computer/Raspberry Pi space is rapidly evolving. Indeed,  
210 a new suite of Raspberry Pi V3 Cameras have just been released, offering increased resolution, dynamic range, and  
211 frame rate. Using these cameras, we may be able to increase the frame rate of tracking into the multiple-hundreds of  
212 hz, which would allow us to more reliably resolve individual tail half-beats. The Raspberry Pi "Global Shutter" Camera  
213 has also recently been released, which is likely also going to be very interesting for behavioural neuroscience, as  
214 the use of a global shutter avoids rolling shutter artifacts that distort images along the frame during rapid motion.  
215 The software introduced here could be further optimized for speed/framerate, for example by moving to a multi-  
216 threaded architecture to distribute the image acquisition and tracking computations (**Zhu et al., 2023; Randlett et al.,**  
217 **2019**), using a compiled language (e.g. C/C++ or Julia), or perhaps by moving image processing onto the Raspberry  
218 Pi GPU.

219 **Conclusion**

220 Here I described our system for tracking the tail of the larval zebrafish during microscopy. Many of the practical con-  
221 siderations of this setup may be specific to our application, and therefore may need modification for use in other  
222 experiments in other labs. However, I feel that the core and simple idea of using an IR-sensitive Raspberry Pi Cam-  
223 era, a simple Python script, coupled with IR LEDs and and IR filter, provides an approachable and flexible solution  
224 that may be widely useful for observing and tracking the behaviour of zebrafish (or perhaps other animals) while  
225 performing imaging experiments. This system's attributes may also make it an ideal tool for community engage-  
226 ment activities such as school outreach programs. It could serve as a platform for learning about microelectronics,  
227 behavioural analyses, machine vision, and hardware design and construction.

228 **Methods**

229 **Animal Ethics Statement**

230 Adult zebrafish used to generate larvae were housed in accordance with PRCI facility approved by the animal wel-  
231 fare committee (comité d'éthique en expérimentation animale de la Région Rhône-Alpes: CECCAPP, Agreement #  
232 C693870602). Behaviour and microscopy experiments were performed at the 5dpf stage, and are thus not subject  
233 to ethical review, but these procedures do not harm the larvae.

234 **Animals**

235 All experiments were performed on larval zebrafish at 5 days post fertilization (dpf), raised at a density of ≈1 lar-  
236 vae/ml of E3 media in a 14:10h light/dark cycle at 28-29°C. Adult zebrafish were housed, cared for, and bred at the  
237 Lyon PRECI zebrafish facility. *mitfa*/Nacre mutant animals (ZDB-ALT-990423-22) were used to prevent pigmentation.

238 Larval zebrafish were mounted and head restrained for 2-photon imaging and behavioural analysis by placing  
239 them in a very small drop of E3 in the lid of a 35mm petri dish (Greiner bio-one, 627102). Molten (≈42°C) 2% low  
240 melting point agarose (Sigma A9414) in E3 Medium was added to the dish in an approximately 10mm-diameter  
241 droplet around the fish, and the zebrafish was repositioned within the solidifying agarose using a gel-loading pipette  
242 tip, such that it was oriented symmetrically for imaging with the dorsal surface of the head at the surface of the  
243 agarose. After the agarose had solidified (≈10 minutes), E3 was added to the dish, and then the agarose around the  
244 tail was cut away. This was done using a scalpel in two strokes emanating laterally from just below the swim bladder

**Table 1.** Bill of Materials

Component	Manufacturer	Cat. Number	≈Price (€)	Supplier/Link
Raspberry Pi Computer	Raspberry Pi Found.	4B Rev 1.4 8gb	95	kubii
Pi NoIR Camera	Raspberry Pi Found.	NoIR v2.1	26	kubii
24" Pi Camera Cable	Samtec	FJ-15-D-24.00-4	18	farnell
880nm IR Bandpass filter	Edmund Optics	65-122	177	Edmund Optics
890nm LEDs	Vishay Semiconductor	TSHF5410	$0.35 \times 10 = 4$	RS Components
18V DC power supply <sup>1</sup>	generic, for IR LEDs	min ≈200mA	15	e.g. amazon.fr
Current Limiting Resistor <sup>1</sup>	generic, for IR LEDs	minimum 1W power	1	e.g. amazon
3D printed parts <sup>2</sup>	Black/opaque, generic	PETG <sup>3</sup>	1	github:pi_tailtrack
M3 screws and nuts	generic	to secure camera	1	
Computer screen, keyboard, mouse	generic	recycle/borrow/steal		

**Total: ≈338€**

<sup>1</sup>These particular specs are not required, but the power supply and resistor must be matched appropriately. See [amplifiedparts.com: LED Parallel/Series Calculator](#)

<sup>2</sup>Parts were printed on an Ender 3 Pro 3D printer: Price ≈200€.

<sup>3</sup>This is the material I had on hand, but likely anything will work (PLA, ABS, Resin, etc)

<sup>245</sup> (illustrated in *Figure 3A*). It is critical to not scratch the dish in the vicinity of the freed tail, which can interfere with  
<sup>246</sup> tail-tracking.

### <sup>247</sup> Tail Tracking Software

<sup>248</sup> Software was written in Python, using the *picamera* library for camera control (*Raspberry Pi Foundation, 2023*). Trail  
<sup>249</sup> tracking was performed using *OpenCV* (cv2 version 4.5.5) (*Bradski, 2000*), and *Numpy* (version 1.19.5) (*Harris et al.,  
250 2020*). All code is provided in the file [record\\_tail.py](#). The main method for tail extraction was Adaptive Thresholding  
<sup>251</sup> (cv2.adaptiveThreshold), using a threshold of -10 and a 33 pixel neighborhood. These parameters can be adjusted  
<sup>252</sup> in real-time using the w/s and a/d keys. The starting coordinate for the tail tracking can be adjusted using the arrow  
<sup>253</sup> keys.

<sup>254</sup> Code for generating the figure panels in *Figure 2* can be found in: [plot\\_tail.ipynb](#). Datasets are available here:  
<sup>255</sup> [pi\\_tailtrack datasets](#).

### <sup>256</sup> Hardware

<sup>257</sup> I used a Raspberry Pi 4 Model B Rev 1.4 computer, running Raspbian GNU/Linux 11 (bullseye). **Table 1** contains  
<sup>258</sup> the details of the hardware components that I used, their approximate price, and an option for supplier (keeping in  
<sup>259</sup> mind that these later two are subject to change and will rapidly become inaccurate).

### <sup>260</sup> Ca<sup>2+</sup> imaging and analysis

<sup>261</sup> 2-photon Ca<sup>2+</sup> imaging was performed and analyzed as described in (*Lamiré et al., 2022*). Briefly, a 5dpf *Tg2(elavl3:GCaMP6s)*  
<sup>262</sup> (ZDB-ALT-180502-2, *Dunn et al. (2016)*) larva was imaged using a 20x 1.0NA water dipping objective (Olympus) on a  
<sup>263</sup> Bruker Ultima microscope at the CIQLE imaging platform (Lyon, LYMIC). Frames were acquired using a resonant scanner  
<sup>264</sup> over a rectangular region of 1024×512 pixels (0.6μm x/y resolution) and piezo objective to scan 12 planes sepa-  
<sup>265</sup> rated at 10μm steps, with a repeat rate of 1.98 hz. The position of the functional imaging stack within the brain was  
<sup>266</sup> stabilized in x/y/z "online" during acquisition using Bruker's PrairieLink API and Python ([brukerPL\\_stable\\_tseries.py](#)).  
<sup>267</sup> The central imaging plane was compared to a high-quality anatomical stack acquired before functional imaging  
<sup>268</sup> using the *registration.phase\_cross\_correlation* function from the *scikit-image* package (*Van der Walt et al., 2014*).

<sup>269</sup> ROIs were identified and fluorescence timeseries extracted using suite2p (*Pachitariu et al., 2016*). The zebrafish  
<sup>270</sup> was stimulated with 60 "dark flash" stimuli at 60 second ISI (*Lamiré et al., 2022*), though responses to these stimuli  
<sup>271</sup> were not incorporated into the analyses presented here, other than to synchronize the behavioural tracking with  
<sup>272</sup> the microscope acquisition timing.

273 To identify neurons tuned to turning direction (*Figure 3C*), the fluorescence trace from each ROI was compared to  
274 vectors derived from the *pi\_tailtrack* recordings of the tail reflecting leftward or rightward turns, respectively. These  
275 "behaviour state" vectors were convolved with the GCaMP response kernel to generate "regressors" reflecting the  
276 predicted  $\text{Ca}^{2+}$  response in neurons that are activated during the relevant behavioural state (as in *Miri et al. (2011)*).  
277 Tuning images were then generated reflecting the Pearson correlation coefficient between the z-scored fluorescence  
278 trace of the ROI and the relevant regressor. Images output from the analysis were adjusted for brightness/contrast  
279 and LUT using FIJI/ImageJ (*Schindelin et al., 2012*). This same approach was used to identify the relationship between  
280 ROIs and "Swim" and "Struggle" motor events (*Figure 3D,E*). Code for generating the figure panels in *Figure 3* can  
281 be found in: [gcamp\\_corr\\_swimming.ipynb](#). Datasets are available here: [pi\\_tailtrack datasets](#).

## 282 Acknowledgements

283 This work was supported by funding from the ATIP-Avenir program of the CNRS and Inserm, a Fondation Fyssen  
284 research grant, and the IDEX-Impulsion initiative of the University of Lyon.

## 285 References

- 286 Ahrens MB, Li JM, Orger MB, Robson DN, Schier AF, Engert F, Portugues R. Brain-wide neuronal dynamics during motor adaptation  
287 in zebrafish. *Nature*. 2012 May; 485(7399):471–477. <https://doi.org/10.1038/nature11057>, doi: 10.1038/nature11057.
- 288 Ahrens MB, Orger MB, Robson DN, Li JM, Keller PJ. Whole-brain functional imaging at cellular resolution using light-sheet mi-  
289 croscopy. *Nat Methods*. 2013 May; 10(5):413–420.
- 290 Akerboom J, Chen TW, Wardill TJ, Tian L, Marvin JS, Mutlu S, Calderón NC, Esposti F, Borghuis BG, Sun XR, Gordus A, Orger MB,  
291 Portugues R, Engert F, Macklin JJ, Filosa A, Aggarwal A, Kerr RA, Takagi R, Kracun S, et al. Optimization of a GCaMP Calcium  
292 Indicator for Neural Activity Imaging. *The Journal of Neuroscience*. 2012 Oct; 32(40):13819–13840. <https://doi.org/10.1523/jneurosci.2601-12.2012>, doi: 10.1523/jneurosci.2601-12.2012.
- 294 Bradski G. The OpenCV Library. *Dr Dobb's Journal of Software Tools*. 2000; .
- 295 Broussard GJ, Kislin M, Jung C, Wang SSH. A flexible platform for monitoring cerebellum-dependent sensory associative learning.  
296 *J Vis Exp*. 2022 Jan; (179).
- 297 Budick SA, O'Malley DM. Locomotor repertoire of the larval zebrafish: swimming, turning and prey capture. *J Exp Biol*. 2000 Sep;  
298 203(Pt 17):2565–2579.
- 299 Burgess HA, Granato M. Sensorimotor Gating in Larval Zebrafish. *The Journal of Neuroscience*. 2007 May; 27(18):4984–4994.  
300 <https://doi.org/10.1523/jneurosci.0615-07.2007>, doi: 10.1523/jneurosci.0615-07.2007.
- 301 Chen TW, Wardill TJ, Sun Y, Pulver SR, Renninger SL, Baohan A, Schreiter ER, Kerr RA, Orger MB, Jayaraman V, Looger LL, Svoboda  
302 K, Kim DS. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*. 2013 Jul; 499(7458):295–300. <https://doi.org/10.1038/nature12354>, doi: 10.1038/nature12354.
- 304 Dunn TW, Mu Y, Narayan S, Randlett O, Naumann EA, Yang CT, Schier AF, Freeman J, Engert F, Ahrens MB. Brain-wide mapping of  
305 neural activity controlling zebrafish exploratory locomotion. *Elife*. 2016 Mar; 5:e12741.
- 306 Elder R, A Guide to Recording 660FPS Video On A 6 dollar Raspberry Pi Camera; 2019. <https://blog.robertelder.org/recording-660-fps-on-raspberry-pi-camera/>, [Online; accessed 5-May-2023].
- 308 Geissmann Q, Garcia Rodriguez L, Beckwith EJ, French AS, Jamasb AR, Gilestro GF. Ethoscopes: An open platform for high-  
309 throughput ethomics. *PLoS Biol*. 2017 Oct; 15(10):e2003026.
- 310 Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R, Picus M,  
311 Hoyer S, van Kerkwijk MH, Brett M, Haldane A, del Rio JF, Wiebe M, Peterson P, Gérard-Marchant P, et al. Array programming with  
312 NumPy. *Nature*. 2020 Sep; 585(7825):357–362. <https://doi.org/10.1038/s41586-020-2649-2>, doi: 10.1038/s41586-020-2649-2.
- 313 Lamiré LA, Haesemeyer M, Engert F, Granato M, Randlett O. Functional and pharmacological analyses of visual habituation  
314 learning in larval zebrafish; 2022.
- 315 Maia Chagas A, Prieto-Godino LL, Arrenberg AB, Baden T. The €100 lab: A 3D-printable open-source platform for fluorescence  
316 microscopy, optogenetics, and accurate temperature control during behaviour of zebrafish, *Drosophila*, and *Caenorhabditis elegans*. *PLoS Biol*. 2017 Jul; 15(7):e2002702.

- 318 **Mathis A**, Mamidanna P, Cury KM, Abe T, Murthy VN, Mathis MW, Bethge M. DeepLabCut: markerless pose estimation of user-defined body parts with deep learning. *Nat Neurosci*. 2018 Sep; 21(9):1281–1289.
- 320 **Mirat O**, Sternberg JR, Severi KE, Wyart C. ZebraZoom: an automated program for high-throughput behavioral analysis and categorization. *Frontiers in Neural Circuits*. 2013; 7. <https://doi.org/10.3389/fncir.2013.00107>, doi: 10.3389/fncir.2013.00107.
- 322 **Miri A**, Daie K, Burdine RD, Aksay E, Tank DW. Regression-based identification of behavior-encoding neurons during large-scale optical imaging of neural activity at cellular resolution. *J Neurophysiol*. 2011 Feb; 105(2):964–980.
- 324 **Muller UK**. Swimming of larval zebrafish: ontogeny of body waves and implications for locomotory development. *J Exp Biol*. 2004 Feb; 207(5):853–868.
- 326 **Pachitariu M**, Stringer C, Dipoppa M, Schröder S, Rossi LF, Dalgleish H, Carandini M, Harris KD. Suite2p: beyond 10,000 neurons with standard two-photon microscopy; 2016.
- 328 **Pereira TD**, Tabris N, Matsliah A, Turner DM, Li J, Ravindranath S, Papadoyannis ES, Normand E, Deutsch DS, Wang ZY, McKenzie-Smith GC, Mitelut CC, Castro MD, D'Uva J, Kislin M, Sanes DH, Kocher SD, Wang SSH, Falkner AL, Shaevitz JW, et al. Publisher Correction: SLEAP: A deep learning system for multi-animal pose tracking. *Nat Methods*. 2022 May; 19(5):628.
- 331 **Portugues R**, Feierstein CE, Engert F, Orger MB. Whole-Brain Activity Maps Reveal Stereotyped, Distributed Networks for Visuomotor Behavior. *Neuron*. 2014 Mar; 81(6):1328–1343. <https://doi.org/10.1016/j.neuron.2014.01.019>, doi: 10.1016/j.neuron.2014.01.019.
- 334 **Randlett O**, Haesemeyer M, Forkin G, Shoenhard H, Schier AF, Engert F, Granato M. Distributed plasticity drives visual habituation learning in larval zebrafish. *Curr Biol*. 2019 Apr; 29(8):1337–1345.e4.
- 336 **Raspberry Pi Foundation**, picamera; 2023. <https://picamera.readthedocs.io/>, [Online; accessed 02-May-2023].
- 337 **Saunders JL**, Ott LA, Wehr M. AUTOPILOT: Automating experiments with lots of Raspberry Pis; 2019.
- 338 **Schindelin J**, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, Preibisch S, Rueden C, Saalfeld S, Schmid B, Tinevez JY, White DJ, Hartenstein V, Eliceiri K, Tomancak P, Cardona A. Fiji: an open-source platform for biological-image analysis. *Nat Methods*. 2012 Jun; 9(7):676–682.
- 341 **Severi KE**, Portugues R, Marques JC, O'Malley DM, Orger MB, Engert F. Neural control and modulation of swimming speed in the larval zebrafish. *Neuron*. 2014 Aug; 83(3):692–707.
- 343 **Štih V**, Petrucco L, Kist AM, Portugues R. Stytra: An open-source, integrated system for stimulation, tracking and closed-loop behavioral experiments. *PLoS Comput Biol*. 2019 Apr; 15(4):e1006699.
- 345 **Tadres D**, Louis M. PiVR: An affordable and versatile closed-loop platform to study unrestrained sensorimotor behavior. *PLoS Biol*. 2020 Jul; 18(7):e3000712.
- 347 **Van der Walt S**, Schönberger JL, Nunez-Iglesias J, Boulogne F, Warner JD, Yager N, Gouillart E, Yu T. scikit-image: image processing in Python. *PeerJ*. 2014; 2:e453.
- 349 **Wikipedia contributors**, Extension tube — Wikipedia, The Free Encyclopedia; 2022. [https://en.wikipedia.org/w/index.php?title=Extension\\_tube&oldid=1118116052](https://en.wikipedia.org/w/index.php?title=Extension_tube&oldid=1118116052), [Online; accessed 28-April-2023].
- 351 **Wolf S**, Dubreuil AM, Bertoni T, Böhm UL, Bormuth V, Candelier R, Karpenko S, Hildebrand DGC, Bianco IH, Monasson R, Debrégeas G. Sensorimotor computation underlying phototaxis in zebrafish. *Nat Commun*. 2017 Sep; 8(1).
- 353 **Zhu Y**, Auer F, Gelhaw H, Davis SN, Hamling KR, May CE, Ahamed H, Ringstad N, Nagel KI, Schoppik D. SAMPL is a high-throughput solution to study unconstrained vertical behavior in small animals. *Cell Reports*. 2023 Jun; 42(6):112573. <https://doi.org/10.1016/j.celrep.2023.112573>, doi: 10.1016/j.celrep.2023.112573.