



## Recording and Quantifying *C. elegans* Behavior

Navin Pokala and Steven W. Flavell

### Abstract

Studies of *C. elegans* behavior have been crucial in identifying genetic pathways that control nervous system development and function, as well as basic principles of neural circuit function. Modern analysis of *C. elegans* behavior commonly relies on video recordings of animals, followed by automated image analysis and behavior quantification. Here, we describe two methods for recording and quantifying *C. elegans* behavior: a single-worm tracking approach that provides high-resolution behavioral data for individual animals and a multi-worm tracking approach that allows for quantification of the behavior of many animals in parallel. These approaches should be useful to a wide range of researchers studying the nervous system and behavior of *C. elegans*.

**Key words** *C. elegans*, Behavior, Tracking software, Behavioral states, Quantitative behavioral analysis

---

### 1 Introduction

The quantitative analysis of animal behavior is a widely used approach in neuroethology, neuroscience, and neurogenetics [1]. For decades, researchers have quantified the behavior of small invertebrate animals to gain insights into basic principles of nervous system development and function. For example, genetic analysis of circadian rhythms in *D. melanogaster* revealed evolutionarily ancient mechanisms that underlie the molecular clock [2]. Behavioral studies of *C. elegans* date back to the 1970s [3, 4] and have been extended and refined over time. Quantitative analysis of *C. elegans* behavior has been a “workhorse” method for genetic studies [5, 6], ethological studies [7, 8], and studies of neural circuits [9] and behaviors [10, 11]. Automated behavior tracking methods pioneered in *C. elegans* [12] have inspired the development of automated methods for other animals [13].

---

Navin Pokala and Steven W. Flavell contributed equally to this work.

We describe here two platforms for tracking and quantifying *C. elegans* behavior: a single-worm tracker and a multi-worm tracker. These methods contain the same basic elements—video recording, image analysis, and behavior quantification—but the protocols and resulting datasets differ considerably. The single-worm tracker uses the openAutoScope microscope platform [14] that builds upon previous single-worm tracking approaches [15, 16]. This method provides high-resolution information about *C. elegans* behavior, quantifying locomotion, posture, defecation, egg-laying, and pharyngeal pumping. Because this method allows for the simultaneous quantification of multiple distinct motor programs, it can be useful for studies of many different behavioral outputs. For example, this recently described approach enabled the identification of a dopaminergic circuit that couples locomotion and egg-laying as animals switch behavioral states [14].

The MultiWormTracker, an offshoot of the Parallel Worm Tracker program [17], permits recording and analysis of several animals simultaneously. The MultiWormTracker is best suited for quantifying locomotion parameters such as speed, reversals, and turns, and measuring how their distributions in populations evolve over time [9, 11]. It is highly flexible, and can function under a variety of different experimental conditions including uniform food lawns [18], finite food lawns [19, 20], off-food plates [21], chemotaxis [22], chemo- and optogenetics [23, 24], and microfluidics [25].

For both methods, we present required materials, standard recording protocols, and required data analysis procedures.

---

## 2 Materials

### 2.1 *C. elegans* Growth and Maintenance

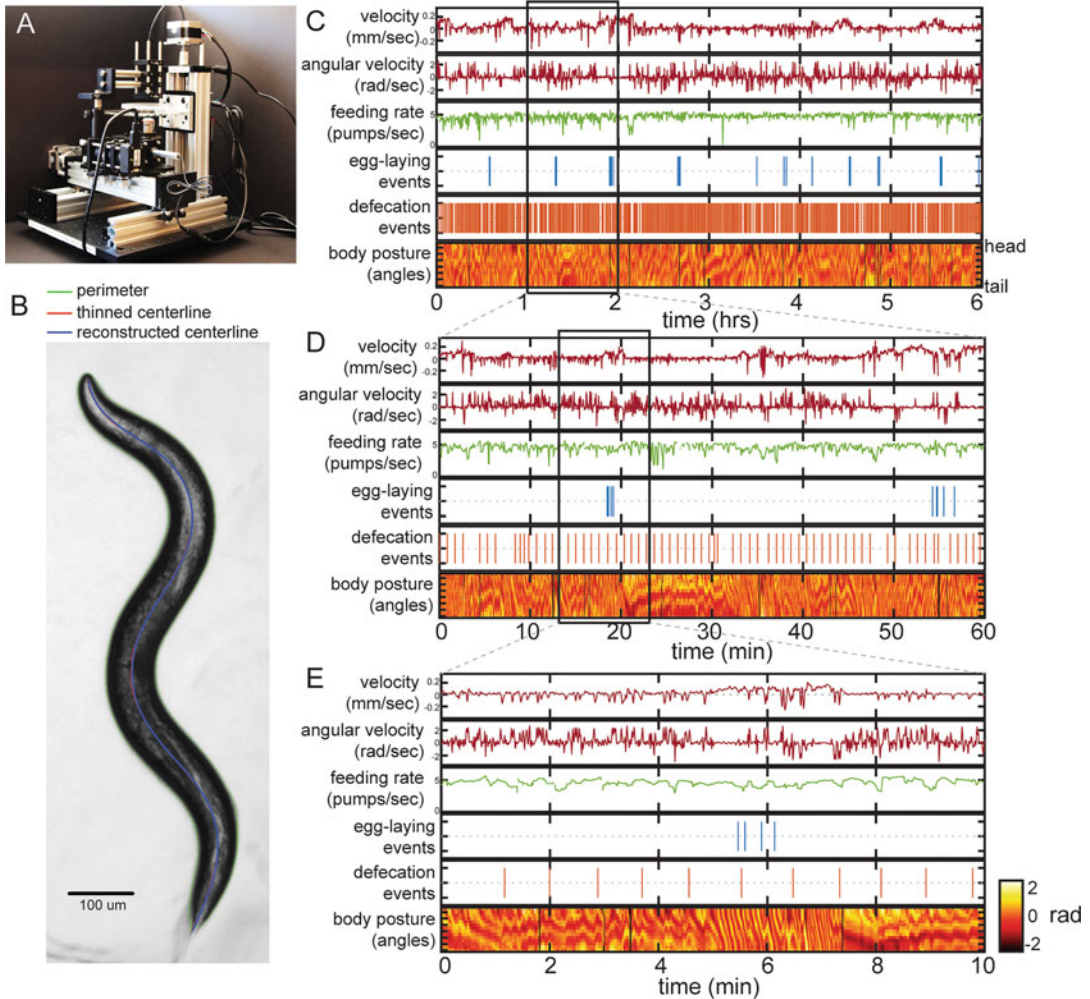
1. Nematode Growth Medium (NGM) plates: Mix 3 g NaCl, 2.5 g peptone, 17 g granulated agar, in 975 mL double-distilled water (deionized to 18 mega-ohm). Autoclave for 30 min and cool to 60 °C while stirring on a hot plate. Add 1 mL of 5 mg/mL cholesterol dissolved in ethanol. Add the following autoclaved salts and buffers in order: 1 mL 1 M CaCl<sub>2</sub>, 1 mL 1 M MgSO<sub>4</sub>, and 25 mL 1 M potassium phosphate pH 6.0. Pour into sterile petri dishes. Let dry overnight and store at 4 °C in plastic storage boxes for up to 6 months [26].
2. *Escherichia coli* OP50 suspension: OP50 bacteria can be obtained from the *Caenorhabditis* Genetics Center. Streak out OP50 from frozen glycerol stock onto a LB or NGM plate and grow overnight at 37 °C. Inoculate a single colony into one bottle of sterile LB (50–500 mL). Leave growing at room temperature without shaking for 2 days with cap slightly ajar

(though on top of bottle). Store at 4 °C for up to 2 months. Practice sterile technique during all steps of bacterial growth since there is no antibiotic resistance marker (*see Note 1*).

3. Copper chloride solution: 20 mM CuCl<sub>2</sub> in double-distilled water.
4. Filter paper: Whatman Grade 1 Qualitative Filter Paper.
5. Staged animals: Both tracking methods work best with young adults. Three days before behavioral recordings, the “staging” procedure to grow animals of a reliable age begins. Pick ~10–15 adult animals to a standard OP50 growth plate and permit them to lay eggs on the plate for 1 h. After the hour, remove all adults from the plate. Allow the plate to sit at room temperature for 72 h to obtain 1-day-old adult animals for recordings. Alternatively, pick L4 animals the day before the assay (*see Note 2*).
6. All-trans retinal: 50 mM all-trans-retinal in 100% ethanol. Store 1 μL aliquots at –80 °C.
7. OP50 bacteria concentrate for optogenetics: Grow 50 mL OP50 as described in **item 2**. Centrifuge the bacteria, and remove all but 10 mL of supernatant. Mix by vortexing. Store at 4 °C for up to 2 months.
8. Retinal feeding for optogenetics: The day before the optogenetics experiment, add 1 mL of OP50 bacteria concentrate to a freshly thawed 1 μL 50 mM all-trans-retinal aliquot (50 μM final retinal concentration). Immediately vortex and spread 200 μL on each unseeded NGM plate (*see Note 3*). Let the liquid absorb before adding animals. Transfer transgenic L4 animals to the plate. Pick a similar number of animals to a non-retinal plate. Store all of these plates in the dark overnight.

## 2.2 Single-Worm Tracking on the openAutoScope

1. Low-peptone NGM plates: Follow recipe for NGM plates in Subheading 2.1, **item 1**, but only add 0.2 g of peptone. All other components and aspects of preparation are identical. Use 10 cm petri dishes (*see Note 4*).
2. Computer hardware, software, and scripts: Use a computer running Windows 10. Download and install Arduino IDE (<https://www.arduino.cc/en/Main/Software>), teensyduino (<https://www.pjrc.com/teensy/teensyduino.html>), and R and R Studio (<https://rstudio.com/products/rstudio/>). Purchase and install National Instruments (NI) Labview (full 2017 or later), as well as NI Vision Acquisition Software and NI Vision Development Module. All custom software and scripts relevant to this protocol are available at <https://bitbucket.org/natecermak/openautoscope/src/master/>.



**Fig. 1** (a) Image of the openAutoScope tracking microscope. (b) Example image of a *C. elegans* animal from the tracking microscope. Green line denotes detected outline of worm; red line indicates worm’s centerline, obtained by thinning the thresholded image; blue line is the centerline as reconstructed from a spline-based 14-parameter representation. (c–e) Example dataset from the tracking microscope, showing the main *C. elegans* motor programs over 6 h (c), 1 h (d), or 10 min (e). (Images are re-printed with permission from Cermak et al., 2020)

3. Microscope: Construction of the openAutoScope consists of several steps. A build tutorial with step-by-step instructions, including detailed images and a full parts list, is available at <https://bitbucket.org/natecermak/openautoscope/src/master/>. Here, we describe the components that are assembled (Fig. 1a):

(a) Optical components: The optical path of the microscope consists of a  $4\times/0.1\text{NA}$  Olympus PLAN objective coupled to a 150 mm tube lens, resulting in  $3.33\times$

magnification. The image is projected onto a monochrome FLIR Chameleon 3 camera with a 0.5 inch sensor, yielding a  $1.84 \times 1.47$  mm field of view at  $1.4 \mu\text{m}/\text{pixel}$  resolution. A 780 nm, 18 mW infrared LED is used for illumination in a transillumination configuration.

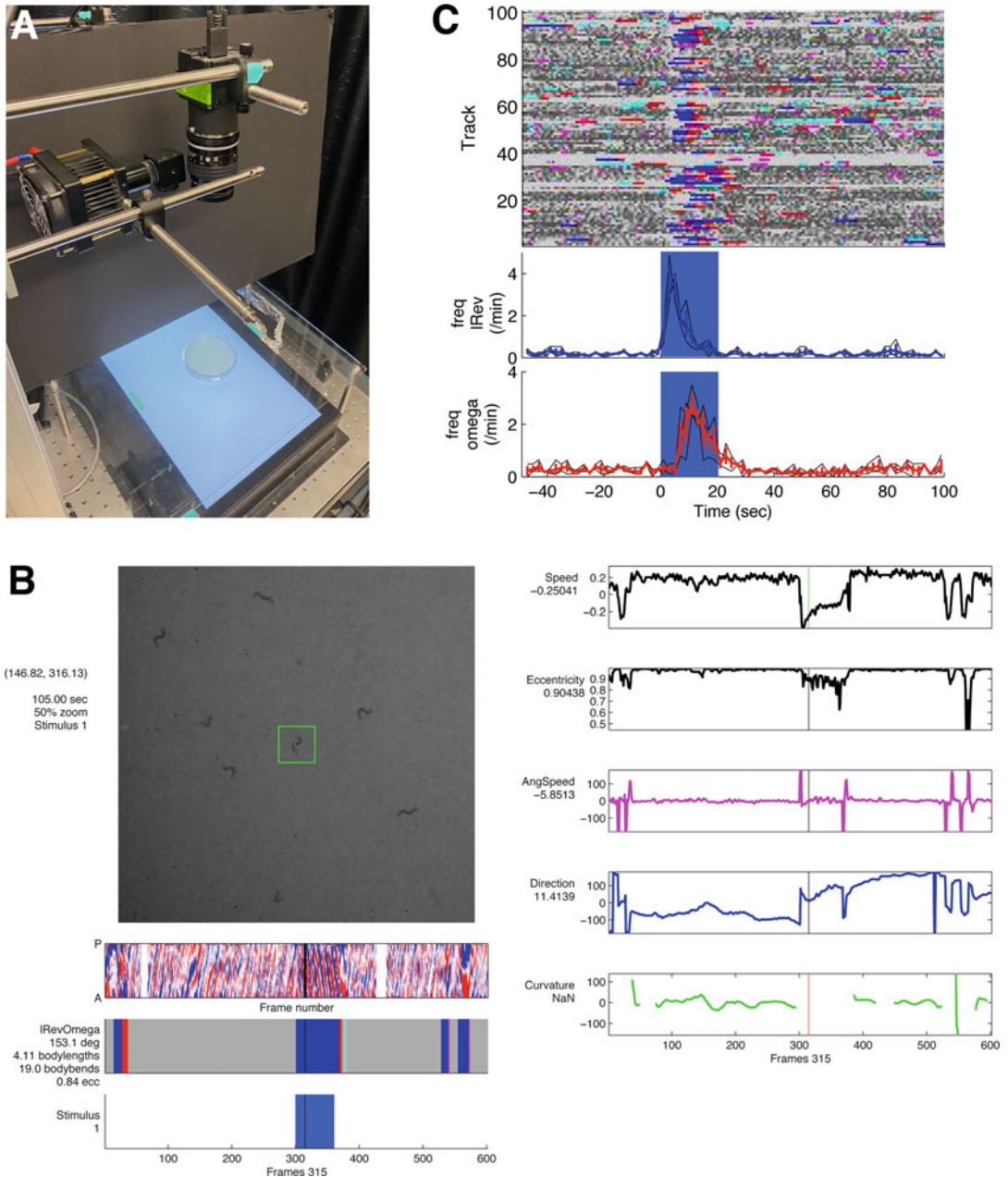
- (b) Laser for optogenetic stimulation: A 4.5 mW, 532 nm laser diode module is used for optogenetic stimulation. The laser is combined with the main optical path via a 550 nm dichroic and illuminates a 1.6 mm diameter region in the sample plane. Laser intensity is set to  $150 \mu\text{W}/\text{mm}^2$ .
- (c) Linear axes: Stage movement is controlled by motorized C-beam linear actuators with  $\sim 150$  mm of travel distance in each direction. The optical components of the microscope are mounted on the X and Y actuators so that the microscope moves (instead of the worm stage). The worm stage is mounted on the Z actuator. The stepper motors that move the actuators have a resolution of  $1.25 \mu\text{m}$  in each linear axis. The DRB8825 stepper motor controller ICs that control these motors are controlled with a Teensy 3.2 microcontroller.
- (d) Printed circuit board (PCB): The components of the microscope are connected on a PCB. The Teensy on the PCB is connected to a computer via a USB cable. The Teensy in turn controls the infrared LED, green laser for optogenetics, and the three linear actuators for microscope movement.
- (e) For computer control of the Teensy: Program the Teensy by opening the custom “scopeController.ino” file in the Arduino IDE.
- (f) Linking camera acquisition to microscope control for closed-loop tracking: Run the custom Labview code “scopeController.vi”.
- (g) Align the laser for optogenetic control and the LED light source. This should be done periodically though it is not required daily. For the laser, place a fluorescent slide on the stage. Turn the laser on in the Stage Control tab of the Labview software, set the display mode to Page 1 (Raw-Image), and adjust the right-angle mirror (using the knobs on it) until the brightest point is in the center of the field of view. After aligning the laser, turn off the laser and turn on the LED. To align the LED, manually move the overhead light source until the brightest spot is in the center of the field of view.

4. Data analysis: Open the custom R project `woRmtools.rproj` and install it locally by clicking Build/Install and restarting software. Open the custom `analyzeWorms.R` file, which has the full data analysis suite.

### **2.3 Multi-Worm Tracking**

1. Thin-lawn plates: Thin-lawn plates are prepared by flooding standard NGM plates with a saturated OP50 culture, removing excess liquid, and incubating at room temperature for 16 h.
2. Copper ring: Copper (II) chloride is a potent worm repellent and can be used to prevent animals from leaving an area. Cut a Whatman filter paper frame of the desired size. The filter paper frame size and shape can be customized. For a standard 60 mm plate, we use a 1.125 inch (28 mm) square punch to create a hole. Trim a ~0.25 cm wide frame around the hole. Using forceps, submerge the frame in a petri dish containing 20 mM  $\text{CuCl}_2$  for a few seconds. Remove the filter paper, and tap on the lid of the dish to remove non-absorbed drops of liquid. Gently place the frame on the assay plate agar. Use the forceps to gently flatten any folds, and ensure that the entire frame is flushed with the agar. Use within 1 h.
3. Worm pick: To transfer animals without food, flatten the ~5 mm of the tip of a platinum worm pick with needle-nose pliers.
4. Camera and recordings: We have used a FLIR Grasshopper 3 camera (FLIR GS3-U3-50S5M-C), controlled by Flycapture software from the manufacturer. We have also used a Pixelink PL-D7715 camera, controlled by Streampix software (Norpix). The camera can be attached to an 18 mm fixed focal length lens, or to a stereomicroscope equipped with a wide-angle lens (Fig. 2a). Video files recorded at 3 Hz and saved as 8-bit grayscale avi files with MPEG-4 video compression work well. The resolution should be 15–40 pixels/mm. If a copper ring is used, plates and zoom should be positioned and adjusted such that the inner edge of the ring is clearly visible. The tracking analysis sensors reversal and turn events that occur within 2.5 mm of the ring, and all behavior 10 s after a ring-induced event [27]. Before beginning actual recordings, adjust the zoom, focus, and lighting with a mock assay plate containing one or two animals.
5. Optogenetic stimulation: We use LED spotlights (for Chr2, 455 nm Mightex PLS-0455-030-S) driven by a computer-controlled power supply (Mightex SLC-SA02-US). The LEDs may be held by standard ring-stand clamps, or by dedicated optics hardware. Adjust the position and orientation of





**Fig. 2** (a) Image of a multi-worm tracking microscope. (b) WormPlayer GUI displaying part of a track of an animal expressing ChR2 in the AVA command neurons. These animals reverse continually during photostimulation, then make an omega turn. This animal is in the midst of an AVA:ChR2-elicited reversal. The green box indicates the selected worm, with blue, green, and red dots indicating the head, mid-body, and tail, respectively. The vertical black line in the strip-charts indicates the current frame. Displayed charts include a body posture kymograph, ethogram, photostimulation state, and instantaneous speed, angular speed, direction, and other parameters. (c) Ethograms and examples of stimulus-onset-averaged event frequency time-courses for animals expressing ChR2 in the ASH sensory neurons. These animals make a long reversal (dark blue) immediately followed by an omega turn (red). The frequency time-courses are averaged over three movies; the black lines indicate the average trace from each movie

the LED and slide the attached lens to ensure that the entire worm arena is illuminated. Using the control software, adjust the LED current to achieve the desired light intensity as measured by a laser power meter. Retinal-fed ChR2-expressing animals respond robustly to a 455 nm LED projecting at  $25 \mu\text{W}/\text{mm}^2$ , while non-retinal-fed control animals show no response. To reduce excitation glare in recordings, the lens can be inserted into a paper cup whose bottom is replaced with a glued filter sheet (with appropriate wavelength filter to prevent transmission of optogenetic light source). Alternatively, a glass filter can be directly fit onto the camera lens. To minimize photoactivation by the transmitted white light illumination, a filter sheet can be placed directly between the transmitted white light source and the plate. For ChR2 activation, we use Roscolux #312 filter sheets.

#### 6. MultiWormTracker Software.

- (a) The MultiWormTracker software runs in MATLAB (Mathworks) and requires the MATLAB Image Processing and Statistics toolboxes. While it works in newer versions (up to MATLAB 2020a), it was developed and tested most extensively using MATLAB 2011b on PCs running Microsoft Windows 10. The MultiWormTracker software can be downloaded from Github (<https://github.com/navinpokala/MultiWormTracker2020>).
- (b) Using Windows, copy the file `put_into_C_windows_system32\pdftk.exe` in the MultiWormTracker2020 directory to `C:\Windows\System32`.
- (c) Launch MATLAB and change to the MultiWormTracker2020 directory. Before you use the program for the first time, edit `MultiWormTracker_setup.m` line 8 where `MultiWormTracker_CODE_PATH` is defined to where you put the MultiWormTracker2020 source code directory.
- (d) Run `MultiWormTracker_setup` in MATLAB each time you want to use the code.
- (e) This will temporarily set the program path to the `MultiWormTracker_CODE_PATH` directory and temporarily remove any other non-core MATLAB paths. This is necessary to avoid conflicts between function names, etc. These path changes are NOT saved; you will need to run `MultiWormTracker_setup` each time unless you manually save the path.



### 3 Methods

The single-worm tracker and multi-worm tracker have different hardware, protocols, and analysis software, so they are described separately.

#### 3.1 *Single-Worm Tracking on the openAutoScope*

##### 3.1.1 *Preparation*

1. One day before recordings, seed the assay plates with OP50 bacteria. Take the desired number of 10 cm low-peptone NGM plates from the fridge, add 200  $\mu$ L OP50, and spread the bacteria into a roughly circular lawn whose edges are  $>1$  cm from the edges of the petri dish. Let plates sit at room temperature overnight. There should be a dry, thin lawn the next morning. (*see Note 5*).
2. On the day of the recording, transfer one animal to a seeded assay plate. Be careful not to transfer large amounts of bacteria with the animal. Move the plate onto the microscope stage face down. If long recordings are desired, remove the lid to prevent condensation. The plate should be placed on  $\frac{1}{4}$ " spacers in this case. Use tape to attach plate to the stage so that it does not move as a result of microscope movement.
3. Open LabView software and open the scopeController.vi program. Set the Control COM port to the appropriate number. Click the white arrow in the upper left to initiate the program. Go to the Saving tab and uncheck all boxes. Go to the Tracking tab and uncheck all of the indicators to turn the tracking function off. Under the Stage Control tab, turn the LED on and click Motors Enabled on. The field of view should now be visible, with the stage function on and closed-loop tracking off. No data will be saved under further specified (*see Note 6*).

##### 3.1.2 *Collect the Lawn Boundary Information*

1. Set the filename of the lawn boundary file. This file stores the coordinates of the lawn boundary for later analysis. The filename should be specified as a .csv file in the Saving tab under "file tracking path."
2. Under the Stage Control tab, use the Up/Down/Left/Right controls to move the stage to the lawn boundary. Under the Camera tab, adjust the exposure time so that the lawn edge is easily visible. Focus up/down if necessary.
3. Under the Saving tab, check the "saving tracking data" indicator next to "file tracking path" to begin saving. Then Stop and Start the LabView program one more time using the red stop and white arrow icons in the upper left (required for running the program free of bugs). Then use the cursor controls to move the field of view along the lawn perimeter. The objective is to keep the lawn boundary in the center of field of view and

go all the way around the perimeter. Progress can be visualized in the Page 6 tab “worm trajectory” panel.

4. After you have completed the circle around the perimeter, go to the Saving tab and uncheck the indicator next to “file tracking path.” This completes the lawn boundary file. From this point forwards, do not adjust the plate position or else the lawn boundary will need to be re-recorded.

### 3.1.3 *Collect Worm Data*

1. Rename all three file names with the names you would like to use for saving the worm data. The .avi file is not actually used. But the other file paths specify the directory for saving worm images and the file name for storing all of the tracking info (stage position, etc.).
2. Find the worm on the plate and use the cursors to get the animal in the field of view.
3. Once the animal is in view, go to the Tracking tab and check all 3 indicators, Process images, Tracking and Periodic autofocus, in the tab to initiate tracking. After this, the outline of the animal should be indicated and the microscope should automatically track the animal.
4. Adjust the exposure time again so that the outline of the worm’s perimeter is precisely correct, including the head. Typically, the ideal brightness results in the head appearing brighter than the tail.
5. If you are using optogenetics in your experiment, use the Laser Control tab to set the parameters of your stimulation protocol. Note that the stimulation protocol only begins when you manually click on “start simulation program,” so you may need to click on this when you start the video or at a later time if your stimulation should begin later in the video (the exact time points when the laser is on/off are logged, so the precise timing will be available during data analysis).
6. Stop/Start the Labview program again and then click “start timed recording” under the Saving tab with the duration of the recording set as desired. All files will be fully saved after the specified time is elapsed.

### 3.1.4 *Analyzing the Data*

1. The important files for data analysis are the lawnboundary.csv file, the trackingdata.csv file (which has stage movement information and more), and the worm images, which are stored as jpeg files in the folder that you specified.
2. In the analyzeWorms.R file, specify the paths of the lawn boundary file, worm tracking file, and the image directory.
3. Run the R code from top to bottom of the analyzeWorms script. The script is largely automated, but there are two points

that will require user input. First, after the majority of the analysis has completed, a matrix will appear in the plots window, showing the intensity of the worm's body along its centerline (as y-axis) over time (x-axis). It will ask whether the head was defined correctly. Since the head is brighter, you should answer yes if the bottom of the image is brighter and no if it isn't. If you enter no, it will invert the image and re-display, asking the same question again. Second, candidate egg-laying events need to be annotated by the user. During this stage, two consecutive images will appear in the plots window and the user will answer questions about whether an egg was laid during the time interval between the two images. First the user is asked to determine whether there is an actual egg-laying event until a true event is detected (enter "v" for a real event; enter "u" to proceed to next candidate event). This step is used to identify the ventral side of the body. After this, the script will display candidate egg-laying events on the ventral side of the body. The user is asked to input how many eggs were laid between these two consecutive frames. Enter 0 if there was no egg laid, or 1 (or higher) if eggs were laid.

4. The R code will save a CSV file at the end with the analyzed data. This file can be opened in any software for further analysis and inspection of behavioral data (Fig. 1b–e; see **Notes 7** and **8**). Time-dependent parameters in the CSV file include: body area and length; body posture (bodyangles and radius of curvature quantified from 14 evenly spaced segments along the anterior-posterior axis); head and mid-body coordinates; head and mid-body distances to food lawn edge; speed; behavior indicators (omega turn, defecation, egg-laying, pharyngeal pumping rate); optogenetic laser status; microscope autofocus status.

## 3.2 Multi-Worm Tracking

### 3.2.1 Preparation

1. The day before the experiment, pick L4 animals to a day-old thin-lawn plate. Each assay will require 20–30 animals. For optogenetic experiments, pick animals to retinal plates as described in Subheading 2.1, **item 8**, and pick a similar number of animals to a non-retinal plate.
2. Prepare thin-lawn food assay plates as described in Subheading 2.3, **item 1**, and incubate at room temperature overnight. Covering the entire plate eliminates lawn edge effects on exploration [19].
3. Food-free assay plates should be allowed to sit at room temperature overnight.
4. The day of the experiment, let food-free and seeded NGM assay plates dry for an hour, face down, tilted on their lids. Wipe off any condensation on the lids.

5. Prior to transferring animals to an assay plate, place a copper ring. Use within 1 h.
6. Using a mock assay plate containing a few worms, adjust the zoom, focus, and lighting of the recording arena.

### 3.2.2 Exploration Assay

On food lawns, *C. elegans* animals switch between bouts of roaming and dwelling [18]. When transferred to a food-free plate, these animals explore a local area by increasing their frequency of spontaneous reversals and turns for 10–20 min, then transition to a dispersal state with long forward runs with few reorientations [21].

1. Transfer 20–30 young adult animals to a thin-lawn food assay plate. Let the animals equilibrate for 30 min (*see* **Note 9**).
2. Gently place the plate under the camera and record for 10 min. These videos will allow quantification of on-food behavior.
3. Set the camera software to record for 1 h, so that minimal action (a single mouse click or keystroke) is required to trigger recording.
4. Place a copper ring on a food-free assay plate.
5. Start a timer for 2 min ( $t = 0$ ). Using a lump of food, transfer the animals from the food assay plate to a food-free plate (rinse plate). Flame the worm pick. Let the tip cool by touching it to the rinse plate agar. Scoop animals that have crawled away from the transfer drop point by sliding the flat pick between the animals and the agar. Quickly transfer the picked animals to the center of the food-free assay plate. Repeat the scoop and transfer until all the animals have been transferred to the assay plate, or when the 2 min are up ( $t = 2$  min).
6. Position the plate face down under the camera. At  $t = 3$  min, trigger recording.
7. Create four directories ('mystrain\_food', 'mystrain\_nofood', 'control\_food', 'control\_nofood'), and save the movies to the relevant directories.
8. Run MultiWormTracker({'mystrain\_food', 'mystrain\_nofood', 'control\_food', 'control\_nofood'}).

### 3.2.3 Multi-Worm Optogenetics

1. Place a copper ring frame on an assay plate relevant to the experiment, and transfer retinal-fed animals. If assaying animals off-food, transfer them as described in Subheading 3.2.2, **step 5**.
2. Depending on the experiment, you may wish to let the animals acclimatize to the plate prior to recording.
3. Wipe any condensation off the lid with a Kimwipe 10 min before recording.
4. Gently place the plate under the camera, with the lid facing up.

5. Start the recording and the photostimulation programs.
6. Create directories for the retinal-fed and non-retinal-fed controls, and save the movies to the relevant directories.
7. Launch MATLAB and change to the MultiWormTracker2020 directory. Run MultiWormTracker\_setup to set the program path. Go to the parent directory of the experiments.
8. Create a stimulus text file that describes when the light (blue, amber, green, or red) is on. An example is given in the README.txt file.
9. Process the movies as described below, but include the stimulus file as an argument. For example, MultiWormTracker({'retinal\_fed','controls'},'stimulus','my\_stimulus.txt'). The linked-Tracks and BinData files have fields that identify frames and times that are photostimulated. Plots that highlight behavior during stimulation times will be automatically generated as .pdf files. Peristimulus time histogram (psth) psth\_Tracks, psth. BinData, and pdf plot files with tracks aligned at the start of stimuli are also produced.

### 3.2.4 Tracking and Analysis

1. Launch MATLAB and change to the MultiWormTracker2020 directory. Run MultiWormTracker\_setup to set the program path. Go to the parent directory of the experiments.
2. To track a single movie that includes a standard 28 mm square copper ring, run MultiWormTracker('my\_movie.avi'). To track a directory of movies that all include a standard 28 mm square copper ring, run MultiWormTracker('my\_directory').
3. To track multiple directories of movies that all include a standard 28 mm square copper ring, run MultiWormTracker({'directory1','directory2','directory3'}).
4. Prior to starting the fully automated tracking, the program will automatically find the all rings and attempt to find 20–30 animals for each movie. If it cannot find a ring, or cannot find 20–30 animals, a graphical user interface (GUI) will pop up with instructions for the user to follow. The program will then launch the fully automated tracking processes.
5. If you do not use the standard 1.125 inch (28 mm) square copper ring, take a short 10 s movie of an object of known size, and save it as “scale\_movie.avi.” Include this as an additional “scale” argument when tracking. For example, MultiWormTracker('mymovie.avi','scale','my\_scale\_movie.avi'). A GUI will guide the user to define the length of the scale object contained in 'my\_scale\_movie.avi'.
6. If your framerate is NOT 3 Hz, include “framerate,” your\_framerate\_in\_Hz as an argument. For example

MultiWormTracker('mymovie.avi','scale','my\_scale\_movie.avi','framerate',5) would be used to analyze a movie recorded at 5 Hz.

7. If number of worms  $<20$  or  $>30$ , then you should include 'numworms', #worms as an argument. For example, MultiWormTracker('my\_movie.avi', 'numworms', 10) will analyze a movie with only 10 animals.
8. A number of files and plots are produced for each movie. The two most useful .mat files are the linkedTracks.mat (tracks of individual animals with descriptors for each frame including instantaneous speed, coordinates, body posture and shape, pirouette parameters such as reversal length and turning angle, and behavior state) and BinData.mat (averaged time-course data).
9. The README.txt file in the downloaded software directory describes the linkedTracks and BinData files in more detail. MATLAB functions useful for analyzing linkedTracks and BinData files are also described.
10. linkedTracks is a structure array with an element for each animal. Fields within linkedTracks are arrays whose elements describe the behavior of the animal in each frame. Fields in linkedTracks encode information such as: time and movie frame; X and Y coordinates for the animal in each frame; heading direction in each frame; instantaneous speed and angular speed; instantaneous path curvature; posture information including body eccentricity, body curvature, head and body angles; parameters including reversal length and change in direction due to a turn for each reorientation event; instantaneous locomotion state of an animal such as forward, pause, reversal, omega turn, and upsiion turn; presence or absence of a stimulus (i.e., optogenetic light) at each frame.
11. WormPlayer('mymovie.linkedTracks.mat') is a GUI-based viewer for examining the tracks of individual animals overlaid onto the movie. The animal in the current track is boxed, with dots indicating the head, center, and tail. The "T" button toggles the track, which is colored by behavioral state. The "Z" button is used to zoom. When the movie is played, indicators on strip-charts show instantaneous behavioral parameters of the animal such as its speed, angular speed, eccentricity, behavioral state, path curvature, and body posture kymograph (Fig. 2b).
12. BinData is a structure whose fields contain averaged behavior time-courses, along with standard deviation and standard error of the mean. Instantaneous parameters (speed, posture, fraction of animals in a given locomotion state) are averaged over 1 s, while event frequencies are averaged over 10 s. Fields in



BinData encode information such as time; forward speed; path curvature; posture (body eccentricity, head and body angles); frequencies of events such as pauses, reversals, and turns; fractions of animals in locomotion states such as forward, pause, reversal, and turn; locomotion state transition probabilities.

13. The freqs.txt and non\_freqs.txt text files contain the data from BinData.mat in a tab-delimited text format which can be opened by other programs for further analysis. The pdf files display plots for a selection of these averaged time-courses data (Fig. 2c).
14. If you analyze a directory, BinData.mat, freqs.txt, and non\_freqs.txt files will be created that contain weighted-average time-courses from all the movies in that directory, along with pdf files that display plots of these data.

---

## 4 Notes

1. We describe protocols here using OP50 bacteria, but many different strains and species of bacteria can be fed to *C. elegans*. This is a critical aspect of experimental design, since the food source of the animal profoundly impacts its behavior [28].
2. The developmental stage of the animal during the recording is important for two reasons. First, animals of different ages display different behaviors, so uniform staging is important for reproducible results. Second, both programs track animals based on their size, and default to young adults. This would need to be adjusted for younger animals that are smaller and more transparent. For single-worm tracking, in openAutoScope under Tracking tab, “criteria” would need to be adjusted. For multi-worm tracking, the minWorm and maxWorm parameters in define\_preferences.m would need to be adjusted.
3. Since 50  $\mu\text{M}$  retinal can retard bacterial growth, a concentrated bacteria slurry is used for feeding animals retinal.
4. The use of low-peptone plates is important to prevent the bacterial lawn from growing too thick. Thick bacterial lawns degrade tracking quality, since they can mask the worm and the worm can leave behind high-contrast tracks that can confuse the tracker.
5. When animals reach the perimeter of the plate, they will be lost by the tracker and the recording will be functionally over. To prevent this, it is helpful to leave  $>1$  cm between the lawn boundary and the plate perimeter. For recordings in the

absence of food, use filter paper with 20 mM CuCl<sub>2</sub> as is described in the multi-worm tracker protocol.

6. There is a green dot in the center of the field of view that indicates the target center position. Beware that if you click in the field of view, this will move this target (indicated by the green dot moving).
7. A small fraction of recordings can yield low-quality data. The main indicator of a low-quality recording is that the worm's posture was not identified in a large fraction of frames. For each recording, it is useful to check the number of "NA" entries in the bodyAngles data. If >10% of the entries are "NA," the video can be rejected on the basis of low-quality data.
8. The body angles in the saved .csv file are not adjusted for dorsal/ventral alignment. The source of this issue is that some animals lie on their left sides during movement and others lie on their right sides. The eggLoc variable can allow the user to align animals, though, since it indicates the body segment where the eggs were laid, and eggs are always laid on the ventral side. From the tip of the nose, segments 1–6 (arbitrarily numbered) are the left side of the animal as it appears on the screen, while the segments 7–12 are the right side. Thus, if one animal has its eggs laid along segments 3–4, but another has its eggs laid along segments 9–10, these animals were lying on opposite sides during recording. Body angles can then be inverted for select animals to then correct for dorsal/ventral alignment.
9. Transferring the worm by picking or other means can impact its behavior, typically stimulating it to move at increased speeds for 15–30 min afterwards. This should be taken into account when designing experimental protocols.

## References

1. Egnor SER, Branson K (2016) Computational analysis of behavior. *Annu Rev Neurosci* 39: 217–236. <https://doi.org/10.1146/annurev-neuro-070815-013845>
2. Konopka RJ, Benzer S (1971) Clock mutants of *Drosophila melanogaster*. *Proc Natl Acad Sci U S A* 68:2112–2116. <https://doi.org/10.1073/pnas.68.9.2112>
3. Brenner S (1974) The genetics of *Caenorhabditis elegans*. *Genetics* 77:71–94
4. Croll NA (1975) Components and patterns in the behaviour of the nematode *Caenorhabditis elegans*. *J Zool* 176:159–176. <https://doi.org/10.1111/j.1469-7998.1975.tb03191.x>
5. Bargmann CI, Hartwig E, Horvitz HR (1993) Odorant-selective genes and neurons mediate olfaction in *C. elegans*. *Cell* 74: 515–527. [https://doi.org/10.1016/0092-8674\(93\)80053-H](https://doi.org/10.1016/0092-8674(93)80053-H)
6. McGrath PT, Rockman MV, Zimmer M et al (2009) Quantitative mapping of a digenic behavioral trait implicates globin variation in *C. elegans* sensory behaviors. *Neuron* 61: 692–699. <https://doi.org/10.1016/j.neuron.2009.02.012>
7. Brown AEX, Yemini EI, Grundy LJ et al (2013) A dictionary of behavioral motifs reveals clusters of genes affecting *Caenorhabditis elegans* locomotion. *Proc Natl Acad Sci U S A* 110:791–796. <https://doi.org/10.1073/pnas.1211447110>

8. Stephens GJ, Johnson-Kerner B, Bialek W, Ryu WS (2008) Dimensionality and dynamics in the behavior of *C. elegans*. *PLoS Comput Biol* 4: e1000028. <https://doi.org/10.1371/journal.pcbi.1000028>
9. Gray JM, Hill JJ, Bargmann CI (2005) A circuit for navigation in *Caenorhabditis elegans*. *Proc Natl Acad Sci U S A* 102:3184–3191. <https://doi.org/10.1073/pnas.0409009101>
10. Swierczek NA, Giles AC, Rankin CH, Kerr RA (2011) High-throughput behavioral analysis in *C. elegans*. *Nat Methods* 8:592–598. <https://doi.org/10.1038/nmeth.1625>
11. Pierce-Shimomura JT, Morse TM, Lockery SR (1999) The fundamental role of pirouettes in *Caenorhabditis elegans* chemotaxis. *J Neurosci* 19:9557–9569. <https://doi.org/10.1523/JNEUROSCI.19-21-09557.1999>
12. Husson SJ (2012) Keeping track of worm trackers. *WormBook* 1–17. <https://doi.org/10.1895/wormbook.1.156.1>
13. Datta SR, Anderson DJ, Branson K et al (2019) Computational neuroethology: a call to action. *Neuron* 104:11–24. <https://doi.org/10.1016/j.neuron.2019.09.038>
14. Cermak N, Yu SK, Clark R et al (2020) Whole-organism behavioral profiling reveals a role for dopamine in state-dependent motor program coupling in *C. elegans*. *eLife* 9:e57093. <https://doi.org/10.7554/eLife.57093>
15. Faumont S, Lockery SR (2006) The awake behaving worm: simultaneous imaging of neuronal activity and behavior in intact animals at millimeter scale. *J Neurophysiol* 95: 1976–1981. <https://doi.org/10.1152/jn.01050.2005>
16. Yemini E, Jucikas T, Grundy LJ et al (2013) A database of *Caenorhabditis elegans* behavioral phenotypes. *Nat Methods* 10:877–879. <https://doi.org/10.1038/nmeth.2560>
17. Ramot D, Johnson BE, Berry TL et al (2008) The parallel worm tracker: a platform for measuring average speed and drug-induced paralysis in nematodes. *PLoS One* 3:e2208. <https://doi.org/10.1371/journal.pone.0002208>
18. Flavell SW, Pokala N, Macosko EZ et al (2013) Serotonin and the neuropeptide PDF initiate and extend opposing behavioral states in *C. elegans*. *Cell* 154:1023–1035. <https://doi.org/10.1016/j.cell.2013.08.001>
19. Calhoun AJ, Tong A, Pokala N et al (2015) Neural mechanisms for evaluating environmental variability in *Caenorhabditis elegans*. *Neuron* 86:428–441. <https://doi.org/10.1016/j.neuron.2015.03.026>
20. Rhoades JL, Nelson JC, Nwabudike I et al (2019) ASICs mediate food responses in an enteric serotonergic neuron that controls foraging behaviors. *Cell* 176:85–97.e14. <https://doi.org/10.1016/j.cell.2018.11.023>
21. López-Cruz A, Sordillo A, Pokala N et al (2019) Parallel multimodal circuits control an innate foraging behavior. *Neuron* 102: 407–419.e8. <https://doi.org/10.1016/j.neuron.2019.01.053>
22. Jin X, Pokala N, Bargmann CI (2016) Distinct circuits for the formation and retrieval of an imprinted olfactory memory. *Cell* 164: 632–643. <https://doi.org/10.1016/j.cell.2016.01.007>
23. Pokala N, Liu Q, Gordus A, Bargmann CI (2014) Inducible and titratable silencing of *Caenorhabditis elegans* neurons in vivo with histamine-gated chloride channels. *Proc Natl Acad Sci U S A* 111:2770–2775. <https://doi.org/10.1073/pnas.1400615111>
24. Gordus A, Pokala N, Levy S et al (2015) Feedback from network states generates variability in a probabilistic olfactory circuit. *Cell* 161: 215–227. <https://doi.org/10.1016/j.cell.2015.02.018>
25. Albrecht DR, Bargmann CI (2011) High-content behavioral analysis of *Caenorhabditis elegans* in precise spatiotemporal chemical environments. *Nat Methods* 8:599–605. <https://doi.org/10.1038/nmeth.1630>
26. Stiernagle T (2006) Maintenance of *C. elegans*. *WormBook*. <https://doi.org/10.1895/wormbook.1.101.1>
27. Hilliard MA, Bergamasco C, Arbucci S et al (2004) Worms taste bitter: ASH neurons, QUI-1, GPA-3 and ODR-3 mediate quinine avoidance in *Caenorhabditis elegans*. *EMBO J* 23:1101–1111. <https://doi.org/10.1038/sj.emboj.7600107>
28. Kim DH, Flavell SW (2020) Host-microbe interactions and the behavior of *Caenorhabditis elegans*. *J Neurogenet* 34(3–4):500–509. <https://doi.org/10.1080/01677063.2020.1802724>