RESEARCH ARTICLE

FIMTrack: An open source tracking and locomotion analysis software for small animals

Benjamin Risse[1☯], Dimitri Berh[2☯], Nils Otto[3], Christian Klämbt[3], Xiaoyi Jiang[2]*

1 School of Informatics, University of Edinburgh, Edinburgh, Scotland, United Kingdom, **2** Faculty of Mathematics and Computer Science, University of Münster, Münster, Germany, **3** Department of Behaviour and Neuronal Biology, University of Münster, Münster, Germany

☯ These authors contributed equally to this work.
* xjiang@uni-muenster.de

Abstract

Imaging and analyzing the locomotion behavior of small animals such as Drosophila larvae or C. elegans worms has become an integral subject of biological research. In the past we have introduced FIM, a novel imaging system feasible to extract high contrast images. This system in combination with the associated tracking software FIMTrack is already used by many groups all over the world. However, so far there has not been an in-depth discussion of the technical aspects. Here we elaborate on the implementation details of FIMTrack and give an in-depth explanation of the used algorithms. Among others, the software offers several tracking strategies to cover a wide range of different model organisms, locomotion types, and camera properties. Furthermore, the software facilitates stimuli-based analysis in combination with built-in manual tracking and correction functionalities. All features are integrated in an easy-to-use graphical user interface. To demonstrate the potential of FIMTrack we provide an evaluation of its accuracy using manually labeled data. The source code is available under the GNU GPLv3 at https://github.com/i-git/FIMTrack and pre-compiled binaries for Windows and Mac are available at http://fim.uni-muenster.de.

This is a *PLOS Computational Biology* Software paper.

Introduction

For most animals the ability to move is essential to survive. A complex nervous system, built-up by neurons and glial cells, allows sophisticated locomotion control. The analysis of locomotion of freely moving animals is crucial to gather insights into the nervous system functionality. Especially *Drosophila melanogaster* larvae and *Caenorhabditis elegans* worms are popular model organisms in neuro- and behavioral biology since sophisticated genetic tools and a well-

OPEN ACCESS

Citation: Risse B, Berh D, Otto N, Klämbt C, Jiang X (2017) FIMTrack: An open source tracking and locomotion analysis software for small animals. PLoS Comput Biol 13(5): e1005530. https://doi.org/10.1371/journal.pcbi.1005530

Editor: Timothée Poisot, Universite de Montreal, CANADA

Received: January 18, 2017

Accepted: April 20, 2017

Published: May 11, 2017

Data Availability Statement: The source code is available under the GNU GPLv3 at https://github.com/i-git/FIMTrack and pre-compiled binaries for Windows and Mac are available at http://fim.uni-muenster.de. The ground truth dataset can be downloaded at http://fim.uni-muenster.de.

Funding: This work has been partially supported by the Deutsche Forschungsgemeinschaft, Cluster of Excellence EXC 1003, Cells in Motion, CiM, Münster, Germany (PP-2014-05, FF-2016-06, CIM-IMPRS fellowship to NO). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

established knowledge base provide advantages like cell specific manipulations and ease behavioral inferences [1, 2]. Different tracking and locomotion analysis tools have been proposed including commercially available (e.g. EthoVision [3]) and custom solutions (e.g. MWT [4], MAGAT [5], SOS [6]). In the past we have introduced a novel imaging technique called FIM [7] to gather high-contrast recordings of the aforementioned model organisms. The associated open-source tracking software FIMTrack has already been used in a variety of studies [7–11] and a video tutorial has been published in [12] to demonstrate its biological usability. For example, FIMTrack has successfully been used to identify a central neural pathway for odor tracking in Drosophila [9] and to study the behavioral changes of *uba-5* knockout C. elegans worms [13].

Here we elaborate on the technical aspects and algorithms implemented in FIMTrack for a better understanding of the resultant quantities. Additionally, we provide an accuracy quantification using manually labeled data.

FIMTrack offers several advantages compared to state-of-the-art tracking tools:

- The assignment of animals across frames is implemented in a modular fashion, offering different combinations of assignment strategies and cost functions, making FIMTrack more flexible for a wider range of model organisms, locomotion types, and camera properties.
- FIMTrack extracts a huge variety of posture and motion-related features with a very high tracking accuracy which is evaluated using labeled data.
- Our tracking program has an intuitive graphical user interface allowing the inspection of most of the calculated features, an option for manual tracking, and an easy integration of stimulus regions.
- FIMTrack does not rely on commercial packages and is available in source code and as precompiled binaries for Windows and Mac.
- The software is implemented in an object-oriented fashion to improve re-usability and enable extensibility.

The main purposes of this paper are:

- Elaborate the algorithmic insights of the widely used FIMTrack software to enable easier usage and extensibility.
- Provide a ground truth-based evaluation of the tracking performance.
- Give an update on the current state of the program featuring a variety of novel functionality compared to its first usage in 2013 [7].
- Introduce FIMTrack as a tool for other communities dealing with other model organisms.

Design and implementation

FIMTrack is written in C++ and is easily extendable since the object-oriented programming paradigm is used. We utilize the OpenCV library and the Qt framework in combination with QCustomPlot (http://qcustomplot.com/) for image processing and the graphical user interface. Generally, FIMTrack consists of three main modules, namely the *tracker*, the *results viewer*, and the *input-output (IO)* module.

Tracker module

The main flow of the tracking module is given in Fig 1 and can be separated into *image processing*, *model extraction*, and *tracking*.

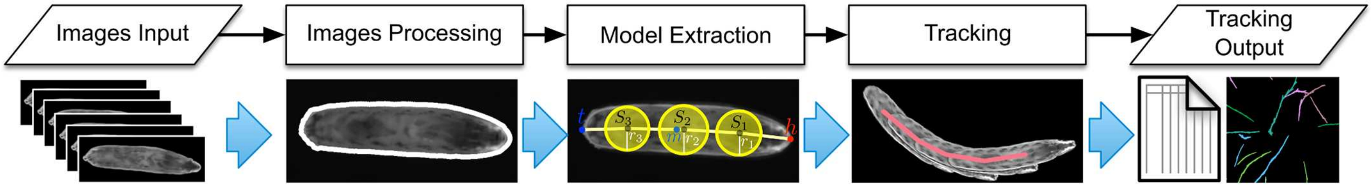


Fig 1. Flow chart of the tracking module.

https://doi.org/10.1371/journal.pcbi.1005530.g001

Image processing. Let $\mathcal{I}_t$ be the gray scale image at time t and assume that N animals in total need to be tracked. Prior to further image analysis we compute a static background image which includes almost all immovable artifacts. Since images produced by FIM have a black background with bright foreground pixels and since we assume that an animal moves more than its own body length during the recording, the calculation of the background image $\mathcal{B}$ can be done using the minimal pixel intensity value over time, resulting in

$$\mathcal{B}(r,c) = \min_t \left(\mathcal{I}_t(r,c)\right)$$

where $\mathcal{I}_t(r,c)$ is the pixel intensity at row r and column c at time t.

Subsequently, the foreground image $\mathcal{F}_t$ containing almost all objects of interest without the artifacts present in the background image $\mathcal{B}$ is obtained by

$$\mathcal{F}_t(r,c) = \begin{cases} \mathcal{I}_t(r,c) & \text{if } |\mathcal{I}_t(r,c) - \mathcal{B}(r,c)| \geq \tau_{\mathcal{B}}(r,c) \\ 0 & \text{otherwise} \end{cases}$$

with $\tau_{\mathcal{B}}(r,c) = \tau + \mathcal{B}(r,c)$ where τ is a user set gray value threshold. Given $\mathcal{F}_t$ the contours c_t^i of the animals are calculated by using the algorithm proposed in [14] resulting in a set of contours $C_t = \{c_t^1, c_t^2, \ldots, c_t^{N_t}\}$. N_t might differ from N since animals can be in contact with each other (leading to merged contours) or impurities on the substrate which are not included in the background image lead to artifacts. However, the contours in C_t can be filtered to identify single animals by assuming that all imaged animals cover approximately the same area. The filtered set $\tilde{C}_t$ of contours is given by

$$\tilde{C}_t = \left\{c_t^i \mid \lambda_{\min} \leq A(c_t^i) \leq \lambda_{\max},\ i = 1, \ldots, N_t\right\}$$

where $\lambda_{\min} < \lambda_{\max}$ are two user defined thresholds and $A(c_t^i)$ is the contour area given by the number of pixels enclosed by c_t^i. Both the contours that fulfill $A(c_t^i) > \lambda_{\max}$ which are assumed to represent colliding animals and contours with $A(c_t^i) < \lambda_{\min}$ which are assumed to be artifacts are ignored in further calculations.

Model extraction. For each contour $c_t^i \in \tilde{C}_t$ we compute a model representation of the associated animal. First, the spine is calculated based on curvature values which are obtained for each point $\mathbf{p}$ on the contour using the first pass of the IPAN algorithm [15]. Given all curvatures $\angle_{\mathbf{p}}$ the two regions with sharpest acute angle are located using a sliding window approach as illustrated in Fig 2a. The point with the sharpest overall mean angle is identified as head $\mathbf{h}$ and the point with the second sharpest mean angle (with an appropriate distance $\delta_{h \leftrightarrow t}$ to $\mathbf{h}$) is identified as tail $\mathbf{t}$. This assignment is done during the model extraction for each frame individually. In order to avoid head and tail switches and to ensure a reliable identification of these points for different organisms like larvae or C. elegans, the positions of the head and tail are refined in a post processing step using posture and motion features over time (see Post processing section).

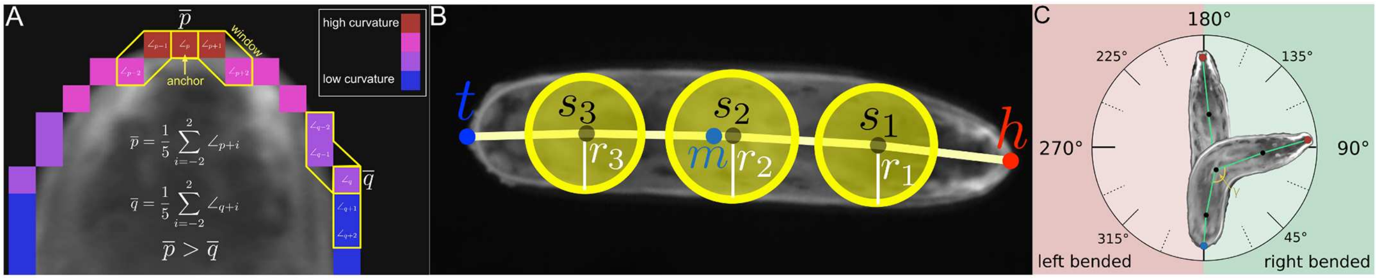


Fig 2. Calculation of the animal representation. (**A**) Example of the sliding window algorithm for points $\bar{\mathbf{p}}$ and $\bar{\mathbf{q}}$ with a window size of 5. Contour points with sharp angles are given in red. (**B**) Animal representation including the notation given in the text. (**C**) Body bending is calculated based on Eq (3). An animal is not bended if $\gamma = 180^\circ$, bended to the left if $\gamma > 180^\circ$ and bended to the right if $\gamma < 180^\circ$.

https://doi.org/10.1371/journal.pcbi.1005530.g002

The initial **h** and **t** identification is used to split the contour into two halves $c^i_{t,1}$ and $c^i_{t,2}$. Without loss of generality, let $|c^i_{t,1}| < |c^i_{t,2}|$. Then $c^i_{t,2}$ is re-sampled, so that $|c^i_{t,1}| = |c^i_{t,2}|$ by utilizing linear interpolation between points in $c^i_{t,2}$. Now each point $\mathbf{p}^j_1 \in c^i_{t,1}$ corresponds to a unique point $\mathbf{p}^j_2 \in c^i_{t,2}$. The spine points $\mathbf{s}_1, \ldots, \mathbf{s}_{L-2}$ are calculated by determining $L - 2$ equidistant pairs $(\mathbf{p}^j_1, \mathbf{p}^j_2)$ along $c^i_{t,1}$ and $c^i_{t,2}$ and by setting $\mathbf{s}_j = \frac{\mathbf{p}^j_2 + \mathbf{p}^j_1}{2}$. The radii r_j are calculated similarly by $r_j = \frac{\|\mathbf{p}^j_2 - \mathbf{p}^j_1\|}{2}$. In addition, the center of mass **m** is calculated based on c^i_t. As a result, each animal i at time t is defined by a model

$$l^i_t = (\mathbf{h}, (\mathbf{s}_1, r_1), \ldots, (\mathbf{s}_{L-2}, r_{L-2}), \mathbf{t}, \mathbf{m}) \tag{1}$$

which is depicted in Fig 2b.

Tracking. Considering a sufficient spatio-temporal resolution tracking can be done by assigning animals between consecutive frames: all active animals $\mathcal{L}_t = \{l^i_t | i = 1, \ldots, N_t\}$ at time t need to be associated with either one detected animal from the set $\mathcal{L}_{t+1} = \{l^j_{t+1} | j = 1, \ldots, N_{t+1}\}$ at time $t + 1$ or removed from the set of active animals. Mathematically this assignment is known as a bipartite matching between $\mathcal{L}_t$ and $\mathcal{L}_{t+1}$. Let the costs for an assignment between an animal i at time t and an animal j at time $t + 1$ be κ_{ij}. This leads to the following cost matrix:

$$\mathcal{C}_t = \begin{array}{c} \\ l^1_t \\ l^2_t \\ \vdots \\ l^{N_t}_t \end{array} \left(\begin{array}{cccc} l^1_{t+1} & l^2_{t+1} & \cdots & l^{N_{t+1}}_{t+1} \\ \kappa_{11} & \kappa_{12} & \cdots & \kappa_{1N_{t+1}} \\ \kappa_{21} & \kappa_{22} & \cdots & \kappa_{2N_{t+1}} \\ \vdots & \vdots & \ddots & \vdots \\ \kappa_{N_t 1} & \kappa_{N_t 2} & \cdots & \kappa_{N_t N_{t+1}} \end{array} \right) \in \mathbb{R}^{N_t \times N_{t+1}}$$

The following three cost measurements are implemented in FIMTrack:

1. Euclidean distance between the center of masses (Fig 3a): $\kappa_{ij} = ||\mathbf{m}^i_t - \mathbf{m}^j_{t+1}||_2$
2. Euclidean distance between the mid-points (Fig 3b): $\kappa_{ij} = ||\mathbf{s}^i_{\frac{L}{2},t} - \mathbf{s}^j_{\frac{L}{2},t+1}||_2$
3. The intersecting area in pixels given two consecutive contours (Fig 3c)

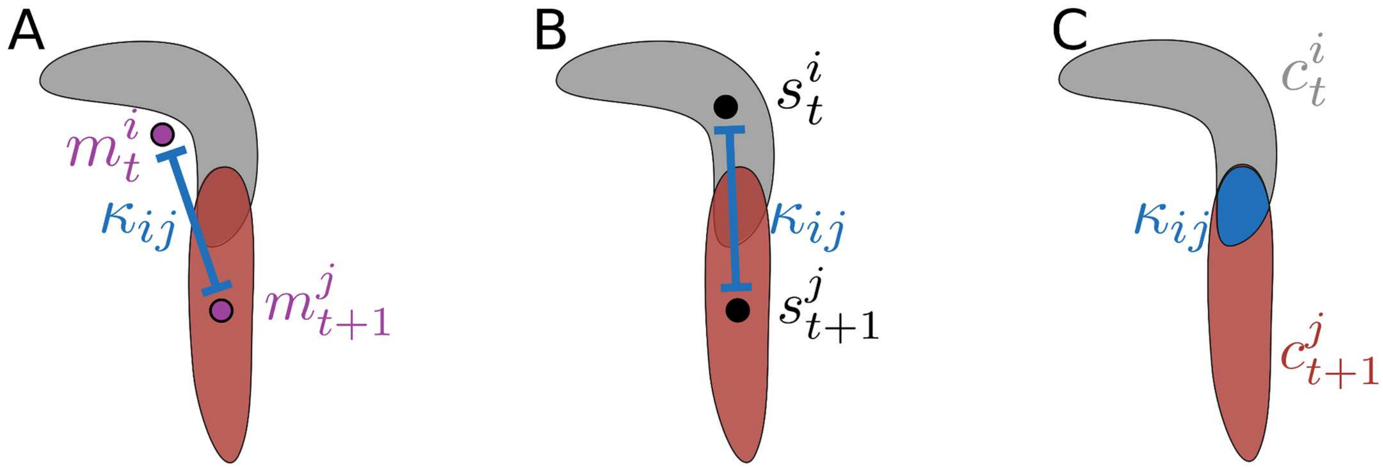


Fig 3. Different cost measurements. (**A**) Center of mass-based cost. (**B**) Spine-based cost. (**C**) Contour-based cost.

https://doi.org/10.1371/journal.pcbi.1005530.g003

Since the center of mass is extracted from the contour, the translation between consecutive frames is very smooth. In contrast, the middle spine point contains more jitter, but is forced to be inside the contour. Costs derived from the overlap of two consecutive contours fulfill both criteria as they are smooth and based on pixels within the contours.

To solve the aforementioned assignment problem two algorithms are implemented in FIMTrack. One is the Hungarian algorithm [16, 17] which has one drawback: Given distance-based costs, the algorithm will find assignments for all $\tilde{n} = \min(N_t, N_{t+1})$ animals. For example, if animal *i* disappears at time *t* while another animal *j* appears at time *t* + 1, the algorithm will assign these two animals even if the Euclidean distance between them is very large. Thus, we check each assignment if at least one point of the model of animal *i* is inside the contour of animal *j*. Otherwise animal *i* is considered to be inactive, the associated trajectory is terminated, and animal *j* is initialized as a new animal.

The second algorithm follows the greedy pattern. Given $\mathcal{L}_t$ and $\mathcal{L}_{t+1}$ the algorithm determines sequentially for each animal $l_t^i \in \mathcal{L}_t$ the best matching animal $l_{t+1}^j \in \mathcal{L}_{t+1}$ given one cost measurement (note that each match has to be unique). To exclude irrational assignments this algorithm requires an additional threshold τ_{greedy} specifying the maximal distance between two consecutive points if distance-based costs are used or the minimal amount of overlap in case of contour-based costs.

The possibility to choose between multiple costs and optimization algorithms extends the range of organisms which can be analyzed even at various spatial and temporal resolutions. For example, during peristaltic forward locomotion of Drosophila larvae, the Hungarian algorithm in combination with overlap-based costs is feasible to associate the larvae over time (Fig 4a). In contrast, rolling larvae with unsuitable temporal resolution lead to false assignments using the Hungarian algorithm with overlapping contour costs: due to strong changes in the body bending and the relatively fast lateral locomotion no overlaps can be detected within contours of consecutive frames (Fig 4b). Similarly, C. elegans moves in a snake like motion so that contour-overlap-based assignments may fail (Fig 4c).

After processing all frames, the overall path of an animal *i* is given by

$$\mathbb{L}^i = (l_{t_1}^i, l_{t_1+1}^i, \ldots, l_{t_2}^i) \tag{2}$$

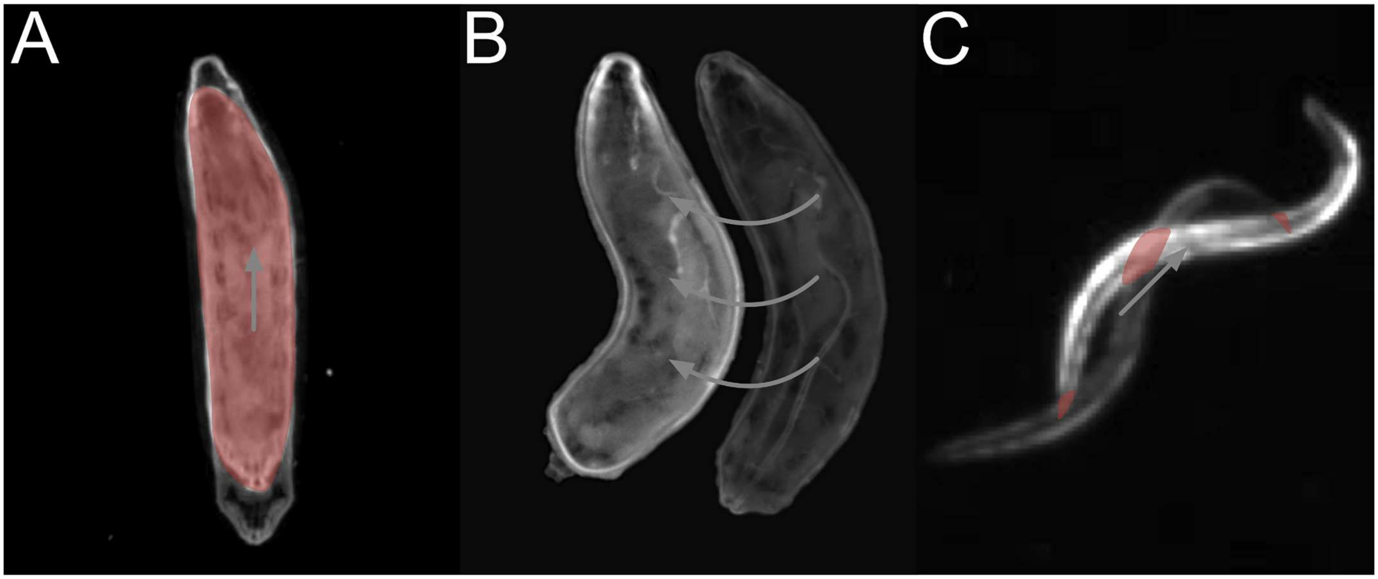


Fig 4. Examples of different assignment strategies. Overlapping regions are given in red and locomotion is indicated by arrows. (**A**) During forward locomotion of Drosophila larvae the association can be done using overlap-based costs and the Hungarian algorithm. (**B**) Given rolling behavior of Drosophila larvae, an assignment using contour overlaps as costs for the Hungarian algorithm might fail due to inappropriate frame rates. (**C**) For tracking the snake-like motion of C. elegans, contour-based assignments might be insufficient.

https://doi.org/10.1371/journal.pcbi.1005530.g004

where t_1 specifies the first frame the animal appears at and t_2 indicates the last valid measurement ($1 \leq t_1 \leq t_2 \leq T$; T represents the total number of frames).

Post processing. The initial definition of the larval orientation is based on two regions with the sharpest acute angles. Due to the non-rigid body wall or low per-animal resolutions this assumption is not true in some frames leading to alternating head/tail assignments. Furthermore, other model organisms like C. elegans can imply other curvature characteristics compared to larvae which leads to the necessity to correct the initial **h** and **t** calculation.

To adjust the assignments in $\mathbb{L}^i$ (Eq 2) first distinct sequences $\tilde{\mathbb{L}}^i \subseteq \mathbb{L}^i$ where the animal is not coiled are determined. Afterwards, a probability indicating whether the head/tail assignment is correct or not is calculated for all tuple $l_t^i \in \tilde{\mathbb{L}}^i$ based on the following constrains:

1. Locomotion conformity (i.e. the mid-point/head vector points in the direction of the locomotion)
2. Bending conformity (i.e. larvae move the head but not the tail during reorientations)

If the head and tail are swapped, the spine points $\mathbf{s}_i$ and the radii $\mathbf{r}_i$ are reversed, too. Furthermore, all spine-point derived features are recalculated. Note that, although these constraints are derived from larval locomotion, the resultant probability is still valid if a C. elegans worm moves forward in more than 50% of the frames. It is worth mentioning that, after identifying **h** and **t**, the position of these points along the spine is fixed by assigning each subsequent head/tail point based on the respective predecessor with smallest Euclidean distance. Furthermore, even if the head and tail are swapped one click in the results viewer module is sufficient to correct the model throughout the entire recording (see Results reveiwer module).

Feature calculation. To quantify the locomotion in more detail several primary, secondary, and motion-related features are calculated by FIMTrack.

Primary features. In addition to the representation of an animal l_t^i (Eq 1) the area A and the perimeter P are calculated.

Secondary features. The main body bending angle γ is calculated based on the head $\mathbf{h}$, the mid spine point $\mathbf{s}_{\frac{\mathbf{L}}{2}}$, and the tail $\mathbf{t}$. Given the vectors $\mathbf{v_1} = \mathbf{h} - \mathbf{s}_{\frac{\mathbf{L}}{2}}$ and $\mathbf{v_2} = \mathbf{s}_{\frac{\mathbf{L}}{2}} - \mathbf{t}$, Eq (3) is used to calculate the bending in degree.

$$\gamma = \arccos\left(\left\langle \frac{\mathbf{v_1}}{\|\mathbf{v_1}\|}, \frac{\mathbf{v_2}}{\|\mathbf{v_1}\|} \right\rangle\right) \cdot \frac{180^\circ}{\pi} \tag{3}$$

As a consequence, an animal is not bended if $\gamma = 180^\circ$, bended to the left if $\gamma > 180^\circ$ and bended to the right if $\gamma < 180^\circ$ (Fig 2c). Since a user-specified number of spine points can be extracted these points can be used to extract other bendings along the spine by using Eq (3) with appropriate $\mathbf{v_1}$ and $\mathbf{v_2}$ (e.g. to quantify the stereotypical S-shape of C. elegans).

Given a threshold τ_{bend}, an animal sweeps to the left if $\gamma \geq 180^\circ + \tau_{\text{bend}}$ and sweeps to the right if $\gamma \leq 180^\circ - \tau_{\text{bend}}$. Furthermore, the spine length S_l is calculated by summing up the Euclidean distances between the head, all spine points, and the tail:

$$S_l = \|\mathbf{h} - \mathbf{s_1}\|_2 + \sum_{i=1}^{L-3} \|\mathbf{s_i} - \mathbf{s_{i+1}}\|_2 + \|\mathbf{s_{L-2}} - \mathbf{t}\|_2$$

In case of coiled animals (Fig 5a) the spine calculation fails to extract the posture correctly. As a consequence, all spine-related features are not reliable. To mark these ambiguous situations, a binary indicator $c?$ is introduced to determine coiled states. The coiled indicator is true if one of the following constraints is satisfied:

1. The perimeter to spine length fraction converges to π ($\frac{P}{L} \approx \pi$; Fig 5b)
2. The circle given by the mid spine point radius divided by the perimeter converges to 1 ($\frac{2r_{\text{mid}}\pi}{P} \approx 1$; Fig 5c)

Motion-related features. Most of the motion-related features are calculated based on the animal's center of mass $\mathbf{m}$ since this point is calculated directly from the contour and does not depend on the spine calculation. The accumulated distance for an animal at time t is calculated

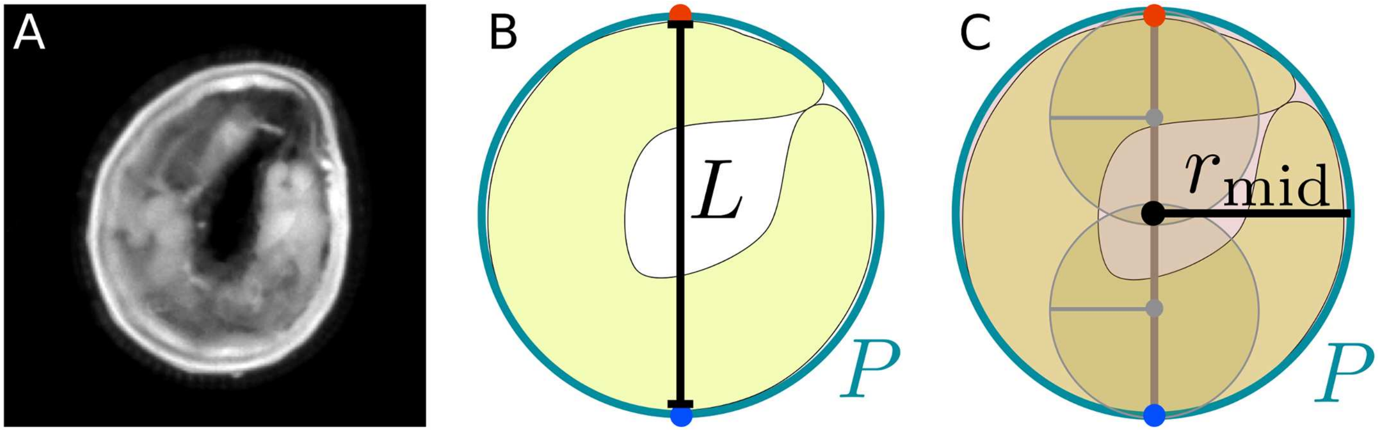


Fig 5. Two constraints are used to determine if an animal is in a coiled state. (**A**) Larva in a coiled state. (**B**) Perimeter P to spine length L fraction. (**C**) Mid spine point radius r_{mid} to perimeter P ratio.

https://doi.org/10.1371/journal.pcbi.1005530.g005

by

$$d_{\mathrm{acc}} = \sum_{i=1}^{t-1} \|\mathbf{m_i} - \mathbf{m_{i+1}}\|_2$$

and the distance to origin is given by

$$d_{\mathrm{org}} = \| \mathbf{m_1} - \mathbf{m_t}\|_2$$

Furthermore, the velocity for time t is calculated by

$$v_m = \frac{\| \mathbf{m}_{\mathbf{t}-\frac{\mathrm{fps}}{\mathbf{2}}} - \mathbf{m}_{\mathbf{t}+\frac{\mathrm{fps}}{\mathbf{2}}}\|_2}{\mathrm{fps}}$$

where fps is the given frame rate. In a similar fashion, the acceleration is obtained by using consecutive velocities:

$$a_m = \frac{\| v_{m,t} - v_{m,t+1}\|_2}{2}$$

To identify if an animal is in a go phase, a binary indicator $g?$ is used. The following constraints must be valid for a go phase:

- The velocity v_m is above a certain threshold
- There is no strong bending γ of the animal's body

To avoid alternating go phase measurements, a user-specified minimal go phase length (τ_{go}) is used to extract continuous phases. This implies that the number of consecutive $g?$ = *true* measurements has to be $\geq \tau_{\mathrm{go}}$ to classify this sequence as a go phase. An animal is in a reorientation phase if $g?$ is false.

Stimulus-related features. In order to extend the capabilities of FIMTrack it is possible to place different stimulus markers on the raw image in the results viewer. The following markers are supported:

- **Point** which is a (x, y) position
- **Line** which is a straight line segment
- **Rectangle** which is an arbitrarily sized axis-aligned 2D rectangle
- **Ellipsoid** which is an arbitrarily sized axis-aligned 2D ellipsoid

For all markers the additional features *distance to stimulus*, *bearing angle to stimulus*, and *is in stimulus region* are calculated for each time point and animal. The distance to a stimulus is given by the Euclidean distance between the center of mass **m** of the animal and the point **p** representing the nearest point on the stimulus. The bearing angle β is obtained by using Eq (3) with $\mathbf{v_1} = \mathbf{m} - \mathbf{t}$ and $\mathbf{v_2} = \mathbf{p} - \mathbf{t}$ (note that the tail **t** is used since it is not affected by head casts).

Given a **point stimulus** the necessary computations are straight forward. The nearest point between the animal and a **line stimulus** is obtained by performing an orthogonal projection of **m** on the line defined by the line segment. If the projected point **p** is not located on the line segment we take one of the two endpoints of the line segment as the nearest point which has the minimal Euclidean distance to **p**. In case of a **rectangular stimulus** the nearest point **p** is calculated by $\mathbf{p} = \arg\min_{\mathbf{p}_i} \|\mathbf{m} - \mathbf{p}_i\|_2$, where $\mathbf{p}_i$ is the associated orthogonal protection of **m** onto each of the four boundaries of the stimulus. Since the calculation of the exact nearest point **p** of **m** on a **ellipsoid stimulus** cannot easily be performed in an analytical fashion, we use an approximation of $\mathbf{p}$. Given an axis-aligned ellipse centered at $\mathbf{c} = (c_x, c_y)$. First, a line $\mathbf{l}$ going through $\mathbf{m}$ and $\mathbf{c}$ is determined. Next the intersection points $\mathbf{p_1}$ and $\mathbf{p_2}$ between $\mathbf{l}$ and the ellipse are obtained and the intersection point with the minimum distance to $\mathbf{m}$ is taken as the approximation of $\mathbf{p}$.

Results viewer module

The *results viewer* module offers the possibility to review the calculated features. The experimenter can load, display, and manually correct the posture and motion-related features or even manually track some animals if they could not be recognized automatically. If an animal model is adjusted manually all features are updated accordingly. The results viewer module itself is divided into three main parts, namely the *image view* (Fig 6a), the *table view* (Fig 6b) and the *animal view* (Fig 6c).

The image view provides a qualitative impression of the tracking results. Most of the calculated features are plotted color coded for each animal as an overlay onto the raw images. Moreover, the user can manually change the calculated model or merge/remove trajectories. This is particularly helpful to resolve ambiguous situations like coiled or colliding animals. In the table view all calculated features can be inspected in a table showing animals in columns and the associated features in rows. The animal view can be used to inspect the results for a single animal in more detail: both a cropped region of a single animal and plots for relevant features can be obtained simultaneously.

IO module

This module is responsible for reading and writing files. Currently the image file formats TIFF and PNG are supported. A CSV file containing all calculated features, a YML file including the same measurements as the CSV file in combination with some additional informations like the processed images, and an image with color coded trajectories are generated after tracking has been performed. It should be mentioned that the CSV format is standardized and can directly be imported into a variety of analysis programs (e.g. MATLAB, Excel, R).

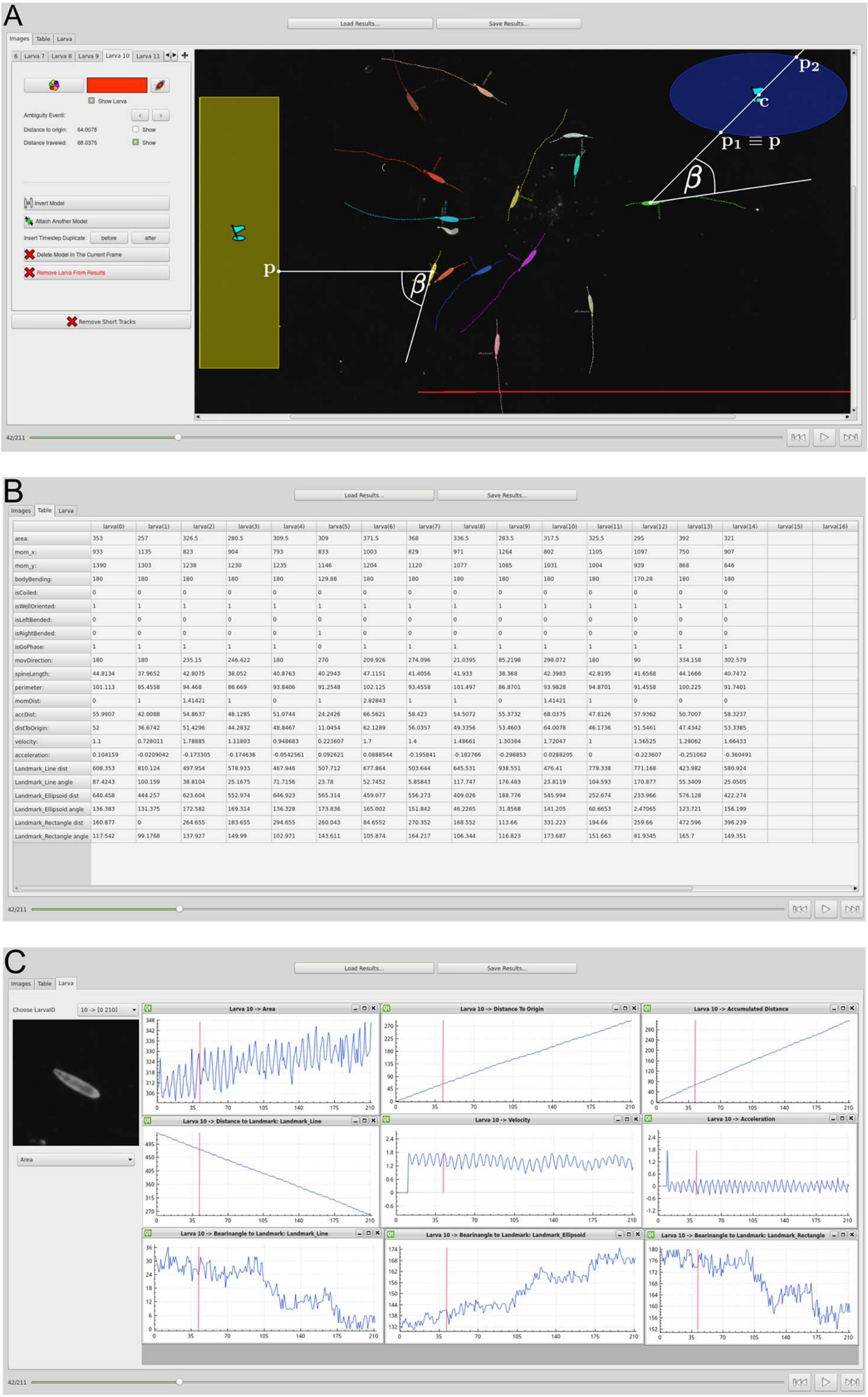
Results

Most of the calculated features rely on a precise calculation of the center of mass (e.g. accumulated distance, velocity, bearing, etc.). Furthermore, good candidates to assess the quality of the calculated model are the mid spine point and the body bending (most of the secondary and motion-related features are derived from the underlying model). Here, we evaluate the accuracy of the software regarding these features.

Ground truth data

An image sequence with a resolution of 2040 × 2048 pixels acquired with a Basler acA2040-25gm camera equipped with a 16mm objective (KOWA LM16HC) containing 15 larvae over 211 frames was used to generate ground truth data (Fig 7a and 7b). All animals were associated with a larval model consisting of the head, tail, and 5 equidistant spine points associated with appropriate radii. Furthermore, the center of mass was calculated based on these models.

For the subsequent analysis we considered the 10 trajectories of the larvae which could be tracked by FIMTrack over all 211 frames (i.e. larvae which crawl at least by there own body length and do not collide).


A
Load Results...
Save Results...
Images
Table
Larva
6
Larva 7
Larva 8
Larva 9
Larva 10
Larva 11
Show Larva
Invert Model
Attach Another Model
Insert Timestep Duplicate:
before
after
Delete Model In The Current Frame
Remove Larva From Results
Remove Short Tracks
p
β
p₂
c
p₁ ≡ p
β
42/211
B
C
Larva 10 -> Area
Larva 10 -> Distance To Origin
Larva 10 -> Accumulated Distance
Larva 10 -> Distance to Landmark: Landmark_Line
Larva 10 -> Velocity
Larva 10 -> Acceleration
Larva 10 -> Bearinangle to Landmark: Landmark_Line
Larva 10 -> Bearinangle to Landmark: Landmark_Ellipsoid
Larva 10 -> Bearinangle to Landmark: Landmark_Rectangle
0
35
70
105
140
175
210

Ambiguity Eventl:

Distance to origin: 64.0078 □ Show

Distance traveled: 68.0375 ☒ Show

	larva(0)	larva(1)	larva(2)	larva(3)	larva(4)	larva(5)	larva(6)	larva(7)	larva(8)	larva(9)	larva(10)	larva(11)	larva(12)	larva(13)	larva(14)	larva(15)	larva(16)
area:	353	257	326.5	280.5	309.5	309	371.5	368	336.5	283.5	317.5	325.5	295	392	321		
mom_x:	933	1135	823	904	793	833	1003	829	971	1264	802	1105	1097	750	907		
mom_y:	1390	1303	1238	1230	1235	1146	1204	1120	1077	1085	1031	1004	939	868	846		
bodyBending:	180	180	180	180	180	129.88	180	180	180	180	180	180	170.28	180	180		
isCoiled:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
isWellOriented:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
isLeftBended:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
isRightBended:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0		
isGoPhase:	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1		
movDirection:	180	180	235.15	246.422	180	270	209.926	274.096	21.0395	85.2198	298.072	180	90	334.158	302.579		
spineLength:	44.8134	37.9652	42.8075	38.052	40.8763	40.2943	47.1151	41.4056	41.933	38.368	42.3983	42.8195	41.6568	44.1666	40.7472		
perimeter:	101.113	85.4558	94.468	86.669	93.8406	91.2548	102.125	93.4558	101.497	86.8701	93.9828	94.8701	91.4558	100.225	91.7401		
momDist:	0	1	1.41421	1	0	1	2.82843	1	1	0	1.41421	1	0	0	0		
accDist:	55.9907	42.0088	54.8637	48.1285	51.0744	24.2426	66.5621	58.423	54.5072	55.3732	68.0375	47.8126	57.9362	50.7007	58.3237		
distToOrigin:	52	36.6742	51.4296	44.2832	48.8467	11.0454	62.1289	56.0357	49.3356	53.4603	64.0078	46.1736	51.5461	47.4342	53.3385		
velocity:	1.1	0.728011	1.78885	1.11803	0.948683	0.223607	1.7	1.4	1.48661	1.30384	1.72047	1	1.56525	1.28062	1.66433		
acceleration:	0.104159	-0.0209042	-0.173305	-0.174636	-0.0542561	0.092621	0.0888544	-0.195841	-0.182766	-0.298853	0.0288205	0	-0.223607	-0.251062	-0.360491		
Landmark_Line dist	608.353	810.124	497.954	578.933	467.946	507.712	677.864	503.644	645.531	938.551	476.41	779.338	771.168	423.982	580.924		
Landmark_Line angle	87.4243	100.159	38.8104	25.1675	71.7156	23.78	52.7452	5.85843	117.747	176.483	23.8119	104.593	170.877	55.3409	25.0505		
Landmark_Ellipsoid dist	640.458	444.257	623.604	552.974	646.923	565.314	459.077	556.273	409.026	188.776	545.994	252.674	233.966	576.128	422.274		
Landmark_Ellipsoid angle	136.383	131.375	172.582	169.314	136.328	173.836	165.002	151.842	46.2265	31.8568	141.205	60.6653	2.47065	123.721	156.199		
Landmark_Rectangle dist	160.877	0	264.655	183.655	294.655	260.043	84.6552	270.352	168.552	113.66	331.223	194.66	259.66	472.596	396.239		
Landmark_Rectangle angle	117.542	99.1768	137.927	149.99	102.971	143.611	105.874	164.217	106.344	116.823	173.687	151.663	81.9345	165.7	149.351		

Choose LarvaID 10 -> [0 210]

Area

Fig 6. Results viewer module. (**A**) Image view with the raw image, an overlay of the color coded features, and two stimuli marker with the notations given in the text. (**B**) Table view. (**C**) Animal view with both a cropped region of a single larva and plots of some features.

https://doi.org/10.1371/journal.pcbi.1005530.g006

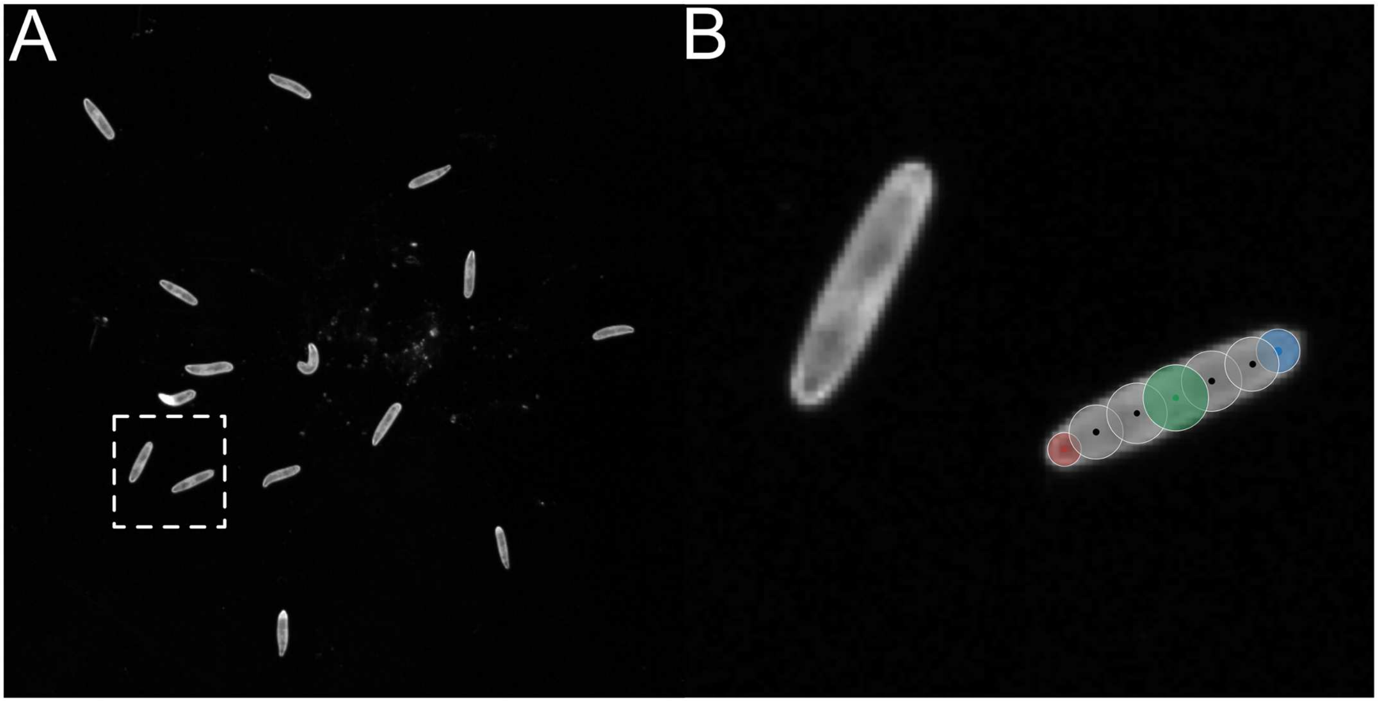


Fig 7. An image from the ground truth dataset and the manually generated model. (**A**) Exemplary image used for ground truth generation. (**B**) Close up of the dashed box from Fig 7a.

https://doi.org/10.1371/journal.pcbi.1005530.g007

Measured deviations

Deviations from the ground truth are determined by calculating the Euclidean distances between the tracking results and ground truth data for both the center of mass and the central spine point. For the body bending, absolute differences are used to determine the accuracy. Table 1 illustrates the deviations.

Obviously the deviation of the center of mass and the deviation of the central spine point are below 2 pixels in mean and median. It should be noted that during tracking no sub-pixel accuracy is used and thus the minimum possible error is 1 pixel if the displacement happens either in x or y direction. For a diagonal displacement the minimum possible error is $\sqrt{2} \approx 1.41$ pixels. In combination with the area of the animals which range from 232.50 to 454.50 square pixels, this leads us to the assertion that deviations below 2 pixels are caused due to noise.

Center of mass. A detailed overview of the center of mass progress is given in Fig 8a. Each boxplot represents the center of mass deviation for the respective larva. None of the measurements has a median deviation above 3 pixels. This suggests that the divergence is rather the

Table 1. Deviations for the examined parameters. Body bending is given in degree, all other parameters are given in pixels. Max* represents the values obtained by including outliers whereas OL gives the number of outliers.

Deviations	Mean (±Std)	Median	Min	Max	Max*	OL
Center of mass	1.86 (±0.23)	1.85	1.21	2.55	2.84	17 (0.80%)
Central spine point	1.84 (±1.10)	1.57	0.50	4.58	16.84	61 (2.89%)
Body bending	3.54 (±7.51)	2.55	0.00	13.16	171.00	46 (2.18%)

https://doi.org/10.1371/journal.pcbi.1005530.t001

result of an inaccuracy of the tracking algorithm but more likely caused by the previously performed image processing and definitely influenced by the non-contour-based center of mass extraction in the ground truth data.

Central spine point. The central spine point location contains more outliers compared to the center of mass measurements. The maximal deviation (including outliers) between a measurement and the ground truth is 16.84 pixels (Table 1). As illustrated in Fig 8b measurements

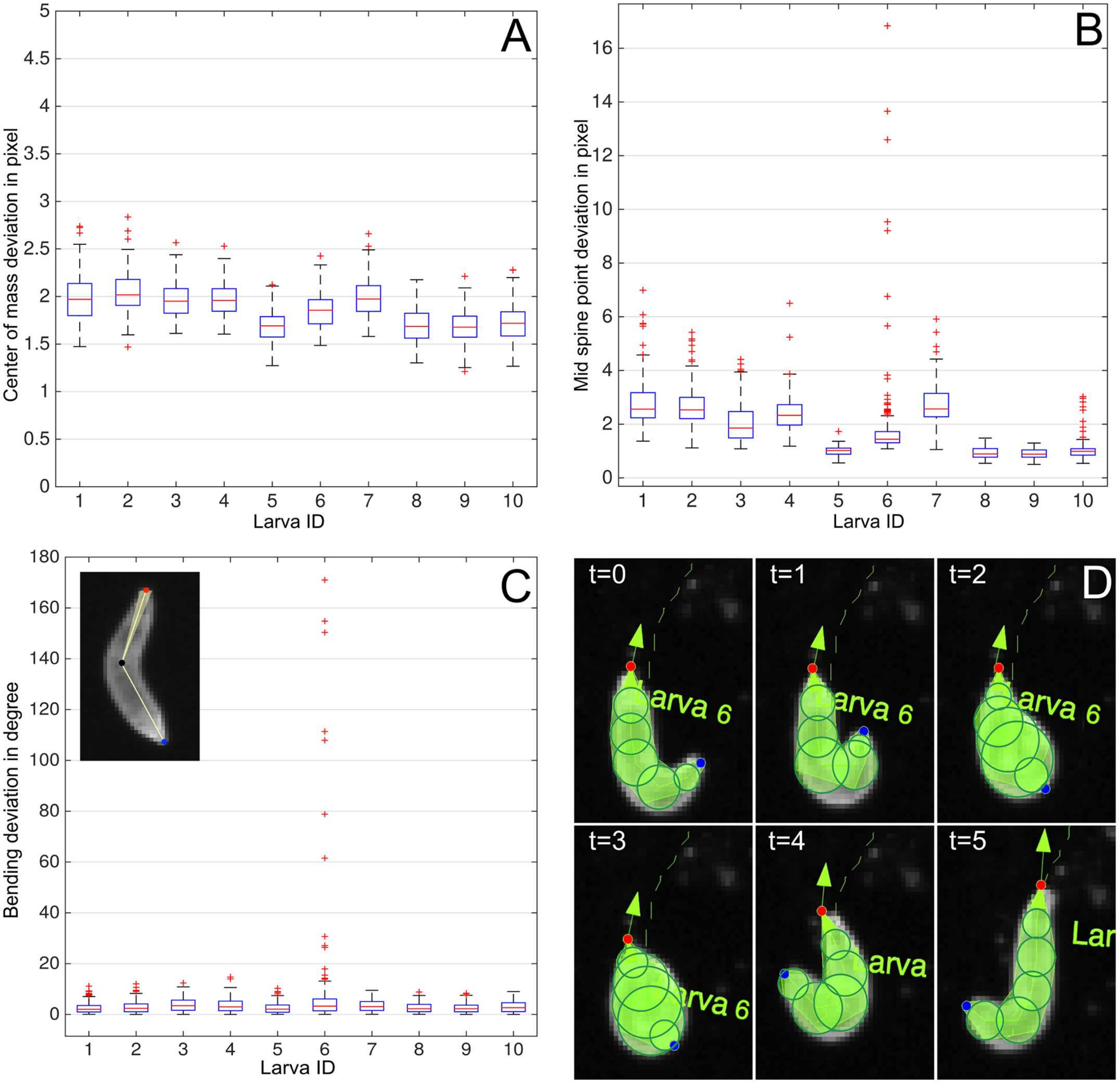


Fig 8. Measured deviations. (**A**) Center of mass deviations. (**B**) Central spine point deviations. (**C**) Body bending angle deviations. The mean divergence of the body bending is sketched by the light yellow area in the larva image at the top left corner. (**D**) The coiled structure of larva 6 (at *t* = 2, 3) causes outliers in the measurements (compare to Table 1). The head is given in red and tail is given in blue.

https://doi.org/10.1371/journal.pcbi.1005530.g008

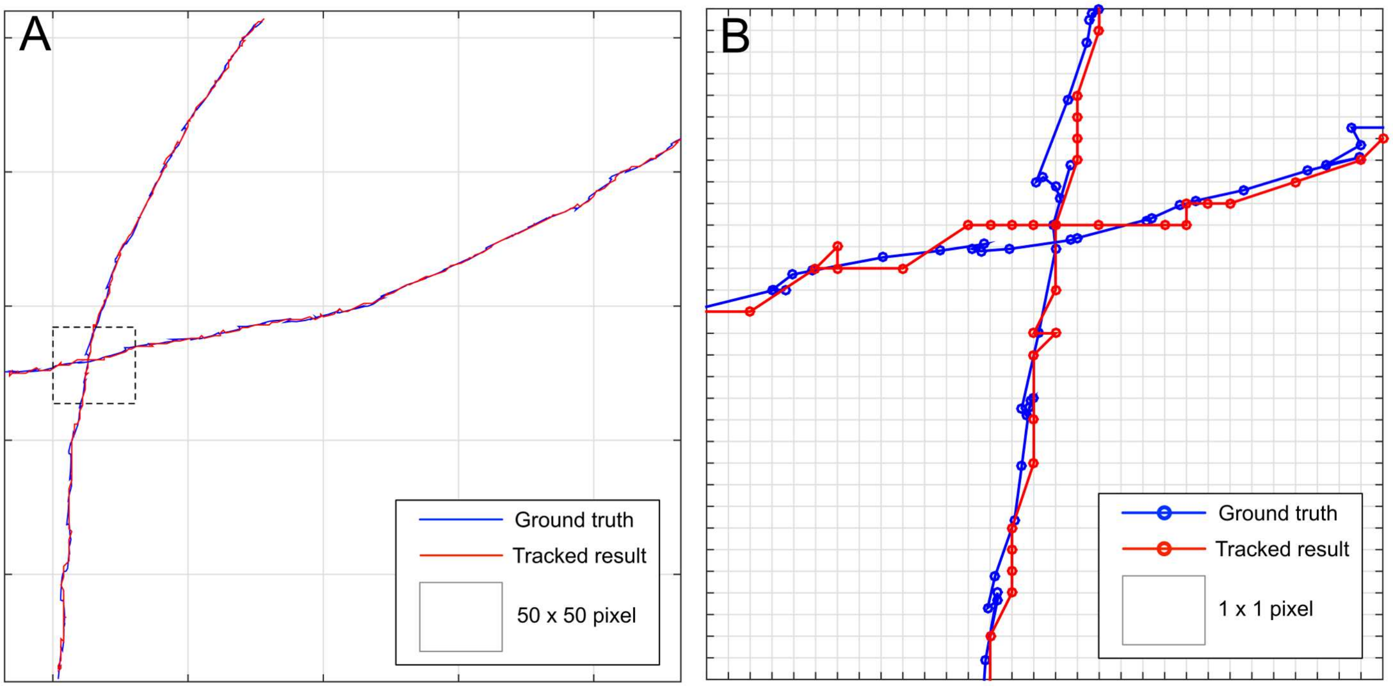


Fig 9. Resultant center of mass trajectories compared to the ground truth paths. (**A**) Center of mass point of the ground truth and tracked larvae. (**B**) Close-up or the dashed box from Fig 9a.

https://doi.org/10.1371/journal.pcbi.1005530.g009

for larva 6 include most outliers. These inaccuracies are caused within several frames in which the animal is coiled resulting in an erroneous spine calculation (Fig 8d). The median spine point deviation is below 2 pixels and after removing the outliers the maximum distance decreases to 4.58 pixels (Table 1).

To further study the accuracy an overlay of ground truth and calculated central spine point trajectories is given in Fig 9. Since no sub-pixel accuracy is used for tracking, the calculated path contains more straight lines interrupted by edges. However, the deviation from the ground truth path is rarely more than one pixel.

Body bending. A high tracking precision can also be observed within the body bending quantification: the mean deviation is below 4° (Table 1) which is depicted in the top left corner of Fig 8c where the head of the larva is given in red, the tail in blue, and the central spine point in black. The mean deviation is indicated by the light yellow area visible at both sides of the spine segment connecting the head and the tail.

By taking a closer look at the plots given in Fig 8c, it can be seen that again only larva 6 includes several frames with a very strong deviation. Since the deviations go up to 171° the head and the central spine point are swapped which can be observed in Fig 8d.

Availability and future directions

FIMTrack is freely available as a pre-built binary package for Windows and Mac at http://fim.uni-muenster.de. Further documentation and exemplary FIM-images for testing purposes are available at the same website. An open-access video tutorial for experimental biologists illustrating the usage of our system with and without stimulation can be found in [12].

The source code of FIMTrack is licensed under the GNU GPLv3 and can be obtained from https://github.com/i-git/FIMTrack. Users implementing new features or extensions are encouraged to submit their work via GitHub's pull request mechanism for inclusion in a coming release.

In the past, several others groups successfully used FIMTrack to differentiate between different behavioral phenotypes (examples for Drosophila larvae can be found in [9, 18, 19] and for C. elegans in [13]). Furthermore, the software has been used as the basis for extensions in order to address more specific biological questions [8, 10, 20].

It should be noted that FIMTrack has been initially developed for FIM images and Drosophila larvae [7]. For example, the algorithms described above only segment the animals if the background is darker than the foreground (i.e. the animals). However, we successfully adopted the algorithm to track images recorded with transmitted light illumination by inverting the images before passing them to FIMTrack. Furthermore, some of the extracted features are only valid for larval behavior like the stop and go classification. Otherwise, since the complete model of the animal (Eq 1) obtained after tracking, prepossessing, and maybe some user adjustments is saved in a standardized file format (i.e. CSV), higher-level features for other model organisms can be derived easily. Finally, FIMTrack does not include a module to resolve colliding animals so that the identities of animals participating in a collision get lost and the trajectories of these animals terminate. After the ending of the collision the associated animals receive new identities and are treated as newly appeared.

In the future, we are going to extend FIMTrack by optimizing the tracking for other model organisms like flatworms. In order to overcome the problem of losing identities and behavioral quantities during animal-animal contact, we are working on a statistical approach capable of resolving colliding animals.

Supporting information

S1 Data. In order to quantify the accuracy of FIMTrack we have manually tracked 15 larvae over 211 frames. The resultant quantities and the used evaluation script are provided in order to guarantee reproducibility of our results. Note that the images can be downloaded at http://fim.uni-muenster.de.
(ZIP)

S1 Text. FIMTrack manual describing the work flow.
(PDF)

Author Contributions

Conceptualization: BR DB NO CK XJ.

Data curation: BR DB NO CK XJ.

Resources: CK XJ.

Software: BR DB.

Validation: BR DB NO.

Writing – original draft: BR DB NO CK XJ.

References

1. Gohl DM, Silies MA, Gao XJ, Bhalerao S, Luongo FJ, Lin CC, et al. A versatile in vivo system for directed dissection of gene expression patterns. Nature Methods. 2011; 8(3):231–237. https://doi.org/10.1038/nmeth.1561 PMID: 21473015

2. Sokolowski MB. Drosophila: Genetics meets behaviour. Nature Reviews Genetics. 2001; 2(11): 879–890. https://doi.org/10.1038/35098592 PMID: 11715043
3. Spink AJ, Tegelenbosch RA, Buma MO, Noldus LP. The EthoVision video tracking system–a tool for behavioral phenotyping of transgenic mice. Physiology and Behavior. 2001; 73(5):731–744. PMID: 11566207
4. Swierczek NA, Giles AC, Rankin CH, Kerr RA. High-throughput behavioral analysis in C. elegans. Nature Methods. 2011; 8(7):592–598. https://doi.org/10.1038/nmeth.1625 PMID: 21642964
5. Gershow M, Berck M, Mathew D, Luo L, Kane EA, Carlson JR, et al. Controlling airborne cues to study small animal navigation. Nature Methods. 2012; 9(3):290–296. https://doi.org/10.1038/nmeth.1853 PMID: 22245808
6. Gomez-Marin A, Partoune N, Stephens GJ, Louis M. Automated Tracking of Animal Posture and Movement during Exploration and Sensory Orientation Behaviors. PLoS ONE. 2012; 7(8): e41642. PMID: 22912674
7. Risse B, Thomas S, Otto N, Löpmeier T, Valkov D, Jiang X, et al. FIM, a novel FTIR-based imaging method for high throughput locomotion analysis. PloS ONE. 2013; 8(1):e53963. https://doi.org/10.1371/journal.pone.0053963 PMID: 23349775
8. Risse B, Berh D, Otto N, Jiang X, Klämbt C. Quantifying subtle locomotion phenotypes of Drosophila larvae using internal structures based on FIM images. Computers in biology and medicine. 2015; 63:269–276. https://doi.org/10.1016/j.compbiomed.2014.08.026 PMID: 25280919
9. Slater G, Levy P, Chan KA, Larsen C. A Central Neural Pathway Controlling Odor Tracking in *Drosophila*. The Journal of Neuroscience. 2015; 35(5):1831–1848. https://doi.org/10.1523/JNEUROSCI.2331-14.2015 PMID: 25653345
10. Otto N, Risse B, Berh D, Bittern J, Jiang X, Klämbt C. Interactions among Drosophila larvae before and during collision. Scientific Reports. 2016; 6:31564. https://doi.org/10.1038/srep31564 PMID: 27511760
11. Kiel M, Berh D, Daniel J, Otto N, ter Steege A, Jiang X, et al. A Multi-Purpose Worm Tracker Based on FIM; in preparation.
12. Risse B, Otto N, Berh D, Jiang X, Klämbt C. FIM Imaging and FIMTrack: Two New Tools Allowing High-Throughput and Cost Effective Locomotion Analysis. The Journal of Visualized Experiments. 2014; (94):e52207. https://doi.org/10.3791/52207 PMID: 25591081
13. Colin E, Daniel J, Ziegler A, Wakim J, Scrivo A, Haack TB, et al. Biallelic Variants in UBA5 Reveal that Disruption of the UFM1 Cascade Can Result in Early-Onset Encephalopathy. The American Journal of Human Genetics. 2016; 99(3):695–703. https://doi.org/10.1016/j.ajhg.2016.06.030 PMID: 27545681
14. Suzuki S. Topological structural analysis of digitized binary images by border following. Computer Vision, Graphics, and Image Processing. 1985; 30(1):32–46. https://doi.org/10.1016/0734-189X(85)90016-7
15. Chetverikov D. A simple and efficient algorithm for detection of high curvature points in planar curves. In: Proc. of the 10th International Conference on Computer Analysis of Images and Patterns; 2003. p. 746–753.
16. Kuhn HW. The Hungarian method for the assignment problem. Naval Research Logistics Quarterly. 1955; 2(1–2):83–97. https://doi.org/10.1007/978-3-540-68279-0_2
17. Munkres J. Algorithms for the assignment and transportation problems. Journal of the Society for Industrial and Applied Mathematics. 1957; 5(1):32–38. https://doi.org/10.1137/0105003
18. Kunc M, Arefin B, Hyrsl P, Theopold U. Monitoring the effect of pathogenic nematodes on locomotion of Drosophila larvae. Fly. 2017. https://doi.org/10.1080/19336934.2017.1297350
19. Hirono K, Kohwi M, Clark MQ, Heckscher ES, Doe CQ. The Hunchback temporal transcription factor establishes, but is not required to maintain, early-born neuronal identity. Neural Development. 2017; 12(1):1–11. https://doi.org/10.1186/s13064-017-0078-1 PMID: 28137283
20. Risse B, Otto N, Berh D, Jiang X, Kiel M, Klämbt C. FIM^{2c}: Multicolor, Multipurpose Imaging System to Manipulate and Analyze Animal Behavior. IEEE Transactions on Biomedical Engineering. 2017; 64(3):610–620. https://doi.org/10.1109/TBME.2016.2570598 PMID: 28113210