

idtracker.ai: tracking all individuals in small or large collectives of unmarked animals

Francisco Romero-Ferrero^{1,2}, Mattia G. Bergomi^{1,2}, Robert C. Hinz¹, Francisco J. H. Heras¹ and Gonzalo G. de Polavieja^{1*}

Understanding of animal collectives is limited by the ability to track each individual. We describe an algorithm and software that extract all trajectories from video, with high identification accuracy for collectives of up to 100 individuals. idtracker.ai uses two convolutional networks: one that detects when animals touch or cross and another for animal identification. The tool is trained with a protocol that adapts to video conditions and tracking difficulty.

Researchers attempting to determine animal trajectories from video recordings face the problem of maintaining correct animal identifications after individuals touch, cross or are occluded by environmental features. To bypass this problem, we previously implemented in idTracker the idea of tracking the identification of each individual by using a set of reference images obtained from the video¹. idTracker and further developments in identification algorithms for unmarked animals^{2–6} have been successful for small groups of 2–15 individuals and in situations with few crossings^{5,7}.

Here we present idtracker.ai, a species-agnostic system able to track all individuals in both small and large collectives (up to 100 individuals) with high identification accuracy—often greater than 99.9%. A graphical user interface walks users through tracking, exploration and validation (Fig. 1a). The system uses two different convolutional networks^{8–10}, as well as algorithms for species-agnostic preprocessing, extraction of training datasets from the video and post-processing (Fig. 1b).

The preprocessing module extracts ‘blobs’, areas of each video frame that correspond either to a single animal or to several animals that are touching or crossing. The blobs are then oriented according to their axes of maximum elongation (Fig. 1c).

The convolutional ‘crossing detector’ network determines whether each preprocessed blob corresponds to a single animal or to a crossing (Fig. 1d; network architecture in Supplementary Table 1). idtracker.ai trains this network using images that a set of heuristics labels with high confidence as single animals or crossings (Methods). Once trained, the network can classify all blobs as single animals or crossings.

The convolutional identification network is then used to identify each individual between two crossings (Fig. 1e; Supplementary Table 1 outlines the network architecture). We measured the identification capacity of this network using 184 single-animal videos, with 300 pixels per animal on average. We randomly selected 3,000 images per animal for training. Tests of the network with 300 new images showed >95% single-image accuracy for up to 150 animals (Fig. 1f; Supplementary Fig. 1 shows the experimental setup, and Supplementary Fig. 2 shows results obtained with the alternative architectures detailed in Supplementary Tables 2 and 3).

By comparison, the accuracy of idTracker¹ degraded more quickly, to 83% for 30 individuals, and computationally the program is too demanding for larger groups.

With videos of collective behavior, however, we typically lacked direct access to 3,000 images per animal for training of the identification network. Hence, to obtain the training images, we developed a cascade of three protocols that were recruited sequentially depending on the difficulty of the video (Fig. 1b; Supplementary Figs. 3 and 4 show experimental setups with zebrafish and flies, and the Supplementary Notes provide details on the algorithms).

Protocol 1 first finds all intervals of the video where all the animals are detected as separate from one another. The protocol then extends the image fragments for each animal by adding the preceding and following image frames up to the previous and next crossing for each animal. We call these extended intervals global fragments; they can contain different numbers of images per animal. Subsequently, the system determines the shortest distance traveled by an individual animal within a global fragment and then chooses the global fragment in which this shortest distance is maximal across the dataset (Fig. 1g). The system then uses this global fragment to train the identification network. Once trained, the network assigns identities in all the remaining global fragments. Afterward, the system uses a set of heuristics to select those global fragments with a high-quality assignment of identities (Methods). If these high-quality global fragments (Fig. 1g) cover <99.95% of the images in the global fragments, then protocol 1 fails and protocol 2 starts.

Protocol 2 accumulates identified images in global fragments as training examples, without human intervention. It starts by retraining the network with the additional high-quality global fragments found in protocol 1. This new network then assigns the remaining global fragments, from which the system selects the high-quality ones. This procedure iterates, always converting high-quality test examples into training examples, until no more high-quality global fragments remain or until 99.95% of the images from global fragments are accumulated. Upon completion, protocol 2 can be declared successful if it finished by accumulating 99.95% of images from global fragments or if >90% of the images in global fragments had been accumulated at the point when no more high-quality global fragments were available. In our example, protocol 2 was successful at the ninth step, when it had accumulated 99.95% of images from global fragments (Fig. 1g).

Post-processing starts with assignment of the remaining images using the final network (Fig. 1h). Then, identification accuracy is estimated using a Bayesian framework that aggregates evidence from multiple individual images in each global fragment

¹Champalimaud Research, Champalimaud Center for the Unknown, Lisbon, Portugal. ²These authors contributed equally: Francisco Romero-Ferrero, Mattia G. Bergomi. *e-mail: gonzalo.polavieja@neuro.fchampalimaud.org

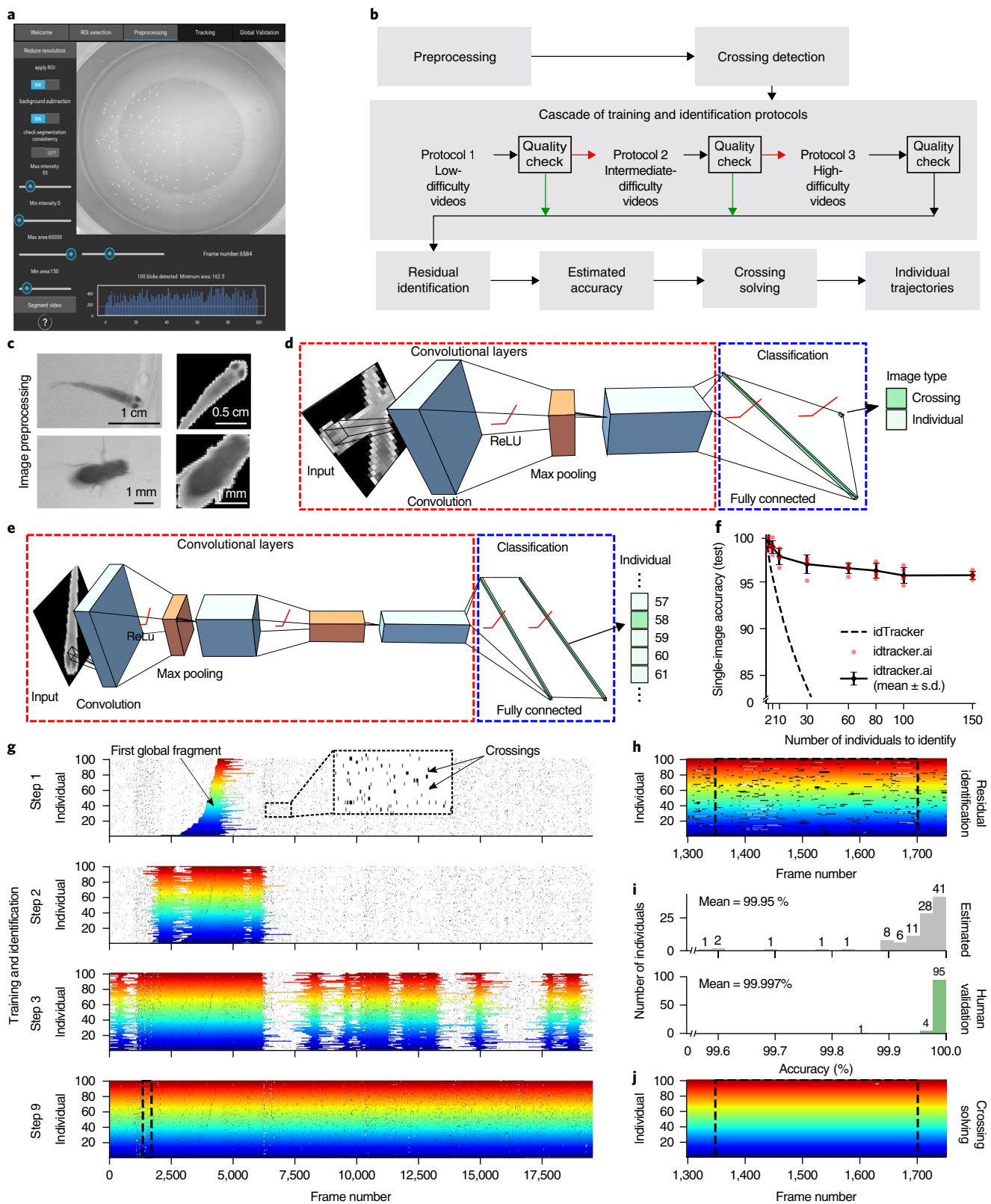


Fig. 1 | Tracking by identification in idtracker.ai. **a**, Graphical user interface. **b**, Diagram of the processing steps. **c**, Preprocessing extracts single-animal and multi-animal blobs. **d**, The crossing-detector network. ReLU, rectified linear units. **e**, The identification network. **f**, Single-image accuracy of idtracker.ai (solid line, mean \pm s.d.; red dots, single trials; $n = 5$) and idTracker¹ (dashed line). **g**, Accumulation of training images in a video of 100 zebrafish at 31 days post-fertilization. Colors indicate each of the 100 individuals, and small vertical black segments indicate animal crossings. In step 1, the identification network trains on the starting global fragment and assigns the other global fragments; then a subgroup of high-quality global fragments is extracted (step 2). Protocol 2 increases the size of the training dataset by iterating training and quality checks, here ending at step 9. **h**, Identification of remaining small segments (lighter colors). **i**, Estimated (top) and human-validated (bottom) accuracies. **j**, Post-processing assignment of crossings (black) and very small nonassigned (white) segments from the data in **h**.

(Supplementary Fig. 5). In our example, the system estimated 99.95% accuracy at this step (Fig. 1i, top); human validation of 3,000 sequential video frames (680 crossings) gave 99.997% accuracy (Fig. 1i, bottom). Animal crossings are then resolved through iterative image erosion and interpolation¹ (Fig. 1j, Supplementary Notes). The human-validated accuracy was 99.988% for the final assignments, including images between and during crossings.

If protocol 2 fails, protocol 3 starts by pretraining only the convolutional part of the identification network, using most of the global fragments. In this first step, the system does not accumulate global fragments, and the classification layer is reinitialized after training with each global fragment. Then, it proceeds in the same way as protocol 2, but training only the classification layer and fixing the parameters of the convolutional layers at the values obtained in the first step. The different stages in the protocol cascade (protocols 1–3 and post-processing) add to the accuracy and computational time (Supplementary Table 4).

We tested idtracker.ai on small and intermediate-size groups of four species (Supplementary Table 5) and on large collectives of zebrafish and flies (Supplementary Table 6). In zebrafish, protocol 2 was always successful for large groups, giving accuracies of $99.96\% \pm 0.06\%$ for 60 individuals and $99.99\% \pm 0.01\%$ for 100 individuals (mean \pm s.d.; $n=3$ for both groups). With flies, the system applied protocol 3 for groups of more than 38 individuals and reached high accuracies (99.997%) for groups of up to 72 individuals. With groups of 80–100 flies, the system reached its limit, but still had $>99.5\%$ accuracy.

We have studied potential limitations of the system. One concern is how large the global fragments need to be. We typically find 300–1,000 global fragments in 10-min videos, with the extraction of each one requiring only one frame with no crossings. Our tests have shown that the pipeline is typically successful when it starts with a global fragment containing >30 images per animal, although it can work with fewer images (Supplementary Fig. 6). Videos of large collectives of up to 100 zebrafish were found to fulfill this condition of >30 images per animal by a large margin (Supplementary Fig. 6). Videos of flies also worked with this number of images, except in recordings of very low locomotor activity acquired in a low-humidity setup (Supplementary Fig. 6, Supplementary Table 7).

A second concern is how performance depends on image quality. We recommend working with around 300 pixels per animal at the segmentation step, but our tests indicated good performance with as few as 25 pixels per animal, which corresponds to 100 pixels per animal at the identification stage owing to a dilation of segmented animals (Supplementary Fig. 7 and Supplementary Table 8). Also, the system is robust to blurring (Supplementary Fig. 8 and Supplementary Table 9), inhomogeneous lighting (Supplementary Fig. 9) and image-compression algorithms (Supplementary Table 10). A reduction of image quality typically increases the computational time (Supplementary Tables 8 and 9). Shorter computational times can be achieved through transfer learning¹¹ (Supplementary Table 11, Methods).

Finally, we illustrate the use of idtracker.ai (Fig. 2). Using an attack score (Methods) applied to two adult male zebrafish staged so as to trigger fight behavior¹² (Fig. 2a), we found the expected pattern of frequent attacks from both fish followed by one dominating (Fig. 2b, top), but also reversals of dominance (Fig. 2b, middle) and dominance of one animal from the start (Fig. 2b, bottom; Supplementary Fig. 10). idtracker.ai also tracked a group of 14 ants, *Diacamma indicum*, despite shadows, light reflections and immobile animals (Fig. 2c). Active ants activate immobile ants (Fig. 2d) in direct proportion to their level of activity (Fig. 2e; Pearson's $R^2=0.75$, two-sided Wald test $P=6 \times 10^{-5}$). We noted that 100 juvenile zebrafish formed mills (Fig. 2f). Different individuals visited the arena

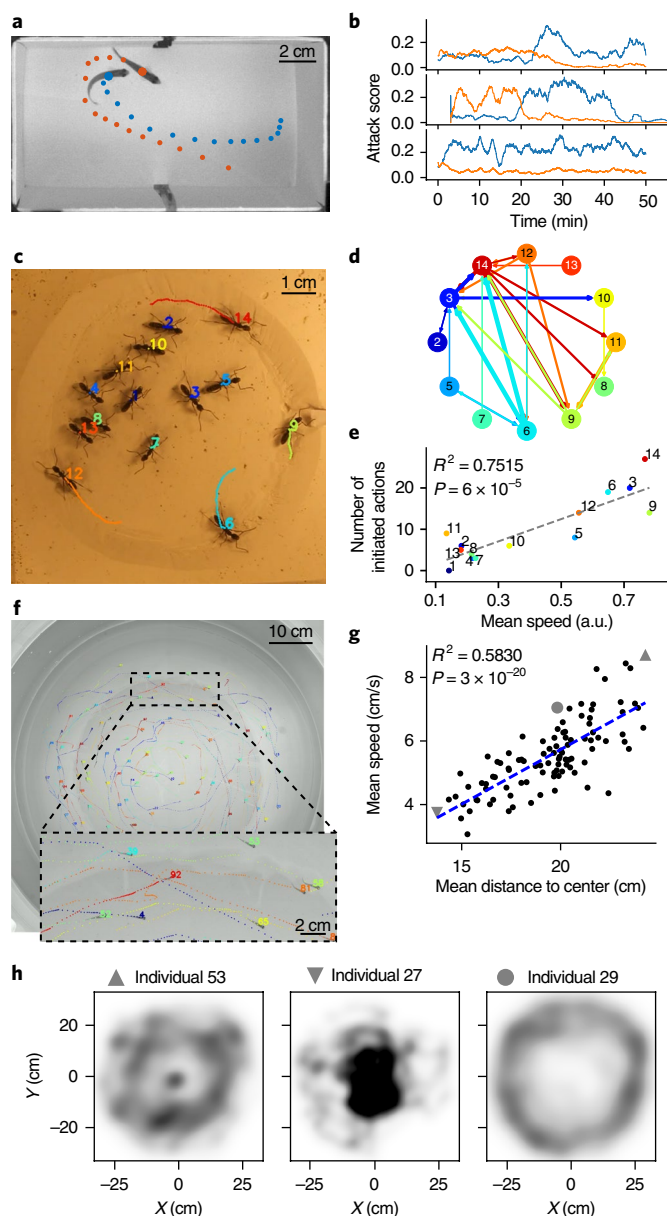


Fig. 2 | Using idtracker.ai to study small and large animal groups. **a**, Two adult male zebrafish staged so as to induce fighting behavior. Dotted lines represent small portions of their trajectories. **b**, Attack scores versus time for two individual male zebrafish. The three plots correspond to three different pairs of fish. **c**, Photo of 14 ants. Photo is a frame from a video courtesy of A. I. Bruce (Monash University, Melbourne, Australia) and N. Blüthgen (Technische Universität Darmstadt, Darmstadt, Germany) tracked with idtracker.ai. **d**, Network of ant interactions. An arrow connects two individuals if the locomotor activity of a source individual caused a response in a target individual. Line thickness represents the frequency of the interactions. **e**, Correlation of the number of locomotor initiations an animal produces with its mean speed. a.u., arbitrary units. **f**, Frame of 100 juvenile zebrafish, and a zoomed-in view of a smaller group of those fish. **g**, Correlation between mean speed and mean distance to the center of the arena. Each dot represents one of the 100 animals in the collective shown in **f**; gray symbols correspond to the individual animals references in **h**. **h**, Probability density of finding an individual in a certain position in the arena for three different fish.

differently, and those who preferred the periphery moved faster (Fig. 2g,h; Pearson's $R^2=0.58$, two-sided Wald test $P=3 \times 10^{-20}$; see Supplementary Fig. 11).

Online content

Any methods, additional references, Nature Research reporting summaries, source data, statements of data availability and associated accession codes are available at <https://doi.org/10.1038/s41592-018-0295-5>.

Received: 30 June 2018; Accepted: 26 November 2018;

Published online: 14 January 2019

References

1. Pérez-Escudero, A., Vicente-Page, J., Hinz, R. C., Arganda, S. & de Polavieja, G. G. *Nat. Methods* **11**, 743–748 (2014).
2. Dolado, R., Gimeno, E., Beltran, F. S., Quera, V. & Pertusa, J. F. *Behav. Res. Methods* **47**, 1032–1043 (2015).
3. Rasch, M. J., Shi, A. & Ji, Z. *bioRxiv* Preprint at <https://www.biorxiv.org/content/early/2016/08/24/071308> (2016).
4. Rodriguez, A., Zhang, H., Klaminder, J., Brodin, T. & Andersson, M. *Sci. Rep.* **7**, 14774 (2017).
5. Wang, S. H., Zhao, J. W. & Chen, Y. Q. *Multimed. Tools Appl.* **76**, 23679–23697 (2017).
6. Xu, Z. & Cheng, X. E. *Sci. Rep.* **7**, 42815 (2017).
7. Lecheval, V. et al. *Proc. Biol. Sci.* **285**, 1877 (2018).
8. LeCun, Y., Bengio, Y. & Hinton, G. *Nature* **521**, 436–444 (2015).
9. Abadi, M. et al. TensorFlow: large-scale machine learning on heterogeneous distributed systems. *TensorFlow.org* <http://download.tensorflow.org/paper/whitepaper2015.pdf> (2015).
10. Rusk, N. *Nat. Methods* **13**, 35 (2016).
11. Pan, S. J. et al. *IEEE Trans. Knowl. Data Eng.* **22**, 1345–1359 (2010).
12. Laan, A., Iglesias-Julios, M. & de Polavieja, G. G. *R. Soc. Open Sci.* **5**, 180679 (2018).

Acknowledgements

We thank A. Groneberg, A. Laan and A. Pérez-Escudero for discussions; J. Baúto, R. Ribeiro, P. Carriço, T. Cruz, J. Couceiro, L. Costa, A. Certal and I. Campos for assistance in software, arena design and animal husbandry; and A. Bruce (Monash University, Melbourne, Australia), N. Blüthgen (Technische Universität Darmstadt, Darmstadt, Germany), C. Ferreira, A. Laan and M. Iglesias-Julios (Champalimaud Foundation, Lisbon, Portugal) for videos of ants, flies and zebrafish fights. This study was supported by Congento LISBOA-01-0145-FEDER-022170, NVIDIA (M.G.B., F.H. and G.G.d.P.), PTDC/NEU-SCC/0948/2014 (G.G.d.P.) and Champalimaud Foundation (G.G.d.P.). F.R.-F. acknowledges an FCT PhD fellowship.

Author contributions

F.R.-F., M.G.B. and G.G.d.P. devised the project and algorithms and analyzed data. F.R.-F. and M.G.B. wrote the code with help from F.H. M.G.B. managed the code architecture and GUI. F.R.-F. managed testing procedures. R.H. built setups and conducted experiments with help from F.R.-F. G.G.d.P. supervised the project. M.G.B. wrote the supplementary material with help from F.R.-F., R.H., F.H. and G.G.d.P., and G.G.d.P. wrote the main text with help from F.R.-F., M.G.B. and F.H.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at <https://doi.org/10.1038/s41592-018-0295-5>.

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence and requests for materials should be addressed to G.G.d.P.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© The Author(s), under exclusive licence to Springer Nature America, Inc. 2019

Methods

Ethics. Animal handling and experimental procedures were approved by the Champalimaud Foundation Ethics Committee (CF internal reference 2015/007) and the Portuguese Direcção Geral Veterinária (DGAV reference 0421/000/0002016) and were performed according to European Directive 2010/63/EU¹³.

Tested computer specifications. We tracked all the videos with desktop computers running GNU/Linux Mint 18.1 64-bit (Intel Core i7-6800K or i7-7700K, 32 or 128 GB RAM, Titan X or GTX 1080 Ti GPUs, and 1 TB SSD disk). Videos can also be tracked using CPUs, with longer computational times.

Animal rearing and handling. For zebrafish videos we used the wild-type TU strain at 31 days post-fertilization (dpf). Animals were kept in 8-liter holding tanks at a density of ten fish per liter and with a 14-h light/10-h dark cycle in the main fish facility. For each experiment, a holding tank with the necessary number of fish was transported to the experimental room, where fish were carefully transferred to the experimental arena with a standard fish net appropriate for their age.

For the fruit fly videos, we used adults from the Canton S wild-type strain at 2–4 d post-eclosion. Animals were reared on a standard fly medium and kept on a 12-h light/dark cycle at 28°C. We placed flies in the arena either by anesthetizing them with CO₂ or ice, or by using a suction tube. We found the latter method to have the least negative effect on the flies' health and to result in better activity levels.

Experimental setups. *Zebrafish video setup.* The main tank was placed inside a box built with matte white acrylic walls (Supplementary Fig. 3a). The lighting was based on infrared and RGB LED strips. A cylindrical retractable light diffuser made of plastic ensured homogeneous illumination in the central part of the main tank. A 20 MP monochrome camera (Emergent Vision HT-20000M) with a 28-mm lens (ZEISS Distagon T* 28-mm f/2.0 Lens with ZF.2) was positioned approximately 70 cm from the surface of the arena. To prevent reflections of the room ceiling, we used black fabric to cover the top of the box (Supplementary Fig. 3b). We used this setup to record zebrafish in groups and in isolation. Videos of groups of 10, 60 and 100 fish were recorded in a custom-made one-piece circular tank of 70-cm diameter, designed in-house. The tank was filled with fish system water to a depth of 2.5 cm. The circular tank was held in contact with the water of the main tank approximately 10 cm above a white background to improve the contrast between animals and background (Supplementary Fig. 3c). A water-recirculating system equipped with a filter and a chiller ensured a constant water temperature of 28°C.

Fruit fly video setup. The setup was placed in a dedicated experimental room with controlled humidity (60%) and temperature (25°C). RGB and IR LEDs placed on a ring around a cylindrical light diffuser guaranteed homogeneous light conditions in the central part of the setup (Supplementary Fig. 3a). Videos were recorded with the same camera as in the zebrafish setup. Black cardboard around the camera reduced reflections of the ceiling on the glass covering the arena (Supplementary Fig. 3b). We used two different arenas made of transparent acrylic, both built to prevent animals from walking on the walls. Arena 1 (diameter, 19 cm; height, 3 mm) had vertical walls that were heated with a white insulated resistance wire (Pelican Wire Company; 28 AWG solid (0.0126 inch), Nichrome 60, 4.4 Ω/ft, 0.015-inch white TFE tape). At 10 V, 0.3 A, the temperature at the walls reached 37°C. Arena 2 (diameter, 19 cm; height, 3.4 mm) had conical walls (angle of inclination, 11°; width of conical ring, 18 mm) (Supplementary Fig. 3c). The best results were obtained with standard top-view recording (Supplementary Table 5). Arena 1 was also used for bottom-view recordings. The top of the arena was a sheet of glass covered with Sigmacote SL2 (Sigma-Aldrich), which prevented the flies from walking on the ceiling. A white plastic sheet was placed below the arena to increase the contrast between flies and background, at a distance of 5 cm below the arena to avoid shadows. To move flies into the arena, we either anesthetized them with CO₂ or ice, or used a suction tube. We found the latter method to have the least negative effect on the flies' health as evidenced by their activity levels.

Individual image dataset. *Individual image dataset setup.* We recorded 184 juvenile zebrafish (TU strain, 31 dpf) in separate chambers (60-mm-diameter Petri dishes). A holding grid with transparent acrylic walls allowed equal spacing between arenas while granting visual access to the neighboring dishes (Supplementary Fig. 2a). To increase image contrast, we used a white acrylic floor placed 5 cm below the holding grid, which acted as a light diffuser to prevent shadows. Four individuals at a time were recorded for 10 min (Supplementary Fig. 2b). On the outer borders we placed additional dishes containing fish to act as social stimuli (Supplementary Fig. 2b). This made the recorded fish swim more than they would have in isolation. From the 46 videos recorded, individual images were labeled according to the individual they represented. Each image was preprocessed according to the procedure detailed in the Supplementary Notes and then cropped as a square image for use in testing the identification network (image size, 52 × 52 pixels; Supplementary Fig. 2c). The dataset comprised a total of ~3,312,000 uncompressed, grayscale, labeled images.

Statistics and reproducibility. Results similar to the ones in Fig. 1c,g–j were obtained for $n = 32$ independent experiments tracked with idtracker.ai. Supplementary Tables 5–7 show the results of every experiment, and Supplementary Fig. 5 shows a comparison of the estimated accuracy and the accuracy after human validation.

In total, we performed $n = 13$ independent fish-fight experiments with different animals. We obtained results similar to those presented in Fig. 2a,b (Supplementary Fig. 10).

We tracked $n = 1$ video of ants (Fig. 2c–e) to illustrate the kind of graph analysis that can be performed with the trajectories obtained with idtracker.ai. We tracked $n = 3$ different videos of 100 juvenile zebrafish. We obtained results similar to those shown in Fig. 2f–h for the other two videos (Supplementary Fig. 11).

In Fig. 2e,g, R^2 was computed as the square of the Pearson's r correlation coefficient, and P is the two-sided P value for a hypothesis test whose null hypothesis is that the slope is zero, obtained by Wald test with t -distribution of the test statistic.

Artificial neural network details. *Architectures.* The crossing-detector network (Fig. 1d) is a convolutional neural network^{8,10}. It has two convolutional layers that obtain from data a relevant hierarchy of filters. A hidden layer of 100 neurons then transforms the convolutional output into a classification of 'single animal' or 'crossing'. idtracker.ai trains this network using images that can be confidently characterized as single or multiple animals (i.e., single animals as blobs consistent with single-animal statistics, and not split into more blobs in the past or future). Further details of the architecture are given in Supplementary Table 1.

The architecture of the identification network (Fig. 1e) consists of three convolutional layers, a hidden layer of 100 neurons and a classification layer with as many classes as animals in the collective. Further details are given in Supplementary Table 1. We tested variations of the architecture by modifying either the number of convolutional layers (Supplementary Table 2) or the number of hidden layer neurons (Supplementary Table 3). Analysis of these networks indicated that the most important requirement for successful identification is that the convolutional part have at least two layers (Supplementary Fig. 1).

The GUI allows users to modify the architecture of this network and its training hyperparameters.

Training. The convolutional and fully connected layers of both networks are initialized via Xavier initialization¹⁴. Biases are initialized to 0.

The deep crossing-detector network is trained using the algorithm and hyperparameters described by Kingma and Ba¹⁵. The learning rate is set at the initial value of 0.005. This network is trained in mini batches of 100 images.

The identification network is trained using stochastic gradient descent, setting the learning rate to 0.005. This network is trained in mini batches of 500 images. In the training set, every image is duplicated via 180° rotation because the preprocessed images can have the head either in the upper-right corner or in the lower-left corner. Further details are given in the Supplementary Notes.

Overfitting was prevented by training with early stopping¹⁶, that is, training the model until the error in the validation dataset reached a minimum. It is possible to use dropout¹⁷ by modifying the GUT's advanced settings. These settings also allow transfer learning¹¹ using a network previously trained on other experiments.

Heuristics. *Classification of individual and crossing blobs.* The dataset used to train the convolutional crossing detector is extracted automatically from the video, following two heuristics that use properties of the blobs obtained during the preprocessing.

A blob b is a collection of connected pixels in a given frame that do not belong to the background. First, a model of the area of individual blobs is constructed, considering the median number of pixels, m_b , and the s.d., σ_b , of the blobs in parts of the videos where the number of blobs corresponds to the number of animals declared by the user. A blob is considered to be a potential individual if its number of pixels differs from m_b by less than $4\sigma_b$; otherwise it is categorized as a potential crossing.

Second, we consider the overlap of blobs in subsequent frames. We say that two blobs overlap in consecutive frames if the intersection of the set of pixels defining every blob is not empty. We classify the image corresponding to the blob b as an individual if (i) b is an individual as defined by the first heuristic, (ii) b overlaps in the immediately previous and subsequent frames with only one blob, and (iii) every blob overlapping in the past and future of b overlaps with at least one blob. We classify the images corresponding to the blob b as crossings if (i) b is a crossing as defined by the first heuristic and (ii) b overlaps with more than one blob in the past or in the future. Or, (i) b is a crossing as defined by the first heuristic and (ii) any blob overlapping in the past or in the future with b overlaps with more than one blob. The Supplementary Notes include a formal definition of the heuristics.

High-quality global fragment extraction. During the protocol cascade, we use the identification network to assign all the global fragments whose images were not in the training dataset. Then, from these global fragments the algorithm extracts those with high quality, which will form part of the next training set. A global fragment is defined as being of high quality if it fulfills the three following

heuristics: First, the identity of all the individual fragments in the global fragment must be certain enough. Second, the identity of all the individual fragments in the global fragment must be consistent with the identity already assigned. Third, all the identities assigned to the individual fragments in the global fragment are unique. The Supplementary Notes include a formal definition of the heuristics.

Image quality conditions. *Resolution.* To test the performance of the system as a function of the number of pixels per animal, we artificially reduced the resolution of every frame, resizing it by a factor $\rho = [0.75, 0.5, 0.35, 0.25, 0.15]$. We reduced Moire effects by resampling using pixel area relation¹⁸.

Blurring. To test the performance of the system under different levels of image blurring, we artificially blurred every frame, using a Gaussian kernel with s.d. $\sigma = [0.5, 1, 2, 3, 4, 5]$. The kernel size was computed automatically from the s.d.¹⁸.

Compression. To test the robustness of the system under the effects of compression algorithms, we encoded the raw videos using the MPEG-4 and H.264 video codecs. Videos were encoded using FFmpeg. We encoded MPEG-4 by setting the FFmpeg's parameter qscale to 1. For the H.264 video codec, we used a constant rate factor of 0, which corresponds to a lossless compression.

Inhomogeneous light conditions. To test the robustness of the system under inhomogeneous light conditions, we switched off the infrared LEDs in two of the four walls and added a black cloth covering half of the cylindrical light diffuser (Supplementary Fig. 3). The videos recorded with the setups described in Supplementary Figs. 3 and 4 were obtained with IR illumination. Recording with this illumination allowed us to use any patterns of visible light that might have been needed in the experiment.

Transfer learning. The transfer-learning¹¹ technique can be applied at the identification stage to reuse knowledge from a network previously trained with similar animals and light conditions. In the advanced settings of the GUI, the user can apply transfer learning and decide whether to train the whole identification network or only the last two fully connected layers. We trained only the last two fully connected layers of the identification network when we applied this method.

Parameters. idtracker.ai needs the user to input parameters only for the segmentation of animals from the background. Users typically need to input only three parameters: a maximum intensity to separate animals from background, a minimum area to discard small objects during segmentation, and the number of animals to be tracked. In some cases, the user might need to input the minimum intensity or the maximum area, but we find this to be uncommon. The system then computes the number of animals in the video and asks the user for confirmation. There is also a set of advanced parameters that are divided in two main categories. It is possible to refine the preprocessing as detailed in <http://idtracker.ai/quickstart.html>, and to modify many of the identification network hyperparameters. By acting directly from the GUI, the user can define the presence and amount of dropout in the fully connected layers, choose an optimizer or choose one of the architectures described in Supplementary Tables 2 and 3. A detailed description of each parameter is provided in Supplementary Table 12 and in the explanation of the algorithm in the Supplementary Notes.

Analysis of tracks. *Attack score in zebrafish fights.* For each frame, we considered one focal animal to be attacking the other if its speed was greater than 1.5 body lengths (BL) per second, the other animal was positioned within an angle of $\pm 45^\circ$ with respect to the focal animal's direction of movement, and the distance between them was less than 2 BL. We then calculated the attack score at time t as the fraction of frames the focal animal spent attacking the other in the window $t \pm 1$ min.

Interaction network in ants. We represent the interaction network of a collective as a directed graph. An arrow between nodes i and k indicates that, during the video, individual i triggered at least a locomotor response in k , and the thickness of that arrow is proportional to the number of triggered locomotor responses.

To detect interactions among individuals, we first computed an activity time series for each of them. We obtained the speed of each individual as the first derivative of the respective trajectory, which we smoothed in time with a moving

average of window size w_s . The Hilbert envelope¹⁹ of the smoothed individual speeds accentuated ramps in the speed time series. Finally, by applying the softmax function, we obtained the frame-wise probability mass function of an individual being active with respect to the collective as a measure of activity for each animal, $a_i(t)$. We then computed the local maxima of each $a_i(t)$. Two subsequent maxima in the activity of an individual are acceptable if they are at least w_s frames apart. Let us consider the i th individual and call the frames corresponding to the local maxima of its activity, $M_i = \{m_{i,1}, \dots, m_{i,n}\}$. For each acceptable local maximum $m_{i,j}$, we considered the activity of the other individuals in the frame interval $[m_{i,j}, m_{i,j} + w_i]$, where w_i is the windows of frame in which the activity of an individual can be triggered by the focal individual. So, if another individual, say, k , reaches maximum activity within this interval and the distance between individuals i and k at frame $m_{i,j}$ is smaller than a fixed radius r , we say that i triggered a locomotor response in k . We used $w_s = 59$ frames (1 s). $w_i = w_s$, and $r = 3$ BL.

Location and average speed in milling groups. We recorded and tracked three groups of 100 juvenile zebrafish while they were milling in a circular tank (Supplementary Fig. 3 presents the details of the setup). The trajectories were smoothed using a Gaussian kernel with an s.d. of two frames. We estimated the location of the center of the tank as the average center of mass over all the animals along the video. For every frame we computed the distance from each individual to the center of the tank. We used kernel density estimation²⁰ to estimate the probability density function of the location in the tank of three representative individuals: the one with highest average distance to the center in the video, the one with the smallest average distance to the center in the video, and a third individual with an intermediate average distance to the center (Fig. 2g).

The speed was computed as the norm of the velocity vector. Using standard linear regression analysis, we computed the correlation across individuals between the average distance to the center of the tank and the average speed (Fig. 2h).

Reporting Summary. Further information on experimental design is available in the Nature Research Reporting Summary linked to this article.

Code availability

idtracker.ai is open-source and free software (license GPL v.3). The source code and the instructions for its installation are available at https://gitlab.com/polavieja_lab/idtrackerai. A quick-start user guide and a detailed explanation of the GUI can be found at <http://idtracker.ai/>. The software is also provided as Supplementary Software.

Data availability

Processed data that can be used to reproduce all figures and tables can be found at <http://idtracker.ai/>. Lossless compressed videos can be downloaded from the same page. Raw videos are available from the corresponding author upon reasonable request. A library of single-individual zebrafish images for use in testing identification methods also can be found at <http://idtracker.ai/>. Two example videos, one of 8 adult zebrafish and one of 100 juvenile zebrafish, are also included as part of the quick-start user guide.

References

- Martins, S. et al. *Zebrafish* **13**, S47–S55 (2016).
- Glorot, X. & Bengio, Y. in *Proc. Thirteenth International Conference on Artificial Intelligence and Statistics* (eds Teh, Y. W. & Titterton, M.) 249–256 (PMLR, Sardinia, Italy, 2010).
- Kingma, D. & Ba, J. *arXiv Preprint* at <https://arxiv.org/abs/1412.6980> (2015).
- Morgan, N. & Bourlard, H. in *Advances in Neural Information Processing Systems 2* (ed Touretzky, D. S.) 630–637 (Morgan Kaufmann, San Francisco, 1990).
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).
- Bradski, G. *Dr. Dobbs Journal* **25**, 120–123 (2000).
- Oppenheim, A. V. & Schaffer, R. W. *Discrete-time Signal Processing* (Pearson, Upper Saddle River, NJ, 2014).
- Scott, D. W. *Multivariate Density Estimation: Theory, Practice, and Visualization* (John Wiley & Sons, Hoboken, NJ, 2015).

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see [Authors & Referees](#) and the [Editorial Policy Checklist](#).

Statistical parameters

When statistical analyses are reported, confirm that the following items are present in the relevant location (e.g. figure legend, table legend, main text, or Methods section).

n/a Confirmed

- ☐ ☒ The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
- ☐ ☒ An indication of whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
- ☒ ☐ The statistical test(s) used AND whether they are one- or two-sided
Only common tests should be described solely by name; describe more complex techniques in the Methods section.
- ☒ ☐ A description of all covariates tested
- ☒ ☐ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
- ☐ ☒ A full description of the statistics including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
- ☒ ☐ For null hypothesis testing, the test statistic (e.g. F , t , r) with confidence intervals, effect sizes, degrees of freedom and P value noted
Give P values as exact values whenever suitable.
- ☒ ☐ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
- ☒ ☐ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
- ☒ ☐ Estimates of effect sizes (e.g. Cohen's d , Pearson's r), indicating how they were calculated
- ☐ ☒ Clearly defined error bars
State explicitly what error bars represent (e.g. SD, SE, CI)

Our web collection on [statistics for biologists](#) may be useful.

Software and code

Policy information about [availability of computer code](#)

Data collection

To acquire the videos we used StreamPix 7 or the open source software BONSAI (<https://bitbucket.org/horizongir/bonsai>). Any software to acquire videos that meets our requirements in the section "General Video Conditions" in the Supplementary Material is valid.

Data analysis

Videos were tracked with idtracker.ai. idtracker.ai is open-source and free software and can be found at the GitLab repository link provided in the Methods section in the manuscript, https://gitlab.com/polavieja_lab/idtrackerai. All the code used to analyze the data is custom made and open-source and can be found at the GitLab repository https://gitlab.com/polavieja_lab/idtrackerai_scripts.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers upon request. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research [guidelines for submitting code & software](#) for further information.

Data

Policy information about [availability of data](#)

All manuscripts must include a [data availability statement](#). This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

Processed data to reproduce all figures and tables can be found at <http://idtracker.ai/>. Lossless compressed videos can be downloaded from the same page. Raw videos are available upon request. A library of single-individual images of zebrafish to test identification methods can be found in the same link. Two example videos, one of 8 adult zebrafish and another of 100 juvenile zebrafish, are also included as part of the quick-start user guide.

Field-specific reporting

Please select the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see [nature.com/authors/policies/ReportingSummary-flat.pdf](https://www.nature.com/authors/policies/ReportingSummary-flat.pdf)

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size

We performed no sample size calculation: we intended to prove that the accuracy of the system was consistently high. idtracker.ai was first developed using in 2 videos until their accuracy was very high (marked with * in Supplementary Table 5 and 6). Then, we tested that this high accuracy was consistent, using 6 videos of large groups of zebrafish and 6 videos of fruitflies, with accuracies always above 99.4%. To show that performance of the system in small groups was also consistently high, we recorded 3 videos of 10 zebrafish, and tracked all of them with high accuracies (>99.99). Accuracy when tracking videos in [1] was similar in idtracker.ai and idTracker (13 videos). All together, they form the largest dataset to test multi-animal tracking. As all accuracies were proved to be consistently high in our recorded videos (>99.4%, N=15 new videos), we did not the need to increase the sample size. To study the limits of the system, we decided to systematically modify the number of pixels per animal and the blurring until the accuracy was below 99%. This gave us 5 videos of different number of pixels per animal for a group size of 8 animals and a group size of 60; and 6 videos with different blurring conditions for 8 animals and another 6 for 60 animals.

Data exclusions

We did not exclude any data.

Replication

All attempts at replication were successful.

Randomization

All animal groups considered were selected randomly from a larger pool.

Blinding

To the assess the performance of the tracking system it is not necessary to make comparisons between groups. Therefore, there was no need for any group allocation procedures. As stated above, only randomization is needed to avoid systematic use of the same animals across videos. Analysis biases were reduced by following standard procedures in machine learning, namely: dividing the dataset in three parts, for training, validation and test.

Reporting for specific materials, systems and methods

Materials & experimental systems

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> Unique biological materials
<input checked="" type="checkbox"/>	<input type="checkbox"/> Antibodies
<input checked="" type="checkbox"/>	<input type="checkbox"/> Eukaryotic cell lines
<input checked="" type="checkbox"/>	<input type="checkbox"/> Palaeontology
<input type="checkbox"/>	<input checked="" type="checkbox"/> Animals and other organisms
<input checked="" type="checkbox"/>	<input type="checkbox"/> Human research participants

Methods

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> ChIP-seq
<input checked="" type="checkbox"/>	<input type="checkbox"/> Flow cytometry
<input checked="" type="checkbox"/>	<input type="checkbox"/> MRI-based neuroimaging

Animals and other organisms

Policy information about [studies involving animals](#); [ARRIVE guidelines](#) recommended for reporting animal research

Laboratory animals

For zebrafish (*D. rerio*) videos we used the wild-type TU strain at 31 days post fertilization (dpf). For the fruit fly (*D. melanogaster*) videos we used adults (males and females) from the Canton S wild-type strain at 2-4 days post-eclosion.

Wild animals

The experiments we performed in this study did not involve wild animals.

Field-collected samples

The experiments we performed in this study did not involve field-collected samples.