Title: Ubitrail: A robust and open-source software for tracking insect locomotion

Authors: Joe D. Gallagher[*1], Michael T. Siva-Jothy[1], and Quentin Geissmann[2]

Author affiliations:

[1] *Department of Animal and Plant Sciences, University of Sheffield, S10 2TN, U.K.*

[2] *Department of Life Sciences, Imperial College, South Kensington, London, SW7 2AZ, U.K.*

[*] Corresponding author: joedgallagher@gmail.com

Abstract

Analysing the locomotor behavior of animals is essential to a diverse range of studies, from learning and memory to reproduction and immunity. Automated tracking systems can provide a high-throughput method of obtaining quantitative behavioural data, but many of the currently available options are inflexible, difficult to use, or expensive. We present an open-source software ('Ubitrail') for tracking the individual movement of insects, and a statistical package ('Rubitrail') for extracting and analysing behavioural metrics of interest. The software is designed to be versatile with regards to species morphology and experimental design and robust to imperfect conditions of video capture, as well as inexpensive and straightforward to use. We provide a demonstration of its capabilities by using it to detect an effect of immune challenge upon locomotion in the mealworm beetle, *Tenebrio molitor*. Finally, we investigate the biological significance of relationships between our extracted behavioural metrics by comparing endogenous beetle locomotion to computer-generated simulations of locomotion.

Keywords: locomotory analysis; video tracking; *Tenebrio molitor;* sickness behaviours

1. Introduction

Locomotor activity impacts almost all aspects of a mobile animal's ecology. Movement underpins key fitness-driving traits, such as foraging, mating, courtship, learning processes, and immunity (Martin, 2004). Many of the most commonly measured behaviours in animals, such as ambulation, freezing (resting immobile), jumping, vectorial information (speed, acceleration), positional information (e.g. site preference, orientation angle) – as well as psychological measures typically considered in only vertebrate studies, such as anxiety, obsession and aggression – are emergent from tracking the movement vectors of an individual, i.e. the organism's spatial coordinates over time.

The quantification of the complex movement patterns of mobile organisms has become an integral subject in biological research, and has been facilitated by recent advances in automated tracking methods. Automated systems provide a much higher throughput than manual methods, and tend to be more reliable due to the consistency of a processing algorithm, which does not suffer observer fatigue or drift (Noldus *et al.*, 2001). Digital methods can also yield behavioural metrics that would be difficult or impossible to quantify manually, such as velocity, acceleration and turning angle, as well as calculating time and spatial location with a high degree of accuracy.

As computing capabilities have increased and costs decreased over the last decade, automated tracking systems have progressed from rudimentary and often unreliable analogue systems which described mostly discrete behaviours, to digitised methods which can detail an array of continuous kinematic variables. Advancements in computer-vision capabilities and the increasing availability and support of open-source libraries, such as OpenCV (Willowgarage, http://opencv.willowgarage.com), are creating a new and accessible ecosystem of highly customisable and affordable tools for biologists to study the behaviour of a much larger variety of organisms. Several digitiser-based video tracking systems are commercially available, but there are drawbacks with many of these.

Firstly, many trackers are developed only for the most well-studied model organisms (e.g. *Drosophila* [Gomez-Marin *et al.*, 2012, Dankert *et al.*, 2009], *C. elegans* [Swierczek, *et al.*, 2011], zebrafish [Beyan & Fisher, 2013], mice [de Chaumont *et al.*, 2012]), being highly tailored to the particular morphology and movement patterns of their target species. Secondly, many of these programs are designed to address specific behavioural paradigms, and thus offer little flexibility, often requiring specialised apparatus or the use of highly specific experimental designs. Thirdly, much available tracking software is proprietary and requires a substantial up-front fee, as well as continuing license fees in some cases (e.g. EthoVision [Noldus *et al.*, 2001], ANY-maze [http://anymaze.com], GroupScan, LoliTrack, PhenoTracker). Of the open-source options available, several use proprietary toolboxes to parse output files (containing the tracked X,Y-coordinates), most notably MATLAB (e.g. Ctrax [Branson *et al.*, 2012], CADABRA, idTracker, Motr, SOS-track). Finally, several open-source software are largely inflexible and onerous to modify, necessitating in-depth knowledge of the relevant programming language as well as a substantial investment of time (e.g. Flydra), or are currently unstable or non-functional due to a lack of maintenance (e.g. SwisTrack [Lochmatter *et al.*, 2008], MotMot [Straw & Dickinson, 2009]).

In this paper, we describe an open-source software, 'Ubitrail', for the automated tracking of insect locomotion and a statistical package, 'Rubitrail', for extracting behavioural metrics. The platform is intended to be: (i) versatile, working with a range of morphologically and ethologically distinct insect species and within a range of non-specialised experimental set-ups; (ii) robust, working under imperfect lighting conditions and handling insect occlusion and other experimental imperfections which can lower accuracy; (iii) simple, with a simple graphical user interface offered for tracking and statistical functions allowing for straightforward analysis straight 'out of the box'; and (iv) affordable, making use of exclusively open-source software and relying on inexpensive hardware. We provide a proof of concept by quantifying changes in locomotion following immune challenge in the mealworm beetle, *Tenebrio molitor*, in order to investigate sickness behaviours. By

comparing this data with computer-generated simulations of locomotion, we help separate meaningful biological relationships between behavioural metrics from mathematical correlations in order to validate the behavioural metrics defined.

2. Methods

2.1. Description of the system

The tracking software, UbiTrail, was written in C++ using the OpenCV library (Willowgarage, http://opencv.willowgarage.com), and under the CodeBlocks design environment (http://codeblocks.org). The software source code and compilers for Unix and Microsoft Windows operating systems are freely available online (http://sourceforge.net/projects/ubitrail), as is the associated R package, Rubitrail, as well as a user manual, sample videos and sample data.

The details of the image analysis process are described in detail in Figure 1. In brief, the software uses a dynamic learning algorithm to learn to identify moving foreground objects during an initial training period (default value of 500 frames, ~25 seconds). In order to solve ambiguities in foreground detection, a likelihood model is built on the fly, based upon several key features of known foreground, including contour shape, pixel colour and distance between current contour and last detected contour, with the most likely single contour being identified as foreground.

2.2. Using the software

UbiTrail currently works with digital video files as input, which can be recorded using an inexpensive USB video camera (any webcam with a resolution of at least 640x480 pixels is suitable) and are easily captured using the open-source multimedia player, VLC (VideoLAN, http://videolan.org/vlc).

After recording a video, the user is able to define a mask to denote the position of areas within the arena, as well as sub-territories within individual areas, if desired. The user is then able to adjust several processing parameters in order to optimise tracking accuracy, such as sensitivity (which determines how likely noise is to be detected as motion) and the number of frames used to train the motion detector.

The software can be implemented either via the command line or using a graphical user interface (GUI). The GUI is a simple assistant which allows the user to interactively define the inputs and output options, preview the defined mask over the video, and visualise the actual tracking process on-the-fly (Figure 2). Command line usage can increase efficiency by allowing the user to iteratively analyse multiple videos without the need for continual input.

The software outputs a CSV file containing an X,Y coordinate, timestamp, area ID and territory ID (if applicable) for each detected object in each frame of a video. Also included is a header containing metainformation, such as name, duration, and number of frames per second, as well as the X,Y-coordinates of each detected area. Video files of the tracking process can also be optionally returned, either as a single video of the global arena or as separate videos for each individual area.

2.3. Rubitrail analysis package

The analysis software, Rubitrail, is a package written for R (R Core Development Team, http://r-project.org). The package extracts multiple features from the raw data outputted by the tracking software, including velocity, turning angles, activity levels and positional information, as well as allowing the user to define their own additional variables for analysis. Whilst all scripts within the package are fully customisable, a single master function is included to aid user accessibility, requiring as input only a list of CSV files for analysis and a scale calibration (pixels/mm).

2.3.1. Pre-processing data

2.3.1.1. Undistortion

Fisheye lenses and low-cost wide-angle lenses can produce a significant degree of barrel distortion in the images they capture, having the potential to impact the validity of detected movements in a tracking software (Figure 3). This can be corrected using a simple algorithmic transformation:

where **r** is the distance of a given pixel to the centre of the uncorrected image and **R** is the distance of the pixel in the corrected image. This transformation can either be applied before tracking analysis by transforming each image frame of the raw video, or after tracking by transforming the detected X,Y-coordinates; Rubitrail utilises the latter method. Ready-made parameter sets for particular cameras can be found online (e.g. http://sourceforge.net/projects/hugin/files/PTLens%20Database), or can be calculated manually by taking a calibration image (see Figure 3) and identifying the most suitable values using the undistortion feature available in a number of image manipulation programs (e.g. ImageMagic [http://imagemagick.org]).

2.3.1.2. Linear interpolation

In frames where insects are occluded by obstacles or glare, or a contour is otherwise not found, X,Y position is inferred using linear interpolation. X,Y-coordinates are not inferred for training frames, where the insect has yet to be detected. In instances where no movement is detected throughout the entire video, a velocity of zero is inferred for all frames whilst all other metrics regarding positional information are defined as NA. In cases where movement is not detected in the final frames of a video (e.g. the insect does not move in the final two minutes of analysis), the X,Y-coordinates for the remaining frames are inferred as the last confirmed location of the insect.

2.3.1.3. Trajectory smoothing

Camera noise, lighting abnormalities, non-locomotory insect movements (e.g. grooming, antennation), and imperfections in foreground segmentation can cause false movements to be identified, increasing the noise in detected X,Y trajectories. Furthermore, lateral oscillation in the detection of moving objects is common, which may be due to alternated movement between the posterior and anterior of an insect between frames (Hen *et al.*, 2004). Both of these factors are manifest in the tracked coordinates as a relatively small jitter, with perturbations no larger than the maximum length of the tracked insect. Two steps were taken to correct for this noise.

Firstly, trajectories were smoothed by using a simple moving median (e.g. Hen et al., 2004) with a window size of 20 data points (1 s) a 1 point step size (0.05 s), which allowed maximum overlap between smoothing windows. These values were found to preserves overall trajectory information and provide greater accuracy in determining velocity, turning angles and overall activity level (Figure 4). Secondly, due to the size of the smoothing window, insects were often falsely determined to have negligible, but non-zero, velocity (> 0 mm/s). A movement threshold was thus implemented to filter such negligible movements from the smoothed velocity data, with near-zero velocities of < 1 mm/s (~2 pixels/s; a value used in similar tracking software [Valente *et al.*, 2007; Robie *et al.*, 2010; Colomb *et al.*, 2012]) being redefined as zero velocity (=0mm/s) (Figure 5).

2.3.2. Extracting metrics

Several key metrics to describe insect locomotion were defined and extracted from the smoothed trajectory data.

2.3.2.1. Velocity metrics

Distance moved was calculated as the Pythagorean distance between smoothed X,Y-coordinates in

successive frames. Summing each movement length over the entire analysis yielded the total distance travelled (mm). Dividing distance travelled per frame by time elapsed gave instant velocity (mm/s), and the first derivative of instant velocity was used to define acceleration (mm/s^2).

2.3.2.2. Angular metrics

Turning angle was calculated as the angle between successive velocity vectors (Figure 6). Considering the movement from P_0 to P_1, α_0 is the absolute movement angle, the turning angle, γ, can be calculated as $\alpha_0 - \alpha_1$. Movement paths of walking insects are generally continuous, and do not have discrete break points that make it easy to define moves; a common solution is to resample movement at regular time intervals and connect successive positions with linear interpolation. Turning angles were therefore calculated from smoothed data which was down-sampled to a rate of 1 frame per second (Figure 6). Meander is a measure of movement tortuosity which combines turning angle with distance travelled, and increased meander has been associated with navigational uncertainty (Collins *et al.*, 1994). Meander was calculated by dividing the turning angle by the instantaneous velocity (θ * mm/s) (Martin *et al.*, 2004). Many animals show a tendency to turn around an arena (Yaski *et al.*, 2011), a behaviour which is often interpreted as an escape response. Escape responses are well-studied in cockroaches, which rapidly turn directly away (180°) from threatening stimuli, such as a puff of wind, and accelerate away (Domenici *et al.*, 2008). A similar response is observed in *Tenebrio molitor* (pers. obs.), although this behaviour may equally be representative of roving behaviour or foraging activity, as opposed to an anti-predation or stress response. Turnaround events were defined as turns of 180° ± 25° which were completed within the space of one second (example highlighted in Figure 6e).

2.3.2.3. Activity metrics

Run length encoding (RLE) was used to temporally smooth velocity in order to derive activity

metrics, allowing identification of stationary and mobile phases. RLE is a form of data compression which identifies patterns in consecutive sequences (runs) of data. For example, a binary sequence of characters, <AAAAABBABBB>, may be run length encoded as, <5A, 2B, 1A, 3B>. Here, information on mobility was calculated by run length encoding smoothed and thresholded velocity data to determine whether movement speeds were above or below a user-defined velocity threshold (1 mm/s) (Figure 7). Owing to noise between frames in detected velocities, a sliding window of 3 s was used to classify movement transitions (see Figure 8); i.e. when velocity was above the defined threshold (>=1 mm/s) threshold for a period of >=3 s, the insect entered a movement phase, and when its speed fell below 1mm/s for a period of >3 s, the insect entered a stationary phase. The absolute number of phases transitions and mean duration of mobile and stationary phases was calculated for each insect.

2.3.2.4. Spatial analysis

'Heat maps' can be outputted to provide a fast and intuitive overview of an insect's location during the course of the experiment (Figure 9). Two metrics, thigmotaxis and exploration, were also developed in order to quantify the amount of time spent in certain zones of the arena. Thigmotaxis is the tendency of many animals to remain in the perimeter of an arena during open-field experiments, moving in the peripheral areas where they can maintain physical contact with the walls of the arena and avoiding central zones whose open space may leave the insect more vulnerable to predation (e.g. Gotz & Biesinger, 1985; Colomb *et al.*, 2012). Exploratory (or roving) behaviour is often defined in vertebrates alongside such metrics as shyness/boldness, aggression and neophobia (Dingemanse *et al.*, 2002), but may be more simply defined as the proportion of the environment traversed.

In order to normalise the spatial locations for each arena, a minimum enclosing circle was fitted to each arena to determine its exact boundaries, before tracked Cartesian coordinates (x,y) for

each arena were converted to polar coordinates (r,θ). To quantify thigmotaxis, each defined minimum enclosing circle was divided into two zones of equal area: an inner disc and an outer ring (Figure 10), an each r,θ-coordinate was defined as belonging in the inner or outer zone based upon its distance from the centre of the area. The thigmotaxis metric ranges from 0 to 1, where 1 represents the insect remaining at the perimeter of the arena for the entire observation period and 0 remaining only in the arena centre.

To quantify exploration, each circular arena was divided into a network of 96 cells of equal area by a series of concentric circles and line segments (Figure 11). The grid cell location of each r,θ-coordinate in a trajectory path is determined, and a measurement of proportion of territory visited (number of unique cells visited / total number of cells) is calculated for each insect over the course of observation.

2.4. Validating the software

2.4.1. Testing tracking accuracy

Implemented smoothing and thresholding procedures acted to eliminate the majority of false artifacts from raw trajectories. To quantify the remaining level of unreliability in the system and measure its accuracy, the tracker was compared to human users. Videos were manually authenticated by producing a series of images at random points during the analysis, and asking human users to estimate the x,y position of objects in the image using a simple interactive C++ application (Figure 13). For the same frames, human-estimated object coordinates were compared with raw object coordinates detected by the software, and with processed object coordinates returned after movement thresholding and smoothing in order to gain a correlative measure of accuracy (Figures 14 & 15).

2.4.2. Comparing endogenous locomotion with computer-generated trajectories

An effective way to investigate meaningful relationships between the different behavioural metrics defined above is to compare endogenous beetle locomotion with computer-generated data (in which the simulated trajectories represent 'perfect' information consisting of known coordinates), helping to separate biological relationships in the data from simple mathematical correlations. Computer-generated trajectories were made using a correlated walk rule by modifying code from the adehabitatLT package (Calenge et al., 2009) and scripts made available by Colomb et al. (2012). Walks were fitted by adjusting values for median velocity, median turning angle and median activity duration until they closely resembled endogenous locomotion. (A sample trajectory is shown in Figure 16.)

Colomb et al. (2012) used two methods of trajectory simulation to compare with the endogenous locomotion of adult *Drosophila melanogaster*: correlated walks and Levy walks. Whilst these two methods differ only in terms of their velocity sampling, they can produce emergent differences in defined behavioural metrics, such as total distance travelled and duration of activity bouts (Colomb et al., 2012). Here, by using a run length encoded binary sequence to model the probability of activity (as opposed to a randomly oscillating binary sequence), we are able to simulate more sustained bursts of movement /non-movement lasting for minutes at a time, which more closely resembles true beetle locomotion.

In order to simulate our experimental data, computer-generated trajectories were bounded within a circular arena. Starting from an initial starting point at the centre of the arena, the first turning angle is chosen randomly, with each proceeding angle sampled from a wrapped normal distribution around the previous angle. The correlation strength between two consecutive turning angles is determined by a concentration parameter, 'rho', between 0 and 1. The distance moved between each generated frame is then determined by sampling from a gamma distribution fitted to endogenous beetle speed data. When a generated x,y-coordinate lies outside the arena limits (i.e. when $R_t > r$, where R is $sqrt(x_t^2 + y_t^2)$ and r is the radius of the bounded arena), it is replaced by the

nearest point within the bounded limits and the next turning angle is resampled randomly.

In order to simulate bouts of movement and inactivity, each speed value was multiplied by a value taken from a run length encoded binary sequence, which was produced in two steps. Firstly, bout durations of activity and inactivity were generated by sampling from two separate gamma distributions which model endogenous beetle activity/inactivity durations (two different distributions were used as bouts of inactivity were generally found to last longer than bouts of activity in *T. molitor*). Secondly, generated bout durations were alternately sampled to create an alternating sequence of inactivity frames (zeros) and activity frames (ones). For example, the inactivity sample <2,3,1> and activity sample <5,2,2> produce the binary sequence <001111100011011>.

The shape (α) and rate (β) parameters of the gamma distributions used to simulate speed and activity/inactivity duration were optimised by modelling endogenous data with the fitdistr() function of the MASS package (Venables & Ripley, 2002). Thirty trajectory simulations were generated to compare with endogenous beetle locomotion, with each lasting 10 minutes at 20 frames per second. In order to approach observed endogenous variation in speed and activity/inactivity duration, the shape and rate values of gamma distributions was set to vary for each simulated trajectory. The parameters chosen for the simulations were as follows: velocity: $\alpha = 0.695 \pm 0.144$, $\beta = 0.119 \pm 0.0335$ (mean ± S.D.); stationary phase duration: $\alpha = 0.221$, $\beta = 0.00764$; mobile phase duration: $\alpha = 0.773$, $\beta = 0.0557$; angle correlation (rho) = 0.994.

2.5. Computational efficiency?

Benchmarking...

2.6. Experimental validation

2.6.1. *Experimental methods*

…...

2.6.2. *Statistical analysis*

All statistical analyses were conducted using R v3.1.2 under Ubuntu (R Development Core Team, 2014). Behavioural metrics are described for each individual as either single values (e.g. thigmotaxis, exploration, number of pauses) or medians of multiple values (e.g. speed, turning angle, pause duration). Medians were used instead of means to describe these continuous metrics as they were clearly non-normally distributed.

A principal component analysis (PCA) of endogenous locomotion revealed no clear correlations between behavioural metrics (Figure 18), so the effects of immune challenge and beetle gender were investigated separately for each behavioural variable. A linear mixed effects was built for each metric using the lme4 package (Bates et al., 2014), with beetle sex and treatment (and their interaction) defined as fixed effects and beetle identity defined as a random effect. P-values were corrected for multiple comparisons in order to control for false discovery (e.g. Benjamini & Hochberg, 1995).

3. Results

3.1. Validation of the tracking software

3.1.1. Automated vs. human tracking

Comparison between insect identification by the software and by human users..... (Figures 14 & 15).

3.1. Endogenous vs. simulated locomotion

Brief discussion of correlation plots..... (Figures 17). Simulated trajectories help determine the chance levels of certain complex emergent properties of locomotion, such as thigmotaxis and exploration, and the how they can be affected by differences in simple locomotion rules, such as movement speed or pause duration.

3.2. Sexual dimorphism in *T. molitor* sickness behaviours

… (Figure 19).

4. Discussion

The developed tracker, Ubitrail, is capable of recording the trajectory of up to 24 insects simultaneously with relatively high spatial (up to 0.5mm / pixel) and temporal resolution (up to 30Hz). A range of biologically meaningful behavioural metrics have been defined in order to produce quantitative data on insect locomotion, including information on velocity, turning angles and location, as well as several more specific behaviours such as turnarounds, thigmotaxis and exploration.

The tracker is: (i) versatile, having been tested on a range of insect species, including *Tenebrio molitor*, *Drosophila* spp. (both adults and larvae), ants (*Lasius niger*), aphids (*Acyrthosiphon pisum*) and bean weevils (*Acanthoscelides obtectus*), (ii) robust, working with relatively low resolution video images (640 x 480 px), imperfect and variable lighting conditions and moderate levels of visual occlusion and background variation, and (iii) accurate, as tracked coordinates of *T. molitor* were found to closely match (<10% of the body length of the insect away from) coordinates determined by human users. Furthermore, the system is inexpensive as the software makes use of only open-source tools and does not require specialised apparatus for experimental set ups or recording. Finally, the inclusion of a graphical user interface (GUI) for video analysis and R package, Rubitrail, for statistical analysis, aim to maximise accessibility to the user and allow for tracking analysis 'straight out of the box'.

References

Bates, D., Maechler, M., Bolker, B., & Walker, S. (2013). lme4: Linear mixed-effects models using Eigen and S4. R package version, 1(4).

Benjamini, Y., & Hochberg, Y. (2000). On the adaptive control of the false discovery rate in multiple testing with independent statistics. *Journal of Educational and Behavioral Statistics*, *25*(1), 60-83.

Beyan, C., & Fisher, R. B. (2013, September). Detection of abnormal fish trajectories using a clustering based hierarchical classifier. In British Machine Vision Conference (BMVC), Bristol, UK.

Branson, K., & Bender, J. (2012). CTRAX–the caltech multiple walking fly tracker. http://ctrax.sourceforge.net/

Calenge, C. (2011). Analysis of animal movements in R: the adehabitatLT Package. *R Foundation for Statistical Computing: Vienna, Austria*.

de Chaumont, F., Coura, R. D. S., Serreau, P., Cressant, A., Chabout, J., Granon, S., & Olivo-Marin, J. C. (2012). Computerized video analysis of social interactions in mice. Nature methods, 9(4), 410-417.

Colomb, J., Reiter, L., Blaszkiewicz, J., Wessnitzer, J., and Brembs, B. (2012). Open Source Tracking and Analysis of Adult Drosophila Locomotion in Buridan's Paradigm with and without Visual Targets. PLoS ONE 7, e42247.

Dankert, H., Wang, L., Hoopfer, E. D., Anderson, D. J., & Perona, P. (2009). Automated monitoring and analysis of social behavior in Drosophila. Nature methods, 6(4), 297-303.

Dingemanse, N. J., Both, C., Drent, P. J., Van Oers, K., & Van Noordwijk, A. J. (2002). Repeatability and heritability of exploratory behaviour in great tits from the wild. Animal Behaviour, 64(6), 929-938.

Domenici, P., Booth, D., Blagburn, J. M., & Bacon, J. P. (2008). Cockroaches keep predators guessing by using preferred escape trajectories. Current Biology, 18(22), 1792-1796.

Gomez-Marin, A., Partoune, N., Stephens, G. J., Louis, M., & Brembs, B. (2012). Automated tracking of animal posture and movement during exploration and sensory orientation behaviors. PloS one, 7(8), e41642.

Götz, K. G., & Biesinger, R. (1985). Centrophobism inDrosophila melanogaster. Journal of Comparative Physiology A, 156(3), 329-337.

Hen, I., Sakov, A., Kafkafi, N., Golani, I., Benjamini, Y. (2004). The dynamics of spatial behavior:

how can robust smoothing techniques help. J Neurosci Methods, 133, 161–72.

Lochmatter, T., Roduit, P., Cianci, C., Correll, N., Jacot, J., & Martinoli, A. (2008, September). Swistrack-a flexible open source tracking software for multi-agent systems. In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on (pp. 4004-4010). IEEE.

Martin, J.R. (2004). A portrait of locomotor behaviour in Drosophila determined by a video-tracking paradigm. Behav. Processes 67, 207–219.

Noldus, L. P., Spink, A. J., & Tegelenbosch, R. A. (2001). EthoVision: a versatile video tracking system for automation of behavioral experiments. Behavior Research Methods, Instruments, & Computers, 33(3), 398-414.

Robie, A.A., Straw, A.D. & Dickinson, M.H. (2010) Object preference by walking fruit flies, Drosophila melanogaster, is mediated by vision and graviperception. The Journal of Experimental Biology 213: 2494–506.

Straw, A. D., & Dickinson, M. H. (2009). Source Code for Biology and Medicine. Source code for biology and medicine, 4, 5.

Swierczek, N. A., Giles, A. C., Rankin, C. H., & Kerr, R. A. (2011). High-throughput behavioral analysis in C. elegans. Nature methods, 8(7), 592-598.

Valente, D., Golani, I., & Mitra, P.P. (2007) Analysis of the trajectory of Drosophila melanogaster in a circular open field arena. PloS one 2: e1083.

Venables, W. N., & Ripley, B. D. (2010). Package ‘MASS’. Online at: http://cran. r-project. org/web/packages/MASS/MASS.pdf.

Yaski, O., Portugali, J., & Eilam, D. (2011). Arena geometry and path shape: When rats travel in straight or in circuitous paths?. Behavioural brain research, 225(2), 449-454.

406 **Figures and figure legends**

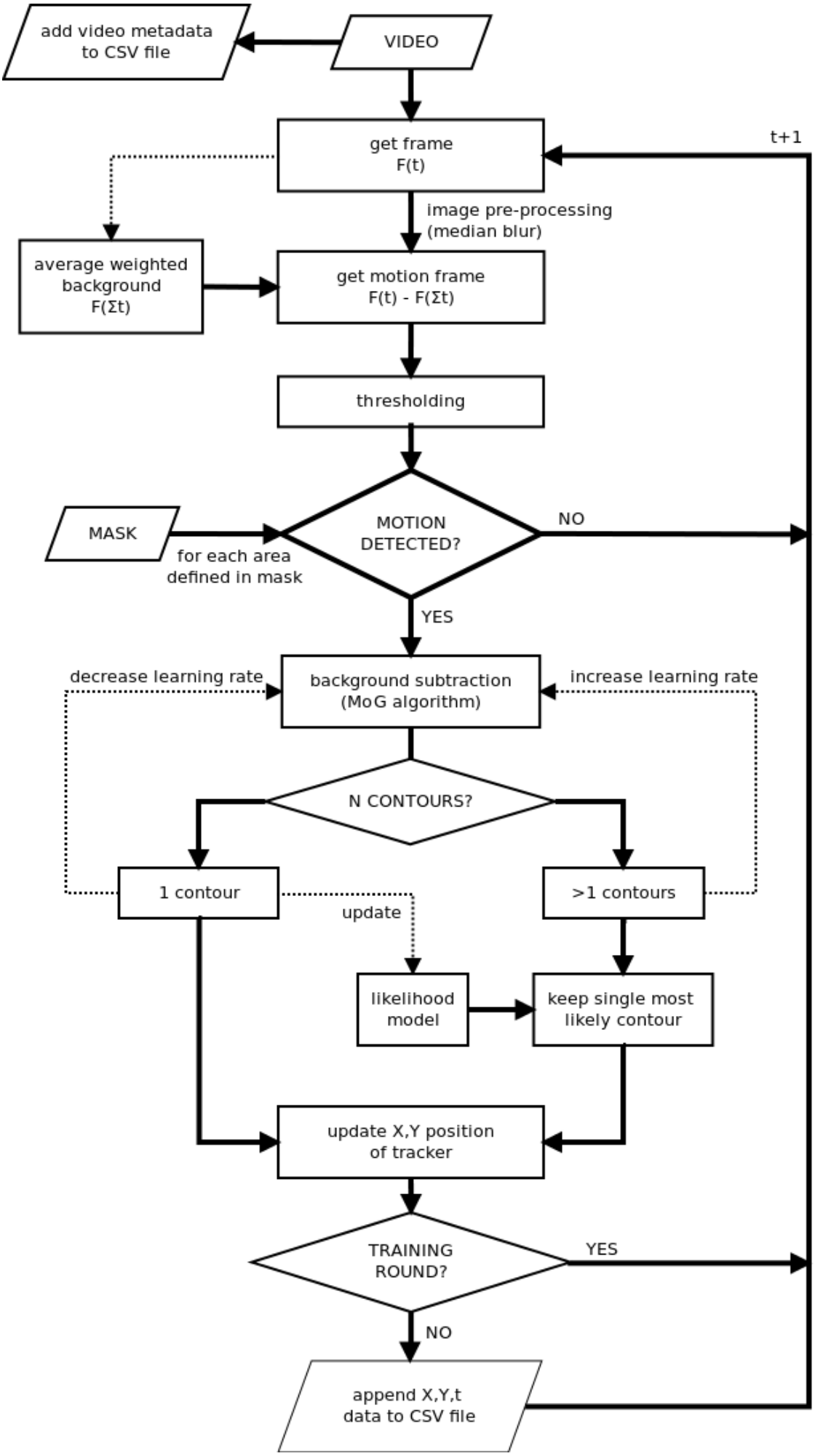

add video metadata to CSV file
VIDEO
get frame F(t)
t+1
image pre-processing (median blur)
average weighted background F(Σt)
get motion frame F(t) - F(Σt)
thresholding
MASK
for each area defined in mask
MOTION DETECTED?
NO
YES
decrease learning rate
background subtraction (MoG algorithm)
increase learning rate
N CONTOURS?
1 contour
update
>1 contours
likelihood model
keep single most likely contour
update X,Y position of tracker
TRAINING ROUND?
YES
NO
append X,Y,t data to CSV file

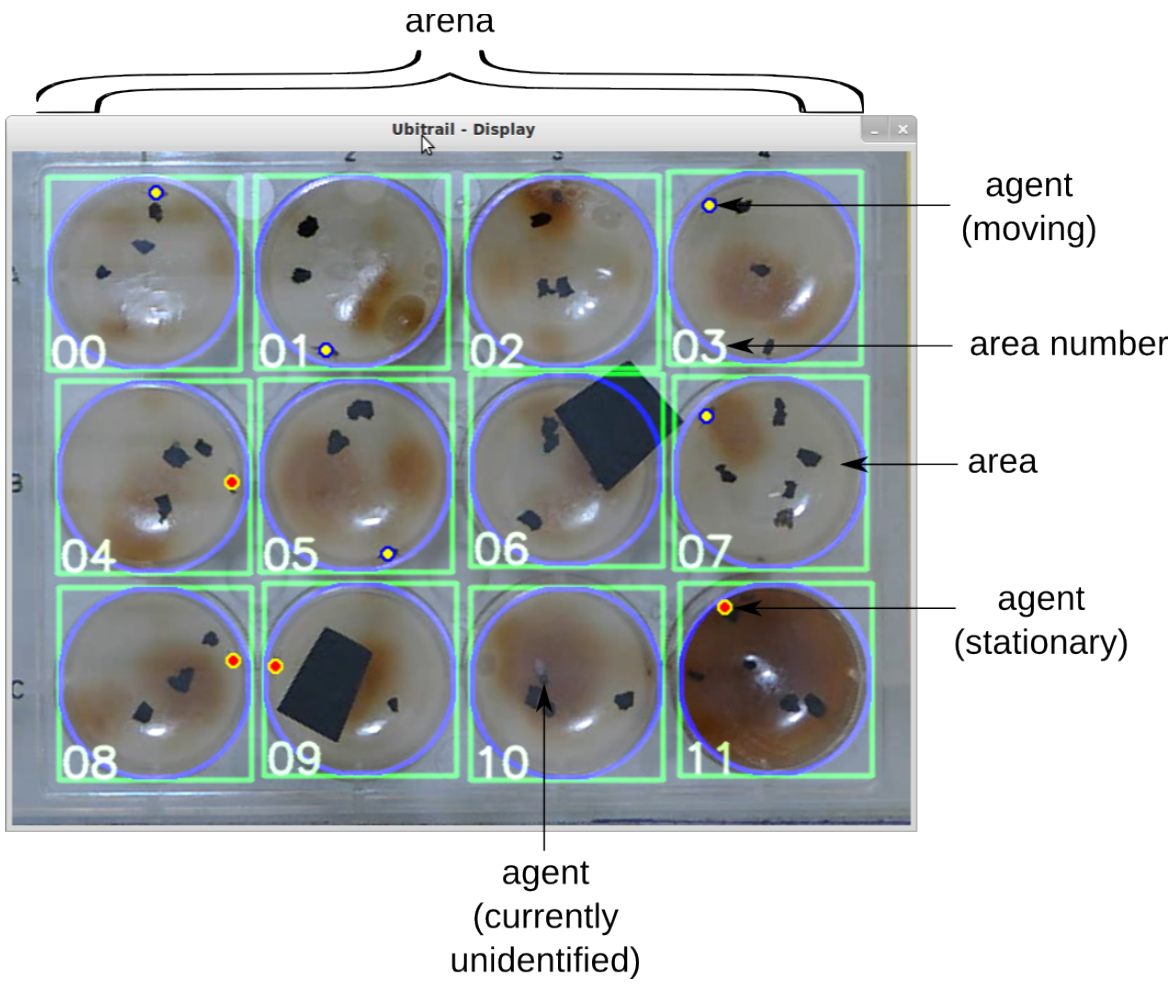
Figure 1. Flow diagram of tracking software process. Ubitrail takes a digital video file as input and returns a CSV file containing tracked X,Y-coordinates for each frame in each defined area, as well as a header containing metainformation extracted from the video. Each frame, F(t), is extracted individually from the video and de-noised using a (9x9) median blur filter. A motion frame is then produced by subtracting the current frame from a running weighted average of previous frames, F(t-1), which is used to model the background. At this point, the mask is applied to split the frame up into individual areas. In areas where motion is detected, a dynamic learning algorithm based upon a mixture of Gaussian (MoG) background subtraction method is used to identify moving foreground objects. The MoG algorithm is trained separately for each individual area, with the rate of learning being increased following ambiguous frames in which the movement of more than one foreground object is detected, and decreased following unambiguous in which movement of exactly one foreground object is detected. In order to solve ambiguities in foreground detection, an on-the-fly likelihood model is built based upon several key features of known foreground objects, including contour shape, mean and standard deviation of pixel colour in the red, green and blue channels, and distance between centre of the current contour and centre of the last detected contour. A log-likelihood, L, is then calculated under the assumption of normal distribution, where:

$$\mathrm{L} = \sum_{i=0}^{n} l_i$$

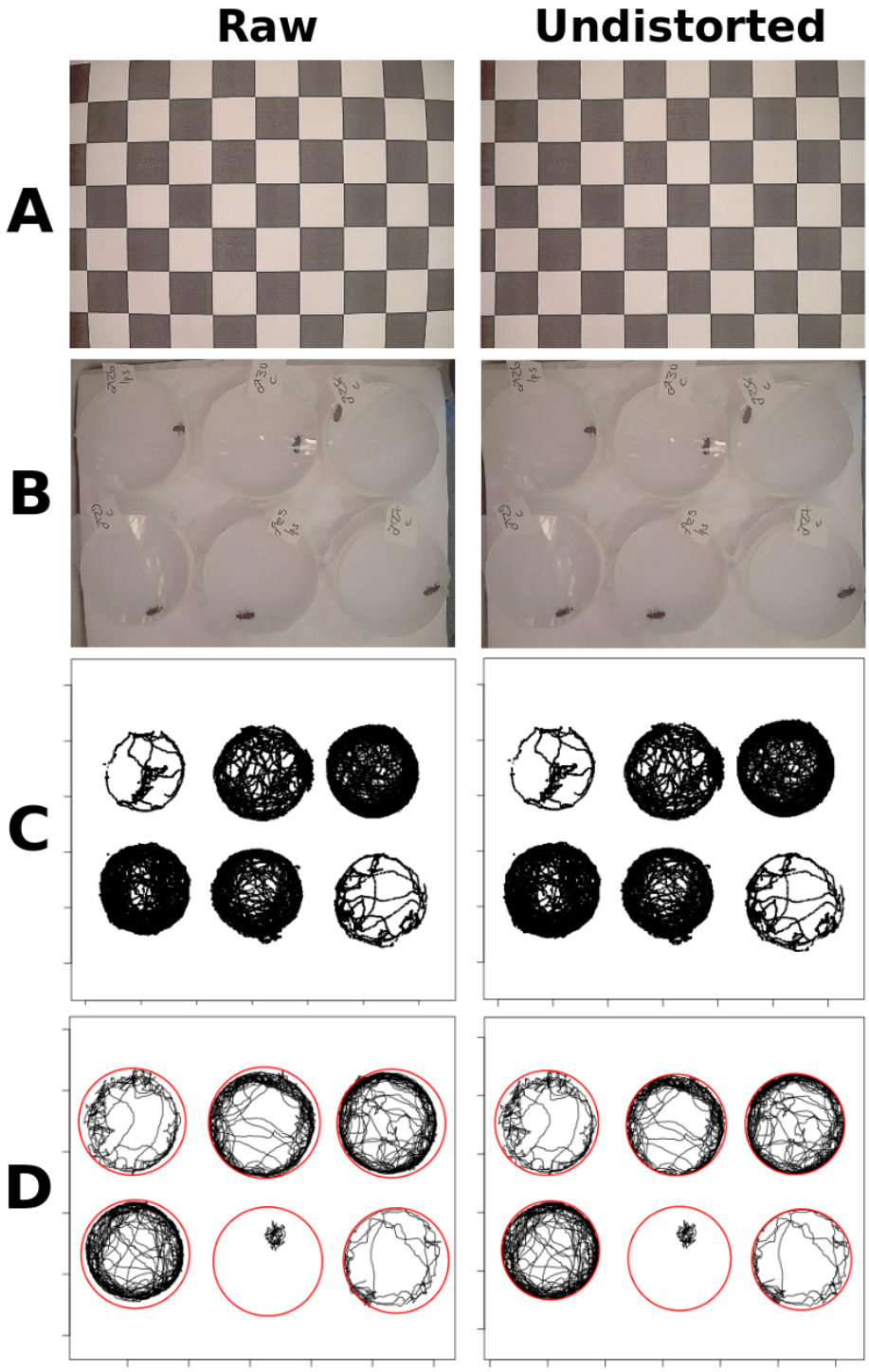
$$\mathrm{l}_i = \ln\left(\frac{1}{s_i\sqrt{2\pi}} e^{-\frac{(x_i - m_i)^2}{2s_i^2}}\right)$$

and where **n** is the total number of features, **i** is a given feature, $\mathbf{x_i}$ = value of feature **i**, $\mathbf{m_i}$ is the psuedo-mean of feature **i**, and $\mathbf{s_i}$ is the pseudo-standard-deviation of feature **i**. When more than one contour is detected, the contour with the maximum log-likelihood is taken as the foreground object. An initial training round (default = 500 frames) is used to train the background subtraction

algorithm and build a suitable likelihood model for foreground detection, ensuring valid foreground detection throughout the video analysis.



436 **Figure 2**. The graphical user interface (GUI) of the Ubitrail software, with key elements labelled.
437 The outline of the mask which defines each individual arena is shown in blue, with its assigned area
438 number depicted inside the green square. The above sample shows a test on *Drosophila*
439 *melanogaster* adults, where variation in background luminosity was created to test for the
440 robustness of background subtraction, and covering objects (unlabelled black objects) were
441 introduced to test for the effects of occlusion upon insect tracking.



442 **Figure 3.** The effects of lens undistortion upon images and tracked coordinates. (A) An image of a
443 chessboard pattern is captured during the recording stage and used to calibrate parameters for an
444 undistortion matrix. The effects of undistortion (raw [left] vs. processed [right]) are shown for (B)
445 raw video frames, (C) tracked movement vectors, and (D) fitting a minimal enclosing circle to
446 arenas.

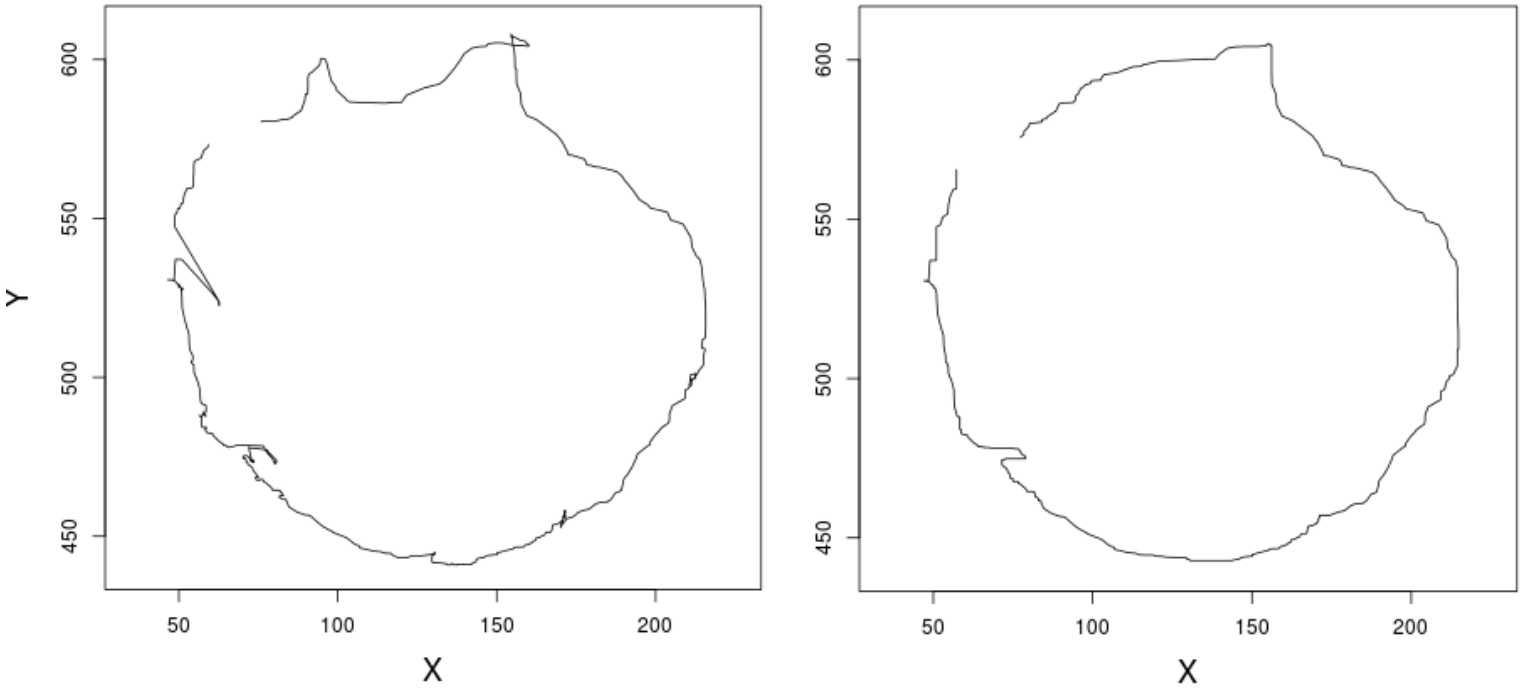


Figure 4. Smoothing of tracked x,y-coordinates. (A) shows a 60s sample of raw trajectories outputted by the tracking software, whilst (B) shows the same trajectory after application of a rolling median with window size of 3s (60 frames).

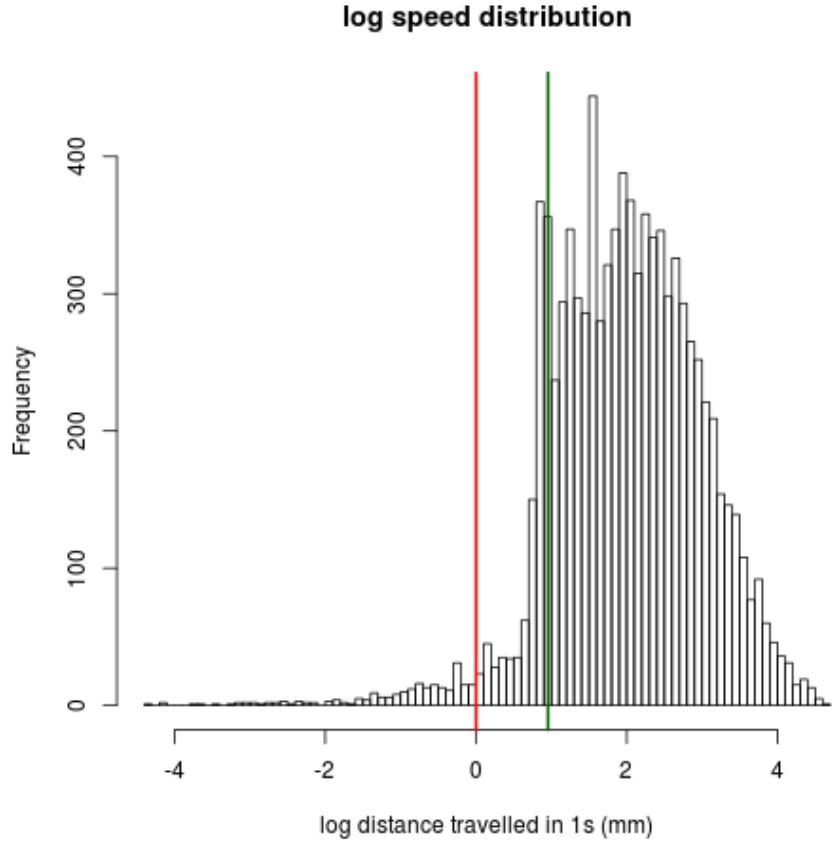
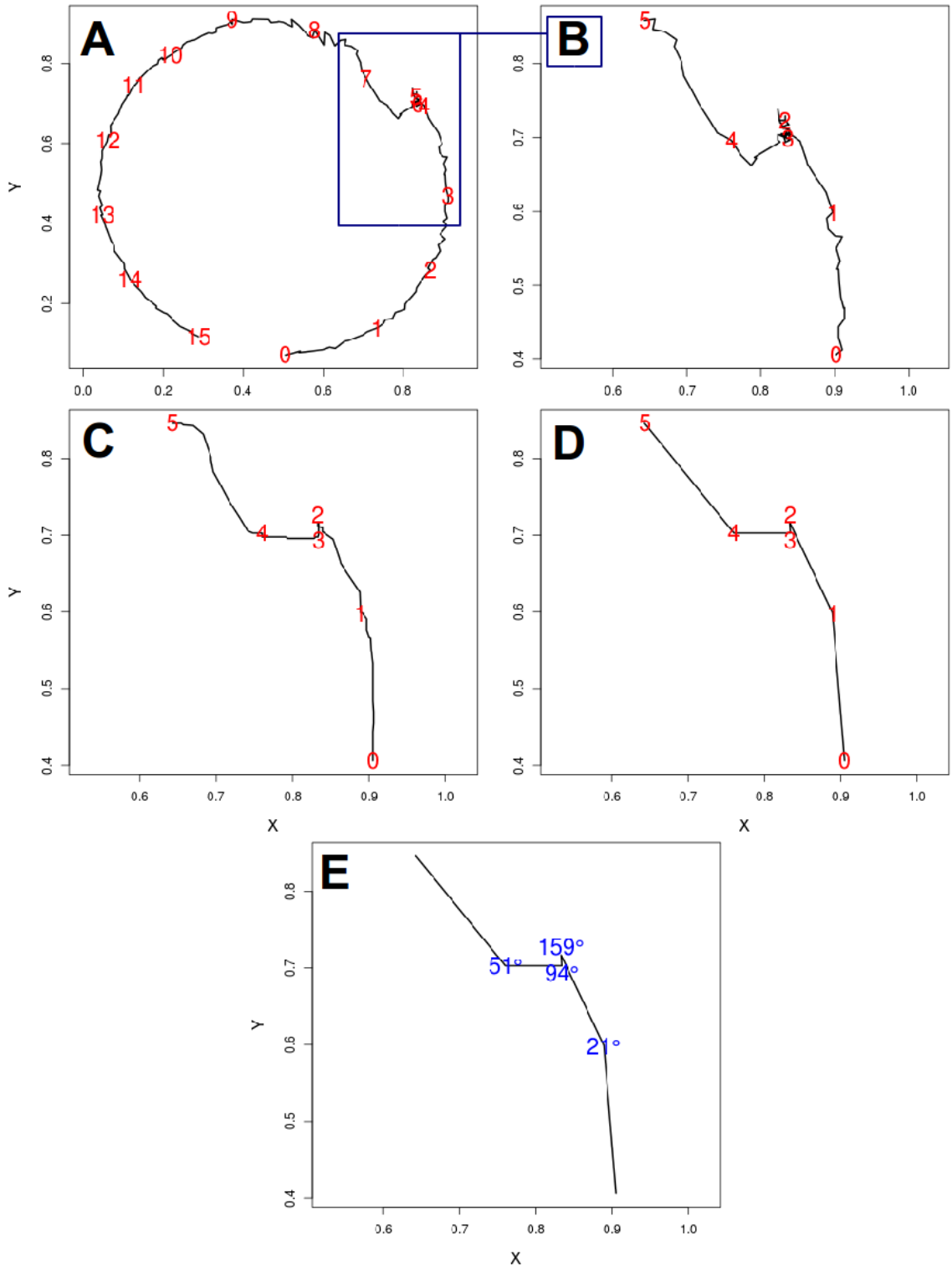
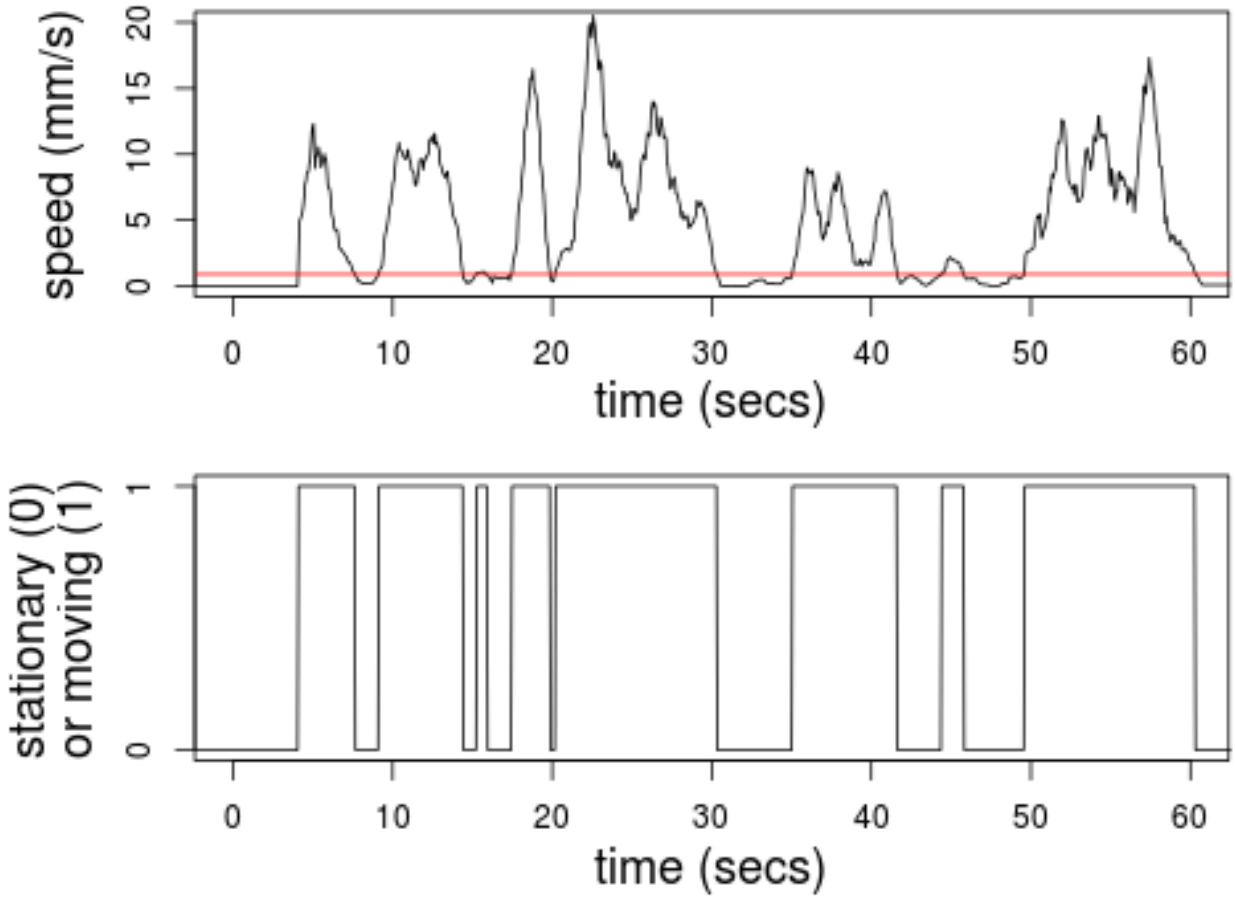


Figure 5. Histogram of smoothed velocity frequency (logarithmic scale), with lower threshold (red line: 0 mm/s) and upper thresholds defined (green line: 1 mm/s)

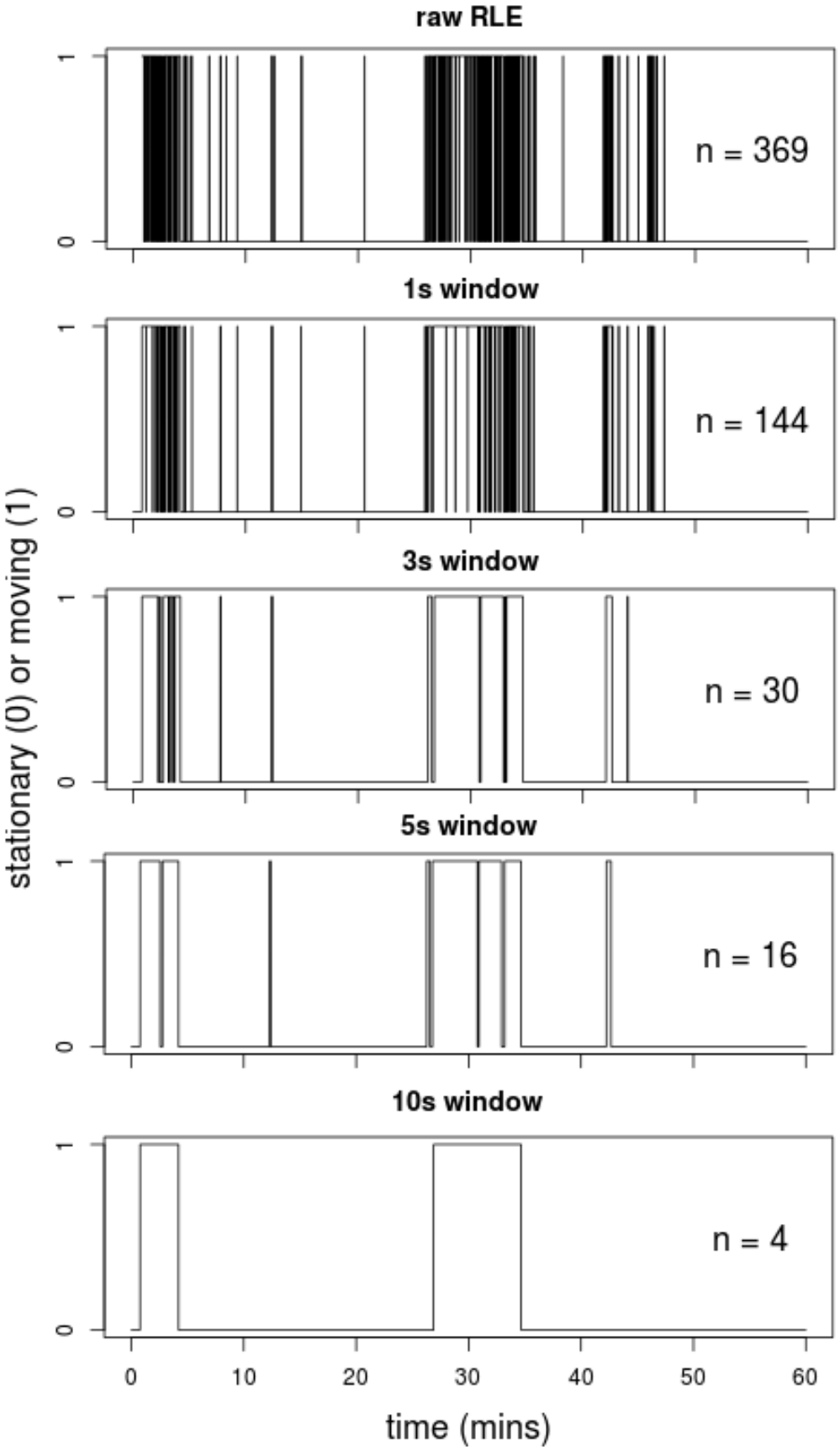


452 **Figure 6.** Calculation of turning angles. (A) shows a 15s sample (300 frames) of raw tracked X,Y-
453 coordinates, with corresponding number of seconds overlaid (red text). The area inside the blue box
454 is a 5s subsample which is zoomed on in (B-E), where (B) shows the same raw X,Y-coordinates
455 with number of seconds (red text), (C) shows coordinates that have been smoothed using a rolling
456 median with a window of 20 frames [1s]), (D) shows coordinates that have been smoothed

457 (window=20) and then resampled at 20Hz (1 frame per s), and (E) shows the final relative turning
458 angles (in degrees; blue text) calculated from smoothed and resampled coordinates in (D).

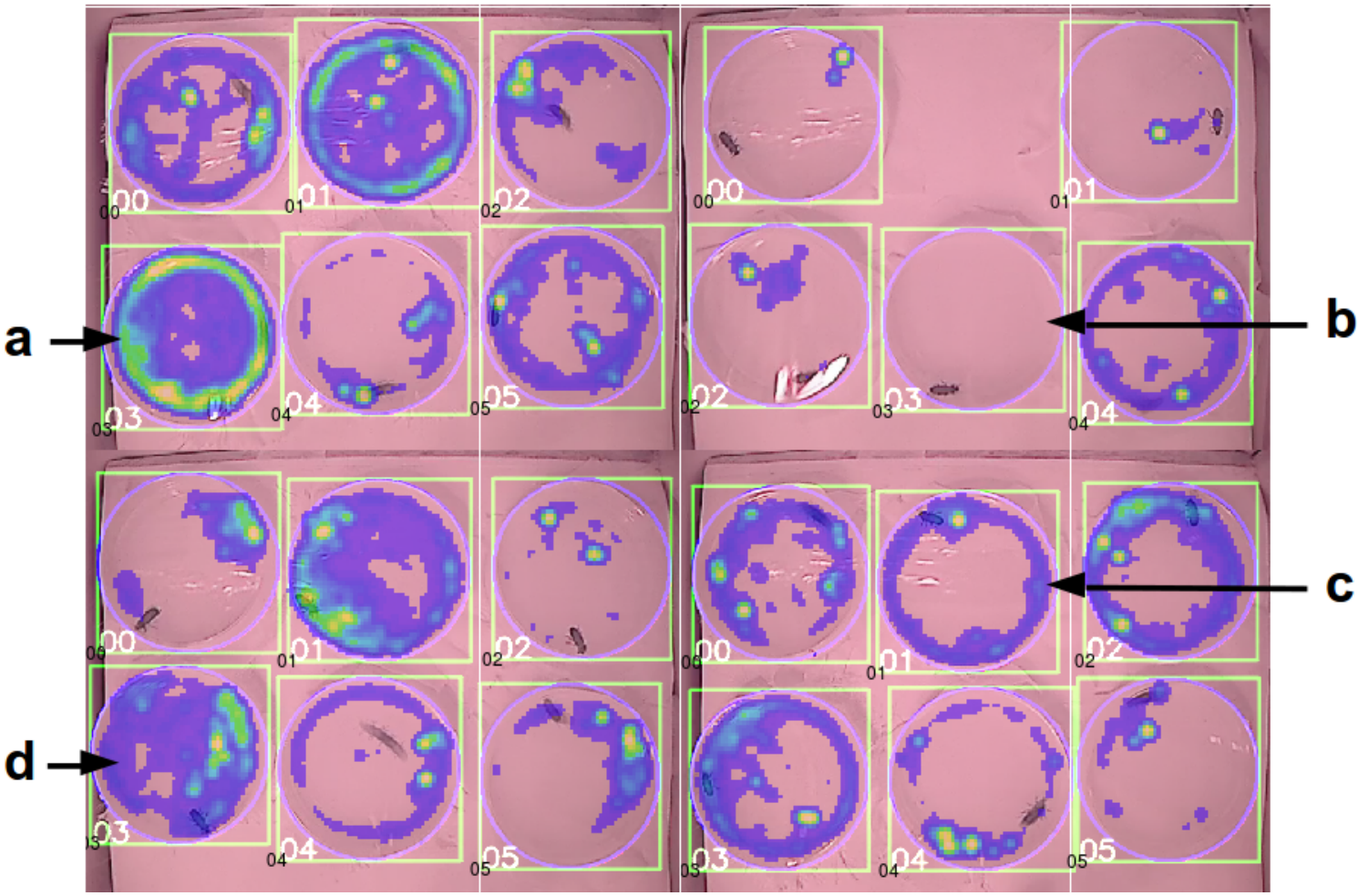


460 **Figure 7**. Determination of insect movement (mobile vs. stationary) using run length encoding. (A)
461 shows a 60s sample of smoothed insect movement speed (mm/s). The red line represents the user-
462 defined speed threshold (here, 1mm/s) below which the insect is classified as being stationary. (B)
463 shows the same sample with speed run length encoded into a binary format, whereby the insect is
464 classified as mobile (1) when moving faster than the speed threshold and stationary (0) when
465 moving slower.



467 **Figure 8.** A sample of different sliding window sizes for smoothing run length encoded movement
468 data, from raw data (1 frame interval) to 10s (200 frame interval). n indicates the number of defined

transitions between mobile and stationary phases, which can be seen become eroded as the size of the sliding window increases. A sliding window of 3s was used in the final analysis as this was found to preserve biologically meaningful pausing events whilst still reducing noise from oversampling.



474 **Figure 9.** Sample 'heatmaps' showing the frequency insect locations over the course of a 60 minute
475 recording, with yellow areas being visited frequently and blue areas infrequently. Each background
476 image is a frame taken from the raw video analysed by the tracker. (a) shows an insect which
477 displayed a high level of exploration as well as a relatively high degree of thigmotaxis. (b) shows an
478 insect which did not display sufficient movement during the recording to be tracked (<500 frames
479 [25s] in which motion was detected). (c) shows an insect with a high degree of thigmotaxis but a
480 relatively low level of exploration. (d) shows an insect with a high level of exploration, although
481 with movement being concentrated primarily on the right hand side of the arena.

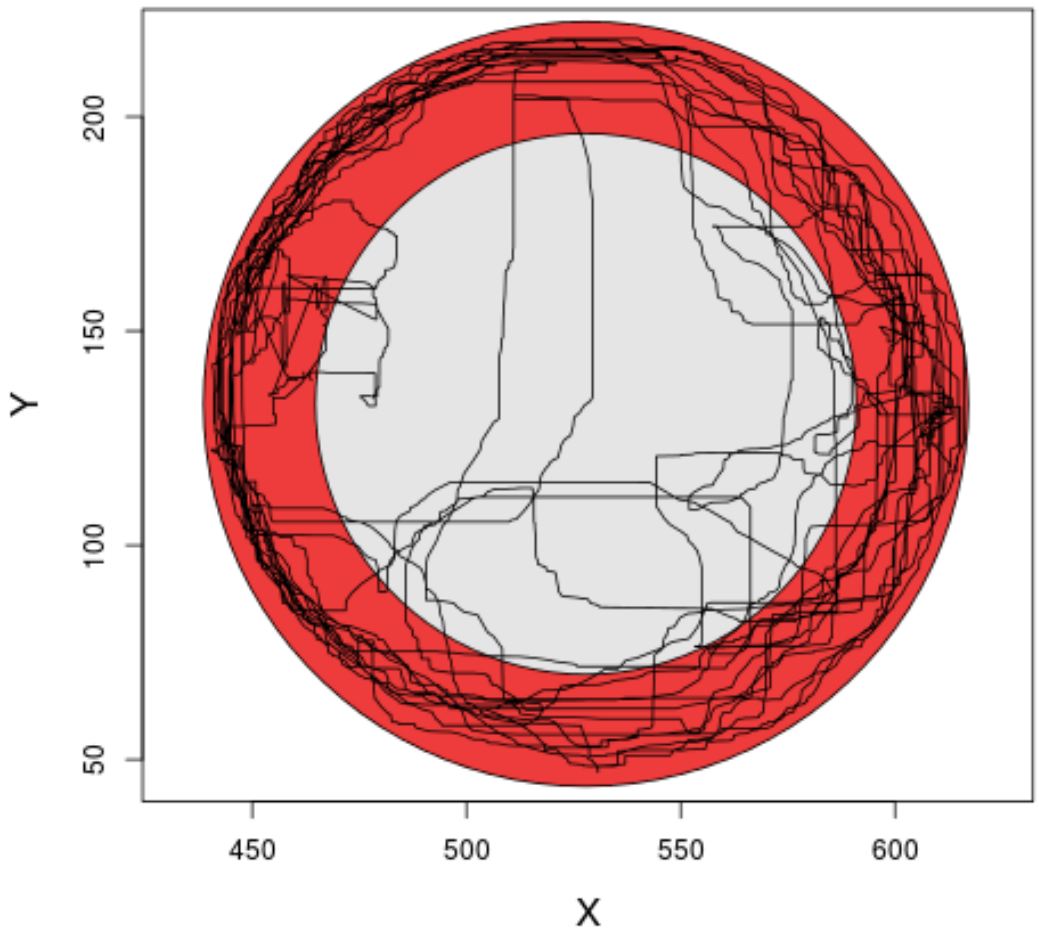


Figure 10. Visualisation of the thigmotaxis metric. A minimum enclosing circle (outer boundary of red ring) is fitted to each circular arena, which is then divided into two zones of equal area, an inner zone (shown in red) and an outer zone (shown in grey). The radius of the inner circle, r_{inner}, is √2 times smaller than the radius of the outer enclosing circle, r_{outer}. Each X,Y-coordinate in a trajectory path (black line) is designated as being in the inner or outer zone based upon its Pythagorean distance from the midpoint of the arena. I.e. a coordinate (x_t,y_t) is classified as being in the outer zone if:

$$\sqrt{(x_{mid} - x_t)^2 + (y_{mid} - y_t)^2} > \frac{r_{outer}}{\sqrt{2}}$$

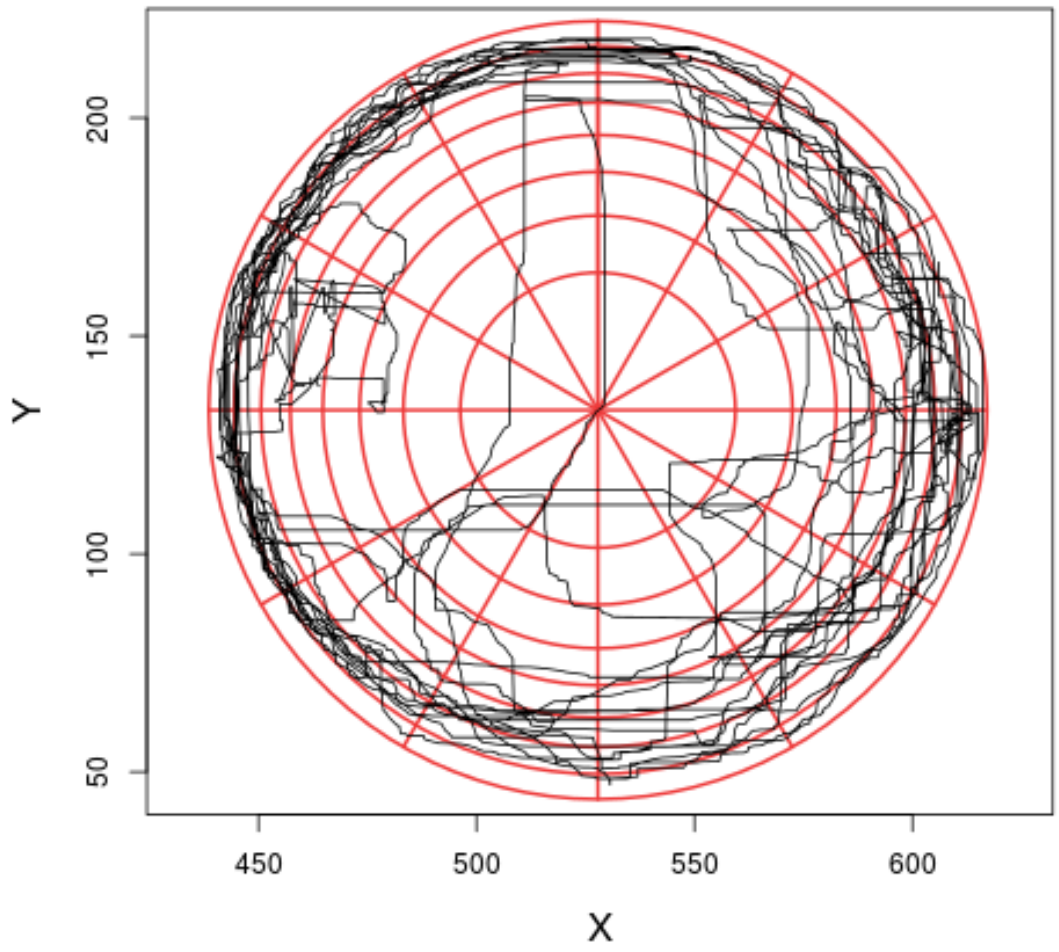


Figure 11. Visualisation of the exploration metric. A minimum enclosing circle is fitted to each circular arena, which is then divided into a network of grid cells of equal area by concentric circles and line segments. Here, 8 concentric circles and 12 line segments (shown above in red) compose a grid of 96 cells. Given a number of circles i:n, the radius of circle i, r_i, is given by the formula:

$$r_i = r_n \cdot \sqrt{\frac{i}{n}}$$

where r_n is the radius of the outermost circle enclosing the arena. Given a number of line segments, j:n, the angle of segment j, θ_j, is given by the formula:

$$\theta_j = \frac{2\pi j}{n}$$

The cell location of each coordinate in a smoothed trajectory path (shown above in black) is then determined, and the total number of unique cells visited by the insect used as a measure of exploration.

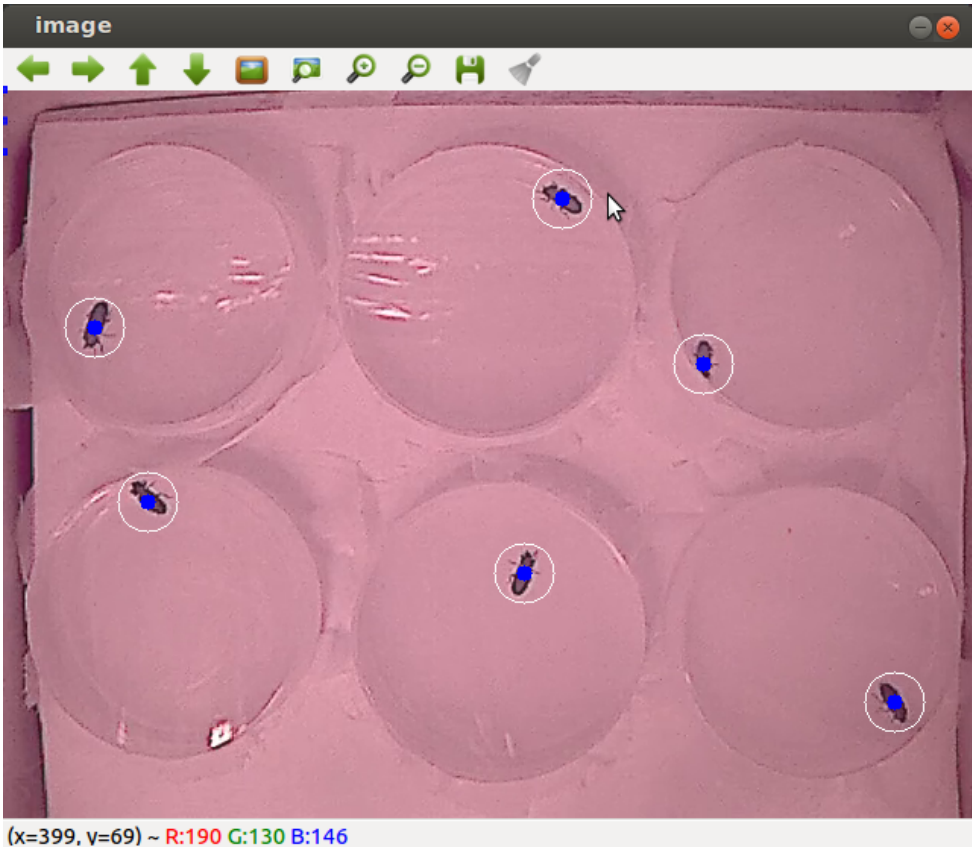


Figure 13. A sample of the human scoring test application used to examine automated tracking accuracy. Raw frames were randomly extracted from a video and opened in a simple C++ application. Users then clicked the point at which they deemed the centre of mass of insect to be (shown as blue dots with white circle). User-defined x,y-coordinates were then compared with coordinates defined by the tracking software for the same frames (see Figures 14 & 15).

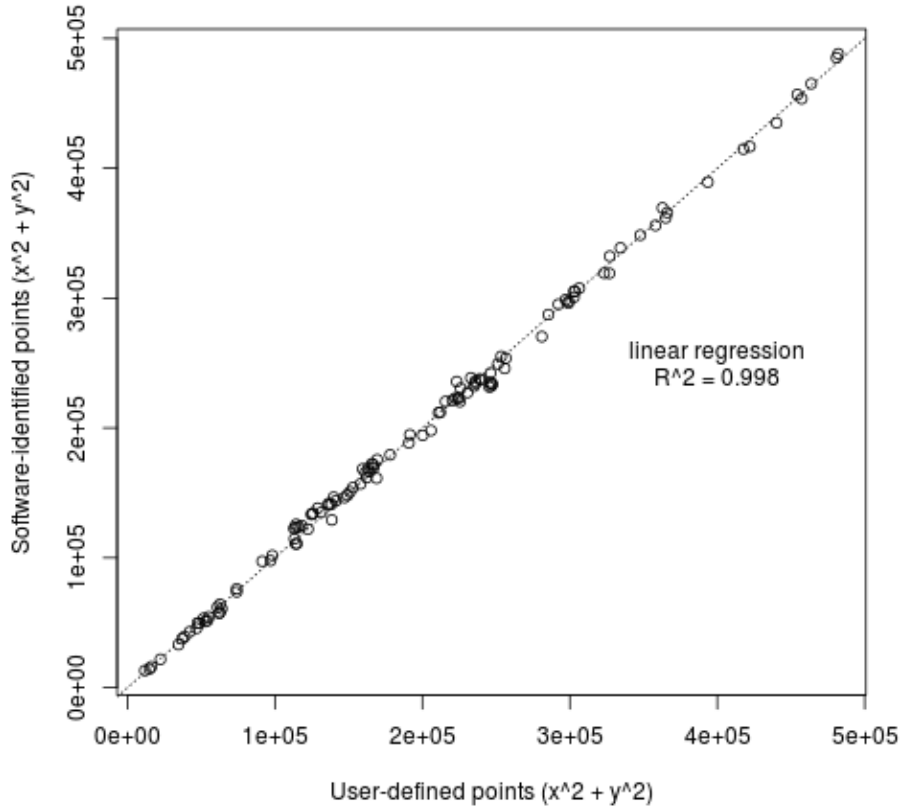
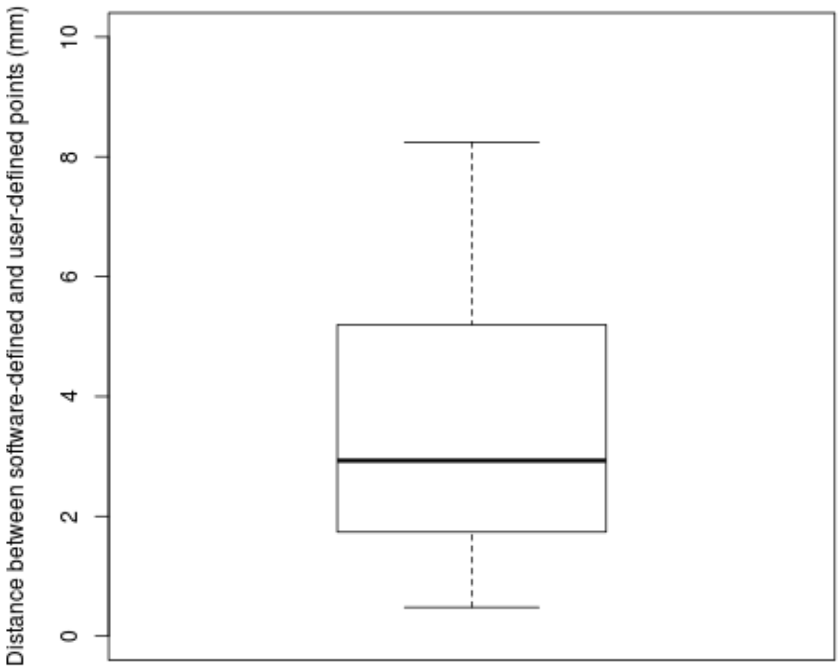
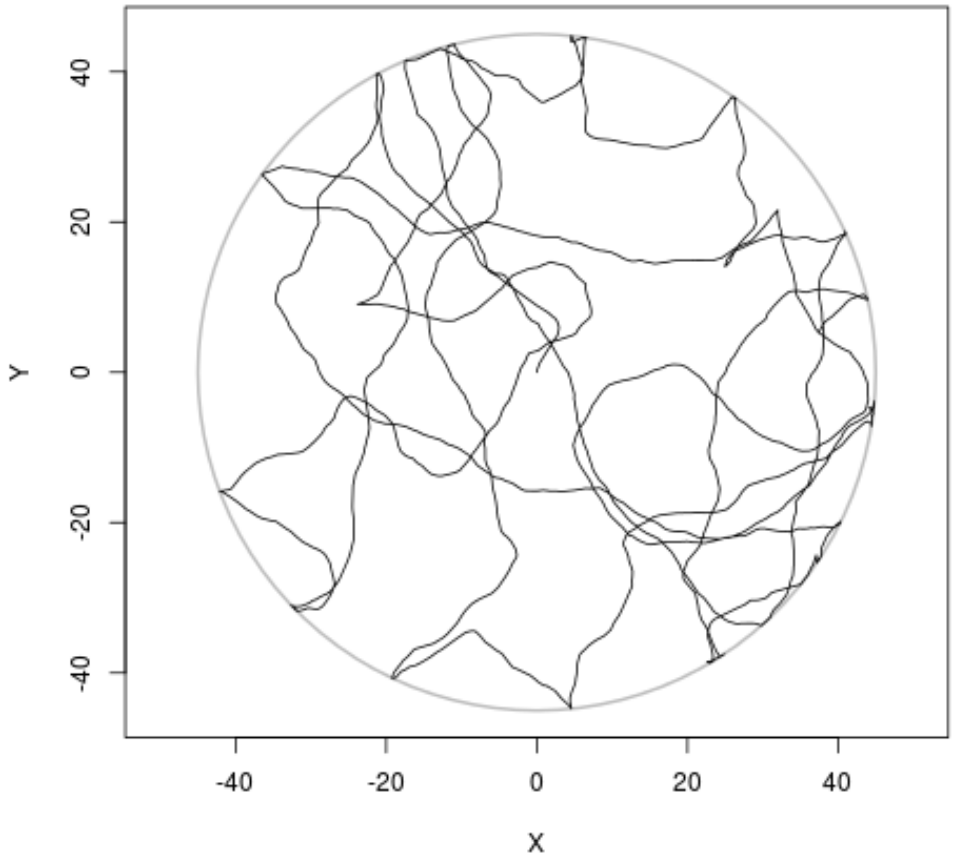


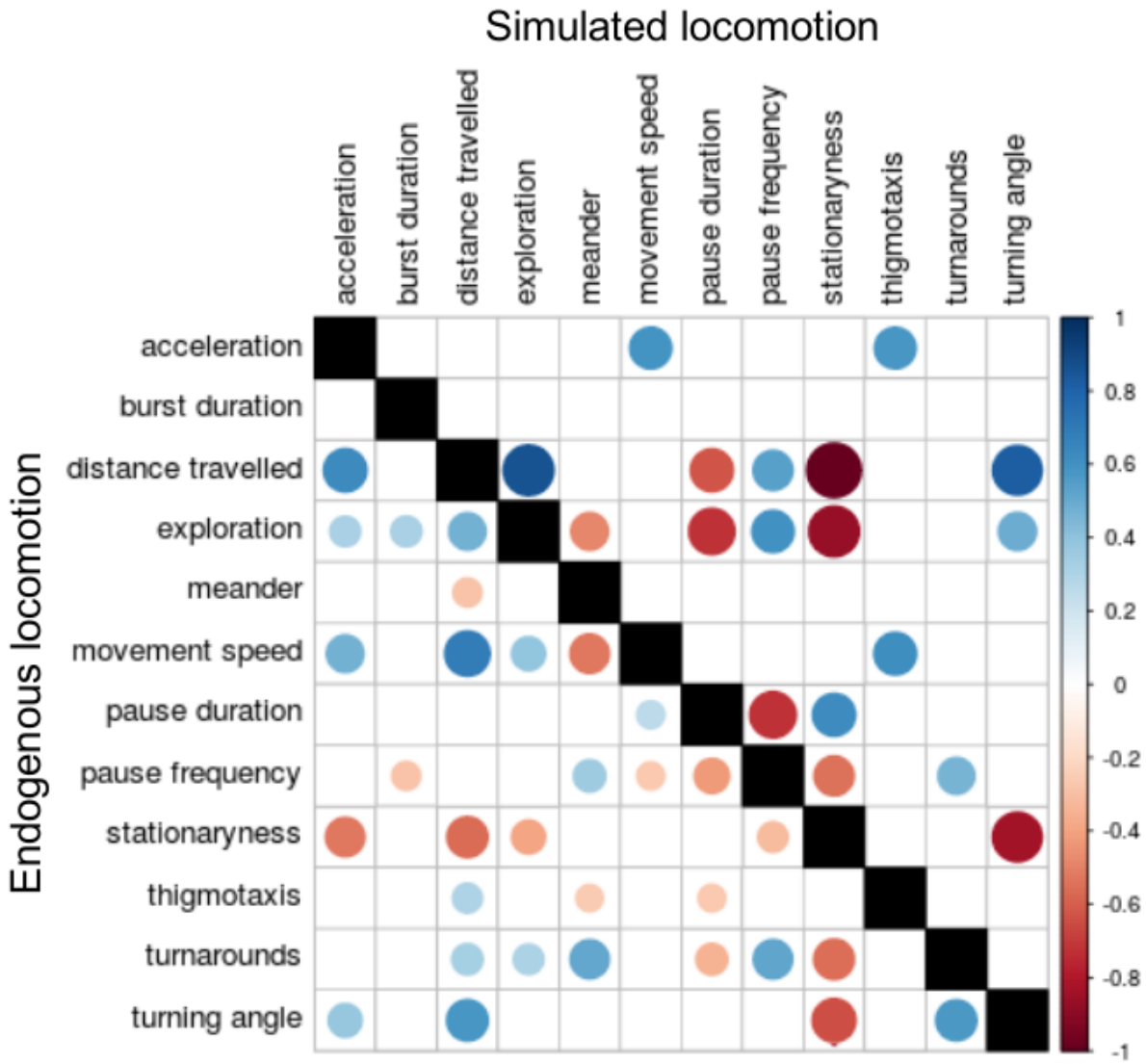
Figure 14. Correlation between user-defined X,Y-coordinates of insect location (groundtruth data) and X-Y-coordinates outputted by the automated tracker for the same frames.

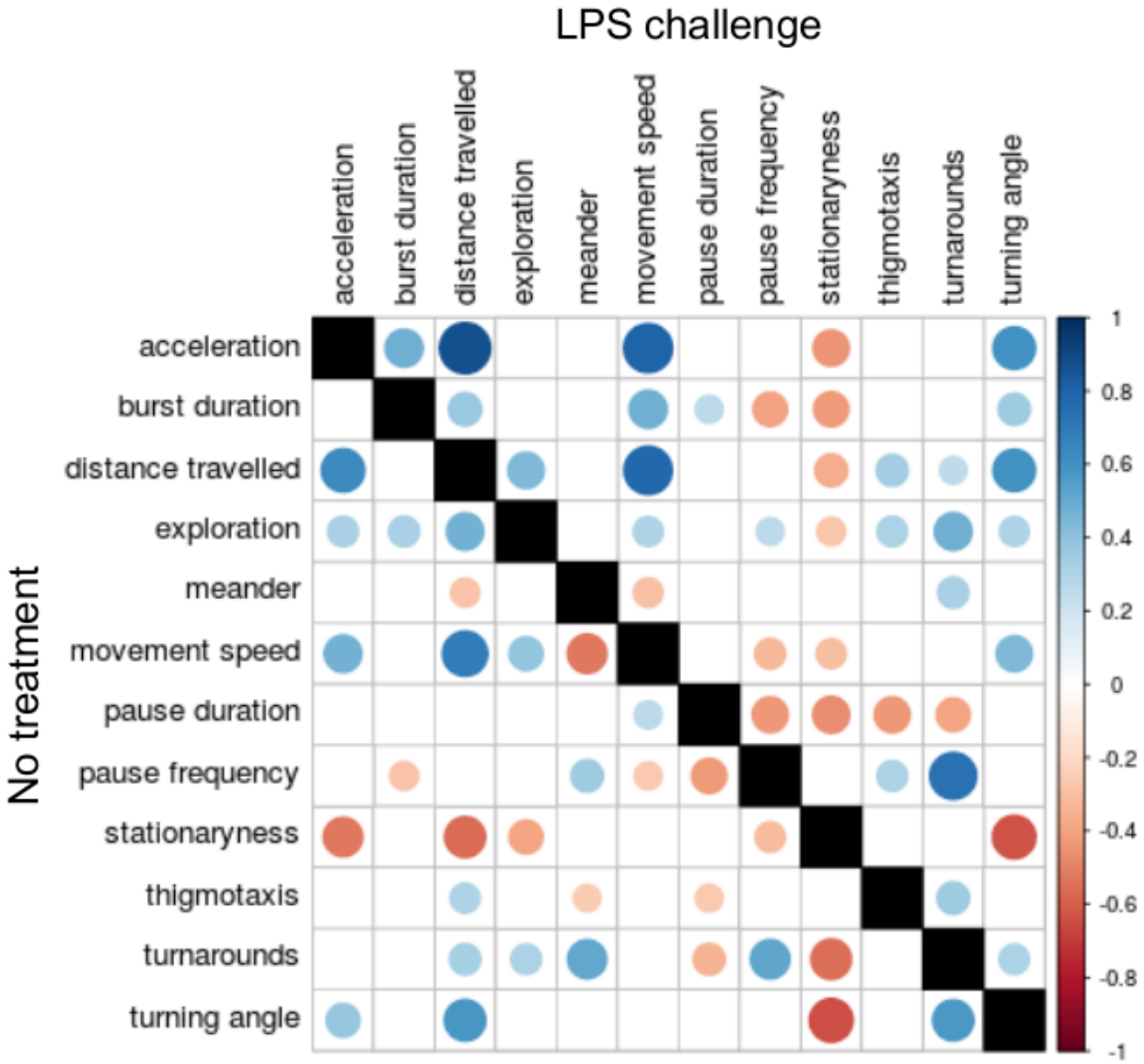


510 **Figure 15.** Boxplot showing Pythagorian distance (in mm) between tracked X,Y-coordinates of
511 insects outputted by the software and coordinates defined by users. Boxplots show the median and
512 interquartile range (IQR), and whiskers represent 1.5*IQR. For comparison, the mean body length
513 of an adult *T. molitor* is 18mm.



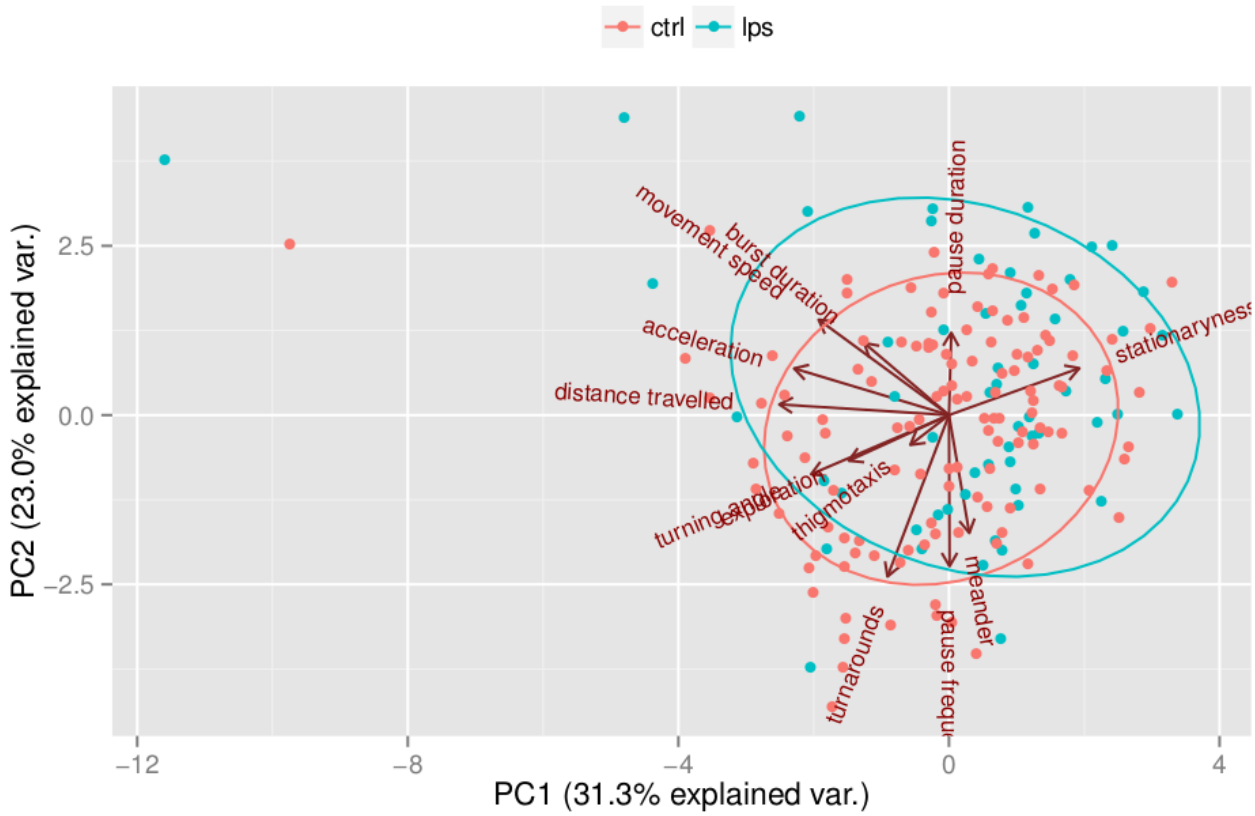
514 **Figure 16.** Sample trajectory of a correlated walk simulation lasting 10 minutes at 20 frames per
515 second. The outer bounds of the simulated arena are defined by the grey circle.


Simulated locomotion
Endogenous locomotion
acceleration
burst duration
distance travelled
exploration
meander
movement speed
pause duration
pause frequency
stationaryness
thigmotaxis
turnarounds
turning angle
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1



518 **Figure 17.** Correlation plots showing the relationships between defined behavioural metrics. The
519 order of the variables was set by clustering them for endogenous locomotion data (A, lower left
520 part). Only significant correlations ($p<0.05$) are shown, with larger circles representing greater
521 significance. Each matrix is divided into two triangles representing the correlations between the
522 same behavioural metrics extracted from different datasets. (A) Correlation matrices comparing
523 endogenous beetle locomotion with simulated correlated walks, showing correlations in untreated
524 beetles (lower left triangle) and correlated walk simulations (upper right triangle). (B) Correlation
525 matrices comparing the effects of an immune challenge upon endogenous beetle locomotion,

showing correlations in unchallenged beetles (lower left triangle) and beetles challenged with LPS (upper right triangle).



529

530 **Figure 18.** Principle component analysis (PCA) showing the effect of immune challenge upon

531 locomotory behaviour in *T. molitor*.

<TODO: Figure of sample results from *T. molitor*>

Figure 19. The effect of immune challenge (unchallenged in blue, LPS challenged in red) on behavioural metrics in *T. molitor* males and females. Bars show means (± S.E.) of single values for C,F,I,J,K. and means (± S.E.) of medians for A,B,D,E,G,H.