Deep learning in automated worm identification and tracking for C. elegan mating behaviour analysis

Conference or Workshop Item

Accepted Version

Akpu, C. H., Wei, H. ORCID: https://orcid.org/0000-0002-9664-5748 and Hong, X. ORCID: https://orcid.org/0000-0002-6832-2298 (2024) Deep learning in automated worm identification and tracking for C. elegan mating behaviour analysis. In: 27th International Conference on Pattern Recognition, ICPR 2024, 1-5 Dec 2024, Kolkata, India, pp. 113-128. doi: https://doi.org/10.1007/978-3-031-78122-3_8 (Antonacopoulos, A., Chaudhuri, S., Chellappa, R., Liu, C. L., Bhattacharya, S. and Pail, U. (eds.) Springer. ISBN: 9783031781216) Available at https://centaur.reading.ac.uk/119635/

To link to this article DOI: http://dx.doi.org/10.1007/978-3-031-78122-3_8

Publisher: Springer

Deep Learning in Automated Worm Identification and Tracking for C. Elegan Mating Behaviour Analysis

Chukwuma Hilary Akpu Jnr, Hong Wei[0000−0002−9664−5748], and Xia Hong[0000−0002−6832−2298]

University of Reading, Reading, RG6 6AY, UK
https://www.reading.ac.uk/computer-sceince/
holaryc@gmail.com; {h.wei, x.hong}@reading.ac.uk

Abstract. This study is concerned with computer vision technologies applied in C. elegans (diminutive nematodes) mating behaviour analysis, more specifically object detection and tracking to find contacts of male and female worms in worm mating videos. Advanced deep learning algorithms, such as YOLOv8 and DeepSORT, are adapted in the automated worm identification and tracking system. A modified DeepSORT algorithm is developed to cope with appearance similarity of C. elegans for improving the tracking accuracy. In addition, a male worm detection and tracking algorithm, utilising the male worm's mobility characteristic, assists the modified DeepSORT in accurate male worm tracking. Finally, worm contact detection is implemented by calculating the Euclidean distance between the male and female worms. The developed system, named as M1 and M2, is trained and evaluated under two sets of data, bounding boxes and segmented worms, respectively. Furthermore, we compared the effectiveness of including SAM segmentation optional module in experiments. The evaluation results have shown that YOLOv8 has excellent performance in worm detection to cope with deformable worm shape, and the modified DeepSORT significantly outperforms the default DeepSORT in worm tracking.

Keywords: Object detection and tracking · C. elegan mating behaviour analysis · deep learning.

1 Introduction

Background This study is concerned with applying computer vision technologies to discover mating behaviours of Caenorhabditis elegans (C. elegans) worms. C. elegans are diminutive and free-living nematodes that have emerged as vital model organisms across diverse scientific disciplines, including neurobiology, developmental biology, and genetics [19]. In the short developmental life cycle, typically three days, C. elegans undergo complete development from an embryo to a sexually mature adult [1]. This significant characteristic of C. elegans has made it popular in investigations to unveil correlations between C. elegans behaviour and the presence of environmental toxins in the soil [8]. By analysing

C. elegans mating behaviours, scientists intend to reveal soil conditions with regard to pollution [4]. Carefully setting experiments with a camera to capture C. elegans mating behaviours made the analysis possible purely based on videos. However, manual methods to observe the whole process are time-consuming, which restrict the scope and efficiency of comprehensive studies. An automated analysis tool based on computer vision technologies, such as object detection and tracking is sought after to discover the mating behaviours.

In recent years, with the development of computer vision technologies, deep neural networks are adapted in object detection and tracking. The emergence of YOLO [23], from the original YOLO to YOLOv8, enables accurate real-time object detection for robotics, autonomous vehicles, and video monitoring applications [27]. DeepSORT [28] has further advanced multiple object tracking with a focus on simple and effective algorithms, which integrate appearance information to improve the performance of the original simple online and real-time tracking (SORT) algorithm [3]. Alternatively, the Segment Anything Model (SAM) is an attempt to lift image segmentation into the era of foundation models [17].

Problem statement The videos taken in the C. elegans reproductive experiments contain six worms, ideally, in each frame, five females and one male. The duration of each video is about 18 minutes. The male worm is active and moving around to approach females, whilst female worms are relatively inactive but with motions. Fig. 1(a) shows six worms in a frame and Fig. 1(d) demonstrates the male worm has contacted a female worm after a few minutes in the video. From Fig. 1, it can be seen that positions and shapes of these worms are changing in the course of the process. This leads to a problem of deformable object detection and tracking. The active male worm may move out of the scene (see Fig. 1(b)), and imprints may appear in the scene (see Fig.1(b), (c), and (d)). These challenge the traditional approaches, e.g. appearance based object detection, motion based tracking, etc. [21, 5–7].

(a) (b) (c) (d)

Fig. 1: Examples of C. elegans positions in experiments (a) Initial positions, (b) Male left the scene, (c) Male back to the scene, (d) Male touching female 5

With the final aim of automated analysis of C. elegans mating behaviours, the initial objective of this study is to detect and record the contacting points between the male and a female worm in the 18 minute videos so that researchers can skip to these points to observe the mating behaviours. Deep learning methods, based on both YOLOv8 and DeepSORT, are considered in object detecting and tracking to investigate how the aforementioned challenges can be dealt with by advanced technologies.

Related work Using a computer to monitor nematodes movement was attempted in the 80's of the last century [9] with a 6809 microprocessor programmed in assembly language under a lighting condition of high contrast. In the area of C. elegans behavioural research, an array of innovative tracking systems and methodologies have emerged, each with distinct attributes and capabilities. Ramot, et al. [22] developed the Parallel Worm Tracker, a platform for quantitative analysis of C. elegans locomotion. This system is capable of tracking multiple worms in sequential video frames and recording their centroid positions. It is also adept at calculating the worm's speed and angular velocity. Simonetta, et al. [26] proposed an automated system that tracks the locomotor activity of C. elegans which is also suitable for circadian locomotion recording and research on aging mechanisms. The system utilizes light microbeams to detect worm movement and convert the frequency of the signal, allowing for a sophisticated analysis of locomotion patterns. Jaensch, et al. [13] presented an automated tracking and analysis system that offers exceptional accuracy in quantifying and tracking the size of Green Fluorescent Protein (GFP)-labelled centrosomes in early C. elegans embryos. It proves its efficiency by effectively processing large datasets with only minor manual corrections required. Dzyubackyk, et al. [10] introduced an algorithm designed for tracking C. elegans embryogenesis utilizing fluorescence microscopy images. The algorithm demonstrated successful segmentation and tracking of nuclei in the image sequence, surpassing the performance of existing methods. It was found to be efficient and user-friendly, employing graph-cut-based energy minimization for improved results. Restif, et al. [24] introduced CeleST, a sophisticated computer vision software tailored for automated tracking and in-depth analysis of C. elegans swimming behaviour. This innovative approach employs adaptive background subtraction to effectively discern and track individual nematodes within video frames, surmounting the challenges posed by intricate and diverse background environments. Despite its impressive capabilities, it is noteworthy that CeleST is designed to exclude worms that are in contact during the tracking process. This unique exclusion, while advantageous in some contexts, may present limitations when applied to specific studies that aim to explore the nuances of worm interactions and contact behaviour.

Javer, et al. [14], introduced the Tierpsy Tracker, a Python-based multi-worm tracker that extracts postural information from worm behaviour videos. By offering enhanced head-tail detection and locally calculated thresholds, the system provides an improved tracking accuracy. Lorimer, et al. [20] developed an approach that excels in detecting changes in worm locomotion behaviour through prediction error analysis. The algorithm localizes changes in worm locomotion behaviour and offers flexibility and sensitivity. Leonard and Vidal-Gadea [20] proposed a cost-efficient and user-friendly C. elegans tracking system. Although designed for classroom usage, the system delivers results almost rivalling more expensive professional systems, making it a cost-effective option for basic worm behaviour studies. Deep learning methods were employed by Banerjee, et al. [2] in their deep-worm-tracker for accurate detection and tracking C. elegans in worm behavioural studies.

Contributions Different from the existing tracking systems, the uniqueness of this study lies in detecting and monitoring C. elegan pairs engaged in mating interactions. This requires identifying individual worms and tracking them throughout their movements with position and appearance changes until the male worm touches one of the female worms. Limitations of robustness and reliability are still issues in dealing with occlusion and object lost/back in the scene by using traditional approaches. Curiously, this study investigates how an integrated deep learning approach copes with the limitations. In this paper, we introduce an automated system that aids the study of C. elegans mating behaviours analysis. The developed system, adapting YOLOv8 and modified DeepSORT, addresses the challenging task of detecting and tracking individual deformable worms, monitoring the trajectory of the male worm, and identifying mating occurrences in recorded videos. A contact detection mechanism is implemented efficiently reporting instances where worms overlap or make contact. The system's overall performance achieves a substantial level of accuracy in worm tracking and contact detection. The technical innovation of the integrated system is demonstrated in developing algorithms to modify DeepSORT, which significantly improves the performance in worm tracking and contact detection.

The rest of the paper is organised as below. Section 2 briefly describes YOLOv8 and DeepSORT. Section 3 introduces our technical approaches, including the system framework, data annotation, and details of implementation. Relevant experiments and testing results are demonstrated in Section 4. Section 5 concludes the study and lays the future work.

2 Preliminary: YOLOv8 and DeepSORT

YOLOv8: YOLO was introduced by Redmon, et. al. [23], representing a significant leap forward in object detection efficiency and effectiveness. The key innovation of YOLO lies in its ability to perform object detection in a single forward pass of various deep neural networks as backbones for different versions, from the original YOLO to recent YOLOv8, thereby achieving real-time processing speed. The network divides the input image into a grid, and each grid cell is responsible for predicting bounding boxes (objects). This grid-based approach significantly reduces the computational overhead, making YOLO highly efficient and suitable for real-time applications.

Essentially, the object detection task is framed as a regression problem, where the neural network predicts bounding boxes and class probabilities directly from the input image. Non-Maximum Suppression (NMS) is then used, which is a post-processing technique [12] to reduce the number of overlapping bounding boxes and improve the overall detection quality. From the original YOLO to recent YOLOv8, the algorithm also introduces anchor boxes to improve the accuracy of object detection. Anchor boxes are predetermined shapes of different aspect ratios that are used to refine the predicted bounding boxes. By introducing anchor boxes, recent versions, such as YOLOv8, are better equipped to handle objects of various shapes and sizes leading to improved localization accuracy.

As a state-of-the-art model, YOLOv8 [15] by Ultralytics was used in this study. Object detection algorithms of YOLOv8 generate multiple bounding boxes around the same object with different confidence scores, followed by NMS which filters out redundant and irrelevant bounding boxes, keeping only the most accurate ones. From input images, the YOLOv8 efficiently outputs a set of detected bounding boxes together with their sufficiently high confidence scores of object detection.

DeepSORT: DeepSort stands for Deep Learning-based SORT (Simple Online and Realtime Tracking), is an advanced object tracking algorithm that combines the principles of deep learning and traditional SORT to achieve highly accurate and efficient tracking results in real-time video sequences. Extended from SORT, a traditional online tracking method that uses Kalman filtering [16] and the Hungarian algorithm [18] for data associations, DeepSORT addresses the limitation of the SORT algorithm by incorporating two pieces of information, motion and appearance into its framework [28], named as $d^{(1)}$ and $d^{(2)}$, respectively.

The motion metric $d^{(1)}$ is implemented with the Mahalanobis distance between predicted Kalman states vector $\mathbf{d}_j$ of jth bounding boxes (consisting of its center position (u, v), aspect ratio r, height h as well as their respective velocities in image coordinates) and newly arrived measurements $\mathbf{y}_i$ of ith track.

$$d^{(1)}(i, j) = (\mathbf{d}_j - \mathbf{y}_i)^T \mathbf{S}_i^{-1} (\mathbf{d}_j - \mathbf{y}_i) \tag{1}$$

where the i-th track distribution is projected into the measurement space by $(\mathbf{y}_i, \mathbf{S}_i)$.

For each bounding box detection $\mathbf{d}_j$, an appearance descriptor $\mathbf{a}_j$ with $\|\mathbf{a}_j\| = 1$, is calculated based on a pre-trained convolution neural network(CNN) [28]. This approach trains appearance features offline with a large number of training samples on a convolution neural network. The appearance metric $d^{(2)}$ measures the smallest cosine distance between the i-track and j-th detection in the image space.

$$d^{(2)}(i, j) = \min\{1 - \mathbf{a}_j^T \mathbf{a}_k^{(i)} | \mathbf{a}_k^{(i)} \in \Re_i\} \tag{2}$$

where $\mathbf{a}_j$ denotes an appearance descriptor in j-th detection, and $\mathbf{a}_k^{(i)}$ represents appearance descriptors in the i-th image space $\Re_i$, which is maintained as the recent 100 associated appearance descriptors for each track.

Finally, the two metrics are combined as:

$$C_{i,j} = \lambda\, d^{(1)}(i, j) + (1 - \lambda) d^{(2)}(i, j) \tag{3}$$

where $0 \leq \lambda \leq 1$ is a hyperparameter. $C_{i,j}$ are used to determine if the detected bounding boxes are assigned to each track. For the details of DeepSORT implementation, refer to [28].

3 Methodology

Relevant system frameworks and algorithms are developed to overcome challenges that arise when dealing with visually similar entities such as C. elegans for their mating behaviours analysis based on videos. Fig. 2 depicts the system framework. YOLOv8, fine-trained by annotated C. elegans data, is for C. elegans detection. It automatically resizes and rescales the input image to match that of the images used for training the detector. The locations of objects detected in the input image are returned as a set of bounding boxes. A modified DeepSORT algorithm is implemented for multiple worm tracking. In the framework, SAM (Segment Anything Model) is optionally attempted to assist DeepSORT in accurate tracking with binary images [17]. For the technical details of SAM, which is beyond the scope of this paper, the readers are referred to [17]. In Section 4, we carried out an ablation study of SAM to obtain empirical results with or without SAM module in Fig. 2. The male worm is identified and tracked in each frame, even when it leaves and comes back to the scene. The system outputs two text files with information on the contact times of the male worm and a female worm, and coordinates of the male worm's trajectory, as well as videos with detected worms in each frame.

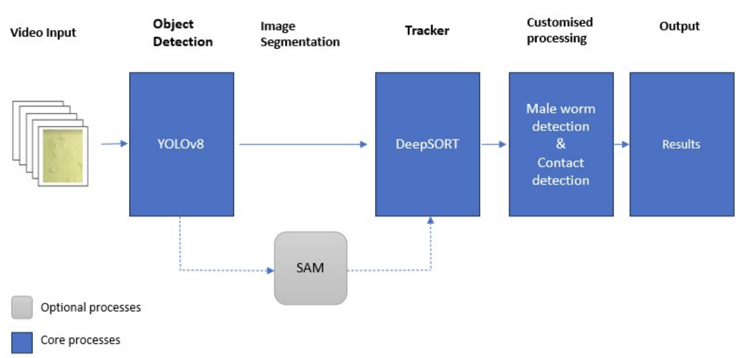
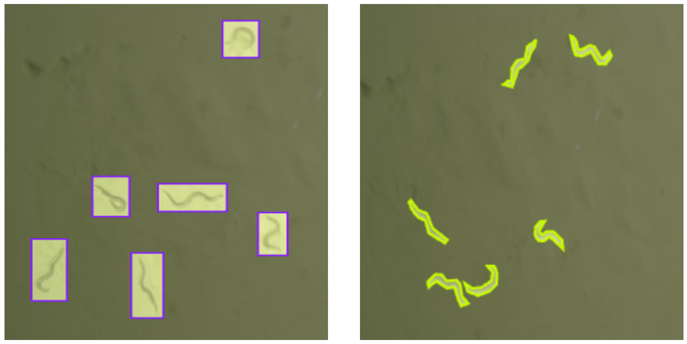


Fig. 2: Framework of the worm detection and tracking system.

3.1 Data annotation

To fine-train YOLOv8, annotated training samples are prepared from the C. elegans mating videos. This endeavour is expedited through the utilization of



(a) (b)

Fig. 3: Training samples for YOLO fine-train (a) Bounding box, (b) Segmented

the RoboFlow platform [25], which eased the annotation and data augmentation process. Two different versions of annotated training samples are collected, bounding boxes (version 1: V1) and segmented encapsulations (version 2: V2), as shown in Fig. 3 (a) and (b), respectively. The initial step involves the conversion of videos into images. This task was achieved through the utilization of the ffmpeg library [11], where images or frames are meticulously extracted from the videos at intervals, typically spanning every 5 to 10 seconds.

In the selection of training samples, diversity is the key element to ensure the robustness of a trained model. V1 contains 846 training samples, whilst V2 only has 208 training samples due to the process being highly time-consuming.

3.2 Modification of DeepSORT

A modified DeepSORT algorithm was proposed in this study to enhance the tracking accuracy. Due to high visual similarities between the worms, identity switching often occurs when these worms come in contact with each other or occlude themselves. This, in turn, affects tracking accuracy. In other instances, it would create a new ID for an existing worm. Knowing that videos in this study consist of 6 worms at maximum, we modified DeepSORT to limit the frequency in which new worm IDs are created after all 6 worms in the video have been identified and are being tracked.

To mitigate identity switches during worm interactions, where visual similarities can lead to confusion, we introduced dynamism to the number of frames for new track identification. The DeepSORT algorithm initiates a new track hypothesis for each detection that cannot be associated with an existing track. These nascent tracks are initially classified as tentative with the algorithm anticipating their consecutive appearance in a specified number of frames before their inclusion in the track set. Failure to meet this criterion results in discarding these tentative tracks. This predefined number for track identification can be adjusted prior to the program's execution. In modifying DeepSORT, we made the predefined number for track identification adjustable during program execution. At the program's onset, this number is set to a relatively low value (e.g., 10 frames) and once all six worms in the video are integrated into DeepSORT's track set, we dynamically increase the number of frames for new track identification to a substantial value (e.g., 400 frames). This means that after all 6 worms are identified if the YOLO model detects a $7th$ object (maybe a worm imprint as shown in Fig. 1), this $7th$ object (wrongly detected) will not be easily included in our modified DeepSORT's track set as opposed to the default DeepSORT. This dynamic adjustment significantly enhances tracking accuracy by minimizing the occurrence of identity switches.

SAM (Segment Anything Model) [17] is explored aiming to improve DeepSORT tracking accuracy. Although experiments with SAM had better tracking performance, the time taken to run such experiments is double that of the experiments without SAM (see Table 2). For this reason, we left SAM to be an optional process, for instances where speed is prioritized. YOLOv8 outputs bounding boxes after C. elegans detection. SAM is applied to each bounding box to segment C. elegan within it. Using this way, the segmented image and the bounding boxes are passed into the DeepSORT tracker. Fig. 4 depicts the outputs of YOLO and SAM in the system.

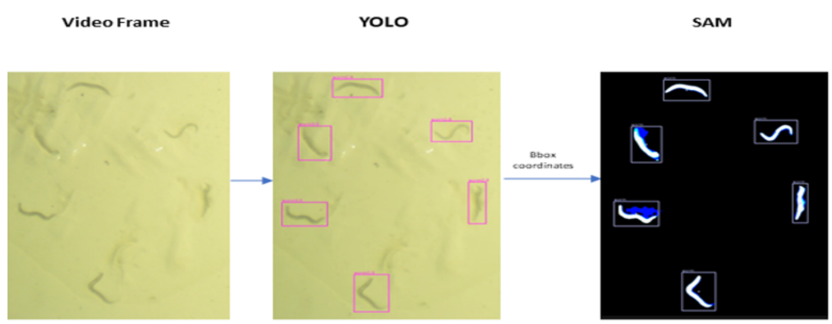


Fig. 4: Outputs of YOLO and SAM in the system

3.3 Male worm detection and contact detection

An integral objective of this worm detection and tracking system is to provide scientists with valuable insights into the moments when contact occurs between male and female worms. The visual similarity between male and female worms may cause problems for the appearance metric used in DeepSORT. A male worm detection algorithm is developed and implemented. As aforementioned, the male worm tends to have higher mobility and move more rapidly compared to those female worms in the videos taken from C. elegans mating events. This behavioural disparity presents a unique opportunity to leverage mobility as a discriminating factor for distinguishing between male and female worms. The approach for male worm detection is presented in Algorithm 1.

Algorithm 1 Male Worm Identification by Mobility

Require: Coordinates of bounding boxes in each frame. Total number of bounding boxes m. A specified frame number n.
Ensure: The male worm coordinates are identified.

1. For bounding boxes $k = 1, ..., m$ in a frame of the video
   (a) Obtain the centre coordinates $u_k^{(i)}$ and $v_k^{(i)}$ in the initial frame.
   (b) At a specified frame number (n):
      i. Obtain the current centre coordinates $u_k^{(c)}$ and $v_k^{(c)}$.
      ii. Compute the Euclidean distance D_k between the current and initial coordinates.

$$D_k = \sqrt{(u_k^{(c)} - u_k^{(i)})^2 + (v_k^{(c)} - v_k^{(i)})^2} \tag{4}$$

2. At frame (n+1), identify the worm with the highest Euclidean distance $\max\{D_k\}$ as the male worm.
3. Return

For contact detection, the initial way involves calculating the Euclidean distance between the centre points of two bounding boxes of worm pairs (one is the male worm). If this distance falls below a specified threshold, it unequivocally signals contact between the worms. The time of contact in the video was determined by multiplying the frame number and the frames per second of the video. While this method provides a quick, straightforward, and intuitive way for contact detection, it also demonstrates certain limitations. In scenarios where two worms are positioned in parallel, their bounding box centre points may exhibit proximity, leading to potential false positives in the contact detection process. To address the limitations, worm segments as connected components are attempted in contact detection. If any segmentation point from the male worm overlaps or is in proximity to any female segmentation point, it is indicative of contact between the two worms, and the time is recorded for the occurrence. The second method highly depends on the accurate segmentation of each worm in the frame. Otherwise, it may introduce contact detection errors.

4 Evaluation and results

Evaluation takes place in three folders, *i.e.* worm detection, tracking, and contact detection. A set of metrics is adopted. Mean Average Precision (mAP) and *mAP(50-95)* are for worm detection, whilst the Multiple Object Tracking Accuracy (*MOTA*) is used for evaluating worm tracking accuracy. The contact detection is done by comparing the system output with the ground truth manually obtained, and contact accuracy and contact $F1$ score are used as metrics.

4.1 Evaluation metrics

Mean Average Precision (mAP) is a commonly used evaluation metric in object detection tasks. It is defined in Equation 5

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{5}$$

where AP_i is the precision for detection class i, defined as $AP_i = \frac{TP_i}{TP_i+FP_i}$. N is the number of detection classes, in this case, six classes represent six worms in a scene. A higher mAP score indicates better performance in object detection, implying that the detection model excels in both precision and recall, where $recall_i = \frac{TP_i}{TP_i+FN_i}$. TP, FP, and FN stands for true positive, false positive, and false negative, respectively.

mAP(50-95) is an extension of the mAP metric, taking Intersection over Union (IoU) in the calculation. IoU is defined as:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{6}$$

mAP(50-95) measures the mean Average Precision over a range of IoU thresholds, typically from 0.5 to 0.95, in increments of 0.05. The following steps are involved in the calculation.

1. For each detection class and for each IoU threshold (beginning from 0.5, incrementing by 0.05 up to 0.95), we compute the AP.
2. The $mAP(50-95)$ score is then obtained by averaging these AP values across all detection classes and across all IoU thresholds (from 0.5 to 0.95).

A high $mAP(50-95)$ score indicates that the model performs well in object detection across various IoU thresholds. This metric is more reliable than the mAP which only uses an IoU value of 0.5, thus indicating that a higher $mAP(50-95)$ score will result in more accurate bounding box predictions.

Multiple Object Tracking Accuracy ($MOTA$) is an evaluation metric in the field of object tracking that provides a holistic assessment of a tracking system's ability to accurately track multiple objects over time. It is defined as:

$$MOTA = 1 - (FN + FP + M_{sw})/GT \tag{7}$$

where

- False Negatives (FN): quantifies the number of worms present in the frame which the tracking system fails to detect.
- False Positives (FP): encompasses objects erroneously identified as worms by the tracker, quantifying instances of incorrect worm detection.

– Male worm switches (M_{sw}): indicates whether the male worm is detected or switched with other worms in a given frame. This parameter carries a weight of 0.25 greater (+25%) than that of identity switches occurring within female worms.
– Ground Truth (GT): represents the actual number of worms present in the frame.

A high $MOTA$ score suggests that the tracking system is performing well in terms of accuracy in object tracking. This implies that it effectively tracks the majority of objects while minimizing missed detections (FN), false alarms (FP), and failure in detecting a male worm (M_{sw}) in a frame.

Contact accuracy in contact detection is defined as

$$\text{Contact accuracy} = \frac{\text{contacts detected}}{\text{overall contacts}} \tag{8}$$

In Equation 8, contacts detected refers to the number of contacts (between male and female worms) that were detected by the system in the testing video, and overall contacts is derived from the ground truth showing the real contact number.

Contact F1 score is expressed in Equation 9.

$$\text{Contact } F1 \text{ score} = \frac{2TP}{2TP + FP + FN} \tag{9}$$

where TP refers to contacts detected as contacts, FP; non-contacts detected as contacts,and FN, contacts that were not detected by the system.

4.2 System evaluation

Evaluation experiments were conducted by using a system with a $7th$-generation Intel processor, 16GB RAM, and a 6GB Nvidia Rtx2060 GPU. A pre-trained YOLOv8n (a nano-sized model) was adapted in the experiments. The fine-tuning of the model utilised the two sets of training samples, V1 and V2, described in Section 3.1. The learning rate was set as 0.01 and the default Adam optimizer is used in training. Two YOLOv8 models, M1 and M2, are established with V1 and V2, respectively. The training took less than an hour for M1 and over 4 hours for M2. Fig. 5 shows the losses against epochs in their training. In M1, box losses and class losses were involved in the training, whilst for M2, segmentation losses were also taken into account. Testing was conducted on both M1 and M2.

Worm detection evaluation In the evaluation experiments, we had three sets of testing data in each model. For Model M1, associated with the V1 dataset, 133 samples were derived along with the training samples from the three videos

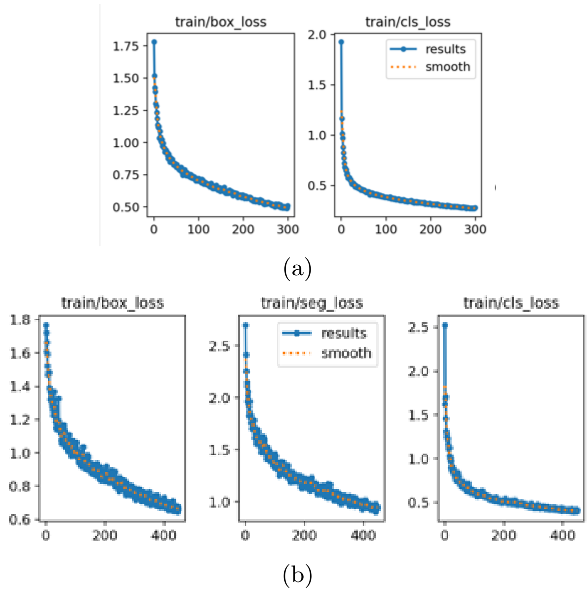


Fig. 5: Losses against epochs in training (a) Model M1, (b) Model M2

as described in Section 3.1; 167 samples were annotated from a new set of videos, and then we applied data augmentation to increase the testing dataset to 298 samples. For Model M2, associated with dataset V2, 46 testing samples were derived along with the training samples; 48 samples from a new set of videos, and 71 samples from data augmentation. As demonstrated in Table 1, M2 shows the consistency of *mAP* and *mAP(50-95)* scores for the three testing datasets. Interestingly for M1, the testing dataset derived from the same videos as the training samples produces a much better *mAP(50-95)* score than those from the testing datasets extracted from different videos. This may be caused by bounding boxes with larger percentage overlaps decreasing in these two testing datasets. Meanwhile, M1 yields superior *mAP* scores. The performance divergence can be attributed to several factors, mainly because of the dissimilarity in training data. It is worth noting that both the V1 and V2 training datasets comprise images extracted from three distinct C. elegans mating videos. However, a pivotal distinction emerges in the volume of training samples. The V1 dataset boasts a larger training sample size, amounting to 846 samples, in contrast to the V2 dataset, which comprises a modest 208 samples. On the one hand, it may indeed imply an enhancement in model performance, as a larger training dataset generally fosters better learning outcomes. Conversely, this scenario can also potentially lead to overfitting, where the model becomes excessively tailored to the idiosyncrasies of the three specific videos present in the V1 dataset. It underscores the significance of dataset diversity in achieving robust model generalization.

Table 1: Worm detection performance

Model	Training sample	Epochs in training	Batch size	Testing sample	mAP	mAP (50-95)
M1	846	300	16	133	0.992	0.858
				167	0.981	0.591
				298	0.980	0.587
M2	208	450	1	46	0.975	0.765
				48	0.960	0.714
				71	0.950	0.713

Worm tracking evaluation Building up on the YOLOv8 models is the DeepSORT for worm tracking with or without optional modular SAM. Ablation experiments were conducted with three different scenarios in worm tracking evaluation.

1. A system that uses the default DeepSORT algorithm.
2. A system that uses the modified DeepSORT algorithm.
3. A system that incorporates the Segment Anything Model (SAM) in conjunction with the modified DeepSORT algorithm.

Overall, 113,400 frames were involved in the worm tracking and contact evaluation, about 10.5 fps (frame per second). The results presented in Tables 2 and 3 were from a test of an eighteen-minute video. With different algorithms applied, the testing time was different. From Table 2, it is observed that with default DeepSORT, the M2 model, trained on a dataset comprising 208 images has the lowest $MOTA$. However, it attains the most commendable performance when coupled with the modified DeepSORT tracker and SAM. Table 2 has clearly shown that in both M1 and M2, the modified DeepSORT outperforms the default DeepSORT significantly. The modified DeepSORT with SAM incorporated gives better $MOTA$ scores, but the tracking operation took a longer time.

Table 2: Worm tracking performance

Model	Tracker	$MOTA$	$\mathbf{M_{sw}}$	Testing time
M1	Default DeepSORT	61.5%	9	50:16
	Modified DeepSORT	82.2%	4	51:06
	Modified DeepSORT (SAM)	85.2%	3	2:12:45
M2	Default DeepSORT	21.4%	30	51:44
	Modified DeepSORT	97.6%	4	56:23
	Modified DeepSORT (SAM)	**97.9%**	4	2:23:45

Worm contact evaluation Detecting contact between male and female worms was evaluated by implementing a manual recording process to establish the ground truth. It documents the time frames encompassing the male worm's interaction with a female worm, including the initiation and cessation of contact. A video of C. elegans mating behaviours was involved in the testing. Contact accuracy and F1 score defined in Equations 8 and 9 are demonstrated in Table 3. Due to the low performance in tracking, the default DeepSORT algorithm was excluded from the tests for both M1 and M2, only modified variants are evaluated for contact detection.

Table 3: Contact detection performance

Model	Tracker	Contact accuracy	Contact *F1* score
M1	Modified DeepSORT	91.7%	0.861
	Modified DeepSORT (SAM)	**93.7%**	**0.898**
M2	Modified DeepSORT	85.3%	0.738
	Modified DeepSORT (SAM)	73.4%	0.347

The M1 model uses bounding boxes for contact detection since it does not have segmentation coordinates while the M2 model uses the segmentation-based contact detection. Although the M2 model coupled with Modified DeepSORT and SAM has shown the highest performance in worm tracking, it exhibits an anomaly, which affects the contact detection accuracy. The reason for this misclassification is due to how image segmentation was implemented, which renders the worms in stark white against a pitch-black background. Thus, when the worms, displayed as white entities, converge, their contours are not clearly visible due to the overlapping white pixels, making it difficult to identify individual worms in contact. Although worm segmentation has shown an advantage in tracking isolated worms, it occasionally makes mistakes in detecting worms during contact. We observe that the bounding-box-based contact detection (with SAM) performs better than the segmentation-based methods. Interestingly, the M1 model with modified DeepSORT tracker and SAM produced the best results with respect to both contact accuracy and contact *F1* score on the video used in evaluation.

5 Conclusion and future work

In this study, we developed a computer vision system which works on worm mating videos to identify and track each worm involved, leading to worm contact detection. By adapting YOLOv8 and DeepSORT algorithms in the system, the modified tracker associated with the developed male worm detection and contact detection algorithms has achieved commendable contact detection accuracy on the evaluated video recording worm mating behaviours. The research problem involves deformable object detection and tracking. It has proved that the integrated deep learning approach can cope with the difficulty. The evaluation results have shown that YOLOv8 has excellent performance in worm detection to cope with deformable worm shapes in both M1 and M2. By incorporating SAM in the tracker, excellent tracking performance was achieved in M2 although substantial time is required in the operation, and the contact detection accuracy is also improved in M1. Future study includes two aspects, investigating a larger number of small sequences to gain more comprehensive views on mating behaviours and to investigate the role of segmented objects in contact detection. With the two approaches, it is expected to make the system more robust and efficient in coping with different scenarios.

Acknowledgements The project was initiated with the collaboration supported by the grant of the University of Reading NERC Discipline Hopping for Environmental Solutions. We would like to express our thanks to Nandini Vasudvan, Zuowei Wang, Eva Kevei, and Xingchen Zhai for providing the data and discussions over the project.

References

1. Altun, Z.F., Hall, D.H.: Introduction to c. elegans. In: WormAtlas (2009). https://doi.org/10.3908/wormatlas.1.1
2. Banerjee, S., Khan, K., Sharma, R.: Deep-worm-tracker: Deep learning methods for accurate detection and tracking for behavioral studies in c. elegans. Applied Animal Behaviour Science **266**, 106024 (2023)
3. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing (ICIP). pp. 3464–3468 (2016)
4. Bhagat, J., Nishimura, N., Shimada, Y.: Worming into a robust model to unravel the micro/nanoplastic toxicity in soil: A review on caenorhabditis elegans. TrAC Trends in Analytical Chemistry **138**, 116235 (2021)
5. Bradley, D., Roth, G.: Adapting thresholding using the integral image. Jounal of Graphics GPU and Game Tools **12**(2), 13–21 (2007)
6. Chauhan, A., Krishan, P., Kumar, D.: Moving object tracking using gaussian mixture model and optical flow. International Journal of Advanced Research in Computer Science and Sofware Engineering **3**(4), 243–246 (2013)
7. Chen, X., Wang, X., Xuan, J.: Tracking multiple moving objects using unscented kalman filtering techniques. In: International Conference on Engineering and Applied Science (ICEAS 2012) (2012)
8. Donkin, S.G., Dusenbery, D.B.: A soil toxicity test using the nematode caenorhabditis elegans and an effective method of recovery. Archives of Environmental Contamination and Toxicology **25**, 145–151 (1993)
9. Dusenbery, D.B.: Using a microcomputer and video camera to simultaneously track 25 animals. Computers in Biology and Medicine **15**(4), 169–175 (1985)
10. Dzyubachyk, O., Jelier, R., Lehner, B., Niessen, W., Meijering, E.: Model-based approach for tracking embryogenesis in *Caenorhabditis elegans* fluorescence microscopy data. In: Proceeding of Annual International Conference of the IEEE Engineering in Medicine and Biology Society. pp. 5356–5359 (2009)
11. FFmpeg-Developers: ffmpeg tool (version be1d324) [software] (2016), http://ffmpeg.org/ Last accessed 28 June 2024

12. Hosang, J., Benenson, R., Schiele, B.: Learning non-maximum suppression. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6469–6477 (2017)
13. Jaensch, S., Decker, M., Hyman, A.A., Myers, E.W.: Automated tracking and analysis of centrosomes in early *Caenorhabditis elegans* embryos. Bioinformatics **26**(12), i13–i20 (2010)
14. Javer, A., Currie, M., Lee, C., Hokanson, J., Li, K., Martineau, C.N., Zimmer, M.: An open-source platform for analyzing and sharing worm-behavior data. Nat. Methods **15**, 645–646 (2018)
15. Jocher, G., Chaurasia, A., Qiu, J.: Ultralytics yolov8, [software] (2023), https://github.com/ultralytics/ultralytics/ Last accessed 28 June 2024
16. Karavasilis, V., Nikou, C., Likas, A.: Visual tracking by adaptive kalman filtering and mean shift. In: Konstantopoulos, S., Perantonis, S., Karkaletsis, V., Spyropoulos, C., Vouros, G. (eds.) Artificial Intelligence: Theories, Models and Applications. pp. 153–162 (2010)
17. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., ..., Girshick, R.: Segment anything. In: Proceedings of the IEEE/CVF Internaional Conference on Computer Vision (ICCV). pp. 4015–4026 (2023)
18. Kuhn, H.: The hungarian method for the assignment problem. Naval Research Logistics Quaterly **2**, 83–97 (1955)
19. Leung, M.C., Williams, P.L., Benedetto, A., Au, C., Helmcke, K.J., Aschner, M., Meyer, J.N.: Caenorhabditis elegans: an emerging model in biomedical and environmental toxicology. Toxological Sciences **106(1)**, 5–28 (2008)
20. Lorimer, T., Goodridge, R., Bock, A.K., Agarwal, V., Saberski, E., Sugihara, G., Rikin, S.A.: Tracking changes in behavioral dynamics using prediction error. PLoS One **16**(5) (2021)
21. Ostsu, N.: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cyberetics **9**(1), 62–66 (1979)
22. Ramot, D., Johnson, B.E., Berry, B.J., Carnell, L., Goodman, M.B.: The parallel worm tracker: A platform for measuring average speed and drug-induced paralysis in nematodes. PLoS One **3** (2008)
23. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 779–788 (2016)
24. Restif, C., Ibáñez Ventoso, C., Vora, M.M., Guo, S., Metaxas, D., Driscoll, M.: Celest: Computer vision software for quantitative analysis of *C. elegans* swim behavior reveals novel features of locomotion. PLoS One **10** (2014)
25. Roboflow: Everything you need to build and deploy computer vision models (2023), https://roboflow.com/ Last accessed 28 June 2024
26. Simonetta, S.H., Golombek, D.A.: An automated tracking system for *Caenorhabditis elegans* locomotor behavior and circadian studies application. J. Neurosci. Methods **161**, 273–280 (2007)
27. Terven, J., Córdova-Esparza, D.M., Romero-González, J.A.: A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. Machine Learning and Knowledge Extraction **5**(4), 1680–1716 (2023)
28. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: 2017 IEEE international conference on image processing (ICIP). pp. 3645–3649. IEEE (2017)