Technical Report

Pattern Recognition and Image Processing Group Institute of Computer Graphics and Algorithms TU Wien Favoritenstr. 9/186-3 A-1040 Vienna AUSTRIA Phone: +43 (1) 58801 - 18661 Fax: +43 (1) 58801 - 18697 E-mail: dpucher@prip.tuwien.ac.at URL: http://www.prip.tuwien.ac.at/

 $\operatorname{PRIP-TR-135}$

June 21, 2016

2D tracking of Platynereis dumerilii worms during spawning

Daniel Pucher, Nicole M. Artner and Walter G. Kropatsch

Abstract

Platynereis dumerilii are marine worms that reproduce by external fertilisation and exhibit particular swimming behaviours during spawning. In this paper we propose a novel worm tracking approach that enables the 2D tracking and feature extraction during the spawning process of these worms. The gathered data will be used in the future to characterise and to compare male and female spawning behaviours.

Contents

1	Introduction to Platynereis dumerilii1.1 Properties of worm anatomy1.2 Organization of the Thesis	2 4 5			
2	Task formulation	6			
3	Existing tracking approaches	8			
4	Experimental setup				
5	Segmentation and tracking	11			
6	Shape representation	12			
	6.1 Definitions	12			
	6.2 Skeletonization and pruning	13			
7	Feature extraction	15			
	7.1 Curvature	17			
	7.2 Normalized shape	19			
8	Evaluation	21			
	8.1 Datasets	21			
	8.2 Morphological operations	22			
	8.3 Curvature estimation	24			
	8.4 Normalized shape	29			
	8.5 Temporal worm analysis	33			
	8.6 Runtimes	35			
9	Conclusion	38			
	9.1 Future work	39			

1 Introduction to Platynereis dumerilii

Platynereis dumerilii are marine polychaete worms (Lophotrochozoa, annelida, nereididae), which swim only when sexually mature, in order to reproduce. The timing of reproductive spawning events in this species is synchronized with the moon phase, whereby spawning in nature occurs primarily during new moon. This together with chemical pheromone signaling allows mature male and female worms to locate one another and engage in spawning behaviors that constitute the nuptial dance. See Figure 1 for an image of a male and female worm.



Figure 1: Image of a male (red) and female (yellow) worm.

The spawning behaviors of male and female worms are important for successful fertilization of the gametes. The spawning process consists of four general phases: Pre-spawning, engaged spawning, gamete release and postspawning. During pre-spawning, male and female worms typically swim independently of one another, usually with lower speeds, and display a linear body shape. Engaged spawning is initiated when male and female worms come into close contact and sense chemical pheromones secreted into the water by the opposite sex. This is accompanied by a noticeable change in swimming behavior for both sexes: swimming speeds increase (particularly for males), and worms either begin to swim in circles, or swim in tighter circles (particularly for females). Other changes in the plane of swimming are more frequently observed in both sexes during engaged spawning behavior. During gamete release, sperm and eggs are secreted into the water, which particularly for female worms, results in a dramatic change in body area, length and overall shape. The time individual spawning phases take varies and depends on the worms and their willingness to engage. Some worm pairs are better matches than others, which can result in shorter spawning phases.

This work is done in cooperation with biologists of the Max F. Perutz Laboratories GmbH. The goal of the biologists is to analyse these spawning behaviours in a quantitative manner, and to characterise and compare male and female-specific spawning behaviours.

1.1 Properties of worm anatomy

The body shape of the worms does not have any branches and is linear or curved, based on the movement. A worm is able to bend to form a circle. Length and area of its shape are ambiguous between male and female worms and can change over time.

Male and female worms can be distinguished by their color and anterior / posterior segment border, which can be seen in Figure 2.



Figure 2: Image of a female (top) and a male (bottom) worm with their segment borders (Scale in cm).

Worms are divided into a head and a tail part by the segment border and the position of the border is different for male and female worms. Relative to their whole body length, male worms have a longer tail than female worms. Therefore, the segment border is closer to the head.

During the different spawning phases the overall shape of the worms changes. Especially for female worms, the body length and the area changes during the gamete release as eggs are secreted into the water. During the different spawning phases the curvature also changes for both worms.

1.2 Organization of the Thesis

This thesis builds on a previous paper [11] submitted to the Computer Vision Winter Workshop 2016. Parts of the original paper have been reused for this work. The thesis is divided into the following sections:

- Section 1 gives an introduction to the Platenereis dumerilii worms, their spawning process and anatomical properties.
- Section 2 states the task of this work and describes the worm features that are extracted from the videos.
- Section 3 discusses existing methods for animal tracking.
- Section 4 gives an overview on the tracking setup.
- Section 5 explains the general segmentation and tracking approach.
- Section 6 gives some definitions that are used throughout the rest of the thesis and explains the skeletonization process in detail.
- Section 7 gives an overview of the feature extraction. The curvature estimation and computation of the normalized shape are explained in more detail.
- Section 8 discusses evaluation results.
- Section 9 concludes the thesis and gives an outlook on future work.

2 Task formulation

The aim is to develop methods that enable the tracking of spawning worms from captured videos and extract features to quantify behaviours. With this information a video should also be cut automatically depending on the spawning phases.

For the tracking, it is important to distinguish male and female worms in every frame of a captured video, label them and keep track of those labels. This thesis focuses on the extraction of features for the analysis of behaviours. The tracking task is simplified by only considering videos with single worms.

Based on the worm properties (see Section 1.1) and their behaviour (see section 1) the following features were selected together with the biologists:

1. Curvature

In general the coarse curvature of the worm's body provides information on the directionality of swimming. For example, a mostly straight linear profile would be indicative of linear swimming, while smoothly curved body profiles would indicate circular swimming. Good resolution of finer-scale body curvatures along the length of the worm is also important. For example, a linear profile with several bends could indicate an acceleration of swimming speed, or 'wriggling' movements, depending on the amplitude of the curvatures. Such wriggling movements can be seen in males when they are stopping to secrete sperm. Similarly, as gametes are released from the tail, mapping fine-scale curvatures at the tip of the tail could be used to map gamete release events, or characterize sex-specific gamete release behaviours. For example, fast small tail flicks in males during sperm release, and curling of the tip of the tail in females just prior to egg release have been observed.

2. Normalized shape

A normalized representation of the worms shape allows the comparison of different worms (or of the same worm at different times in a video). This gives the biologists a new method in analysing spawning behaviours independent of the anatomy of the individual worms.

3. Length and area

During the gamete release phase the length of the body and its area change, especially for female worms. Therefore, these features are a good indicator for the beginning of this phase.

4. Velocity

The velocity is a good indication for the beginning of the engaged spawning, as the swimming speeds increase for both sexes. Furthermore, it is valuable information for the tracking.

5. Trajectory

The trajectories give information on the interaction between the two worms. Furthermore, for individual worms, the curvature of the trajectory can be compared to the curvature of the worm. A high correlation indicates a circular movement and increases the robustness of the curvature estimation.

3 Existing tracking approaches

The tracking of animals and the extraction of features to quantify behaviours is not a new field of application. Caenorhabditis elegans (C. elegans) are roundworms that have been used as model systems in neuroscience for years and the demand for robust computational methods has lead to a number of different tracking systems like Nemo [14], OptoTracker [12] or a tracking system developed by Chatenay and Schafer [2].

These worm trackers are capable of tracking worms and extracting a variety of different features. Unfortunately, they are developed for C. elegans worms who differ in their appearance as well as their locomotion from Platynereis dumerilii. Furthermore, some of them are only capable of tracking single worms while others terminate tracking of animals if they collide and assigns new tracks after they separate again. This does not guaranty a continuous trajectory of a single worm for a whole video sequence, which is an important requirement for our behaviour analysis. Other animal tracking projects like AnTracks (www.antracks.org) or "Visual Ants Tracking" by Ying [15] are capable of tracking animals, but do not extract any features.

Therefore, this thesis proposes a new system that is capable of tracking Platynereis dumerilii worms and offers feature extraction including a new method to compute normalized shape forms.

4 Experimental setup

An existing setup at the laboratory is used for the worm tracker. It consists of a light-tight box, a mounted infrared camera and an ordinary PC to capture the videos. Worms are placed inside a cylindrical bowl refered to as arena. Figure 3 shows the arena with two worms.



Figure 3: Image of the arena with two worms taken from a captured video.

Videos are taken at a size of 1280x960 pixels with 43 frames per second. Since the spawning in nature occurs at night and the biologists want to reproduce this environment in the lab, it is important that the camera captures the videos in infrared light. Regarding 3D movements of the worms, the single camera setup has some limitations, as they might conceal parts of their body from the cameras viewpoint, resulting in a flawed representation. For example, the worms tend to turn sideways when moving fast. This results in a change of the body area as the worms are viewed from the side. In addition, the worms segment border (see Section 1.1) is not visible in such cases. Figure 4 shows three frames of the same worm in the same video just a few seconds apart. These frames illustrate frequent variations in the worms appearance.



Figure 4: In the first frame a) the worm turned sideways, therefore the segment border is not visible and the body area is different to the other frames b) and c). In those frames, the segment border is visible.

Changes in body area also happen when a worm overlaps itself and is therefore not entirely visible to the camera.

Analysis of spawning videos have shown, that worms mostly move horizontally near the water surface. Therefore, the collaborators decided to use this single camera setup and neglect the few cases where the gathered data is flawed due to 3D movement. Nevertheless, the setup might change in the future using three cameras instead of one to solve the issue with the 3D movement.

5 Segmentation and tracking

This work focuses on the tracking of single worms. Although, only single worms are segmented and tracked at the moment, it is still possible to analyse separate spawning behaviours in male and female worms. For this the biologists add eggs or sperm manually to the arena and the worms react to them. This allows to study isolated spawning behaviours.

The proposed segmentation and tracking approach assumes continuous movements of the worms. To track a single worm it is first segmented from the background. This is done with a simple background subtraction for every frame of the video. For the subtraction, it is important that there is at least one frame at the beginning of the video with an empty arena, which serves as the background image. As this image serves as the background image for the whole video, it is assumed that the arena does not move nor that the illumination changes during the video. This assumptions are met by the current setup (see Section 4).

After the background subtraction the resulting image is converted to a binary image, based on a global threshold level of 0.1. This threshold was defined empirically. To remove noise and smooth the boundary of the binary worm shape, a number of morphological operations are applied: morphological closing, to remove small holes, is followed by morphological opening, to remove small objects. A majority filter with a 3-by-3 neighborhood is applied to the binary worm shape after the opening and closing to create a smoother boundary. The majority filter sets a pixel to 1 (foreground) if the majority of pixels in its neighborhood have the value 1 (foreground) and to 0 (background) otherwise. The resulting binary image is a collection of regions that correspond to changes in relation to the empty arena. Ideally there is only one region representing the worm. As the worm produces some noise when moving in the arena (particles or bubbles in the water, reflections on the edge of the arena), there might also be some noise in the binary image. Therefore, only regions whose area is above a given threshold are considered as worms. This threshold is based on the minimum and maximum area of the worms in the test videos (see Section 8.1). As the regions generated by noise are very small, this approach works well with the current setup.

For the tracking, the worms are labeled at the beginning of a captured video. Then, the distance between the head positions in consecutive frames is calculated and the label is assigned based on the smaller deviation. This approach already works well for single worms, but it is too simple to track pairs of worms, as they tend to overlap and the distance of head positions alone is not a robust criterion.

6 Shape representation

6.1 Definitions

The skeleton describes the center line of a worm, is delimited by two endpoints and is represented by an ordered list of points in an 8-connected curve $S = \langle p_1, ..., p_{n_{skel}} \rangle$, $p_i = (x_i, y_i)$ with $i = 1, ..., n_{skel}$ where n_{skel} is the number of points on the skeleton.

An alternative and more compact encoding for the skeleton would be the Freeman chain code [6], but since the coordinates of the skeleton points are necessary to compute the curvature, chain codes are inappropriate.

The points p_i of the skeleton S are ordered from head to tail by comparing the endpoints of the skeleton in the current frame with the endpoints in the previous frame. The initial ordering is defined by the selection of head and tail through the user in the first frame.

Section 6.2 gives a detailed description of the skeletonization approach.

The head position p_{head} of a worm is the first point of the skeleton S: $p_{head} = (x_1, y_1) \in S.$

6.2 Skeletonization and pruning

Given the binary region of the worm, there are two common approaches to compute its skeleton: morphological thinning and morphological skeletonization with the medial axis transform (MAT) algorithm. Both approaches were tested and they give different results as the morphological thinning generates fewer spurious branches. See Figure 5 for a comparison between the two approaches for a sample worm.



Figure 5: Illustration of the worm skeletons (white) computed from the binary region (outlined in red). The left skeleton was computed using morphological thinning and the right one using skeletonization (MAT) technique.

Since the skeleton of a worm is defined as a center line between two endpoints, spurious branches are removed if the computed skeleton has more than two endpoints. A pruning algorithm is used, that preserves the longest overall branch of the skeleton, which conforms to the non-branching property of the worm. The algorithm currently only works if the original skeleton has 2 or more endpoints and no loops.

First, the longest overall branch of the skeleton is determined. This is done by computing the maximum geodesic distance on the skeleton for every endpoint. After these distances are computed, the endpoints associated with the largest geodesic distance give the endpoints of the longest overall branch. This branch corresponds to the diameter of the skeleton. If there are more than two endpoints with maximum geodesic distance, the longest overall branch is not unique. This can happen, if there are two branches with equal geodesic distance on one of the worms ends. In such cases, one of the endpoints is chosen arbitrarily. This arbitrary decision does not have an effect on the overall length of the worm. However, it effects the estimation of the curvature as both branches usually point in different directions.

After the two endpoints of the longest overall branch have been found, they are marked as the "true" endpoints. All other endpoints and their corresponding branches (a branch from an endpoint to the nearest branch point) are deleted. This process is repeated until there are two endpoints left. The result of this algorithm is the skeleton, the longest overall branch without any spurious branches.

After all spurious branches are removed, the difference between the two approaches (thinning and MAT) lies in the position of the endpoints and the skeleton line.

In this thesis, the skeleton obtained by the morphological thinning is used since the MAT approach creates more spurious branches and the result of pruning effects the estimation of the curvature if there is no unique longest branch in the original skeleton.

7 Feature extraction

Features are extracted for every frame of the captured video and are based on the binary region and/or the skeleton of a worm.

1. Curvature

The curvature is estimated along the skeleton S with a method based on osculating circles. Curvature values are computed for every point on the skeleton. Subsection 7.1 gives a detailed description of the curvature estimation.

2. Normalized shape

The normalized shape is computed using the skeleton S and a distance transform of the binary worm region. Distance values at the position of the skeleton give the minimum distance to the region border and are used as radii to draw the normalized shape using circles. Subjection 7.2 gives a detailed description on the computation of the normalized shape.

3. Length

The length of a worm is calculated with the geodesic distance between the endpoints of the skeleton S plus the radii of the circles at the first and last skeleton point. It is important to include the circle radii as the skeleton endpoints do not always touch the outline of the binary worm region.

4. Area

The area of a worm is the sum of all foreground pixels of the binary worm region. It is the zeroth moment.

5. Velocity

Velocity v is defined as the rate of change of position with respect to time $v = \frac{\Delta x}{\Delta t}$. The change in position Δx is measured between two consecutive frames and is given by the Euclidean distance between the two head positions p_{head} in the frames. The time Δt is given by the time between two frames.

6. Trajectory

The trajectory of a worm in a video sequence is described by a chronological list of head positions p_{head} from every frame of the video sequence. The tangent of the trajectory provides information on the direction of the worms movement and its curvature provides information on circular movements. Figure 6 shows a section of a trajectory from a single worm video.



Figure 6: Part of a video frame with an overlay of the head trajectory. The trajectory is taken from a short sequence of a single worm video.

7.1 Curvature

According to Hermann and Klette [9], the estimation of the curvature along a discrete curve can roughly be divided into three categories: the derivative of the tangent angle, the derivative of the curve and the radius of the osculating circle. In this thesis a method based on osculating circles is used as it is fast and the implementation is simple. Gray [7] defines the osculating circle of a curve C at a given point P in the continuous space as the circle that has the same tangent as C at point P as well as the same curvature. These circles are approximated with the circumscribed circles of triangles on the discrete skeleton curve. Casey [4] defined the circumscribed circle as the unique circle that passes through each of the triangle's three vertices.

Given the definition for the skeleton S, let the neighbourhood k be $1 \leq k \leq \lfloor \frac{n_{skel}}{2} \rfloor$ if n_{skel} is odd and $1 \leq k \leq (\lfloor \frac{n_{skel}}{2} \rfloor - 1)$ if n_{skel} is even. For each point p_i on S a triangle is defined between the three points p_{i-k} , p_i and p_{i+k} . Then the radius r of the triangles circumscribed circle allows to calculate the curvature at p_i by $\frac{1}{r}$. See Figure 7 for a visualization.



Figure 7: Illustration of the circumscribed circle (blue) for a single point p_i on a skeleton. The circle passes through every vertex of the triangle (red) formed by the points p_{i-k} , p_i and p_{i+k} with k = 10.

The radius of the circumscribed circle is defined as $r = \frac{abc}{4*area}$, where a, b and c correspond to the side lengths of the triangle and *area* is the area of the triangle. The area of a triangle is given by:

$$area = \frac{1}{2} * \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}$$

with $(x_1, y_1) = p_{i-k}, (x_2, y_2) = p_i$ and $(x_3, y_3) = p_{i+k}$

The sign of the area gives an indication on the orientation of the triangle and therefore an indication on the direction of the curvature. Finally, the curvature c is given by the inverse of the radius $c = \frac{1}{radius}$. It is positive if the curvature is on the right side and negative if the curvature is on the left side of the skeleton curve. See Figure 8 for a visualization of the curvature.



Figure 8: Image of a worm with its skeleton (top) and a plot of the estimated curvature of the worm for a fixed neighbourhoods k (bottom).

7.2 Normalized shape

To compute the normalized shape representation, we follow a recent strategy which is known as co-registration, where shapes are first straightened or flattened to then register different views/deformations of the same normalized shape [1].

The normalized shape representation of a worm is achieved with a backward medial axis transform approach. It is based on the distance transform of the binary worm image, which labels each pixel of the worm with the Euclidean distance to the nearest boundary pixel. For every point $p_i = (x_i, y_i)$ of the skeleton S, the corresponding value of the distance transform is looked up. Those distances then serve as the radii for the circles and are related to the radii of the circles used by MAT. See Figure 9 for a visualization.



Figure 9: Part of the distance transform of a worm with circles drawn for four points on the skeleton.

Since the original skeleton of a worm might have spurious branches that are removed during the skeletonization process, the distances for the spurious branches are not looked up, causing inaccuracies in the normalized shape representation. For the normalized shape representation, the distances between the circle center points correspond to the distances between these points on the skeleton. This way the proportions are maintained. Figure 10 shows the results of this method, where in the first visualization only the outlines of a sub-sampling of circles are drawn to show the general idea.



Figure 10: Plots of the normalized representation of a worm using the outlines of 24 circles to visualize the general idea (top) and a complete shape visualization with all 115 filled circles (bottom) of that worm.

8 Evaluation

8.1 Datasets

For the evaluation two datasets were used. The first dataset (referred to as "video dataset") consists of 7 different single worm videos with female worms only. All videos have a resolution of 1280x960 pixels and a framerate of 43 frames per second. Table 1 shows additional information of the video dataset.

video	duration	max worm area	min worm area
1	01:05 min	2674 px	890 px
2	01:24 min	3351 px	1260 px
3	$00:44 \min$	3640 px	201 px
4	01:29 min	$6071 { m px}$	1396 px
5	$01:37 \min$	2104 px	466 px
6	03:02 min	2486 px	$550 \mathrm{px}$
7	02:07 min	3219 px	821 px

Table 1: Video dataset

The second dataset (referred to as "image dataset") consists of 100 images selected from the video sequences of the video dataset. The images show the single worms in different stages of the spawning process. Figure 11 visualizes the body area of every worm in the image dataset.



Figure 11: Body area for all images in the image dataset. The subdivision (dotted lines) corresponds to the 7 different videos the images were taken from.

8.2 Morphological operations

Three different morphological operations are used to remove noise and smooth the boundary of the binary worm image: opening to remove small objects (noise), closing to remove small holes and majority to smooth the boundary of the worm region.

To evaluate the morphological operations, the difference between the original binary worm region and the morphed region was computed for every worm image in the image dataset. The difference was evaluated for two cases: The first case describes the pixels that were added to the original worm region and the second case describes the pixels that were removed from the original worm region. Figure 12 shows the changes to the original worm region after all morphological operations have been applied.



Figure 12: Changes after a morphological closing followed by a morphological opening and majority is applied on the original worm region.

Figure 13 shows the two best and the two worst examples of worm regions after the morphological operations.



Figure 13: Images of the best and worst results obtained by the morphological operations. Pixels that were added to the original worm region are highlighted in green, pixels that were removed are highlighted in red.

8.3 Curvature estimation

An important factor in the accuracy of curvature estimation is the neighbourhood k. A discrete circle with a radius of 40 pixels created with Bresenham's circle algorithm [3] was used to test the accuracy. Figure 14 shows the results. As the error gets too big for small neighbourhoods, it starts at $0.05 * n_{skel}$. With increasing neighbourhood k it can be observed that the error gets smaller. The same is true for a constant neighbourhood k but an increasing radius, which corresponds to the multigrid convergence theorem, where the accuracy is expected to increase as the grid resolution (or in our case the circle radius) increases [10, Chapter 10].



Figure 14: Plot of the avg- and max-error for the curvature estimation of a circle with radius 40 and increasing neighbourhood k.

So the accuracy gets better with increasing k. Unfortunately, this accuracy comes at a price, as small curvatures are overlooked in the process if k is too big. This corresponds to the sampling theorem as the neighbourhood defines a sampling window for the curvature estimation. Another problem with a constant neighbourhood k are points at the beginning and the end of the skeleton curve. For points p_a with $a \leq k$ there are no neighbourhood points p_{a-k} defined. The same is true for points p_b with b > n - k where no neighbourhood points p_{b+k} are defined. This problem is currently solved by disregarding those points on the curve.

To evaluate the accuracy for different curvatures and alternating curvature directions, a synthetic line was created. The line is composed of 7 semi-circles with different radii who are connected horizontally (see Figure 15). The ground truth of the line can be computed with the radii of the semi-circles.



Figure 15: Synthetic line composed of 7 semi-circles.

Figure 16 shows the effect of different neighbourhoods k on the estimation of the curvature. The chosen neighbourhoods correspond to the optimal neighbourhoods $\frac{r*\pi}{2}$ for the semi-circles 2 (k = 219), 4 (k = 125) and 6 (k = 31).



Figure 16: Estimation of the real cruvature (gt) with three different neighbourhoods k.

With k = 125 and k = 219 the estimation at the center of the larger semi-circles (1,2,3 and 4) is better in comparison to k = 31. Unfortunately, this also results in a higher error in the transitions between two consecutive semi-circles. Furthermore, smaller curvatures are overlooked. With k = 31the overall estimation is more accurate although the curvature estimation for the smallest semi-circle (7) is still error prone. Therefore, the same evaluation is repeated with a smaller neighbourhood k.

Figure 17 shows the results of the curvature estimation with an optimal neighbourhood k = 15 for the smallest semi-circle (7). This shows a good estimation for the two smallest semi-circles (6,7), but it also results in higher errors for the larger semi-circles.



Figure 17: Estimation of the real cruvature (gt) with a small neighbourhood k = 15 px.

Figure 18 shows the curvature estimation with the optimal neighbourhood k = 282 for the largest semi-circle (1). This results in higher errors for the estimation of the smaller circles and errors in the transitions.



Figure 18: Estimation of the real cruvature (gt) with a large neighbourhood k = 282 px.

To evaluate the effect of discretization on the results, a second synthetic line was created by rotating the line in Figure 15 by 45 degrees counterclockwise. Figure 19 compares the result for the curvature estimation of both lines with a fixed neighbourhood k = 31.

On the tested synthetic line the neighbourhood k = 31 gives the best result and shows good estimations for all semi-circles but the smallest (7). For the smaller semi-circles (6,7) the neighbourhood k = 15 shows better results. Since the possible bending radii of the worms depend on their size, the overall accuracy can be improved by adjusting the neighbourhood k to the size of the worm when computing the curvature. To restrict the actual neighbourhood, 31 is set as an upper bound and 15 as a lower bound.

In this thesis, the value for the neighbourhood k is defined with respect to the number of skeleton points as $k = max(15, min(31, n_{skel} * 0.15))$. The factor 0.15 was defined empirically.



Figure 19: Estimation of the real cruvature (gt) with a neighbourhood k = 31 px for both lines.

8.4 Normalized shape

To evaluate the normalized shape representation the original binary worm region is compared with a shape projection. The shape projection is created by drawing circles for every point p_i of the original skeleton S with the associated radius obtained by the normalized shape computation. Figure 20 shows the error rates of the shape projection compared to the original worm body area. The minimum error is 1.02 %, the maximum error is 8.89 % and the mean-error for all images is 3.2 %. Figure 21 shows the two best and the two worst results for the shape projection.



Figure 20: Error rate of the shape projection for every image of the evaluation dataset. The subdivision corresponds to the different videos the images were taken from.



Figure 21: Images of the two best (a and b) and the two worst (c and d) results obtained by the shape projection. The projected shapes are drawn in white and the original worm body areas in red.

The results shown in Figure 21 correspond to the images 3 (a), 40 (b), 45 (c) and 83 (d) of the evaluation dataset. The worm in image 83, where the shape projection produces the biggest error, is one of the smallest worms in the dataset (see Figure 11). This suggests, that the error rate of the normalized shape correlates with the worm size. Therefore, the errors for other worms that are of similar size were checked. Table 2 shows the results which indicate that there is no correlation between the size of a worm and the accuracy of the normalized shape representation.

image	area	error
80	640 px	2.97~%
81	601 px	3.00~%
82	614 px	6.35~%
83	630 px	8.89~%

Table 2: Error values for the shape projection

Since the normalized shape is computed with a backward medial axis transform approach, it seems natural to assume that the shape projection is free of errors when based on an already projected shape. Experiments have shown, that this is not the case. Figure 22 shows the error rates of another shape projection based on the first projected shape. The majority of shape projections is error prone.



Figure 22: Difference in the area of the second shape projection compared to the first projected shape (blue). Difference in the area of the first projection compared to the original worm area is also shown (red). The subdivision corresponds to the different videos the images were taken from.

This experiment was repeated 20 times, revealing that the shape projection gets better after some iterations for the majority of shapes in the dataset. Figure 23 shows the error rates after the 7th iteration.



Figure 23: Error rate of the 7th shape projection compared to the 6th projected shape. The subdivision corresponds to the different videos the images were taken from.

These experiments show that the normalized shape projection is not the inverse of the MAT and depends on the shape of the worm. Figure 24 shows the shape projections for the two worst cases of the image dataset that still show a difference of the area in the shape projection after 20 iterations.

For 96 out of 100 images the shape projection has a 0% area difference in the 20th iteration. Independent of the worm shape and the iteration, the difference in % with regard to the previous projection is always below 9%.



Figure 24: Change of the shape projection after different iterations. The red pixels highlight the changes to the previous picture.

8.5 Temporal worm analysis

Temporal experiments were performed on the video dataset to show changes of features over time and how they correspond to the spawning phases. Around the time of the gamete release where the female worms secrete their eggs into the water they get smaller and therefore shorter. This can also be observed in plots of the worm length and gives the biologists information on the phase of the spawning process without having to look at the video material.

Figure 25 shows an example of the change of the worm length between frames 600 and 1000 where the gamete release starts. To hide outliers of the original worm length in the plot, the data was smoothed with a moving average filter.



Figure 25: Smoothed worm lengths for a female worm right around the time of the gamete release (marked in red).

Figure 26 shows an example on how the length of a female worm changes during an entire spawning process. For this plot the original data was not smoothed to also show special outliers. Annotation A marks a special case where the worm is overlapping itself resulting in a faulty binary area and skeleton. The problem here is the 3D movement of the worm.



Figure 26: Change of worm length over time. During the gamete release the worm gets shorter. Annotations A and B mark special cases.

Another special case, where the 3D movement results in erroneous data is marked with annotation B. Here the end of the tail is not visible to the camera which makes the worm appear shorter in the video.

8.6 Runtimes

Runtimes of the main operations for the segmentation, tracking and feature extraction were computed for all videos in the videos dataset. Figure 27 shows the mean runtimes per frame in a barchart. Figures 28 and 29 show boxplots of the runtimes.



Figure 27: Barchart of the mean runtimes per frame of the main operations for the tracking and feature extraction.

For the background subtraction the runtime is constant for all videos as it is performed on the whole image, i.e. independent of the worm. Blob detection depends on the amount of noise that is present after the background subtraction and the skeletonization depends on the shape of the worms. Both are different for every video in the dataset.

Computing the normalized shape and curvature has the same runtime for all videos and depends on the number of skeleton pixels of the worms.

For an input video, the mean runtime for all operations is 4.8 times real time.



Figure 28: Boxplot of the runtimes per frame for the operations background subtraction (1), blob detection (2), skeletonization (3), normalized shape (4) and curvature (5).



Figure 29: Boxplot of the runtimes per frame for the operations background subtraction (1), blob detection (2), skeletonization (3), normalized shape (4) and curvature (5).

9 Conclusion

This thesis presents a novel approach for the analysis of the spawning process of Platynereis dumerilii worms, that enables both tracking and feature extraction in captured videos.

The tracking is new for this type of worms, as alternative approaches focus on different worms or other animals. The feature extraction also offers a new method to compute normalized shape representations that follows a recent strategy, which is known as co-registration. This can give the biologists a new method to study the changes in the worms shape during the spawning process independent of their deformation.

The accuracy of the normalized shape representation depends on the original worm shape in the video and can be free of errors in the best case. In general, the error was always below 9% of the original shape for all tested worms.

The evaluation of the curvature estimation has pointed out the importance of the neighbourhood parameter k and its influence on the accuracy of the estimation. One mature flaw is also the lack of curvature information at the ends of the skeleton. This information is important to the biologists, since tail flicks in male worms and curling of the tip of the tail in female worms can be observed during gamete release.

Temporal experiments on worm videos have shown how extracted features can help to identify the different phases of the spawning process.

9.1 Future work

1. Tracking of two worms

The method currently used to track single worms also works for two worms if they are physically separated, but as they get close to each other or overlap, the current method might fail. In the future, the method will be extended to consider cases where the worms are close to each other or even overlap. Ideas to achieve this include the comparison of more features than just the head positions of consecutive frames. A combination of all other features could yield an appropriate approach in distinguishing male and female worms.

2. Segmentation

There exist special cases, where a single worm overlaps itself due to 3D movement in the water. This results in regions and skeletons, which do not represent the worm correctly and therefore the extracted features are flawed as well. One solution could be the watershed method [8] as it might be superior to the simple threshold based segmentation method.

3. Skeletonization

The skeletonization works only if there is no loop in the original skeleton. Such a loop might occur if the worm forms a circle. This will be addressed in the future.

The pruning approach currently chooses endpoints of the skeleton arbitrarily if there is more than one branch in the skeleton with maximum geodesic distance. This negatively influences the curvature estimation if the wrong endpoints are chosen.

4. Curvature estimation

The curvature estimation has some flaws and is not robust enough. In the future, we will look into alternative approaches to compute the curvature of discrete curves. Roussillon and Lachaud [13] base their method around maximal digital circular arcs. Other methods rely on digital straight segments (DSS) recognition [9] [5].

References

- N. Aigerman, R. Poranne, and Y. Lipman. Lifted bijections for low distortion surface mappings. ACM Trans. Graph., 33(4):69:1–69:12, July 2014.
- [2] J. B. Arous, Y. Tanizawa, I. Rabinowitch, D. Chatenay, and W. R. Schafer. Automated imaging of neuronal activity in freely behaving caenorhabditis elegans. *Journal of Neuroscience Methods*, 187(2):229 234, 2010.
- [3] J. Bresenham. A linear algorithm for incremental digital display of circular arcs. Commun. ACM, 20(2):100–106, Feb. 1977.
- [4] J. Casey. A sequel to the first six books of the elements of Euclid, containing an easy introduction to modern geometry, with numerous examples. 7th edition, revised and enlarged. Dublin. Hodges. University Press Series. 184 S, (1895).
- [5] D. Coeurjolly, S. Miguet, and L. Tougne. Discrete Curvature based on Osculating Circles Estimation, 2001. 4th International Workshop on Visual Form 2001, Capri, Italy, Springer Lecture Notes in Computer Science 2059, pages 303-312.
- [6] H. Freeman. On the encoding of arbitrary geometric configurations. *Electronic Computers, IRE Transactions on*, EC-10(2):260-268, June 1961.
- [7] A. Gray. Modern Differential Geometry of Curves and Surfaces with Mathematica. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 1996.
- [8] B. Han. Watershed segmentation algorithm based on morphological gradient reconstruction. In Information Science and Control Engineering (ICISCE), 2015 2nd International Conference on, pages 533–536, April 2015.
- [9] S. Hermann and R. Klette. A comparative study on 2d curvature estimators. In *Computing: Theory and Applications, 2007. ICCTA '07. International Conference on*, pages 584–589, March 2007.

- [10] R. Klette and A. Rosenfeld. *Digital Geometry*. Morgan Kaufman Publishers, 2004.
- [11] D. Pucher, W. G. Kropatsch, N. Artner, S. Bannister, and K. Tessmar-Raible. 2d tracking of platynereis dumerilii worms during spawning. In *Proceedings of the 21st Computer Vision Winter Workshop*. Slovenian Pattern Recognition Society, 2016.
- [12] D. Ramot, B. E. Johnson, T. L. Berry, Jr, L. Carnell, and M. B. Goodman. The parallel worm tracker: A platform for measuring average speed and drug-induced paralysis in nematodes. *PLoS ONE*, 3(5):e2208, 05 2008.
- [13] T. Roussillon and J.-O. Lachaud. Accurate curvature estimation along digital contours with maximal digital circular arcs. In 14-th International Workshop on Combinatorial Image Analysis (IWCIA), LNCS, pages 43–55. Springer, 2011.
- [14] G. D. Tsibidis and N. Tavernarakis. Nemo: a computational tool for analyzing nematode locomotion. BMC Neuroscience, 8(1):1–7, 2007.
- [15] F. Ying. Visual ants tracking. PhD thesis, University of Bristol, 2004.