**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: csansone@prip.tuwien.ac.at URL: http://www.prip.tuwien.ac.at/

PRIP-TR-139

July 26, 2016

An algorithm for tracking swimming worms

Carmine Sansone

### Abstract

During spawning, the marine worms Platynereis dumerilii exhibit certain swimming behaviours, which are described as a 'nuptial dance'. We propose a 2D tracking approach that enables the extraction of data to quantify behaviours. With Platynereis dumerilii worms, associations based on appearance are challenging since they are deformable and regularly change shape and luminosity depending on their movement. During the spawning phase, the worms exhibit complex interactions leading to occlusions which interrupt the continuous track of a worm. During the occlusion the worms overlap themselves several times and change the movement direction. To maintain the individual identities within a video sequence, we propose an approach where appearance models are used to compare a set of feature before and after an occlusion occurs to get correct associations of worm identities. The chosen features are: the normalized shape (which is computed using a novel approach), the area, the length, the luminosity and the position. To correctly define the head and the tail points of the worms a trajectory analysis is done and depending on the movement of the worms the head and the tail points are detected.

The method that we propose can be easily re-adapted to track similar type of worms. The evaluation done shows how promising is this new approach for the tracking of two Platynereis dumerilii worms.

# 1 Introduction

Platynereis dumerilii [1, 2] is a particular marine worm that swims only during the reproduction process. In this species, the timing of reproductive spawning events is synchronized with the moon phase. It is interesting to understand how the molecules of this worms (genes, proteins and hormones, as well as signalling pathways and cell types) are involved in the regulation and the synchronization of the reproduction time according to the moon phase. [5, 6]. Descriptive studies of the spawning behaviour of Platynereis, and other related Nereid worm species exist and date back to the early 1950s. These previous investigations have also identified some key chemical pheromones that stimulate egg and sperm release and induce spawning behaviour itself. For example, female worms secrete uric acid and this is known to stimulate sperm release from male worms. Similarly, different stereoisomers of 5-methyl-3-heptanone stimulate spawning activity in male and female worms [7]. However, nothing is known about the genes, proteins and neuronal circuitry in the brain and peripheral nervous system that connect the detection of spawning pheromone signals to the stimulation of spawning behaviours, and ultimately the release of gametes into the water for fertilization.



Figure 1: Image of a female (top) and a male (bottom) worm (taken from [6]).

This is interesting because many marine animals synchronize their reproductive spawning events with the lunar cycle [9]. The Platynereis dumerilii worms are easy to grow in the laboratory and can be used for controlled experiments, so they make an ideal model system for studying the underlying mechanism of lunar reproductive timing. Figure 1 shows a colour image of two Platynereis dumerilii worms.

The goal is to develop a quantitative spawning analysis method that is able to characterize the differences in behaviour of male and female worms during spawning. To define the behaviour of the worms, some worm features need to be extracted from the video sequence and stored in appropriate data structure: speed, trajectory, area, length and number of touches. The extracted data can be analysed from the biologists to experimentally test the candidate genes, proteins, hormones and pheromones to understand their importance and role in regulating spawning behaviours.

Currently biologists are obligated to watch all the video sequences and write by hand a report about the behaviour of the worms. There are two main problems with this procedure:

- This procedure is time consuming for a human. At the moment the biologists cannot analyse video sequences with a long duration like twenty-four hours but just videos that have a short duration like ten minutes. As consequence of this problem no one has analysed the worms when they are not mating. There is some interest to analyse the worms during the entire life cycle.
- This procedure is time consuming for a human. At the moment the biologists cannot analyse video sequences with a long duration like twenty-four hours but just videos that have a short duration like ten minutes. As consequence of this problem no one has analysed the worms when they are not mating. Nonetheless there is some interest to analyse the worms during the entire life cycle.
- In the reports written by hand there is an approximate description about the behaviour of the worms and it is hard to compare the behaviour of the worms in different video sequences. The evaluation of a video done by a human is not properly formalized in quantitative measurements but just with a qualitative description. Comparing the

results of different analyses is even more difficult when the reports are written by different researchers. The problem is that the values of the desired features cannot be measured by a human so the final comment about the behaviour of the worms depends also on the sensibility of the operator (the worms move fast/very fast, the worms do small/big circle, etc.).

For these reasons an algorithm that is able to automatically define the behaviour of the worms in a formal way is needed.

# 1.1 Overview of this document

The first chapter of the thesis explains the motivation of this work and the importance of the research done on the genetic of the Platynereis dumerilii worms. Some basic concepts regarding the behaviour and the constitution of this worms are given. After this introductory information, the challenged problem is defined from an high point of view as well as the role of this work in the research of the biologists that are working on the genetics of the Platynereis dumerilii worms.

The second chapter provides an overview of the best practices and the common methodologies used in the tracking problematics. These techniques include specifications regarding: representation of background, representation of foreground objects, extraction of the features used to model the tracked objects and Kalman filter: how it works and how to use it. After this general overview some among the most important algorithms, in the literature and on the marker, that aim at the worms tracking are given. Each of these algorithms is compared to the algorithm proposed in this thesis remarking why it cannot be used to solve the tracking of the Platynereis dumerilii worms.

The third chapter defines in detail the proposed method and how it is organised in modules. The main three modules are used respectively to achieve the following objectives: worms detection, features extraction and worm tracking. An analysis of the computation complexity of the method is given and some optimization about the execution time and the needed space to store the output of the algorithm are proposed. At the end of the chapter the limits of the proposed method are defined. In particular the class of problems that can be solved using this algorithm are defined.

In the fourth chapter the proposed method is designed to be implemented in any procedural programming language. Every parameter of the algorithm is described defining how the output of the elaboration changes depending on every parameter variation. This design has been used to implement a Matlab application, now used by the biologists of the Max F. Perutz Laboratories of Vienna.

The fifth chapter show the results obtained by the algorithm on the provided data set that is composed by twenty-five videos for a duration of more than two hours. In the first part of the chapter the standard measurement indexes used to evaluate the algorithm are described. In the last part of the chapter the obtained results are organized in tables.

The last part of the thesis is dedicated to the conclusion and the future work prospective.

This work has been done during my Erasmus experience in Vienna as a collaboration between the Pattern Recognition and Image Processing (PRIP) research group of the Institute of Computer Graphics and Algorithms in Vienna and the MIVIA research group of the Department of Information Engineering, Electrical Engineering and Applied Mathematics in Salerno 1. The Max F. Perutz Laboratories (MFPL) research group of the Vienna Biocenter has participated to this work proposing the problematic and providing the videos data set. Contents, figures, plots and tables present in this thesis are also present in the paper Shape Normalizing and Tracking Dancing Worms submitted to IAPR International Workshops on the Structural and Syntactic Pattern Recognition (SSPR 2016) conference.

# 2 State of the art

In this chapter we provide a formulation of multiple object tracking problem and a review of the existing methods used to partially solve this problem. After this general overview about the MOT some among the most famous and recent algorithms for the worms tracking are described remarking the goals achieved and the limitation of each of these algorithms.

# 2.1 Video tracking

Video tracking can be defined as the process that estimates the location of one or more objects over time using a camera. The main challenges of a video tracker application can be summarized as follow in three categories [10]:

- Clutter: the appearance of the target is similar to the appearance of other objects in the scene or to the appearance of the background.
- Pose changing: the appearance of the target projected in a plan image changes when the target does 3D movements as rotations, translations and deformations.
- Ambient illumination: direction, colour and intensity of the ambient light influence the target appearance. These variations are particularly challenging for outdoor scenes.
- Noise: the image acquisition process can introduce noise which depends on the quality of the sensors used to acquire the image.
- Occlusions: it can happen that the target appears partially or totally occluded during video sequences. The target can appear occluded because it moves behind a static object as a wall, a column etc, or because other moving objects obscuring the view of the target. When the occlusions are partials, an appearance model based on some global features (e.g. the histogram) can be used to detect it. When total occlusions occur information about the appearance of the target is not sufficient for the tracking. In this case information about the typical motion behaviours can be used to estimate the position of the target. When the target reappears after an occlusion an appearance model can be used to correctly detect the target and re-initialise the track.

# 2.1.1 Problem formulation of tracking

As discussed in [10] the single object tracking problem can be defined as the estimation of a time series x over a set of discrete time instants:

$$x = \{x_k : x_k \in E_s, x_k \in E_s\}\tag{1}$$

where  $E_s$  is the state space and x is also known as trajectory of the target in  $E_s$ . The information encoded in  $x_k$  depends on the application. Let  $I_k$  be the frame (image plane) k of a video sequence. Usually  $I_k$  is mapped in a vector  $o_k \in E_o$  that summarizes the information of the target relevant for the tracking, where  $E_o$  is the observation space. This mapping process is called feature extraction or detection.

Usually, in the tracking problems, the state of the target is defined as a

position measure of the target in the plan image. In this case  $x_k$  can be defined as the coordinates of a particular point of the target:

$$x_k = (u_k, v_k) \tag{2}$$

where  $u_k$  is the vertical coordinate and  $v_k$  is the horizontal coordinate. Another option is to bound the target with a box defining the state as:

$$x_k = (u_k, v_k, h_k, w_k) \tag{3}$$

where  $u_k$  and  $v_k$  are the coordinates of the up-left corner of the box,  $h_k$  is the height of the box and  $w_k$  is the width of the box. The target could be also bound by an ellipse; in this case one more parameter is necessary,  $\theta_k$ , that defines the rotation of the ellipses:

$$x_k = (u_k, v_k, h_k, w_k, \theta_k) \tag{4}$$

More complex representation of the position of the target can be used, for instance a chain of points on the contour of the target.

The formulation of the single object tracking problem can be extended to formulate the multiple objects tracking problem. Let  $n_k$  and  $m_k$  be the number of target objects and the number of observations in the frame k. The state  $x_k$  can be extended as a set  $X_k$  that defines the multiple-target state:

$$X_k = x_{k,1}, x_{k,2}, \dots, x_{k,n} \in F(E_s)$$
(5)

where  $F(E_s)$  is the collection of all the finite subsets of  $E_s$ . The definition of observation given for a single target  $o_k$  can be extended to multiple targets:

$$O_k = o_{k,1}, o_{k,2}, \dots, o_{k,n} \in F(E_o)$$
(6)

where  $F(E_o)$  is the collection of all the finite subsets of  $E_o$ .

#### 2.1.2 Feature extraction

The performance of the tracking strictly depends on the quality of the features extracted from the image [11].

# Colour

The colour is the most common low level feature. In RGB model red, green and blue can be used. In this model a colour is represented as a point in a three-dimensional space where the three axes correspond to the red, green and blue primaries. The colour black is defined as (0, 0, 0) and the white is defined as (1, 1, 1). The HSV (hue, saturation, value) model also represent a colour as a point in a three-dimensional space but in this case the coordinate system is cylindrical. This representation can be useful because the values of H and S do not change with surface orientation, illumination direction and illumination intensity. The colour histogram can be used as object feature. It is based on the colour and it has the advantage to be invariant to rotation and scaling (if normalized).

### Edge

The local intensity changes are due to different reflectance properties of the object with respect to the surrounding background. This spatial intensity changes can be quantified and used to perform the edge detection as showed in Figure 2.



Figure 2: Edge detection

Given an plan image of the video sequence  $I_k$  it can be interpreted as a two dimensional function  $I_k(u, v)$ . The intensity changes can be obtained performing the partial derivative:

$$\nabla I_k(u,v) = \left[\frac{\partial I_k(u,v)}{\partial u}\frac{\partial I_k(u,v)}{\partial v}\right]$$
(7)

An easy way to detect edge is doing a threshold on the image-gradient magnitude. In the literature there are some well-known methods that can be used to perform the edge detection based on the approximation of the gradient components as: difference operations, Sobel filter, Prewitt filter, Canny, etc. Other methods are based on the second-order derivatives as: Laplacian, Difference of Gaussian (DoG), Laplacian of Gaussian (LoG), etc. *Motion* 

Another common low level feature is the motion. It is important to remark the difference between the 2D motion and the apparent motion. The 2D motion is the projection of the 3D real-world motion into the image plane. Apparent motion is what we consider to be motion analysing the intensity changes. In contrast to the previous features to define the motion is necessary to analyse more than one frame. The result of the subtraction is zero if there is no motion. Considering two successive frames, the simplest way to detect motions is subtracting the intensity values of the second image from the intensity values of the first one.



Figure 3: Optical flow example

More sophisticate techniques can be used to obtain a motion measurement as the optical flow [12]. An example of motion detection is given in Figure 3.

#### Interest points

Interest points are points of the image that can be automatically identified across a video sequence when pose and illumination conduction may change. Most of the interest-point detectors [10] select highly distinctive local patterns such corners, prominent edges, etc. Maravec detector defines the interest points searching the small square regions with the greater intensity variation in eight directions. The problem with this detector is that it is not rotation and scale invariant. The rotation invariation can be improved using the Harris corner detector but the scale invariant problem still remains. The easy way to solve the scale invariant problem is to extract features at a variety of scales, and then matching features at the same level. A well-known solution for the scale invariant is the Lowes Scale-Invariant Feature Transform (SIFT) [13].

#### Uniform regions

A uniform region is defined as a set of connected pixels sharing some properties as colour or motion. To obtain these uniform regions usually a cluster of selected low level features is done. The clustering algorithm can be divided in two categories: boundary based and region based. The boundary based post elaborate gradient information to locate boundaries that define partitions. Some of the most common algorithms in this category are: snakes, scissors and level sets. Region based algorithms estimate the similarity among features. Some of the most common algorithms in this category are: split and merge, region growing, amplitude thresholding and watershed techniques. 2.1.3 Background models When the camera is stationary the most widely used technique to detect the foreground pixels is the background subtraction. Using this technique is possible to detect the foreground pixels (moving pixels) in the current frame computing a subtraction between the background model and the current frame. If during the video sequence the background does not change it can be model with just an image otherwise more complex models have to be used. All of the methods that aim at the dynamic modelling of the background compute two main steps: an initialization phase done one time before the start of the tracking and an update phase done during the video analysis. The following two methods do the update of the background model using statistics of the pixel values over time (as mean and standard deviation): Gaussian Mixture Model [14] and Kernel Density Estimation. Using a dynamic model for the background it is possible to face global illumination changes. A high update rate can properly manage fast global illumination changes but has also the disadvantage that objects that stay stationary on the scene, even if for a little time, became part of the background model and are not detected as foreground. The quality of the results obtained with this technique depends on the rate of the luminosity changes. For example if in an outdoor application the clouds obscure the sun for a short time, the background subtraction technique could fail. In some application it could be easier to model the appearance of a specific class of objects than to model the appearance of the background.

## 2.1.3 Target models

A model of an object target includes information about appearance and shape of the target. The target model can be computed in different ways: defined using a priori information, defined using a snapshot information and defined using a set of training set samples.

#### Shape

The simplest model of the target shape is a point that usually corresponds to the object centroid. Other shape approximation model can be obtained creating a bound of the objects using a box or an ellipse. More sophisticate models try to approximate the 3D volume occupied by the target. This approximation can be done using a priori information about the 3D shape of the target or using multiple calibrated cameras with overlapping views. In this case the object can be bound with cuboid or ellipsoid. The obtained approximation of the shape can be used to manage in a more robust way multiple objects and occlusions. The previously discussed models can be combined to create an articulated model of the target shape. The kinematic joint constraints of the target articulation are modelled as constraints among the connected rigid models that constitute the target model. Using an articulated model it is possible to execute a complete body tracking. For a deformable object the articulated model cannot be robustly used to estimate the object shape because it is based on the canonical joints modelling. There are some models in literature that better model deformable objects as: fluid models, contours model and point distribution models. Using a fluid model the object target is divided in homogeneous regions that are tracked separately. There is no motion constraint among the identified regions. Using a contours model the tracking of the target is done tracking some points along the object contours of the target. To improve the tracking results some constrains on the contours of the object can be imposed. The point distribution model is the most accurate shape representation; it is able to represent deformations inside the object boundary.

## Appearance

In general the model used to represent the appearance can be used just for a

specific object in a specific application and cannot be generalized for class of objects. The most common target appearance representation is the template that codes the position and the colour (or the grey scale values) of all the pixels of the target area. To compare the template with an image region some index measurement can be used as the L1-norm, L2-norm or the normalized cross correlation. Before the comparison process between the selected image region and the template, some transformations on the image region could be done as translation and scaling. Instead of code information for all the pixels among the target area the histogram model codes the appearance of the target in a compact way extracting from the target a transformationinvariant description. The colour histogram encodes statistical distribution of the pixel values. This makes this representation invariant to scaling and rotation and also robust to partial occlusion. A tracking application that is based uniquely on this model could fail in case of luminosity changes, background clutter and 3D movements of the target object. To manage this situation a histogram model based on gradient [15] or/and orientation information is used together with the colour histogram model. As discussed for the background representation also for the target representation it can be necessary to define dynamic model instead of static model. A generic solution could be representing every parameter of the appearance model as a mixture of Gaussians that is update over time. If a set of training samples of the target is available the appearance changing can be extracted. This appearance changing model can be used to predict changes in the appearance of the target.

#### 2.1.4 Kalman filter

This filter was formulated by Rudolph E. Kalman in 1960 [16]. Kalman filter is used to estimate the state of a linear dynamic system. The estimation obtained with this filter is optimal in the sense that it minimizes the estimated error covariance. The inputs of the Kalman filter are measurements linearly related to the state system, corrupted by white noise. Kalman filter assumes that a discrete time linear dynamic system is modelled with the following equations:

$$x_{k+1} = A_k * x_k + B_k * u_k + w_k \tag{8}$$

$$z_k = H_k * x_k + v_k \tag{9}$$

## Where:

- $x_k$  is an array that defines the state of the system at time k. Generally the state defines the position and the velocity or the position, the velocity and the acceleration.
- $u_k$  is a vector that contains all the control inputs received at time k.
- $w_k$  is an array that defines for every state parameter a noise term. This noise is defined as zero mean white process noise with covariance:  $E[w_k * w_k'].$
- $z_k$  is an array containing the measures at time k.
- $v_k$  is an array containing the measurement noise terms for each observation in the measurement vector. This noise is defined as zero mean white measurement noise with covariance:  $E[v_k * v_k]$ .
- $A_k$  is the state matrix that defines the effect of the state parameters at time k on the system state at time k + 1.
- $B_k$  is the input matrix that defines the effect of the inputs at time t on the system state at time k.
- $H_k$  is a transformation matrix that maps the state vector parameters into the measurement domain.

After an initialization there are two main operations done in succession by the Kalman filer: a prediction update and a measurement update. The measurement update is done to correct the state and covariance estimations with the measurement. This operation can be interpreted as a correction of the state equations based on the measurement obtained. Using this correction method, when the system run steadily, the influence of the initialization is almost irrelevant. Predictions of the movement of the tracked objects can be used to improve the identification of the target objects among the video sequence: the predictions can be used as a motion feature in the target model. Also the detection process can be improved: using the predictions the algorithm can analyse just the portion of the image in which the objects should appear without scanning the entire image to detect the target objects.

# 2.2 Worms tracker

In the last years several methods have been proposed for tracking moving objects such as people [17], pedestrians on the street [18, 19], vehicles [20, 21, 22, and sport players in the court [23]. In the state of the art it is possible to find some algorithms that aim at tracking of flock of animals as: birds [24], bats [25], ants [26, 27], fishes [28, 29] and cells [30, 31]. Only a modest attention has been devoted to the tracking of worms. Indeed, it is worth to point out that traditional tracking methods cannot be simply applied for solving the tracking of swimming worms: the worms change their direction and their speed in a very fast way, thus their movement is not easily predictable. Furthermore, the appearance of the worms change very rapidly, thus the history of the moving objects cannot be simply used for increasing the reliability of the tracking algorithm, as in [17]. As summarized in this book [32] in the state of the art there are some algorithms for the tracking and the features extraction of worms. These algorithms can be organized in three categories: algorithms that aim at the tracking of one worm, algorithms that aim at the analysis of a group of worms and algorithms for the worms tracking.

# 2.2.1 Single worm trackers

These algorithms aim at the tracking and the features extraction of a single worm. The main goal is to analyse many videos in which just one worm appears and automatically define some features and behaviours of the worms that can be used to classify them.

## Worm Tracker 2.0

This algorithm has been implemented by the Schafer lab (Cambridge University) and has been proposed for the first time to the worm community in 2007. The current version can be downloaded on this website 2. The installation of Worm Tracker 2.0 is free but require a MATLAB Compiler Runtime Environment and Dino-Lite type microscope/webcam. The current release supports analysis of the worm's area, length, width, thickness, transparency and the brightness of head and tail.

# CoLBeRT (control locomotion and behaviour in real time) CoLBeRT is an algorithm implemented by Samuel lab in 2011. The design

set up for this system include a digital micro-mirror device (DMD, from Texas Instruments) to reflect a diode-pumped solid-state laser in order to achieve selective illumination. This system is the fastest real-time single worm tracker to date [32], capable of spatially restricted optogenetic manipulations [33].

### Nemo (Nematode movement)

Nemo is a software tool developed by the Nektarios Tavernarakis group in 2007 [34]. The goal of this software is to elaborate a video sequence with one worm and record for every frame of the video some features of the worm as the position of the centroid, the perimeter, the area and the pruned skeleton. Some key points on the border of the arena are tracked as well to understand when the arena is moved. Other similar algorithm for the tracking and the features extraction of one worm can be found here [6, 35, 36, 37].

# 2.2.2 Algorithms that analyse a group of worms

The algorithms that aim at the tracking of a group of worms use the background subtraction technique and some morphological operations to obtain a binary image in which the worms are described as connected regions in the foreground. The centroid of all the connected regions present in the binary image is tracked. When two or more regions are occluded the information about these regions is lost. The reason is that in the context in which these algorithms are used the focus is on the behaviour of the worms group and not on the exact trajectory of a every single worm presents in the image.

## The Parallel worm tracker

The Parallel worm tracker (PWT) has been implemented by the Miriam Goodman lab [38] to analyse the speed of the centroid points of 50 worms in parallel. The trajectory of a worm is stored only if it remains not occluded for more than a fixed time. This analysis is useful to enabling quantification of drug-induced paralysis: the speed of the worms is used as threshold to count the number of paralyzed worms. The last version of the algorithm is able to detect pirouettes in the trajectory of the worms as described in [39].

# The Parallel worm tracker

This algorithm has been implemented by the Rex Kerr and Catharine Rankin labs [40]. The goal of the algorithm is to analyse spontaneous movement on food, chemotaxis and habituation of response to tap stimulation. The core of the hardware system is a high-end digital camera (Falcon 4M30, 4 Megapixel (2352 1728), 31 fps) that renders the use of a microscope system unnecessary. With this hardware set up the algorithm is able to track up to 120 worms.

The optomechanical system for imaging or manipulation of neuronal activity in freely moving animals This system has been implemented by Lockery lab in 2011 [41]. The system is based on a non-invasive neuronal optogenetic manipulations [33]. The main goal is the visualization of neuronal activity in unterhered combined with tracking and quantification of locomotory behaviour.

The work [42] defines a methodology for morphometric and locomotory analysis of freely behaving worm populations. Worm movements are expressed in terms of three primitive motion patterns: peristaltic progression, deformation, and translation. These patterns define an overall locomotory space of C. elegans and to describe its general dynamic movements. The model has been incorporated into algorithms for segmentation and simultaneous tracking of multiple worms in a field, some of which may be interacting in complex ways. A recursive Bayesian filter is used for tracking. Unpredictable behaviours associated with interactions are resolved by multiple-hypothesis tracking. Unfortunately this method works just with the C. elegans worms. Other similar algorithms for the tracking and the analysis of a group of worms can be found here [43, 44].

## 2.2.3 Worms tracking

The main goal of these algorithms is to define the trajectory of two or more worms in a video sequence managing the occlusions. The PhD thesis of Kuang Man Huang [45] proposes a methodology for the tracking of C. elegans worms where it is possible to take trace of the worms also if they are occluded. Some motion patterns are defined to model the worm movements. The worms are modelled with rectangle blocks arranged on the deformable shape of the worms. An algorithm based on the dynamic programming is used to arrange the rectangle blocks inside the worm shapes. In case of occlusion one thousand body poses and some morphological operations are used to obtain the original shapes of the overlapped worms and keep track of them after the occlusion. In contrast to Huang's approach, our method does not try to segment the worms during occlusion, but aims at re-identifying them afterwards. We assume that during the occlusion there is no information that can be obtained from the frame. If two worms are occluded, instead to try to define which part of the connected region belongs to the first worm and which part belongs to the second worms, the features of the worms after the occlusion are analysed to figure out the correct label association. They also propose a method to define the head and the tail point of the worms. This method can be used just for C. elegans worms because the method is based on the fact that these particular worms have an accumulation of fat in the head so the worms head area is usually brighter than the tail area. The method that we propose for the definition of the head and tail points of the worms can be also used for different type of worms because the decision is taken studying the trajectory of the end-points of the worm skeleton. The method proposed by Hoshi [46] is able to detect the worm's head and tail. The method is based on the assumption that the head swings more than the tail so the head of the worm can be always distinguished from the tail because of the different speed. This method cannot be used because it is applicable only for the C. elegans (or similar) worms.

# 2.3 Platynereis dumerilii worms

Platynereis worms reproduce themselves by external fertilization. Worm eggs and sperm are secreted from the mature male and female worms and fertilize in the water. Therefore, for a successful reproduction of two Platynereis worms, male and female worms need to effectively locate one another within the large and complex marine environment. Coordinating the timing of spawning to the lunar cycle allows the worms to become sexually mature on the same day. Secondly, the worms release chemical pheromones to locate one another in the water as they swim around to look for mates. Finally, the series of spawning behaviours performed by male and female worms, commonly called the nuptial dance, are important for mixing the sperm and eggs together to facilitate successful fertilization. A general review can be found in [47]. The spawning process consists of four general phases: pre-spawning, engaged spawning, gamete release and post-spawning. 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 behaviour 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 behaviour. 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. Male and female worms can be distinguished by their colour and anterior/posterior segment border, which can be seen in Figure 4.



Figure 4: Image of a female (top) and a male (bottom) worm (taken from [8]).

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 of the head trajectory also changes for both worms.

# 3 Proposed method

In this chapter we propose a method for the tracking and the analysis of swimming worms. The first part of the chapter explains the main issue with the worms tracking. The second part of the chapter describes the proposed method. In the third part of the chapter the results of the evaluation done are showed.

# 3.1 Problem definition

The main problems with the tracking of two worms and the recognition of the male and female worm are:

- worms are deformable object;
- worms can touch each other or overlap themselves;;
- worms shape changes depending on the way in which the worms swim;
- worms luminosity changes depending on the area of the arena in which they swim;

There is also a space and a computational complexity constraint. The videos may have duration of twenty-four hours and the data about the worms have to be stored in an appropriate data structure. The elaboration can be done offline but the time request for the elaboration of the video should not exceed the duration of the video plus some hours. The videos of the worms are in grey scale values with a size of 1280x960 pixels and a variable frame rate: from 30 to 60 frames per second (43 frames per second in average). Before describing how the tracking and the features extraction are done two preliminary elaborations are described: the initialization phase and the preelaborate the video. The operations that belong to the pre-elaboration phase are done for every frame of the video. The goals of the initialization are: a background mask image (needed in the pre-elaboration phase) and the definition of the coordinates of the head points of the worms in the first frame (needed for the tracking). After this two phases a binary image is obtained in which the worms appear as two connected component in foreground. The tracking and the features extraction is done on the binary images.

# 3.2 Initialization phase

The initialization is needed to obtain a background mask image and to know the position of the head of the female worm and of the head of the male worm in the first frame of the video. The initialization phase follows these steps:

- 1. The position of the head of the female worm is manually marked;
- 2. The position of the head of the male worm is manually marked;
- 3. The part of the image that represents the arena is manually marked as ROI (region of interest). To define the ROI, a point that belongs to the border of the arena has to be marked. One point is enough because the arena has a circular shape and the centre of the arena is the centre of the image. When a point that belongs to the border of the arena has been marked is easy to define a circle that represents the border of the arena. The circle is defined as the set of pixels that have the same distance from the centre as the marked point. All the pixels of the image inside this circle represent the ROI.

Figure 5 shows an example of system initialization in which can be observed that the user has marked three points on the first frame of the video sequence.



Figure 5: Initialization phase

To build the background mask image from the first frame the following

step are done:

- 1. Threshold based on the luminosity of the pixels;
- 2. Every connected component that has an area bigger than a threshold (minimal worm area) is set as part of the background;
- 3. All the pixels out of the ROI are set as part of the background.

Figure 6 shows how these three steps are done.













Figure 6: Background mask building process

The white pixels represent the background. The background mask obtained in this phase is used in the background subtraction process to obtain a binary image where the worms are in foreground. The position of the head points is needed because the algorithm cannot automatically recognize the female and the male worm without any a priori information. In the feature work a method to understand the behaviour of the worms is described. With this method it would be possible to automatically recognize the male worm and the female worm analysing the trajectory of the worms without any a priori information about them. The tracking of the worms start after this initialization phase.

# 3.3 Worm detection

The goal of the worm detection phase is to obtain a binary image with two connected regions representing the two worms, or one connected region if the two worms are crossing each other. To reach this goal the following operations are done:

- 1. Threshold based on the luminosity of the pixels;
- 2. Background subtraction;
- 3. Removing connected components with an area smaller than a fixed threshold.

Figure 7 shows how these three steps are done. In the background mask b) the white pixels represent the background instead in c-d) the white pixels represent the foreground.



Figure 7: a) Original frame. b) Background mask image. c-d) Worm detection.

As background model is sufficient to use a static background mask image because the background does not change during the video. Assuming that the worms are not overlapped, for each connected region the elaboration proceeds according to the following steps:

# 1. Creating the skeleton of the connected region

To create the original skeleton the algorithm executes more thinning operations in sequence until the connected region has a thickness of one pixel. In the work of Pucher [8] has been observed that using the medial axis transformation process (MAT) [48] instead of the thinning process produces a skeleton with more branches, as showed in Figure 8.

## 2. Finding the end-points of the skeleton

When the skeleton has no brunches is easy to find its two end-points. When there are more than two end-points (because of the branches of the skeleton) just the two end-points that are connected with the longest path have to be chosen. The algorithm defines a distance matrix in which is defined the geodesic distance from each end-point to every other. The maximum element of the matrix defines the couple of end-points with the maximum geodesic distance. If there are more than two endpoints with maximum geodesic distance, the longest overall branch is not unique.



Figure 8: Worm skeleton computed using morphological thinning operation (left) and using (MAT) technique (right) (taken from [8]).

This can happen, if there are two branches with equal geodesic distance on one of the worms ends, as showed in Figure 8 (right illustration). In such cases, one of the end-points is chosen arbitrarily.

### 3. Pruning the skeleton from the branches

The branches are removed from the skeleton until the skeleton is reduced to one path that connects the couple of the end-points found in the previous step. The Figure 9 shows how these three steps are done to pruning the worm skeletons.



Figure 9: Pruning skeletons process

More information about the pre-elaboration phase can be found in the work of Pucher [8].

# 3.4 Feature extraction

To track the worms when they swim separated the only feature taken in account is the position. The correct association is done comparing the position of the worms with a prediction obtained with the Kalman filter [49]. To track the worms when they are occluded the information about the position cannot be robustly used. To manage the occlusion the proposed tracking method considers the following worm features: normalized shape, area, length, luminosity and trajectory. In this paragraph every feature is described. To better understand the variability and the values' range of these features, for every feature a plot is showed. All the plots have been generated from the same video sequence that has duration of three minutes and fifteen seconds and a variable frame rate, 43 frames per second in average.

# Area

The area is computed as the number of the pixels that define the connected region of the worm in the binary image.



Figure 10: Worms area

From the Figure 10 it is possible to observe how high the variability of this feature is over time. In just three frames the worms can change their way of swimming by showing their front or their side. These changing have a high influence on the area values. Also 3D movements influence the area value. The area is a very important feature because from the area plot the biologists can easily understand when the female worm releases the eggs. In the last part of the plot showed in Figure 10 the female collapse on itself until the end of the video so the algorithm holds the last available area value.

#### Length

The length is computed as an approximation of the geodesic distance between the two end-point of the skeleton. Starting from an end-point of the worm skeleton the geodesic distance is obtained summing the Euclidean distance between every point and the successive one. Using the geodesic distance as measurement of the length of the worms instead of that the number of pixels that compose to the worm skeleton a better approximation of the real length of the worm is obtained. To obtain a better approximation the radii of the circles at the first and last skeleton point are added to the length. It is important to include the circle radii as the skeleton endpoints do not always touch the outline of the binary worm region. More details about the computation of the geodesic distance can be found in Soille work [50].



Figure 11: Worms length

As with the area plot also from the length plot is possible to understand the period in which the female release the eggs. The length value is more stable than the area because when the worms can change their way of swimming the length value remains almost the same. The 3D movements have a high influence on the length values. Figure 11 shows an example of the worms length over a video sequence.

#### Luminosity

The luminosity is computed as the mean grey scale value of the pixels that belong to the worm skeleton. It is better to consider the luminosity of the skeleton instead of the luminosity of the entire region that represents the worm because there is some noise on the border of the region that alternates the real value of the luminosity of the worm. The noise is presents because of the fins of the worm that sometime are excluded from the connected region and sometime are included.



Figure 12: Worms appearance

Figure 12 shows an example of the worms appearance over a video sequence.





Figure 13: Touches occurrences

Figure 13 show the touch occurrences plot: every time that the worms touch each other the function is incremented by one. This feature is the most reliable to understand when the worms are mating. From the Figure 13 it can be observed that the mating time start after the frame 5.000 and ends before the frame 6.500.

# Velocity

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



Figure 14: Male speed



Figure 15: Female speed

From Figure 14 and Figure 15 it can be observed that the female worm has a more stable trajectory than the male worm. The female has almost regular speed instead the male has a high discontinues speed since he swims and stops over and over again.

# Head trajectory

The trajectory of a worm is defined as chronological list of head point positions. For every frame in the video sequence there is an entry in the list that defines the position of the head point of the worm, from the first frame to the last one Figure 16 shows a section of a trajectory.



Figure 16: Part of a video frame with an overlay of the head trajectory (taken from [8])

From Figure 18 and Figure 17 it can be observed that usually the female worm has a more regular trajectory than the male worm.



Figure 17: Female trajectory



Figure 18: Male trajectory

In fact the female worm move in circle during all the video instead the male worm has different behaviour. At the beginning of the video the male worm swims along the border of the arena because he does not detect the presence of the female worm in the arena so he tries to escape from the arena searching for a female worm. After this phase the male worm detect the female worm and start to swim near it with irregular trajectory. When the male start to release sperm he also moves in circle as the female. After the mating both the worms remain quiet.

## Normalized shape

From the binary image the medial axis transformation is computed for both the connected regions that represent the two worms. Figure 19 gives a visualization of this process.



Figure 19: Part of the distance transform of a worm with circles drawn for four points on the skeleton (taken from [8]).



Figure 20: Normalized shape (taken from [8]).

After this transformation, for every point of the skeleton worm a radius value is defined. In Figure 20 is showed how it is possible to obtain the normalized shape of the worm drawing a circle for every point of the skeleton in according to the correspondents radius value. Starting from an end-point of the worm skeleton that can be considered the first point of the skeleton, the geodesic distance between every skeleton point and the first skeleton point is computed. For every point of the skeleton it has been defined: the radius value and the geodesic distance respect the first point of the skeleton. A discrete function is built considering on the abscissa axis the geodesic distance of every skeleton point to the first point and on the ordinate axis the corresponding radius value. Interpolating the radius values a new continuous function is defined. To compare two normalized shape the radius values have to be compared in according to the same abscissa axis, for this reason the described function is sampled with a unitary sampling step.


Figure 21: Interpolation and sampling process

The result of this elaboration is the normalized shape feature used for the tracking of the worms. An example of the described process is showed in Figure 21.

### Situation that increase features' values changing

In this section some situations that cause high variation in the features' values are described.



Figure 22: A worm changing the swimming mode

Figure 22 shows an example of situation in which the worm changes the swimming mode: from a strict line trajectory to a circular one. It can be observed than when the worm swims in a strict line show the front and the fins are clearly visible. When the worm swims in a circular way the worm keep the fins near the body so they are not visible. Furthermore when the worm swim in a circular way just the side is showed. For this changing in the swimming mode the values of the area feature changes very fast. The value of the area of the worm in the Figure 22 are respective 451, 445, 401 and 340. There is a difference of more than one hundreds pixels in the area values in just six successive frames. Figure 23 shows an example of situation in which the worm changes the occupied part of the area. It can be observed

than when the worm occupies the parts of the arena near the border its luminosity is highest than when it swims in the canter of the arena. In the example shows in Figure 23 the average luminosity values of the pixels that belong to the worm skeleton are respective: 45, 48, 55 and 65. There is a difference of ten in the luminosity values in just twelve successive frames.



Figure 23: Different contrast in different parts of the arena

The eggs release is a situation that cause variation in the features values of the female worm. This process is slower respect the previous two situations. When the female releases the eggs the area and the length decrease while the luminosity increases. After the eggs release process the female is smaller than before and this is why the area and the length decrease during this process.



Figure 24: Eggs release process

After the eggs release process the female is brighter than before because when the female has released the eggs the tail and the centre part of the body became empty and transparent. Figure 24 shows an example of this process. In this example the features values of the female worm area respective:

- area: 316, 207 and 168 (pixels)
- length: 73, 61 and 32 (pixels)
- mean grey scale value of the skeleton: 61, 63 and 68.

This variation of the features values happen in 1393 frames (almost 30 seconds). It can be observed that the appearance of the female worm before the eggs release is completely different than after the eggs release. 3D movements are a cause of high variation in the appearance of the worms. This situations are more unusual than the previous one in fact the worms prefer to swim on the surface of the water and do 3D movement just when they are obligated. Figure 25 shows an example of 3D movement.



Figure 25: An example of 3D movement

# 3.5 Tracking of the worms

To describe what happens in the arena, different situations are defined. Each situation represents a different state of the arena. The following image shows how the state of the arena can evolve:



Figure 26: States of the arena

For each situation, the elaboration is different because there are different types of problems to solve:

- When the worms swim separated the correct association among the end-points of the worm skeleton in the frame at time t and the points that represent the tail and the head of the two worms in the frame at time t-1 have to be found. In this case the tracking of the worm is done using movement predictions of the worms obtained with the Kalman filter.
- When the worms touch each other or when they are overlapped there is not much information about the behaviour of the worms that can be got. In this case the problem is to define what should be the values of

the worms feature and try to predict the positions of the head points of the worms.

• When the worms reappear separated after an occlusion the problem is to label the worms as male worm and female worm. There is also the problem to label the two end-points of the worm skeletons as head point and tail point. To solve these problem two methods are proposed: gender re-identification and head-tail definition.

### **3.5.1** Worm representation

For the tracking of the worms some points along the worm skeletons are defined as key points. These key points are tracked during the frame sequence. In this paragraph the notation used to reference these points is described.

Let  $w(\tau,k,id)$  be a function that defines the position of the points of the worms skeleton where is a real number defined in [0, 1], k is an integer number that represents the frame number of the video and id is an identification label. In case of two worms tracking:  $id \in [male, female]$ . For a fixed frame  $\hat{k}$  of the video and for a fixed worm  $\hat{id}$  the position of the tail point of the worm is  $w(0,\hat{\kappa}+1, \hat{id})$  and the position of the head point of the worm is  $w(1,\hat{\kappa}+1, \hat{id})$ . The worm skeleton can be divided into three parts: tail points  $\tau \in [0, \frac{1}{3}]$ , body points  $\tau \in [\frac{1}{3}, \frac{2}{3}]$ , and head points  $\tau \in [\frac{2}{3}, 1]$ .

The trajectory of the head and tail of a worm is not a continuous wave and a lot of vibrations and oscillations are present. For this reason it is not possible to correctly predict the position of these points using the Kalman filter. The points that belong to the body represent the part of the worm skeleton in which less vibrations and waves are observed. The tracking of the worms is done using a movement prediction, obtained with the Kalman filter, of the following skeleton points:  $w(\frac{1}{3},k,id)$  and  $w(\frac{2}{3},k,id)$ .

### 3.5.2 Track the worms when they swim separated

When the worms swim separated there are two connected regions in the binary image that represents the two worms. In the frame  $\hat{k}$  the position of the following points are know:  $w(\frac{1}{3},\hat{k},male), w(\frac{2}{3},\hat{k},male), w(\frac{1}{3},\hat{k},female)$  and  $w(\frac{2}{3},\hat{k},female)$ .

The position of the same points in the frame k+1 is unknown. In the frame

 $\hat{k}+1$  for both the worms two points called  $p_1$  and  $p_1$  are defined in the following way:

- 1. One between the two skeleton end-points is chosen to be the first endpoint.
- 2.  $p_1$  is the point along the worm skeleton that has a distance of  $\frac{1}{3}$  from the first end-point.
- 3.  $p_2$  is the point along the worm skeleton that has a distance of  $\frac{2}{3}$  from the first end-point.

The tracking problem is formulated in the following way: find the correct association among the points defined in the frame k and the points defined in the frame  $\hat{k}+1$ . Figure 27 shows how these points are marked in the frame  $\hat{k}$  and in the frame  $\hat{k}+1$ .



Figure 27: Associations problem

The first step of the elaboration is to predict, for both the worms, the position of the points  $w(\frac{1}{3}, \hat{k}, id)$  and  $w(\frac{2}{3}, \hat{k}, id)$  using the Kalman filter. Let  $k(\tau, k, id)$  a function that defines the predicted position of the point  $w(\tau, k - 1, id)$  based on the trajectory of the same point [49].



Figure 28: Information that the association manager takes in account

The number of the all possible associations among the points in the frame  $\hat{k}+1$  and the Kalman prediction is eight as showed in Table 1.

Hypothesis	$k\left(\frac{2}{3},\widehat{k},male\right)$	$k\left(\frac{1}{3},\widehat{k},male\right)$	$k\left(\frac{2}{3}, \widehat{k}, female\right)$	$k\left(\frac{1}{3}, \widehat{k}, female\right)$
Hp1	$p1_{region1}$	p2 <sub>region1</sub>	$p1_{region2}$	$p2_{region2}$
Hp2	$p2_{region1}$	$p1_{region1}$	$p1_{region2}$	$p2_{region2}$
Hp3	$p1_{region1}$	$p2_{region1}$	$p2_{region2}$	$p1_{region2}$
Hp4	$p2_{region1}$	$p1_{region1}$	$p2_{region2}$	$p1_{region2}$
Hp5	$p1_{region2}$	$p2_{region2}$	$p1_{region1}$	$p2_{region1}$
Hp6	$p2_{region2}$	$p1_{region2}$	$p1_{region1}$	$p2_{region1}$
Hp7	$p1_{region2}$	$p2_{region2}$	$p2_{region1}$	$p1_{region1}$
Hp8	$p2_{region2}$	$p1_{region2}$	$p2_{region1}$	$p1_{region1}$

Table 1: Possible associations

For every prediction the error is calculated as the Euclidean distance between the position of the prediction and the corresponding point in frame

	$k\left(\frac{2}{3},\widehat{k},male\right)$	$k\left(\frac{1}{3},\widehat{k},male\right)$	$k\left(\frac{2}{3},\widehat{k},female\right)$	$k\left(\frac{1}{3}, \widehat{k}, female\right)$
$p1_{region1}$	(1,1)	(1,2)	(1,3)	(1,4)
$p2_{region1}$	(2,1)	(2,2)	(2,3)	(2,4)
$p1_{region2}$	(3,1)	(3,2)	(3,3)	(3,4)
$p2_{region2}$	(4,1)	(4,2)	(4,3)	(4,4)

 $\hat{k}+1$  as showed in Table 2.

For every prediction the corresponding error is calculated as the Euclidean distance between the position of the prediction and the position of the predicted point in frame  $\hat{k}+1$ . For every hypothesis four errors are taken in account: the errors of the head and tail predictions of the male worm and the errors of the head and tail predictions of the female worm. The hypothesis with the lowest root mean square error (RMSE) is chosen. The root mean square error has been preferred to the mean absolute error [51] because the prediction errors follow a norm distribution (also the error model used in the Kalman filter is defined in according to a norm distribution). Furthermore it has been proved on the evaluation done that using the root mean square error instead of the mean absolute error brings better results.

Possible association	Root mean square error
Hp1	$\frac{1}{4} * \sqrt[2]{(1,1)^2 + (2,2)^2 + (3,3)^2 + (4,4)^2}$
Hp2	$\frac{1}{4} * \sqrt[2]{(1,2)^2 + (2,1)^2 + (3,3)^2 + (4,4)^2}$
Hp3	$\frac{1}{4} * \sqrt[2]{(1,1)^2 + (2,2)^2 + (3,4)^2 + (4,3)^2}$
Hp4	$\frac{1}{4} * \sqrt[2]{(1,2)^2 + (2,1)^2 + (3,4)^2 + (4,3)^2}$
Hp5	$\frac{1}{4} * \sqrt[2]{(1,3)^2 + (2,4)^2 + (3,1)^2 + (4,2)^2}$
Hp6	$\frac{1}{4} * \sqrt[2]{(1,4)^2 + (2,3)^2 + (3,1)^2 + (4,2)^2}$
Hp7	$\frac{1}{4} * \sqrt[2]{(1,3)^2 + (2,4)^2 + (3,2)^2 + (4,1)^2}$
Hp8	$\frac{1}{4} * \sqrt[2]{(1,4)^2 + (2,3)^2 + (3,2)^2 + (4,1)^2}$

Table 3: Potential error of the hypothesis

When the correct association of the points  $w(\frac{1}{3}, \hat{k}, id), w(\frac{2}{3}, \hat{k}, id)$  and  $w(\frac{1}{3}, \hat{k}+1, id),$  $w(\frac{2}{3},\hat{k}+1,id)$  have been found, it is easy to define the position of the points  $w(0,\hat{k}+1,id)$  and  $w(1,\hat{k}+1,id)$ . One of the two end-points of the worm skeleton is chosen as start point. From the start point every skeleton point is analysed one by one. If the first point met during this walk is  $w(\frac{1}{3}, k+1, id)$ it means that the start point represents the position of the tail of the worm and the other end-point represents the position of the head of the worm, instead if the first point met during this walk is  $w(\frac{2}{3},\hat{k}+1,id)$  means that the start point represents the position of the head of the worm and the other end-point represents the position of the tail of the worm. It is important to keep in mind that first the association among the points  $w(\frac{1}{3}, \hat{k}, id)$  and  $w(\frac{2}{3}, \hat{k}, id)$  is found and after the position of the head and of the tail of the worms in the frame  $\hat{k}+1$  is defined. There are some cases in which doing directly the tracking of the points w(0, k, id) and w(1, k, id), bring the algorithm to takes wrong decisions because the worms can turn the head in the opposite direction respect to the movement. This method is used to keep trace of the worms when they do not cross each other. This method can be optimized considered the fact that the worms cannot appear in the current frame too far from their positions in the previous frame so if the pairs of

points in the frame  $\hat{k}$  that represent the male worm are far away from the pairs of points in the frame  $\hat{k}$  that represent the female worm the possible hypothesis became four instead of eight (first four rows of Table 1).

Figure 29 show an example of tracking between two successive frame:  $\hat{k}$  and  $\hat{k}+1$ . Figure 29-a) shows the results of the tracking of the frame  $\hat{k}$ , where the position of the head and tail points are known for both the worms (pink and blue crosses). Figure 29-b) show the detection of the two worms in the frame  $\hat{k}+1$  where it is not known which points are the head and tail points of the male worm and which points are the head and tail points of the female worm (red crosses).



Figure 29: a) Results of the tracking of the frame  $\hat{k}$ . b) Analysis of the frame  $\hat{k}+1$ .

Figure 30 provides a visualization of all the possible associations among the Kalman predictions and the positions of the worms in the frame  $\hat{k}+1$ . The blue circles are the Kalman predictions of the head and tail of the male worm. The pink circles are the Kalman predictions of the head and tail of the female worm. The red crosses are the positions of the worms in the current frame. The yellow lines define the distances among the Kalman predictions and the correspondence worms positions.



Figure 30: Example of comparison among the Kalman predictions and the positions of the worms

In Figure 30 it can be observed that for every possible hypothesis the root mean square error is defined. In this example the correct hypothesis, the first one, is chosen because with this hypothesis the RMSE is minimized.

### 3.5.3 Track the worms when they cross each other

When the worms touch each other or when they are overlapped there is not much information about the behaviour of the worms that can be got from the binary image. The only thing that the algorithm does in this situation is to update the position of the tail and of the head points of the two worms using a prevision based on Kalman filter. To improve the accuracy of the Kalman filter prediction the algorithm relocates the position of the predictions inside the connected component that represent the two worms overlapped. In fact we know that the head and the tail of the two worms have to be inside the connected component.



Figure 31: Relocation of Kalman predictions.

The position of the Kalman prediction is used just for the online visualization during the elaboration. When the elaboration of the entire video the values of the worm features for the frames in which the worms are occluded are defined doing a linear interpolation among the values of these features in the last frame before the occlusion and the value of the same features in the first frame after the occlusion. In the feature a way to get the skeletons of the worms also when in the binary image there is just one connected region for both the worms could be defined. Two worms are considered occluded when they cross each other or when they touch each other. There are many other cases in which the worms have to be considered occluded:

- When one of these two points  $k(\frac{1}{3},\hat{k},male)$  and  $k(\frac{2}{3},\hat{k},male)$  is too near to one of these points  $k(\frac{1}{3},\hat{k},female)$ ,  $k(\frac{2}{3},\hat{k},female)$ . This situation happens when the frame rate is too low and the two worms swim near along two circular trajectories.
- When the tail point of the worm touches the head point of the same worm. This situation happens when the worm close itself in a circular shape, in this case the information about the end-points of the worm is lost. This situation has to be considered as an occlusion. The occlusion end when the two end-points of the worm reappears separated.
- When the Euclidean distance between these two points  $k(\frac{1}{3},\hat{k},id)$  and  $k(\frac{2}{3},\hat{k},id)$  is too short. This situation happens when the worm collapse on itself (dead worm).

### 3.5.4 Track the worms when they reappear separated

The important thing about the crossing of the worms is to be able to recognize the male worm and the female worms when they reappear separated as well as to identify the points that represent the tail and the head of the worms. The elaboration proceeds in two main steps:

- 1. Gender re-identification: in the frame at time t we have two different regions. The problem is to decide which region represents the male worm and which region represents the female worm;
- 2. Identify the position of the head and tail points: for each region the algorithm studies the trajectory of the two end-points of the region skeleton and decides which end-point is the tail and which end-point is the head.

To do these two steps the algorithm elaborates some frames before the occlusion and some frames after the occlusion as showed in Figure 32. During this time the algorithm analyse the features of the two worms before the occlusion and the features of the two connected regions after the occlusion. This information is used to define the gender of the two connected regions. During the observation time also the trajectory of the skeleton end-points of the two connected region are analysed. This analysis is used to define the head and the tail of the worms after the occlusion. If the worms cross each other during the "observation time" or there is not enough frames in the video to continue this analysis the algorithm makes a decision with the information got until that time. For the tail and head detection more the observation time is long and the more the probability to take a correct decision increases but also the elaboration time will increase. At the end of this analysis the complete trajectory of the head and of the tail points of the two worms is known. Since the algorithm have to produce a video in which the trajectory of the two worms are showed, though we know the trajectory of the two worms we have to restart the elaboration from the fist frame analysed during the observation time and generate a frames sequence in which the trajectory of the two worms is showed with appropriate markers. For this reason if we increase the observation time the elaboration will be increased because the frame that we analyse during the observation time have to be re-elaborated to draw the trajectory of the two worms. For the gender re-identification the observation time should be short because the value of the features worms

change very fast. Observing just few frames before the occlusion and few frames after the occlusion, the two worms do not have enough time to change the values of their features. For this reason observation windows of different size are used: one for the gender re-identification and one for the identification of the head and tail points.

### 3.5.5 Gender re-identification

An occlusion is defined as a set of frames in which the worms are overlapped and represented in the binary image by one connected component. In the frames before the occlusion the value of the worm features are known, during the occlusion there is no information about the worm that can be obtained from the binary image, after the occlusion there are two connected regions in the binary image. The problem is to define the gender of the two connected regions that represent the worms after the occlusion. The idea is to compare some features of the two worms before the occlusion with the features of the two regions. The features taken in account are: normalized shape, area, mean grey-scale value, length, position of the centroid. A description of the features can be found in the paragraph 3.4. Before the occlusion the features of the worms are used to build two models: one for the male worm and one for the female worm. To compare the worm models with the region features, a measure of similarity and a comparison methodology is defined. To increase the strength of the decision, five frames after the occlusion are taken in account. A decision about the correct gender re-identification is made for every one of these frames. After the comparison, the association that has been chosen most often during the analysis is taken as correct. Figure 32 shows how the frames are divided and analysed.



Figure 32: Gender re-identification process

The number of frames to analyse has to be chosen according to the rate in which the chosen features change over time. For Platynereis dumerilii worms, some of the features depend on how and where the worms are moving. When they swim in a straight line, the body area is bigger than when they are moving in circles. The luminosity also changes depending on where they swim. It is higher in the middle of the arena than at the border. Therefore a decision of five frames was chosen empirically for Platynereis dumerilii worms in the current tracking set-up. The algorithm uses the same approach showed in the fourth paragraph to take track of the two blobs. To build the male and female model, the average value of the features in the last five frames before an occlusion is calculated. The position feature is defined with a movement prediction obtained with the Kalman filter [52]. This feature defines where the worm should be after the occlusion and it is used in case the occlusion has a short duration. The other features define how the worms should appear after the occlusion. As information about the position the point w(0, 5)k,id is used because it can be easily obtained from the skeleton of the region without introducing errors in the gender re-identification process. For every frame in the observation window the features of the two regions are cooperated with the male and female models. After this comparison a similarity matrix is built as showed in the Table 4.

	$model_1(male)$	$model_2$ (female)
$region_1$	<i>s</i> <sub>11</sub>	S <sub>12</sub>
region <sub>2</sub>	s <sub>21</sub>	s <sub>22</sub>

 Table 4: Similarity matrix

The element  $s_{ij}$  of the matrix is computed in this way:

$$s_{ij} = \sqrt{\frac{\alpha_n * (s_{ij}^n)^2 + \alpha_a * (s_{ij}^a)^2 \alpha_m * (s_{ij}^m)^2 + \alpha_l * (s_{ij}^l)^2 + \alpha_p * (s_{ij}^p)^2}{\alpha_n + \alpha_a + \alpha_m + \alpha_l + \alpha_p}}$$
(10)

The measure of similarity that we propose to use is a quadratic weighted mean. To compare different measurements in just one index every measurement of similarity (normalized shape, area, mean grey scale value, length and position) is normalized to one, and also the weights  $\alpha_n$ ,  $\alpha_a$ ,  $\alpha_m$ ,  $\alpha_l$  and  $\alpha_p$ are defined between zero and one.

#### Normalized shape

 $s_{ij}^n$  is the factor that takes account of the normalized shape of the  $region_i$ and the normalized shape of the of the  $model_j$ . The paragraph 2.1 explains in detail how the normalized shape is computed. The normalized shape of a worm can be interpreted as a discrete signal r(k): on the abscissa axis there are some integer values from one to N and on the ordinate axis there are the radius values. N is defined as the integer value that better approximate the geodesic length of the worm skeleton. Assuming that the normalized shape of the  $region_i$  has the same length of the normalized shape of the  $model_j$ then the similarity index between these two features is computed using the Pearson correlation coefficient [53, 54]:

$$s_{ij}^{n} = \frac{1}{N-1} * \sum_{n=1}^{N} \left( \frac{r_i(k) - \mu_i}{\sigma_i} \right) * \left( \frac{r_j(k) - \mu_j}{\sigma_i} \right)$$
(11)

Where  $r_i(\mathbf{k})$  is the normalized shape of the  $region_i$ ,  $r_j(\mathbf{k})$  is the normalized shape of the  $model_j$ ,  $\mu_i$  and  $\sigma_i$  are the average and the variance of the function  $r_i(\mathbf{k})$  and  $\mu_j$  and  $\sigma_j$  are the average and the variance of the function  $r_j(\mathbf{k})$ . If the two normalized shapes,  $N_i$  and  $N_j$ , have different length the value N is defined as the minimum between  $N_i$  and  $N_j$ . The comparison is done several times taking in account the signal with the shortest length and cut versions of the longest signal. The cuts are done in according to a window of N elements shifted on the longest signal. For every possible shift a new cut is obtained and the Pearson formula is applied on the shorted signal and on the cut version of the longest signal. The maximal value obtained after the comparisons is chosen as similarity measurement. Using this shift system the 3D movements are management, Figure 33 shows an example: in the first image the worm shows the complete front side of the body so the normalized shape represents the entire front side of the worm body having a length of 66 pixels.



Figure 33: Example of normalized shape correlation in 3D movements

In the successive frames the worm does a 3D movement and the normalized shape represents just the front side of the worm head. The normalized shape that represents the head will has a shortest length than the normalized shape that represents the body. In according to the shift system the normalized shape that represents the head is compared several times along the worm body. When the normalized shape that represents the head is compared with the part of the worm body that represents the head (last part of the normalized shape) a high value of similarity measurement is got.

### Area

 $s_{ij}^a$  is the factor that takes account of the current area of the  $region_i$  and the area of the  $model_i$ .

$$a_{ij}^{a} = 1 - \frac{|d_{ij}^{a}|}{d_{max}^{a}}$$
 (12)

Where  $d_{ij}^a$  is the difference between the area of the  $region_i$  and the area of the  $model_j$ . To obtain the  $s_{ij}^a$  the value  $d_{ij}^a$  is normalized with the factor  $d_{max}^a$  that represents the maximal value of the area of a worm.

### Mean grey scale value

 $s_{ij}^m$  is the factor that takes account of the mean grey value of the skeleton of the  $region_i$  and the mean grey value of the skeleton of the  $model_j$ .

$$s_{ij}^m = 1 - \frac{|d_{ij}^m|}{255}$$
 (13)

Where  $d_{ij}^m$  is the difference between the mean grey value of the skeleton of the  $region_i$  and the mean grey value of the skeleton of the  $model_j$ . To obtain the  $s_{ij}^m$  the value  $d_{ij}^m$  is normalized with a factor that represents the maximal grey scale value of the skeleton of a worm.

### Geodesic length

 $s_{ij}^l$  is the factor that takes account of the length of the skeleton of the  $region_i$ and the length of the skeleton of the  $model_j$ .

$$s_{ij}^{l} = 1 - \frac{|d_{ij}^{l}|}{d_{max}^{l}}$$
 (14)

Where  $d_{ij}^l$  is the difference between the length of the skeleton of the  $region_i$  and the length of the skeleton of the  $model_j$ . To obtain the  $s_{ij}^l$  the value  $d_{ij}^l$  is normalized with the factor  $d_{max}^l$  that represents the maximal length of a worm skeleton.

### Position

 $s_{ij}^p$  is the factor that takes account of the position of the centroid of the  $region_i$  and the prediction of the position of the centroid of the  $model_j$ .

$$s_{ij}^p = 1 - \frac{|d_{ij}^p|}{d_{max}^p} \tag{15}$$

Where  $d_{ij}^p$  is the Euclidean distance between the position of the central point of the skeleton of the  $region_i$  and the prediction of the position of the central point of the skeleton of the  $model_j$ . To obtain the  $s_{ij}^p$  the value  $d_{ij}^p$  is normalized with the factor  $d_{max}^p$  that represents the maximum distance that a worm can cover in a time equal to the duration of the occlusion. There are two possible associations:  $region_1$  is the male worm and  $region_2$  is the female worm  $Hp_1$  or vice versa  $(Hp_2)$ . A measure of confidence is assigned to both the hypotheses:  $s_{11}+s_{22}$  to  $Hp_1$  and  $s_{12}+s_{21}$  to  $Hp_2$ . The association with the greater confidence value is chosen as result of the gender re-identification.

### 3.5.6 Trajectory analysis for head and tail definition

At this point of the elaboration is known which region is the male worm and which region is the female worm. The problem now is to define for both the worms which point between the two end-points of the skeleton is the head and which one is the tail. Two methods are proposed that can be used to reach this goal. The first method is based on the assumption that the worms are alive, the second one is based on the assumption that the worms are dead or losing vitality. The first method studies the trajectories of the skeleton end-points for some frames after the occlusion. This phase is called trajectory analysis. The second method compares the positions of the worms after the occlusion with the positions of the worms before the occlusion. In this case the trajectories of the skeleton end-points cannot be robustly used to identify the head and the tail of the worms because in this case the worms are dead and the movements depend on the water waves. The idea is based on the fact that when a worm swims in toward the tail point is pulled from the head point. For each worm the algorithm defines which point (among the two end-points) pulls and which point is pulled. For every frame during the trajectory analysis the tracking of the following points of the worm skeleton is done:  $w(\frac{1}{3},k,id)$ ,  $w(\frac{1}{2},k,id)$  and  $w(\frac{2}{3},k,id)$ . To track these points during the observation time the method based explained in the paragraph 3.2 is used. The Figure 34 shows how these points are represented for a fixed worm *id* in two successive frames  $\hat{k}$  and  $\hat{k}+1$ .

$$w\left(\frac{1}{3},\hat{k},\hat{id}\right) \\ w\left(\frac{1}{2},\hat{k},\hat{id}\right) \\ w\left(\frac{1}{2},\hat{k},\hat{id}\right) \\ w\left(\frac{1}{2},\hat{k},\hat{id}\right) \\ w\left(\frac{1}{3},\hat{k}+1,\hat{id}\right) \\ w\left(\frac{1}{$$

Figure 34: Information taken in account to identify the head and the tail points.

As shown in Figure 34 the distance between the point  $w(\frac{1}{3}, \hat{k}+1, \hat{id})$  and the point  $w(\frac{1}{2}, \hat{k}, \hat{id})$ , is smaller than the distance between the point  $w(\frac{2}{3}, \hat{k}+1, \hat{id})$ and the point  $w(\frac{1}{2}, \hat{k}, \hat{id})$  therefore the point  $w(0, \hat{k}+1, \hat{id})$  is voted as tail point. This voting is repeated for each one of the frames analysed during the observation time. At the end of the trajectory analysis the point with highest number of votes becomes the tail point. If during the trajectory analysis the worms do not move in backward for a time greater than half of the duration of the trajectory analysis the outliers are avoided. The duration of the trajectory analysis have to be chosen as a trade-off between the probability to take the correct decision and the elaboration time. Therefore a decision of thirty frames was chosen empirically for Platynereis dumerilii worms in the current tracking set-up.

It has been chosen to study the trajectory of the points  $w(\frac{1}{3},k,id)$  and  $w(\frac{2}{3},k,id)$ and not the trajectory of the points w(0,k,id) and w(1,k,id) for the following reason: image to define Cartesian reference system centred on  $w(\frac{1}{2},k,id)$  that moves according to the trajectory of this point. What can be observed is that the relative trajectory of the points  $w(\frac{1}{3},k,id)$  and  $w(\frac{2}{3},k,id)$  with respect to this reference system is almost zero instead the relative trajectory of the points w(0,k,id) and w(1,k,id) with respect to this reference system is not zero because a worm can always turns left and right the head and the tail.

Sometime one of the worms (or both of them) dies after the mating phase. In such a case the worm stops to swim and starts to move through the movement of the water. The trajectory analysis described in the previous paragraph cannot be used to manage this situation, because the assumption that the tail point is pulled from the head point is not valid. For the dead worms we propose to use the same approach defined in the paragraph 3.2 but with the simplification that we consider just one worm. To define the head and the tail points after an occlusion we compare the positions of the head and of the tail points of the worm in the last frame before the occlusion with the positions of the end-points of the worm skeleton in the first frame after the occlusion. There are four possible associations among these points and the association with the smallest mean square error is chosen as correct. A dead worm detector defines when a worm shows a dead behaviour. Assuming that the size of the trajectory analysis N, a worm is considered dead if the following conditions occur:

- Speed: the average speed of the point  $w(\frac{1}{2},k,id)$  in N frames after the occlusion is lower than a threshold.
- Movement: the distance between the point  $w(\frac{1}{2},k,id)$  in the last frame before the occlusion and the position of the same point in the first frame after the occlusion is lower than a threshold.

Usually when a worm is dead does strange and slows movements as rotations or waves with the body or just collapse on itself. As measure of the worm speed it has been chosen the speed of the point  $w(\frac{1}{2},k,id)$  and not the speed of the end-points skeleton because the speed of the points w(0,k,id)and w(1,k,id) could be highest than the threshold because of the rotations and the waves.

A threshold on the average speed of the worm is imposed for two reasons:

- 1. If the speed of the worm is low, is high probable that the worm is dead.
- 2. If the speed of the worm in some frames after the occlusion is high, it can be assume that with the trajectory analysis there is a good chance to take the correct decision.
- A threshold on the movement of the worm is imposed for two reasons:
- 1. The threshold on the speed is not sufficient to detect dead worm because there are some situations in which the speed is high but the worm is dead, these situations happen when the dead worm does rotation or waves.
- 2. If the position of the worm after the occlusion is near to the position of the same worm before the occlusion is high probable to take the correct decision about the identification of the head and tail points using the altered version of the association manager instead that with the method based on the trajectory analysis. For this reason it has been chosen to

consider the movement of the worm during the occlusion and not the movement of the worm after the occlusion.

Figure 35 shows an example of occlusion management.



Figure 35: Frames sequence in which the worms cross each other.

## 3.6 Computational complexity

In this paragraph we define the computational complexity of the proposed method. In this analysis we not consider the fact that some operations can be performed on the GPU and also we not consider particular architecture as Multiple Instruction Multiple Data or Single Instruction Multiple Data. To evaluate the computational complexity of the proposed method we consider the upper limit under the assumption of worst case. In the worst case the worms appear in every frames with the maximal possible length and the maximal possible area values. For this reason we consider the maximal value of every variable taken in account along the video sequence. In this way we can consider the value of every variable as an unknown value that is always the same for every frame of the video sequence. Furthermore in this analysis we do not consider eventually constant factors. The following table summarize the variables used in this evaluation.

Variable	Description		
а	Number of pixels that define the area of a worm		
S	Number of the pixels that belong to the skeleton of a worm		
e	Number of boundary edges in the connected component		
C	that defines the worm in the binary image.		
b	Number of branches in a worm skeleton		
k	Number of frames that compose the video sequence		
n	Height of the frames expressed in pixels		
m	Width of the frames expressed in pixels		



As described in the previous section the method is composed by three main functional blocks executed in succession: detection, features extraction and tracking. The computation complexity of the method can be defined as the max among the computation complexities of these three main functions. To simplify this evaluation we consider the operations related to the worms skeletons in the features extraction phase.

### Detection

The operations that have to be computed during the detection process are: threshold, background subtraction, closing and connected components labelling. Regarding the threshold operation, every pixel of the grey scale image has to be checked so the computation complexity of this operation is O(k \* n \* m). Also for the background subtraction an elementary operation has to be computed for every pixel of the image so the complexity of this operation is O(k \* n \* m). The morphological operation closing is computed as a dilatation operation followed by erosion operation. The operations of dilatation and erosion have the same computational complexity: a kernel of a fixed size (generally 3x3 or 5x5) is superimposed on every pixel of the binary image; if the pixels of the binary image that are under the kernel respect a fixed condition the pixel of the binary image that is under the origin of the kernel is set to one or to zero depending on the imposed condition. Dilatation and erosion have a computational complexity of O(k \* n \* m) so also the closing operation has a computational complexity of O(k \* n \* m). Regarding the connected components labelling all the pixels of the binary image have to be scanned two times. During the first scan every foreground pixel receive a region label, during the second scan the equivalent labels are substituted by a unique one. After the second scan every foreground pixel has a label and there are not equivalent label. In the end the connected components labelling process has a linear time with the number of pixels. We can conclude that the detection phase has a computational complexity of O(k \* n \* m).

### Features extraction

During the features extraction process, for both the connected component present in the binary image the following operations are done: measure the area of the connected component, compute the skeleton, pruning of the skeleton, compute the normalized shape, compute the geodesic length of the skeleton and compute the mean grey scale value of the pixels that belong to the skeleton. To compute the geodesic length of the worm we compute the Euclidean distance between every point of the skeleton and the following one. Summing these Euclidean distances the algorithm estimates the geodesic distance of the worm. Using the Newton's method based on the Schoolbook long multiplication method the computational complexity of this operation is  $O(k * s^2)$  where s is the maximal number of pixels of the worm skeleton and k is the number of frames. To calculate the area of the worms the algorithm has to count the number of pixels of the connected components, in the binary image, that define the two worms. This operation is linear with the maximal number of pixels that define a worm in the binary image: O(k\*a). The skeleton is obtained performing the medial axis transformation. According to the work of Lee [56] the computational complexity of the MAT in case of planar shape is  $O(k * e * \log e)$  where e is the maximal number of boundary edges in the connected component that define the worm in the binary image. The normalized shape can be easily obtained in a constant time from the MAT taking in account the radii values of the pixels that belong to the pruned version of the skeleton. The time needed for the pruning of the skeleton is quadratic with the number of the branches of the skeleton. For every branch there is an end point in the skeleton. For every end point, the geodesic distance with all the other end points of the skeleton has to be computed; this is why the pruning has a quadratic time with the number of the branches. Pruning the skeleton is an operation that has a computation complexity of  $O(k * b * s^2)$ .

We can conclude that the computation complexity of the features extraction phase is:  $O(k * b * s^2) + O(k * a) + O(k * e * \log e)$ 

### Tracking

There are two different tracking process computed depending on the state of the worms: the worms swim overlapped or the worms swim separated. When the worms swim separated the tracking is based on the analysis of the Kalman predictions. The four predictions obtained with the Kalman filter in the previous frame (for both the worms prediction of the head and tail points) are compared with the current positions (head and tail) of the two worms. It is evident that in this case the tracking process has a computational complexity that strictly depend on the computational complexity of the Kalman filter. As explained in [49] the Kalman prediction are obtained computing some matrix operations. The size of these matrixes is fixed because the number of states (position and velocity) is fixed as well as the number of input and output of the discrete system considered. For this reason we can consider that in this case the computational complexity of the tracking phase. When the worms are overlapped the algorithm has to manage the occlusion. After the occlusion the algorithm compute the gender re-identification and the head and tail definition. Using the features values of the two worms before the occlusion two model are created. These two models are compared with the features values of the worms after the occlusion. This process can be done in a constant time. For the head and tail definition the trajectories of the end-points of the skeleton in a fixed number of frames after the occlusion are analysed. This analysis can be done in a constant time.

We can conclude that the computational complexity of proposed method is:  $O(k * n * m) + O(k * b * s^2) + O(k * a) + O(k * e * \log e)$ 

## 3.7 Optimization

### 3.7.1 Output storage optimization

In this paragraph an estimation of the space needed to store the algorithm output is done. At the end of the paragraph some methods are proposed to reduce the space needed. Using the output of the algorithm it has to be possible to generate a binary video with the two worms in foreground. Furthermore it has to be possible to generate the graphs with the feature values over the time. The estimation is done in according to the following assumptions:

- Size of the video is 640 x 480 (the same size used in the evaluation).
- Max value of the worm area is 800 pixels.
- Max value of a radius in the normalized shape of a worm is 50 pixels.
- Max value of the length worm is 120 pixels (skeletonLength).
- The worms touch each other one time every 244 frames (touchRate).
- Max speed of a worm is 50 pixels per second.

For both the worms and for every frame the information stored by the algorithm at the end of an elaboration is the following:

- Current frame 32 bits;
- Coordinates of the head point 2\*10 bits;
- Coordinates of the skeleton points skeletonLength \* 2\*10 bits;
- Normalized shape skeletonLength \* 6 bits;

The length of the worm can be obtained from coordinates of the skeleton points. The area of the worm can be obtained from the normalized shape. The speed of the worm can be obtained from the coordinates of the head point. The luminosity of the worm does not need to be stored because it is used just for the tracking and it is not relevant for the biologists. For every frame we need to store number of the current frame because of the frame rate can be not constant. There is no need to store the number of the times that the worms have touched each other for every frame because it is sufficient to store the information about when the touched have occurred. Considering a video that have duration of 24 hours with a frame rate of 43 frames for second and doing the previous assumptions the space needed to store the output of the algorithm can be estimate as following:

$$\begin{split} N.offrame &= 3.715.200;\\ MaleSpace &= N.offrames*(2*10+skeletonLength*(2*10+6))\\ FemaleSpace &= N.offrames*(2*10+skeletonLength*(2*10+6))\\ Currentframe &= N.offrames*32\\ TouchesSpace &= N.oftouches*touchRate*32\\ Space &= MaleSpace+FemaleSpace+TouchesSpace+Currentframespace \end{split}$$

Current frame = 14,86 MBytes  $MaleSpace = FemaleSpace = 1,166 * 10^{1}0bits = 1,45GBytes$  TouchesSpace = 60,9Kbytes  $TatalSpace \approx 2,9Gbytes$ 

Table 6 shows the space needed to store the original videos of the dataset and compare this number with the space needed to store the output of the algorithm in according to the assumptions done that are valid for the considered dataset. The space needed for the output of the algorithm can be calculated in according to the previous formulas considering a video with 320.630 frames that it is equal to the sum of frames in all the videos of the dataset.

$$\label{eq:currentframe} \begin{split} Currentframe &= 1.282, 5Kbytes = 1, 2825MBytes\\ MaleSpace &= FemaleSpace = 1, 0068 * 10^9 bits = 125, 8473MBytes\\ TouchesSpace &= 5, 2562Kbytes = 0, 0053Mbytes\\ Space &\approx 253, 03Mbytes \end{split}$$

To reduce the space needed to store the output of the algorithm two optimizations are proposed: the first optimization reduce the number of bits needed to store the coordinates of the heads points of the two worms and the second optimization reduce the number of bits needed to store the coordinate of the skeletons points of the two worms.



Figure 36: The position of the worms is defined respect a system of Cartesian axis cantered in the up-left corner of the image.

For every frame, the coordinates of the head point are stored respect a system of Cartesian axes centred in the up-left corner of the image as showed in Figure 36. We propose to store the coordinates of the head point in the first frame of the video respect these Cartesian axes and for all the successive frames we propose to store the coordinates of the head point in a relative way as showed in Figure 37. For a fixed frame k a system of Cartesian axes is defined. These axes are centered in the position of the head point of the worm. The position of the head point of the worm in the frame k +1 can be defined respect to Cartesian axes previously defined. For the assumption done a worm cannot cover a distance bigger than 50 pixels between two successive frames so using this new representation, for every frame of the video after the first one, the coordinate of the head point can be stored with

2\*6 bits instead of 2\*10 bits.



Figure 37: The positions of the worms are defined respect two system of Cartesian axes centred in the positions of the head points in the previous frame.

For every frame the coordinates of the skeleton points are stored respect the fixed point of the image (0,0) (up-left corner). From the previous optimization the coordinate of the head point (first point of the skeleton) can be formulated respect the fixed point of the image (0,0). There is no need to store for all the successive pixel of the skeleton the coordinate respect the point (0,0) because it is know that every point of the skeleton is a 8-neighbour respect the previous one. From the second point of the skeleton to the last one the coordinate of the skeleton points can be stored using the Freeman Chain Code. With this coding the coordinate of the skeletons points can be stored with 2\*3 bits instead of 2\*10bits.

The following table shows the space needed to store the output of the algorithm for the complete dataset using the two described optimization.

Current frame = 1.282, 5K bytes = 1, 2825 MBytesMaleSpace = FemaleSpace = 465.554.760 bits = 58, 1943 MBytesTouchesSpace = 5, 2562 Kbytes = 0, 0053 Mbytes $Space \approx 117.676 Mbytes$ 

	Space needed	
Complete Data Set	4.500,00	Mbytes
Output of the algorithm	252,98	Mbytes
Optimized output of the algorithm	117,67	Mbytes

Table 6: Needed space to store the output of the algorithm

In the work [57] a novel approach is proposed to represent the worm in a compressed way. The method is based on the, so called, eigenworms. The posture of a worm can be represented with just four numbers: the amplitudes along each dimension when the shape is projected onto the eigenworms.

### 3.7.2 Elaboration time optimization

The bottle neck of the computational complexity are the morphological operations that we use during the pre-elaboration phase to generate the binary image. The idea to speed up the elaboration is to generate the binary image in parallel with the elaboration that we do on that image for the features extraction. We can divide the instructions in two set: the set of the instructions that aim to generate the binary image using morphological operations (set B) and the set of the instructions that aim to interpret the binary image extracting the worms features from the binary image and doing the worms tracking (set T). From the evaluation it can be seen that the time required for the elaboration of the instructions of the set B is much higher than the time to elaborate the instructions of the set T. The following image shows how a sequential elaboration is done where:

- $B_i$  means elaboration of the instructions that belongs to the set B to create the binary image of the frame i;
- $T_i$  means elaboration of the instructions that belongs to the set T to extract features and track the worms in the frame i

$$\xleftarrow{\text{B1}} \xleftarrow{\text{T1}} \xleftarrow{\text{B2}} \xleftarrow{\text{T2}} - - - \xleftarrow{\text{Bn}} \xleftarrow{\text{Tn}}$$

Figure 38: Sequential elaboration.

We can improve the elaboration time considering that the production of the binary images can be anticipated. Generate the binary images from the original frames is a process that can be done out of order instead the elaborations done on the binary images have to be done in order from the first binary image to the last one. The idea is to do the elaboration of these two different types of elaboration in parallel. The following image show how the elaboration time can be reduced using a parallel approach:

Figure 39: Parallel elaboration.

### 3.8 Limits of the method

The module that does the pruning of the skeleton of the connected regions works only if there are no loops in the original skeleton. The method showed for the head and tail definition can be used in every type of application in which the following conditions occur simultaneously:

- The skeleton of the object has a shape of a line with just two end-points.
- During the trajectory analysis the worm should move in forward for a time greater than half of the duration of the trajectory analysis.
- The points used to analyse the trajectory of the worm cannot move in the opposite direction respect the direction of the movement of the middle point of the worm skeleton.

To compute the normalize shape it is necessary that the connected region that represent the worm in the binary image can be reduced to a skeleton without branches. For the tracking of the worms when they swim separately the proposed method is based on the Kalman filter so the limits of the Kalman filter have a directly repercussion on the limits of the proposed method. In particular a technological limitation is imposed: the camera used to record the videos has to be able to record videos with a sufficient frame rate in according to the speed of the worms. The gender re-identification method strictly depends from the chosen features and from the number of analysed frames. Given a set of N feature an N-dimensional space can be considered. Every dimension of the space defines the values of one of the chosen feature. As discussed in the previous paragraphs  $model_1$  and  $model_2$  represents the values of the features of the male and of the female worm before the occlusion. They are represented in this N-dimensional space as two points that we can call  $m_1$  and  $f_1$ . For a fixed frame after the occlusion the values of the features of the two worms are also represent as two points in the N-dimensional space that we can call  $m_2$  and  $f_2$ . If the sum of the Euclidean distances between the points  $m_1$  and  $m_2$  and the points  $f_1$  and  $f_2$  is smaller than the sum of the Euclidean distances between the points  $m_1$  and  $f_2$  and the points  $f_1$  and  $m_2$  the correct decision is taken otherwise a wrong decision is taken. The gender re-identification method can be used in every application in which exist a set of features that can be used to build models, before an occlusion, that properly represent the tracked objects after the occlusion.

# 4 Design and implementation

### 4.1 System architecture

In the previous chapter the method that allows the worm tracking and features extraction is described. In this chapter we propose our implementation of the method. The environment chosen to implement the algorithm is MATLAB (matrix laboratory). MATLAB is optimized to compute matrix manipulations which make it a perfect tool for image processing and video analysis applications. The design of the algorithm is based on functional modules and a top-down approach is used to describe the modules that compose the system. From the highest point of view the system can be represented with three main modules: worm detection, features extraction, worms tracking. In the following paragraphs every one of this modules is described the sub-modules that compose it.



Figure 40: Main system overview

The worm detection module bring as input the video frame an produce a binary image with a black background and connected regions in foreground. These connected regions are marked as blobs. The features extraction module proceeds with the extraction of the features from these blobs. The last step is done by the worms tracking module that property label the blobs with the gender allowing the taking of them during the video sequence. Furthermore for every blob the module has to mark the head and the tail points.

### 4.1.1 Worm detection

This module implements the background subtraction technique to obtain a binary image in which the worms appear in foreground as connected regions.

- Input: a grey-scale frame taken from the video sequence.
- Output: binary image with the worms in foreground.
- Initialization parameters:
  - Minimum and maximum luminosity value of the worms.
  - Minimum value of the worm area.
  - Background mask.

It is important to remark that to build the background mask a ROI has been used. Every pixel out of the ROI has been marked as background in the background mask. In this way every time that the background subtraction technique is applied, the pixels of the image that are out of the ROI are automatically excluded from the elaboration. For this reason there is no need to put the ROI as an input parameter of this method because it is implicitly included in the background mask. The operations done by this module are:
- 1. Do a double luminosity threshold based on the minimum and maximum luminosity values of the worms.
- 2. Subtract to the obtained image the background mask.

In the binary image obtained from the previous steps some noise could appears in foreground. It could be also possible that the connected regions that represent the worms contain some small holes inside. To correctly detect the worms the following operations are done:

- 1. Threshold on the area of the connected regions based on the minimum value of the worm area.
- 2. A morphological operation of closing.

The result of this elaboration is a binary image with two connected regions that represent the two worms or just one connected region if the worms are touching each other.

### 4.1.2 Features extraction

The input of this module is the output of the previous one: a binary image with the blobs that represents the worms in foreground. If the worms do not touch each other, for every detected connected regions this module define the value of all the features discussed in the paragraph 3.4.

- Input: a binary image in which the worms appear in foreground.
- Output: detection of the worms inside the binary image and definition of the feature values of the worms.



Figure 41: Features extraction module

If the worms are touching each other no feature value is extracted and the output of the module is the identification of the single connected region that represents the two worms. If the worms are swimming separately for both the connected regions the value of all the features are extracted.

#### 4.1.3 Worms tracking

From the previous module, the connected regions that represent the two worms are detected and all the worm features are extracted from these regions. The goal of this module is to track the head points of these regions and label them as male worm and female worm.



Figure 42: Worms tracking module

When there are two connected regions means that the worms swim separated. In this case the sub-module Tracking based on Kalman filter is used. When there is one connected region means that the worms touch each other and are represented by the same connected region. In this case the submodule Occlusion manager is used. Regarding the tracking of the worms when they swim separated, the output of the tracking for the previous frame is used to obtain the Kalman predictions. These predictions are compared with the positions of the regions obtained from the worm detection of the current frame. The eight hypothesis method explained in the paragraph 3.5.2 is used to define the correct association among these points. When the correct association among this points has been defined it became automatically know the gender labels of the two connected regions and also the head and tail labels. The occlusion manager module has to re-identify the gender of the two worms after an occlusion and also, for both the worms, define the head and tail positions.



Figure 43: Occlusion manager module

Both the sub-modules Gender re-identification and Head and Tail identification need to analyse the features of the worms in some frames after the occlusion. To use these two modules the tracking of the two connected regions represented in the binary image after the occlusion is needed. This tracking is done using the module Tracking based on Kalman filter. This module is the same module used for the worms tracking with the only difference that instead to label the regions as male and female worm the regions are labelled as  $region_1$  and  $region_2$ . After the tracking of the two regions all the features of these two regions has been stored and the modules Analysis of successive frames and Head and Tail identification can be used. The module Gender re-identification use the values of the features before the occlusion to build two models: a model that represent the male worm and a model that represent the female worm, as described in the paragraph 3.5.4. This models are compared with the values of the features tracked after the occlusion. The comparison method is described in the paragraph 3.5.4. After this comparison the two regions:  $region_1$  and  $region_2$  are labelled as male and female worm. The module Head and Tail definition analyse the trajectory of the two regions tracked after the occlusion and define which are the head points of the two regions. The analysis of the trajectory of the two regions is done as described in the paragraph 3.5. In the end, the occlusion manager is used to track the regions present in the binary image in the frames after the

occlusions. The features of these regions are extracted during the tracking and then are used to correctly re-identify the gender and the position of the head points of the two worms.

## 4.2 Algorithm configuration

This is the description of the algorithm parameters:

- FEATURE\_OBSERVATION\_WINDOW\_SIZE
  - Defines the size of the observation windows used to create the male and the female model to aim at the gender re-identification. It also represents the number of the frames that are analysed after an occlusion to find the correspondence between the worm models and the regions.
- TRAJECTORY\_OBSERVATION\_WINDOW\_SIZE Defines the size of the observation windows used to study the trajectory of the end-points of the skeleton regions after an occlusion to identify the head and the tail points.
- DEAD\_WORM\_OBSERVATION\_WINDOW\_SIZE Defines the size of the observation windows used to understand if some of the regions present in the binary image after an occlusion represent a dead worm.
- $DEAD_WORM\_SPEED$  and  $MAX\_DEAD\_WORM\_MOVEMENT$ If in the frames that belong to the dead worm observation window a region has a speed lowest of  $DEAD\_WORM\_SPEED$  value and the distance covered during the occlusion (distance between the last position before the occlusion and the first position after the occlusion of the point w( $\frac{1}{2}$ ,k,id). is greater than  $MAX\_DEAD\_WORM\_MOVEMENT$ , the region is detected as dead worm. Obviously this elaboration is done after the gender re-identification otherwise it is not possible to associate the first position of the region after the occlusion with the last position of the worm before the occlusion.

### • *MIN\_TAIL\_HEAD\_DISTANCE*

If the Euclidean distance between the points w(1, k, id) and w(0,k, id) is less than this value means that the worm is closing himself in a circle. This situation is managed as an occlusion because if the skeleton

of the worm has a circle shape it is not possible to get the end-points of the skeleton and identify the head and tail points. The worm remains occluded until the end-points of the its skeleton reappear distant from each other for more than this value.

#### • *MIN\_T\_H\_KALMAN\_DISTANCE*

If the Euclidean distance between the points  $w(\frac{1}{3},k,id)$  and  $w(\frac{2}{3},k,id)$ is less than this value means that the worm is collapsing on himself. This situation generally happens when the worms die. This situation is managed as an occlusion because if the worm collapses in a ball shape there is any information about the skeleton and the end-points that can be got. The worm remains occluded until the points  $w(\frac{1}{3},k,id)$  and  $w(\frac{2}{3},k,id)$  reappear distant from each other for more than this value.

#### • *MIN\_KALMAN\_DISTANCE*

If the Euclidean distance between the points  $k(\frac{1}{3},k,male)$  and  $k(\frac{1}{3},k,female)$ or  $k(\frac{1}{3},k,male)$  and  $k(\frac{2}{3},k,female)$  or  $k(\frac{2}{3},k,male)$  and  $k(\frac{1}{3},k,female)$ or  $k(\frac{2}{3},k,male)$  and  $k(\frac{2}{3},k,female)$  is less than this value means that the movement predictions are too approximate and there is a high probability to get an error during the association process. This situation generally happens when these three conditions occur simultaneously:

- The worms swim too fast for the frame rate of the video;
- The worms swim near without touching each other;
- The worms do circular trajectories.

When this situation happens, the association method explained in the paragraph 3.2 is not robust because movement predictions are too near. The frames in which this situation happens are considered as frames in which the worms are occluded and the situation is managed as if the worms were crossing each other.

### • *HEAD\_TAIL\_LENGTH\_PERCENTAGE*

This value defines how the worm body can be subdivided and also which points along the skeleton points are used as key points to track. Setting this value to  $\frac{1}{3}$  means that the total length of the skeleton can be divided in three parts that have the same length:

- Tail: from the point w(0,k,id) to the point  $w(\frac{1}{3},k,id)$ ;

- Body: from the point  $w(\frac{1}{3},k,id)$  to the point  $w(\frac{2}{3},k,id)$ ;
- Head: from the point  $w(\frac{2}{3},k,id)$  to the point w(1,k,id);

The tuning of the parameters of the algorithm has been done on the videos that belong to the folder: "160409\_*Spawning\_pairs\_Carmine*" because in these videos appear almost every possible situation and the evaluation has been done maintaining the same configuration of these parameters.

The following table show the configuration used for the evaluation of the algorithm:

Parameters	Default values
FEATURE_OBSERVATION_WINDOW	5
TRAJECTORY_OBSERVATION_WINDOW	100
DEAD_WORM_OBSERVATION_WINDOW	10
DEAD_WORM_SPEED	2,6
MAX_DEAD_WORM_MOVEMENT	39
MIN_TAIL_HEAD_DISTANCE	12
MIN_T_H_KALMAN_DISTANCE	12
MIN_KALMAN_DISTANCE	18
HEAD_TAIL_LENGTH_PERCENTAGE	0,33
α <sub>n</sub>	1
α <sub>a</sub>	1
α <sub>m</sub>	1
α <sub>1</sub>	1
α <sub>p</sub>	0
Min worms area	50
Max worms area	800

Table 7: Parameters of the algorithm

# 5 Experimental evaluation

To evaluate the performance of the methodological approach proposed in this work a Matlab application has been realized. The application has been tested on the data set of worm videos provided by the Max F. Perutz Laboratories of Vienna.

## 5.1 Data set description

The videos of the worms are in black and white with a size of 1280x960 pixels and a variable frame rate: from 30 to 60 frames per second (43 frames per second in average). To speed up the elaboration time every video has been resized to 640x480 pixels. From the complete data set of videos have been removed: the parts in which appears just one worm, the parts in which the worms are dead (nothing happens), the parts in which the biologists introduce with a syringe chemical elements in the arena. After this filtering we can summarize the information about the data set in this way: the number of videos is 25, the total duration of the videos is 2 hours, 7 minutes and 16 seconds, and the total number of frames is 320.630 (elaborated videos and more material about the evaluation done and the obtained results can be found on our web site "http://www.prip.tuwien.ac.at/research/worms.php"). The biologists have organized the videos of the data set in folders in according to the data in which the videos have been recorded. The following table describes how the videos have been organized:

Folders name	Number of videos	Total duration of the videos	Total number of frames
1511_11&12_spawn_60fps_Noldus_1280x960	9	01:06:50	173.886
160409_Spawning_pairs_Carmine	5	00:22:03	57.605
150905_60fps_Movies_Noldus	4	00:21:06	54.857
150826&27_Movies_Noldus Setup	4	00:10:05	18.175
150821_30fps_movies_Steph&Daniel	3	00:07:06	16.107

Table 8: Organization of the videos in folders

Original video folder name	Id
160409_Spawning_pairs_Carmine	Trial1
150905_60fps_Movies_Noldus	Trial2
150821_30fps_movies_Steph&Daniel	Trial3
1511_11&12_spawn_60fps_Noldus_1280x960	Trial4
150826&27_Movies_Noldus Setup	Trial5

Table 9: Videos folder convection

### 5.2 Sequence Frame Detection AccuracyDistance

In the work of Kasturi [58] an index to measure the accuracy of the detection is defined. This index can be used in all the tracking applications in which a ground truth is available and the objects tracked are represented in the video sequence by points. In our case the two worms can be represented by the head points and the ground truth is available because it has been defined manually watching the videos of all the data set. Let D be a matrix having a dimension of 2x2 that defines the Euclidean distance between the detected head points of the two worms and the exact position of the two worms in according to the ground truth. The D matrix is normalized to one dividing every element for the length of the diagonal of the image. For a fixed frame t of the video sequence, the accuracy of the detection is defined by the index: Frame Detection AccuracyDistance (FDA-D) in according to Equation 16.

$$FDA - D(t) = \frac{\sum_{i=1}^{N_{mapped}^{t}} (1 - d'_{i})}{\frac{N_{G}^{t} + N_{D}^{t}}{2}}$$
(16)

$$FDA - D(t)_{male} = 1 - \frac{\sqrt{(male_{x(t)_D} - male_{x(t)_G})^2 + (male_{y(t)_D} - male_{y(t)_G})^2}}{diagonal}$$
(17)

$$FDA - D(t)_{female} = 1 - \frac{\sqrt{(female_{x(t)_D} - female_{x(t)_G})^2 + (female_{y(t)_D} - female_{y(t)_G})^2}}{diagonal}$$
(18)

$$FDA - D(t) = \frac{FDA - D(t)_{male} + FDA - D(t)_{female}}{2}$$
(19)

Where  $N_{mapped}^t$  is the number of number of mapped object sets in the frame t,  $N_G^t$  is the number of objects present in the frame t in according to the ground truth,  $N_G^t$  is the number of objects present in the frame t

in according to the detection of the algorithm,  $d'_i$  represents the distance between the i-th mapped pair. This measurement be averaged over the set of all frames in a sequence by:

$$SFDA - D(s) = \frac{\sum_{t=1}^{N_{frames}} FDA - D(t)}{\sum_{t=1}^{N_{frames}} \exists (N_G^t ORN_D^t)}$$
(20)

For the average over all the set of sequences we have:

$$ASFDA - D = \frac{\sum_{s=1}^{S} SFDA - D(s)}{S}$$
(21)

Where S is the total number of all the video sequences in the data set.

### 5.3 Multiple Object Tracking Accuracy (MOTA)

In the work of Kasturi [58] an index to measure the accuracy of the tracking is defined. This index can be used in all the tracking applications in which a ground truth is available and the objects tracked are represented in the video sequence by points. In our case the two worms can be represented by the head points and the ground truth is available because it has been defined manually watching the videos of all the data set. The MOTA measurement takes in account: the missed counts, the false alarm counts and the number of switches in the system output ID for a given reference ground truth.

$$MOTA = 1 - \frac{\sum_{i=1}^{N_{frames}} (c_m(m_i) + c_f(f_{p_i}) + \log_e(id_{switches}))}{\sum_{i=1}^{N_{frames}} N_G^i}$$
(22)

For a fixed frame *i*:  $m_i$  is the number of missed tracks,  $f_{p_i}$  is the total number of false alarm tracks and  $id_{switches}$  is the total number of ID switches made by the system output for any given reference ID and  $N_G^i$  is the number of tracked objects in the frame i in according to the ground truth. For the evaluation done, the number of missed tracks is always zero because we do not consider an error when the algorithm misses the track of the worms when the worms touch each other (because the algorithm is able to re-identify the worms after the occlusion). We have decided to be more strict weighting the penalty of a switch in a linear way instead of a logarithm way. The reason is that in our application just two objects have to be tracked so the variable  $id_{switches}$  become a binary variable: one if the algorithm does the switch and zero otherwise. When the algorithm considers a worm occluded because it

closes itself in a circle, for the ground truth there are always two worms that have to be tracked. For this reason the value of  $N_G^i$  is zero if the worms touch each other (occlusion) and two otherwise. According to these considerations the number of the Multiple Object Tracking Accuracy can be reformulated as:

$$MOTA = 1 - \frac{N.ofswitches + N.offalse\_positive}{2*N.offrames\_without\_occlusions}$$
(23)

### 5.4 Features ranking

In this paragraph an evaluation of the chosen features is done. Standard ranking methods as Support Vector Classification [34] cannot be used in our case. The reason is that the values of the features that define a model for the male and for the female worm are always different among the videos and also during the same video sequence. For this reason in the proposed method new models that describe the two worms are built and used for every worm interaction. The ground truth that defines for every one of the 1.308 gender re-identification decisions the correct gender re-identification has been defined manually, watching the original videos. The method that we use to rank the features is the following: evaluate every feature individually defining the number of correct gender re-identification decisions that have to be taken. The ranking is done sorting the features in according to the number of correct decisions taken. The following tables summarize the obtained results:

Trial 1				
Rank	Feature	Number of correct gender re-identification decisions	Percentage of correct gender re-identification decisions	
	Luminosity	212	97,25 %	
2	Length	192	88,07 %	
3	Normalized shape	189	86,70 %	
4	Area	186	85,32 %	

		Trial 2	
Rank	Feature	Number of correct gender re-identification decisions	Percentage of correct gender re-identification decisions
	Length	122	95,31 %
2	Normalized shape	121	94,53 %
3	Area	117	91,41 %
4	Luminosity	109	85,16 %

Trial 3				
Rank	Feature	Number of correct gender re-identification decisions	Percentage of correct gender re-identification decisions	
	Area	148	94,27 %	
2	Length	145	92,36 %	
3	Normalized shape	145	92,36 %	
4	Luminosity	128	81,53 %	

Trial 4					
Rank	Feature	Number of correct gender re-identification decisions	Percentage of correct gender re-identification decisions		
1	Area	655	97,62 %		
2	Length	653	97,32 %		
3	Luminosity	641	95,53 %		
4	Normalized shape	633	94,64 %		

Trial 5				
Rank	Feature	Number of correct gender re-identification decisions	Percentage of correct gender re-identification decisions	
1	Luminosity	138	99,28 %	
2	Length	136	97,84 %	
3	Area	127	91,37 %	
4	Normalized shape	122	87,77 %	

Max F. Perutz Laboratories of Vienna Data Set				
Rank	Feature	Percentage of correct gender re-i dentification decisions		
1	Length	1.248/1308	95,41 %	
2	Area	1.233/1308	94,27 %	
3	Luminosity	1.228/1308	93,88 %	
4	Normalized shape	1.210/1308	92,51 %	

Table	10:	Features	ranking
-------	-----	----------	---------

## 5.5 Obtained results

The evaluation has been done on a Personal Computer with the following features:

- Operating system: Windows 7, 64 bit version;
- RAM: 8 GB;
- CPU: Intel Quad Core i5-760, 2,8 GHz;
- GPU: Nvidia GeForce GTX 460;
- Motherboard: Asus P7H55.
- Hard Disk: Western Digital Black WD1002FAEX, 1 TB, 7200 RPM.

The following table shows the results obtained for every videos folder:

Id	Trial1	Trial2	Trial3	Trial4	Trial5
Number of videos	5	4	3	9	4
Duration	00:22:03	00:21:06	00:07:06	01:06:50	00:10:05
Total number of frames	57.605	54.857	16.107	173.886	18.175
Number of gender re-identifications	218	128	157	666	139
Number of wrong gender re-identifications	0	0	1	1	1
Percentage of correct gender re-identifications	100%	100%	99,36%	99,85%	99,28%
Number of head and tail identifications	513	319	341	1.494	292
Number of wrong head and tail identifications	8	10	3	16	5
Percentage of correct head and tail identifications	98,44%	96,86%	99,12%	98,92%	98,29%
Frames with wrong gender identification	0	0	2.879	8.821	3.510
Percentage of frames with correct gender identification	100%	100%	82,12%	94,93%	80,69%
Frames with wrong head and tail identification	5.322	4.006	764	2.522	863
Percentage of frames with correct head and tail identification	90,76%	92,70%	95,26%	98,55%	95,25%

## Table 11: Evaluation results

The following table summarizes the results obtained on the whole data set:

Number of videos	25
Duration	02:07:10
Total number of frames	320.630
Number of gender re-identifications	1.308
Number of wrong gender re-identifications	3
Percentage of correct gender re-identifications	99,77%
Number of head and tail identifications	2.959
Number of wrong head and tail identifications	42
Percentage of correct head and tail identifications	98,58%
Frames with wrong gender identifications	15.210
Percentage of frames with correct gender identifications	95,26%
Frames with wrong head and tail identifications	13.477
Percentage of frames with wrong head tail associations	95,80%

#### Max F. Perutz Laboratories of Vienna Data Set

Table 12: Evaluation results in summary

It would seem strange that the percentage of correct gender re-identifications is greater than the percentage of wrong head and tail identifications and the percentage of the frames with a correct gender re-identifications is lower than the percentage of frames with a wrong head and tail identifications. The reason is that a wrong head and tail identification does not have repercussions on the following one because every one of this decision is based on different and independent trajectory studies; instead if a wrong gender re-identification decision is taken there is a high probability that the worms gender will continue to be confused in the following gender re-identification decisions. As regards the gender re-identifications we have observed that in the most of the videos we have got zero gender re-identification errors and in three videos, from a certain frame to the end of the video, the gender of the worms has been confused. As regards the head and tail identifications we have observed that in the majority of the case in which the algorithm takes a wrong decision the worms lose vitality and start to stay quite or do not move. In these situations the trajectories of the worms are quite casual and depend on the movement of the water. Instead when the worms swim in a natural way is very unlikely that the algorithm takes wrong decision. The following table show the values of the ASFDA-D obtained for all the video sequences of every Trial folder.

Trial	SFDA-D	ΜΟΤΑ
Trial 1	0,998	1
Trial 2	0,999	1
Trial 3	0,955	0,99
Trial 4	0,970	0,99
Trial 5	0,923	0,99

Table 13: Evaluation results in summary

From the evaluation done, the obtained value of the ASFDA-D on all the video sequences of the data set is: 0,977. The following table show the values of the MOTA obtained for all the video sequences of every Trial folder. From the evaluation done, the obtained value of the MOTA on all the video sequences of the data set is: 0,99.

## 6 Conclusion

In this work, a method to extract features and track Platynereis dumerilii is proposed. It is able to handle occlusions and maintain the identity of the tracked worms with the help of a novel feature, the normalized shape. The normalized shape allows the comparison of the shape of worms regardless of their deformation. It is used along with other features to correctly re-identify the worms after occlusions. The proposed method for the re-identification and the trajectory analysis to assign head and tail are not limited to the presented application but can be applied to other tracking problems. The decision indeed takes into account the trajectory of the end-points of the skeleton and does not focus on anatomical or shape properties. Especially the normalized shape is a suitable representation for all kinds of non-rigid objects having a main axis.

The method shown for the gender re-identification of worms can be adapted for other tracking problems, where the correct associations has to be made after an occlusion. Assuming that the objects which are visible before an occlusion belong to a particular class, models for every class can be built. Using these appearance models and the comparative method shown in this paper the best possible decision can be taken defining the assignment class of the objects after the occlusion. For different applications and association problems the features used to represent the object types can be different. This also allows the algorithm to handle more than two objects and will be used in the future to track more than two worms.

Experimental evaluations on videos of the duration of more than two hours showed that the proposed approach is able to reliably analyse the nuptial dance of the worms. In 99.77% of the cases the gender of the worms was correctly re-identified after an occlusion. The head and the tail where correctly identified in 98.58% of the cases.

Using the outputs of the algorithm generated from the videos dataset, the biologists of the Max F. Perutz laboratory of Vienna are finally able to continue the study of the genetic of the Platynereis dumerilii worms. In fact the last research work [5] was published in 1974. The research was stopped for so many years because it was too hard for a human to extract relevant worms features by hand watching many hours of videos. Now the biologists have the possibility to figure out the impact of the different genetic alterations for all the videos present in the dataset by analysing the values of the extracted features from plots and tables.

## 6.1 Future work and biologists expectations

Now that a methodological approach for the features extraction and the tracking of the worms has been defined, the next step would be to define a method that aim to understand the behaviour of the worms analysing the worms trajectories and all the features describes in the previous part of the document. Biologists are interested in understanding how the behaviour of the worms changes with different genetic alterations of the worms DNA so the main goal of the future work would be to define an algorithm that automatically recognize these different type of worms behaviours. To automatically identify these different worms behaviours a good start-point would be to find recurring pattern among the trajectories of the worms and to use them to understand the behaviours of the worms. Using this method it will be possible to divide the worms videos according to semantic information about the behaviour of the worms, for instance a possible semantic partition of a video could be:

- 1. Pre-spawning;
- 2. First meet;
- 3. Non-engaged pre-spawning;
- 4. Engaged pre-spawning;
- 5. Engaged spawning;
- 6. Post-spawn after-release;

With the proposed method the biologists have the possibility to automatically analyse videos that have duration of days. A novel approach to store the output of the elaboration, optimizing the space needed, should be defined.

# References

- R. Shingai, Durations and frequencies of free locomotion in wild type and GABAergic mutants of Caenorhabditis elegans, Neuroscience Research, 2000.
- [2] J. Zantke, S. Bannister, V. B. V. Rajan, F. Raible and K. T. Raible, Genetic and Genomic Tools for the Marine Annelid Platynereis dumerilii, Genetics, 2014.
- [3] M. Bono, D. M. Tobin, M. W. Davis, L. Avery and C. I. Bargmann, Social feeding in Caenorhabditis elegans is induced by neurons that detect aversive stimuli, Nature, 2002.
- [4] M. G. Bentley, P. J. Olive and K. Last, Sexual Satellites, Moonlight and the Nuptial Dances of Worms: the Influence of the Moon on the Reproduction of Marine Animal, Earth, Moon, and Planets, 1999.
- [5] S. Brenner, The genetics of Caenorhabditis elegans, Genetics, 1974.
- [6] T. C. e. S. Consortium, Genome sequence of the nematode C. elegans: a platform for investigating biology, Science, 1998.
- [7] R. Shingai, Durations and frequencies of free locomotion in wild type and GABAergic mutants of Caenorhabditis elegans, Neuroscience Research, 2000.
- [8] D. Pucher, 2D tracking of Platynereis dumerilii worms during spawning, Vienna: Technical report, Pattern Recognition and Image Processing Group, Vienna University of Technology, 2016.
- [9] H. Laura Anne, S. Emily, K. Rex, Z. Guotong and W. R. Schafer, Serotonin Modulates Locomotory Behavior and Coordinates Egg-Laying and Movement in Caenorhabditis elegans, Neurobiology, 2001.
- [10] E. Maggio and A. Cavallaro, Video Tracking: Theory and Practice, John Wiley and Sons, 2011.
- [11] M. Nixon, Feature Extraction and Image Processing for Computer Vision (Third Edition), Elsevier, 2012.

- [12] S. Ince and J. Konrad, Occlusion-Aware Optical Flow Estimation, IEEE Transactions on Image Processing, 2008.
- [13] W. L. Zhao and C. W. Ngo, Flip-invariant SIFT for copy and object detection, IEEE Transactions on Image Processing, 2013.
- [14] A. Elgammal, D. Harwood and L. Davis, Non-parametric Model for Background Subtraction, 6th European Conference on Computer Vision Dublin, 2000.
- [15] S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, In Proceeding of the IEEE Conference on Computer Vision and Patter Recognition, 1998.
- [16] R. E. Kalman, A New Approach to Linear Filtering and Prediction Problems, Transactions of the ASME - Journal of Basic Engineering, 1960.
- [17] M. Vento, P. Foggia, G. Percannella, A. Saggese and R. D. Lascio, A real time algorithm for people tracking using contextual reasoning, Computer Vision and Image Understanding, 2013.
- [18] B. Yang, C. Huang and R. Nevatia, Learning Affinities and Dependencies for Multi-Target Tracking using a CRFModel, Proceedings of IEEE Computer Vision and Pattern Recognition, 2012.
- [19] S. Pellegrini, A. Ess, K. Schindler and L. Gool, You'll never walk alone: Modeling social behavior for multi-target tracking, 2009 IEEE 12th International Conference on Computer Vision, 2009.
- [20] C. Benjamin, B. David, M. Philip and M. Jitendra, A real-time computer vision system for vehicle tracking and traffic surveillance, Transportation Research Part C: Emerging Technologies, 1998.
- [21] D. Koller, J. Weber and J. Malik, Robust Multiple Car Tracking with Occlusion Reasoning, Berkeley: Technical Report, EECS Department, University of California, 1993.
- [22] M. Betke, E. Haritaoglu and L. S. Davis, Real-time multiple vehicle detection and tracking from a moving vehicle, Machine Vision and Applications, 2000.

- [23] W. L. Lu, J. A. Ting, J. J. Little and K. P. Murphy, Learning to Track and Identify Playersfrom Broadcast Sports Videos, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013.
- [24] W. Luo, T. Kim, B. Stenger, X. Zhao and R. Cipolla, Bi-label Propagation for Generic Multiple Object Tracking, 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014.
- [25] M. Betke, D. E. Hirsh, A. Bagchi, N. I. Hristov, N. C. Makris and T. H. Kunz, Tracking Large Variable Numbers of Objects in Clutter, 2007 IEEE Conference on Computer Vision and Pattern Recognition, 2007.
- [26] F. Ying, Visual Ants Tracking, Bristol: Ph.D. dissertation, Computer Science, University of Bristol, 2004.
- [27] Z. Khan, T. Balch and F. Dellaert, An MCMC-based Particle Filter for Tracking Multiple Interacting Targets, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005.
- [28] C. Spampinato, Y. H. Burger, G. Nadarajan and R. Fisher, Detecting, Tracking and Counting Fish in Low Quality Unconstrained Underwater Videos, The International Conference on Computer Vision Theory and Applications, 2008.
- [29] E. Fontaine, A. H. Barr and J. W. Burdick, Model-Based Tracking of Multiple Worms and Fish, ICCV Workshop on Dynamical Vision, 2007.
- [30] E. Meijering, O. Dzyubachyk, I. Smal and W. A. Cappellen, Tracking in Cell and Developmental Biology, Seminars in Cell and Developmental Biology, 2009.
- [31] K. Li, E. D. Miller, M. Chen, T. Kanade, L. E. Weiss and P. G. Campbell, Cell population tracking and lineage construction with spatiotemporal context, Medical Image Analysis, 2008.
- [32] H. Steven J., C. Wagner Steuer and S. a. A. G. Cornelia, Keeping track of worm trackers, Oliver Hobert, 10 September 2012. [Online]. Available: http://www.wormbook.org/chapters/tracking.pdf.

- [33] O. Yizhar, L. E. F. T. J. Davidson, M. Mogri and K. Deisseroth, Optogenetics in Neural Systems, Neuron, 2011.
- [34] G. D. Tsibidis and N. Tavernarakis, Nemo: a computational tool for analyzing nematode locomotion, BMC Neuroscience, 2007.
- [35] E. Yemini, High-throughput, single-worm tracking and analysis in Caenorhabditis elegans, Cambridge: Ph.D. dissertation, Emmanuel College, University of Cambridge, 2011.
- [36] S. Nicholas, G. Andrew C., R. Catharine H. and K. Rex A., High-throughput behavioral analysis in C. elegans, Nature Methods, 2011.
- [37] W. S. Jason and W. Z. Wen, Track-A-Worm, An Open-Source System for Quantitative Assessment of C. elegans Locomotory and Bending Behavior, Public Library of Science, 2013.
- [38] D. Ramot, B. E. Johnson, T. L. Berry, J. L. Carnel and M. B. Goodman, The Parallel Worm Tracker: A Platform for Measuring Average Speed and Drug-Induced Paralysis in Nematodes, Public Library of Science, 2008.
- [39] S. H. Chalasani, N. Chronis, M. Tsunozaki, J. M. Gray, D. Ramot, M. B. Goodman and C. I. Bargmann, Dissecting a circuit for olfactory behaviour in Caenorhabditis elegans, Nature, 2007.
- [40] G. Tsechpenakis, L. Bianchi, D. Metaxas and M. Driscoll, A novel computational approach for simultaneous tracking and feature extraction of C. elegans populations in fluid environments, IEEE Transactions on Biomedical Engineering, 2008.
- [41] S. Faumont, G. Rondeau, T. R. Thiele, K. J. Lawton, K. E. McCormick, M. Sottile, E. S. H. W. M. R. O. Griesbeck, C. Q. Doe and S. R. Lockery, An Image-Free Opto-Mechanical System for Creating Virtual Environments and Imaging Neuronal Activity in Freely Moving Caenorhabditis elegans, Public Library of Science, 2011.
- [42] L. A. Hardaker, E. Singer, R. Kerr, G. Zhou and W. R. Schafer, Serotonin modulates locomotory behavior and coordinates egg-laying

and movement in Caenorhabditis elegans, Journal of Neurobiology , 2003.

- [43] R. Dhawan, D. B. Dusenbery and P. L. Williams, Comparison of lethality, reproduction and behavior as toxicological endpoints in the nematode Caenorhabditis elegans, Journal of Toxicology and Environmental Health, 1999.
- [44] N. Roussel, C. Morton, F. P. Finger and B. Roysam, A computational model for C. elegans locomotory behavior: application to multiworm tracking, IEEE Transactions on Biomedical Engineering, 2007.
- [45] H. M. Kuang, Tracking and Analysis of Caenorhabditis Elegans Behavior Using Machine Vision, San Diego: Ph.D. dissertation, Electrical and Computer Engineering, University of California, 2008.
- [46] K. Hoshi and R. Shingai, Computer-driven automatic identification of locomotion states in Caenorhabditis elegans, Journal of Neuroscience Methods, 2006.
- [47] F. Raible and K. T. Raible, Platynereis dumerilii, Current Biology, 2014.
- [48] Y. Chang, J. H. Kao, J. M. Pinilla, J. Dong and F. B. Prinz, Geometric Modelling: Medial Axis Transform (MAT) of General 2D Shapes and 3D Polyhedra for Engineering Applications, International Federation for Information Processing, 1998.
- [49] N. Kovvali, M. Banavar and A. Spanias, An Introduction to Kalman Filtering with MATLAB Examples, Morgan and Claypool, 2013.
- [50] P. Soille, Morphological Image Analysis: Principles and Applications (2nd Edition), Secaucus, NJ: Springer-Verlag, 2003.
- [51] T. Draxler and R. Chai, Root mean square error (RMSE) or mean absolute error (MAE)? Arguments against avoiding RMSE in the literature, Geoscientific Model Development, 2014.
- [52] D. Comaniciu, V. Ramesh and P. Meer, Kernel-based object tracking, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003.

- [53] R. A. Fisher, Statistical Methods for Research Workers, 13th Ed., Hafner, 1958.
- [54] J. W. Osborne, Best Practices in Quantitative Methods, Louisville: SAGE Publications, Inc , 2008.
- [55] D. T. Lee, Medial Axis Transformation of a Planar Shape, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1982.
- [56] A. E. X. Brown, E. I. Yemini, L. J. Grundy, T. Jucikas and W. R. Schafer, A dictionary of behavioral motifs reveals clusters of genes affecting Caenorhabditis elegans locomotion, Proceedings of the National Academy of Sciences, 2012.
- [57] R. Kasturi, D. Goldgof, P. Soundararajan and J. Zhang, Framework for Performance Evaluation of Face, Text, and Vehicle Detection and Tracking in Video: Data, Metrics and Protocol, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009.
- [58] J. Brank, M. Grobelnik, N. M. Frayling and D. Mladenic, Feature Selection Using Linear Support Vector Machines, Microsoft Research, 2002.