Tracking earthworm locomotion using image processing

Sai Saradha Kalidaikurichi Lakshmanan[1], Frank Merat[1]

[1]Department of Electrical Engineering and Computer Science,

Case Western Reserve University, Cleveland, OH, Email: sxk1195@case.edu

ABSTRACT

The locomotion of earthworms – a legless species – occurs through a series of longitudinal contraction and elongation of parts of their hydro-skeleton – known as 'peristaltic' motion. There is significant interest in constructing bio-inspired legless robots, capable of navigation through peristalsis. To effectively bio-mimic earthworm motion, it is instructive to track several features of a worm's topography as it moves. In this project, we use morphological, registration and segmentation techniques to track the worm boundary, total skeletal length, total area, head and tail points over a complete peristalsis cycle. Head and tail points are identified using integral image analysis and template matching normalized cross-correlation respectively. As the peristalsis wave travels down the length of the earthworm, the propagation front elongates (driving the worm forward), identified by an increase in the over-all area and skeletal length. The average displacement of the head is higher than the tail over the time frame studied. Using SURF based image registration and cross-correlation, the clitellum is identified and its centroid tracked as a function of time. Finally, we propose a method to identify and track individual elongated segments of the worm. Segment identification is performed using adaptive thresholding and watershed transform techniques. This project establishes a proof-of-concept on how image processing can be utilized to parameterize several features of earthworm locomotion in relation to its dimensions. Quantitatively understanding this unique locomotion will be beneficial in designing robots that can replicate peristalsis to access spaces with minimal egress. All our experiments have been written and tested in MATLAB.

KEYWORDS

Earthworm, Peristaltic motion, tracking, SURF, Integral Image.

INTRODUCTION

Earthworms, ubiquitously found in soils, are flexible, cylindrical, segmented species that exhibit motion despite being legless. A sliding motion coupled with an axial elongation and contraction of their individual segments – called 'peristaltic' motion, characterizes their movement. The individual segments of the worm can undergo axial extension (expansion step) as well as radial expansion (contraction step). Unlike other legless species such as snakes which have a vertebrae, earthworms not have a rigid exoskeleton, but instead possess a flexible hydrostatic, fluid filled skeleton that allows for peristaltic waves to travel down the worm's body that enable propulsion. The ability to exhibit this contractive-expansive motion enables the worms to navigate through tortuous paths, bend over sharp angles and pass through tiny spaces. Therefore, researchers have taken inspiration from this organism to develop legless robots capable of exhibiting peristaltic motion. Proposed applications for these robots are in non-invasive intestinal tract surgeries [1], robotic endoscopies, troubleshooting industrial pipes [2] carrying hazardous chemicals, navigating through rubble for search operations etc.,

To mimic earthworm locomotion, we need to understand the dynamics of the individual segments of the worm as it moves. Specifically, research groups are interested in identifying the coupling between segment length-diameter-area as a function of peristalsis wavelength. In this project, we demonstrate that image processing techniques can be used to track several features of an earthworm as it exhibits peristalsis.

Biological researchers have utilized highly efficient algorithms to track the motion of *c. elegans* worms [3] –a microscopic species that does not have a segmented body. Furthermore, a *c. elegans* is more symmetrical along its skeleton at any given point. On the other hand, an earthworm presents several complications. Firstly, it has a complicate body consisting of segments of varying dimensions. Secondly, boundaries between segments are often not clearly identifiable. Third, as a earthworm propagates, it has an anisotropic surface – that is – part of its body (towards the propagation front) consist of highly elongated segments, a portion of its body is highly contracted, consisting of radially expanded compacted segments, while another portion remains in an undeformed state and in the midst is an irregular rectangular mid-section called the clitellum. Therefore, effectively tracking an earthworm is a significant challenge. Finally, while *c. elegans* are often captured using light microscopes, which have well defined intensity profiles and depth of field focusing, earthworm videos often suffer from non-uniform background illumination, varying focus that render global analysis infeasible.

The peristaltic motion of the worm has been studied using geometric methods as in [4] – by way of identifying centroids along the length of the worm and utilizing inscribed circles to represent effective cross-sectional width. Second, image processing techniques such as thresholding-segmentation-representation can be used to uniquely identify the worm features and track them explicitly. The latter is contingent to the quality of the video/images such as lighting, frame rate etc.

In this project, we first convert the individual frames into binary image using basic morphological operations and global thresholding methods. Using morphological operations again, we get the boundary, outline and the skeleton images of the frame. To track the worm over individual frames we extract important points of interest from the worm such as the centroid, corner points and the terminal (head and tail) points. In addition, we use SURF feature based registration method to identify the clitellum – a non-segment unit that is present mid-way along the worm's length. We found an interesting method to extract the head point using the integral image concept. Finally, we demonstrate how watershed segmentation coupled with image representation can be used to identify the worm segments. We distinctly identify several elongated segments of the worm closer to its head. Further refining of these methods would yield identifying more resolvable segments and parametrizing features such as radius of curvature as a function of worm length etc., Additionally, combining these image processing techniques with well-known geometric features of the worm – such as – constraints on the dimension of each segment, limits on number of segments in a contracted portion etc., are instructive next steps. In the following sections, we talk about the basic image processing techniques used in the paper and then discuss the results of the experiments.

Structure of the Earthworm

Figure 1 is an image of an earthworm that features the distinct portions of its body. The earthworm is cylindrical, tube like and highly flexible [5]. Throughout its length, the earthworm is divided into segments with thin grooves separating adjacent segments. Each segment can elongate and compress enabling locomotion. The clitellum (marked by a black bounding box) - the non-segmented band like unit - is a part of the reproductive system of the earthworm and generally separates the front and back part of the worm. During locomotion, a peristalsis wave travels from the head to the tail causing an elongation of the segments closer to the head following by a contraction of those segments that simultaneously drags the rest of the body forward. The earthworm has other dorsal pores and a mouth that are not visible in the figure. Another distinctly visible feature is the ventral blood vessel line that runs through the worm's length and is referred to as the 'skeletal length' in this project.

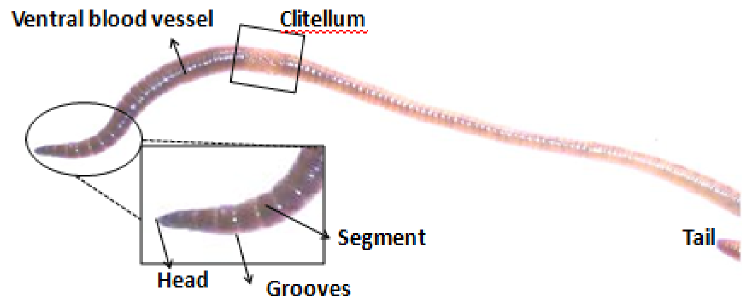

Figure 1: Image of an earthworm showing the distinct features along its body.

Image Morphology

Oftentimes, the properties of the objects in the image are not directly extractable and a lot of preprocessing is required for us to be able to realize these underlying properties such as shape, color and texture. The preliminary step to extract this information is to process the image so that it can be further processed to analyze how these objects can be best represented and described. Mathematical morphology [6] is one tool that helps in extracting the useful components of the image by mathematical operations. A structural element is used to operate on the binary or grayscale image and produces an output image of the same size as that of the input image. The morphological methods used in this project are – tophat transform, image filling, boundary identification and skeletonization. We will now briefly explain some of these concepts here.

Images that have poor contrast balance have this issue of degraded isolation of the peaks and valleys. Tophat transform is very useful in extracting these small details in an image. This transform involves two kinds of operations, opening and closing. The opening operation is first performed with a structural element and the resulting image is then subtracted from the original image (closing). This can mathematically be expressed as,

$$I \hat{\circ} S = I - I \circ S$$

Where $I \hat{\circ} S$ is the opening operation

The openhat operation preserves sharp peaks and enhances the image. Performing this operation on the original image results in a filtered image with pixels that are smaller than the structural element and are brighter than the neighbors.

Image filling is used to change pixels with 0s to 1s until the boundary is reached. It is a flood-fill operation that fills the holes in the images. The mathematical expression for hole filling can be given as,

$$X_k = (X_{k-1} \oplus B) \cap A^c$$

Where A is the image and the holes in A are filled.

Skeletonization is a method that reduces the foreground pixels to a great extent but retains the connectivity of the pixels resulting in the skeletal form of the original image. It is also known as Medial Axis Transform (MAT). While there are many techniques to generate the skeleton of the image, this process here involves repeated thinning operations by eroding the pixels until no more pixels can be removed. The skeleton is a very useful shape and property descriptor.

The initial processing procedure can be summarized as below:

1. *Take the grayscale image and perform tophat transform*
2. *Threshold the sharpened transformed image and create a binary image (Adaptive Thresholding)*
3. *Remove all connected pixels that are fewer than P pixels*
4. *Four regions of interest of the size in the previous step are selected - top, bottom, left and right*
5. *Fill holes in the resulting image*
6. *Get the boundary by doing a morphological remove operation*
7. *Extract the skeleton of the image*

Integral Image

Integral image [7] also known as summed area table, is calculated as – the value of a pixel in location (x,y) is equal to the sum of all pixels to the left and above this pixel. Integral images are interesting mainly because they speed up the calculations to a great extent reducing the order of complexity and at the same time also serve as useful feature descriptors of a region. The integral image at location (x,y) can be given as,

$$ii\,(x,y) = \sum_{x' \le x,\, y' \le y} i(x',y'),$$

In general, using the integral image, any rectangular sum can be computed in four array references and this implies that the difference between two rectangular sums can be calculated in eight array references. As expressed in [3],

F

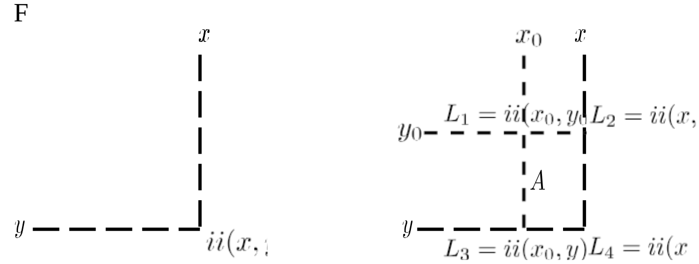

Figure 2 a(left) and right(b) : Integral Image computation

The value of ii(x,y) in fig. 2a is the obtained as described above while fig. 2b shows how to compute the sum of the sum of the sample values on the rectangle bounded by the rectangle. It can be computed as $L_4+L_1 – (L_2+L_3)$. In this project, we have used integral image to compute the head point of the worm. This was very interesting because computing the integral image made it very easy to determine the head point of the worm. The details of how we have used this concept follows in the results section below.

Speed Up Robust Features (SURF) based registration

It is important to identify features that are scale and rotation invariant and the SURF algorithm [8] identifies them given an image. Briefing the process of how SURF works from [3] by Bay et all, the first step is to identify the points of interest in the image using the determinant of Hessian matrix, use integral image technique that we discussed above to get these hessian matrices. It is evident now that dealing with integral images makes the computation less expensive. The matrix with thresholded determinants is called blob response map. Outliers are removed by using non-maxima suppression with the map itself and with the maps above and below it. To determine the direction of the features identified, haar transforms are used and finally features vectors are generated. In this project, we have used the SURF features for registering the template of clitellum with the incoming frames. Since the clitellum in the earthworm has quite distinct properties, identification is comparatively easier. Even as the worm undergoes motion, the clitellum is identifiable and this means that we could use the coordinates of the clitellum to track the worm in the video.

Normalized Cross Correlation (NCC)

The NCC algorithm matches two images, the reference and the template by performing a discrete cross correlation between the two images at every possible location in the reference image. NCC is a simpler method but is considerably effective compared to the other similarity measure methods (SAD, SSD etc.). The range of the correlation value is between -1 and +1 and this is because the correlation computation is equivalent to finding the dot product between two unit vectors,. Since it is normalized, NCC is less sensitive to the luminance variation.

General definition of the normalized cross correlation between two images f and g can be presented as,

$$\hat{f} = \frac{f - \bar{f}}{\sqrt{\sum (f - \bar{f})^2}} \qquad \hat{g} = \frac{g - \bar{g}}{\sqrt{\sum (g - \bar{g})^2}}$$

$$\boxed{\mathrm{NCC(f,g)} = C_{fg}(\hat{f}, \hat{g}) = \sum_{[i,j] \in R} \hat{f}(i,j)\hat{g}(i,j)}$$

Where C_{fg} the correlation is,

$$C_{fg} = \sum_{[i,j] \in R} f(i,j)g(i,j)$$

In this project, we have used the NCC algorithm to find the tail point in the image. Samples of the tail of the worm are saved as a binary image and this acts as a template image. We then look for this template in the original frame. Once it identifies the tail position, bounding box is marked and the coordinates of the bounding box define the tail portion. To identify the tail points that we wish to track, we look for the corner points detected by the Minimum Eigenvalue method and track only those corner points that lie within this identified tail region.

Adaptive Thresholding

In this project where tracking and segmentation are the two main goals, thresholding plays a very significant role in determining how accurate our model is. Thresholding is itself a form of segmentation. In this project, thresholding is used to convert the grayscale image to a binary image. The most common method is global thresholding using Otsu's method. However, as we will later see in the results section, global thresholding did not threshold the images correctly due to the variation in illumination across the frame. For this reason, we use adaptive thresholding. Adaptive thresholding takes into account the spatial variations in illumination. We use a moving average filter followed by a median filtering to threshold the given frame.

Watershed Segmentation

We utilize the watershed transforms to identify distinct segments in the earthworm. Figure 3 shows that adjacent segments of the earthworm are separated by a very thin boundary, which is often not easily visible due to poor contrast or variable lighting. Furthermore, in the contracted portion of the earthworm, the segment width is almost as wide as the boundary separating them. Therefore, segmenting neighboring segments is contingent upon effectively identifying the boundary separating them. While there are several tools to identify objects of interest from the background – such as thresholding, gradient smoothing, edge detection etc., - many of them rely on the intensity histogram having separable modal distributions. However, Fig. 3 precludes this possibility. In situations where the objects of interest are in close proximity to each other, or, are in contact with each other, the watershed transformation provides a means to draw 'ridge lines' or 'boundaries' separating the two objects.

Figure 3: Segments of the earthworm

The watershed segmentation follows the following algorithm. First, an intensity gradient or the 'intensity distant matrix' of the image is computed. The gradient image identifies boundaries where a significant change in intensity is present. The watershed transform then identifies local minima and climbs up the 'gradient' slope until a point of contact between neighboring local minima is obtained. At the point of contact, a 'node' or 'ridge line' is drawn. In this way, ridge lines are drawn between neighboring local minima thereby effectively segmenting the object from the background and from its adjacent neighbor.

In this report, we compute the intensity gradient matrix by simply computing a distant matrix which computes the distance from any pixel to the nearest non-zero pixel in a binary image. The watershed transform is then applied to the complement of the distant matrix which yields the ridge lines or boundaries between region of low and high intensities. From the watershed results, we compute set of all connect points (points whose neighbors have non-zero intensities) and non-connected background points (points which have atleast one zero neighbor) by using ridge lines which separate the two regions. Each region of connected points is a different segment, while the non-connected points are the background. To further improve watershed segmentation, we can implement advanced techniques for identifying the distant matrix. Some of these could be by using marker based segmentation, geometry and variable intensity based segmentation.

RESULTS AND DISCUSSION

In a video, the worm could be in any part of the frame and in varying orientations. Its locomotion is not stable and cannot be generalized to a specific pattern. Coming up with a common algorithm that would be invariant to the aforementioned changes requires detailed videography of discrete portion of the worm coupled with advanced image representation algorithms. Given the difficulty of the problem and the time constraints, we only consider a few scenarios to study the motion of the worm, but do provide substantial evidences that the methods experimented here, with few modifications, could be extended to challenging scenarios also.

Tracking

Shown in fig 4 are the tophat filtered (left) and adjusted (right) images of the worm in fig1 (right). As we can see from fig 1, the video has poor contrast balance, so it would be essential to perform sharpening and other contrast stretching techniques to equalize the different illumination based regions in the image. Figure 5 now is a comparison of the binary image thresholded using a Global thresholding method (Otsu's) and the same image thresholded using the adaptive thresholding method described above. As we can see from the image, with adaptive thresholding we are able to identify the segments much better than the global thresholding. With no contrast stretching, global thresholding is not able to identify the segments in the rear portion of the frame at all. As explained before, this is because of the fact that the image has different contrast in different regions and a locally adapted thresholding would be the best method on these kinds of images.

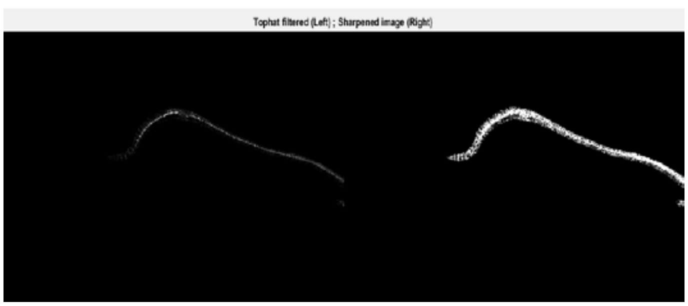

Figure 4: Tophat filtered image and binary image thresholded by Otsu's method

Figure 5: Global and local thresholding - comparison

The thresholded image is then filled for holes using the MATLAB function 'bwareaopen' and any region less than 50 pixels is treated as holes and they are filled with the pixel value 1. We then obtain the boundary line of the worm and the skeleton using the morphological operations performed above and fig.6 and 7 shows the result of these operations.

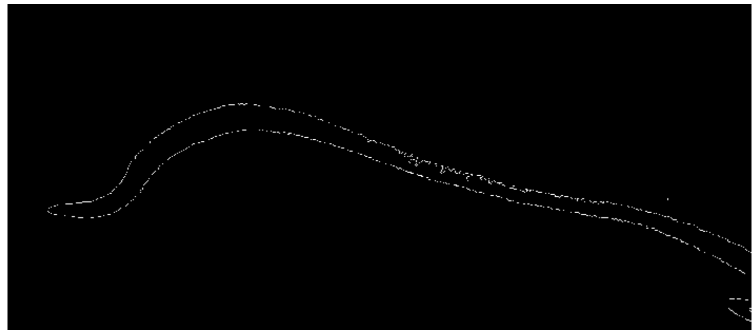

Figure 6: Boundary image

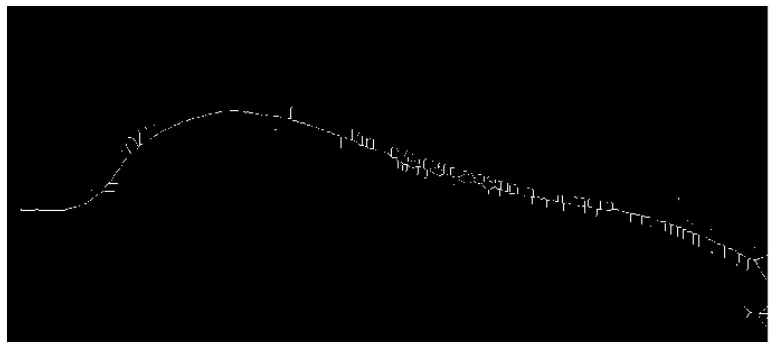

Figure 7: Skeleton image

For tracking the worm in the video, corner points are important, not all of them, but definitely the head and the tail. The first step to get the head and tail points is to get the set of corner points. Figure 8 shows the corner points obtained using the minimum Eigen value method.

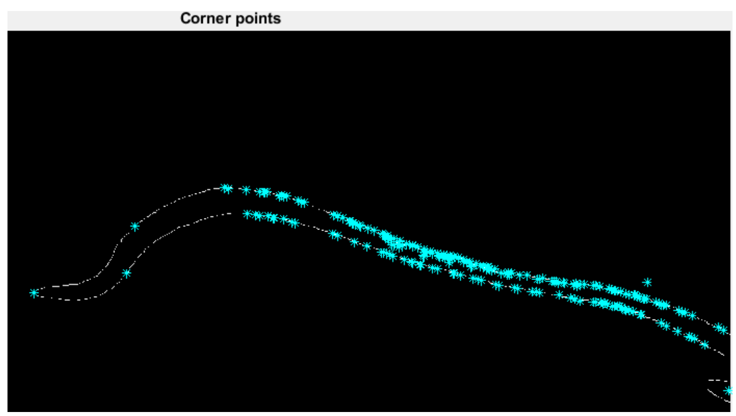

Figure 8: Corner points obtained using MinimumEigenValue Method

As we can see from the figure, the number of corner points is too high. Our points of interest, the head and tail points have to be extracted from these set of points. To do this, we use the integral image concept and the NCC method discussed above. First, to obtain the head point we propose a novel method of calculating the integral image of the frame.

Since the frame has only the worm and there is not much background noise, the integral image obtained is as shown in fig 9. This image is very interesting in that the head point is easily visible (highlighted in red). Here we assume that the worm moves from right to left (would also work for left to right movement by flipping the image) in the frame. As we calculate the integral image by summing up the values to the left of the frame, the head point becomes clearly visible. The head point can then be easily obtained by taking the coordinate positions of the first occurrence of 1. This is our head point. Figure 10 shows the integral image blended on the worm and we can see that the point marked is the head point. This gives us our head coordinates.

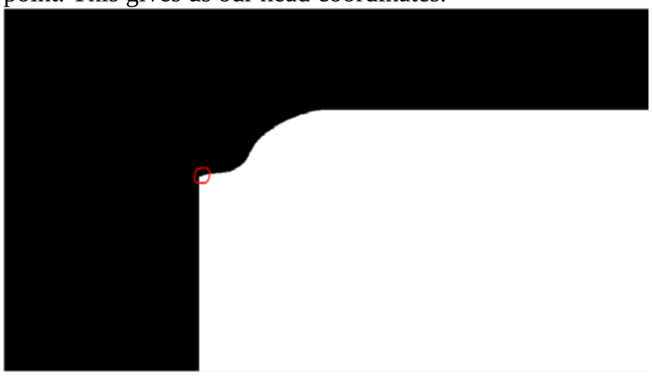

Figure 9: Integral image of the worm image

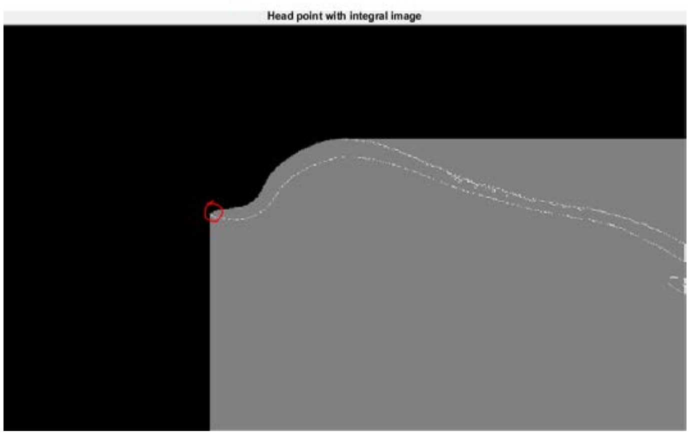

Figure 10: Head point identification; Integral image over the worm image

This works fine in cases where the head of the worm is the first portion on the frame. It might not work really well in cases where the body of the worm is ahead of the head. It would be interesting to try this proposed method in different cases.

Now, for the tail point NCC based matching technique is used. Before beginning the experiment, we collected a sample of the binary image of the tail portion as the template. Taking the highest correlation value the bounding box around the tail points is identified. The matching worked pretty well for all frames tested and we were able to obtain the corner points. However, this method has to be tested for robustness. Figure 11 shows the tail portion identified. The tail coordinate points to be tracked are then finalized by the same method as that calculated for the head point.

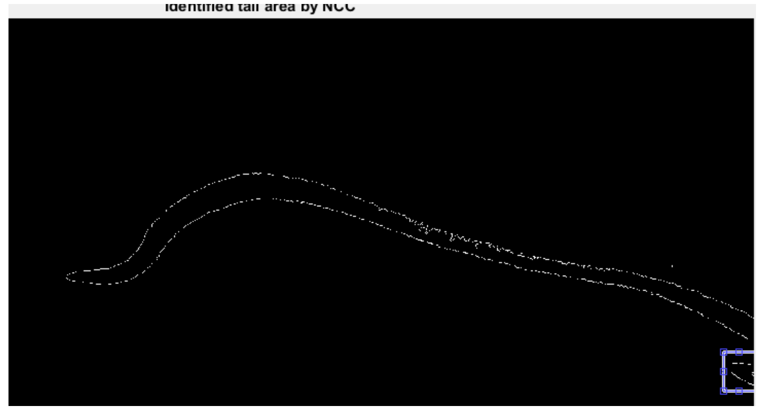

Figure 11: Identifying tail portion using NCC

The next step is to get the coordinates for tracking the clitellum. As discussed earlier, the clitellum is easy to identify and hence we use the SURF features based method, where we have a reference clitellum and match that in the frame to obtain a bounding box. Figure 12 shows the result of SURF based matching. Once the corner points in the bounding box are identified, we compute the centroid of the bounding box and that acts as the tracking point for the clitellum.

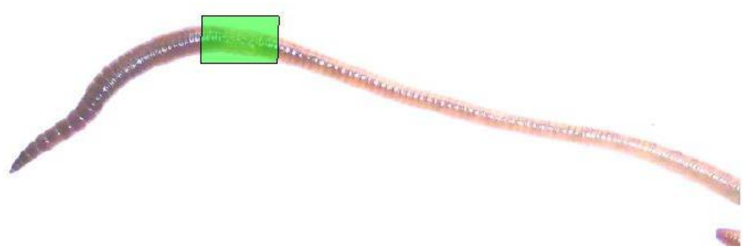

Figure 11: Identifying the clitellum using SURF

The centroid points of the image are also taken for tracking. We now have all the coordinate points to track and the final image with all the boundaries, skeleton and the points combined is shown in fig 12.

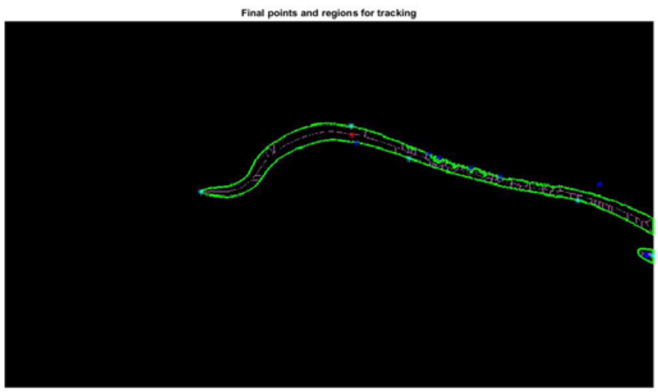

Figure 12: Image with all identified points and regions

Tracking can now be done in two ways – Performing all these operations on every incoming frame. When we tested this, we discovered that it takes approximately 3 seconds for a single frame to be processed. These experiments were tested on a Windows 7 Operating System with 8GB RAM and only CPU (No GPU). This indicates that a better way of tracking would be to use the Kalman filtering technique as in [11] or optical flow algorithm such as the KLT as in [12]. Given more time, we would definitely be interested in tracking the motion of the earthworm efficiently.

Individual segment identification

Identifying the individual segments is of significant importance for reasons discussed in the introduction section. We tried two methods here – simple segmentation followed by boundary identification using the matlab 'bwboundaries' and the other method of watershed segmentation and the latter proved to be promising. Figure 13 shows the result of the boundaries identified by the 'bwboundaries' function in MATLAB after segmenting the image. Though the segments have been correctly identified, this was possible because of the clear separation in the segments in this section of the worm due to elongation. In the same figure, we can see that the first few segments were not clearly segmented because they are closely connected.

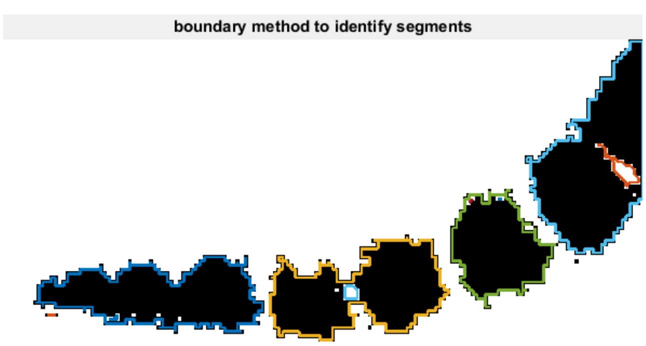

Figure 13: Failure identifying boundaries on the segmented image

The binary image generated using the adaptive thresholding approach discussed earlier is used for watershed segmentation. The result of the watershed segmentation performed can be seen in fig. 14. The individual segments have been identified which can be seen through the ridge lines that have been constructed along every segment.

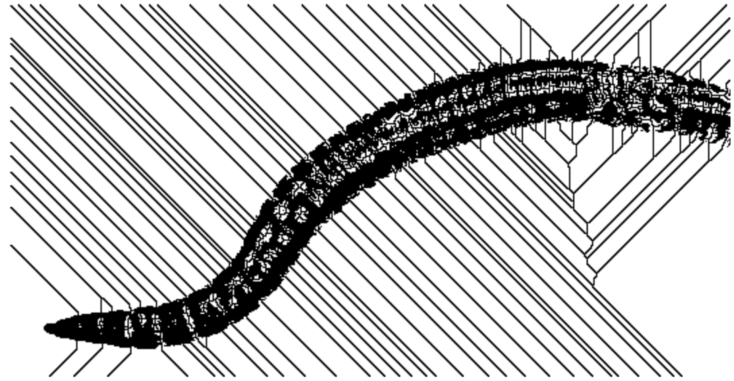

Figure 14: Watershed Segmentation

Figure 15 shows the result of the watershed segmentation on the entire worm. We can see that the worm has been segmented approximately correctly, though not entirely. More time and experimenting with this method might provide us better segmentation results which can then be used to study the interesting properties of the worm motion.

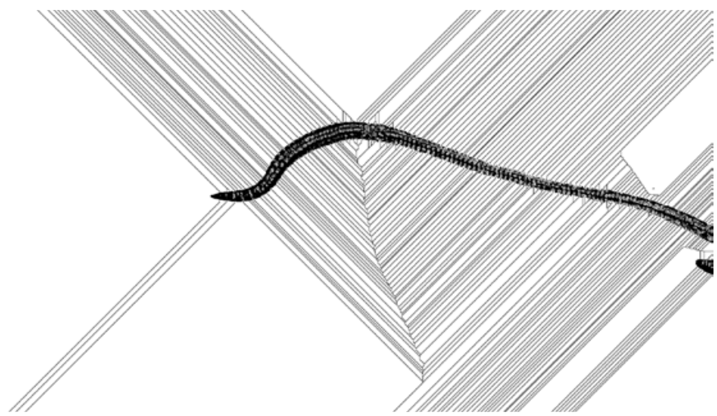

Figure 15: Watershed Segmentation on the entire image

Computing few basic properties with the data

The purpose of studying the motion of the earthworm is to construct robots that can bio-mimic this motion. Hence we were interested in studying some of the properties based on the data we have. These methods are not accurate since these methods themselves are not accurate, but they capture few basic characteristics well.

Figure 16 shows the plot of the change in area as a function of frames. From the graph, we can observe the pattern in the in the variation in area of the worm – during elongation phase, the area is found to increase while during contraction there is a decline in the area (Images corresponding to the plot obtained are attached in the zip folder). This result was obtained by taking every 25th frame with a total of about 14 frames. The reason for taking every 25th frame is to account for the change in motion. Since these videos have a frame rate of approximately 60 frames per second, motion would be better captured by picking selective frames.

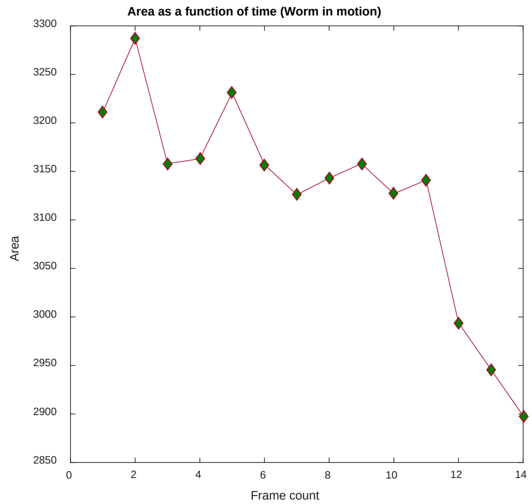

Figure 16: Change in area as a function of time (framecount)

The length of the worm is also an interesting characteristic to study and fig 17 shows the plot of length as a function of time (frames). The length is calculated by taking the skeletal image and computing the distance between the pixels with pixel value 1 (distance measured along the curve). We see a similar trend in the way the length varies – increases during elongation and decreases during contraction phase.

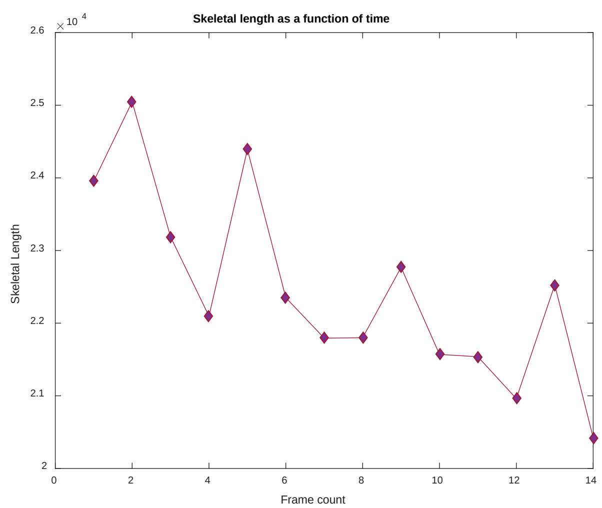

Figure 17: Skeletal length vs Time

Figure 18 plots the (x,y) position of the head and tail point over the frames tracked and shows an interesting property – We see that the head point moves significantly along the (x,y) plane while the tail point remains relatively stationary. This demonstrates the segmented motion of the earthworm - where - over any given time frame, only a portion of the worm moves while the remaining portions remains relatively stationary

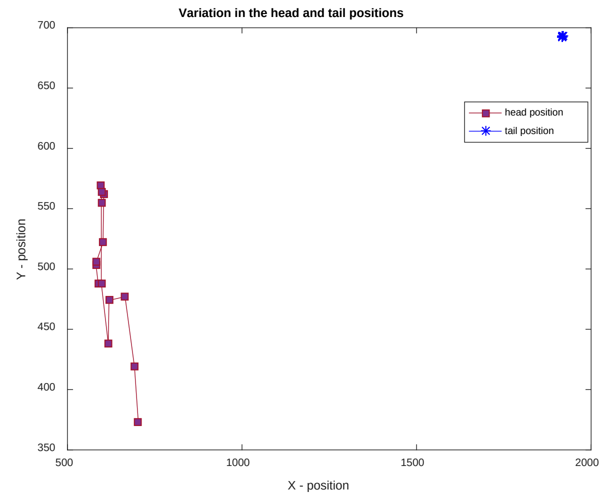

Figure 18: Head displacement vs tail displacement

SUMMARY

In this project we implemented a number of image processing algorithms studied in the course in tracking and segmenting the earthworm. This project was divided into two major subsections – Tracking the full worm and identifying the individual segments. For tracking, we identified the boundary, centroid points, skeleton line, head and tail points, centroid of the clitellum and computed them over individual frames. A novel method of identifying the head point using the integral image proved to be helpful. NCC and SURF based approaches for identifying the tail and the clitellum respectively was also successful. For segmentation, the simpler boundary based segmentation and the watershed algorithm were discussed. We also noticed the interesting patterns on the variation in the area, skeletal length, that they increase during elongation and decrease during contraction phase. The observation on the segmented motion with only the head moving with the tail remaining stationary is very interesting. There is still a lot more methods to experiment and lot more trials to be done before a robust and generalized model can be built for tracking and studying the locomotion of the earthworm. We would be very interested to continue work on this project and use image processing and computer vision to help understand yet another marvel of the nature – the peristaltic motion.

ACKNOWLEDGMENTS

This work was submitted as a final project for EECS 490, Prof. Francis Merat's class in Digital Image Processing. Prof. Merat was very supportive all through the course and the project. This topic is an ongoing research project in the biologically-inspired Robotics group at Case by Dr. Kathryn Daltorio and group. Dr. Daltorio provided great support and her initial guidance really helped in getting the idea of this project. The videos and images used for this project were provided by her and we would like to sincerely thank Dr. Daltorio.

REFERENCES

[1] Kassim I, Ng W S, Feng G, Phee S J, Dario P, Martin P and Mosse C A 2003 Review of locomotion techniques for robotic colonoscopy Proc.

[2] Yamashita A, Matsui K, Kawaniski R, Kaneko T, Murakami T, Omori H, Nakamura T and Asama H 2011 Self-localization and 3-D model construction of pipe by earthworm robot equipped with omni-directional rangefinder Proc. IEEE Int Conf. on Robotics and Biomimetics pp 1017–23

[3] Tracking the swimming motions of C. elegans worms with applications in aging studies, Restif C, Metaxas D

[4] Efficient worm-like locomotion: slip and control of soft-bodied peristaltic robots, Kathryn A Daltorio, Alexander S Boxerbaum, Andrew D Horchler, Kendrick M Shaw, Hillel J Chiel and Roger D Quinn

[5] Dorgan K M 2010 Environmental constraints on the mechanics of crawling and burrowing using hydrostatic skeletons Exp. Mech. 50 1373–81

[6] Digital Image Processing, 3rd Edition, Rafael C. Gonzalez, Richard E. Woods

[7] Paul Viola, Michael Jones, Rapid Object Detection using a Boosted Cascade of Simple Features, 2001

H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In European Conference on Computer Vision, May 2006. 1, 2