2019

Automated image analysis systems to quantify physical and behavioral attributes of biological entities

Upender Kalwa
Iowa State University

Automated image analysis systems to quantify physical and behavioral attributes of biological entities

by

Upender Kalwa

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Electrical Engineering (Bioengineering)

Program of Study Committee:
Santosh Pandey, Major Professor
Gregory Tylka
Meng Lu
Jaeyoun Kim
Long Que

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2019

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

ACKNOWLEDGMENTS

Firstly, I would like to express my sincere thanks to my advisor, Dr. Santosh Pandey, for providing his continuous guidance and support throughout my research. He has been a great mentor and I am grateful for his patience and consideration in coaching me along the way in my research and made me overcome my technical shortcomings. I am thankful for the opportunity he has given to me to pursue my research interests and for his useful career advice. I have learnt a lot from our collaborations and interactions with Dr. Gregory Tylka and his group members in Plant Pathology. I would also like to thank my committee members; Dr. Jaeyoun Kim, Dr. Long Que, Dr. Meng Lu, and Dr. Gregory Tylka for their guidance and insightful suggestions over the course of my research.

Secondly, I would like to thank my fellow graduate students, colleagues, technicians, and undergraduate students for their support and contributions. I would like to especially thank Dr. Taejoon Kong, Chris Legner, and Dr. Zach Njus for their help and support. I am also thankful for the collaborations with Jared Jensen, Elizabeth Wlezien, and Dr. Greg Tylka, which helped me gain a deeper understanding of soybean crop research. I am thankful to Dr. Michael Kimber for helping me understand the importance of human parasitic research.

Finally, I would like to extend my heartfelt gratitude to my family for their love and encouragement me all these years, providing me the confidence to become the person I am today. This academic success would not have been possible without them. This research was partially funded by Defense Threat Reduction Agency (HDTRA1-15-1-0053) and U.S. National Science Foundation (NSF IDBR-1556370).

ABSTRACT

All life forms in nature have physical and behavioral attributes which help them survive and thrive in their environment. Technologies, both within the areas of hardware systems and data processing algorithms, have been developed to extract relevant information about these attributes. Understanding the complex interplay of physical and behavioral attributes is proving important towards identifying the phenotypic traits displayed by organisms. This thesis attempts to leverage the unique advantages of portable/mobile hardware systems and data processing algorithms for applications in three areas of bioengineering: skin cancer diagnostics, plant parasitic nematology, and neglected tropical disease.

Chapter 1 discusses the challenges in developing image processing systems that meet the requirements of low cost, portability, high-throughput, and accuracy. The research motivation is inspired by these challenges within the areas of bioengineering that are still elusive to the technological advancements in hardware electronics and data processing algorithms. A literature review is provided on existing image analysis systems that highlight the limitations of current methods and provide scope for improvement.

Chapter 2 is related to the area of skin cancer diagnostics where a novel smartphone-based method is presented for the early detection of melanoma in the comfort of a home setting. A smartphone application is developed along with imaging accessories to capture images of skin lesions and classify them as benign or cancerous. Information is extracted about the physical attributes of a skin lesion such as asymmetry, border irregularity, number of colors, and diameter. Machine learning is employed to train the smartphone application using both dermoscopic and digital lesion images.

Chapter 3 is related to the area of plant parasitic nematology where automated methods are presented to provide the nematode egg count from soil samples. A new lensless imaging system is built to record holographic videos of soil particles flowing through microscale flow assays. Software algorithms are written to automatically identify the nematode eggs from low resolution holographic videos or images captured from a scanner. Deep learning algorithm was incorporated to improve the learning process and train the software model.

Chapter 4 is related to the area of neglected tropical diseases where new worm tracking systems have been developed to characterize the phenotypic traits of *Brugia malayi* adult male worms and their microfilaria. The worm tracking algorithm recognizes behavioral attributes of these parasites by extracting a number of features related to their movement and body posture. An imaging platform is optimized to capture high-resolution videos with appropriate field of view of *B. malayi.* The relevance of each behavioral feature was evaluated through drug screening using three common antifilarial compounds.

The abovementioned image analysis systems provide unique advantages to the current experimental methods. For example, the smartphone-based software application is a low-cost alternative to skin cancer diagnostics compared to standard dermoscopy available in skin clinics. The lensless imaging system is a low-cost and high-throughput alternative for obtaining egg count densities of plant parasitic nematodes compared with visual counting under a microscope by trained personnel. The *B. malayi* worm tracking system provides an alternative to available *C. elegans* tracking software with options to extract multiple parameters related to its body skeleton and posture.

CHAPTER 1.

INTRODUCTION

Research Motivation

In the late 16th century, the first microscope was developed and in the next century, Antonie van Leeuwenhoek made significant contributions using a simple lens microscope which led to the origin of microbiology. He was able to achieve 270 times magnification allowing him to view the world of microorganisms in a drop of water. Since then, this technology has been used in various fields to look at the detailed structure of cells, tissues, nematodes, lesions, among many other things. As technology improved, digital imaging became possible and was soon integrated with microscopes allowing people to record images and videos of biological specimens. The implementation of digital imaging technology allows scientists to harness the full potential of computational tools to analyze images. This allowed greater insights into the structure, biological, behavioral, and physical attributes of specimens. These computational tools, often referred to as image processing techniques, automate the interpretation of images by extracting physical attributes including size, shape, color, and texture. The quantification of these features has enabled object identification, anomaly detection, and point of care diagnostics. Within the realm of bioengineering, there are several areas of application where preference is still given to conventional microscopy because of standardization, convenience or availability. There is scope to introduce alternative imaging and software technologies to these areas of application where the hope is to improve accuracy, throughput, automation, cost or processing speed. In this thesis, we identified three such areas of application within bioengineering where new systems were developed to improve existing techniques with faster quantification of physical and behavioral traits.

The combination of digital imaging systems and image processing techniques has greatly enhanced research in several areas of medicine such as radioactive imaging, magnetic resonance imaging, x-ray and computed tomography. Compared to these areas, skin diagnostics has received less attention which still relies greatly on the use of dermoscopes and image processing algorithms to classify the cancerous nature of skin lesions. Existing literature suggests that early detection of skin cancer with frequent testing and adequate knowledge can significantly help improve chances of survival. Today, early or frequent skin testing for cancerous moles is generally limited by the lack of accessibility to skin clinics and the expensive costs of current dermoscopic instruments. Some at-home skin testing kits are available but are still expensive and limited studies have been conducted to judge their effectiveness in predicting the cancerous nature of moles. Here the research motivation was to develop a simple, smart, user-friendly method to test whether a skin mole is cancerous or not by using basic steps of recording a mole image, extracting its key physical attributes, and telling the user of the decision outcome.

Using standard microscopes, it is inherently difficult to automate the process of imaging multiple samples over a wide field of view. Various alternatives to microscopes for biological imaging came into existence, such as high-resolution scanners and lens-free imaging platforms. High-resolution scanners provide an option for imaging a wide field of view with some magnification during post-processing. Similarly, lens-free imaging platforms obviate the need for expensive lighting and objective lens with some image-processing subsequent to image capture. To bring out the benefits of both high-resolution scanners and lens-free imaging platforms compared to standard microscopes, we found an application area in plant nematology

where most of the egg counting was performed using a McMaster slide under a microscope. Conventional methods of counting eggs of plant parasitic nematodes in soil requires a number of steps to prepare the sample which are laborious and time-consuming. Thereafter, the visual detection of eggs is done by a trained person. During our discussions with Dr. Greg Tylka and his group, we concluded that new techniques for automating some of the steps in egg counting could be valuable for soil diagnostic laboratories. Here the research motivation was to develop new egg counting methods that obviate the need for standard microscopes while still processing a large number of soil samples in relatively shorter time.

Digital imaging with computational algorithms has been used to record the behavior of model organisms such as *C. elegans* and zebrafish. The progress in this field has led to linking genetic makeup to perceived phenotypes in model organisms and holds the promise to help understand the genetic basis of human diseases modelled in these microscale organisms. However, most worm tracking systems built for *C. elegans* cannot be applied for parasitic nematodes as they show a range of behavioral traits which cannot be grouped into identifiable features. For example, *Brugia malayi* is a mosquito-borne human parasite that is challenging to characterize in laboratory settings. This parasite has a low survival rate outside the human host and so not much is known about its behavior or response to stimuli. Tracking the body or posture of Brugia malayi is challenging as it exhibits a range of unpredictable body shapes. Here the research motivation was to develop a new digital imaging platform with software algorithms to record videos of *Brugia malayi* and identify key behavioral features towards predicting the dose response of various antifilarial compounds.

Literature Review

In bioengineering, image analysis systems have been gaining significant attention over the past few years, specifically in the context of portability, low-cost, high throughput, and signal processing. Image analysis systems often consists of an image acquisition hardware system to capture raw images and data analysis software to amplify the desired signal from background noise.

In general, the first major step in image analysis is to identify object(s) of interest from a recorded image or a video frame. The object is extracted from the image and a variety of physical and behavioral metrics such as size, shape, color, and positions are computed. These metrics are then used to characterize the behavior of biological specimen of interest. In this section, a review on the current image analysis systems and their shortcomings is presented in the three application areas of bioengineering: skin cancer diagnostics, plant parasitic nematology, and neglected tropical diseases.

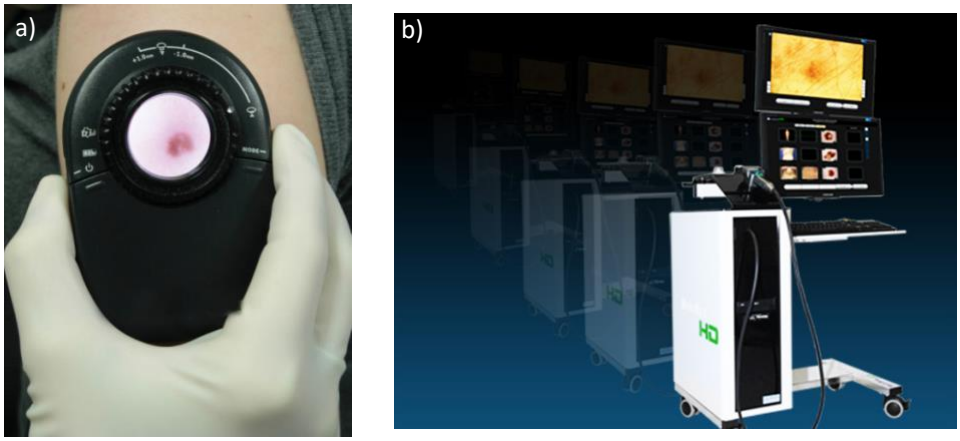


Figure 1.1 Diagnosing melanoma using dermoscope. a) Handheld dermoscope b) MoleMax HD [8,9].

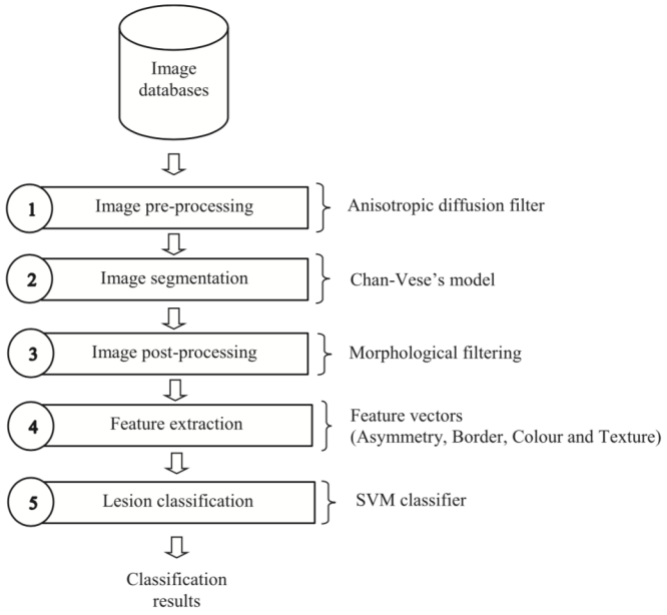


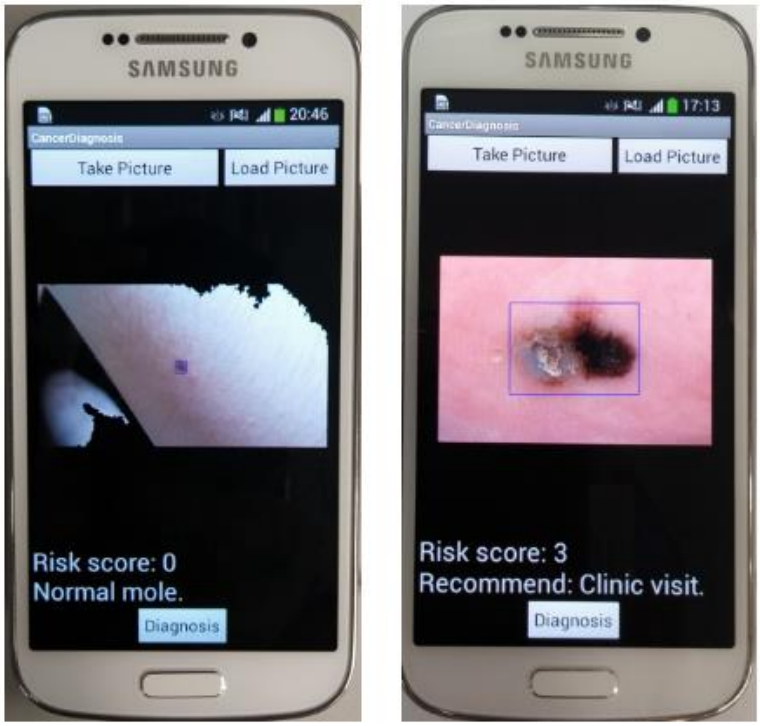
Figure 1.2 A typical flow chart of a computer-aided diagnostic system [10].

Skin cancer diagnostics

Among the different types of skin cancers, melanoma is the deadliest form of skin cancer with high mortality rates [1,2]. Literature shows that detecting at an earlier stage significantly improves the chances of survival [1,2]. Typically lesion images are captured by a commercial dermoscope which can be either handheld or desktop-based systems (Fig 1.1). Different computer-aided design (CAD) and diagnostic systems have been developed for the detection and analysis of these lesion images [3–7]. A typical CAD system for skin mole diagnostics consists of four steps: preprocessing, segmentation, feature extraction, and classification. In preprocessing step, smoothing filters are used to reduce the amount of noise and other artifacts present in the image. In the segmentation step, the lesion or mole is isolated from the background. In the feature extraction step, multiple features are extracted from the segmented lesion. The extracted features are fed to a mathematical model to get a score or to a machine learning classifier model to determine if a lesion is benign or melanoma. The

performance of the CAD system is evaluated by testing them on publicly available datasets of skin moles. The abovementioned steps remain the same for most CAD systems, while variations in the exact algorithms and procedure could be different.

A number of data processing algorithms exist to make the identification of skin moles more accurate and tuned to realistic situations. For example, in a previously reported work [10], the researchers have used an anisotropic diffusion filter to reduce the effects of hair present for the preprocessing stage and Chan-Vese's model as the segmentation algorithm. The authors have extracted asymmetry, border, color, and texture information from the images and ran them through a support vector machine (SVM) classifier to classify the lesion as shown in Fig 1.2. Similarly, for the preprocessing step, different filters such as Gaussian and median filters were used to reduce the background noise [16]. Different segmentation algorithms such as edge detection [11], thresholding [12], and active contour [10] methods were used to identify the mole structure from the image. For the feature extraction step, the feature characteristics such as asymmetry, border irregularity, color, texture, and diameter were extracted from the lesion images [10,13,14]. These were then passed through machine learning classifiers such as kNN [15], SVM [16], decision trees [15], and artificial neural networks [17] for classification.



(a) Non-cancer (b) Cancer

Figure 1.3 A screenshot of the smartphone application classifying a cancer and non-cancer sample [18].

While most of the CAD systems developed previously for melanoma detection run on desktop PCs or workstations, there have been a few attempts to use a smartphone for early melanoma detection [18,19]. Fig 1.3 shows a screenshot of a smartphone application for cancer diagnosis. Some of the limitations of these previously reported are testing the systems on small image data sets or different databases and not designed for at-home diagnostics which are addressed in this thesis.

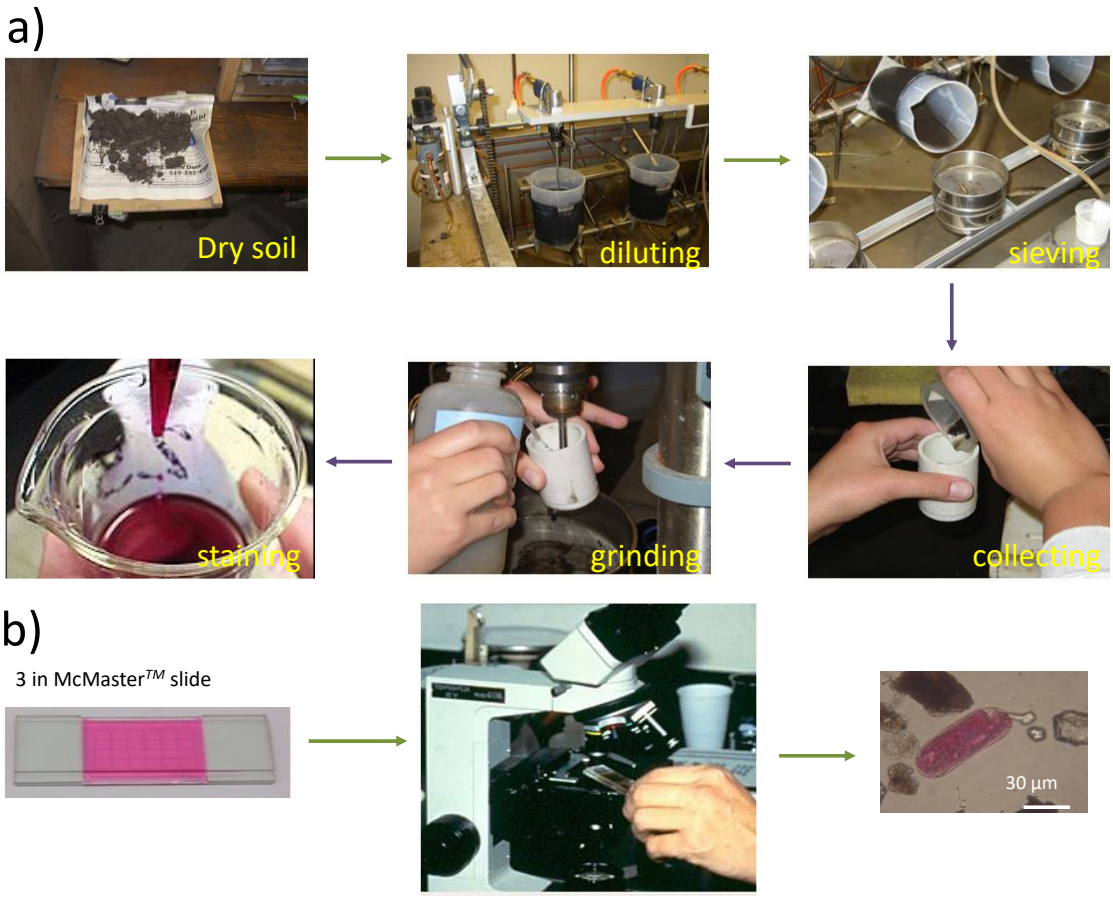


Figure 1.4 Procedure to count nematode eggs from a field sample. a) Protocol to extract and stain eggs from a dry soil sample. b) Manual method to count SCN eggs [23,24].

Plant parasitic nematology

The soybean cyst nematode (SCN), *Heterodera glycines*, is the major contributor to yield losses of soybeans in the United States. A common way to assess the severity of the infestation level is by computing the nematode egg density from a field soil sample [20,21]. For this purpose, the cysts (dead females) are collected from the sample. The eggs within are extracted and are stained using a dye. The protocol after obtaining the field sample to staining the nematode eggs is shown in Fig 1.4a. In the manual process, 1 mL of the stained egg solution is transferred to a nema slide. A trained technician counts the stained eggs on the slide by looking through a microscope as shown in Fig 1.4b.

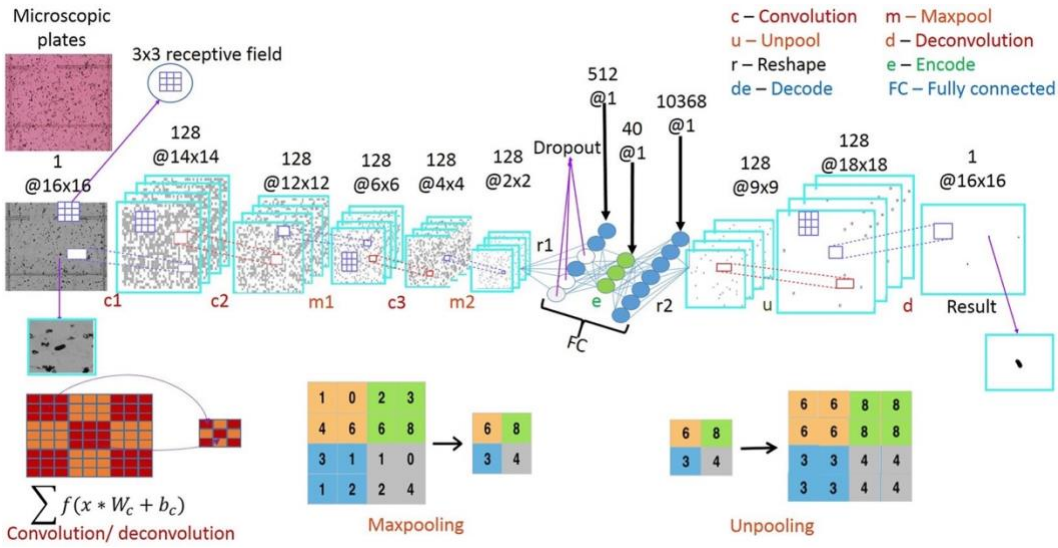


Figure 1.5 Deep convolution network to identify eggs in an image [22].

Since the manual method of egg counting is cumbersome and time consuming, researchers have automated the egg counting process using deep learning network [22]. In this method, initially the images of nema slide are captured through a microscope and a trained technician annotates the eggs. The annotated dataset is used in training an autoencoder deep learning network to identify and count eggs. The images are split into sub images and are fed through the network to produce a binary image with eggs if they are present as shown in Fig 1.5.

While the abovementioned methods efficiently count the eggs, there are some shortcomings. There are substantial manual hours with labor costs and only a small volume can be imaged at a time and then extrapolated to obtain total egg count. The microscopes are also expensive compared to scanners.

Neglected tropical diseases

Lymphatic filariasis, a neglected tropical disease, is caused by parasitic nematodes including *Brugia malayi* and over a billion people are at the risk of infection [25–28]. The parasites (larval stage) enter the human body when a mosquito carrying them takes a blood

meal. These parasites stay in the lymph vessels, mature into adults, and release microfilaria into the blood stream of the host. These microfilariae are ingested by other mosquitoes and transmit the disease. There have been worm tracker systems previously developed for drug screening purposes. Most of the worm tracker programs available today are developed to track *Caenorhabditis elegans*, a widely used model organism. One among them is the "Multi-worm tracker" which can record *C. elegans* crawling on the agar plate and can track their movement parameters in real-time [29]. The setup for this software is shown in Fig 1.6. The software calculates a variety of behavioral attributes such as: area, position, head position, spine length, spine curvature, speed, and angular speed. Using these attributes, the authors were able to distinguish different phenotypes of *C. elegans* [29].

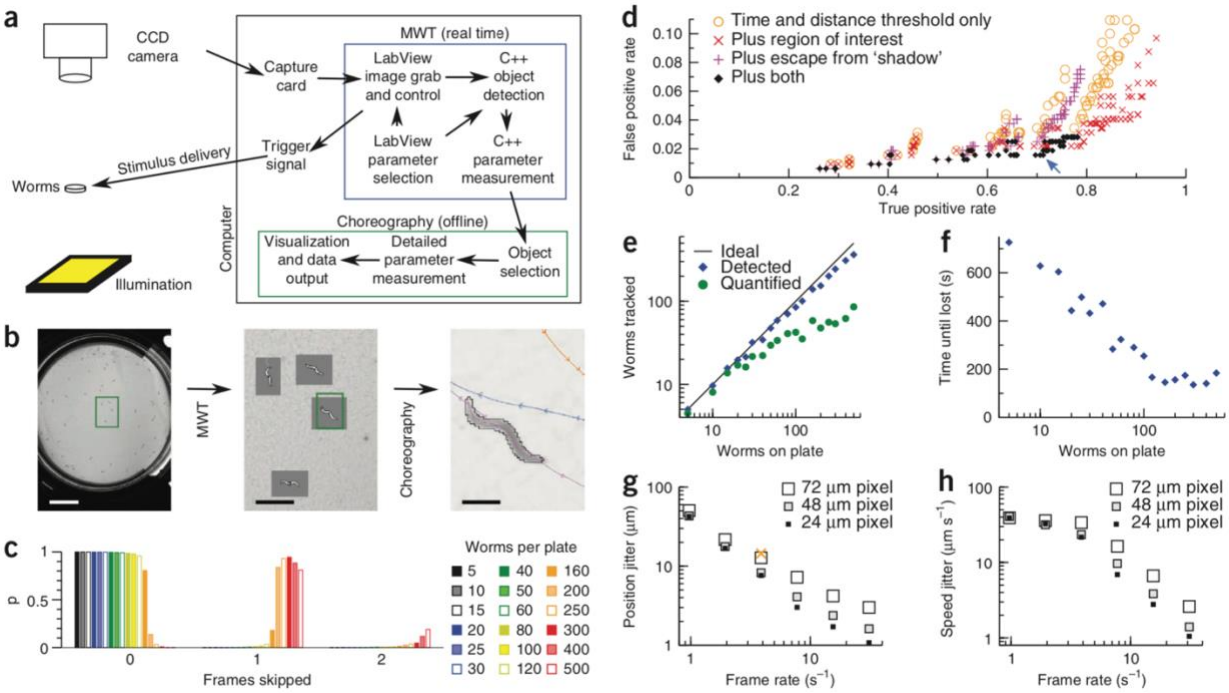


Figure 1.6 Setup and performance of the multi worm tracker software [29].

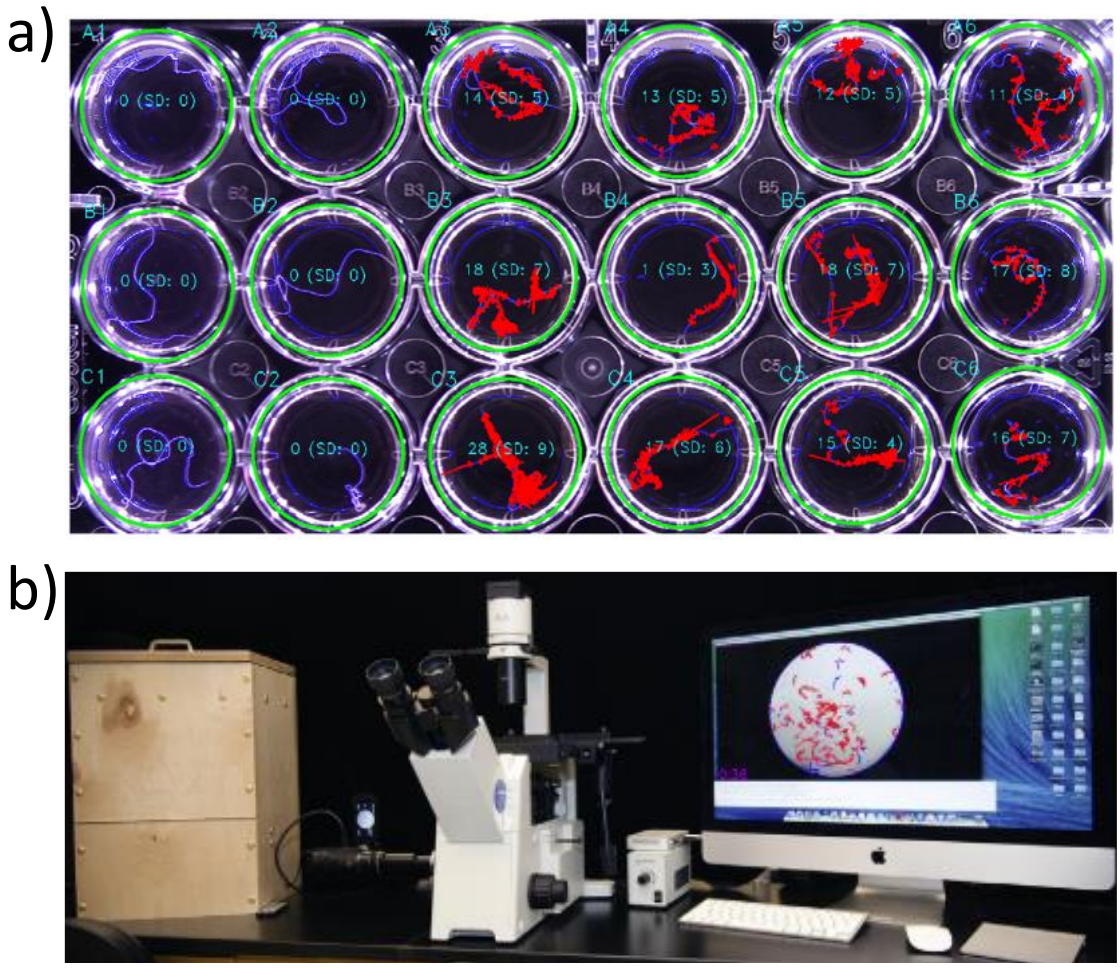


Figure 1.7 Setup and results of *B. malayi* worm tracker systems. a) WormAssay software results on female adult *B. malayi* b) Worminator setup for recording the videos of parasites [30,31].

In addition to *C. elegans*, some worm trackers for *B. malayi* have also been developed. One among them is known as "WormAssay" using which drug efficacy of three drugs was tested on female adult *B. malayi*. In their research, the authors developed an imaging platform to record videos of adult parasites after exposing them to drugs for more than 24 hours along with a custom software to analyze the videos. From these videos, metrics such as velocity and a custom metric to detect low levels of movement were extracted for each worm. The worm tracker software results are shown in Fig 1.7a. Using this system, the researchers were able to

achieve good dose response curves for the drugs [30]. Another system for drug screening of parasitic nematodes is called "Worminator" [31]. Researchers have used this system to screen drug compounds for microfilariae and L3 stages of three different filarial nematodes. The setup consisted of an inverted microscope, an x-y stage, a dark filed illuminator holder, and a computer running custom software as shown in Fig 1.7b. The videos of parasite movements were recorded after exposing them to drugs for more than 12 hours. The custom software calculated the motility inhibition percentage which was used to calculate the dose response curves for the different drug experiments.

While there are worm tracker systems developed for *B. malayi*, they extract only a couple of features for observing the phenotypical behavior which may not explain the phenotypical behavior as good as the *C. elegans* worm trackers. There are worm tracker systems developed for *C. elegans* that are more advanced and extract multiple features, however, they are designed to work with *C. elegans* as the assumptions of size, shape, and movement pattern are incorporated in their design which makes it challenging for these systems to be used for *B. malayi* parasites. In this thesis, we addressed these problems and developed software modules for both adult and microfilaria stages.

Dissertation Organization

This thesis presents automated imaging systems for applications in three areas of bioengineering which are still elusive to technical advancements in hardware electronics and data processing algorithms. The imaging systems helped us extract different sets of physical and behavioral features from the following biological entities: skin moles and lesions, eggs of soybean cyst nematodes, and the human parasite, *Brugia malayi*.

Chapter 2 deals with the development of a smartphone application to identify early signs of melanoma from skin lesion images. A user captured an image directly from a

smartphone with custom lens attachment and uploaded the image into the software application. The uploaded image was analyzed using multiple image processing steps to extract physical features such as asymmetry, border irregularity, number of colors, and diameter. A machine learning model was used to train the classifier in the application to classify an image as benign or malignant from the set of features. We validated the application by testing its performance on two public datasets which are dermoscopic and digital lesion images. The chapter is modified from journal paper "Skin Cancer Diagnostics with an All-Inclusive Smartphone Application", Symmetry 11.6(2019):790.

Chapter 3 discusses the development of new methods to clean debris and automated image analysis methods to count soybean cyst nematode eggs from soil samples. In the first image analysis method, we used a high-resolution scanner to scan the images of filter papers containing eggs and debris from the soil sample. These images were split into train and test datasets. A deep learning model was trained using the train dataset to classify and count the eggs from debris. We validated the model by comparing its results with manual methods. In the second method, we used a lensless imaging and a microfluidic flow chip setup to count the eggs. A soil sample was loaded into a syringe connected to the flow chip and a holographic video was recorded as the sample passed through the flow chip. The holographic videos were reconstructed and processed by an image analysis program that identified and counted the eggs based on its features such as size and shape. Its performance was compared with the scanner-based and manual methods.

Chapter 4 discusses the development of an image analysis software package to characterize the phenotypic behavior of *B. malayi* adult and microfilaria stages. The imaging software consists of two modules. The first module, BrugiaAdultTracker, calculated a number

of features from adult male videos recorded using a microscope. We explored the importance of these features individually by calculating the dose response of three commonly used drugs against adult male parasitic nematodes. The second module, BrugiaMFTracker, was developed to extract whole body information of *B. malayi* microfilaria by constructing a worm model. A number of features pertaining to posture and movement were extracted from the skeleton of the worm model. The imaging software provides a new method to track the behavior of nematodes that particularly do not follow the sinusoidal movement patterns of model *C. elegans.*

References

1. Giotis I, Molders N, Land S, Biehl M, Jonkman MF, Petkov N. MED-NODE: A computer-assisted melanoma diagnosis system using non-dermoscopic images. Expert Syst Appl. 2015;42: 6578–6585. doi:10.1016/j.eswa.2015.04.034

2. Abbas Q, Celebi ME, García IF. Hair removal methods: A comparative study for dermoscopy images. Biomed Signal Process Control. Elsevier Ltd; 2011;6: 395–404. doi:10.1016/j.bspc.2011.01.003

3. Yu L, Chen H, Dou Q, Qin J, Heng PA. Automated Melanoma Recognition in Dermoscopy Images via Very Deep Residual Networks. IEEE Trans Med Imaging. 2017;36: 994–1004. doi:10.1109/TMI.2016.2642839

4. Barata C, Ruela M, Francisco M, Mendonca T, Marques JS. Two systems for the detection of melanomas in dermoscopy images using texture and color features. IEEE Syst J. 2014;8: 965–979. doi:10.1109/JSYST.2013.2271540

5. Pennisi A, Bloisi DD, Nardi D, Giampetruzzi AR, Mondino C, Facchiano A. Skin lesion image segmentation using Delaunay Triangulation for melanoma detection. Comput Med Imaging Graph. Elsevier Ltd; 2016;52: 89–103. doi:10.1016/j.compmedimag.2016.05.002

6. Majtner T, Yildirim-Yayilgan S, Hardeberg JY. Efficient Melanoma Detection Using Texture-Based RSurf Features. Int Conf Image Anal Recognition Springer Int Publ. 2016;9730: 30–37. doi:10.1007/978-3-319-41501-7

7. Møllersen K, Kirchesch H, Zortea M, Schopf TR, Hindberg K, Godtliebsen F. Computer-Aided Decision Support for Melanoma Detection Applied on Melanocytic and Nonmelanocytic Skin Lesions: A Comparison of Two Systems Based on Automatic Analysis of Dermoscopic Images. Biomed Res Int. 2015;2015. doi:10.1155/2015/579282

8. Rigel DS, Russak J, Friedman R. The evolution of melanoma diagnosis: 25 years beyond the ABCDs. CA Cancer J Clin. 2010;60: 301–316. doi:10.3322/caac.20074

9. MoleMax HD [Internet]. Available: https://www.dermamedicalsystems.com/products_systems.htm#molemax_hd

10. Oliveira RB, Marranghello N, Pereira AS, Tavares JMRS. A computational approach for detecting pigmented skin lesions in macroscopic images. Expert Syst Appl. Elsevier Ltd; 2016;61: 53–63. doi:10.1016/j.eswa.2016.05.017

11. Barcelos CAZ, Pires VB. An automatic based nonlinear diffusion equations scheme for skin lesion segmentation. Appl Math Comput. Elsevier Inc.; 2009;215: 251–261. doi:10.1016/j.amc.2009.04.081

12. Ma Z, Tavares JMRS. A Novel Approach to Segment Skin Lesions in Dermoscopic Images Based on a Deformable Model. IEEE J Biomed Heal Informatics. 2016;20: 615–623. doi:10.1109/JBHI.2015.2390032

13. Jain S, Jagtap V, Pise N. Computer aided melanoma skin cancer detection using image processing. Procedia Comput Sci. Elsevier Masson SAS; 2015;48: 735–740. doi:10.1016/j.procs.2015.04.209

14. Jaworek-Korjakowska J. Novel method for border irregularity assessment in dermoscopic color images. Comput Math Methods Med. 2015; doi:10.1155/2015/496202

15. Doukas C, Stagkopoulos P, Kiranoudis CT, Maglogiannis I. Automated skin lesion assessment using mobile technologies and cloud platforms. Proc Annu Int Conf IEEE Eng Med Biol Soc EMBS. 2012; 2444–2447. doi:10.1109/EMBC.2012.6346458

16. Oliveira RB, Filho ME, Ma Z, Papa JP, Pereira AS, Tavares JMRS. Computational methods for the image segmentation of pigmented skin lesions: A review. Comput Methods Programs Biomed. Elsevier Ireland Ltd; 2016;131: 127–141. doi:10.1016/j.cmpb.2016.03.032

17. Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, et al. Dermatologist-level classification of skin cancer with deep neural networks. Nature. Nature Publishing Group; 2017;542: 115–118. doi:10.1038/nature21056

18. Do TT, Zhou Y, Zheng H, Cheung NM, Koh D. Early melanoma diagnosis with mobile imaging. Conf Proc . Annu Int Conf IEEE Eng Med Biol Soc IEEE Eng Med Biol Soc Annu Conf. 2014;2014: 6752–6757. doi:10.1109/EMBC.2014.6945178

19. Ramlakhan K, Shang Y. A mobile automated skin lesion classification system. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI. 2011. pp. 138–141. doi:10.1109/ICTAI.2011.29

20. Allen TW, Bradley CA, Sisson AJ, Byamukama E, Chilvers MI, Coker CM, et al. Soybean Yield Loss Estimates Due to Diseases in the United States and Ontario, Canada, from 2010 to 2014. Plant Heal Prog. 2017;18: 19–27. doi:10.1094/PHP-RS-16-0066

21. Akintayo A, Lee N, Chawla V, Mullaney M, Marett C, Singh A, et al. An end-to-end convolutional selective autoencoder approach to Soybean Cyst Nematode eggs detection. 22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining. 2016. doi:10.1145/1235

22. Akintayo A, Tylka GL, Singh AK, Ganapathysubramanian B, Singh A, Sarkar S. A deep learning framework to discern and count microscopic nematode eggs. Sci Rep. 2018;8: 9145. doi:10.1038/s41598-018-27272-w

23. SCN egg extraction protocol [Internet]. Available: https://www.plantpath.iastate.edu/scn/protocols_extracteggs

24. SCN egg counting [Internet]. Available: https://www.plantpath.iastate.edu/tylkalab/routine-soil-processing-scn-egg-counts

25. Bilsland E, Bean DM, Devaney E, Oliver SG. Yeast-Based High-Throughput Screens to Identify Novel Compounds Active against Brugia malayi. Geary TG, editor. PLoS Negl Trop Dis. 2016;10: e0004401. doi:10.1371/journal.pntd.0004401

26. Rao RU, Moussa H, Weil GJ. Brugia malayi: effects of antibacterial agents on larval viability and development in vitro. Exp Parasitol. 2002;101: 77–81. doi:10.1016/S0014-4894(02)00019-X

27. Kassis T, Skelton HM, Lu IM, Moorhead AR, Dixon JB. An Integrated In Vitro Imaging Platform for Characterizing Filarial Parasite Behavior within a Multicellular Microenvironment. Mitre E, editor. PLoS Negl Trop Dis. 2014;8: e3305. doi:10.1371/journal.pntd.0003305

28. Tompkins JB, Stitt LE, Ardelli BF. Brugia malayi: In vitro effects of ivermectin and moxidectin on adults and microfilariae. Exp Parasitol. Elsevier Inc.; 2010;124: 394–402. doi:10.1016/j.exppara.2009.12.003

29. Swierczek NA, Giles AC, Rankin CH, Kerr RA. High-throughput behavioral analysis in C. elegans. Nat Methods. 2011;8: 592–598. doi:10.1038/nmeth.1625

30. Marcellino C, Gut J, Lim KC, Singh R, McKerrow J, Sakanari J. WormAssay: A Novel Computer Application for Whole-Plate Motion-based Screening of Macroscopic Parasites. Geary TG, editor. PLoS Negl Trop Dis. 2012;6: e1494. doi:10.1371/journal.pntd.0001494

31. Storey B, Marcellino C, Miller M, Maclean M, Mostafa E, Howell S, et al. Utilization of computer processed high definition video imaging for measuring motility of microscopic nematode stages on a quantitative scale: "The Worminator." Int J Parasitol Drugs Drug Resist. Australian Society for Parasitology; 2014;4: 233–243. doi:10.1016/j.ijpddr.2014.08.003

CHAPTER 2.

SKIN CANCER DIAGNOSTICS WITH AN ALL-INCLUSIVE SMARTPHONE APPLICATION

Modified from the journal article
Upender Kalwa, Christopher Legner, Taejoon Kong, and Santosh Pandey. "Skin Cancer Diagnostics with an All-Inclusive Smartphone Application." Symmetry, 11(6), p.790, 2019

Abstract

Among the different types of skin cancer, melanoma is considered the deadliest and is difficult to treat at advanced stages. Detection of melanoma at earlier stages can lead to reduced mortality rates. Desktop based computer-aided systems have been developed to assist dermatologists with early diagnosis. However, there is significant interest in developing portable, at-home melanoma diagnostic systems which can assess the risk of cancerous skin lesions. Here we present a smartphone application that combines the image capture capabilities with preprocessing and segmentation to extract the ABCD features of a skin lesion. Using the feature sets, classification of malignancy is achieved through support vector machine classifiers. By using adaptive algorithms in the individual data processing stages, our approach is made computationally light, user friendly, and reliable in discriminating melanoma cases from benign ones. Images of skin lesions are either captured with the smartphone camera or imported from public datasets. The entire process from image capture to classification runs on an Android smartphone equipped with a detachable 10x lens and processes an image in less than a second. The overall performance metrics are evaluated on a public database of 200 images with Synthetic Minority Over-sampling Technique (SMOTE) (80% sensitivity, 90% specificity, 88% accuracy, and 0.85 AUC) and without SMOTE (55% sensitivity, 95% specificity, 90% accuracy, and 0.75 AUC). The evaluated performance metrics and computation times are comparable or better than previous methods. This all-inclusive

smartphone application is designed to be easy-to-download and easy-to-navigate for the end user, which is imperative for the eventual democratization of such medical diagnostic systems.

Introduction

Skin is the largest organ in the human body and comprises of two distinct layers: epidermis and dermis. While the epidermis protects the body from harsh exposures (such as ultraviolet radiation, infection, injuries, and water loss), the dermis provides nutrition and energy to the epidermis through a network of blood vessels [1–3]. As with every organ in the body, the skin is prone to different forms of cancer. The two most common skin cancers are the basal cell carcinoma and squamous cell carcinoma arising from epidermal cells called keratinocytes [4]. A third, deadlier form of skin cancer is malignant melanoma developing from epidermal cells called melanocytes. Today melanoma is notoriously frequent because of increasingly high rates of incidence that lead to a majority of skin cancer deaths [5,6].

To some extent, skin cancer is preventive and regular screening of skin moles, either in the clinic or at-home, is beneficial for curtailing the progress of the disease. However, current guidelines for screening skin cancer in the United States are inconsistent across different health organizations [7]. For instance, while the American Cancer Society recommends checking for skin cancer during periodic self-examinations by primary care physicians, the American Academy of Dermatology suggests that patients perform skin self-examination without sufficient clarity on the nature and frequency of screening. In a survey involving over 1600 physicians, it was concluded that the most effective skin cancer screening resulted when high-risk patients demanded a complete skin examination and the physicians also had sufficient medical training [7].

As with most cancers, early detection of melanoma can lead to reduced mortality according to several survey studies. In one study, 572 melanoma cases were detected over a

10-year timespan [7]. In another study, 18000 patients were checked for melanoma over a 17-year timespan. Both studies suggested that the chances of detecting melanoma early on are higher in established patients who routinely visit a skin clinic and are educated on the benefits of routine skin examinations [7]. Besides routine physical examination by a primary care physician or dermatologist, skin self-examination in at-home settings is valuable for the early diagnosis of melanoma. A thorough skin self-examination involves detailed diagnosis of all body parts including the back of the body and scalp areas. In addition, imaging technologies aid in accurate diagnosis at an early stage leading to better treatment and management strategies for melanoma.

The methods to evaluate skin growth for potential prognosis of melanoma have evolved over the past few decades. Before the 1980s, melanomas were generally identified by naked-eye observation of changes in gross mole features such as large size, bleeding, or ulceration [8]. In the case of suspicious lesions, biopsy of the lesion was done by removing the lesion for further analysis. This invasive method is still the most accurate method for diagnosis of melanoma, but requires the use of trained personnel and expensive equipment. During that time period, early prognosis was difficult because of the lack of technological advancements in imaging hardware and software tools. As time progressed, non-invasive techniques slowly became adopted that entailed less expensive equipment with good accuracy. The most commonly used non-invasive technique is dermoscopy or epiluminescence microscopy where the skin lesions are examined. A dermoscope is an optical instrument that uses a light source to cancel out skin surface reflections [1,9–12]. This gives access to the in-depth structures and colors of the lesion. This device is connected to a computer and captures images or videos of the lesion that are later used for diagnosis. A sensitivity of 89% has been reported using the

dermoscopy method; an improvement over the 70-85% reported for the naked-eye inspection approach [3]. However, the cost is prohibitive for generous usage as the dermoscope has to be coupled to expensive equipment (such as stereo microscopes) to evaluate the malignancy of the lesion.

Attempts to democratize skin diagnostics have been demonstrated that use cheaper alternatives to stereo microscopes as an imaging source. A method called "mobile teledermatology" employed mobile phones to take digital images of the lesion but needed coupling with pocket dermoscopic devices to compensate for the poor-quality optics in early generation mobile devices. The acquired images were transferred to teleconsultants via virtual private networks (VPN) located at remote locations for analysis and evaluation [13,14]. The two areas of improvement involve; (i) better hardware to capture high-resolution images and (ii) smarter computer-aided diagnosis (CAD) systems to accurately identify melanoma from dermoscopic images. Most of the previously reported CAD systems work on desktop PCs or workstations and assist the physicians to identify cancerous lesions at an early stage so that the treatment regimen can start right away. These CAD systems have generally been tested on dermoscopic or microscopic lesion images, even though they could be integrated with smartphones. Today mobile phones are equipped with high processing power, more storage capacity, high resolution image sensors, and larger memory [15,16]. This should enable mobile phones to capture images and run large computational tasks on the image directly on the device itself.

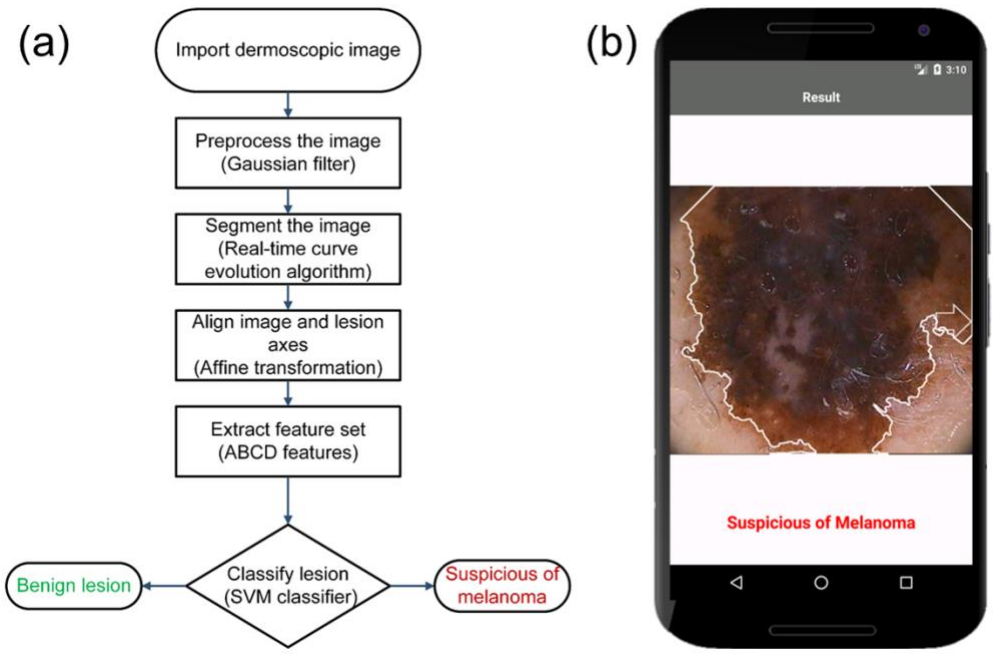


Figure 2.1 Overview of our smartphone application for the prognosis of melanoma. a) Flowchart shows the different steps involved in the processing of lesion images: The user selects an imported image on the smartphone and selects BEGIN to start the processing stages. The CAD system first preprocesses the image using a Gaussian filter and segments the image using the geometric curve evolution algorithm. The image is rotated to align with the major axes. ABCD features are extracted from the rotated image and classified as benign or melanoma using a SVM classifier. b) Screenshot of the final screen of the smartphone application running on the Android Operating System shows a sample lesion classified as melanoma.

In this work, we developed a smartphone application that functions as an image capture and diagnostic tool for at-home testing of skin cancer. The smartphone optics are enhanced by an inexpensive, commercially available 10x lens. A flowchart of the steps in the entire process is illustrated in Fig 2.1(a). A screenshot of the smartphone application classifying a lesion image from a public dataset is shown in Fig 2.1(b). Images of skin lesions are either captured with the smartphone camera or imported from the public dataset. Thereafter, four image processing steps are implemented: preprocessing, segmentation, feature extraction, and classification. A Gaussian filter reduced the noise, followed by a segmentation algorithm using curve evolution with fast level-set approximations to extract the lesion from the image. A linear

affine transformation aligned the lesion axes with the image axes. From the transformed image, the ABCD (Asymmetry, Border irregularity, Color variegation, and Diameter) rule is used to extract features and input them into a support vector machine (SVM) classifier to classify the lesion as benign or melanoma. The development of such all-inclusive skin diagnostic applications is anticipated to gain momentum in coming years, considering the present scenario of health care reforms, expensive costs of hospital visits, and the high mortality rates from melanoma.

Material and methods

Our smartphone application is designed to have relative ease of operation without compromising the accuracy in predicting melanoma cases. The application is intended to have a minimal number of intervention steps from the user with simplistic graphical representation of the classified results. Algorithms should be preferred that can run efficiently on a mobile phone without overloading the computing device. The four stages of image processing performed by the smartphone application are: preprocessing, segmentation, feature extraction, and classification.

Preprocessing

Typically, a dermoscopic image may contain artefacts such as hairs, ruler markings, air bubbles, and uneven illumination. The first stage towards classifying the malignancy of the skin mole involves preprocessing of the captured image where the intent is to remove the effects of the above-mentioned artefacts, reduce noise, and enhance the image contrast in the image.

In the preprocessing step, color transformation is first performed where the RGB image is converted to a different color space that separates color (chroma) from the intensity (luma) component. The commonly used color spaces are HSV (H represents hue, S represents

saturation, and V represents value in the cylindrical coordinate system), Y'UV (Y' represents luminance, U and V represent the chromaticity components in the Cartesian system) and Lab (L represents lightness, A and B represent the chromaticity components in the Cartesian system). We converted the image from the RGB color space to the Y'UV color space which separates the color component from the illumination component of the image, and is known to perform better in these respects than other color spaces such as HSV, HIS (H represents hue, S represents saturation, and I represents intensity in the cylindrical coordinate system), and LAB [17]. This conversion enables the smartphone application to perform consistently under varied illumination conditions (such as indoor and outdoor lighting) and makes the detection of lesion borders less prone to illumination effects. A RGB image is converted to the Y'UV color space based on the National Television System Committee (NTSC) standard using the following equations [18]:

$$Y' = 0.299\,\mathrm{R} + 0.587\,\mathrm{G} + 0.114\,\mathrm{B} \tag{2.1}$$

$$U = -0.147\,\mathrm{R} - 0.289\,\mathrm{G} + 0.436\,\mathrm{B} \tag{2.2}$$

$$V = 0.615\,\mathrm{R} - 0.515\,\mathrm{G} - 0.100\,\mathrm{B} \tag{2.3}$$

Next, to reduce the effects of artefacts, several filters are available that help smoothen the image. Commonly used filters are Gaussian [9,19–22], median [16,23,24], and anisotropic diffusion filters [25,26]. We used a 2-D Gaussian filter G for a point (x, y) represented by Eq. (2.4) below [27]:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\left(\frac{x^2+y^2}{2\sigma^2}\right)} \tag{2.4}$$

where σ_x and σ_y are the standard deviations and μ_x and μ_y are the means across both dimensions. The Gaussian filtering produces a resultant image by performing convolution of

the filter with the image. The size of the filter is determined by its kernel value. A large kernel value significantly blurs the image and weakens the borderline along with noise, whereas a small kernel value does not reduce the noise to a desirable extent. We found that a kernel of $k = 5$ and standard deviation of $\sigma = 1$ provided the best results.

Segmentation

In the segmentation step, the lesion boundary of the lesion is identified from the preprocessed image, which is then used to extract physical features of the lesion. A number of segmentation algorithms have been reported in the literature, such as edge detection [26,28,29], thresholding [30,31], and active contour methods. Previous segmentation algorithms were highly sensitive to noise and thus required high contrast images, in addition to inputs from the user to adjust the segmented region. Today, active contour algorithms have gained popularity, where a deformable curve progresses until it fits the boundary of the region of interest (ROI). Active contour algorithms are categorized as parametric or geometric based on the curve tracking method. In the parametric active contour model, the deformable curve is guided by energy forces with internal energy controlling the curve's expansion or shrinkage. The image energies (such as image intensities, gradients, edges, and corners) are used to guide the curve to the ROI. Although parametric models have worked even when the ROI has weak borders, there are challenges in handling ROIs with large curvatures and topological changes [10,23]. The geometric active contour model improves upon parametric models by adapting to topological changes. One popular geometric active contour model is known as the Chan-Vese model [32], [25]. Generally speaking, active contour models involve solving partial differential equations (PDE) for curve evolution creating a computational burden [33].

Our choice of segmentation algorithms was focused around geometric active contour models for the reasons mentioned above, but we desired techniques outside the PDE realm that

were computationally light and could run efficiently on a smartphone. We used a modified Chan-Vese model that runs in real- time with fast level-set approximation [34]. In this model, a curve $\phi(x, y)$ denotes a level set function over a grid u_o and is expressed by Eq. (2.6) below:

$$\phi(x,y) = \begin{cases} -3, & if\ (x,y)\ is\ an\ interior\ point \\ -1, if\ (x,y) \in L_{in} \\ 1, if\ (x,y) \in L_{out} \\ 3, & if\ (x,y)\ is\ an\ exterior\ point \end{cases} \quad (2.5)$$

where (x, y) represents a point location on the grid. The lists L_{in} and L_{out} contain points inside $(\phi < 0)$ and outside $(\phi > 0)$ of the curve separated by pixels and allow localization of the curve. The 'interior points' are the points inside L_{in} and the 'exterior points' are the points outside L_{out}. The model updates the curve during each evolution until it fits the boundary of the object of interest. The process of curve evolution is composed of a data dependent cycle and a smoothing cycle. Both cycles are repeated for N_a, N_s iterations for each evolution. In the data dependent cycle, a field speed F_d represented by Eq. (2.6) is calculated for all the points in L_{in} and L_{out}.

$$F_d(x,y) = \lambda_2(|u_0(x,y) - c_2|^2) - \lambda_1(|u_0(x,y) - c_1|^2) \quad (2.6)$$

The parameters λ_1 and λ_2 are fixed integer values, and $c_1$and c_2 are the mean values inside and outside the curve. The pixel intensity at a point (x, y) on the grid is given by $u_o(x, y)$. For each point in L_{out}, if $F_d > 0$, the point is switched from L_{out} to L_{in} and redundant points are deleted. Similarly, for each point in L_{in} , if $F_d < 0$, the point is switched from L_{in} to L_{out} and redundant points are deleted. In the smoothing cycle, a speed F_{int} represented by Eq. (2.7) is calculated for all the points in L_{in} and L_{out}.

$$F_{int}(x,y) = \begin{cases} 1, if\ G \otimes H(-\phi)(x,y) > \frac{1}{2}\ \forall\ x\ \in L_{out} \\ -1, if\ G \otimes H(-\phi)(x,y) > \frac{1}{2}\ \forall\ x\ \in L_{in} \\ 0, otherwise \end{cases} \quad (2.7)$$

where G and H represent the 2-D Gaussian filter and Heaviside functions, and $H(-\phi)$ indicates the object region of the curve (ϕ). For each point in the L_{out}, if $F_{int} > 0$, the point is switched from L_{out} to L_{in} and redundant points are deleted. Similarly, for each point in L_{in}, if $F_{int} < 0$, the point is switched from L_{in} to L_{out} and redundant points are deleted. The stopping condition of this segmentation algorithm is defined when the lists (L_{in}, L_{out}) are not updated after the first cycle or a set number of iterations is reached. In our case, $\lambda_1 = 2,\ \lambda_2 = 1$ were determined to be the best parameters after multiple trial and error tests.

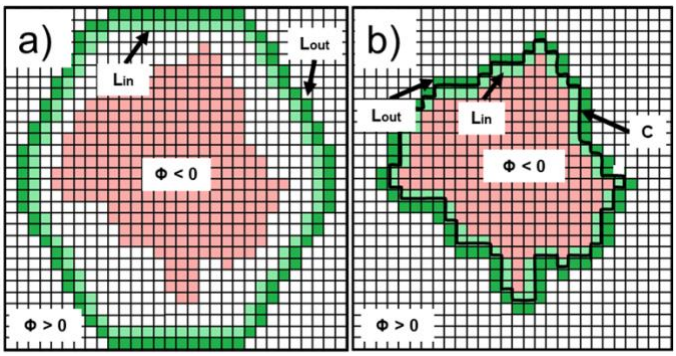


Figure 2.2 Illustration of the geometric active contour curve evolution process during segmentation to identify a lesion. a) The object of interest is drawn on a 2-D grid image. The initial elliptic curve with same height as the image and roughly 80% of the image width along with initial contours of L_{in} and L_{out} are overlaid on the image. The L_{in}, L_{out} contours represent the outer and inner curves with one-pixel gap. All the points in the image are assigned values based on the level-set function ϕ. At each iteration, the curves move inner towards the object determined by the speeds. b) The final contours of L_{in} and L_{out} along with the boundary C of the object after a certain number of iterations is shown.

A cartoon depicting the curve evolution from the initial set curve is shown in Fig 2.2. In Fig 2.2(a), the lesion image is shown as a 2-D grid with the object of interest depicted in light red. The initial lists (L_{in}, L_{out}) are shown by the light green and dark green colors, where

an initially segmented region, based on ϕ, is overlaid on the original image. During each iteration, the points in two lists get updated in the direction that minimizes the differences in mean values (c_1, c_2). After a certain number of iterations, the curve represented by C fits to the boundary as shown in Fig 2.2(b).

Feature extraction

In the 1980s, a group at New York University coined the ABCD acronym to categorize the morphological and clinical features of a skin mole or lesion [35]. This acronym stands for Asymmetry, Border irregularity, Color variegation, and Diameter greater than 6 mm [31,36–38]. The ABCD rule is best suited to differentiate early, thin tumors from benign, pigmented lesions. Besides the ABCD rule, there are other methods and algorithms to detect early melanoma. Pattern analysis has been employed with epifluorescence and video microscopy to categorize the type of skin lesion based upon its general appearance, surface, pigmented patterns, border, and depigmentation [39,40]. Pattern analysis and ABCD rule are the oldest and widely adopted methods for melanoma detection [41]. The C.A.S.H. algorithm identifies the Color, Architectural disorder, Symmetry, and Homogeneity/Heterogeneity of mole structures [42]. The C.A.S.H. algorithm has a lower specificity compared to the ABCD rule [8,43]. The Menzies method images the pigmented skin lesions using an immersion oil and categorized the mole based on the symmetry of pattern and one color [44]. The Glasgow 7-point checklist performs diagnosis on three major features (change in size of lesion, irregular pigmentation, and irregular border) and four minor features (inflammation, itching sensation, diameter greater than 7 mm, and oozing of lesions)[1,3,8,45,46]. Because of its inherent complexity, the Glasgow 7-point checklist is less widely adopted and has a lower pooled sensitivity compared to the ABCD rule [47]. Another method extends the ABCD rule to incorporate the evolution of the lesion (E parameter) by adding patient's description of

lesion change (e.g. enlargement, elevation and color change) [45,48]. Although the ABCDE rule has been validated in clinical practice, however no randomized clinical trials have shown that there is an improvement in the early detection of melanoma [45]. In addition, image acquisition methods have also been developed to differentiate the amount of light absorbed, transmitted or backscattered by the melanin content of the lesion. Examples of such image acquisition methods are hyperspectral imaging, reflectance confocal microscopy, and optical coherence tomography [43]. However, these image acquisition methods are yet to be standardized to accurately calibrate the absorbance or reflectance from the imaging window [43]. Information about the inflammatory process in the skin have also been retrieved by the use of ultrasound technology and electrical bioimpedance measurements. However, ultrasound images are difficult to interpret and the electrical impedance of the skin can vary greatly based on age, gender, and body location [8]. In addition, advanced dermoscopy and photography tools are commercially available for these applications (e.g. digital epiluminescence (dermoscopic) microscopy (DELM), SIAscope™, MIRROR DermaGraphix™ software, DigitalDerm MoleMapCD™) along with accessory camera (e.g. Molemax™, SolarScan™, VivaCam™). These equipment are cost prohibitive for at-home diagnostics [49].

In our smartphone application, lesion features are extracted following the ABCD rule. Because of its simplicity in implementation, the ABCD rule is widely adopted and taught in dermatology classes. Furthermore, among all the computerized methods for melanoma detection, the ABCD rule is the most popular and most effective algorithm for ruling out melanoma [47].

The ‘Asymmetry’ feature stems from the fact that lesion images taken using a dermoscope are generally not symmetric with the major x-y axes of the images. However, to

judge if there is any asymmetry in shape, the lesion axes must be aligned to the major axes of the image. To first accomplish this alignment, it is necessary to transform the segmented image (u_o) by finding the lesion's minimum enclosing rectangle and extracting the rectangular matrix from the image. This matrix provides the major and minor axes, along with the tilt angle (θ) of the rectangle. Next, we calculate the rotation matrix (M) from the tilt angle as shown in Eq. (2.9).

$$\alpha = sf\cos(\theta), \beta = sf\sin(\theta) \tag{2.8}$$

$$M = \begin{bmatrix} \alpha & \beta & (1-\alpha)C_x - \beta C_y \\ -\beta & \alpha & \beta C_x + (1-\alpha)C_y \end{bmatrix} \tag{2.9}$$

The parameters α and β are calculated from the scale factor and tilt angle as shown in Eq. (2.8). The scale factor (sf) determines whether the image should be cropped while rotating or scaling the image so that no information is lost. The application automatically adjusts the scale factor based on the area and position of the lesion in the image. The values (C_x, C_y) represent the centroid position of the lesion. From the rotation matrix (M), the segmented image is transformed to get the rotated image (R) as shown in Eq. (2.10) [2]:

$$R(x,y) = u_o(M_{11}x + M_{12}y + M_{13}, M_{21}x + M_{22}y + M_{23}) \tag{2.10}$$

where $M_{ij}\ \forall\ i = 1,2\ j = 1,2,3$ represents the corresponding value at the location (i,j) in the rotation matrix.

The asymmetry score is calculated from a total of eight parameters. The first two parameters, vertical and horizontal asymmetry, are calculated by overlapping the binary form of the warped segmented image with the mirror images in horizontal and vertical directions. The sum of all the non-zero pixels in the image is computed and divided by two, assuming that the asymmetrical area will be the same across horizontal and vertical axes. The asymmetry

level (AS) is calculated as the percentage of the non-zero pixels in the overlapped image over the lesion area and is represented by Eq. (2.11),

$$AS = \frac{NOR}{A} \times 100 \tag{2.11}$$

where *NOR* represents the non-overlapped region (non-zero pixels) and *A* represents the lesion area or the total sum of non-zero pixels in the binary image. The remaining six parameters refer to the asymmetry in structure and are calculated as the distance between lesion centroid and the weighted centroids of the color contours (obtained from the Color variegation feature).

The 'Border irregularity' feature is generally defined as the level of deviation from a perfect circle and measured by the irregularity index (I) as shown in Eq. (2.12),

$$I = \frac{P^2}{4\pi A} \tag{2.12}$$

where *P* and *A* are the perimeter and the area of the lesion, respectively [31,41]. The minimum value of the irregularity index is the one that corresponds to a perfect circle. As the lesion shape deviates from that of a perfect circle, the value of the irregularity index increases.

The 'Color variegation' feature denotes the different number of colors of the lesion from the HSV (Hue, Saturation, and Value) image. This is calculated by iterating through each pixel of the lesion, extracting its hue value, and grouping all the pixels that have hue values within a specified range. Our color set includes the following colors: white, red, light brown, dark brown, blue-gray, and black. The HSV values for these colors are determined by trial and error. In general, a benign mole has one or two colors while a melanocytic mole may have more than three colors.

The ‘Diameter greater than 6 mm’ feature refers to the size of a lesion in the suspicious case of melanoma. However, even with lesions having a diameter less than 6 mm, the mole should be analyzed for early risks of melanoma. The diameter of the lesion is calculated as shown in Eq. (2.13):

$$D = 2a\gamma \tag{2.13}$$

where a is the side length of the minimum area rectangle in pixels, and γ is the conversion factor from pixels to millimeters. The γ value is calculated using the parameters of the imaging system, such as focal length and the distance from object to the lens in the system.

Classification

The extracted features are passed to a classifier which categorizes whether the lesion is suspicious of melanoma or benign in nature. To generate an optimum classifier, the ABCD feature sets from all the images are randomly divided into training and test sets. Then, a supervised machine learning classification model learns to classify the lesions into different classes based on the input training set, this is generally referred to as the training step. The test set, excluding the classes, is applied on the generated model to create classes which are compared with the ones in the test set to evaluate the model’s performance.

The classification algorithm for our smartphone application prioritized robustness, supporting libraries developed for different platforms (Desktop, smartphones), with better performance and faster classification. We chose the SVM classification algorithm because it satisfied all the above conditions and is demonstrated to work better than other classifiers in other studies [51]. Besides SVM, different classifiers have been implemented for this purpose, including k-nearest neighbor (kNN) [51,52], decision trees [51], and artificial neural networks [53,54]. The SVM algorithm constructs a discrimination plane (hyperplane) in high

dimensional space that best separates the input data into different classes. The training data with m samples is labelled as $\{x_i, y_i\}$, where $x_i \in$ input features $(\boldsymbol{X})$, $y_i \in$ input classes $(\boldsymbol{y})$. Let the separating hyperplane constructed by x be defined as $f(x) = w \cdot x + b$ where w is the normal distance to the hyperplane, $|b| / ||w||$ is the perpendicular distance from the origin, and $||w||$ is the Euclidean norm of w. The hyperplanes can be formed by x that satisfy $f(x) = 0$ in such a way that the positive samples satisfy $f(x) > 0$ and negative samples satisfy $f(x) < 0$. Depending on the type of data, the following constraints can be formulated [55]:

$$y_i(f(x_i)) - 1 \geq 0 \; \forall \; i = 1, \dots, m \;\; \text{(Linearly separable)} \tag{2.14}$$

$$y_i(f(x_i)) - 1 \geq -\xi_i, \; \xi_i \geq 0 \; \forall \; i = 1, \dots, m \; \text{(Linearly non} - \text{separable)} \tag{2.15}$$

For the linearly non-separable data, the variables ξ_i referred to as slack variables are added such that $\sum_i \xi_i$ sets the upper bound on the total number of errors. An extra cost parameter C is added to assign penalty for errors. The algorithm chooses the optimum hyperplane based on the largest margin which is calculated as the sum of shortest distances from closest positive and negative sample to the hyperplane. The largest margin is obtained by forming two parallel hyperplanes $H1$ and $H2$. The points that satisfy $\alpha_i > 0$ are called support vectors. These parallel hyperplanes are obtained by minimizing the $||w||^2$ subjected to the inequality constraints (Eq. (2.14) or Eq. (2.15)) depending on the type of data. This minimization is defined by Lagrangian functions for different types of data [55]:

$$L_{ls} = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i y_i \alpha_j y_j x_i \cdot x_j, \qquad \alpha_i \geq 0 \tag{2.16}$$

$$L_{lns} = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i y_i \alpha_j y_j x_i \cdot x_j, \qquad 0 \leq \alpha_i \leq C \tag{2.17}$$

The equation (2.16) for linearly separable data (L_{ls}) satisfies the conditions $w = \sum_{i=1}^{m} \alpha_i y_i x_i$ and $\sum_{i=1}^{m} \alpha_i y_i = 0$, where $\alpha_i, i = 1, \ldots, m$ are positive Lagrange multipliers for each of constraints for linearly separable data. The equation (2.17) for linearly non-separable data (L_{lns}) is subjected to conditions $\sum_{i=1}^{m} \alpha_i y_i = 0, i = 1, \ldots, m$.

For non-linear data, first the samples are mapped to a high dimensional space (H) defined as $\phi: X \rightarrow H$. A kernel function defined as $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ is used to calculate the dot product of the samples in the higher dimension. The Lagrangian function is modified as [55]:

$$L_{nl} = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i y_i \alpha_j y_j K(x_i, x_j), \qquad 0 \leq \alpha_i \leq C \tag{2.18}$$

The equation (2.18) satisfies the conditions $\sum_{i=1}^{m} \alpha_i y_i = 0$, where $\alpha_i, i = 1, \ldots, m$. The kernel function can be set by the user. We tested our smartphone application with three different kernels: Linear, Radial basis function (RBF), and a third-degree polynomial function.

Results

We initially trained and tested our smartphone application on the publicly available PH2 database [56]. The database has been analyzed by expert dermatologists with added information such as segmented lesions, identified colors, and their clinical diagnosis. The database consists of 200 dermoscopic images (80 atypical nevi, 80 common nevi, and 40 melanomas) taken by a Mole Analyzer system with a 20x magnification. The RGB images are 8-bit with a resolution of 768 × 560 pixels. To test the images in the database, our smartphone application uses the computer vision library OpenCV for Java. The segmentation algorithm is

implemented in C++ and embedded into the smartphone application using Android NDK (native development kit).

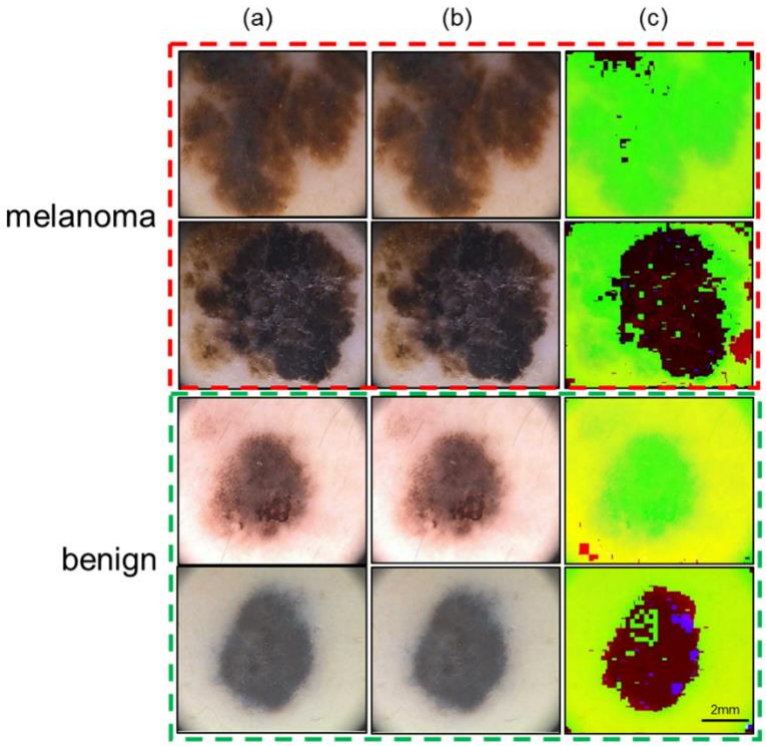


Figure 2.3 The preprocessing stage involves applying a Gaussian filter and color transformation. a) The column shows the representative images from the publicly available PH2 dataset. The top two rows are images for melanoma cases and bottom two rows are images for benign cases. b) The column shows the results after applying Gaussian filter with kernel value of 5 and standard deviation of 1. c) The column shows the results after converting the color space from RGB to YUV. Scale bar = 2 mm.

Representative images after the preprocessing stage are depicted in Fig 2.3. In Fig 2.3(a), the first two rows show images of melanoma cases and the last two rows show images of benign cases. We applied the Gaussian filter and tested the dataset with values of the kernel (k) ranging from 3 to 11, while maintaining a standard deviation (σ) value as 1. We found that $k = 5$ and $\sigma = 1$ gave the best results as shown in Fig 2.3(b). A color transformation from the RGB to YUV color space was performed on the images (Fig 2.3(c)).

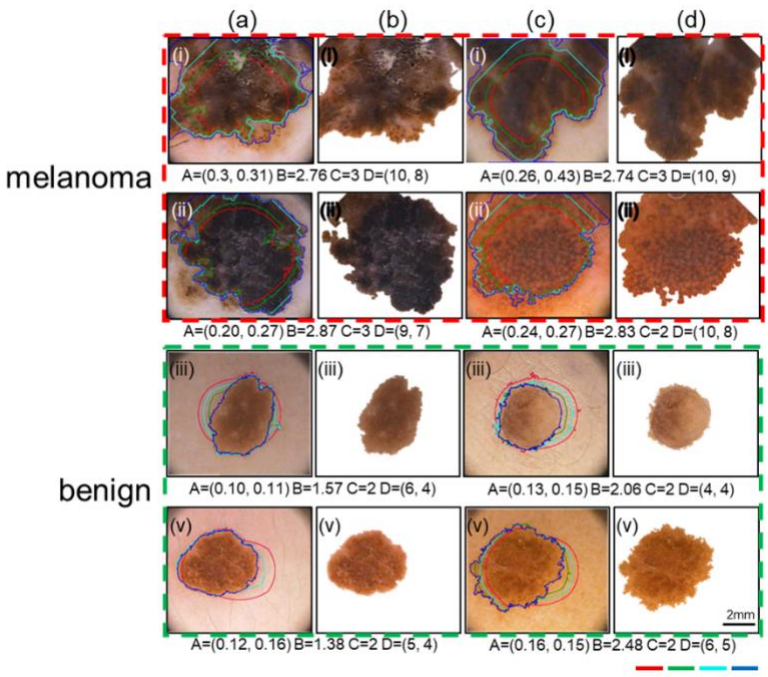


Figure 2.4 The segmentation stage identifies the lesion from the background using the geometric active contour algorithm. The top two rows are images for melanoma cases and bottom two rows are images for benign cases. Images in (a) and (c) columns show the original images overlaid with the resulting curve after evolution of 50 iterations (Red), 100 iterations (Green), 200 iterations (Cyan), and 400 iterations (Blue). The images in (b) and (d) columns show the corresponding final segmented images. The values of the ABCD features are listed for these images. Scale bar = 2 mm.

Thereafter, the procedure for segmentation is applied to the images in the dataset. The algorithmic parameters $\alpha, \beta,$ and γ are used to specify the importance of the Y, U, and V color channels with respect to segmentation and can be adjusted to add more weight to a specific channel, if needed. The initial contour was set at 65% of the image's width and height, and a maximum of 400 iterations were required to find the final contour. Visuals of the curve evolution during segmentation of representative benign and melanoma cases are shown in Fig 2.4. The first two rows refer to the melanoma cases and next two rows refer to the benign cases. The curve at different iteration points (50, 100, 200, and 400) is overlaid on the original image in different colors (red, green, cyan, and blue) as shown in Fig 2.4(a, c). It is interesting to

observe that the algorithm performs outward evolution for melanoma images and inward evolution for benign images as shown in Fig 2.4(a, c). The final curve is represented by the blue color. The segmented image based on this final curve for these images is shown in Fig 2.4(b, d). If the lambda values (λ_1, λ_2) are set to values (> 2) seems to be affecting the segmentation significantly. It is however worth noting that, in melanoma cases, the segmentation curve is slightly under fitted in some cases as shown in Fig 2.4(a, c). In addition, the ABCD values are shown for these images. From the figure, it can be inferred that these values are higher in the melanoma cases.

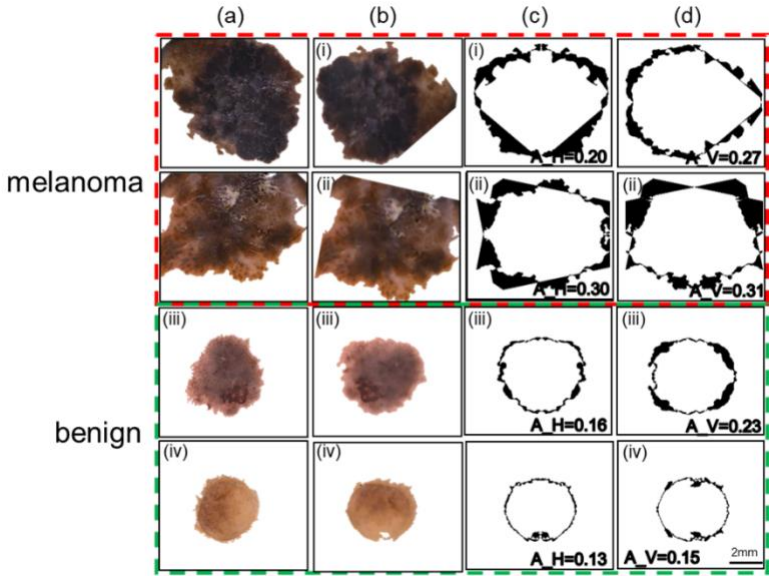


Figure 2.5 The asymmetry in shape is illustrated on representative lesion images. The top two rows are images for melanoma cases and bottom two rows are images for benign cases. (a) The column represents the images obtained after segmentation stage. b) The images are warped (rotated) so that the lesion axes are aligned with the image axes. c) Horizontal asymmetry is calculated by superimposing the horizontally flipped lesion onto the original lesion and marking the non-overlapped regions with black pixel values. d) Vertical asymmetry of the lesion is calculated by superimposing the vertically flipped lesion onto original lesion and marking the non-overlapped regions with black pixel values. The values of horizontal and vertical asymmetry (A_H and A_V, respectively) are listed within the images. Scale bar = 2 mm.

Our procedure for calculating the asymmetry in shape is depicted in Fig 2.5 for two representative benign and melanoma cases. The top two rows represent the melanoma cases while the next two rows show the benign cases. Fig 2.5(a, b) show the images after segmenting the lesions from the original images and rotating them to align with the image axes. Fig 2.5(c, d) show the results of the horizontal and vertical asymmetries, in shape, overlaid with their values. The vertical and horizontal asymmetry values, represented by darkened pixels, are higher for melanoma cases than for benign cases.

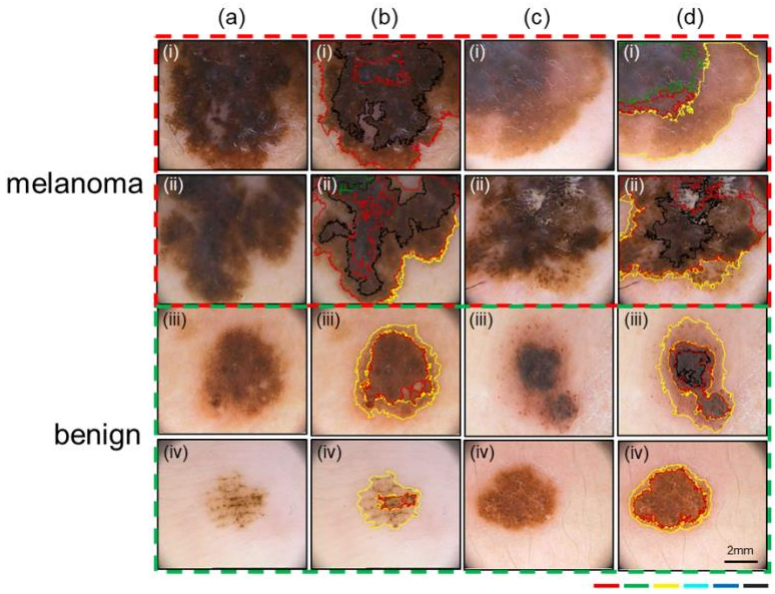


Figure 2.6 The color variegation feature is illustrated showing the different color contours on representative lesion images. The top two rows represent melanoma cases and bottom two rows represent benign cases. (a, c) The images shown here are the original images from the PH2 dataset. (b, d) The original images are overlaid with the different color region borders detected by the smartphone application. The colors red, green, yellow, cyan, blue, and black correspond to dark brown, blue gray, light brown, white, red, and black color parameters. Scale bar = 2 mm.

The method to estimate the number of colors (for color variegation) is applied to the dataset, and representative images (five melanoma and five benign cases) are shown in Fig 2.6(a, c). For most of the benign cases, the number of colors is limited to two (light brown and dark brown). However, in the case of melanoma, there are generally more than two colors

present. All six color parameters are labelled with different colors: dark brown (red), blue gray (green), light brown (yellow), white (cyan), red (blue), and black (black). The original images with color contours drawn for both melanoma and benign are illustrated in Fig 2.6(b, d).

Table 2.1 Evaluation parameters (i.e. sensitivity, specificity, accuracy, and area under curve (AUC)) were calculated on the PH2 dataset for three SVM kernels (i.e. Linear, Radial basis function, and Polynomial). The RBF kernel provides the best performance compared to the other two kernels.

SVM Kernel	**With SMOTE**				**Without SMOTE**			
	Sensitivity	Specificity	Accuracy	AUC	Sensitivity	Specificity	Accuracy	AUC
Linear	79	82	80	0.81	50	92	84	0.71
Radial basis function (RBF)	80	90	88	0.85	50	95	86	0.72
Polynomial	79	75	76	0.71	55	95	90	0.75

The ABCD features extracted from all the images are split into training and testing sets with a 70:30 ratio. Because of the low ratio of melanoma to benign cases in the dataset (1:4), we used a popular oversampling algorithm called Synthetic Minority Over-sampling Technique (SMOTE) [57] to synthetically generate more samples for melanoma to update the training set. The features in the training set are then scaled by subtracting the mean and dividing by the standard deviation for each feature independently maintaining zero mean and unit variance. This scaling is also applied to the testing set using the same mean and standard deviation values from the training set before testing against the classifiers. We evaluated three different kernels for the SVM classifier: Linear, Radial basis function (RBF), and Polynomial against evaluation parameters (sensitivity, specificity, accuracy, area under curve (AUC)) on the PH2 dataset. We plotted the receiver operating curves (ROC) and calculated the associated area under the curve (AUC) values. The evaluation metrics for the three kernels are shown in Table 2.1 for the cases of with and without SMOTE. With SMOTE, the RBF kernel performed better than the other two alternatives (linear and polynomial) for the four evaluation parameters

(80% sensitivity, 90% specificity, 88% accuracy, and 0.85 AUC). Without SMOTE, the polynomial kernel performed slightly better than the other two kernels for the evaluation parameters (55% sensitivity, 95% specificity, 90% accuracy, and 0.75 AUC). For studies on early melanoma detection, very high sensitivity is desired. Our calculations indicate that the sensitivity and AUC values with SMOTE were better than those without SMOTE, and the RBF kernel provided the best results with SMOTE.

Table 2.2 For each ABCD feature, the evaluation parameters (i.e. sensitivity, specificity, accuracy, and precision) were calculated and compared with results from combining all the features on the PH2 dataset.

Features	**Parameters**			
	Sensitivity	Specificity	Accuracy	Precision
Asymmetry	23	84	54	62
Border	81	63	72	68
Color	96	42	69	58
Diameter	90	71	80	75
Overall	80	90	88	91

The importance of each ABCD feature was evaluated by training the SVM classifier with the RBF kernel. The results are shown in Table 2.2. When only considering Color variegation, 96% of the melanoma cases were correctly identified, though many false positives were present. Similarly, based on asymmetry alone, 84% of the benign cases were correctly identified, but the increased frequency of false negatives lowered the accuracy and sensitivity values. The diameter feature seemed to have better performance when compared to others. A classifier combining all the ABCD features yielded improved precision (91%) and sensitivity (80%). The associated ROC curves for Table 2.2 and for multiple SVM kernel parameters are provided in the supplemental file.

For all images, a comparison of computational times (in milliseconds) was performed for each processing stage on a desktop (Intel Xeon-E5 CPU, 32 GB RAM, Windows 10) and an Android Phone (Samsung S6). On the desktop, a program was written that runs in Python and makes use of the OpenCV library for Python to perform the image analysis. To speed up the computation time, the same segmentation program used in the smartphone is wrapped using C++ bindings for Python. The results are shown in Table 2.3. Apart from the segmentation step, the remaining stages take similar computation times for both the benign and melanoma cases on a smartphone and desktop. In both devices, the segmentation step takes longer in melanoma cases (300-400 iterations) when compared to most benign cases (200-250 iterations). The segmentation would normally take longer on a smartphone than on a desktop, but due to the nature of the wrapper call and some additional steps required by the desktop program to extract the lesion boundary, the segmentation duration is longer on the desktop application.

Table 2.3 The computational time of each image processing stage in our smartphone application is listed as it is run on a Desktop (Intel Xeon-E5 CPU, 32 GB RAM, Windows 10) and Android Phone (Samsung S6) using the PH2 dataset.

Stage/Device	**Benign**		**Melanoma**	
	Desktop PC	Android Phone	Desktop PC	Android Phone
Preprocessing	78 ± 10	116 ± 18	68 ± 18	109 ± 6
Segmentation	283 ± 139	208 ± 106	415 ± 179	288 ± 92
Feature Extraction	27 ± 7	41 ± 13	31 ± 9	47 ± 14
Feature Classification	10 ± 2	19 ± 5	10 ± 2	19 ± 5
Total Time (ms)	398	384	524	463

For all images, a comparison of computational times (in milliseconds) was performed for each processing stage on a desktop (Intel Xeon-E5 CPU, 32 GB RAM, Windows 10) and

an Android Phone (Samsung S6). On the desktop, a program was written that runs in Python and makes use of the OpenCV library for Python to perform the image analysis. To speed up the computation time, the same segmentation program used in the smartphone is wrapped using C++ bindings for Python. The results are shown in Table 2.3. Apart from the segmentation step, the remaining stages take similar computation times for both the benign and melanoma cases on a smartphone and desktop. In both devices, the segmentation step takes longer in melanoma cases (300-400 iterations) when compared to most benign cases (200-250 iterations). The segmentation would normally take longer on a smartphone than on a desktop, but due to the nature of the wrapper call and some additional steps required by the desktop program to extract the lesion boundary, the segmentation duration is longer on the desktop application.

The computational time estimated for our smartphone application is lower than those previously reported. For instance, Andrea Pennisi *et.al.* reported that their system took 1.990 sec on an Intel i3-2370M CPU and 4 GB of RAM desktop system to classify an image as melanoma [28]. Aleem *et.al.* stated that their method took 14.938 sec on preprocessed 640 × 480 images using an Android smartphone [22]. Majtner *et.al.* reported that the computation time to extract the features alone was around 0.6 to 3.3 sec [19]. Oliveira *et.al.* noted that their CAD system took around 8 sec per image to classify the lesion on a desktop computer equipped with Intel i5-650 CPU with 8 GB of RAM [25]. Do *et.al.* reported that their smartphone application classified the lesion, taken by a Samsung Galaxy S4 Zoom smartphone, in less than 5 seconds after the image is resized with the longer edge at 512 pixels maintaining the same aspect ratio [16]. In comparison, our smartphone application takes less

than one second to completely process and classify an image of dimensions 768 × 560 pixels without compromising the accuracy in classification.

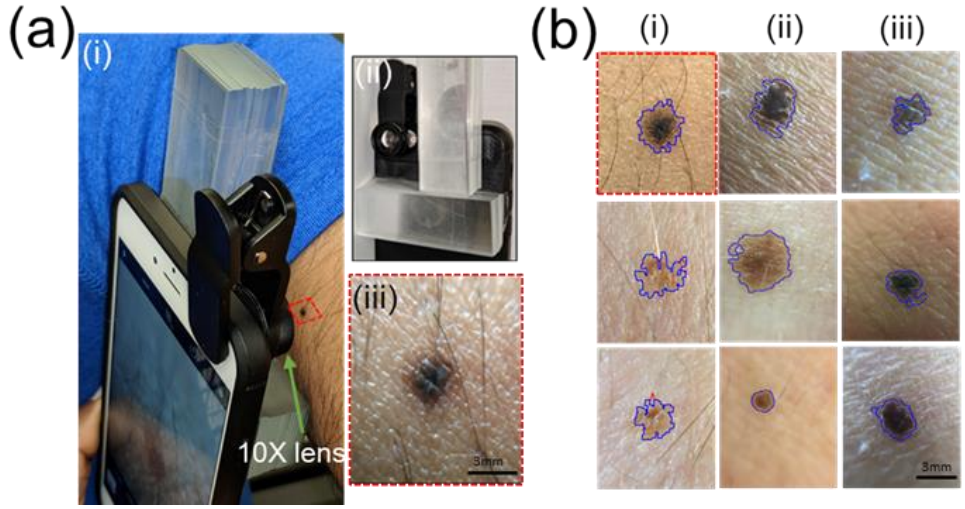


Figure 2.7 The smartphone application is used to capture, process, and classify live images of skin lesions. a) The picture shows the setup that includes a support structure (L = 75 mm, W = 75 mm, H = 13 mm) to capture a person's mole with our application (i). The glass support structure (ii) helps to align the camera onto the mole to capture the image at the appropriate focal distance (iii). A Macro 10x lens is attached to a smartphone rear camera and focused on a mole on the hand. Our application captures the image and processes it to determine the malignancy of the mole. b) Representative results for moles from eight volunteers are shown here overlaid with segmentation contours. All the moles were classified as benign here. Scale bar = 3 mm.

After training and testing the smartphone application on images in the PH2 database, we wanted to completely run the developed platform on live images captured using the phone's in-built hardware. The phone is attached with a 10x lens ($11, AMIR camera lens kit, Shenzhen Amier Technology) to allow us to take microscopic images of the skin moles. Eight individuals, each having different skin tones, volunteered to have images of their moles captured by a smartphone (Fig 2.7(a)). The operational procedure is as follows. First the 'Mole Detection' application is opened, the 'Begin' button is pressed providing the user with two features ('Take a photo' and 'Get stored pictures'). When the 'Take a Photo' feature is chosen,

the application takes control of the smartphone's camera allowing it to capture an image at the user's discretion. The user points the smartphone camera towards the lesion and captures an image after it is focused. This happens when the lens is roughly 13 mm away from the target lesion. This value is also used to recalculate the pixel to mm conversion factor and update it in the application. While capturing an image, a support structure allows to align the smartphone perpendicularly to the lesion at the optimal focal distance (Fig 2.7a(i)). An example L-shaped glass support structure (L = 75 mm, W = 75 mm, H = 13 mm) is shown in Fig 2.7a(ii) with a sample mole image taken using this support structure (Fig 2.7a(iii)). We have seen that focused images (focal distance = 13 mm) captured under white light or indirect sunlight are adequately processed by the smartphone application. Once the image is captured, it is resized to 1024 × 768 using bilinear interpolation before storing it. The user is then provided with two options ('Retake' or 'Proceed'). If the user is unsatisfied with the image quality, they can retake a new image by selecting the 'Retake' option. Otherwise, the user can select the 'Proceed' option to perform the diagnosis of the lesion. The user is presented with the final classified results (i.e. benign or suspicious of melanoma) along with values of the ABCD features. Fig 2.7(b) shows the original mole images overlaid with segmentation contours. It can be observed that the segmentation results are more sensitive to borders, showing more irregularity than it appears visually. This may be due to the high resolution (2988 × 5312 pixels) of the S6 camera, which is not improved after resizing to 1024 × 576 when using bilinear interpolation and a smoothing filter. Even with this drawback, all images were successfully processed and classified as benign. A supplemental file demonstrates the usage of our smartphone application on both the PH2 dataset and live images. In addition, we have tested our system on digital lesion images MED-NODE public dataset [5] and our sensitivity, selectivity, and accuracy values are 70%,

80%, and 75%. Some of the sample processed mole images from this dataset are included in the supplemental file. The software is available at *https://github.com/ukalwa/melanoma-detection* for the interested readers who want to build upon our current prototype.

Discussion

Looking at the reported melanoma cases over the past few years, there are some insightful points that can be enumerated. According to the statistics released by the American Cancer Society for the year 2018, there are 91,270 new cases (55,150 males; 36,120 females) of melanoma in the U.S. Out of this population, there have been an estimated 9,320 deaths (5,990 males; 3,330 females). The states with the highest number of new melanoma cases in 2018 are California (9,830), Florida (7,940), New York (4,920), Texas (4,440), and Pennsylvania (4,320) [58]. Upon comparing to the statistics from 2014, there were 76,100 new cases (43,890 males; 32,210 females) of melanoma within the U.S. with an estimated 9,710 deaths (6,470 males; 3,240 females) [59]. The above data indicates that in the U.S. both the number of new melanoma cases and fatalities have increased in the past five years. Furthermore, the states reporting higher incidence of melanoma correspond with sunny regions and larger populations, where it is challenging to screen and educate the masses about skin cancer and early diagnosis. It can reasonably be assumed that the statistics on melanoma incidence and fatalities would be worse in the developing countries where access to diagnostic/preventive medical resources are still sought. More importantly, the statistics indicate that there is still an urgent need for portable melanoma screening devices that can be readily adopted.

Smartphone-based skin cancer recognition remains a challenging area of research, and this has slowed the commercialization and general availability of portable melanoma screening devices. Some of the technological limitations of the previously reported methods include: (a)

the application used a non-smartphone based camera with a dermoscope to photograph lesions [13,14,19,20,29], (b) the application was tested on small image sets or different databases making it difficult to directly compare the results [15,16,19,22], (c) the application demonstrated a single feature extraction [2,30,60], (d) the application ran only on a desktop and thus was not considered real-time [13,19,21,25,54], and (e) the application had average accuracy, sensitivity, or selectivity [22,50,51]. We showed that our smartphone application is able to overcome the above limitations with better or comparable computation times and accuracies to those reported earlier.

The scope of our work can be extended in terms of performance and usage. The performance parameters (i.e. accuracy, sensitivity, selectivity, and processing time) could be improved by using better thresholding algorithms for boundary detection, testing other color space parameters (shape, color, and texture distributions), and running it on a large variety of images. The application runs solely on a smartphone, so its usage could entail tracking the progression of specific moles over a period of time. To do this, the user would capture and save mole images at different time intervals over a two-week period, recording the ABCD parameter values for each image. From these recorded values, it would be possible to observe any changes in the shape, size, or color of the mole that may then be evaluated by a healthcare professional. The application's reliability and reproducibility can be tested on persons with different skin colors, under different background illumination conditions, and with different stages of lesion malignancy. With the sophistication of smartphone optics today, it may be possible to directly visualize a lesion's distribution in the different skin layers leading to accurate identification of lesion attributes that may be missed especially during the early stages of melanoma. Due to the challenge of forming collaborative relationships with skin clinics,

most researchers have relied on public databases for their research. This makes direct comparisons of performance metrics across different methods difficult. The present study can be expanded beyond the ABCD rule to understand the role of each feature in the eventual classification of melanoma or benign lesions. The commonly used discriminators include shape features (e.g. asymmetry, aspect ratio, maximum diameter), color features in the color spaces (mean distance, variance, maximum distance), and texture features (e.g. grayscale co-occurrence matrix and texture descriptors). While all three feature types are equally relevant in accurate discrimination and classification, the color features have been shown to perform better than the texture features [21].

Conclusions

The current prototype comprises a smartphone application to capture or import images of skin lesions, perform feature extraction based on the ABCD rule, and classify their malignancy based on the SVM classifier. The application is tested on 200 dermoscopic images from the PH2 database and the benign moles of two individuals. The entire process from image capture to classification runs entirely on an Android smartphone equipped with a detachable 10x lens, and has a processing time within one second per image. Easy-to-use navigation buttons are incorporated at the front-end to assist the user through the various processing steps. For the PH2 database, the overall performance is better with SMOTE (80% sensitivity, 90% specificity, 88% accuracy, and 0.85 AUC) compared to without using SMOTE (55% sensitivity, 95% specificity, 90% accuracy, and 0.75 AUC). Future scope of improvement lies in training with even larger image datasets, having access to individuals with possible melanoma cases, and testing under varied environmental conditions and disease stages.

References

1. Korotkov K, Garcia R. Computerized analysis of pigmented skin lesions: A review. Artif Intell Med. Elsevier B.V.; 2012;56: 69–90. doi:10.1016/j.artmed.2012.08.002

2. Jaworek-Korjakowska J. Novel method for border irregularity assessment in dermoscopic color images. Comput Math Methods Med. 2015; doi:10.1155/2015/496202

3. Scharcanski J, Celebi ME. Computer vision techniques for the diagnosis of skin cancer. Springer Science & Business Media; 2013. doi:10.1007/978-3-642-39608-3

4. Abuzaghleh O, Faezipour M, Barkana BD. SKINcure: An Innovative Smartphone-Based Application to Assist in Melanoma Early Detection and Prevention. Signal Image Process an Int J. 2014;15: 1–13. doi:10.5121/sipij.2014.5601

5. Giotis I, Molders N, Land S, Biehl M, Jonkman MF, Petkov N. MED-NODE: A computer-assisted melanoma diagnosis system using non-dermoscopic images. Expert Syst Appl. 2015;42: 6578–6585. doi:10.1016/j.eswa.2015.04.034

6. Abbas Q, Celebi ME, García IF. Hair removal methods: A comparative study for dermoscopy images. Biomed Signal Process Control. Elsevier Ltd; 2011;6: 395–404. doi:10.1016/j.bspc.2011.01.003

7. Mayer JE, Swetter SM, Fu T, Geller AC. Screening, early detection, education, and trends for melanoma: Current status (2007-2013) and future directions Part II. Screening, education, and future directions. J Am Acad Dermatol. Elsevier Inc; 2014;71: 611.e1-611.e10. doi:10.1016/j.jaad.2014.05.045

8. Rigel DS, Russak J, Friedman R. The evolution of melanoma diagnosis: 25 years beyond the ABCDs. CA Cancer J Clin. 2010;60: 301–316. doi:10.3322/caac.20074

9. Erkol B, Moss RH, Stanley RJ, Stoecker W V, Hvatum E. Automatic lesion boundary detection in dermoscopy images using gradient vector flow snakes. Ski Res Technol. 2005;11: 17–26. doi:10.1111/j.1600-0846.2005.00092.x

10. Mendonça T, Marçal ARS, Vieira A, Nascimento JC, Silveira M, Marques JS, et al. Comparison of segmentation methods for automatic diagnosis of dermoscopy images. Proceedings of the 29th Annual International Conference of the IEEE EMBS. 2007. pp. 6572–6575. doi:10.1109/JSTSP.2008.2011119

11. Kopf AW, Salopek TG, Slade J, Marghoob AA, Bart RS. Techniques of cutaneous examination for the detection of skin cancer. Cancer. 1995;75: 684–690. doi:10.1002/1097-0142(19950115)75:2+<684::AID-CNCR2820751411>3.0.CO;2-B

12. Green A, Martin N, Pfitzner J, O'Rourke M, Knight N. Computer image analysis in the diagnosis of melanoma. J Am Acad Dermatol. 1994;31: 958–964. doi:10.1016/S0190-9622(94)70264-0

13. Massone C, Hofmann-Wellenhof R, Ahlgrimm-Siess V, Gabler G, Ebner C, Peter Soyer H. Melanoma Screening with Cellular Phones. PLoS One. 2007;2. doi:10.1371/journal.pone.0000483

14. Kroemer S, Frühauf J, Campbell TM, Massone C, Schwantzer G, Soyer HP, et al. Mobile teledermatology for skin tumour screening: Diagnostic accuracy of clinical and dermoscopic image tele-evaluation using cellular phones. Br J Dermatol. 2011;164: 973–979. doi:10.1111/j.1365-2133.2011.10208.x

15. Karargyris A, Karargyris O, Pantelopoulos A. DERMA/Care: An advanced image-processing mobile application for monitoring skin cancer. Proc - Int Conf Tools with Artif Intell ICTAI. 2012;2: 1–7. doi:10.1109/ICTAI.2012.180

16. Do TT, Zhou Y, Zheng H, Cheung NM, Koh D. Early melanoma diagnosis with mobile imaging. Conf Proc . Annu Int Conf IEEE Eng Med Biol Soc IEEE Eng Med Biol Soc Annu Conf. 2014;2014: 6752–6757. doi:10.1109/EMBC.2014.6945178

17. Al-Tairi ZH, Rahmat RW, Iqbal Saripan M, Sulaiman PS. Skin segmentation using YUV and RGB color spaces. J Inf Process Syst. 2014;10: 283–299. doi:10.3745/JIPS.02.0002

18. Ennehar BC, Brahim O, Hicham T. An Appropriate Color Space to Improve Human Skin Detection. INFOCOMP; Vol 9 No 4 December, 2010. 2010; Available: http://infocomp.dcc.ufla.br/index.php/INFOCOMP/article/view/317

19. Majtner T, Yildirim-Yayilgan S, Hardeberg JY. Efficient Melanoma Detection Using Texture-Based RSurf Features. Int Conf Image Anal Recognition Springer Int Publ. 2016;9730: 30–37. doi:10.1007/978-3-319-41501-7

20. Amelard R, Glaister J, Wong A, Clausi DA. Melanoma Decision Support Using Lighting-Corrected Intuitive Feature Models. In: Scharcanski J, Celebi ME, editors. Computer Vision Techniques for the Diagnosis of Skin Cancer. Berlin, Heidelberg: Springer Berlin Heidelberg; 2014. pp. 193–219. doi:10.1007/978-3-642-39608-3_7

21. Barata C, Ruela M, Francisco M, Mendonca T, Marques JS. Two systems for the detection of melanomas in dermoscopy images using texture and color features. IEEE Syst J. 2014;8: 965–979. doi:10.1109/JSYST.2013.2271540

22. Aleem M, Hameed N, Anjum A. m-Skin Doctor : A Mobile Enabled System for Early Melanoma Skin Cancer Detection Using Support Vector Machine. eHealth 360°, Springer International Publishing. 2017. pp. 468–475. doi:10.1007/978-3-319-49655-9

23. Oliveira RB, Filho ME, Ma Z, Papa JP, Pereira AS, Tavares JMRS. Computational methods for the image segmentation of pigmented skin lesions: A review. Comput Methods Programs Biomed. Elsevier Ireland Ltd; 2016;131: 127–141. doi:10.1016/j.cmpb.2016.03.032

24. Garnavi R, Aldeen M, Celebi ME, Varigos G, Finch S. Border detection in dermoscopy images using hybrid thresholding on optimized color channels. Comput Med Imaging Graph. 2011;35: 105–115. doi:10.1016/j.compmedimag.2010.08.001

25. Oliveira RB, Marranghello N, Pereira AS, Tavares JMRS. A computational approach for detecting pigmented skin lesions in macroscopic images. Expert Syst Appl. Elsevier Ltd; 2016;61: 53–63. doi:10.1016/j.eswa.2016.05.017

26. Barcelos CAZ, Pires VB. An automatic based nonlinear diffusion equations scheme for skin lesion segmentation. Appl Math Comput. Elsevier Inc.; 2009;215: 251–261. doi:10.1016/j.amc.2009.04.081

27. Deng G, Cahill LW. An adaptive Gaussian filter for noise reduction and edge detection. 1993 IEEE Conf Rec Nucl Sci Symp Med Imaging Conf. 2014; 1615–1619. doi:10.1109/NSSMIC.1993.373563

28. Pennisi A, Bloisi DD, Nardi D, Giampetruzzi AR, Mondino C, Facchiano A. Skin lesion image segmentation using Delaunay Triangulation for melanoma detection. Comput Med Imaging Graph. Elsevier Ltd; 2016;52: 89–103. doi:10.1016/j.compmedimag.2016.05.002

29. Blackledge J, Dubovitskiy DA. MoleTest TM : A Web-based Skin Cancer Screening System. The Third International Conference on Resource Intensive Applications and Services. 2011. p. 162. Available: http://eleceng.dit.ie/blackledge

30. Ma Z, Tavares JMRS. A Novel Approach to Segment Skin Lesions in Dermoscopic Images Based on a Deformable Model. IEEE J Biomed Heal Informatics. 2016;20: 615–623. doi:10.1109/JBHI.2015.2390032

31. Jain S, Jagtap V, Pise N. Computer aided melanoma skin cancer detection using image processing. Procedia Comput Sci. Elsevier Masson SAS; 2015;48: 735–740. doi:10.1016/j.procs.2015.04.209

32. Chan TF, Vese LA. Active contours without edges. IEEE Trans Image Process. 2001;10: 266–277. doi:10.1109/83.902291

33. Shi Y, Karl WC. A real-time algorithm for the approximation of level-set-based curve evolution. IEEE Trans Image Process. 2008;17: 645–656. doi:10.1109/TIP.2008.920737

34. Şuta L, Bessy F, Veja C, Vaida MF. Active contours: Application to plant recognition. Proc - 2012 IEEE 8th Int Conf Intell Comput Commun Process ICCP 2012. 2012; 181–187. doi:10.1109/ICCP.2012.6356183

35. Friedman RJ, Rigel DS, Kopf a W. Early detection of malignant melanoma: the role of physician examination and self-examination of the skin. CA Cancer J Clin. 1985;35: 130–151. doi:10.3322/canjclin.35.3.130

36. She Z, Liu Y, Damatoa A. Combination of features from skin pattern and ABCD analysis for lesion classification. Ski Res Technol. 2007;13: 25–33. doi:10.1111/j.1600-0846.2007.00181.x

37. Yagerman SE, Chen L, Jaimes N, Dusza SW, Halpern AC, Marghoob A. "Do UC the melanoma?" Recognising the importance of different lesions displaying unevenness or having a history of change for early melanoma detection. Aust J Dermatol. 2014;55: 119–124. doi:10.1111/ajd.12143

38. Zhou Y, Smith M, Smith L, Warr R. A new method describing border irregularity of pigmented lesions. Ski Res Technol. 2010;16: 66–76. doi:10.1111/j.1600-0846.2009.00403.x

39. Sáez A, Acha B, Serrano C. Pattern Analysis in Dermoscopic Images. Computer Vision Techniques for the Diagnosis of Skin Cancer. 2014. pp. 23–48. doi:10.1007/978-3-642-39608-3_2

40. Pehamberger H, Steiner A, Wolff K. In vivo epiluminescence microscopy of pigmented skin lesions. I. Pattern analysis of pigmented skin lesions. J Am Acad Dermatol. 1987;17: 571–583. doi:10.1016/S0190-9622(87)70239-4

41. Forsea AM, Tschandl P, Zalaudek I, del Marmol V, Soyer HP, Argenziano G, et al. The impact of dermoscopy on melanoma detection in the practice of dermatologists in Europe: results of a pan-European survey. J Eur Acad Dermatology Venereol. 2017;31: 1148–1156. doi:10.1111/jdv.14129

42. Henning JS, Dusza SW, Wang SQ, Marghoob AA, Rabinovitz HS, Polsky D, et al. The CASH (color, architecture, symmetry, and homogeneity) algorithm for dermoscopy. J Am Acad Dermatol. 2007;56: 45–52. doi:10.1016/j.jaad.2006.09.003

43. Leachman SA, Cassidy PB, Chen SC, Curiel C, Geller A, Gareau D, et al. Methods of Melanoma Detection. Melanoma. 2016. pp. 51–105. doi:10.1007/978-3-319-22539-5_3

44. Menzies SW, Ingvar C, Crotty KA, McCarthy WH. Frequency and morphologic characteristics of invasive melanomas lacking specific surface microscopic features. Arch Dermatol. 1996; doi:10.1001/archderm.132.10.1178

45. Tsao H, Olazagasti JM, Cordoro KM, Brewer JD, Taylor SC, Bordeaux JS, et al. Early detection of melanoma: Reviewing the ABCDEs. J Am Acad Dermatol. 2015;72: 717–723. doi:10.1016/j.jaad.2015.01.025

46. Shaw HM, Rigel DS, Friedman RJ, Mccarthy WH, Kopf AW. Early diagnosis of cutaneous melanoma: revisiting the ABCD criteria. Jama. 2004;292: 2771–2776. doi:10.1001/jama.292.22.2771

47. Harrington E, Clyne B, Wesseling N, Sandhu H, Armstrong L, Bennett H, et al. Diagnosing malignant melanoma in ambulatory care: A systematic review of clinical prediction rules. BMJ Open. 2017;7. doi:10.1136/bmjopen-2016-014096

48. Robinson JK, Turrisi R. Skills training to learn discrimination of ABCDE criteria by those at risk of developing melanoma. Arch Dermatol. 2006; doi:10.1001/archderm.142.4.447

49. Gadeliya Goodson A, Grossman D. Strategies for early melanoma detection: Approaches to the patient with nevi. Journal of the American Academy of Dermatology. 2009. doi:10.1016/j.jaad.2008.10.065

50. Ramlakhan K, Shang Y. A mobile automated skin lesion classification system. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI. 2011. pp. 138–141. doi:10.1109/ICTAI.2011.29

51. Doukas C, Stagkopoulos P, Kiranoudis CT, Maglogiannis I. Automated skin lesion assessment using mobile technologies and cloud platforms. Proc Annu Int Conf IEEE Eng Med Biol Soc EMBS. 2012; 2444–2447. doi:10.1109/EMBC.2012.6346458

52. Pennisi A, Bloisi DD, Nardi D, Giampetruzzi AR, Mondino C, Facchiano A. Melanoma detection using delaunay triangulation. Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI. 2016. pp. 797–798. doi:10.1109/ICTAI.2015.117

53. Esteva A, Kuprel B, Novoa RA, Ko J, Swetter SM, Blau HM, et al. Dermatologist-level classification of skin cancer with deep neural networks. Nature. Nature Publishing Group; 2017;542: 115–118. doi:10.1038/nature21056

54. Yu L, Chen H, Dou Q, Qin J, Heng PA. Automated Melanoma Recognition in Dermoscopy Images via Very Deep Residual Networks. IEEE Trans Med Imaging. 2017;36: 994–1004. doi:10.1109/TMI.2016.2642839

55. Burges CJC. A Tutorial on Support Vector Machines for Pattern Recognition. Data Min Knowl Discov. 1998;2: 121–167. doi:10.1023/A:1009715923555

56. Mendonca T, Ferreira PM, Marques JS, Marcal ARS, Rozeira J. PH2 - A dermoscopic image database for research and benchmarking. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS. 2013. pp. 5437–5440. doi:10.1109/EMBC.2013.6610779

57. Chawla N V., Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: Synthetic minority over-sampling technique. J Artif Intell Res. 2002;16: 321–357. doi:10.1613/jair.953

58. Cancer Facts & Figures. In: American Cancer Society [Internet]. 2018 [cited 23 Aug 2018]. Available: https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2018/cancer-facts-and-figures-2018.pdf

59. Cancer Facts & Figures. In: American Cancer Society [Internet]. 2014 [cited 23 Aug 2018]. Available: https://www.cancer.org/content/dam/cancer-org/research/cancer-facts-and-statistics/annual-cancer-facts-and-figures/2014/cancer-facts-and-figures-2014.pdf

60. Xie F, Bovik AC. Automatic segmentation of dermoscopy images using self-generating neural networks seeded by genetic algorithm. Pattern Recognit. 2013;46: 1012–1019. doi:10.1016/j.patcog.2012.08.012

CHAPTER 3.

NEW METHODS OF CLEANING DEBRIS AND HIGH-THROUGHPUT COUNTING OF CYST NEMATODE EGGS EXTRACTED FROM FIELD SOIL

Modified from the journal article

Upender Kalwa, Christopher Legner, Elizabeth Wlezien, Gregory Tylka, and Santosh Pandey. "New methods of cleaning debris and high-throughput counting of cyst nematode eggs extracted from field soil" PLOS One, 2019, (Accepted with minor revisions)

Abstract

The soybean cyst nematode (SCN), *Heterodera glycines*, is the most damaging pathogen of soybeans in the United States. To assess the severity of nematode infestations in the field, SCN egg population densities are determined. Cysts (dead females) of the nematode must be extracted from soil samples and then ground to extract the eggs within. Sucrose centrifugation commonly is used to separate debris from suspensions of extracted nematode eggs. We present a method using OptiPrep as a density gradient medium with improved separation and recovery of extracted eggs compared to the sucrose centrifugation technique. Also, computerized methods were developed to automate the identification and counting of nematode eggs from the processed samples. In one approach, a high-resolution scanner was used to take static images of extracted eggs and debris on filter papers, and a deep learning network was trained to identify and count the eggs among the debris. In the second approach, a lens-less imaging setup was developed using off-the-shelf components, and the processed egg samples were passed through a microfluidic flow chip made from double-sided adhesive tape. Holographic videos were recorded of the passing eggs and debris, and the videos were reconstructed and processed by custom software program to obtain egg counts. The performance of the software programs for egg counting was characterized with SCN-infested soil collected from two farms, and the results using these methods were compared with those obtained through manual counting.

Introduction

Many nematodes (microscopic roundworms) are soil-dwelling plant parasites that infect the roots of plants and cause billions of dollars of crop loss worldwide on an annual basis. *Heterodera glycines*, the soybean cyst nematode (SCN), is the most damaging soybean pathogen in the United States and Canada, causing hundreds of millions of dollars in crop loss annually [1]. The SCN infects the roots of soybeans and siphons nutrients from the plants, leading to stunted growth and reduced crop yields. The amount of damage and yield loss caused by the nematode is related to several factors, including the number of eggs (egg population density) in the soil. Knowing the SCN population density in the soil can be useful in guiding the use of management strategies and assessing the success of management efforts in SCN-infested fields.

To determine the population density of SCN in a field, one or more multiple-core soil samples are collected. On a sample-by-sample basis, the cysts (egg-filled dead SCN females) are extracted from samples, and the eggs are extracted from the cysts, then counted using a microscope. Two methods commonly used to extract nematode cysts from the soil are (i) wet sieving and decanting, and (ii) elutriation. Wet sieving and decanting [2] involve suspending soil in water, agitating the suspension, allowing the heavier soil particles to settle to the bottom of the container, and pouring the suspension through two sieves. The top sieve (usually with 850-µm-diameter pores) will capture root fragments and other debris, which will be discarded, and the bottom sieve (usually with 250-µm-diameter pores) will capture nematode cysts and cyst-sized debris. Smaller debris (<250 µm in diameter) will pass through both sieves and be discarded. For the elutriation method, soil is suspended in a column or cone of upward-flowing water. Heavy soil particles remain near the bottom of the flowing water suspension, while cysts

and less dense objects float and pour out of the top of the column or cone to be captured on sieves [3] as with the wet-sieving and decanting method.

Currently, two methods commonly used for extracting eggs from nematode cysts are (i) by grinding the cysts in a plastic or glass tube with a stainless-steel pestle [4] or Teflon tissue homogenizer pestle and (ii) by grinding the cysts on a 250-µm-pore sieve with a rubber stopper [5]. Both methods result in the capture of eggs and similarly sized debris on 25-µm-pore sieves. Accurately and efficiently counting the eggs to determine the population density can be difficult and inefficient when considerable amounts of debris are recovered with the eggs. Separating the eggs from the debris often is necessary to make counting possible. Also it is desirable to separate and discard debris from eggs in suspension when the eggs are to be used in laboratory experiments, such as when assessing development and studying hatching of nematode juveniles from the eggs [6–8].

Sucrose centrifugation [9] is a method commonly used to separate plant-parasitic nematode juveniles and eggs in suspension from debris. Although sucrose centrifugation is inexpensive and easy, it may not be the most efficient method and it may have adverse effects on the nematodes. Exposing nematode eggs and juveniles to sucrose solutions, with a high osmotic potential, may harm the eggs and juveniles if exposure to the sucrose is prolonged. Also, if the nematodes are not thoroughly rinsed with water following centrifugation in sucrose solution, sucrose residues may remain and promote bacterial and fungal growth on the eggs and juveniles. Deng et al. [10] suggested an alternative density gradient method to sucrose centrifugation using an iodixanol solution, OptiPrep™ (also known as Visipague™ in medical uses). Testing the effects of a single concentration of OptiPrep™, they found that the efficiency of extraction of the reniform nematode (*Rotylenchulus reniformis*) and post-extraction mobility

of the recovered nematodes were both 100% greater than when the sucrose centrifugation method was used.

Following the extraction and cleaning of egg samples, the eggs are counted to determine the population density of nematodes. Counting is completed manually through microscopic observation, requiring trained personnel. The process is time intensive, laborious, and prone to human error. It would be ideal if this portion of the process could be automated to address these shortcomings. Furthermore, automated egg counting could reduce labor costs and processing fees compared to manual counting methods where the expense increases with sample quantity,

In order to increase efficiency and automation of working with the eggs of cyst nematodes, the objectives of this work were: 1) to develop a method of purifying eggs from suspension with debris and 2) to develop new methods to automate counting of eggs once extracted from cysts and purified. In the first counting approach, a high-resolution scanner takes images of the processed sample (i.e. stained eggs with debris) dispersed on filter paper that was then run through deep learning algorithms to automatically identify and count the eggs. In a second counting approach, a benchtop, lensless imaging setup takes real-time, holographic videos of the processed sample passing through a flow chip and is analyzed with a custom software program to determine the egg count. We performed detailed characterization of the proposed methods, while attempting to minimize the number of manual steps.

Materials and methods

Sample preparation

Soil samples were collected from two fields in Muscatine and Story County in Iowa. The sample preparation involved two steps: egg extraction and egg staining.

To extract the eggs, each soil sample was poured in a bucket filled with water (approximately 2 liters), mixed thoroughly, and allowed to settle. The soil suspension was then poured through a 20-cm-diameter sieve with 850-µm-diameter pores above a 20-cm-diameter sieve with 250-µm-diameter pores [2]. The debris and egg-filled cysts (dead nematode females) were collected on the 250-µm-pore sieve and then transferred to a 3.7-cm-diameter, 250-µm-pore sieve and were crushed using a motorized rubber stopper to release the eggs [5]. These eggs (along with similar sized debris) were collected on a 15-cm-diameter sieve with 37-µm-diameter pores and then transferred into a microwavable container [11].

For staining the eggs, a stain solution was prepared by adding 3.5 g acid fuchsin (F8129, Sigma Aldrich) and 250 mL glacial acetic acid (ARK2183, Sigma Aldrich) to 750 mL of distilled water and stirred well [4]. One drop of the stain and four to five drops of 1 M HCl solution were added to the plastic beakers containing the eggs. The beakers and their contents were heated in a microwave for 15 seconds to stain the eggs [12].

Sample cleaning method using OptiPrep™-based centrifugation

Fig 3.1a is a schematic representation of the protocol for sample cleaning using OptiPrep™ as the density gradient medium. The protocol consists of two steps: centrifugation and separation. During the centrifugation step, 3 mL of the density gradient medium (OptiPrep™) at a specific volume percentage was put in a 15 mL centrifuge tube (Corning™). Thereafter, 5 mL of the stained egg sample was pipetted on top of the OptiPrep™ solution, forming an emulsion interface layer between the solutions of different densities as shown in

Fig 3.1a(i.). The combined sample was centrifuged at 840 G for 2 minutes. Particles with higher density than the gradient solution passed through the emulsion layer and were pelleted at the bottom. Particles with comparable density to the gradient solution were concentrated in the interface layer, while particles having lower density floated in the top layer of the solution. The contents of the centrifuge tube were categorized into three layers: top (4 mL), interface (2 mL), and bottom (2 mL). In the separation step, each of the three layers were pipetted into three different 15 mL test tubes and diluted to 11 mL with water, as shown in Fig 3.1a(ii.).

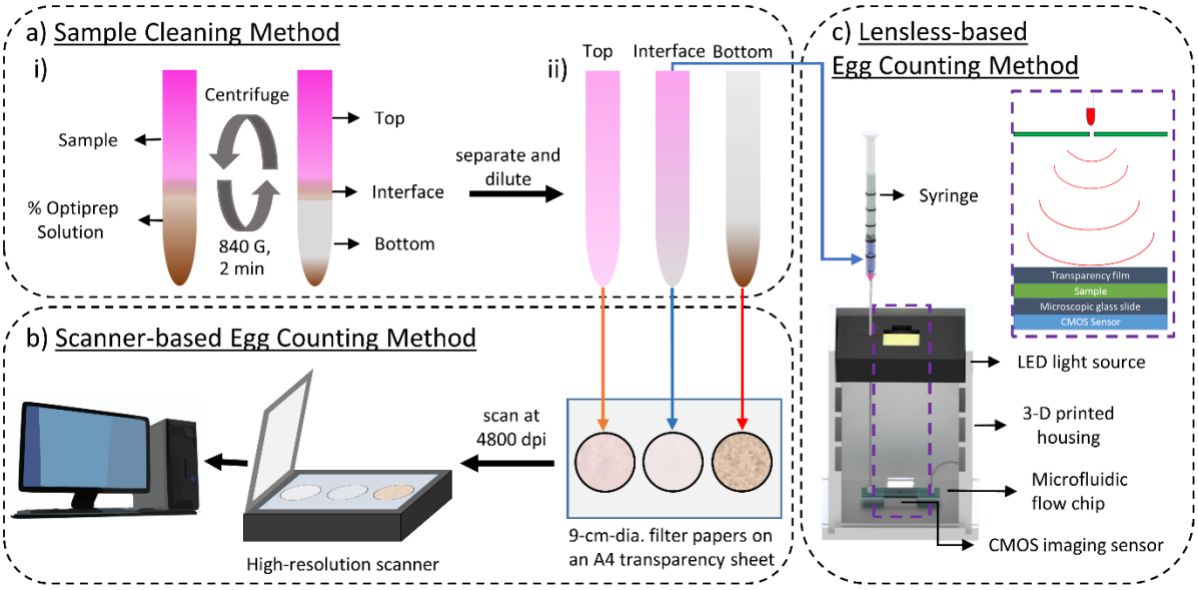


Figure 3.1 Overview of the proposed methods of sample cleaning and egg counting. a) Sample cleaning used a density gradient medium (OptiPrep™) and consisted of two steps: centrifugation and separation. i) An illustration of the density gradient centrifugation is shown where the three distinct layers (top, interface, and bottom) were visible after the centrifugation step. ii) The three layers were separated and diluted with water to 11 mL. b) A scanner-based egg counting method is shown where three solutions were dispersed on separate filter papers, allowed to air dry, and then placed face-down on an A4-sized (21.6 × 27.9 cm) transparency sheet. The filter papers were scanned at 4800 dpi and analyzed by a deep learning network model to obtain a count of the eggs. c) The lensless setup consisted of a light-emitting diode (LED) to illuminate the processed sample flowing through a microfluidic flow chip and a CMOS image sensor to record videos of the eggs in real-time. A custom software program, written in python, analyzed the videos and produced the egg count.

Scanner-based egg counting method:

Hardware and software components

Fig 3.1b illustrates the scanner-based egg counting method. Here the three distinct layers of processed sample (top, interface, and bottom) obtained after centrifugation in OptiPrep™ were poured on separate circular filter papers (Grade 41, Whatman™, 90 mm diameter) and allowed to air dry at room temperature (approximately 22 ˚C). The filter papers were placed on an A4-sized (21.6 × 27.9 cm) transparency sheet face-down and placed on a flatbed scanner (Epson Perfection v750 Pro) connected to a desktop computer. Using the scanner software (Epson Scan software, Version 3.921), each piece of filter paper was selected with a bounding box and scanned as shown in Fig 3.1b. The following settings were chosen: reflective scanning mode, 4800 dpi, and 24-bit color. All other imaging parameters were set to their default values. The scanned images were saved as JPEG files to the computer hard drive.

Data collection and pre-processing

Each scanned image of the filter paper (17759 × 17759 pixels) was split into 4900 patch images (256 × 256 pixels). The colorspace of each patch image was converted from RGB (Red, Green and Blue) to HSV (Hue, Saturation, and Value) to distinguish the different colors because the HSV colorspace separated the color information (chroma) from the image intensity (luma). A range of HSV values was selected to identify objects of a similar color to the stained eggs within the patch image. Then by applying thresholds on the object's physical dimensions (i.e. width, height, shape, and area), all of the eggs were detected in the patch image. A label image with the eggs was created and stored along with the patch image on the hard drive. The above process was repeated for all the patch images, and subsequently used for training and testing the deep learning network. The patch and label images were resized to 128 × 128 pixels to reduce the number of training parameters and model size. The pixels were normalized to

ensure that all the features were given equal importance. The total data set consisted of 60 filter paper images from 20 different soil samples, yielding 294,000 patch images, which were randomly divided into training and test data sets (80:20 split) to be used to develop the deep learning model.

Deep learning network architecture

To automatically learn the features related to the SCN eggs from patch and label images, we have employed a convolutional autoencoder network — a specific type of autoencoder network that used convolutional layers [13,14]. In general, a 'convolutional network' is composed of four layers: convolution, activation, pooling, and dropout layers. The convolution layer had neurons with weights and biases, which were updated after every iteration by a backpropagation algorithm in the training process. The activation layer consisted of a non-linearity function and performed mathematical operations on the input. An example activation function is the ReLU (Rectified Linear Unit) and has been shown to greatly accelerate the training process [13]. The pooling layer performed non-linear down-sampling on the input image by extracting the maximum or average of all the non-overlapping sub-regions in the image. This layer reduced the number of training parameters and memory footprint of the network. The dropout layer selected a random set of neurons determined by a percentage probability and set their inputs to zero making them unusable in the decision-making process of the network. Besides the convolutional network, the 'autoencoder neural network' presented an unsupervised learning platform to provide an approximate mapping of the inputs and outputs. It consisted of the 'encoder' and 'decoder' paths. The 'encoder' path compressed the input information by downsampling and learned important features, while the 'decoder' path reconstructed an approximate higher dimensional output utilizing upsampling operations.

A schematic of our network architecture is shown in Fig 3.2, which is a modified version of the U-Net convolutional autoencoder model [15]. We added dropout layers to prevent overfitting, replaced unpadded convolutions with padded convolutions to avoid cropping operations, and replaced the soft-max activation with sigmoid activation. We reduced the feature maps to half of the original size, which decreased the number of parameters.

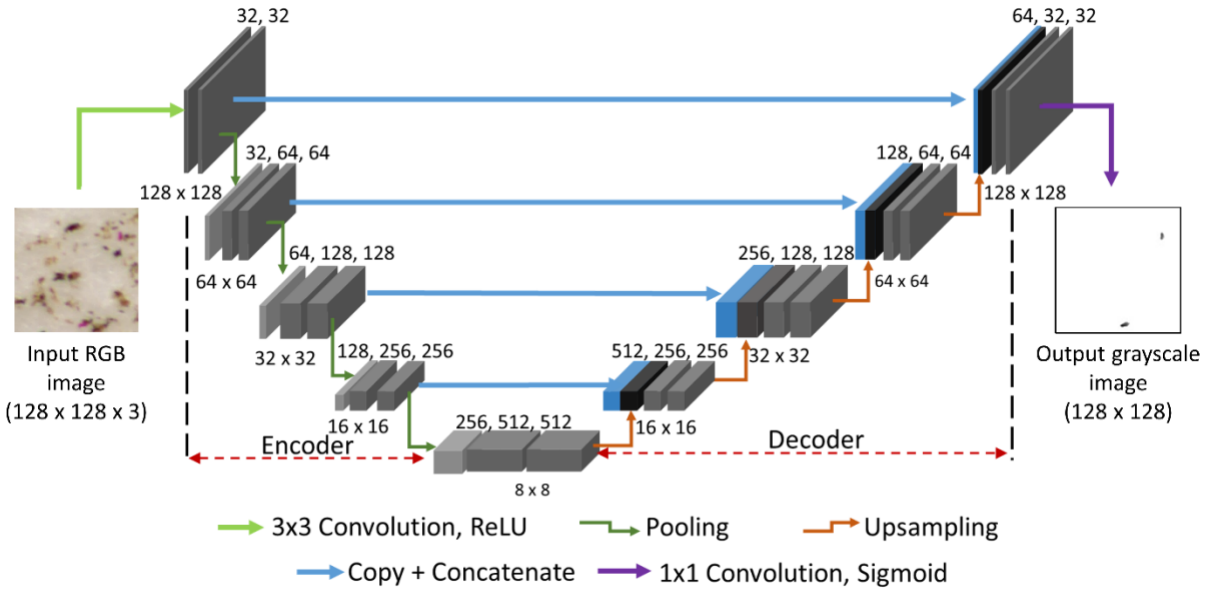


Figure 3.2 The convolutional autoencoder network model. a) Each box represents a three-dimensional feature map (W, H, and D). The depth (D) of each box is mentioned on top and the x-y sizes (W, H) on the bottom. The blue box represents the feature map copied from the encoder step and is concatenated with the feature map (black box) generated by upsampling the previous layer. The input to the network is a 3-channel RGB image and the output is a 1-channel grayscale image. The arrows represent different operations.

As shown in Fig 3.2, the deep learning network took 3-channel RGB patch images as inputs and produced 1-channel grayscale label images as outputs. The network consisted of nine layers; each layer had two convolutional layers and a max pooling layer. In the 'encoder' path, each step performed repeated convolutions with a 3×3 kernel followed by ReLU activation function and a 2×2 max pooling operation. In addition, the fourth and fifth layers contained dropout layers with probabilities of 0.4 and 0.3, respectively. The filters were

doubled as the downsampling operation was performed to move to the next layer. The number of filters started from 32 in the first layer and extended up to 512 in the fifth layer. Thereafter, the 'decoder' path was initialized wherein, at each step, an upsampling operation and 2×2 convolution was performed and followed by concatenation of the corresponding feature map from the 'encoder' path. Then, similar to the step in 'encoder' path, repeated convolutions with a 3×3 kernel followed by ReLU activation was performed. A final 1×1 convolution and sigmoid activation was performed to map the label image.

Training and testing the deep learning network model

The patch images and their corresponding label images were used to train the network with Adaptive Moment Estimation (ADAM), which is a type of stochastic optimization of cross entropy loss [16]. In total, the network had 7,760,097 trainable parameters and was processed using an NVIDIA™ Geforce GTX 1070 Ti graphics processing unit (GPU) with 8 gigabyte memory and 2432 CUDA cores. The batch size was set to 32 and the training was performed for 100 epochs. The model implementation was written in Python using Keras [17] and Tensorflow [18] libraries.

The test dataset comprising the patch images was run through the above trained model and the corresponding grayscale images were generated. A thresholding technique yielded binary images which were passed through a blob labelling algorithm to get the egg counts in the patch images [19].

Lensless imaging method:

Hardware and software components

Fig 3.1c shows the setup for the lensless egg counting. The basic principle of lensless imaging is discussed elsewhere [20] and has been used to image microscopic objects such as cells, bacteria, and even nematodes without the need for expensive microscopes. Our lensless

imaging hardware consisted of the lighting, imaging, and processing modules. (i) The lighting module comprised a light emitting diode (LED, wavelength λ = 616 µm, Vishay Intertechnology) that was aligned with a 100 µm pinhole (Edmund Optics) and sealed within a pinhole mount. This arrangement allowed the light from LED to be emitted through the pinhole producing spatial coherence [20]. The LED was connected in series with a 60 Ω resistor and is powered by the 5 V GPIO (general purpose input/output) pin on the Raspberry Pi 3 Model B (i.e. RPi) board. Fig 3.3a shows the wiring diagram. (ii) The imaging module comprised a Pi camera (8 MP) with a CMOS sensor. The default housing of the camera was removed using a standard razor blade to expose the CMOS sensor. Fig 3.3a shows the camera connected to the CSI (camera serial interface) port on RPi. (iii) The processing module comprised a portable RPi microcomputer having a 1.2 GHz 64 bit central processing unit (BCM2837), 1 GB random-access memory, and built-in Wi-Fi unit. The RPi ran on the Raspbian Jessie operating system (Debian "Jessie" based/Linux kernel) loaded onto a 64 GB microSD card (SanDisk™). Camera support was enabled in the settings, and open-source video streaming software was installed to access the live video feed remotely [21]. The RPi was powered by a micro USB wall-mount power supply (5 V, 2.1 A). A 3-D rendering of the platform that housed the various modules is shown in Fig 3.1c and served to reduce interference from ambient light. The platform was printed on a 3-D printer (da Vinci 1.0 Pro, XYZprinting) using a Polylactic acid filament (1.75 mm diameter, XYZprinting).

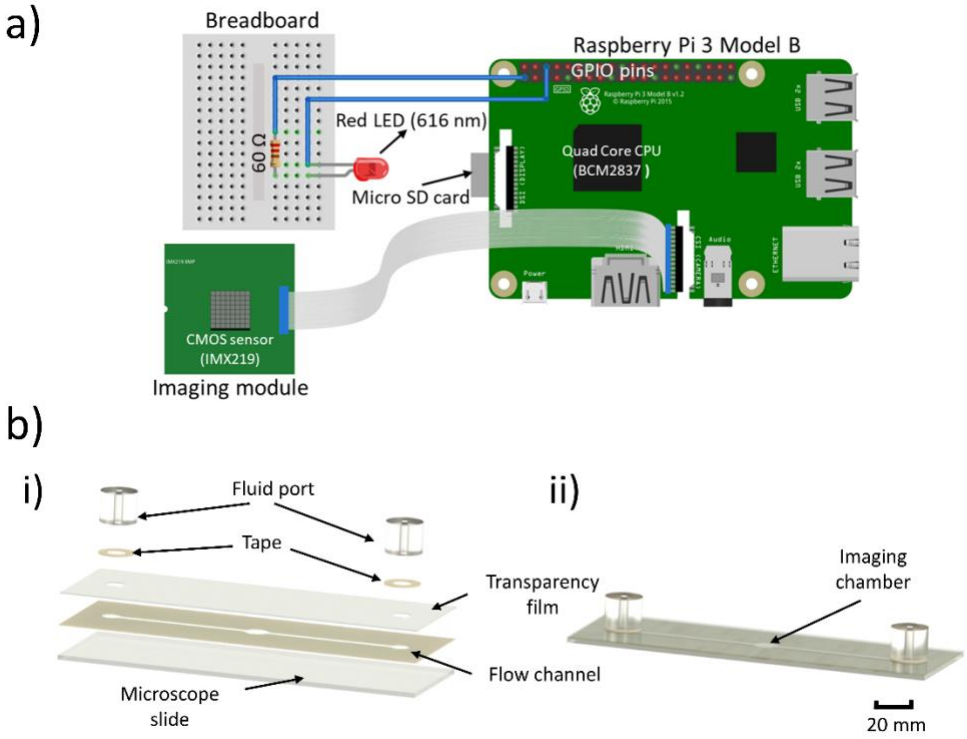


Figure 3.3 Wiring diagram for lensless imaging and design of the microfluidic flow chip. a) The CMOS sensor was connected to RPi through the CSI port. The 5V GPIO pin of RPi was connected to the anode of the LED while the ground pin from RPi was connected to the 60 Ω resistor. The cathode of the LED was connected to the other end of 60 Ω resistor. b) (i) An illustration of the microfluidic flow chip is shown. The microfluidic flow design was cut on a piece of double-sided tape using a cutting machine, and the tape was bonded to a microscope slide. A transparency film with input and output accesses was cut and adhered to the top side of the tape. Two circular tape pieces with holes cut in the center were attached to the fluid ports and placed around the access holes in the transparency sheet. (ii) A cartoon of the assembled microfluidic flow chip is shown with all of the layers bonded together. Scale bar = 20 mm.

After sample cleaning as described earlier, 1 mL of the processed sample was loaded in a 3 mL syringe (BD Biosciences) and connected to a syringe pump (KDS-100, KD Scientific). The standard needle of the syringe was replaced by a dispensing needle (16 gauge 2”, Howard Electronic Instruments) to avoid clogging. The syringe was positioned vertically and taped with a vibratory motor (3 V/60 mA, 6500 RPM, Jameco Reliapro) to prevent settling of debris in the tubing. The 3.3 V GPIO pin was used to power the vibratory motor. A plastic tubing (inner diameter: 16 gauge) dispensed the processed sample from the syringe to a

microfluidic flow chip. The RPi was turned on and video recording was enabled as the eggs and debris passed through the flow chip. Another plastic tubing was connected to the output port to direct the processed sample into a waste reservoir. After all the processed sample passed through the flow chip, the video recording was terminated.

Design of the microfluidic flow chip

The microfluidic flow chip was intended to be made using low-cost and simple-to-use materials and tools, thereby eliminating the need for micromachining or microfabrication techniques such as photomask design, lithography, spin coating, developing, curing, and etching. The flow chip comprised three layers: the base, channel, and cover layers. A microscope slide (25 × 75 × 1.0 mm, Fisherbrand™) served as the base layer. The channel layer was designed and cut from double-sided tape (25 × 75 × 0.2 mm, 3M™). The cover layer was constructed from a transparency sheet (Apollo™). Initially, commercial software (Studio™, Silhouette America) was used to create the design of the microfluidic flow channel (1.4 × 50 × 0.2 mm) with a central imaging chamber (2.1 × 2.4 × 0.2 mm). The different layers are shown in Fig 3.3b(i). Double-sided tape was attached to the cutting mat and loaded into the cutting machine (Cameo™, Silhouette America) to cut the flow channel onto the tape. The cut tape was then removed, aligned, and bonded to the base microscope slide. The design of the cover layer had holes for an inlet and an outlet which were aligned with the ends of the flow channel. The same cutting process described above was used to create the cover layer in a transparency sheet, which was placed and bonded to the channel layer. PDMS ports were used as input and output ports and sealed to the transparency film using circular rings of double-sided tape as shown in Fig 3.3b(i). The assembled flow chip is shown in Fig 3.3b(ii).

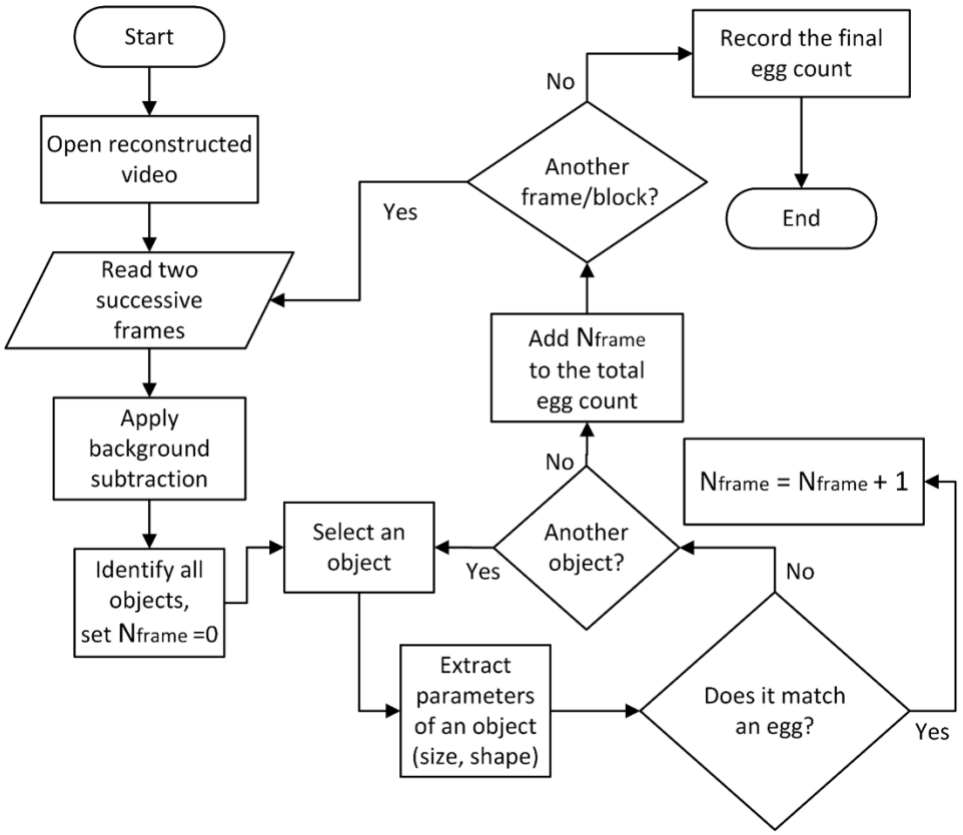


Figure 3.4 The algorithm for the lensless SCN egg counting method is depicted as a flow chart.

Data collection and pre-processing

The videos recorded from the RPi were reconstructed using a custom software program written in Python. As shown in Fig 3.4, the program loaded one video at a time and read all frames sequentially. The objects of interest (eggs) in a frame were identified by subtracting the current frame from the previous frame and filtering the noise using a median filter. This step eliminated any static content in the current frame including imperfections in the flow channel, settled debris, and image noise. Only the moving objects remained after this step, including eggs and debris. The resultant image was in a binary format with object pixels colored in white against a black background. From the binary image, different physical dimensions (i.e. width, height, shape, and area) of the identified objects were determined and compared with those of

a typical SCN egg. This helped identify if the selected object was a SCN egg. The process continued until all of the detected objects in the image were validated. The program then read the next frame and repeated the process until the final frame was reached. Thereafter, the total SCN egg count was reported for the processed sample.

Principle of reconstructing holographic images

The spatially incoherent light from the LED passed through an aperture resulting in a partially coherent light called the *reference wave* $U_R(x, y, z)$. The reference wave illuminated the objects located at distance z_1 in the channel of the flow chip. The incident light was scattered to generate the *object wave* $U_O(x, y, z)$. The interference between the reference wave and object wave produced holographic patterns. The intensity $I(x, y, z)$ of the holograms at a vertical distance z is described in Eq. (3.1) [22]:

$$I(x,y,z) = |U_R(x,y,z)|^2 + |U_O(x,y,z)|^2 + U_R{}^*(x,y,z)U_O(x,y,z) + U_R(x,y,z)U_O{}^*(x,y,z) \quad (3.19)$$

The first and second terms in Eq. (3.1) represented the background and scattered light intensity. The scattered intensity is generally weaker than that of the background. The third and fourth terms represented the interference maxima and minima [22]. The CMOS sensor was placed at a distance z_2 from the light source ($z_2 > z_1$) and recorded the intensity of the holograms [20,22]. However, the phase information was lost, which made it difficult to discern objects from a digital holographic image.

Digital reconstruction algorithms were used to convert the holographic image to a microscope-like image. The reconstruction commonly known as 'holographic reconstruction' converted the holographic image in the object plane to the detector plane. For our application, we have used the Fresnel diffraction method which utilized a single Fourier transform ($\mathcal{F}$) to

back propagate the wave a distance $z = z_2 - z_1$ to the reconstruction plane, $U_z(x, y, z)$. Eq. (3.2) describes how the reconstruction plane was obtained by Fourier transforming the element-wise multiplication of image intensity and combining it with a Fresnel approximated transfer function and a phase term. This operation can be written as [23]:

$$U_z(x, y, z) = \frac{e^{jkz}}{j\lambda z} e^{\frac{j\pi}{\lambda z}(x^2+y^2)} * \mathcal{F}\{I(x, y, z) * e^{\frac{j\pi}{\lambda z}(x^2+y^2)}\} \tag{3.20}$$

where λ is the wavelength and $k = \frac{2\pi}{\lambda}$ is the wave number. The pixel size $\Delta\eta$ in the reconstruction plane was directly proportional to the reconstruction plane as shown in Eq. (3.3).

$$\Delta\eta = \frac{\lambda z}{N\Delta_p} \tag{3.21}$$

where Δ_p is the pixel pitch of the CMOS sensor and N is the minimum dimension of the holographic image size. The reconstructed images at this stage often looked blurred because of the presence of zeroth order frequencies, which can be removed if the phase is known. An approximate phase was obtained using the Fresnel approximation. There are other techniques to recover the phase that involve taking multiple images of the same object by varying parameters such as wavelengths, angles of the LED, and z values [24,25]. Some other techniques use iterative methods to retrieve the phase. These alternatives are also effective but require additional imaging components and computations.

Results

OptiPrep™-based density gradient centrifugation

A qualitative analysis of the debris distribution was performed to characterize the effects of OptiPrep™ during the centrifugation step. Using the protocol described in Fig 3.1a, different OptiPrep™ solutions (0%, 20%, 40%, 50%, 60%, and 80% by volume) were used.

Images of the centrifuge tubes before and after the centrifugation step were taken for the different OptiPrep™ solutions and cropped as shown in Fig 3.5 (i,ii). The three layers were separated after centrifugation, diluted with water to 12 mL, and put on separate filter papers. Fig 3.5 (iii) shows a plot of the mean pixel intensity levels as a function of distance along the center axis of the tube. In these plots, the orange and purple lines denote the mean pixel intensity levels before and after the centrifugation step, respectively.

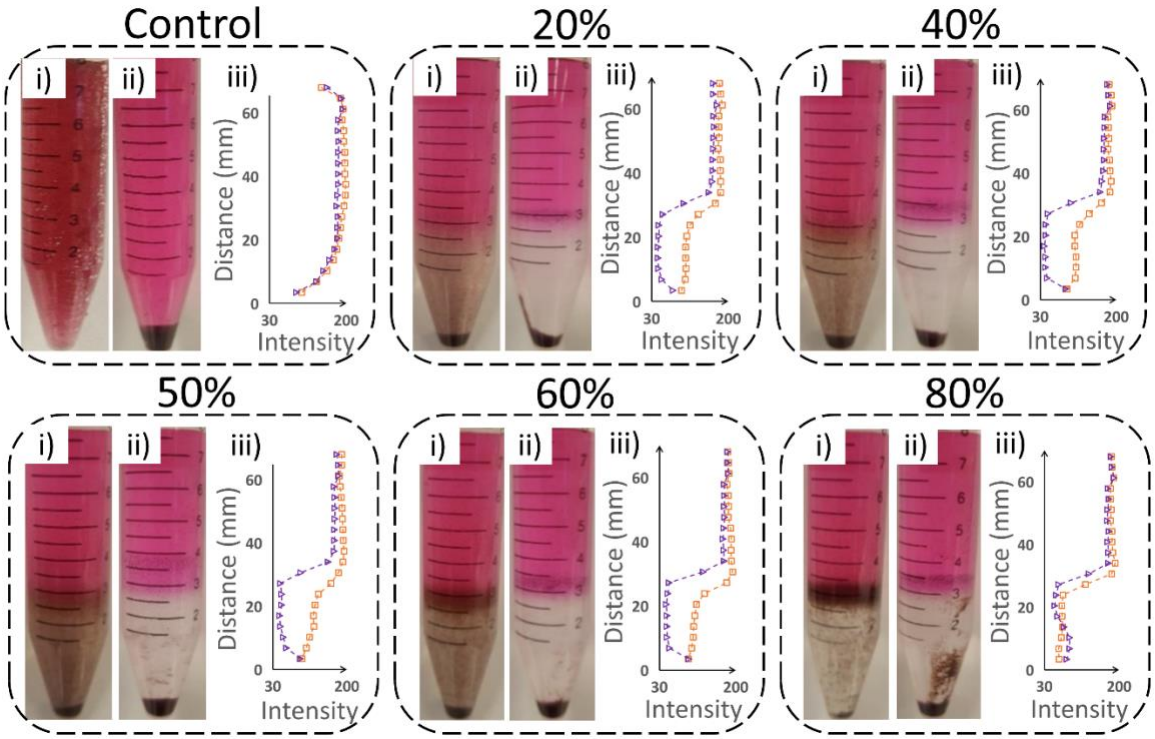


Figure 3.5 Distribution of debris in the OptiPrep™ solution before and after centrifugation. (i., ii.) Cropped images of the centrifuge tube before and after centrifugation. Iii.) The mean pixel intensity of the samples in the centrifuge tube as a function of vertical distance (in mm) was plotted for before (orange) and after (purple) centrifugation. The different OptiPrep™ solutions used were 0%, 20%, 40%, 50%, 60%, and 80% by volume. The control (i.e. 0% OptiPrep™) does not have an interface layer, and so the supernatant is treated as the top and the pellet as the bottom layer.

Compared to the control case (i.e. 0% OptiPrep™) (Fig 3.5 Control), the sample cleaning step using OptiPrep™ resulted in a marked visual separation of the debris. Before

centrifugation, the tube had a relatively consistent debris distribution (Fig 3.5 Control i.) whereas after centrifugation, all of the suspended debris were concentrated into the pellet at the bottom of the tube (Fig 3.5 Control ii.). Since the control sample did not contain the OptiPrep™ solution, there was no distinct interface layer. Here the mean intensity plots before and after centrifugation are similar because no OptiPrep™ solution was present (Fig 3.5 Control iii.).

All test tubes pictured were 15 mL tubes. With 20% OptiPrep™, there was a clear contrast between the images of the tube before (Fig 3.5 20% i.) and after centrifugation (Fig 3.5 20% ii.). The mean pixel intensity plot showed a definitive change at the interface before and after centrifugation (Fig 3.5 20% iii.). The interface layer for 40% and 50% OptiPrep™ appeared visually similar to that of the 20% concentration. However, with increasing OptiPrep™ concentrations (60% and 80%), there was a noticeably increasing amount of debris trapped at the interface layer both before and after centrifugation (Fig 3.5 60% i.-ii. and Fig 3.5 80% i.-ii.). This separation of debris at the interface layer was consistent with the mean color intensity plot (Fig 3.5 80% iii.).

Scanner-based method: Egg recovery in the interface layer

Each image of the filter paper corresponding to different OptiPrep™ solutions (20%, 40%, 50%, 60%, and 80%) was run through a custom software program, written in python, to identify the nematode eggs and give a count of the eggs within the image. The program initially takes a full image of the filter paper and then partitions it into patch images and sub-patches to identify and count the eggs. Fig 3.6a and Fig 3.6b show representative images of two filter papers having an uncleaned sample and a cleaned sample, respectively. In both cases, the program was able to differentiate stained eggs from the debris.

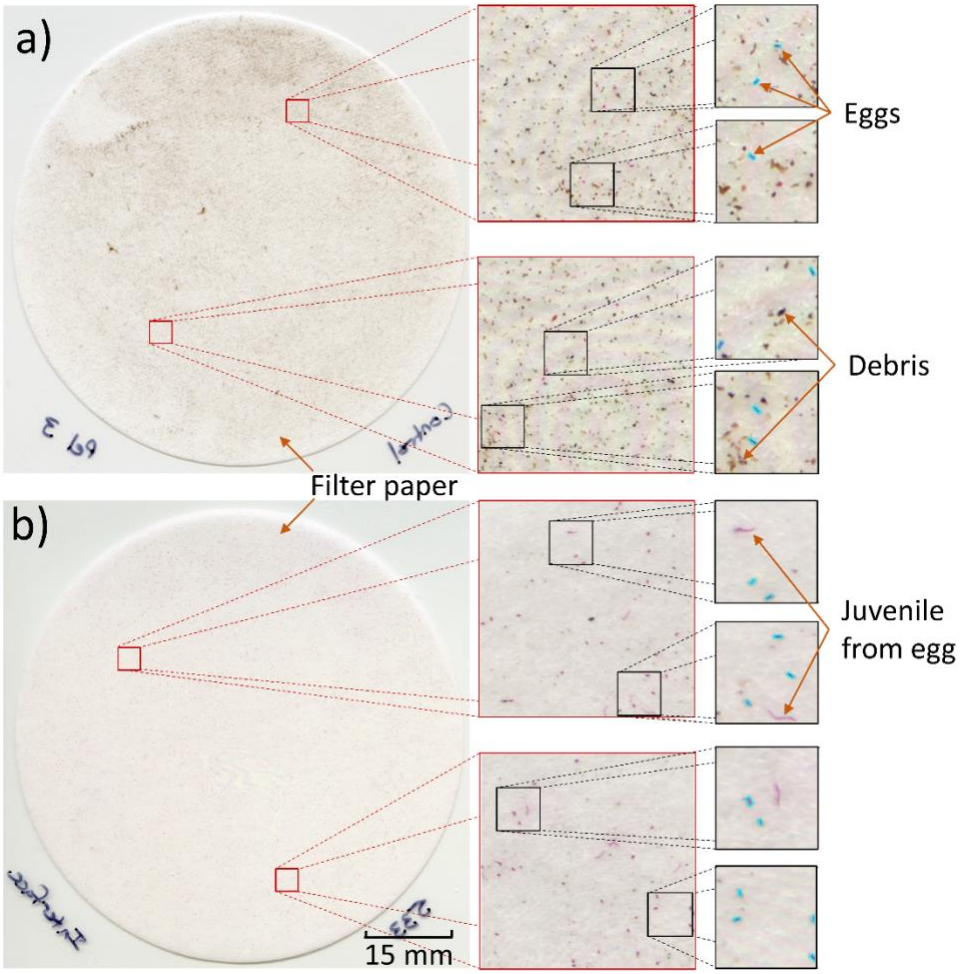
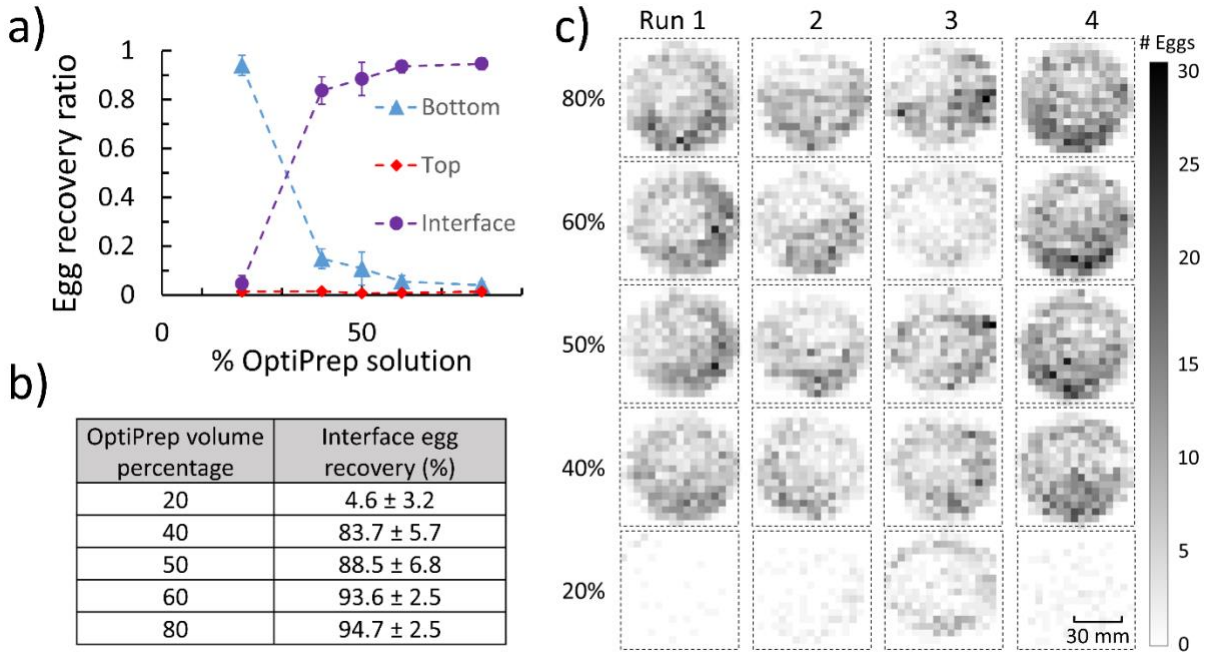


Figure 3.6 Software program to automatically detect SCN eggs among the debris in the scanner-based method. The software program starts with the full image of the filter paper and partitions it into patch images and sub-patches to identify and count the SCN eggs. a) An example of identifying eggs in a debris-laden, uncleaned sample is shown. b) An example of identifying eggs in an OptiPrep™ cleaned sample is shown where it is even possible to differentiate between stained eggs and juvenile nematodes which have recently emerged from the eggs. Scale bar = 15 mm.

For a given volume of the processed sample, the total SCN egg count was obtained by combining the egg counts for the top, interface, and bottom layers. The egg recovery ratio was calculated as a ratio of the egg count in each of the three layers to the total egg count as depicted in Fig 3.7. This was repeated for four processed samples (n = 4). The average egg recovery

ratio and standard deviation were plotted for the top, interface, and bottom layers corresponding to the different OptiPrep™ solutions.



OptiPrep volume percentage	Interface egg recovery (%)
20	4.6 ± 3.2
40	83.7 ± 5.7
50	88.5 ± 6.8
60	93.6 ± 2.5
80	94.7 ± 2.5

Figure 3.7 Sample cleaning using OptiPrep™ and SCN egg recovery from the three layers. The graph shows the ratio of SCN eggs recovered from the three layers (top, interface, and bottom layers) at different concentrations (volume percentages) of OptiPrep™. At 50% and higher OptiPrep™ concentrations, the egg recovery ratio was greater than 0.8 in the interface layer. b) The egg recovery percentage in the interface layer is listed for different percentages of OptiPrep™ by volume for four separate runs of processed samples. c) Raster plots help to visualize of the spatial distribution of the eggs on the filter papers (corresponding to the interface layer) across four extraction runs as the volume percentage of OptiPrep™ was varied. The egg count for each patch image is denoted as a dark pixel in the raster plot.

The graph in Fig 3.7a shows that the egg recovery ratio in the top layer was negligible after the OptiPrep™ sample cleaning as virtually no eggs were found in this layer. In the interface layer, the egg recovery ratio improved with increasing OptiPrep™ concentration and approached 95% for the case of the 80% concentration of OptiPrep™. Meanwhile, the egg recovery in the bottom layer decreased with increasing OptiPrep™ concentrations and approached 4% for the case of the 80% OptiPrep™ solution. For the remaining studies with

OptiPrep™, the 50% OptiPrep™ solution was used as the egg recovery at the interface layer was greater than 80% and the sample purity was the highest. Beyond an OptiPrep™ concentration of 50%, sample purity starts to drop as was demonstrated by the increase in material remaining at the interface layer after cleaning (Fig 3.5 60% ii.).

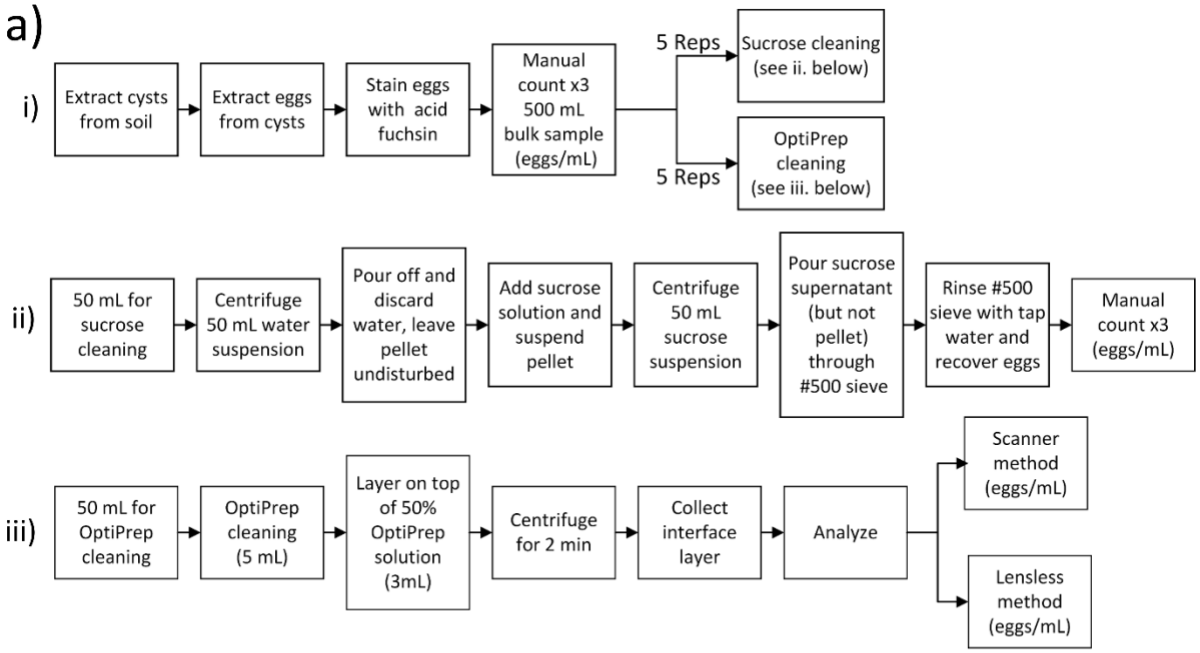
The graph in Fig 3.7a shows that most eggs were recovered from the interface layer of the centrifuge tube after sample cleaning with OptiPrep™. Fig 3.7b tabulates the average percentage of egg recovery at the interface layer for the associated volume percent OptiPrep™ solutions (n = 4). The spatial distribution of eggs in the images of the filter papers (corresponding to the interface layer) are shown as raster plots in Fig 3.7c. Four extraction runs were conducted, each with a fresh batch of stained and extracted SCN eggs and a pre-specified OptiPrep™ solution. The raster plots show the distribution of the SCN eggs (denoted as dark pixels) across the 324 sections of the images generated by the software. The raster plots show that the eggs (or dark pixels) were distributed across most of the filter paper, which helped facilitate the identification of individual eggs. Comparing the raster plots for 20% OptiPrep™ with those of higher OptiPrep™ concentrations, the raster plots appear darker as the OptiPrep™ concentration increases, suggesting that the egg recovery ratio increased in the interface layer as OptiPrep™ concentrations increased.

Performance of sample cleaning methods: OptiPrep™ versus sucrose

The OptiPrep™ sample cleaning method was compared with the sucrose centrifugation technique, which is the conventional method of separating nematodes and eggs from debris [9]. The samples were treated as described in Fig 3.8(a). A stock sucrose solution was made by dissolving 454 gm of sucrose in 1 L of tap water. Approximately 500 cm^3 of SCN infested soil was manually processed (using techniques in the 'Sample Preparation' section) to extract the SCN cysts, then eggs and debris. The samples (i.e. liquid containing eggs and debris) were

diluted to 500 mL and mixed thoroughly before obtaining the manual egg counts for this bulk sample using a nematode counting slide and dissecting microscope with a magnification of 100×.

a)



b)

Farm Location	Bulk[a] (Nematode counting slide)	Sucrose[a] (Nematode counting slide)	OptiPrep[a] (Scanner)	OptiPrep[a] (Lensless)
Muscatine County, IA	129 ± 21	55 ± 4	99 ± 3	79 ± 17
Story County, IA	41 ± 9	20 ± 2	44 ± 4	40 ± 7

[a]Counts in eggs/mL

Figure 3.8 Comparison of the sample cleaning and SCN egg counting methods.i-iii) Steps performed to extract cysts, then eggs from cysts, stain eggs, perform sample cleaning (OptiPrep™ and sucrose cleaning methods), and egg counting (manual, scanner-based, and lensless imaging). b) The table lists the average egg count (n = 5) and standard deviation for the egg count from scanner-based and lensless methods compared to the standard nematode slide count on the uncleaned bulk sample.

The bulk egg sample was stained with acid fuchsin and split into ten different 50 mL test tubes. The contents in five of these test tubes were cleaned using the OptiPrep™ cleaning method (using a 50% by volume OptiPrep™ solution). The egg counts for these five samples

were obtained using the lensless method and the scanner-based egg counting method. The egg counts were reported as the number of eggs per milliliter.

The remaining five test tubes of samples were cleaned by the sucrose centrifugation technique. Each tube was centrifuged for 3 minutes and the excess water was decanted off. The pellet was thoroughly mixed in 50 mL of the stock sucrose solution and centrifuged for one additional minute to separate the eggs from the soil pellet. The sucrose solution supernatant, containing the eggs, was poured over a 37-µm-pore sieve and rinsed in water to remove the excess sucrose. Three 1 mL sub-samples of the processed egg samples were placed on a nematode counting slide, and the SCN eggs were manually counted under a microscope and averaged to estimate the egg count per milliliter.

Soil samples were collected from two fields in Iowa – one each in Muscatine and Story Counties. Soil was collected from two different places in Iowa to get a range of soil textures with which to test the egg cleaning methods. Both methods of sample cleaning (i.e. OptiPrep™ and sucrose based) were conducted on ten different samples (five from Muscatine and five from Story). The egg counting was done by three methods – nematode counting slide, scanner-based, and lensless based. The mean and standard deviation of the egg count obtained from the different sample cleaning methods (i.e. bulk, sucrose, OptiPrep™) and counting methods (i.e. nematode slide, scanner, lensless setups) are tabulated in Fig 3.8(b).

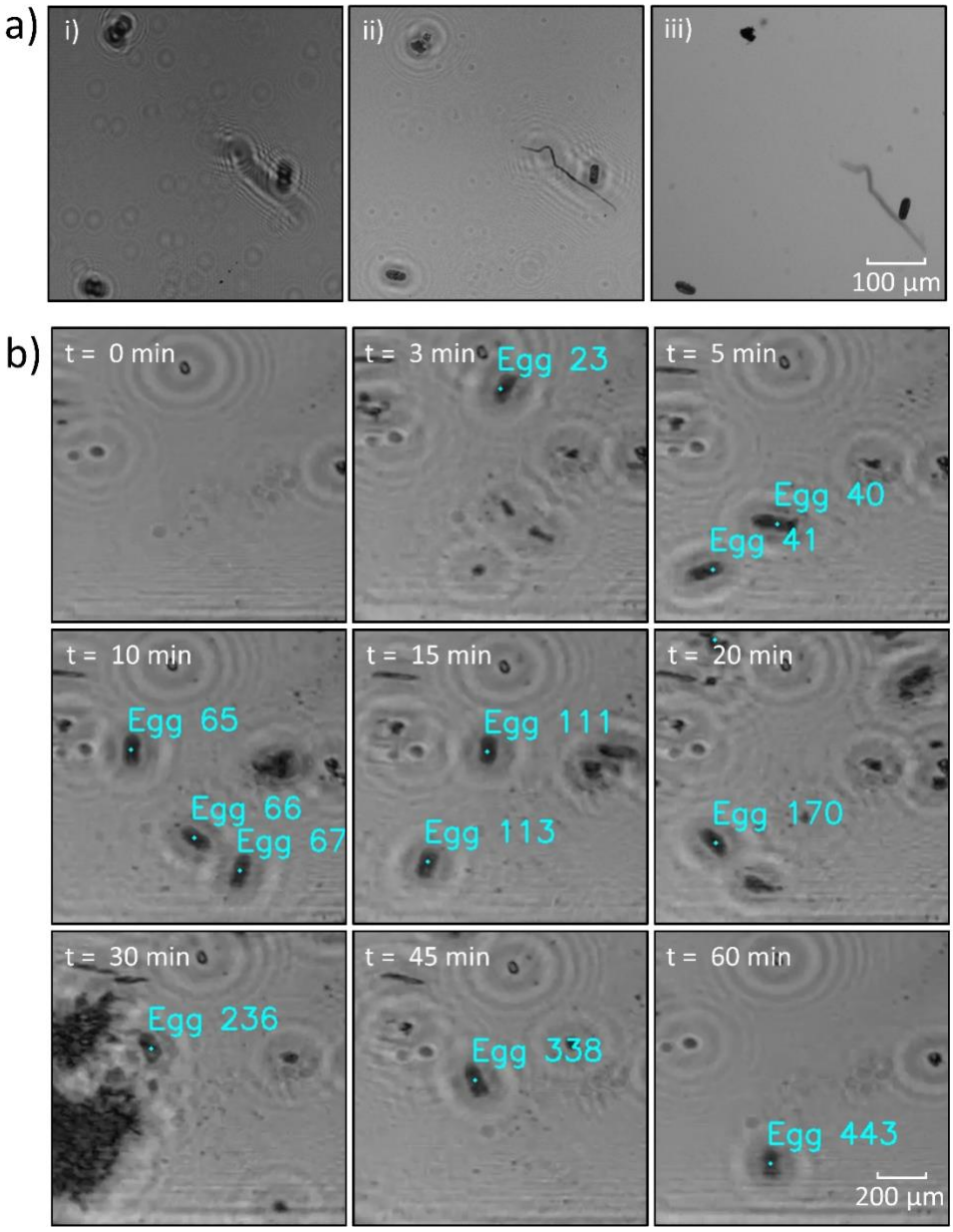


Figure 3.9 Reconstruction of the holographic videos and a time-lapsed visualization of the lensless imaging method.i) The raw image of a sample area was recorded by the lensless imaging setup (z = 8 mm). ii) The reconstructed image was produced using the Fresnel diffraction method. iii) The sample area was imaged with a stereo microscope with bright-field illumination at 50× magnification. Scale bar = 100 µm. b) The images were taken from a representative holographic video recorded by the lensless imaging setup and reconstructed thereafter. A small area of the flow channel is shown here. The eggs detected by the software were labeled with a distinct egg number. Scale bar = 200 µm.

Lensless imaging method: Reconstruction of holographic videos

After sample cleaning using 50% OptiPrep™ by volume, a 1 mL solution of the interface layer was loaded into a syringe. The dispensing needle of the syringe was connected to the input port of the microfluidic flow chip. The syringe pump was turned on at a flow rate of 1 mL per hour. As the liquid and particles began to pass through the flow channel, the RPi was turned on and the holographic video was recorded. At the end of the video recording, the video was transferred to a remote workstation and processed by a MATLAB script to generate the reconstructed video. Reconstruction of a single frame of the video took approximately 4 seconds on the CPU (Intel Xeon E5, 32GB RAM).

Fig 3.9a illustrates the enhancement in object clarity after image reconstruction. Fig 3.9a(i.) shows a section of the image captured from the lensless imaging setup (i.e. raw image). After reconstruction, the clarity and focus of the raw image was considerably enhanced, and the reconstructed image is shown in Fig 3.9a(ii.). For comparison, the bright-field image of the same area was captured by a stereo microscope at 50× magnification and is shown in Fig 3.9a(iii.).

The reconstructed video had sufficient resolution and clarity to identify the nematode eggs in every frame. Fig 3.9b shows time-lapsed image frames of a representative video that was recorded from the lensless imaging setup and reconstructed thereafter. The image frames refer to a small area of the flow channel. The eggs detected in each frame are marked in cyan color with their distinct egg number. The video was recorded at 1 frame per second and the flow rate was set to 1 mL/hour.

Performance of automated egg counting methods

We compared the accuracy of the egg counts obtained from the software program with manual egg counts as shown in Fig 3.10. In total, there were fifteen images of filter papers corresponding to the five different OptiPrep™ concentrations and their three individual layers (i.e. top, interface, and bottom). Initially, the images of filter papers were divided into 324 patch images (1024 × 1024 pixels) and saved in separate folders on the computer. To obtain the manual egg counts, a user viewed each patch image and counted the number of eggs (Fig 3.8). This process was repeated for all fifteen of the filter paper images. Next, to obtain egg counts from the software program, each patch image was further split into 256 × 256 pixel sub-patches and analyzed using the trained deep learning model. Since the deep learning model was trained for 128 × 128 pixel images, the sub-patches were resized before being passed to the learning model. Fig 3.10a shows the total egg counts calculated by the software program and by the manual method for all the filter paper images. The R^2 value between these two sets of data was high indicating that there was good agreement between the two sets of counts. We also investigated the correlation between the software and manual egg counts at a specific OptiPrep™ concentration (50%). There were a total of 972 patch images across the three layers, and the correlation between them was high as well, as shown in Fig 3.10b. This high correlation demonstrates that our software program can provide the egg count in processed samples with accuracy that is comparable to visual detection.

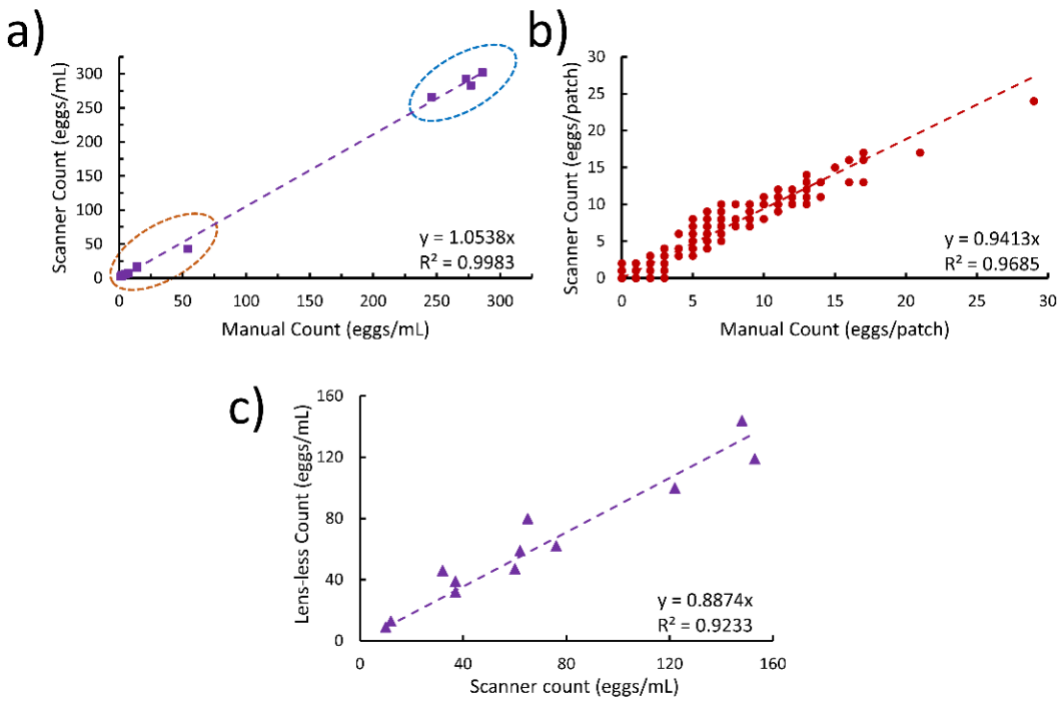


Figure 3.10 Performance of the software programs for automated SCN egg counting. The plot shows the SCN egg counts obtained for the three layers (top, interface, and bottom) for all different OptiPrep™ concentrations. The data points circled in orange belong to samples with fewer eggs. Similarly, the points circled in blue belong to samples with a large number of eggs. b) The plot shows the correlation between manual and software egg counts for all image patches of the three layers at 50% OptiPrep™ by volume. The data were collected from three filter papers, each subdivided into 324 patch images. c) The egg counts obtained from the two methods are compared: software program for lensless imaging method and the scanner-based egg counting method. Each processed sample was passed through the lensless imaging setup and further analyzed by the scanner-based method. There is high correlation between the egg counts from the two methods. The plotted data corresponds to 12 different processed samples.

To test the correlation of the software for egg counting used with the lensless imaging setup, multiple samples (n = 12) were used. The processed samples, after cleaning with 50% OptiPrep™, were passed through the lensless imaging setup. The raw holographic videos were recorded as described earlier. After reconstruction, the videos were run through the software program to identify the eggs in every frame and provide the total egg count for the entire video. The processed sample was retrieved at the outlet of the flow channel and passed through the scanner-based egg counting method. The egg counts from the lensless method and the scanner-

based method were plotted for all the samples as shown in Fig 3.10c. There was high correlation between the egg counts from both the methods with egg counts determined by direct microscopic observation.

Discussion

The sample cleaning method using OptiPrep™ has benefits over the sucrose centrifugation technique that is commonly used. A definite advantage of using OptiPrep™ is the high egg recovery achieved for the 40% to 80% OptiPrep™ solutions (greater than 80% recovery) as compared to those from the sucrose centrifugation technique (less than 40% recovery) (Fig 3.7). Another benefit of the OptiPrep™-based sample cleaning method is the ability to clean small volumes of sample solutions (5-10 mL in a 15 mL test tube) and to condense the eggs into the interface layer (2 mL) within the centrifuge tube. The eggs in the interface layer can then be counted to estimate the egg count for a much larger volume of the original sample. Furthermore, OptiPrep™ is easy to mix with water to alter the density of the liquid used for the cleaning step and it is iso-osmotic whereas sucrose is hyperosmotic, which could be detrimental to eggs for use in certain experiments [10,26]. Despite the above-mentioned benefits, the price of the OptiPrep™ medium is much higher ($250 USD for 250 mL) than sucrose, which poses a challenge for large-scale adoption of this method.

Automation in the protocols for sample cleaning and egg counting can significantly improve the consistency in results, making it less variable and less prone to human bias. Our sample cleaning steps are semi-automatic in that the sample preparation is performed manually while the software calculates the eggs present in the sample. With current technologies, it is difficult to fully automate the steps of extracting eggs from field samples and removing debris with density gradient centrifugation. However, we have developed two methods to automate the egg counting: one scanner-based and the other lensless. The scanner-based method to

obtain egg counts has advantages over the manual counting method using nematode counting slides and a microscope. Manual counting requires considerable time, expertise, and patience of a trained expert. In contrast, the scanner-based method automatically scans the filter papers and counts the eggs in the images using a trained deep learning model. Also, it is possible that cross contamination can occur in the manual counting method because of insufficient or improper cleaning of the nematode counting slide as it is reused multiple times over many samples. Cross contamination is eliminated in the scanner method as the substrate is a disposable filter paper that is not reused. The filter paper substrate is relatively inexpensive (~$0.48 USD) when compared to a nematode counting slide (~$45 USD). As a drawback, the scanner-based method requires the sample to be stained, thereby killing the eggs. Also, the processed sample is distributed on a filter paper and the counted eggs are difficult to recover and reuse after imaging. Thus, the scanner-based egg counting method is not well-suited for experiments where samples containing live eggs or nematodes are to be counted for further use, such as in greenhouse experiments.

The lensless egg counting method has several benefits over the scanner-based method. The lensless imaging setup can enumerate both unstained and stained samples, along with the option to recover the counted sample for later use after the video recording is completed. Also, the flow chip can be reused over multiple samples as passing water through the flow channel removes any debris. The imaging setup is inexpensive (less than $100 USD) relative to the cost of buying a standard microscope for manual counting of eggs. The flow chip is fabricated using non-conventional techniques of cutting and bonding double-sided tapes, which reduces the fabrication costs (less than $10). In comparison, microfluidic chips made from lithography and polymers can be expensive, especially if several chips are desired to process multiple

samples. The lensless imaging method is functionally more automated than the scanner method as a user only needs to load a syringe containing the egg sample into a syringe pump, set the right volume and speed parameters, and start the video recording device. The main disadvantage of this method is the sample processing time. We have tested the system at flow rates of 1 mL per hour. The reconstruction of the recorded holographic video can take a few hours and the egg counting software takes another 10 minutes to count the eggs. The entire reconstruction process could be sped up significantly if a GPU was used for image processing [20].

The deep learning model provides additional robustness to automated egg counting [27]. Our results show that the model can be trained to handle a variety of soil samples having different levels of debris. In our case, the model was successfully trained on samples from the interface and bottom layers collected at different OptiPrep™ volume percentages (Fig 3.5). After training, the model was robust enough to identify and count eggs in the uncleaned control sample (Fig 3.5). The egg counts obtained from the test dataset was comparable to the manual egg counts obtained from the bulk sample (Fig 3.10). The application of the deep learning model can be expanded using transfer learning to automatically identify other particles of interest in soil samples such as nematode worms, cysts, fungal spores, and weed seeds.

Conclusions

In conclusion, we developed methods to improve the efficiency of SCN egg counting by sample cleaning and imaging on scanner and lensless setups. We also created software programs to count the eggs. The use of OptiPrep™ during density gradient centrifugation helped capture most of the SCN eggs in the interface layer. The egg recovery ratio was greater than 0.8 in the interface layer for the case of 50% OptiPrep™ by volume. Then the processed samples (i.e. eggs with debris) were either placed on filter papers to record static images with

a flatbed scanner or passed through a microfluidic flow chip to record real-time, holographic videos by a lensless imaging setup. In the scanner-based method, a convolutional autoencoder network recognized the nematode eggs from static scanner images with reasonable accuracy as confirmed by visual observations. For the lensless method, the software program successfully reconstructed the holographic videos and identified the nematode eggs in the processed sample, which was subsequently confirmed by the scanner-based method. The cost of the materials was kept low by using off-the-shelf components (such as CMOS sensor, Raspberry Pi, LED), microfluidic flow chips made from inexpensive double-sided adhesive tapes, and standard filter papers. Furthermore, the use of image processing and deep learning tools circumvents the need for supporting personnel skilled at counting nematode eggs on nematode counting slides. With our new methods, manual intervention is only needed during sample preparation and loading; the software programs handle the remaining operations automatically (i.e. image or video capture, detection and counting, data storage) over a broad range of nematode egg numbers (10 – 300 eggs/mL).

References

1. Allen TW, Bradley CA, Sisson AJ, Byamukama E, Chilvers MI, Coker CM, et al. Soybean yield loss estimates due to diseases in the United States and Ontario, Canada, from 2010 to 2014. Plant Heal Prog. 2017;18: 19–27. doi:10.1094/PHP-RS-16-0066

2. Gerdemann JW. Relation of a large soil-borne spore to phycomycetous mycorrhizal infections. Mycologia. 1955;47: 619. doi:10.2307/3755574

3. Byrd DW, Barker KR, Ferris H, Nusbaum CJ, Griffin WE, Small RH, et al. Two semi-automatic elutriators for extracting nematodes and certain fungi from soil. J Nematol. 1976;8: 206–12. Available: http://www.ncbi.nlm.nih.gov/pubmed/19308224%0Ahttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC2620186

4. Niblack TL, Heinz RD, Smith GS, Donald PA. Distribution, density, and diversity of *Heterodera glycines* in Missouri. J Nematol. 1993;25: 880–6.

5. Faghihi J, Ferris JM. An efficient new device to release eggs from *Heterodera glycines*. J Nematol. 2000;32: 411–413.

6. Beeman AQ, Tylka GL. Assessing the effects of ILeVO and VOTiVO seed treatments on reproduction, hatching, motility, and root penetration of the soybean cyst nematode, *Heterodera glycines*. Plant Dis. 2018;102: 107–113. doi:10.1094/PDIS-04-17-0585-RE

7. Hajihassani A, Dandurand L-M. An improved technique for sorting developmental stages and assessing egg viability of *Globodera pallida* using high-throughput complex object parametric analyzer and sorter. Plant Dis. 2018;102: 2001–2008. doi:10.1094/PDIS-09-17-1428-RE

8. Tylka GL, Niblack TL, Walk TC, Harkins KR, Barnett L, Baker NK. Flow cytometric analysis and sorting of *Heterodera glycines* eggs. J Nematol. 1993;25: 596–602. Available: http://www.ncbi.nlm.nih.gov/pubmed/19279815

9. Jenkins WR. A rapid centrifugal-flotation technique for separating nematodes from soil. Plant Dis Report. 1964;48.

10. Deng D, Zipf A, Tilahun Y, Sharma GC, Jenkins J, Lawrence K. An improved method for the extraction of nematodes using iodixanol (OptiPrep ™). J Microbiol. 2008; 167–170.

11. Beeman AQ, Njus ZL, Pandey S, Tylka GL. Chip technologies for screening chemical and biological agents against plant-parasitic nematodes. Phytopathology. 2016;106: 1563–1571. doi:10.1094/PHYTO-06-16-0224-R

12. Tylka GL. Acid Fuchsin Stain Preparation [Internet]. 2012. Available: https://www.plantpath.iastate.edu/tylkalab/content/acid-fuchsin-stain-preparation

13. XU Z, Cheng XE. Zebrafish tracking using convolutional neural networks. Sci Rep. Nature Publishing Group; 2017;7: 42815. doi:10.1038/srep42815

14. Masci J, Meier U, Cireşan D, Schmidhuber J. Stacked convolutional auto-encoders for hierarchical feature extraction. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2011. pp. 52–59. doi:10.1007/978-3-642-21735-7_7

15. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics). 2015;9351: 234–241. doi:10.1007/978-3-319-24574-4_28

16. Kingma DP, Ba J. Adam: A method for stochastic optimization. Proc 12th Annu Conf Genet Evol Comput - GECCO '10. 2014; 103. doi:10.1145/1830483.1830503

17. Chollet François. Keras: The python deep learning library. keras.io. 2015. doi:10.1086/316861

18. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, et al. TensorFlow: A system for large-scale machine learning. 12th USENIX Symposium on Operating Systems Design and Implementation. 2016. pp. 265–283.

19. Structural analysis and shape descriptors [Internet]. 2018. Available https://docs.opencv.org/2.4/modules/imgproc/doc/structural_analysis_and_shape_descriptors.html

20. Mudanyali O, Tseng D, Oh C, Isikman SO, Sencan I, Bishara W, et al. Compact, light-weight and cost-effective microscope based on lensless incoherent holography for telemedicine applications. Lab Chip. 2010;10: 1417. doi:10.1039/c000453g

21. Silvanmelchior. RPi_Cam_Web_Interface [Internet]. github.com; 2014. Available: https://github.com/silvanmelchior/RPi_Cam_Web_Interface

22. Isikman SO, Greenbaum A, Lee M, Bishara W, Mudanyali O. Modern trends in imaging viii: lensfree computational microscopy tools for cell and tissue imaging at the point-of-care and in low-resource settings. Anal Cell Pathol. 2012;35: 229–247. doi:10.3233/ACP-2012-0057

23. Dovhaliuk RY. Review of digital holography reconstruction methods. In: Angelsky O V., editor. Thirteenth International Conference on Correlation Optics. SPIE; 2018. p. 5. doi:10.1117/12.2300759

24. Bao P, Situ G, Pedrini G, Osten W. Lensless phase microscopy using phase retrieval with multiple illumination wavelengths. Appl Opt. 2012;51: 5486–94. doi:10.1364/AO.51.005486

25. Zuo C, Sun J, Zhang J, Hu Y, Chen Q. Lensless phase microscopy and diffraction tomography with multi-angle and multi-wavelength illuminations using a LED matrix. Opt Express. 2015;23: 14314. doi:10.1364/OE.23.014314

26. Van Bezooijen J. Methods and techniques for nematology. Wageningen University; 2006.

27. Akintayo A, Tylka GL, Singh AK, Ganapathysubramanian B, Singh A, Sarkar S. A deep learning framework to discern and count microscopic nematode eggs. Sci Rep. 2018;8: 9145. doi:10.1038/s41598-018-27272-w

CHAPTER 4.

BRUGIA MALAYI: AUTOMATED TRACKING AND ANALYSIS OF BEHAVIORAL PHENOTYPES

Abstract

Lymphatic filariasis is a neglected tropical disease caused by parasitic nematodes including *Brugia malayi*. The adult parasites dwell in the lymphatic vessels and lymph nodes and can stay for several years obstructing the fluid flow and releasing microfilaria into the blood stream. This causes swelling or lymphedema and can eventually lead to elephantiasis or hydrocele. As such there is a need to find potential drug candidates against these parasites (adult and microfilaria stages) which requires a much closer inspection of their phenotypic behavior. The current drug screening techniques for different stages of *B. malayi* are either laborious or involve extracting only a couple of features that do not capture the entire behavioral spectrum of these parasites. We have developed a worm tracker package that consists of two modules (BrugiaAdultTracker and BrugiaMFTracker) to characterize the phenotypical behavior of both adult and microfilaria parasites. The first module extracted six key features (centroid velocity, angular velocity, rate of Euler number, rate of eccentricity, rate of extent, and worm path curvature) from the parasite videos exposed to three common drugs and the dose response curves were plotted. Four (centroid velocity, angular velocity, rate of extent, and rate of eccentricity) of the six features showed similar *IC50* values and are in accordance with the *IC50* values published in the literature. The second module extracted the whole-body information from the parasites and built a model using which it computed the body posture and movement features. The software package is simple to use, is robust, requires minimal user intervention, saves data for cross validation, and provides additional insights into the phenotypical responses.

Introduction

Lymphatic filariasis (elephantiasis) is a neglected tropical disease known to affect several millions of people worldwide [1]. This disease is caused by the following parasitic nematodes: *Wuchereria bancrofti*, *Brugia malayi*, and *Brugia timori* [1–6]. The parasite lives within the mosquito up to the third larval stage. After the mosquito bites the human skin, the nematode enters the human body [2,5,7]. Thereafter, these parasites progress to the lymphatic vessels where they mature into the adult stage and release sheathed microfilaria into the blood stream [7]. The microfilaria can be further ingested by other mosquitoes feeding on the same human host and thus act as new transmission vectors [5,7]. As the disease progresses, the host shows symptoms such as an altered immune system and lymphatic dilation (swelling), leading to pain and physical disability [2].

Current anti-filarial treatments (i.e. mass drug administration) to combat lymphatic filariasis include drugs such as diethylcarbamazine (DEC), albendazole, and ivermectin [1,3,8,9]. Combinations of two or more drugs administered at different doses and different time points have shown to effectively eliminate microfilaria from the human body. For instance, drugs (DEC and albendazole) administered together were effective in eliminating microfilaria after multiple doses, while administering three drugs (DEC, albendazole, and ivermectin) together in a single dose made the patients free of microfilaria entirely for two years [10,11]. However, these drugs were ineffective in killing the adult parasites dwelling in the lymphatic vessels [2,3,8]. Recent studies have shown that another drug, doxycycline, can target *Wolbachia* bacteria inside the adult filarial nematodes and thus sterilizing and kill them [1,7,8,10]. Other drugs (e.g. nitazoxanide, tizoxanide, and auranofin) also have been recognized as potential macrofilaricides through *in vitro* screening [3,8]. While treatments are available that eliminate microfilaria, there is still no available filarial nematodes that would be

effective within the human body. This necessitates the need to identify new potential treatments that are efficient, safe, and viable using high throughput drug screening methods.

The progression of lymphatic filariasis within the human host is attributed to adult parasitic nematodes and microfilaria [7]. Therefore, understanding their behavior in different drug compounds is an essential step towards fighting the disease. In conventional drug screening methods, the parasites or their microfilaria are assayed in well plates in different drug concentrations and incubated for 24-72 hours prior to imaging [2,12]. The effectiveness of the drugs is often analyzed by visual observation under a microscope or using a worm tracking software [10,13]. Typical software analysis methods estimate the centroid speed of the parasites from recorded videos [7,13]. There are very limited number of worm tracking software applied to these parasites because of inherent problems in automatically characterizing behavioral parameters without user intervention [3].

A number of worm tracking software have been developed for *Caenorhabditis elegans*, a widely used model organism, to quantify its behavior under various stimuli and movement [14-16]. The worm tracking software generally extract information about the centroid or the body skeleton [14–20]. In centroid-based trackers, the centroid position of a worm is identified and metrics such as centroid velocity and thrashing are calculated from videos recorded at the population level [21–24]. In contrast, skeleton-based trackers extract the posture of the whole body from videos of individual worms which can be used to describe a number of phenotypes [19,25–30]. These systems record the worms at higher magnifications [21,29]. In general, these *C. elegans* worm trackers are developed on known estimates of the worm length and nature of movement. Because of these assumptions, the *C. elegans* worm trackers are too simplistic to handle the complex postures and multiple omega (Ω) turns made by *B. malayi*.

Here, we developed a worm tracking software package to characterize the *B. malayi* adult parasites and microfilaria. The software package consists of two modules. The first module (BrugiaAdultTracker) was designed to characterize the phenotypic behavior of adult *B. malayi* by computing a number of features related to its three-dimensional movement. We validated the module on adult male worms using three drugs (albendazole, ivermectin, and fenbendazole). The following features regarding worm movement were extracted automatically by the software module: centroid velocity, angular velocity, rate of Euler number, rate of eccentricity, rate of extent, and worm path curvature. Based upon these features, dose response curves were plotted to understand the relevance of each feature in estimating the drug efficacy. The second software module (BrugiaMFTracker) was developed to extract the information regarding the body skeleton of *B. malayi* microfilaria. The software module was used to extract microfilaria features such as body curvature, number of bends, directed motion, and point velocities on the body. The combination of both the software modules provides an automated method to track *B. malayi* parasites using a unique set of features which can be used to test the potency of anti-filarial for drug screening

Materials and methods

Assay preparation

All the chemicals were purchased from Sigma Aldrich Inc. (St. Louis, MO) and product identification codes (#) are mentioned below. Adult *Brugia malayi* male worms (TRS Labs Inc., Atlanta, GA) were assayed in RPMI-1640 (25 mM HEPES, 2 g/L $NaHCO_3$) (*#R5886*) supplemented with 1% (v/v) Antibiotic/Antimycotic (*#A5955*). 30 mM and 10 mM stock solutions of fenbendazole (*#35032*) and ivermectin (*#I8898*) were prepared in dimethyl sulfoxide (DMSO) (*#D2650*). Similarly, two stock solutions, 30 mM and 100 mM, of albendazole (*#A4673*) were prepared. The working concentrations of fenbendazole (3 μM, 10

μM, 30 μM, 100 μM, and 300 μM), ivermectin (0.03 μM, 0.3 μM, 1 μM, 3 μM, 5 μM, and 10 μM), and albendazole (30 μM, 100 μM, 300 μM, and 1000 μM) were serially diluted in RPMI media from stock solutions. A solution containing 1% DMSO in RPMI media was used as a negative control. The worms were transferred to these solutions and maintained in a 37ºC, 5% CO_2 incubator for 24 hours prior to experimentation. All the concentrations and the control experiments were run in triplicate. Dose response analysis and IC_{50} values were calculated using variable non-linear regression model in GraphPad® Prism 7 and Excel®.

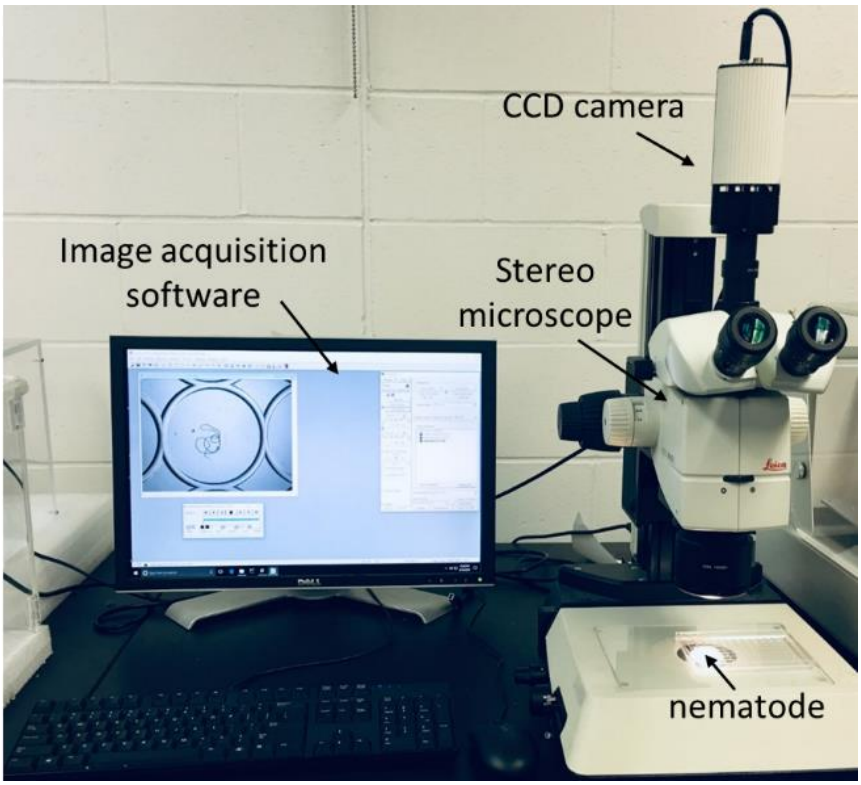


Figure 4.1 The experimental setup for the video acquisition of *Brugia malayi* adult male worms. A *B. malayi* adult male worm placed in the well plate lid was recorded using image acquisition software, a CCD camera, and a microscope. The acquisition software was set to record videos with a 696 × 520 pixel resolution at 5 frames per second for four minutes.

Image acquisition

The hardware components for the image acquisition setup included a stereo microscope (M205C, Leica Microsystems Inc.), a desktop computer (Intel Xeon E3, 8 gigabytes of

random-access memory, and 500 gigabytes of storage), and a microscope camera (QiCAM, QImaging Inc.) as shown in Fig 4.1. MATLAB® 2016a along with its Image Processing toolbox was installed on the desktop computer. QCapture Pro® (QImaging Inc.) was used for the recording videos.

For the *B. malayi* adult male experiments, the cover lids of 96-well plate (Corning Inc.) were used as they were found to have optimal dimensions and field of view for our imaging experiments. The worms were transferred from the incubator to individual wells with cover slips on top. Videos were recorded for individual worms at 8x magnification with a 696 × 520 pixel resolution.

B. malayi adult tracking

The BrugiaAdultTracker software module was developed to analyze videos of adult male *B. malayi*. The module extracts worm features (such as centroid velocity, angular velocity, rate of Euler number, rate of eccentricity, rate of extent, and worm path curvature), and records the data into Excel spreadsheets. In addition, the software module also records real-time positions of the worm centroid which can then be overlaid on the original video for visual inspection.

Image processing and feature extraction

A flowchart describing the steps involved in the feature extraction process used by the BrugiaAdultTracker software is shown in Fig 4.2a. The software reads the initial frame of a video as shown in Fig 4.2b (i). The well boundary was identified from this frame using the Circle Hough Transform (CHT) algorithm (Fig 4.2b (ii)). A mask was generated from this boundary to isolate the well area from the image as shown in Fig 4.2b (iii). In the event that the software was unable to detect the well, a fallback option that lets the user manually select the well area was provided.

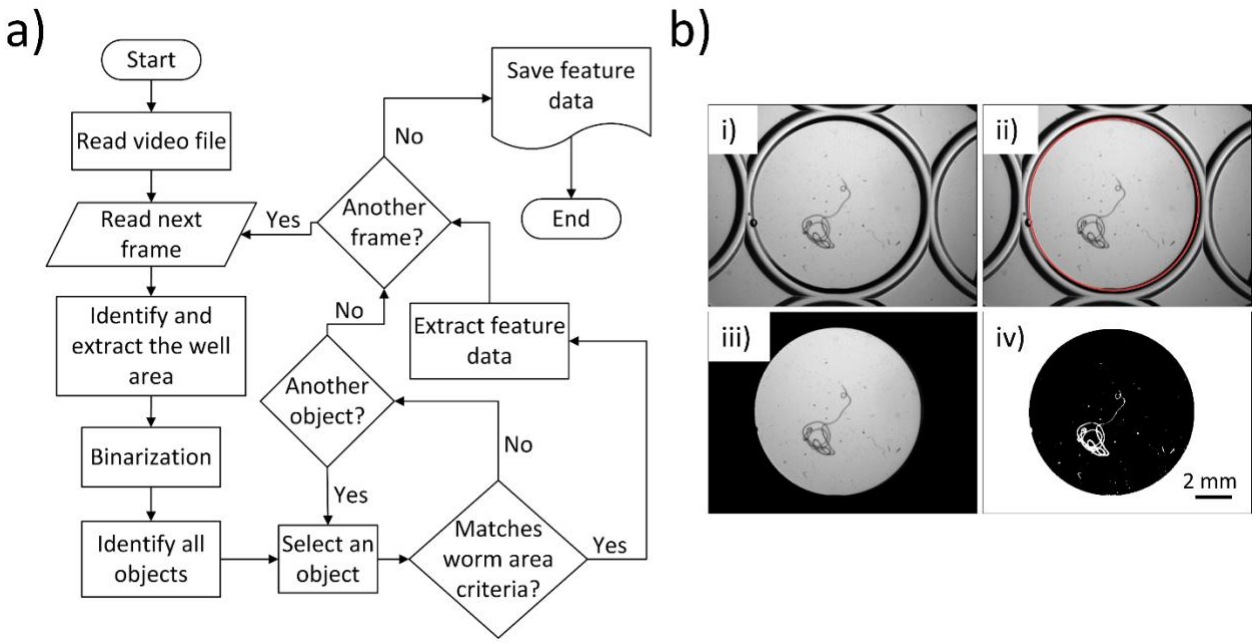


Figure 4.2 Feature extraction process of the BrugiaAdultTracker software. a) A flowchart illustrating the series of steps performed by the BrugiaAdultTracker software to extract and save multiple features from the worm videos. b) Illustration of the sequence of steps performed to identify a worm from a sample video frame. i) The raw frame read by the software from a video. ii) The well containing the worm was identified using the circular Hough transform algorithm and labelled in a red color. iii) A mask was created from the identified well removing other wells and noise. iv) A percent thresholding technique was performed to binarize the image where identified blobs were labelled with white.

The software processes each frame sequentially extracting the feature data. Each frame was binarized using a percent thresholding technique where the average pixel intensity of the frame is multiplied by a set percentage as shown in Fig 4.2b (iv). This method accounts for different light conditions better than setting a fixed threshold value. The binary image was filtered using blob area to isolate the worm from any other objects present. The worm position and blob feature data were extracted and recorded for the frame.

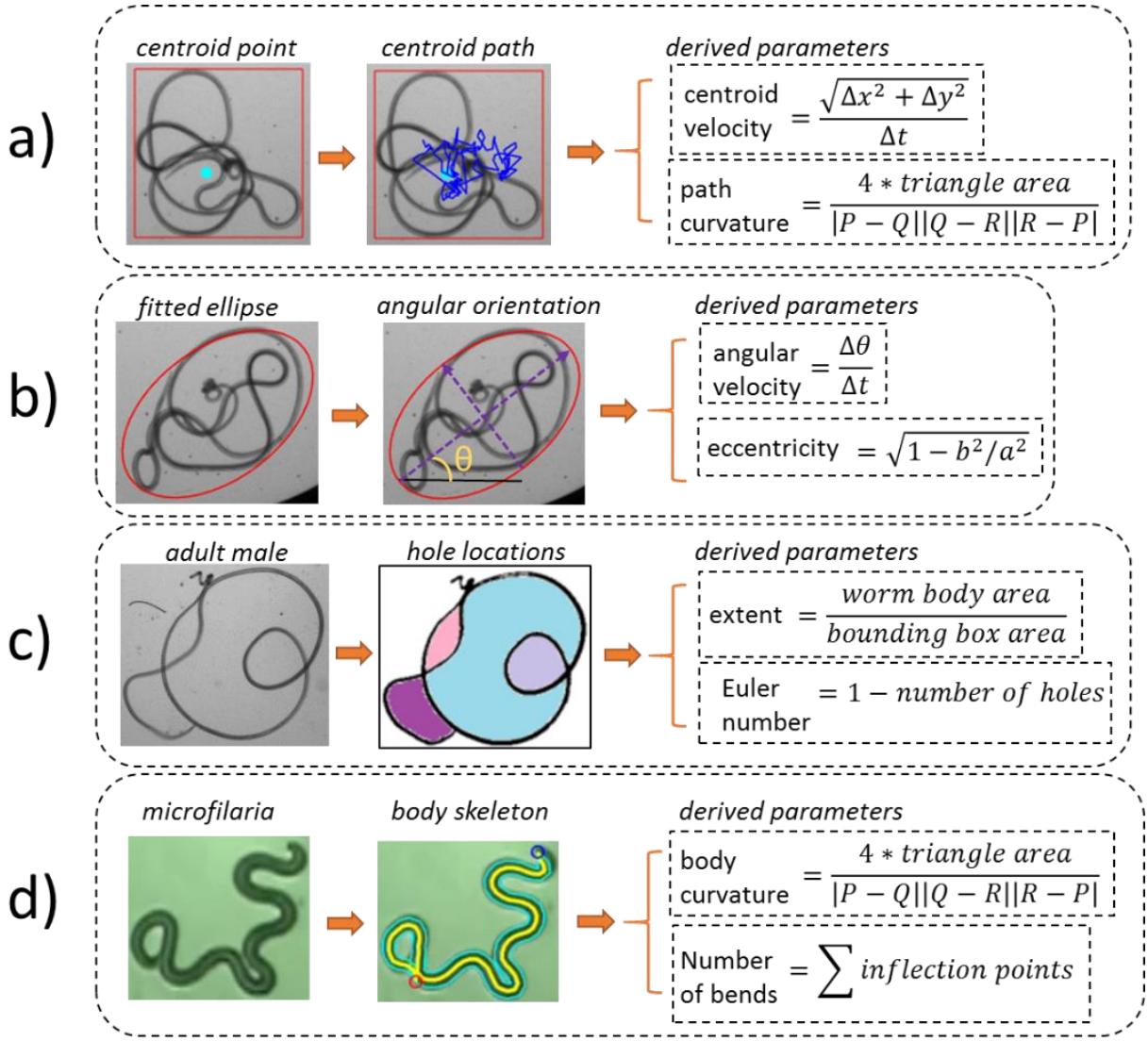


Figure 4.3 Explanation of the methods and extracted features used to analyze *B. malayi* adult male and microfilaria. a) The weighted centroid of the worm was determined from which the centroid velocity and path curvature were calculated. b) A minimum ellipse was fit to the worm and major and minor axes lengths were extracted along with its angular orientation. These metrics were used to calculate angular velocity and eccentricity. c) The extent was calculated from the ratio of the worm area to the area of the minimum bounding rectangle fit to the worm for the adult worm. The number of loops formed by the worm was estimated by the Euler number which is related to the number of holes in worm object. d) The points in the skeletal curve generated for the worm by the BrugiaMFTracker were used to calculate body curvature and the number of bends.

The extracted data included several different properties (centroid, angular orientation, Euler number, and extent). The centroid position was extracted by calculating the weighted center of the blob's minimum bounding rectangle, a close approximation of the worm centroid.

The Euclidean distance between the centroid positions in two subsequent frames was calculated and converted from pixels to millimeters. The 'centroid velocity' was calculated as the ratio of this distance and frame interval, 0.2 seconds. The 'path curvature' was determined by the Menger curvature at the centroid position as shown in Fig 4.3a. An ellipse was fitted to the worm blob to extract the angular orientation (θ) and major (a) and minor (b) axes lengths. The 'angular velocity' was calculated as the ratio of difference between angular orientations in subsequent frames and the frame interval (Fig 4.3b). The eccentricity of an ellipse was calculated as shown in Fig 3b. The extent was defined as the ratio of worm blob area to the bounding rectangle area (Fig 4.3c). The Euler number was defined as the difference of the number of blobs and number of holes as shown in Fig 4.3c. The features 'rate of eccentricity', 'rate of extent', and 'rate of Euler number' were extracted by calculating the ratio of the difference of the corresponding values from subsequent frames and frame interval. All features were saved to Excel spreadsheets for data visualization and analysis.

***B. malayi* microfilaria tracking**

The BrugiaMFTracker software was developed to read and process *B. malayi* microfilaria videos taken at 40x magnification and build a worm model. The skeletal points were used to extract features such as body curvature, number of bends, directed motion, and velocities of head, tail, and centroid.

Worm model

To build a model for *B. malayi*, the motion of the nematode can be approximated in two-dimensional space and the change in nematode length while stretching and contracting is ignored. The midline of the nematode was modeled as a cubic spline curve, $M(u)$, with an associated angle vector $\Theta(u)$. The tangent (e_1) and normal (e_2) vectors of the midline were defined as shown in Eq. (4.1) [20]:

$$e_1 = \begin{bmatrix} cos(\Theta(u)) \\ sin(\Theta(u) \end{bmatrix} \qquad e_2 = \begin{bmatrix} -sin(\Theta(u)) \\ cos(\Theta(u) \end{bmatrix} \tag{4.1}$$

The boundary of the worm (W) was constructed around the midline curve using Eq. (4.2) [20]:

$$W(u,v) = \lambda\left(\int_0^u e_1(u)du + vR(u)e_2(u)\right) + \vec{T}$$
$$u \in \left[-\frac{L}{2}, \frac{L}{2}\right] \quad v \in [-1,1] \tag{4.2}$$

where λ is the scale factor with units of $\frac{pixels}{mm}$, $\vec{T}$ is the translational vector, L is the worm's length. The vector, $R(u)$, contains the distances from midline to the worm's boundary along its length. The scale factor was calculated from the microscope magnification. The worm's length was dynamically determined after the software processed the first frame.

Image processing and feature extraction

The flowchart depicting the video processing steps was shown in Fig 4.4. The software read the RGB frames sequentially and converted them to grayscale. A mean pixel intensity value was calculated for the grayscale image and a percent thresholding technique was used to binarize the image. The worm object was identified from the resulting binary image by identifying the largest object and its boundary was extracted and smoothed with a moving average filter for two iterations.

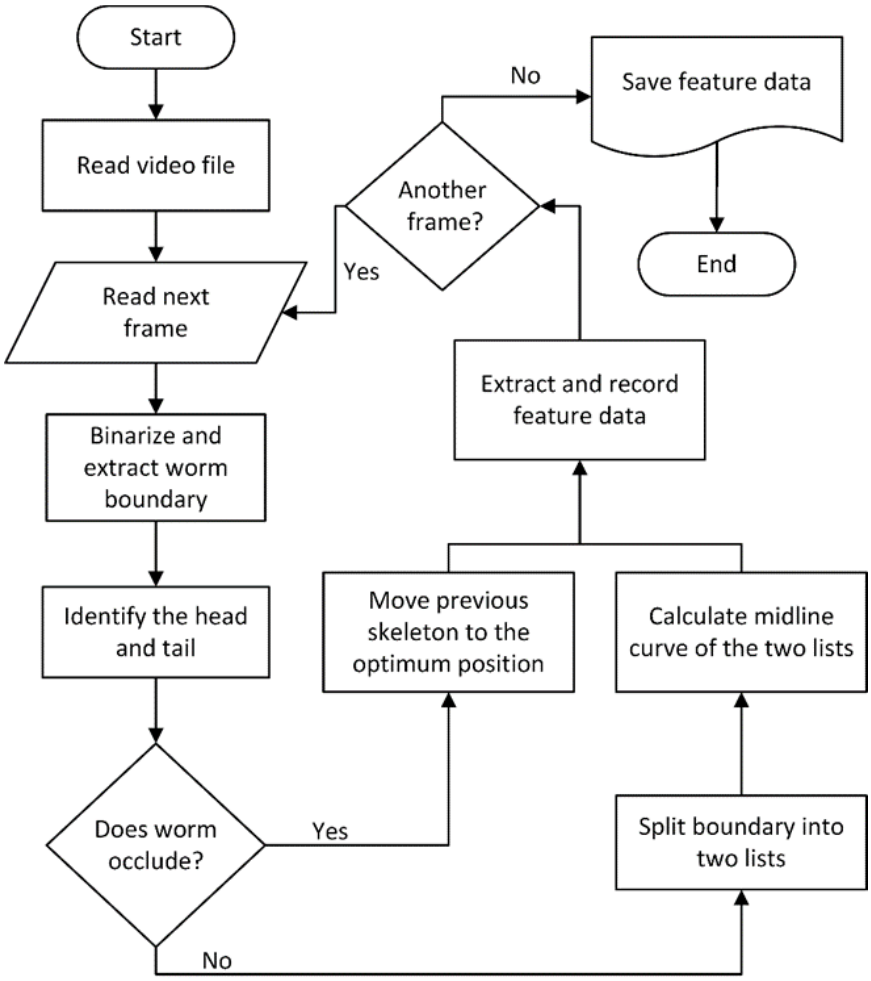


Figure 4.4 A flowchart describing the processing steps taken by the BrugiaMFTracker software to extract features from *B. malayi* microfilaria.

The next step was to determine the head and tail positions from the worm boundary. Sharper bends correspond to higher negative curvature values which occur at the head and tail locations. The curvature at each point in the boundary list was calculated using the Menger curvature algorithm, which takes three points and calculates the inverse of radius of the circumcircle of the points. The function to calculate the curvature (C) for three points (a, b, c) is shown in Eq. (4.4) [31]:

$$A = (b.x - a.x) * (c.y - a.y) - (b.y - a.y) * (c.x - a.x) \quad (4.3)$$

$$C(a,b,c) = \frac{4A}{|\,a-b||b-c||c-a|} \tag{4.4}$$

where $a, b, c \subseteq \mathrm{R}^{\mathrm{n}}$ and A in Eq. (4.3) is the signed area of the triangle formed by the three points. The average value of the curvature in the array was calculated and the points with values lower than two standard deviations of the average value were isolated and clustered into two groups by k-means clustering algorithm [32]. Clustering avoids faulty identification due to the points adjacent to the head and tail with high curvature values. The points corresponding to the lowest curvature values from the two clusters were identified and assigned as head and tail. The head and tail positions determined by the software were marked and displayed to the user for validation before processing the entire video. The boundary was smoothed further and processed again if the user deemed the positions inaccurate.

The software checks for worm occlusions as the occlusion scenario was handled differently. If the worm doesn't occlude, using the head and tail information, the boundary points were separated into dorsal and ventral lists respectively. Depending on the worm orientation and the occurrence of occlusions, the lists can be of different sizes. The shorter of the dorsal and ventral lists was determined and inward facing normal vectors were calculated for each point in that list. The points in the longer list intersecting these vectors were determined using Eq. (4.5). An approximate centerline $M(i)$ was constructed by calculating the midpoints of the points in the two lists as shown in Eq. (4.6). The final skeletal curve and points were generated by fitting a cubic B-spline curve. Accordingly, worm model was created from the width profile and skeletal curve.

$$f(i) = \underset{x\,\epsilon\,[0,S_2)}{\operatorname{argmax}} \left| side_{short,i} - side_{long,x} \right| \tag{4.5}$$

$$M(i) = \frac{side_{long,f(i)} + side_{short,i}}{2} \qquad i \in [0, S_1) \tag{4.6}$$

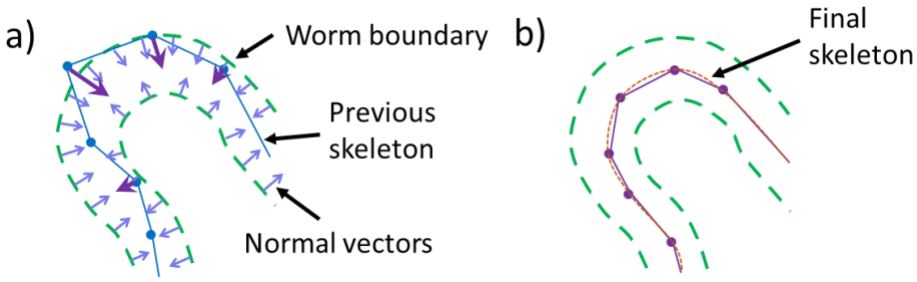


Figure 4.5 A cartoon depicting the process to determine the worm skeleton from the previous skeleton and worm boundary information in the event of an occlusion. a) Inward facing normal vectors (light blue arrows) were generated for the worm boundary (green) of the current frame. Closest points in the worm boundary to the previous skeleton (blue) were determined and displacement vectors (purple) were calculated. b) The previous skeletal points were moved in accordance with the corresponding displacement vectors to new positions (purple). B-spline curve was fit to these points to create the final skeleton (red).

If the worm has occlusions, there is a loss of border information surrounding the occluded area. To resolve this problem, the software uses the skeletal curve from the previous frame. The normal vectors facing inward for each point in the boundary list were calculated. Fig 4.5a shows a portion of the worm boundary in the current frame overlaid with the previous skeletal curve and the inward facing normal vectors of the boundary points. Using Eq. (4.5), the points in the skeleton intersecting these vectors were determined. The distances between the boundary points and the skeletal points were then calculated. If the distances were not the same as determined by the width profile of the worm model (purple arrows in Fig 4.5a), the skeletal points were then moved in the direction of the normal vectors of the corresponding boundary points until their distances matched their widths. This allowed the skeletal points of the previous frame that were outside of the body, to be migrated back into the body as shown in Fig 4.5b. When an occlusion occurs parallel to the skeletal curve, there is a chance of overlap of skeletal points. To prevent this, the software identifies the overlapped points by calculating

the distances from each point to every other point in the skeleton curve. The software moves the points in opposite directions until they are within the tolerance limits. The final skeleton curve (orange color in Fig 4.5b) was formed by fitting a B-spline curve on the skeletal points (purple color in Fig 4.5b).

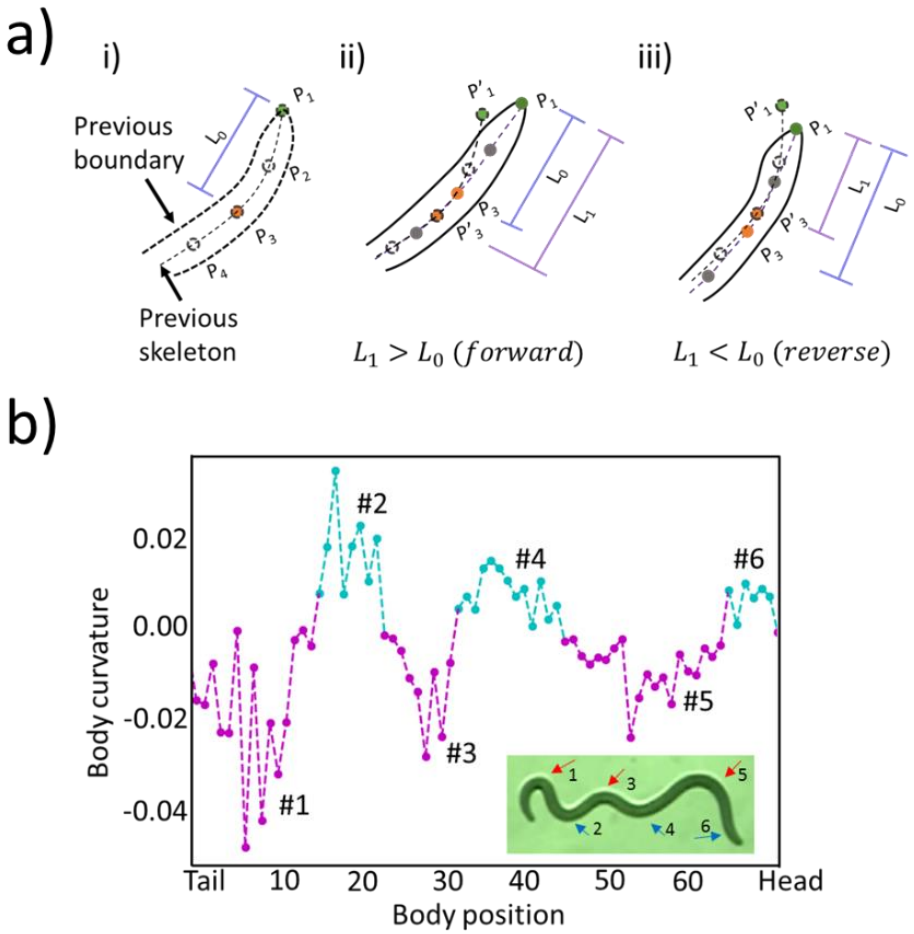


Figure 4.6 Illustration of the methods to extract features from *B. malayi* microfilaria worms. a) A cartoon explaining the method to determine forward and reverse motion at the head portion of the worm. i) Shows the head portion of the worm in the previous frame with P_1 as the head and distance L_0 as the Euclidean distance between P_1 and P_3. ii-iii) The distance L_1 was calculated between the head of the current frame and third point of the previous frame (P'_3). The worm was determined to be in forward motion if this distance is greater than the L_0 of the current frame and in reverse motion for the opposite case. b) Plot showing the curvature at each skeletal point of the worm along with the bends identified. The respective worm image is also shown for reference highlighting the different bends. The points with positive curvature values are marked in cyan color and points with negative values are marked in pink color.

Feature analysis metrics

Directed motion

To determine the direction of the worm motion, the head and tail portions of the worm were considered. Fig 4.6a (i) shows the head portion of the worm from the previous frame. In the current frame, the distance between the third point (P_3) and the head (P_1) or tail was calculated (L_0). Similarly, the distance between the third point of the previous frame (P_3') and the current head (P_1) or tail was calculated (L_1). If this distance was smaller in the current frame than in the previous frame ($L_1 > L_0$), then the worm was considered to be moving forward (Fig 4.6a (ii)). If the distance was larger ($L_1 < L_0$), then the worm was reversing its motion (Fig 4.6a (iii)). In the tail portion of the worm, these criteria were reversed.

Number of bends

B. malayi and other large nematodes can form more bends when compared to shorter nematodes. This allows them to form complicated shapes. The number of bends was quantified by calculating the curvature values of each skeletal point. The 'body curvature' of the nematode was determined by calculating the signed curvature values of each point in the skeleton. The curvature values of a single frame of a video were calculated and plotted as shown in Fig 4.6b along with the corresponding video frame. The positive curvature values were marked in cyan and negative values in magenta colors. A portion of the skeleton was treated as a bend if there was a sign change in the curvature values in the hysteresis window (n=3). The number of points for this window was determined empirically. Fig 4.6b shows the labelled bends on the curvature plot and the corresponding worm image.

Results

B. malayi adult assay

The worms were assayed in different concentrations of three drugs (fenbendazole, ivermectin, and albendazole) known to affect these parasites [1,3,8,9]. For each drug concentration, the worm behavior was recorded for 4 minutes at 5 frames per second. This time duration was sufficiently long to observe the drug-induced worm behavior [2]. Fig 4.7 shows sample video frames at different time intervals for each representative worm exposed to a different drug concentration. Over the course of the video, the worms exhibited complex postures under minimal drug concentrations similar to when they were in negative control (Fig 4.7). However, as the drug concentration was increased, the worms showed minimal to no movements suggesting the degradation of worm's physical ability.

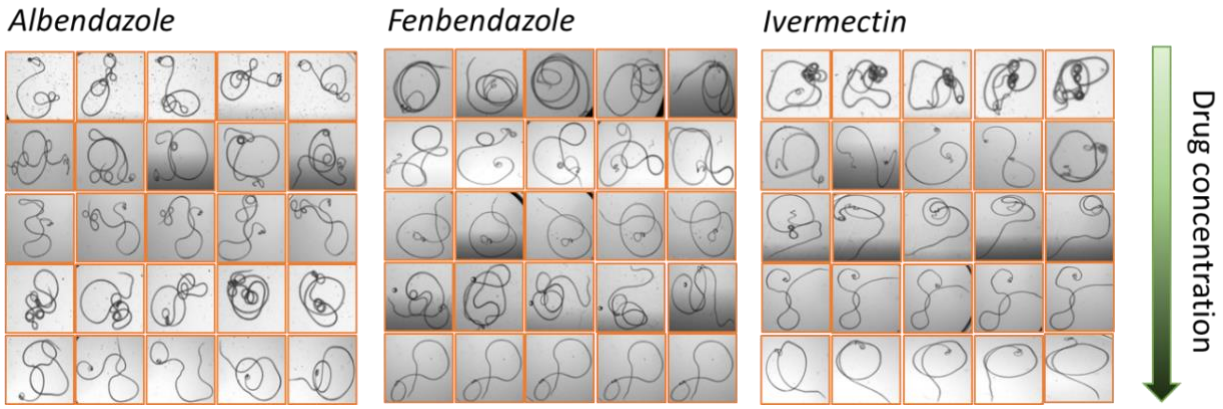


Figure 4.7 Demonstration of the shapes made by the *B. malayi* adult worms at different drug concentrations. Three anthelmintics (albendazole, fenbendazole, and ivermectin) were tested. Each row represents the worms exposed to a specific drug concentration. The drug concentration increases from top to bottom in logarithmic scale. Sample frames were extracted at random from a representative video of each drug concentration experiment.

The feature data (centroid velocity, angular velocity, rate of Euler number, rate of eccentricity, rate of extent, and worm path curvature) from all the videos were extracted for investigation and shown as raster plots in Fig 4.8. The experiments corresponding to different drug concentrations were marked and labelled on the plots. It can be seen from the figures that

as the drug concentrations were increased, the feature values moved to the lower end of the color spectrum affirming the visual analysis (Fig 4.7). It can also be seen that the negative control values for different features were higher when compared to the drug concentrations for all but the path curvature values. This was because the worm paths exhibit high levels of sharpness at higher concentrations of drugs [16] due to their minimal movement. It is worth mentioning that in some of the control and lower drug concentration runs, the nematodes barely showed any movement contrast to what was expected indicating higher experimental variability. Similarly, in some experiments of ivermectin and albendazole concentrations, the feature values were very low.

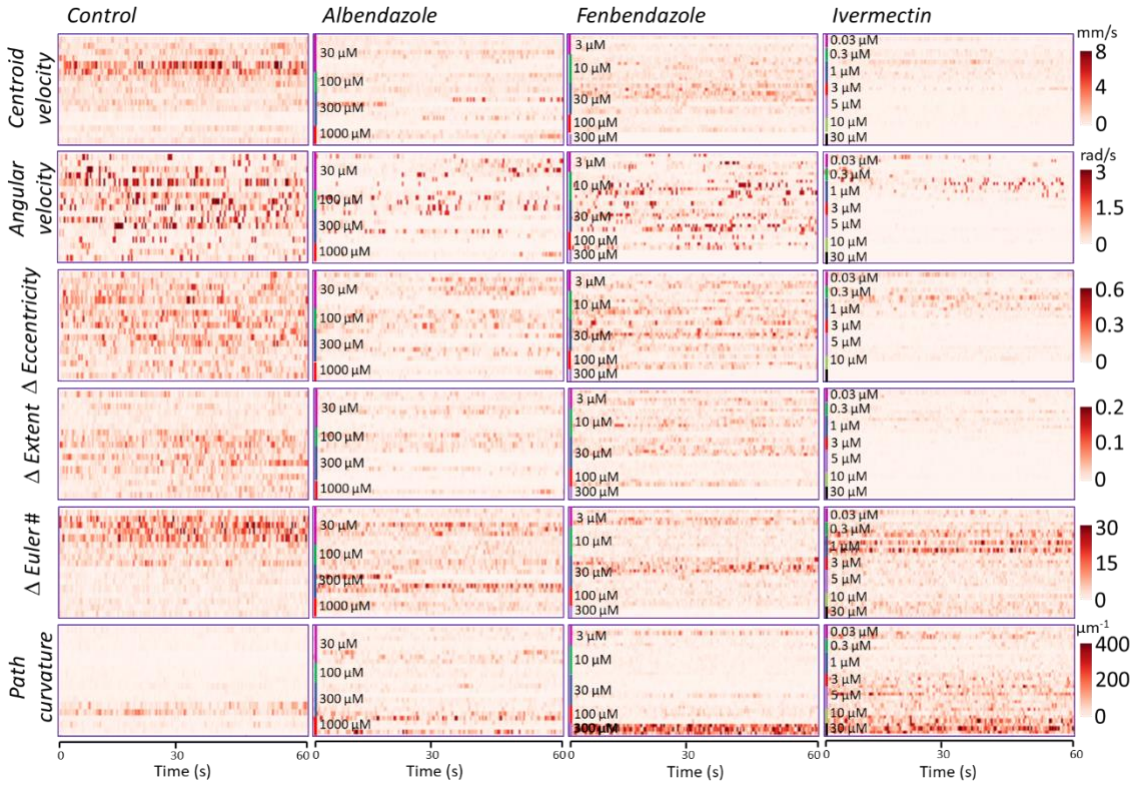


Figure 4.8 The features extracted by the BrugiaAdultTracker software for negative control and drug experiments were shown as raster plots. The runs corresponding to different concentrations were overlaid on the plots for each drug.

Table 4.1 Dose response analysis of the features extracted by the BrugiaAdultTracker software for Brugia malayi adult male worms and the three anthelmintics tested. Using GraphPad Prism 7, a variable slope nonlinear regression model was fitted to the responses at multiple drug concentrations for each feature and drug combinations. *IC50* and R^2 values represent the drug concentration at which the percentage response is 50 % and the coefficient of determination for the model. The drug concentration experiments were triplicates.

Feature	***IC*$_{50}$ (µM)**			**R^2**		
	fenbendazole	ivermectin	albendazole	fenbendazole	ivermectin	albendazole
Centroid Velocity	99.73	2.768	291	50.06	60.92	46.54
Angular Velocity	107.3	2.468	290.3	43.83	72.22	59.64
Δ Eccentricity	104.3	2.408	298.6	47.42	66.43	56.36
Δ Extent	102.9	2.368	291.1	32.41	78.01	40.57
Δ Euler Number	108.1	3.04	333.2	25.37	19.59	47.33
Path Curvature	139.1	13.39	342.5	96.18	69.75	48.04

The dose responses of *B. malayi* adult male worms to the three drugs were calculated for each feature and shown in Fig 4.9. The corresponding *IC50* (i.e. drug concentration at which the dose response was 50%) and R^2 values were tabulated in Table 4.1. Fig 4.9 shows that the feature responses associated with ivermectin and albendazole drugs were significantly lower than the negative control responses. It can be observed from Table 4.1 that the *IC50* values for the first four features (centroid velocity, angular velocity, rate of eccentricity, and rate of extent) correspond very well. It was already established in the literature that centroid velocity was a key parameter in calculating dose responses for other types of nematodes [24].

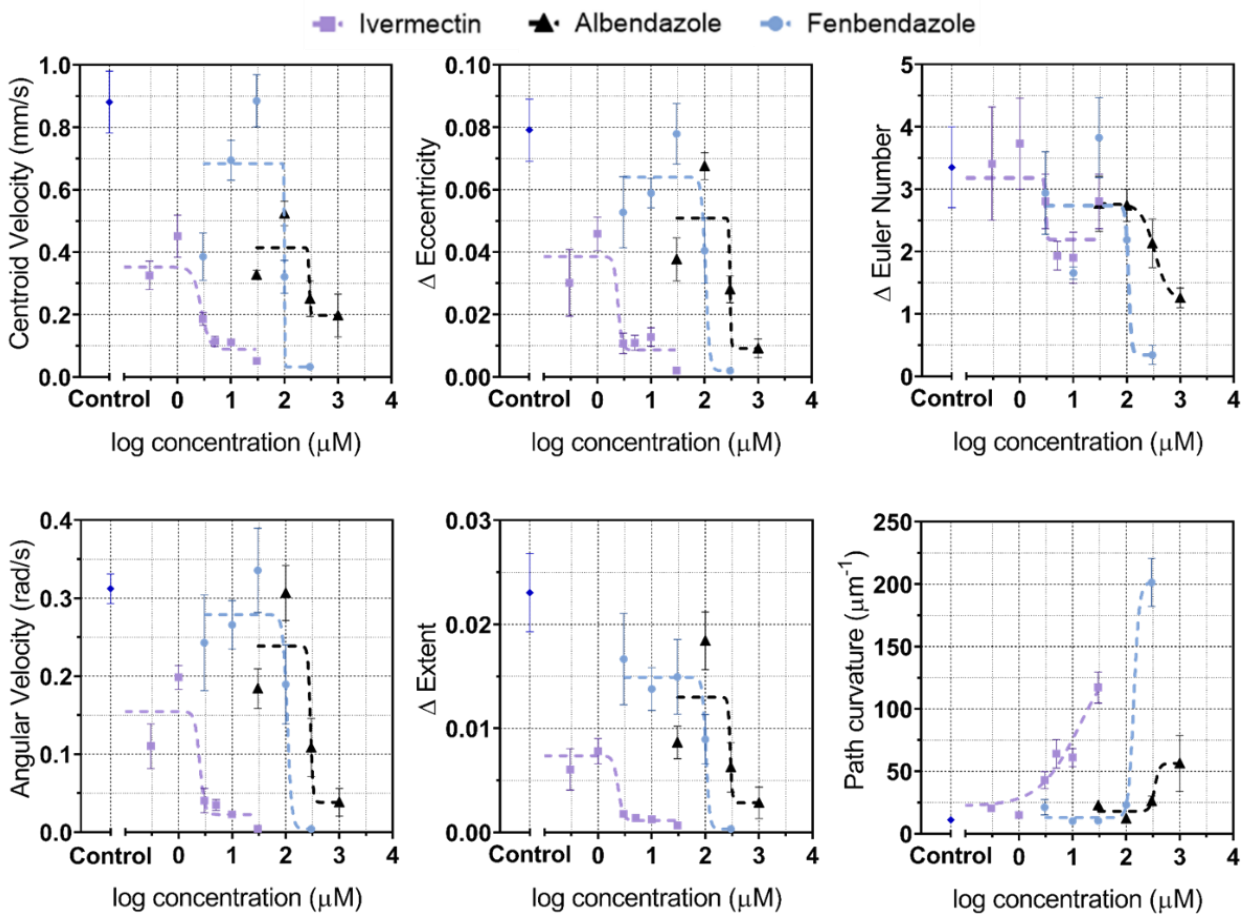


Figure 4.9 Illustration of the dose responses of the three anthelmintics (ivermectin, albendazole, and fenbendazole) for each feature extracted by the BrugiaAdultTracker software. The drug concentration experiments were triplicates. A variable slope nonlinear regression model provided by GraphPad Prism 7 was used to generate the curves.

In addition, based on the correlation between the four features it can be concluded that the features (angular velocity, rate of eccentricity, and rate of extent) can be considered key parameters for evaluating dose responses. The 99 % confidence intervals (data not shown) were calculated using these four features. The *IC50* values for the features (rate of Euler number and path curvature) however are significantly out of range when compared with these confidence intervals. In the case of rate of Euler number, it can be attributed to the fact that some of the R^2 values were very low (25.37% and 19.59%). For path curvature case, the R^2 values were very good and yet the *IC50* values were very different. This allowed us to conclude

that path curvature was not a good indicator for dose response calculation. The R^2 values for the key features for fenbendazole, ivermectin, and albendazole ranged from 32 – 50 %, 60 – 78 %, 40 – 60 %. This suggested that dose responses of adult male worms to ivermectin fit better compared to the other drugs. This also shows that since the R^2 values were not very high (> 90 percentage), investigation of multiple feature data was necessary to calculate accurate *IC50* values. These values for the three drugs using the four features are as follows: fenbendazole – 103.5 µM, ivermectin – 2.5 µM, albendazole – 292.75 µM. These values were similar to the data published in the literature [2].

***B. malayi* microfilaria assay**

The *B. malayi* microfilaria worms were recorded at 640 × 480 pixel resolution for 20 seconds at 30 frames per second. These worms were recorded on an agar nutrient pad (negative control) with cover slip on top to restrict their movement to two dimensions. Multiple features (directed motion, number of bends, and velocity) were extracted from the videos. A visual representation of the worm positions and the directed motion throughout the video for all the videos on an image plane were shown in Fig 4.10. The color spectrum represents the body positions of the worm skeleton from head to tail. The two colors (blue and orange) in the figure indicate the forward and reverse motion. The overall motion in the forward and reverse direction were overlaid on the plots as percentages for each video. It can be seen that worms corresponding to three plots (Fig 4.10 b, d-e) showed significant forward motion. Whereas worms corresponding to the other three plots (Fig 4.10 a, c, and f) had similar amounts of forward and reverse motion suggesting a probing pattern. It was seen that the worms in these videos elongated and contracted repeatedly while not moving significantly. The 'directed motion' feature can be used to segregate the worms into two groups: active (exhibiting larger

forward movement) and semi active (exhibiting a probing pattern). This can be used to study drug efficacies on worms exhibiting multiple levels of activity.

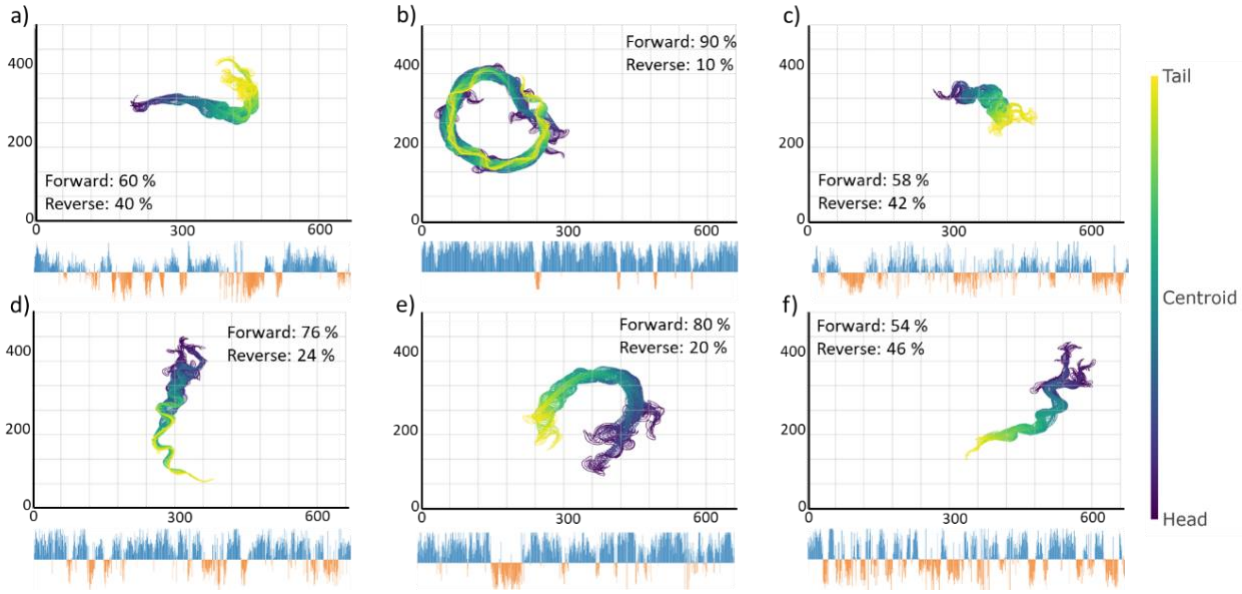


Figure 4.10 Visualization of the *B. malayi* microfilaria movement for the negative control runs on an x-y plane. The skeleton from head to tail was created of the worm and used to generate the plots. The movement of the worms in forward or reverse directions determined by directed motion feature were also shown where the blue color indicates forward, and orange indicates reverse movement. The overall percentage movement in the forward and reverse directions was specified as 'Forward' and 'Reverse' respectively.

The number of bends of the worm for each video frame were calculated and shown as a raster plot for all the videos in Fig 4.11a. It can be observed that the number of bends was roughly 5 – 10 for most of the frames and occasionally it increased drastically towards 15 – 20 bends. This increase was due to the self-occlusions. The sensitivity of this feature can be varied by adjusting the hysteresis window. In the literature, omega (Ω) shapes were used to identify different phenotypes for other type of nematodes (*C. Elegans*) [33]. The *B. malayi* worms being larger than *C. Elegans* can form multiple omega shapes. The 'number of bends' feature encompasses this information which can be used to identify different phenotypes for *B. malayi*.

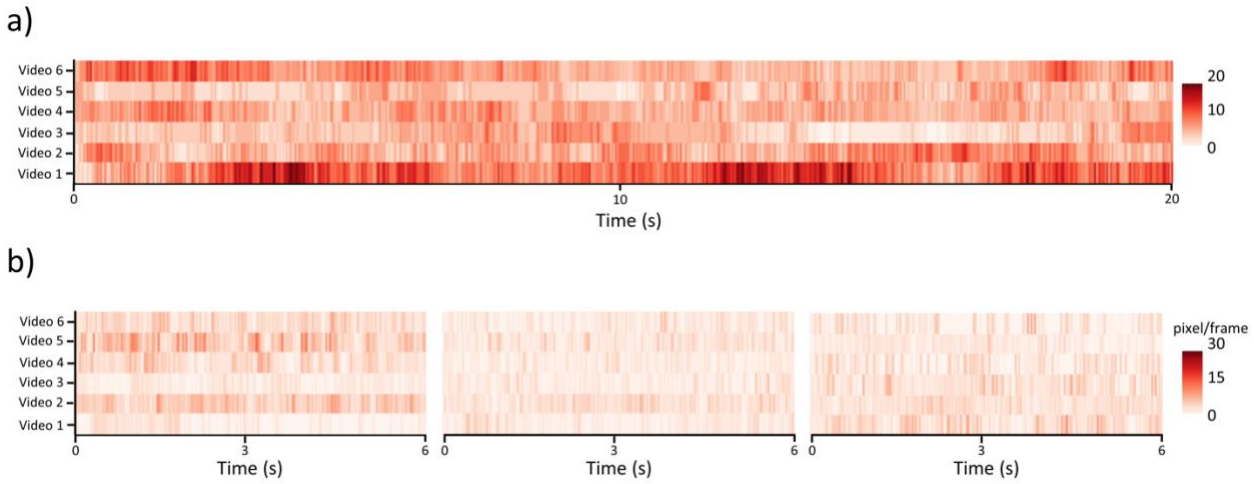


Figure 4.11 Illustration of the features extracted for the negative control experiments of *B. malayi* microfilaria. The duration of each video was around 20 seconds at 30 frames per second. Worm skeletons were generated, and the features were extracted. a) The number of bends made by the worm skeleton in each frame of a video was calculated and shown as a raster plot. Worm occlusions were indicated by the darker regions. b) The velocities of worm skeleton at head, centroid, and tail positions were calculated and shown as raster plots.

The velocities of head, centroid, and tail positions of the worm were plotted for all the videos in Fig 4.11b. From the figure, it can be inferred that the head velocities were higher when compared to the values seen at centroid and tail positions. This was expected as the head initiated the movement which was gradually experienced by the other body positions with lower values. It can be seen that the active worms (Video 2, 4, and 5) tend to have larger velocities when compared to semi-active worms. It can also be seen that in these three videos, there are certain frames where the values were significantly higher than the rest of the frames. It was observed that the worms exhibited a probing pattern where only the head portion was moving while the rest of the body remained stationary. The centroid velocities reflected the joint behavior of both the head and tail which can be seen in the figure. It can be concluded that multiple velocities at different body positions can be used to identify specific worm behavior and attributes.

Discussion

Literature shows that the adult parasitic nematodes and their microfilaria are responsible for the spread of lymphatic filariasis [7]. Different drugs are tested for anthelmintic activity against the parasites using *in vitro* assays to find potential drugs [7,8,10,11]. So, we developed a tool that aids in drug screening process for both adult *B. malayi* and microfilaria. The software package created is simple to use, flexible, designed to work with different image acquisition software, operates in real-time and doesn't need additional setup.

The modules in our software package use different worm tracking approaches (centroid-based and skeleton-based) to analyze the parasites of different sizes. Different approaches were chosen because of the following reasons. The adult parasites are long, round worm like nematodes that show significant three-dimensional activity. This makes it challenging to record the videos of adult parasites at higher magnifications which is necessary for skeleton-based approach. The microfilaria on the other hand are larger than *C. elegans* but significantly smaller than adult parasites allowing us to record them at higher magnifications to effectively extract their skeleton information. Previously reported methods use the same worm tracking approach (centroid-based) to characterize phenotypical behavior of nematodes of different stages [2, 13]. As research on *C. elegans* has shown, whole-body information was better at observing the phenotypical response than the centroid based approaches.

We have demonstrated that the worm tracker module, BrugiaAdultTracker, was able to characterize the phenotypical responses of adult male worms on three drugs (albendazole, ivermectin, and fenbendazole) by calculating *IC50* values of the dose response curves of each feature. The *IC50* values for four (centroid velocity, angular velocity, rate of eccentricity, and rate of extent) of the six features were in accordance with the values reported in the literature.

This work can be extended to include testing of the anthelmintic activity of other compounds and on parasites other than *B. malayi*.

In addition, we have also demonstrated the extraction of multiple features (body curvature, number of bends, directed motion, and point velocities) from skeleton body information of *B. malayi* microfilaria. To the best of our knowledge, this is the first attempt to construct a skeleton model and extract movement features from parasitic microfilaria of *B. malayi*. This work can be applied as a tool for drug screening analysis of microfilaria or other parasitic microfilaria types.

Conclusions

We have presented a worm tracker package comprising of two modules (BrugiaAdultTracker and BrugiaMFTracker) to analyze the phenotypical responses of *B. malayi* adult and microfilaria. The BrugiaAdultTracker module was tested on adult parasites assayed in three drugs (albendazole, ivermectin, and fenbendazole). Multiple features are extracted from the recorded videos and the module provides the option to save additional information for inspection. The dose response curves were plotted using the different features and it is shown that features other than centroid velocity can be used for phenotypical behavioral analysis. The BrugiaMFTracker module extracted the whole-body information from the microfilaria and built a worm model using which it computed the body posture and movement features. The software package is robust and requires minimal user intervention. Future scope lies in running multiple drug screening experiments and testing drug combinations to improve the drug efficacy.

References

1. Clare RH, Cook DAN, Johnston KL, Ford L, Ward SA, Taylor MJ. Development and Validation of a High-Throughput Anti- Wolbachia Whole-Cell Screen. J Biomol Screen. 2015;20: 64–69. doi:10.1177/1087057114551518

2. Marcellino C, Gut J, Lim KC, Singh R, McKerrow J, Sakanari J. WormAssay: A Novel Computer Application for Whole-Plate Motion-based Screening of Macroscopic Parasites. Geary TG, editor. PLoS Negl Trop Dis. 2012;6: e1494. doi:10.1371/journal.pntd.0001494

3. Bilsland E, Bean DM, Devaney E, Oliver SG. Yeast-Based High-Throughput Screens to Identify Novel Compounds Active against Brugia malayi. Geary TG, editor. PLoS Negl Trop Dis. 2016;10: e0004401. doi:10.1371/journal.pntd.0004401

4. Ballesteros C, Tritten L, O'Neill M, Burkman E, Zaky WI, Xia J, et al. The Effects of Ivermectin on Brugia malayi Females In Vitro: A Transcriptomic Approach. Weiss BL, editor. PLoS Negl Trop Dis. 2016;10: e0004929. doi:10.1371/journal.pntd.0004929

5. Kassis T, Skelton HM, Lu IM, Moorhead AR, Dixon JB. An Integrated In Vitro Imaging Platform for Characterizing Filarial Parasite Behavior within a Multicellular Microenvironment. Mitre E, editor. PLoS Negl Trop Dis. 2014;8: e3305. doi:10.1371/journal.pntd.0003305

6. Mostafa E, Storey B, Farghaly AM, Afify HAE-H, Taha AAE-R, Wolstenholme AJ. Transient effects of levamisole on Brugia malayi microfilariae. Invertebr Neurosci. Springer Berlin Heidelberg; 2015;15: 5. doi:10.1007/s10158-015-0181-0

7. Lu IM, Kassis T, Rogers AM, Schudel A, Weil J, Evans CC, et al. Optimization of culture and analysis methods for enhancing long-term Brugia malayi survival, molting and motility in vitro. Parasitol Open. 2018;4: e3. doi:10.1017/pao.2017.19

8. Rao RU, Huang Y, Fischer K, Fischer PU, Weil GJ. Brugia malayi: Effects of nitazoxanide and tizoxanide on adult worms and microfilariae of filarial nematodes. Exp Parasitol. Elsevier Inc.; 2009;121: 38–45. doi:10.1016/j.exppara.2008.09.020

9. Rao RU, Moussa H, Weil GJ. Brugia malayi: effects of antibacterial agents on larval viability and development in vitro. Exp Parasitol. 2002;101: 77–81. doi:10.1016/S0014-4894(02)00019-X

10. Tompkins JB, Stitt LE, Ardelli BF. Brugia malayi: In vitro effects of ivermectin and moxidectin on adults and microfilariae. Exp Parasitol. Elsevier Inc.; 2010;124: 394–402. doi:10.1016/j.exppara.2009.12.003

11. Laing R, Gillan V, Devaney E. Ivermectin – Old Drug, New Tricks? Trends Parasitol. Elsevier Ltd; 2017;33: 463–472. doi:10.1016/j.pt.2017.02.004

12. Smith HL, Paciorkowski N, Babu S, Rajan T. Development of a Serum-Free System for the in Vitro Cultivation of Brugia malayi Infective-Stage Larvae. Exp Parasitol. 2000;95: 253–264. doi:10.1006/expr.2000.4531

13. Storey B, Marcellino C, Miller M, Maclean M, Mostafa E, Howell S, et al. Utilization of computer processed high definition video imaging for measuring motility of microscopic nematode stages on a quantitative scale: "The Worminator." Int J Parasitol Drugs Drug Resist. Australian Society for Parasitology; 2014;4: 233–243. doi:10.1016/j.ijpddr.2014.08.003

14. Yemini E. High-throughput, single-worm tracking and analysis in Caenorhabditis elegans. 2013; doi:10.17863/CAM.15948

15. Baek JH, Cosman P, Feng Z, Silver J, Schafer WR. Using machine vision to analyze and classify Caenorhabditis elegans behavioral phenotypes quantitatively. J Neurosci Methods. 2002;118: 9–21. doi:10.1016/S0165-0270(02)00117-6

16. Ding X, Njus Z, Kong T, Su W, Ho C-M, Pandey S. Effective drug combination for Caenorhabditis elegans nematodes discovered by output-driven feedback system control technique. Sci Adv. 2017;3: eaao1254. doi:10.1126/sciadv.aao1254

17. Huang K, Cosman P, Schafer W. Automated tracking of multiple c. elegans with articulated models. 2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro. IEEE; 2007. pp. 1240–1243. doi:10.1109/ISBI.2007.357083

18. Huang KM, Cosman P, Schafer WR. Automated tracking of multiple C. elegans with articulated models. 2007 4th IEEE Int Symp Biomed Imaging From Nano to Macro - Proc. 2007; 1240–1243. doi:10.1109/ISBI.2007.357083

19. Swierczek NA, Giles AC, Rankin CH, Kerr RA. High-throughput behavioral analysis in C. elegans. Nat Methods. 2011;8: 592–598. doi:10.1038/nmeth.1625

20. Fontaine E, Burdick J, Barr A. Automated Tracking of Multiple C. Elegans. 2006 International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE; 2006. pp. 3716–3719. doi:10.1109/IEMBS.2006.260657

21. Wang SJ, Wang Z-W. Track-A-Worm, An Open-Source System for Quantitative Assessment of C. elegans Locomotory and Bending Behavior. Gilestro GF, editor. PLoS One. 2013;8: e69653. doi:10.1371/journal.pone.0069653

22. Husson SJ. Keeping track of worm trackers. WormBook. 2013; doi:10.1895/wormbook.1.156.1

23. Stephens GJ, Johnson-Kerner B, Bialek W, Ryu WS. Dimensionality and Dynamics in the Behavior of C. elegans. Sporns O, editor. PLoS Comput Biol. 2008;4: e1000028. doi:10.1371/journal.pcbi.1000028

24. Ramot D, Johnson BE, Berry TL, Carnell L, Goodman MB. The parallel worm tracker: A platform for measuring average speed and drug-induced paralysis in nematodes. PLoS One. 2008;3: 6–12. doi:10.1371/journal.pone.0002208

25. Nagy S, Goessling M, Amit Y, Biron D. A Generative Statistical Algorithm for Automatic Detection of Complex Postures. Faisal AA, editor. PLOS Comput Biol. 2015;11: e1004517. doi:10.1371/journal.pcbi.1004517

26. Greenblum A, Sznitman R, Fua P, Arratia PE, Sznitman J. Caenorhabditis Elegans Segmentation Using Texture-Based Models for Motility Phenotyping. IEEE Trans Biomed Eng. 2014;61: 2278–2289. doi:10.1109/TBME.2014.2298612

27. Moore BT, Jordan JM, Baugh LR. WormSizer: High-throughput Analysis of Nematode Size and Shape. Gill M, editor. PLoS One. 2013;8: e57142. doi:10.1371/journal.pone.0057142

28. Yemini E, Jucikas T, Grundy LJ, Brown AEX, Schafer WR. A database of Caenorhabditis elegans behavioral phenotypes. Nat Methods. 2013;10: 877–879. doi:10.1038/nmeth.2560

29. Yemini E. High-throughput, single-worm tracking and analysis in Caenorhabditis elegans (Doctoral thesis). 2011;

30. Fontaine E, Barr AH, Burdick JW. Tracking of multiple worms and fish for biological studies. ICCV Work Dyn Vis. 2007;

31. Léger JC. Menger curvature and rectifiability. Ann Math. 1999;149: 831–869. doi:10.2307/121074

32. Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R, Wu AY. An efficient k-means clustering algorithm: analysis and implementation. IEEE Trans Pattern Anal Mach Intell. 2002;24: 881–892. doi:10.1109/TPAMI.2002.1017616

33. Krajacic P, Shen X, Purohit PK, Arratia P, Lamitina T. Biomechanical profiling of Caenorhabditis elegans motility. Genetics. 2012;191: 1015–1021. doi:10.1534/genetics.112.141176

CHAPTER 5.

CONCLUSIONS

There is considerable interest in building instrumentation for image acquisition and data analysis towards unravelling physical and behavioral attributes of biological samples. Within the broad field of bioengineering, there are several application areas where automated instrumentation has enhanced the experimental throughput and sampling time while reducing costs. In this thesis, I developed imaging instrumentation and data analysis tools to address the limitations in skin cancer diagnostics, plant parasitic nematology, and neglected tropical disease. The common theme for the developed instrumentation was the incorporation of portability, low-cost imaging, and high-throughput analysis of phenotypic traits. Specifically, the smartphone-based skin cancer diagnostic system provided a portable, at-home setup to capture high-quality images of skin lesions along with a software application to judge the cancerous nature of skin lesions through classification of extracted feature sets. Similarly, the combination of scanners, lens-free imaging setup, and machine learning algorithms provided the nematode egg count from soil samples in an automated, low-cost manner that obviates the need for trained personnel. In addition, methods were developed to identify the multi-faceted behavior of Brugia malayi nematodes using a new set of movement features which were then used to screen the effects of three drugs on the worms.

The performance of abovementioned instrumentation was tested on available samples; be it images of skin moles, soil samples with nematode eggs or videos of Brugia malayi worms. The efficacy of the developed methods could be further improved by testing on larger datasets, which is often a challenging issue in bioengineering. Recent trends in bioengineering show that experimental testing of new tools with diverse sample sets and large sample sizes have a

greater impact in the scientific community with higher rate of adoption. We hope the presented methods provide a proof-of-concept for advancing the instrumentation capabilities within the identified areas of bioengineering with more rigorous testing of performance.