UNIVERSITY OF CALIFORNIA, SAN DIEGO

A Machine Vision and Statistical Learning System for Studying *C. elegans* phenotypes

A dissertation submitted in partial satisfaction of the

requirements for the degree Doctor of Philosophy

in

Electrical Engineering (Signal and Image Processing)

by

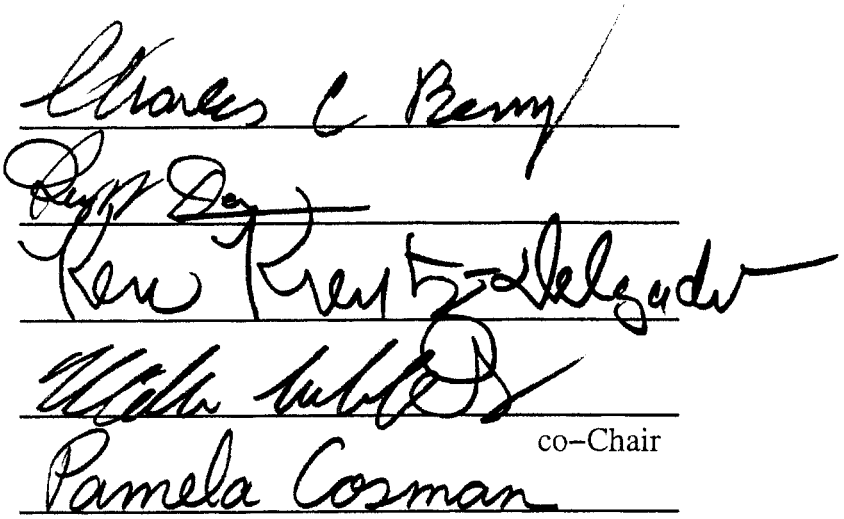
Wei Geng

Committee in charge:

Professor Pamela Cosman, Chair
Professor William R. Schafer, Co-Chair
Professor Charles C. Berry
Professor Kenneth Kreutz-Delgado
Professor Sujit Dey

2004

The dissertation of Wei Geng is approved. And it is acceptable in quality and form for publication on microfilm:



co–Chair

Chair

University of California, San Diego

2004

To my wife

Contents

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

First, I would like to thank my advisors Professors Pamela Cosman and William R. Schafer who pioneered the field and guided me through the challenging path. I am very grateful to Pam for being generous with her time and effort to make my graduate study a successful and rewarding experience, and for her numerous efforts to improve my writing skills. To Bill for his inspiring enthusiasm and unreserved support.

I am indebted to Professor Charles Berry for many discussions and some crucial ideas that led to the success of this project. I also wish to thank my doctoral committee members, Professors Kenneth Kreutz-Delgado and Sujit Dey for taking the time to review my work, ask questions and give advice.

Many thanks are due to Dr. Joong-Hwan Baek, Dr. Zhaoyang Feng, Laddan Hashemian, Clare Huang, Dan Poole, Megan Palm, and Marika Orlov for participating in the different aspects of the project. To Song Cen, Athanasios Leontaris, Qinghua Zhao, Yushi Shen, and Ben Farber for friendly discussions and computing related help in the Information Coding Lab.

To my parents, Zhaojun and Yuqin and my brother Jian I am grateful for their constant love and support.

And finally, I dedicate this dissertation to my beloved wife Yitan. Without her unconditional love and support, I can not imagine going through this journey.

I thank the Caenorhabditis Genetics Center for the *C. elegans* strains. This work was supported by a grant from the National Institute on Drug Abuse.

PUBLISHED MATERIAL

The text of Chapters 2, 3, 4, in part, is a reprint of material that appears as "Quantitative Classification and Natural Clustering of *C. elegans* Behavioral Phenotypes" by W. Geng, P. Cosman, J.-H. Baek, C. Berry, and W.R. Schafer, in *Genetics*, vol. 165, pp. 1117–1126, 2003, and a conference paper "Image Feature Extraction and Natural Clustering of Worm Body Shapes and Motion Characteristics" by W. Geng, P. Cosman, J.-H. Baek, C. Berry and W.R. Schafer, in the *IASTED International Conference on Signal and Image Processing (SIP 2003)*, Honolulu, Hawaii, 2003. The text of Chapters 2, 3, 5, in part, is a reprint of the material that appears as "Automatic Tracking, Feature Extraction and Classification of *C. elegans* Phenotypes" by W. Geng, P. Cosman, Z. Feng, C. Berry, W.R. Schafer, in *IEEE Transactions on Biomedical Engineering*, in press, 2004, and a conference paper "Automated Worm Tracking and Classification" by W. Geng, P. Cosman, C. Huang, and W.R. Schafer, in the *37th Asilomar Conference on Signals, Systems and Computers*, pp. 2063-2068, November 2003, Pacific Grove, California. The dissertation author was the primary researcher, and Dr. Pamela Cosman and Dr. William R. Schafer supervised the research which forms the basis for these chapters.

The text of Chapter 2, 3, 6, in part, is a reprint of the material that will appear as "Egg Onset Detection Using Deformable Template Matching" by W. Geng, P. Cosman, W.R. Schafer, in the *IASTED International Conference on Computer Graphics and Image Processing (CGIM2004)*, Kauai, Hawaii 2004; and has been submitted for publi-

cation to the *EURASIP Journal of Applied Signal and Image Processing* as "*C. elegans* Egg-laying Detection and Behavior Study Using Image Analysis" by W. Geng, P. Cosman, M. Palm, and W.R. Schafer. The dissertation author was the primary researcher, and Dr. Pamela Cosman and Dr. William R. Schafer supervised the research which forms the basis for this chapter.

VITA

October 27, 1972	Born, Xi'an, China
1993	B.S., Beihang University, Beijing, China
1993-1995	Research Assistant, National Lab on Machine Perception, Peking University, Beijing, China
1997	M.S. in Electrical and Computer Engineering, Northeastern University, Boston, MA.
1997	Member of Technical Staff, Boston Technology, Andover, MA.
1997-2001	Embedded Software Engineer, Hewlett-Packard Co., San Diego, CA.
2001-2004	Graduate Student Researcher, University of California, San Diego.
2004	Senior Analytical Engineer, ID Analytics, Inc., San Diego, CA.
2004	Ph.D., University of California, San Diego.

PUBLICATIONS

1. W. Geng, P. Cosman, M. Palm, and W. R. Schafer. "*C. elegans* Egg-laying Detection and Behavior Study Using Image Analysis." Submitted to *EURASIP Journal on Applied Signal and Image Processing*, January 2004.

2. W. Geng, P. Cosman,C. Berry, Z. Feng, and W. R. Schafer. "Automated Tracking, Feature Extraction and Classification of *C. elegans* Phenotypes." *IEEE Transactions on Biomedical Engineering*, in press, 2004.

3. W. Geng, P. Cosman, and W. R. Schafer. "Egg Onset Detection Using Deformable Template Matching." to appear in the *IASTED International Conference on Computer Graphics and Image Processing(CGIM2004)*, Kauai, Hawaii, August 2004.

4. W. Geng, P. Cosman, J. H. Baek, C. Berry, and W. R. Schafer. "Quantitative Classification and Natural Clustering of *C. elegans* Behavioral Phenotypes." *Genetics*, vol. 165, pp. 1117–1126, 2003.

5. W. Geng, P. Cosman, J. H. Baek, C. Berry, and W. R. Schafer. "Image Features and Natural Clustering of Worm Body Shapes and Motion." *Proceedings of the Fifth IASTED International Conference on Signal and Image Processing (SIP)*, pp. 342-347, Honolulu, HI, August 13-15, 2003.

6. W. Geng, P. Cosman, C. Huang, and W. R. Schafer. "Automated Worm Tracking and Classification." *Proceedings of the 37th IEEE Asilomar Conference on Signals, Systems and Computers*, pp. 2063-2068, Pacific Grove, CA, November 2003.

7. M. Florentine, S. Buus, and W. Geng. "Toward a Clinical Procedure for Narrowband Gap Detection I: A Psychophysical Procedure." *Audiology*, vol. 39, pp. 161–167, 2000.

8. M. Florentine, S. Buus, and W. Geng. "Psychometric Functions for Gap Detection in a yes-no Procedure." *Journal of Acoustic Society of America*, vol. 106, pp. 3512–3520, 1999.

9. M. Florentine, S. Buus, and W. Geng. "Psychometric Functions for Gap Detection." *16th International Congress on Acoustics and 137th meeting on Acoustic Society of America*, Seattle, WA, 1998.

FIELDS OF STUDY

Major field: Electrical Engineering

Studies in Signal and Image Processing

Professor Pamela C. Cosman

Professor William R. Schafer

ABSTRACT OF THE DISSERTATION

A Machine Vision and Statistical Learning System for Studying *C. elegans* phenotypes

by

Wei Geng

Doctor of Philosophy in Electrical Engineering

(Signal and Image Processing)

University of California, San Diego, 2004

Professor Pamela Cosman, Chair

Professor William R. Schafer, Co-Chair

The nematode *C. elegans* has powerful genetics, a well-described nervous system, and a complete genome sequence; thus, it is well suited to analysis of behavior and development at the molecular and cellular levels. In particular, the ability to functionally map the influence of particular genes to specific behavioral consequences makes it possible to use genetic analysis to functionally dissect the molecular mechanisms un-

derlying poorly understood aspects of nervous system function. However, many genes with critical roles in neuronal function have effects on behavior that to a casual observer appear very subtle or difficult to describe precisely. Therefore, to fully realize the potential of *C. elegans* for the genetic analysis of nervous system function, it is necessary to develop sophisticated methods for the rapid and consistent quantitative characterization of mutant phenotypes, especially those related to behavior.

This dissertation addresses several key issues in building a computer vision system which can accomplish this rapid and consistent characterization of mutant types. We propose novel and practical methods for segmenting and tracking the worm body and its head and tail positions. We design a comprehensive set of 253 features that characterize the worm phenotypes measured from the image video sequences. We evaluate and design several clustering and classification methods to demonstrate the system can be used effectively for quantitatively characterizing the phenotypic patterns caused by mutations or pharmacological treatments in *C. elegans*. The egg-laying detection algorithms are also developed using this system and behavioral changes surrounding egg onsets are also studied. Together, these constitute a completely automated *C. elegans* tracking and identification system.

Chapter 1

Introduction

1.1 Background and Significance of Computer Vision Based *C. elegans* study

1.1.1 *C. elegans* as an Excellent Model for Studying the Molecular and Cellular Basis of Nervous System Function and Behavior

The nematode *Caenorhabditis elegans* is widely used for studies of nervous system function and development. *C. elegans* is a free-living worm, approximately 1 μm in length, that lives in the soil and feeds on bacteria. It has a simple nervous system containing 302 neurons, and the precise position, cell lineage, and synaptic connectivity of each of these neurons is known [80][81][92]. Despite its anatomical simplicity, the *C. elegans* nervous system mediates surprisingly diverse and intricate patterns of behavior. The sense organs of *C. elegans* are capable of perceiving and responding to a wide range of environmental conditions, including heavy and light touch, temperature, volatile odorants, osmotic and ionic strength, food, and other nematodes. Each of these

sensory modalities in turn regulates many aspects of the animal's behavior, including the rate and direction of movement, the rates of feeding, egg-laying, defecation, and the process of mating. Because a particular neuron can be positively identified based on its position, it is possible to eliminate the function of an individual neuron or group of neurons through single cell laser ablation. Moreover, because of their short generation time, completely sequenced genome, and accessibility to germline transformation, these animals are highly amenable to molecular and classical genetics. Thus, in *C. elegans* it is relatively straightforward to evaluate the functions of particular neurons or gene products by characterizing the effects of mutations or neuronal ablations on the animal's behavior. For these reasons, it is an excellent model organism for studying the molecular and cellular basis of nervous system function and behavior. *C. elegans* is among the most widely studied model organisms.

This approach to understanding nervous system function relies on the ability to obtain precise quantitative computer-based measurements of behavioral abnormalities seen in mutant and cell-ablated animals. Although gross behavioral defects can often be discriminated qualitatively by simple observation, precisely defining these differences can be challenging without quantitative measurements of parameters such as the curvature and amplitude of body bends. Furthermore, many behavioral abnormalities are reliably detected only using computer-based methods. For example, some behavioral events, such as oviposition, occur on a time scale that precludes evaluation by long-time human observation [88][95]. In other cases, differences between mutant and normal strains are not apparent to the eye, but can readily be discriminated through quantitative computer-based analysis. A number of mutants exhibiting altered egg-laying patterns or hyperactive locomotion fall into this category.

1.1.2 *C. elegans* is Ideally Suited for Comprehensive Phenotype Studies

1. **Hundreds of well-characterized loss-of-function alleles exist in isogenic strain backgrounds.** A major advantage of *C. elegans* is the abundance of existing mutant lines with visibly different phenotypes. At present, loss-of-function mutants defining approximately 500 genes have been identified in *C. elegans*; of these approximately 300 have been cloned [71]. Canonical alleles of nearly all of these mutant genes are publicly available from the Caenorhabditis Genetics Center, and annotations describing the nature of the mutant phenotypes are accessible on the *C. elegans* internet database WormBase. Significantly, the vast majority of these mutant lines are derived from the wild-type Bristol (N2) strain; since *C. elegans* is a self-fertilizing hermaphrodite, this line is inbred to such a degree that it is essentially homozygous at all loci. Thus, outside of secondary mutations acquired during mutagenesis or propagation in the laboratory, nearly all *C. elegans* mutants are isogenic in genetic background to one another and to the wild-type N2. This provides an excellent starting point for the construction of any phenotype database that seeks to correlate particular visible abnormalities with specific genetic defects.

2. **Automated methods can be used for comprehensive phenotyping of morphological, developmental and behavioral characters.** The relative simplicity of nematode anatomy and behavior makes it feasible to apply automated methods to precisely and thoroughly characterize key aspects of *C. elegans* mutant phenotypes. Since nematodes lack appendages and move almost entirely in two dimensions (dorsoventral and anteroposterior), a worm's shape as well as its be-

havior can be effectively captured by video recordings of animals crawling across a flat surface such as an agar plate. With the development of specialized image processing and analytical tools, it is possible to obtain an information-rich, comprehensive behavioral signature from data acquired by an automated tracking and imaging system. Such methods can both facilitate high throughput data collection as well as allow objective scoring of a large number of phenotypic characters.

3. ***C. elegans* is a well-established molecular model system for development, cell biology and neuroscience.** Crucially, *C. elegans* is not only notable for its ease of manipulation in the laboratory, but also for its remarkable track record as a model system for the identification of well-conserved developmental and signal transduction mechanisms. For example, the conserved mechanisms for programmed cell death [37][21] were first revealed through genetic studies in *C. elegans*. Likewise, the developmental roles of small regulatory RNAs were first discovered in studies of nematode heterochronic mutants [51][52]. Many proteins playing critical roles in nervous system function and development were first identified genetically in *C. elegans*. In addition, *C. elegans* mutants harboring mutations affecting previously identified nervous system proteins have been invaluable for evaluating function in an intact, living animal, and in some cases have served as models for human disease [76] [12][54][50][59][66][74]. Recently, genetic pharmacology in *C. elegans* has been successfully used to study the mechanisms of action for psychotropic drugs, including therapeutic agents and drugs of abuse [15][82] [89][91].

1.1.3 The Importance of Quantitative Analysis in the Characterization of *C. elegans* Phenotypes

A critical aspect to the genetic analysis of nervous system function is the availability of reliable assays to detect behavioral abnormalities. However, standard assays for abnormalities in complex behaviors such as locomotion are highly imprecise and subjective. For example, mutations in over 100 genes have been described that cause abnormal or uncoordinated movement [71]. In the published literature (e.g. [38]), these uncoordinated (Unc) mutants are usually classified into a number of descriptive categories, including kinkers (animals that fail to propagate a smooth sine wave down the body during locomotion), coilers (animals that tend to coil up during movement), shrinkers (animals that contract dorsal and ventral body muscles simultaneously), loopy mutants (animals that make sine waves of abnormally large amplitudes) and slow/sluggish animals (animals that move slower to varying degrees than normal animals). Since these abnormalities are almost always scored subjectively by a human observer, it is not uncommon for the same Unc mutant to be described differently by different researchers. Moreover, the imprecise nature of these descriptions means that two mutants with very different phenotypes are often assigned the same classification. For example, both *unc-29* mutants (defective in a nicotinic ACh receptor [25]) and *unc-2* mutants (defective in a neuronal voltage-gated calcium channel [76]) are classified as weak kinkers [38], despite the fact that these animals exhibit movement patterns that are visibly quite different to an expert observer. Thus, even though mutants affecting a common molecular target generally have qualitatively similar behavioral phenotypes, it is difficult if not impossible to assess which mutants have genuinely similar phenotypes based on published descriptions alone.

Another problem that arises during analysis of worm behavioral phenotypes is that mutants with physiologically relevant defects in nervous system function often exhibit only subtle alterations in behavior. Such subtle behavioral phenotypes can often be accurately scored only through the use of special phenotyping tools. For example, mutants with deletions of the *flp-1* gene, which encodes a homologue of a human opioid modulatory peptide, exhibit slightly hyperactive and loopy movement [65]. However, although these abnormalities can be quantified by comparing videotapes of *flp-1* and wild-type worms, they are extremely difficult to recognize by visual inspection; in fact, only the availability of a PCR assay makes it possible to reliably score for the *flp-1* deletion in a genetic cross. Other genes whose knockout phenotypes are extremely subtle to the casual observer include those required for serotonin synthesis [79][53] as well as those encoding the *C. elegans* AMPA and NMDA receptor homologues [11].

One way these problems can be surmounted is to use video capture, storage, and analysis systems to aid manual analysis. By recording the behavior of individual animals, often for long time intervals, it is possible to rigorously identify and quantitate deviations from wild-type behavior that are difficult to discern by eye. For example, the movement defect caused by mutations in a neuropeptide Y receptor was not apparent until the rate of movement was measured and compared to wild-type strains using image analysis software [18]. A different system that tracked the movements of individual animals in a chemotactic gradient [68] was instrumental in identifying the sensory defect in *lim-6* mutant animals. Finally, manual analysis of videotapes confirmed the hyperactive phenotypes of the *flp-1* neuropeptide deletion mutants [65]. Automated motion analysis systems have proven equally useful in genetic studies of behavior and drug response mechanisms in the fruit fly Drosophila [60] [2]. Together, these studies demonstrate the usefulness of sophisticated, quantitative behavioral assays in the analysis of many

conserved neuronal signaling pathways.

Much like human vision, a machine vision system combines image/video sensing, image processing, and image analysis together to make sense of input images or video sequences. Powered by the computing technology revolution, machine vision has experienced explosive growth beyond traditional industrial inspection and measurement areas, and expanded into new fields such as surveillance, transportation, and multimedia applications. Statistical pattern recognition and classification techniques often work as a post-processing step of a machine vision system to recognize and/or to classify objects or patterns. A well-designed machine vision and statistical pattern recognition and classification system can not only work automatically and tirelessly, but can also provide new insights into characteristics that human observers are not able to identify or quantify. The system also provides more quantitative measurements and assessments of both input data and output results. Together, they provide excellent tools to study *C. elegans* problems.

In this dissertation, an automated computer vision and statistical learning system for studying *C. elegans* phenotypes and specific behaviors was developed. It includes data acquisition, tracking, and data analysis. In this chapter, we discuss related work in the field, and give an overview of the system and the outline of the dissertation.

Figure 1.1 and Table 1.1 show representative images and descriptions for wild and 15 different mutant types. All worms are young adults.

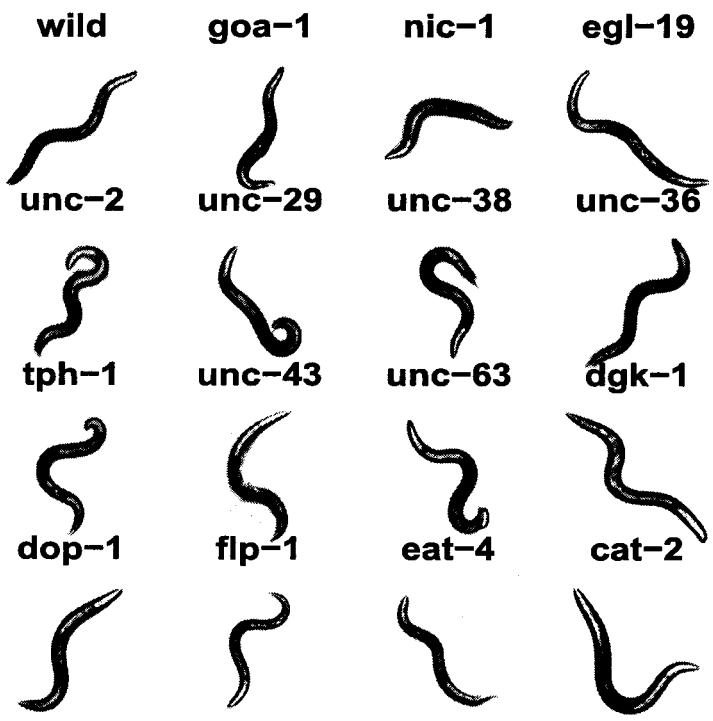


Figure 1.1: Representative images of wild type and 15 mutant types. Descriptions of these mutants are summarized in Table 1.1.

1.2 Related Work

Even though there are no systems that are as automated and comprehensive as the system developed in this dissertation, there were reported computer-driven systems for automated recording and/or analysis of a specific *C. elegans* behavior.

The systems developed by [18][16] are designed to observe multiple animals at low magnification and track the position of each animal over time. Such systems make it possible to measure large-scale behavioral features such as the rate and direction of movement and the frequency of reversals in direction. However, because the animals are observed at low magnification, it is not possible to obtain more detailed information about their body posture and morphology.

There are also systems developed to study specific behaviors. For example, there are systems in [88] that reveal the alternative behavioral states controlled by serotonin in egg-laying behavior systems [18] that study solitary (slow moving) and social feeding (fast moving) behaviors, systems [68] that investigate the behavioral mechanism of chemotaxis by studying the speed and turning rate during chemotaxis in gradients of the attractants ammonium chloride or biotin; systems [60] that report progressive increases in both locomotor activity and stereotyped behavior known as "reverse tolerance" or "behavioral sensitization" caused by repeated intermittent doses of cocaine. Each system is capable of measuring some specific behavioral parameters, but there is no automated system that is designed to classify a large number of mutant types by evaluating large numbers of behavioral parameters simultaneously.

Even though it is not directly related to *C. elegans*, it is interesting to notice

Table 1.1: Descriptions of wild and 15 mutant types.

Strain	Defective molecule	Description	Source
Wild-type	N/A	Normal	Mendel et al., *Science, 1995; [59] Segalat, et al.*, Science, *1995 [74]*
goa-1	α subunit of G-protein G_o	hyperactive, defective male mating	Mendel et al., *Science*, 1995 [59]
nic-1	nic-1-type1 glycosyl-transferase	dumpy, moves poorly	J. Kim and W. Schafer, unpublished
egl-19	egl-19-L-type VGCC α1 subunit	moderate bloating, slow and floppy	Lee et al., *EMBO J.*, 1997 [50]
unc-2	unc-2-non-L-type VGCC α1 subunit	weak kinker, sluggish, thin	Schafer et al., *Nature*, 1995 [76]
unc-29	unc-29-nicotinic receptor β subunit	weak kinker, head region stiff, moves better in reverse	Fleming et al., *J. Neurosci.*, 1997 [25]
unc-38	unc-38-nicotinic receptor α subunit	weak kinker, sluggish, slightly dumpyish	Fleming et al., *J. Neurosci.*, 1997 [25]
unc-36	unc-36-voltage-gated calcium channel (VGCC) $\alpha 2/\delta$ subunit	very slow, thin loopy at rest	Brenner, *Genetics*, 1974 [10]
tph-1	tph-1-tryptophan hydroxylase	bloated, slow moving	Sze et al., *Nature*, 2000 [83]
unc-43	unc-43-CaMKII	slow, lazy, slightly rippling movement	Reiner et al., *Nature*, 1999 [70]
unc-63	unc-63-nicotinic receptor subunit	weak kinker, slow, inactive	Lewis et al., *Nurosci.*, 1980 [55]
dgk-1	dgk-1-diacylglycerol kinase	Hyperactive for locomotion and foraging	Nurrish et al., *J. Neurosci.*, 1999 [67]
dop-1	dop-1-D1 dopamine receptor	locomotion normal	Sanyal et al., *EMBO Journal*, 2004 [73]
flp-1	flp-1-Fa-related neuropeptide	hyperactive, loopy uncoordinated movement	Nelson, et al., *Science*, 1998 [65]
eat-4	eat-4(ky5)-vesicular glutamate transporter	foraging abnormal	Nelson, et al., *Science*, 1998 [65]
cat-2	cat-2-tyrosine hydroxylase	defective in food-swallowing behavior	Lee et al., *J. Neurosci.*, 1999 [50]

systems [2] that show that acute responses to cocaine and nicotine are blunted by pharmacologically induced reductions in dopamine levels by measuring the effect of psychostimulants on fly behavior.

As a preliminary study to this dissertation, [1] developed a system designed to follow an individual animal at high magnification. To keep the animal from leaving the field of view, a tracking program directs the movement of a motorized stage to maintain the worm in the center of the field. In this way, it is possible to follow the position of the animal over long time periods and comprehensively measure multiple features that define behavioral and morphological abnormalities of nematode mutants. By using 94 such features, the system was able to classify representative mutant types using a binary decision tree algorithm (CART). However, although this system performed well at distinguishing visibly different mutant phenotypes, it was less effective at distinguishing types with more subtle differences.

1.3 Research Overview and Dissertation Structure

In this dissertation, an automated video capture and data analysis system is developed. Our approach can be divided into several well-defined stages, presented in Figure 1.2. After video images acquisition, the images are first segmented to isolate the worm body from the background and remove noise and undesired components. Next, the head and tail are recognized for entire video sequences. Feature extraction is applied, to extract the useful information from the segmented objects and the head and tail locations. Finally, a classifier or clustering procedure operates on the characteristics extracted by the previous stages. To study a specific behavior, a set of image analysis algorithms can be

applied before or after the feature extraction step depending on the specific task.

The remainder of this dissertation is organized as follows. Chapter 2 explains the algorithms we developed for segmenting the worm body from the background and tracking worm location and movement by recognizing the head and tail in the video sequences. Chapter 3 covers all the features that are measured and calculated from either the individual image frames or through the entire video sequence. Chapters 4 and 5 discuss the k-means based natural clustering and Random Forests classification schemes and their performance. Comparisons to other methods are also evaluated in these chapters. To study egg-laying events using our system, Chapter 6 presents the algorithms we developed to detect egg-laying events automatically and some behavioral change results. Finally, Chapter 7 summarizes contributions made in the dissertation and lists a few future research directions.

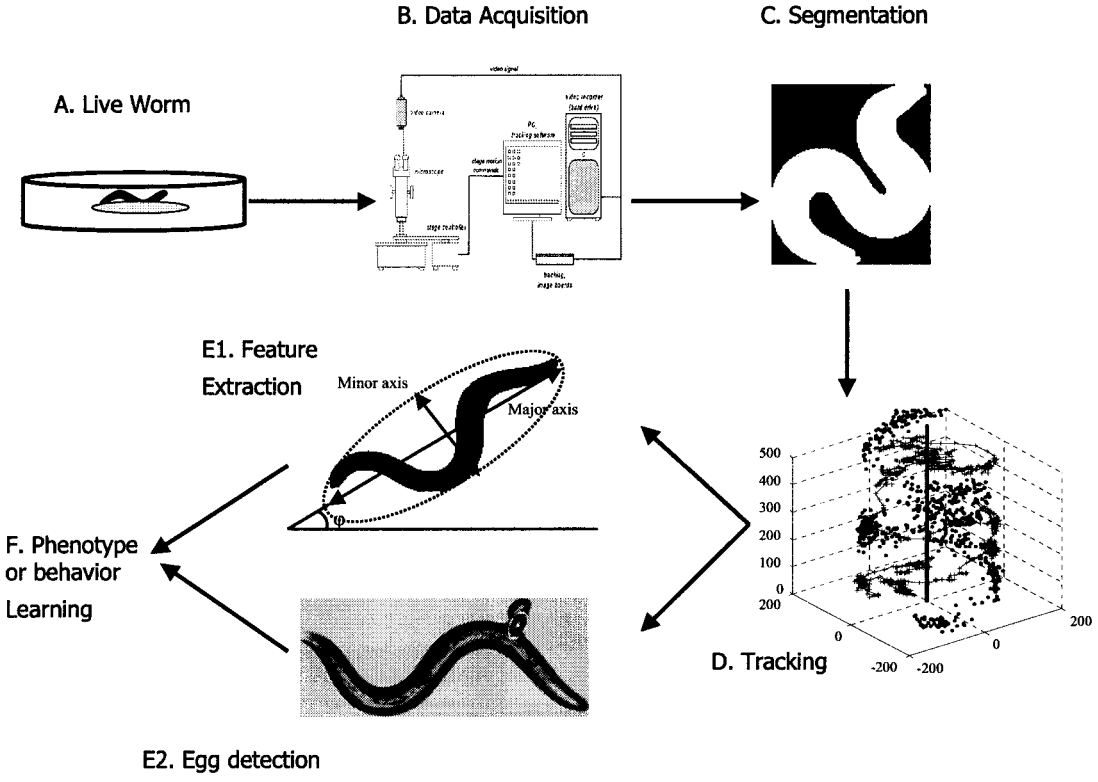


Figure 1.2: (A) Fourth-stage larvae are picked the evening before the experiment and tracked the following morning on a fresh plate. (B) A data acquisition system containing a high power microscope and a stage controller is used to track and record the worm locomotive information. (C) Image processing steps remove noise and separate worm bodies from the background. (D) head and tail tracking. (E1) Feature extraction step extracts a total of 253 features from the binary and gray image sequence. (E2) Automated egg-laying detection. (F) The data are then fed through the learning stage for classification, clustering or specific behavioral study.

1.4 Summary of Contributions

This dissertation addresses several key issues for building a comprehensive computer vision and statistical learning system for studying *C. elegans* behavioral phenotypes. These issues include how to segment the worm body from the background correctly; how to track worm movement and position; how to automatically extract useful features that characterize phenotypes; how to use statistical methods to learn from the features, tasks such as distinguishing one phenotype from another (classification), mapping from phenotype to its underlying genotype (clustering), etc; and how to design an image analysis algorithm to study egg-laying. The following list summarizes contributions made in this dissertation.

1. **Characteristic features:** To precisely characterize the behavioral phenotypes of a large number of mutants, characteristic features need to be designed and realized. The dissertation proposes a comprehensive set of 253 features to measure a wide range of morphological and locomotion characteristics. The important features are also identified.

2. **Segmentation and Tracking algorithms:** As a foundation of any computer vision system, object segmentation and tracking algorithms are crucial to the system performance. In this dissertation, we propose some novel methods to segment and track the worm movement and location accurately. They not only reduce the noise of the measurements of the system, but also enable a large number of additional features to be obtained.

3. **Clustering:** To demonstrate that the measured features (therefore phenotypes) accurately reflect the underlying genetic differences and also quantify the similarities of *C. elegans* mutant phenotypes, and to determine how phenotypic similarity as defined by our system correlates with the involvement of mutant gene products in a common biological function, a k-means based clustering method is developed in this dissertation.

4. **Classification:** We study and evaluate a variety of classification methods, and determined that classification based on Random Forests can not only outperform classification by a human expert dramatically, but can also provide insight about the phenotypes.

5. **Egg-laying behavior study:** One of the most important behaviors for the analysis of neuronal signal transduction mechanisms is egg-laying. In this dissertation, a series of image analysis methods are developed to detect egg-laying events automatically. As an integrated part of the system, we demonstrate that egg-laying events can also be efficiently studied by incorporating the extracted features so that the behavioral changes before and after egg-laying events can be discovered.

Chapter 2

Segmentation and Tracking

As in Figure 2.1, the image processing pipeline for studying *C. elegans* follows the typical machine vision steps such as video acquisition, object segmentation, object tracking. Depending on the individual application, the subsequent domain dependent processing steps varies for each application. For example, the domain dependent processes would include Object Classification, Behavior Classification, and Scene Description steps in a typical automated video surveillance task. In our applications, these subsequent domain dependent processings include clustering, classification, or specific behavioral studies. Because of the highly deformable nature of the *C. elegans* body, many of the conventional segmentation and tracking algorithms designed for rigid bodies are not applied. We have found that incorporating some unique constraints such as body width and movement into the segmentation and tracking system results in a better performance. Section 2.2 describes the data acquisition system. Section 2.3 and 2.4 elaborate upon the segmentation and tracking process. Finally, Section 2.5 summarizes the results.

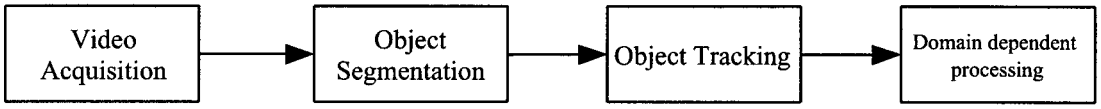


Figure 2.1: Typical machine vision system flow chart.

2.1 Strains and Culture Methods

Routine culturing of *C. elegans* was performed as described [10]. Animals were grown on standard nematode growth medium (NGM) seeded with E. coli strain OP50; all experiments were conducted in the presence of freshly-seeded OP50 lawns. All worms analyzed in these experiments were young adults; fourth-stage larvae were picked the evening before the experiment and tracked the following morning after cultivation at 22°. Since locomotion behavior shows reproducible and stereotyped changes following the transfer of an animal to a new culture dish [35], experimental animals were transferred to new plates and allowed to acclimate for 5 minutes before beginning tracking to ensure a valid comparison between experiments. We used wild type worms and fifteen mutants: N2-wild-type; *goa-1*($n1134$)-*Go*α subunit; *nic-1*($lj22$)-type1 glycosyltransferase; *unc-36*($e251$)-VGCC $\alpha 2/\delta$ subunit; *unc-38*($x20$)-nAChR α subunit; *unc-29*($x29$)-nAChR β subunit; *egl-19*($n582$)-L-type VGCC $\alpha 1$ subunit; *unc-2*($mu74$)-non-L-type VGCC $\alpha 1$ subunit; *tph-1*($mg280$) -tryptophan hydroxylase; *unc-63*($x13$)-nAChR α subunit; *dgk-1*($nu62$)-diacylglycerol kinase; *unc-43* ($e755$)-CaMKII; *dop-1*($ev748$)-D1 dopamine receptor; *flp-1*($yn2$)-Fa-related neuropeptide; *eat-4*($ky5$)-vesicular glutamate transporter; *cat-2*($e1112$)-tyrosine hydroxylase.

2.2 Data Acquisition System

C. elegans locomotion is tracked with a Zeiss Stemi 2000-C stereomicroscope mounted with a Cohu High Performance CCD video camera. A high-quality monochrome image acquisition board (National Instruments, Inc., IMAQ PCI/PXI-1409) is used to grab the image and convert to digital format. A computer-controlled tracker (Parker Hann Automation Corp., OEMZL4 stage controller) is also used at the same time to maintain the worms in the center of the optical field of the stereomicroscope during observation. (excluding the microscope, the components for this system cost approximately $\$10,000$). To record the locomotion of an animal, an image frame of the animal is snapped every 0.5 second for at least five minutes.

Among those image pixels with values less than or equal to the average value minus three times the standard deviation, the largest connected component is found. The image is then trimmed to the smallest axis-aligned rectangle that contained this component, and saved as eight-bit gray level data. The dimensions of each image, and the coordinates of the upper left corner of the bounding box surrounding the image are also saved simultaneously as the references for the location of an animal in the tracker field at the corresponding time point when the images are snapped. The stereomicroscope is fixed to its largest magnification (50 X) during operation. Depending on the type and the posture of a worm, the number of pixels per trimmed image frame varied. These smaller images reduce the storage space requirement by 90% on average. The number of pixels per millimeter is fixed at 312.5 pixel/mm for all worms.

2.3 Segmentation of the Worm Body

The segmentation process is presented in Figure 2.2, and a simplified graphic illustration is shown in Figure 2.3. The first operation is a local thresholding using a 5×5 moving window. The center pixel inside the moving window is assigned to 1 when the mean value of the window is less than 70% of the background pixel value or the standard deviation is larger than 30% of the mean value. Otherwise, the center pixel is assigned to 0 as background.

Next, the sequential algorithm for component labeling is used to remove unwanted small objects [40]. A morphological closing operator (binary dilations followed by erosions) [33] cleans up the spots inside the worm body. In order to avoid occasional false contours and exterior holes (formed by severe worm body bending as shown in Figure 2.4C) being filled by excessive closing operations, we also generate a reference binary image in parallel by filling the holes that have compactness (defined as $perimeter^2/area$) greater than 25 after local thresholding. Since exterior holes tend to be round, the compactness was used to avoid filling large exterior holes (> 100 pixels). Thus, two binary images are generated after local thresholding. The one with the closing operation sometimes contains excessive pixels, whereas the binary image after the hole filling operation tends to have fewer pixels than desired on other occasions.

The difference between these two binary images provides a good indication of whether or not the segmentation is successful and of which binary image is better to use. Figure 2.4 and Figure 2.5 illustrate the segmentation process. Comparing Figures 2.4 and 2.5, we see that the hole-filling result (Figure 2.4D) is more correct than the closing result (Figure 2.4C) for one case, but the closing result (Figure 2.5C) is superior

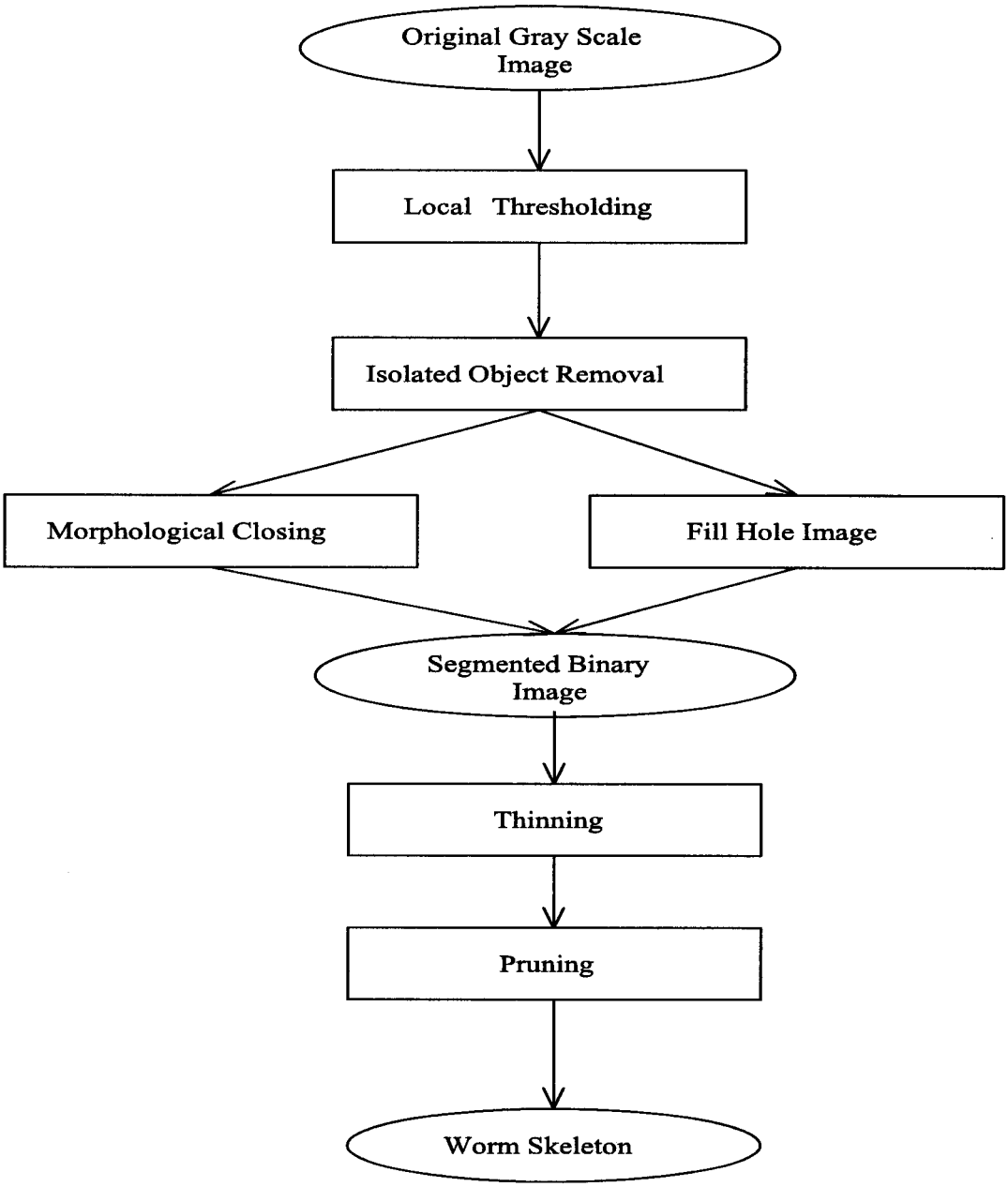


Figure 2.2: General description of the segmentation process.

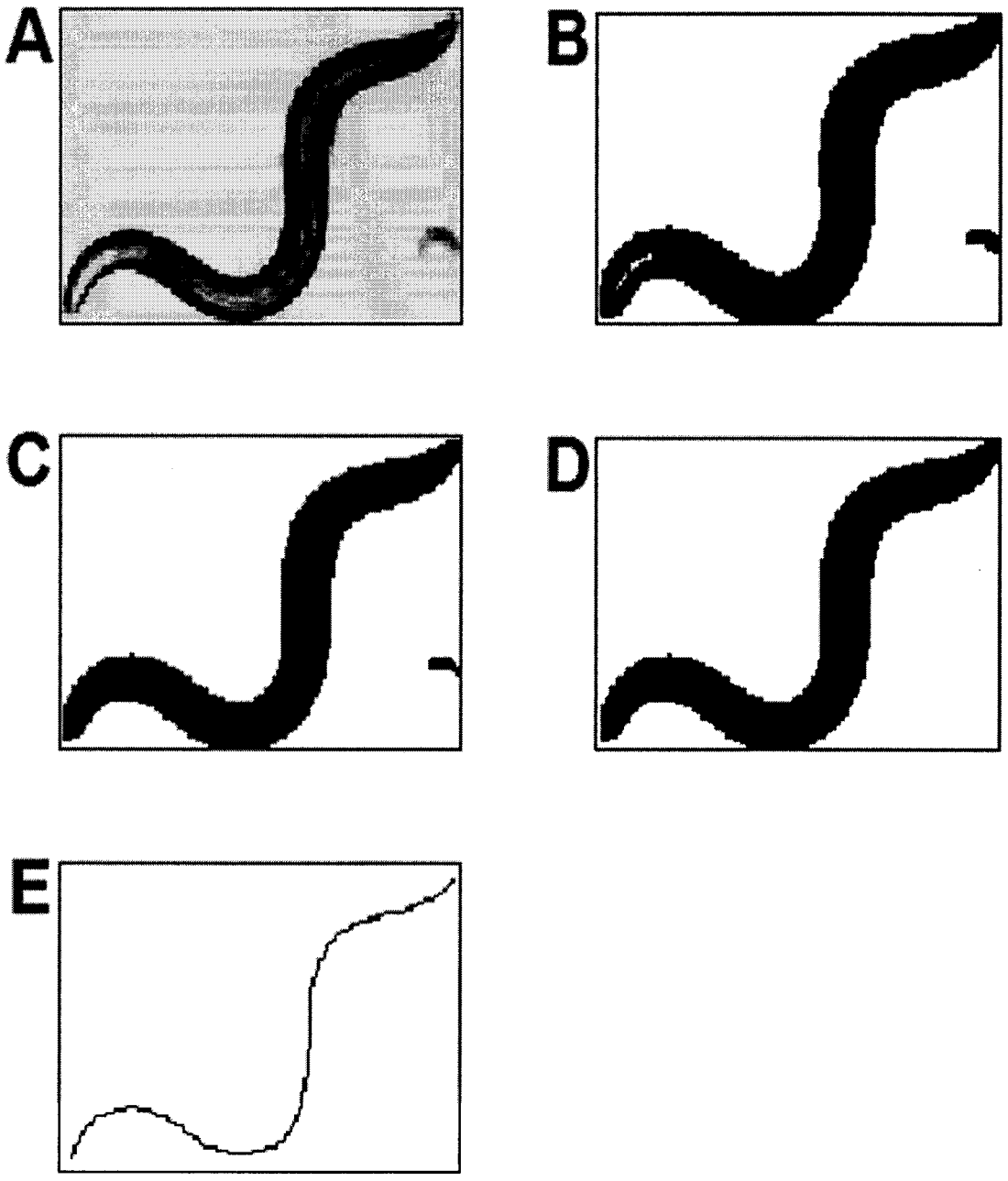


Figure 2.3: A simplified graphic illustration of the segmentation and skeletonizing process. (A) Original gray level image. (B) Binarized image. (C) Binary image after closing operation. (D) Binary image after small object removal. (E) Skeleton.

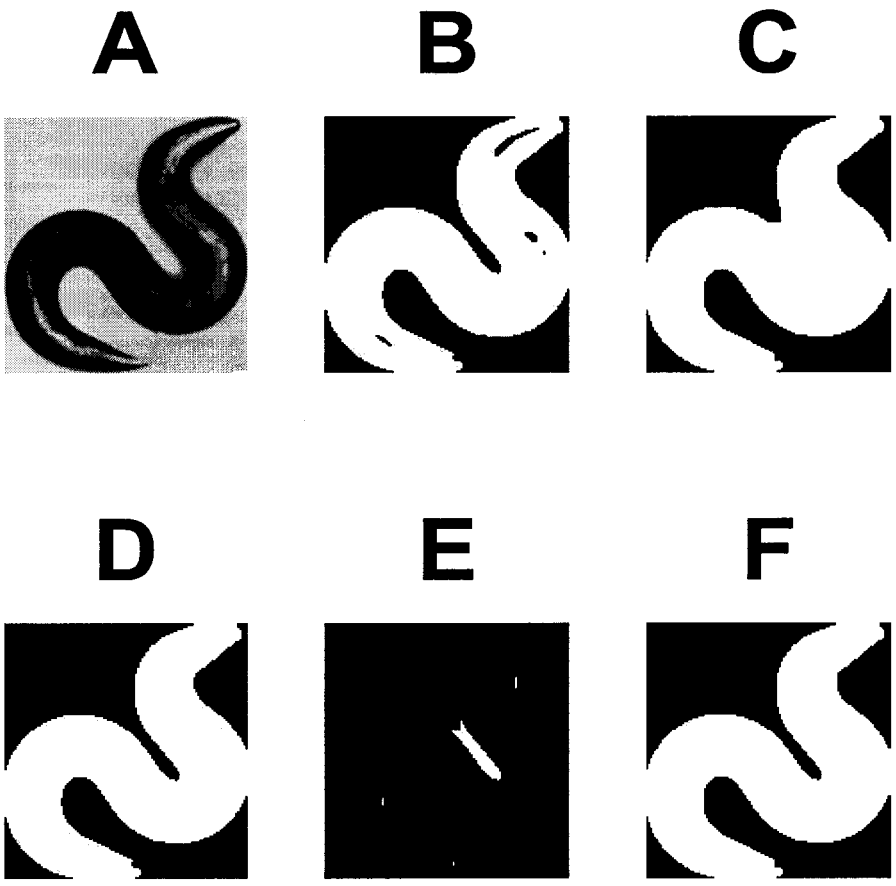


Figure 2.4: (A) Gray level image. (B) Original binary image after local thresholding operation. (C) Binary image after closing operation. (D) Binary image after hole filling operation. (E) Difference between C and D. (F) Final binary image with excess pixels removed.

to hole-filling (Figure 2.5D) in another case. By generating both hole-filling and closed versions of each image, and analyzing the difference between them, the algorithm is able to determine which pixels to include in the final binarization. For example, in Figure 2.4E, the excessive pixels caused by the closing operation are represented as the largest connected component in the difference image. The worm bodies along both sides of the arrow object's second eigen-direction are wider than 20 pixels (the typical worm body width), indicating the existence of excessive pixels. The second eigen-direction is the direction that is perpendicular to the principal component direction, and is calculated as follows:

$$\theta = \pi/2 + \tan^{-1}(2 * \sum_{k=1}^{n}(x_k * y_k)(\sum_{k=1}^{n} {x_k}^2 * \sum_{k=1}^{n} {y_k}^2)), \tag{2.1}$$

where (x_k, y_k) are the coordinates of the pixels in the object after centering.

In Figure 2.5, the missing pixels inside the worm body caused by thresholding are represented as the curved object in the difference image. The worm body portions along both sides of the curved object's second eigen-direction are much narrower than 20 pixels, indicating missing pixels inside the worm body.

Following binarization, a morphological skeleton is obtained by applying a skeletonizing algorithm [94]. Redundant pixels on the skeleton are eliminated by thinning. To avoid branches on the ends of skeletons, the skeleton is first shrunk from all its end points simultaneously until only two end points are left. These two end points represent the longest end-to-end path on the skeleton. A clean skeleton can then be obtained by growing out these two remaining end points along the unpruned skeleton by repeating a dilation operation (Figure 2.6A-D).

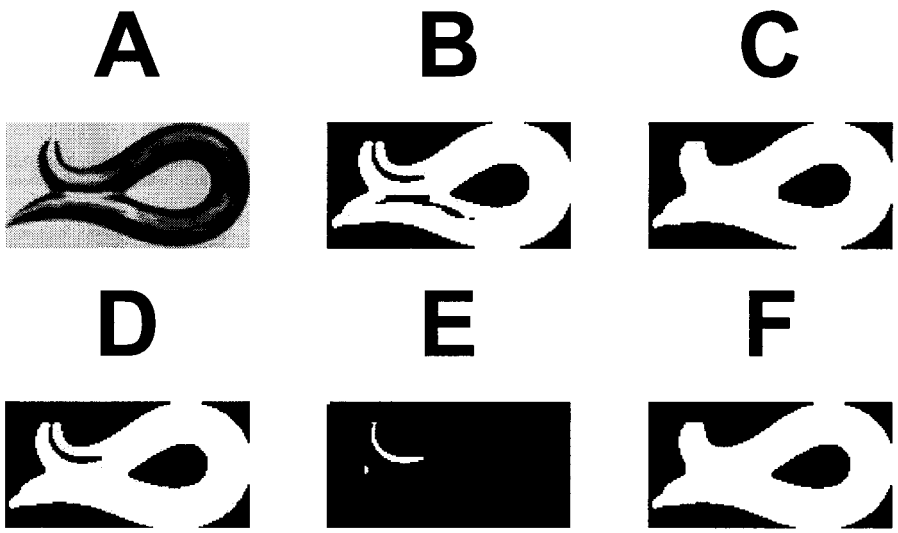


Figure 2.5: (A) Gray level image. (B) Binary image after local thresholding operation. (C) Binary image after closing operation. (D) Binary image after hole filling operation. (E) Difference between C and D. (F) Final binary image with inside cracks filled.

2.4 Tracking and Head and Tail Recognition

Even though a simple tracking system was able to follow the movement of the worm centroid, the head and tail information were not extracted in our earlier work. Because of the highly deformable nature of the worm's body, many conventional image matching and tracking algorithms do not apply to this problem. To address these problems, we have applied three spatial and temporal clues that human observers use to recognize the head and tail sections. Even though the entire worm body could travel a large distance (in camera coordinates) between two consecutive recording frames which were taken 0.5 seconds apart, the head and tail locations relative to the body centroid (worm body coordinates) tend to change little, much as a rigid body would behave (Figure 2.7). The

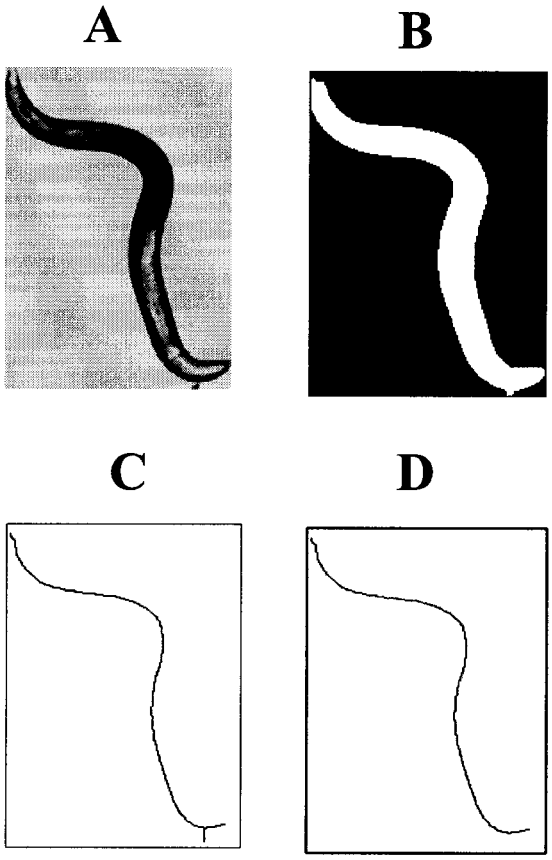


Figure 2.6: A-D illustrate the skeleton generating process. (A) Gray level image acquired from a video sequence containing the worm body and part of the track. (B) Corresponding binary image after thresholding. (C) Skeleton after applying skeletonizing algorithm and redundant pixel removal. (D) Clean skeleton after pruning.

other two clues are: the worm's tail area is darker than the head (having to do with fat distribution), and the head moves more frequently than the tail (having to do with foraging behaviors). The detailed procedure, illustrated in Figures 2.8, 2.9, 2.10, and 2.11, is as follows:

1. From recorded gray level images, the above segmentation procedure is applied. For each video frame, the gray level image and its corresponding binary image

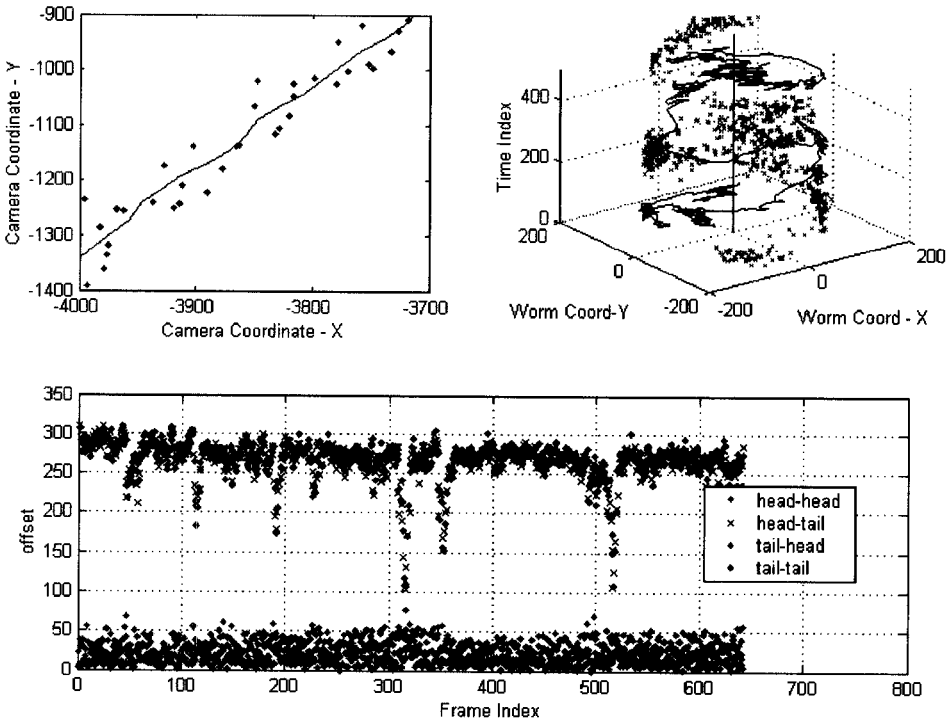


Figure 2.7: Worm movement characteristics and their usage for tracking. (A) A portion of track in camera view. Solid line represents the worm body centroid movement. x and · represent worm's tail and head location respectively, as they wiggle around the travel direction. (B) 3-D plot of head and tail movement in worm coordinates. The centroid movement is represented as the vertical line in the $(0, 0, t)$ location. The tail locations (+) are connected, showing the circular movement around the centroid. The head locations, marked by dots, tend to locate opposite the corresponding tail locations. (C) Bottom plot shows the location offset of heads and tails in worm coordinates for two consecutive frames. The head-head and tail-tail correspondences have smallest offsets of the four.

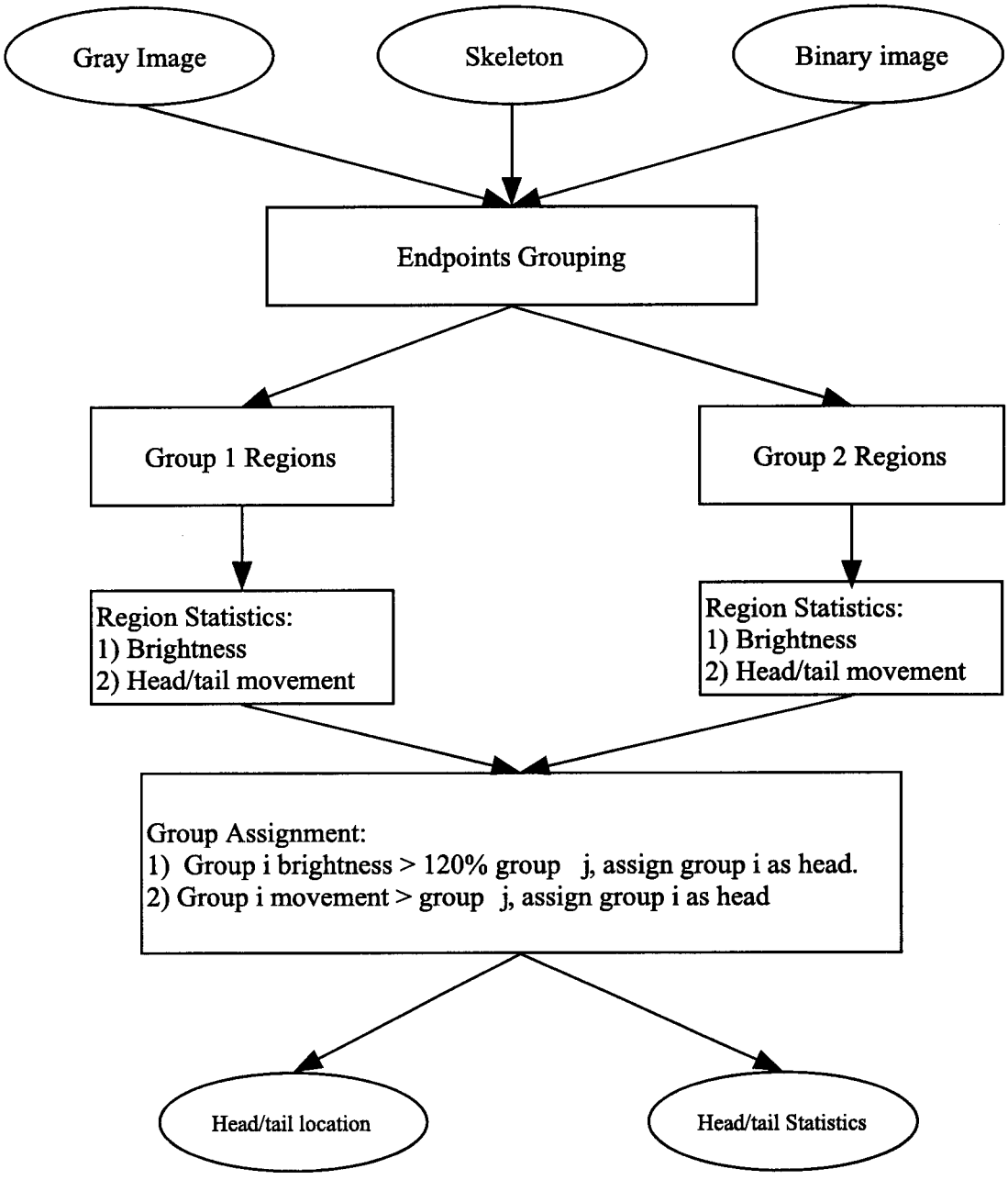


Figure 2.8: Tracking and head and tail recognition algorithm flow chart.

and skeleton are stored.

2. The two end points of the skeleton are potential head and tail locations. We assign the end points to two groups for each uninterrupted video segment according to the following rules:

   Let $endpt1x(t)$ denote the x coordinate of the end point in frame t that was assigned to group 1. Similar definitions hold for $endpt1y(t)$, $endpt2x(t)$ and $endpt2y(t)$. Now we use $endpt1(t) = [endpt1x(t), endpt1y(t)]$ to denote the vector of spatial coordinates for end point 1 in frame t. Let $endptA(t+1)$ and $endptB(t+1)$ denote the vectors of spatial coordinates for the two end points in frame $t+1$ that have not yet been assigned to group 1 or group 2. Let $(I, J) = \arg\min_{(i,j)} dist(endpti(t+1), endptj(t))$, $(i \in \{A, B\}, j \in \{1, 2\})$. Then $endptI(t+1)$ will be assigned to group J provided that $(\overline{I}, \overline{J}) \neq \arg\max_{(i,j)} dist(endpti(t+1), endptj(t))$, $(i \in \{A, B\}, j \in \{1, 2\})$. The condition statement is to avoid head and tail locations being accidentally flipped. If the condition is not met, the current frame is marked as "undecided" and the grouping process restarts from the next frame to avoid potentially spreading errors. Figures 2.9 and 2.10 illustrate the algorithm in details.

3. To isolate the head and tail sections from the rest of the body, we identify two points on the skeleton that are at 1/6 skeleton-length away from each end point. We compute the best fit line to 9-pixel-long segments from the skeleton list surrounding the two identified pixels. The lines are then rotated by 90 degrees to get perpendicular lines. Lines that are +5 and 5 degrees off from the perpendicular

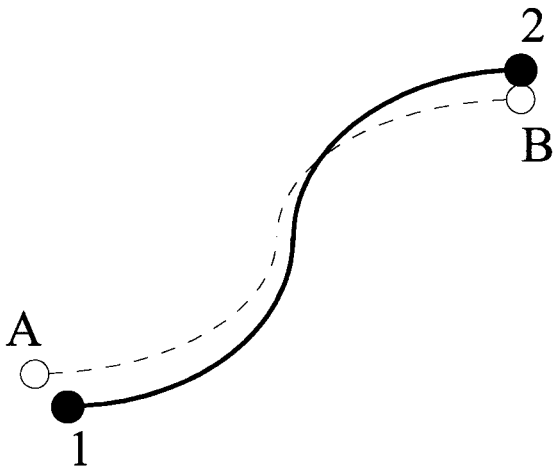


Figure 2.9: Skeleton $1-2$ is the skeleton from frame t. Skeleton $A-B$ is the skeleton in frame $t+1$. Of the four distances: $dist(1,A)$, $dist(1,B)$, $dist(2,A)$, $dist(2,B)$, the smallest is $dist(1,A)$. So endpoint A gets assigned to group 1 because $dist(2,B)$ is not the maximum of the four.

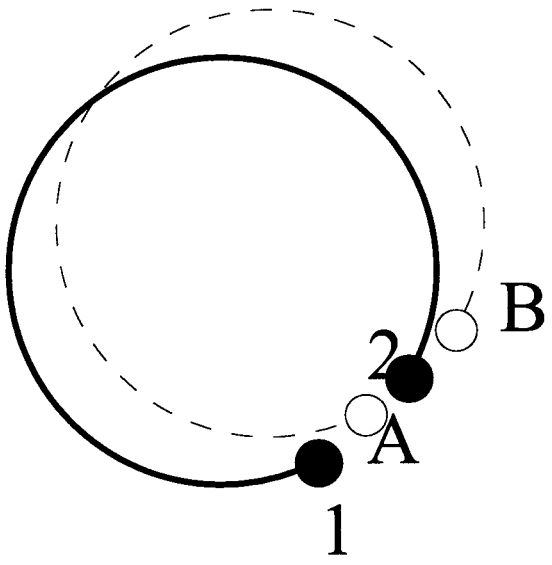


Figure 2.10: Now $dist(2,A)$ is the smallest of the four, but endpoint A does not get assigned to group 2 because $dist(1,B)$ is the largest of the four distances. This is an undecided frame, and the grouping process will re-initialize at this frame.

are also generated. The line with the shortest distance traversing the binary image is chosen as the separation line between the head/tail and the rest of the body. The end sections are separated from the rest by deleting binary pixels along the separation lines.

4. Using the binary image and the end point locations as an index to the gray level image, we calculate the median brightness of the two end sections for each frame. The means of these values for group 1 and 2 are calculated for the segment. If the difference between these two mean values is at least 20% of the larger mean value, the group with the higher average brightness value is labeled as the head.

5. Mutant types with digestive abnormalities have smaller brightness differences between head and tail. For these (brightness difference $\leq$ 20%), a secondary decision rule is introduced to compare the local movement distance for the two end points. The group with higher total movement distance is labeled as the head. This procedure was applied independently for each video segment. Segments are separated by missing frames, failed segmentation, or undecided frames.

2.5 Results

The tracking and head and tail recognition procedure was tested on 161 5-minute video sequences (sampled at $2Hz$) from 16 mutant types including more than $111,000$ image frames. The videos were played back with the worm's tail marked by the algorithm for a human observer to verify. Experimental results are shown in Table 2.1. The method

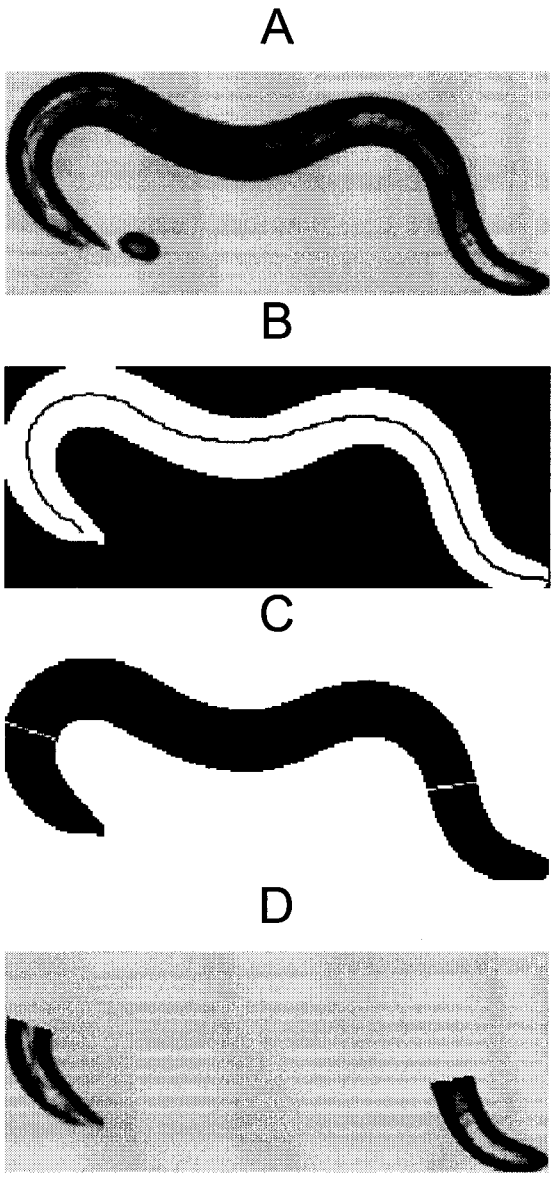


Figure 2.11: Image processing and head/tail extraction procedure. (A) Original gray level image. Notice there is an egg object nearby that needs to be removed by the cleaning operation. (B) Binary image after segmentation and cleaning. The worm skeleton generated from thinning, pruning process is superimposed on the binary image. Two end points of the skeleton are candidates of head and tail locations. (C) Two perpendicular lines (to skeleton line fitting) at 1/6 of skeleton location. Deleting the pixels along these separation lines divides the worm body into head, tail and middle sections. (D) Head and tail sections of the gray level image can be easily obtained by indexing cutoff portions from the binary image from (C).

produces excellent results as the average correct identification rate is around 98%. For 13 of the worm types the correct identification rate is 100%. For one mutant type (*unc-36*), the tail is often lighter than the head, leading to a 24% misidentification rate for this one type.

2.6 Summary

In this chapter, we described the data acquisition system and novel segmentation and tracking algorithms developed to analyze the mutant movement videos. Results (Table 2.1) show that our automated algorithm achieves high performance.

Part of this chapter has appeared in the following publications.

- W. Geng, P. Cosman, C. Huang, and W. R. Schafer. "Automated Worm Tracking and Classification." *Proc. of the 37th IEEE Asilomar Conference on Signals, Systems and Computers*, pp. 2063-2068, Pacific Grove, CA, November 2003.
- W. Geng, P. Cosman, C. Berry, Z. Feng and W.R. Schafer, "Automatic Tracking, Feature Extraction and Classification of *C. elegans* Phenotypes", *IEEE Transactions on Biomedical Engineering*, in press, 2004.

I was the primary researcher and the co-author. Dr. Pamela C. Cosman and Dr. William R. Schafer directed and supervised the research which forms the basis for this chapter.

Table 2.1: Head and tail identification results. Data were collected from 161 5-minute video sequences ($2Hz$) from 16 distinct mutant types. First column shows the mutant type. Second column shows the total number of frames in the videos. The number of frames that had head recognized as tail due to the tail section being lighter is listed in column 3. The number of frames that had head recognized as tail due to grouping errors is listed in column 4. The average error rate is around 2% for 111, 233 frames tested.

Worm type	**Total Frames**	**Recognition Wrong**	**Grouping Wrong**	**Error Percent**
goa-1	6193	0	1	0
unc-29	4679	0	0	0
w.t.	6057	5	0	0
egl-19	5503	1	0	0
cat-2	4908	0	5	0
dop-1	4954	0	0	0
dgk-1	4892	0	0	0
eat-4	5014	0	0	0
flp-1	4942	0	30	0
nic-1	11817	1	0	0
unc-38	4853	3	0	0
unc-63	4926	0	0	0
unc-43	9908	251	0	0.03
tph-1	10552	51	0	0
unc-2	10361	69	0	0.01
unc-36	10559	2613	3	0.24
Total	**111233**	**3030**	**33**	**0.02**

Chapter 3

Feature Extraction

Many of the *C. elegans* mutants have distinct behaviors, posture, or morphological characteristics. For example, *nic-1* is short and fat, while *egl-19* has an elongated body shape. Some mutants such as *unc-29* and *unc-38*, however, have very similar characteristics that are difficult to distinguish by eye. To take full advantage of the computer vision system illustrated in Chapter 1, we design a comprehensive set of 253 features characterizing the mutant phenotypes. These features generally fall into several categories as shown in Figure 3.1. Section 3.1 describes the feature extraction principles. The size related features are described in Section 3.2. Section 3.3 explains the body shape and posture related features. Section 3.4 illustrates the movement related features. Section 3.5 covers the brightness features. Section 3.6 touches on the complex behavior related features. We conclude the chapter with a summary Section 3.7.

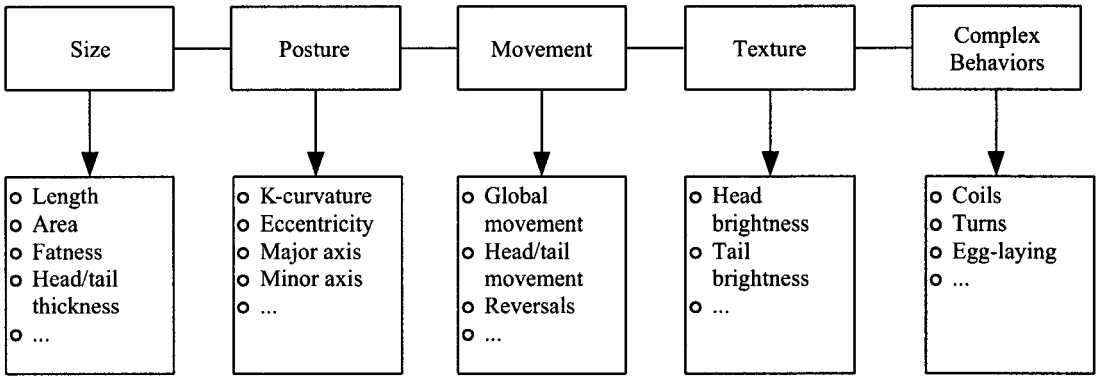


Figure 3.1: Feature categories and typical representatives.

3.1 Feature Extraction Overview

All of the software for binarization, skeletonization, and feature extraction is coded either in C or MATLAB and implemented on UNIX machines. Some features (e.g., the area of the worm, that is, the number of pixels which make up the single binary object in the frame) could be computed on a single frame; these are computed for all 600 frames in the sequence. The average value, the maximum value and the minimum value are then computed for these 600 measurements. Other features could not be extracted from a single frame, for example, the movement between two frames, or the movement within 10 seconds (20 frames). Since there are 600 frames total in a sequence, the movement between two frames could be computed 300 times if we take pairs of frames in a non-overlapping fashion, or it could be calculated 599 times taking pairs of frames in a sliding window or overlapping fashion. Likewise, for the movement within 20 frames, we could compute 581 values for overlapping 20-frame intervals. Quantities of this type are calculated in a sliding window fashion. The average, max, and min are computed from this set of numbers.

Some of the maximum and minimum values are outliers introduced by noise or errors during image capture and processing. To avoid using these extreme values, it is more useful to summarize the group statistics with such quantities as the $90th$ and $10th$ percentile values out of the population of 600 numbers. For the remainder of this dissertation, the terms max and min are used to denote the $90th$ and $10th$ percentile values. The measured features include the minimum, maximum, and average values of the following: distance moved in 0.5, 5, 10, 15, 20, 25, 30 seconds and 5 minutes., number of reversals in 10, 20, 40, 60, 80, 100, 120 sec and 5 minutes, worm area, worm length, width at center and head/tail, ratio of width to length, fatness, eccentricity and lengths of major/minor axes of best-fit ellipse, height and width of minimum enclosing rectangle (MER), ratio of MER width and height, ratio of worm area to MER area, angle change rate, head/tail/center brightness, local head/tail/center movement relative to centroid, and head-centroid-tail angle. The area, angle change rate, and movement features are calculated separately for the head, tail, center, and entire worm body. We now describe in detail how several of these features are extracted from the image data. For a complete list of the features, descriptions, and statistics, see Table A.1.

3.2 Body Size

1. Body Size

   The worm's body size is obtained by counting the number of "on" pixels in the binary image. Similarly, the size of the worm's head, tail and middle part are obtained from the cutoff areas of the tracking procedure outputs (See Figure 3.2).

2. Length

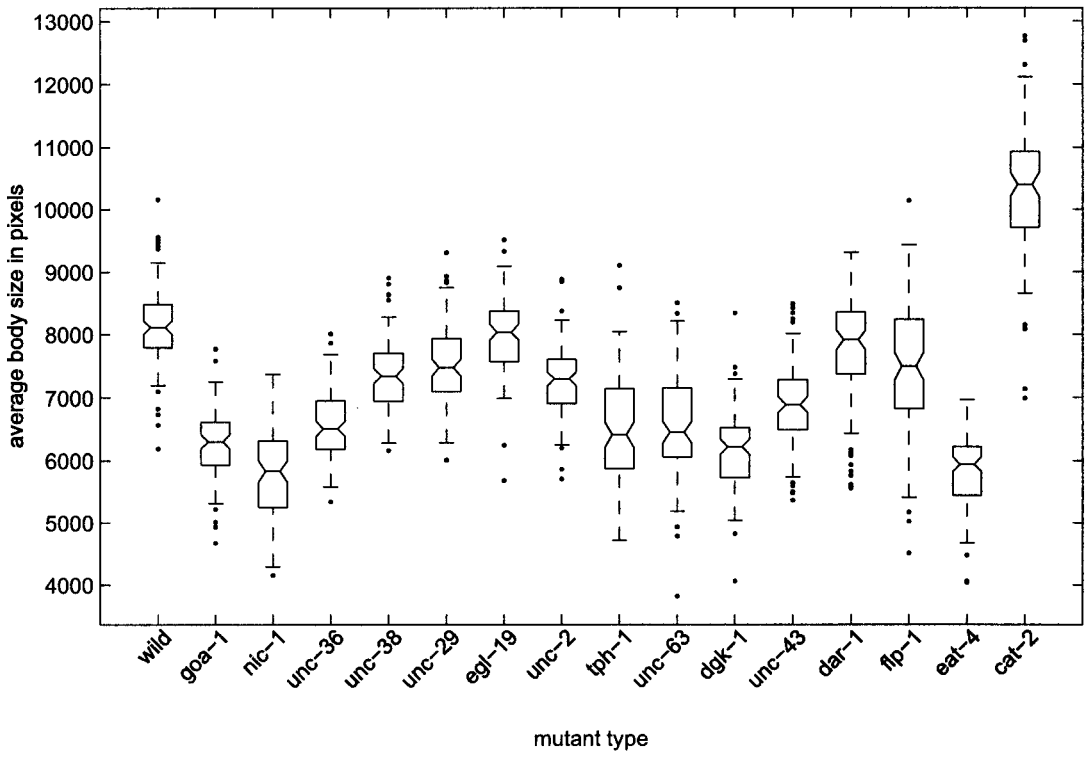


Figure 3.2: Average body size of 6,000 frames for each mutant type. The unit is number of pixels. In all cases, the box extends from the first quartile (25th percentile) to the third quartile (75th percentile), and the horizontal line within the box indicates the median. The lower and upper error bars indicate 10th and 90th percentiles respectively; each outlier is indicated with a dot symbol. Note the *cat-2* has significantly larger body size than the rest of the mutant types.

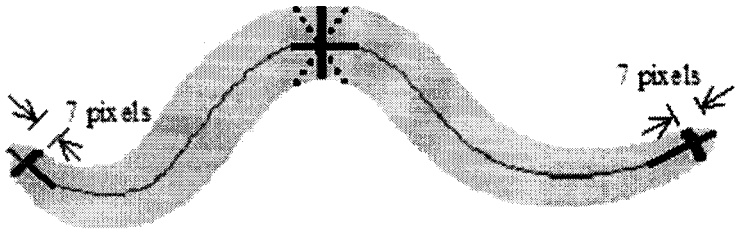


Figure 3.3: Width measurement.

The animal's length is defined as the number of pixels in the image skeleton from the skeletonizing output.

3. Thickness, width, and fatness

The worm thickness (width/length) is measured at the center, head, and tail positions of the worm skeleton (the center position is the value at the center of the skeleton pixel list; the head and tail positions are defined as the position which is 7 pixels away from head and tail end points identified by the tracking algorithm).

In order to measure the center width, we first take a 9-pixel-long segment from the middle of the skeleton list, and compute the best fit line for the segment by a line fitting algorithm. Then we rotate the line by 90 degrees to get a perpendicular line to it (see Figure 3.3). We traverse the perpendicular line in both directions from the center position until we reach the edges of the worm body, and then compute the distance between the two edges. We also rotate the perpendicular line by -5 and +5 degrees, and measure the width in those two directions. The minimum value of the three measurements is considered to be the center width.

Similarly, in order to measure the head/tail width, we take two 9-pixel-long segments from each end of the skeleton list. After getting the best fit lines for the segments, we find the designated head/tail position by going back 7 pixels from the end of the worm body along the best fit line. Then we compute the width at these two measuring positions (one at each end) by traversing the perpendicular lines to the best fit lines. The minimum value of the two measurements is considered to be the head/tail width. We also define the worm's fatness as the ratio of worm area to length.

3.3 Body Shape

1. Angle change rate

   The angle change (Figure 3.4), an important feature for distinguishing different worm types, is defined as

$$R = \frac{1}{n-1} \frac{\sum_{i=1}^{n-1} \theta_i}{L}, \tag{3.1}$$

   where L is the worm length, $\theta_i = \tan^{-1} \frac{y_{i+2}-y_{i+1}}{x_{i+2}-x_{i+1}} - \tan^{-1} \frac{y_{i+1}-y_i}{x_{i+1}-x_i}$, and $(x_i, y_i), (x_{i+1}, y_{i+1})...$ are the location of consecutive points that are 5 pixels apart along the worm skeleton, and n is the number of such points along the skeleton. A larger angle change rate means that a worm has sharper body bends. Figure 3.5 (A-B) shows typical skeletons from two different mutant types. The angle change rate is 15.51 for the *unc-2* skeleton in Figure 3.5A, compared with for 8.45 for the *egl-19* skeleton in Figure 3.5B. The angle change rates are also calculated separately for head, tail, and center regions. The angle change rates of all worm samples are shown in Figure 3.6.

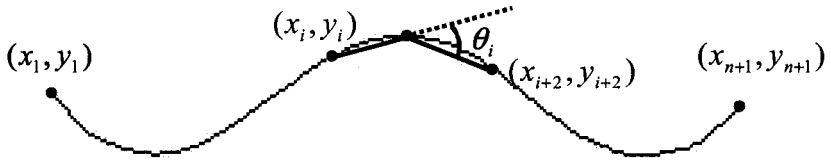


Figure 3.4: Angle change rate calculation.

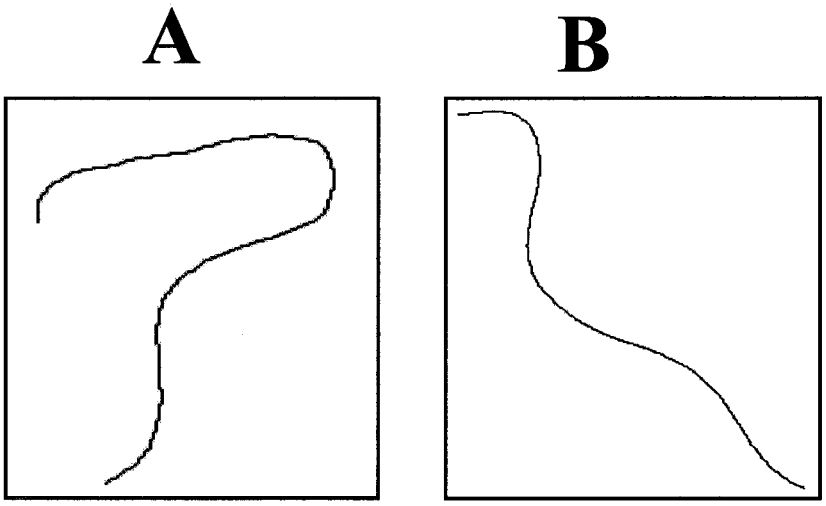


Figure 3.5: Comparison of the skeletons from two mutant types. (A) a typical *unc-2* skeleton. (B) a typical *egl-19* skeleton.

2. Best fit ellipse and minimum enclosing rectangle

   The best fit ellipse calculation follows the eigen direction calculation for the object, as shown in Figure 3.7,

$$\lambda_1 = \frac{\sum_{k=1}^{n} x_k{}^2 + \sum_{k=1}^{n} y_k{}^2 + \sqrt{\sum_{k=1}^{n} x_k{}^2 - \sum_{k=1}^{n} y_k{}^2 + 4 * \sum_{k=1}^{n} x_k y_k}}{2} \qquad (3.2)$$

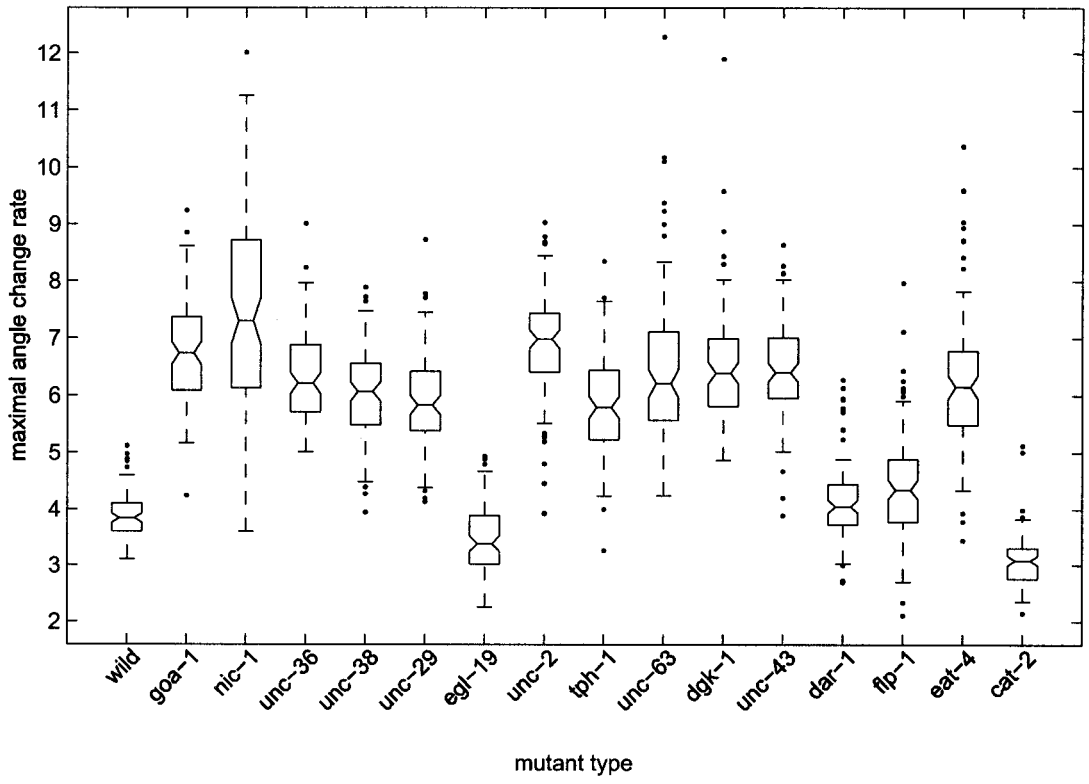


Figure 3.6: Average angle change rate of 6,000 frames for each mutant type. Within each frame, the maximal angle change is used for calculation. In all cases, the box extends from the first quartile (25th percentile) to the third quartile (75th percentile), and the horizontal line within the box indicates the median. The lower and upper error bars indicate 10th and 90th percentiles respectively; each outlier is indicated with a dot symbol. Note the *egl-19* and *cat-2* have significant lower angle change rate than the rest of the mutant types, indicating a less curvaceous body posture during movement.

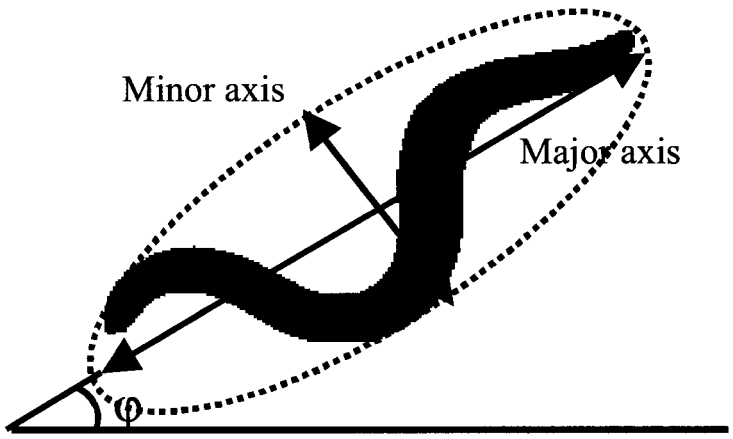


Figure 3.7: Best fit ellipse.

$$\lambda_2 = \frac{\sum_{k=1}^{n} {x_k}^2 + \sum_{k=1}^{n} {y_k}^2 - \sqrt{\sum_{k=1}^{n} {x_k}^2 - \sum_{k=1}^{n} {y_k}^2 + 4 * \sum_{k=1}^{n} x_k y_k}}{2} \tag{3.3}$$

where (x_k, y_k) are the coordinates of the pixels in the object after centering. The major axis length λ_1 and minor axis length λ_2 are the two eigenvalues of the shape. After rotating the shape according to the angle of the principal eigenvector, the minimal enclosing rectangle (MER) is the rectangular box surrounding the shape as shown in Figure 3.8. The eccentricity is defined as the ratio of the distance between the foci of the ellipse and its major axis length. The MER and best fit ellipse give an indication of whether the worm tends to take on elongated positions with low amplitude waves, or, on the contrary, tends to have deeper body bends or looped body positions.

3. Symmetry

To measure the unbalanced muscle behavior of uncoordinated mutants, we characterize the way a worm body deviates from a perfect symmetry. These features

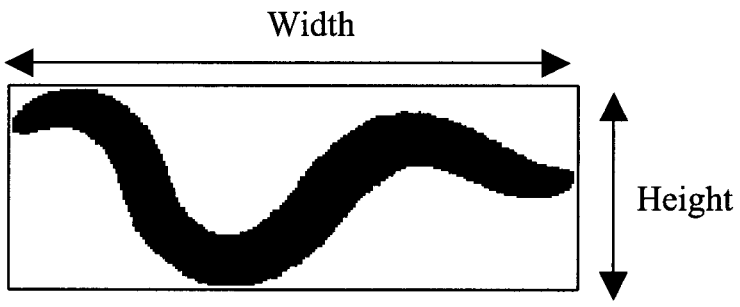


Figure 3.8: Minimal enclosing rectangle.

include the amplitude (defined as the absolute distance from points on the skeleton to the line connecting the head and tail), the sum of signed distances to the line connecting the head and tail, the angle between the line connecting head to centroid, and the line connecting tail to centroid, and the distances between head and centroid, and between tail and centroid (see Figure 3.9A).

3.4 Movement

1. Global Movement

Global movement measures the distance and speed of the worm's entire body movement. It can be measured simply by following the trajectory of the animal's centroid over time. To measure speed, the centroid position data are sampled over a constant time interval, and the worm's displacement is proportional to its average speed during that interval. Interval durations used in our experiments range from 0.5 seconds (1 frame), to 5 minutes (the total time of observation). The movement measured at 0.5 second intervals is shown in Figure 3.10.

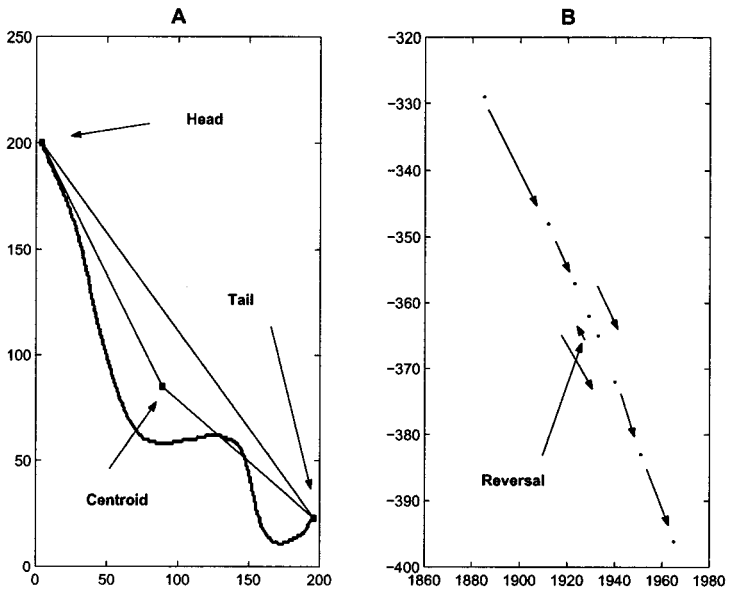


Figure 3.9: Feature examples. (A) Worm skeleton, head, tail, and centroid locations. The length, angle of head to tail, head to centroid, tail to centroid lines provide symmetry information. Worm amplitude can also be measured. (B) A portion of track left by centroid. The reversal location is marked.

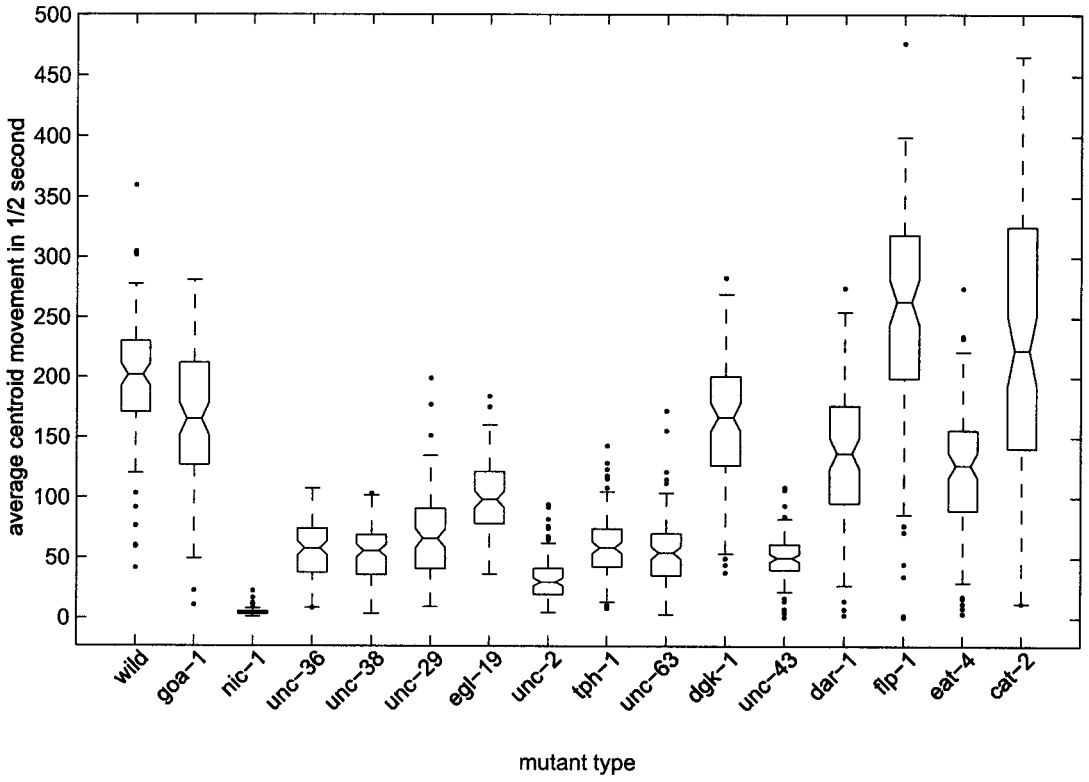


Figure 3.10: Average centroid movement in 0.5 second. In all cases, the box extends from the first quartile (25th percentile) to the third quartile (75th percentile), and the horizontal line within the box indicates the median. The lower and upper error bars indicate 10th and 90th percentiles respectively; each outlier is indicated with a dot symbol. Notice *nic-1* has significantly slower movement.

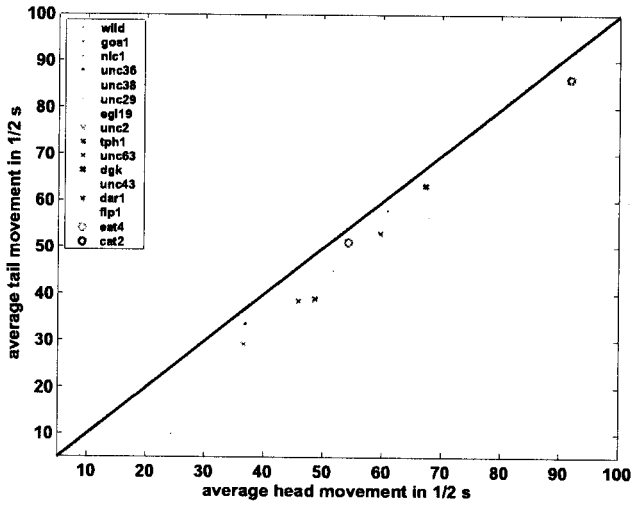


Figure 3.11: Average head vs. tail movement in 0.5 second averaged for all samples in each mutant type. The units are in pixel displacement. Notice all centers are in the lower right region indicating the head moves slightly more than the tail. Also notice both *flp-1* head and tail move much faster than *nic-1*.

2. Local Movement

Besides the features characterizing the global movement using the absolute distance traveled by the worm body centroid over various fixed time intervals, we also measure the relative offset of the head with regard to the centroid across the frames as an indication of the worm's head movement. This offset is defined as the movement of the head when the worm centroids are aligned on top of each other from one frame to the next. The tail movement is also measured. These measurements calculate how much the individual body parts move relative to the rest of the body (see Figure 3.11).

3. Reversals

Reversals are interesting characteristics during movement. They are characterized by the distance and frequency of the worm moving back into the recent previous path. We keep a moving window to record the previous 20 centroid locations. A reversal is detected when the new centroid is closer to any of the 19 previous centroid locations than to the most recent past as shown in Figure 3.9B.

3.5 Brightness

Variations in fat distribution and absorption of nutrients cause some mutant types to become more transparent than others. The transparency can be measured by the median pixel value of the head, center, tail, and whole body regions as shown in Figure 2.11. The head and tail brightness features are shown in Figure 3.12.

3.6 Behavioral Features

The amount of time a worm spends in a coil as well as how often it coils are unique behavioral characteristics of several types of worms. A coiled body posture creates a hole in the image where the worm loops or touches itself. To identify coiled postures, we search for holes in the worm image by performing connected component labeling on the inverted image [40]. Counting up the number of connected objects will always give a value of at least one for the background; thus the number of holes is equal to the number of connected components minus one. In our subsequent analysis, we count the number of frames the worm is in a coiled posture as well as the number of times the worm switches from a non-coiled to a coiled posture (i.e., the number of runs). The

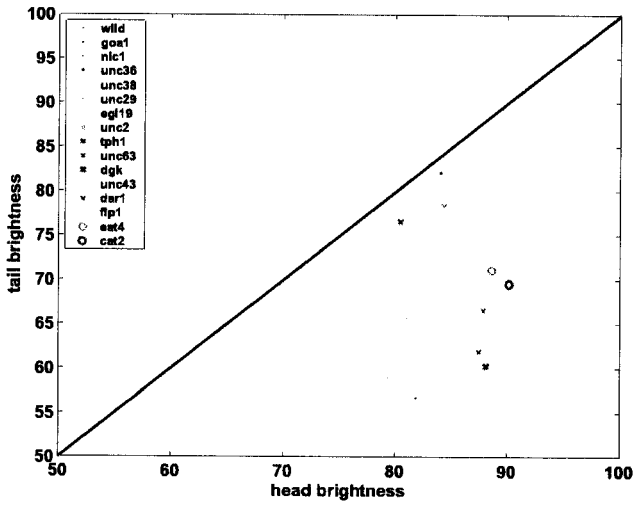


Figure 3.12: Head vs. tail brightness averaged for all samples in each mutant type. The units are in pixel values $[0 - 255]$. Notice all centers are in the lower right region indicating that the head is significantly brighter than the tail.

length of time the worm remains coiled is characterized by finding the minimum, maximum, and average of the run lengths. We also count the total number of times the worm briefly loops and the total number of frames the worm has multiple loops.

There are other behavior features that are potentially useful for defining phenotypes. These features include defecation, pharyngeal pumping, and social behaviors. Due to the complexity of the behaviors, these are left to future research.

3.7 Summary

In summary, the 253 features include 131 morphological features (thickness, fatness, MER, Angle Change Rate, etc), 75 speed features (min, max and average speed over 1, 5, 10, 20, 30, 40sec, etc), 35 texture features (head, tail, center brightness, etc) and 12

other behavioral features (rate of reversals, omega shape, looping, etc).

The feature extraction is a crucial step in defining mutant behavioral phenotypes. It is one of the key objectives of our research. A good set of features ultimately controls the performance of the clustering and classification results. A good segmentation and tracking design will not only allow more features to be calculated, but also reduces noises that are introduced in the measurement stage.

In the subsequent chapters, we will discuss the clustering, classification, and egg-laying studies using these features. The important features will also be identified in these chapters. Table A.1 lists the descriptions and statistics for all 253 features.

Chapter 4

Natural Clustering

Natural clustering (or grouping, cluster analysis, data segmentation) is a set of techniques used to understand the complex natural multivariate relationships among the data. Grouping data in clusters, such that those objects within each cluster are more closely related to one another than to objects assigned to different clusters, can provide an informal means for assessing dimensionality, identifying outliers, and suggesting interesting hypotheses concerning relationships.

In this chapter, we develop a clustering procedure to demonstrate the features described in Chapter 3, characterize the mutant phenotypic behaviors and illustrate the correlations between phenotypes and the underlying genotypes. We start with an overview of common clustering procedures. Then feature scaling, clustering, and stopping rules are discussed in Sections 4.3-6. We conclude this chapter with a summary section 4.7.

4.1 Natural Clustering Overview

The common clustering techniques generally fall into hierarchical and nonhierarchical categories. Hierarchical clustering techniques proceed by either a series of successive mergers or a series of successive divisions. Typical methods in this category include Ward's, agglomerative, and divisive methods [36] [20][45]. In these methods, there is no provision for a reallocation of samples that may have been "incorrectly" grouped at an early stage.

Nonhierarchical clustering techniques are designed to group samples into a collection of K clusters. The number of clusters, K, may either be specified in advance or determined as part of the clustering procedure. Nonhierarchical methods start from either an initial partition of samples into groups or an initial set of seed points. The most popular method in this category is the k-means algorithm [57] and its variations such as k-mediods [46], Self-Organizing Maps(SOM) (a constrained version of K-means) [48], and EM (a soft boundary version of K-means) [19]. Unlike the hierarchical clustering, the samples can be reallocated to other clusters as the number of clusters K changes.

There are also many methods designed for displaying transformed multivariate data in low-dimensional space. The popular methods include Principle Component Analysis (PCA) [20], Multidimensional Scaling (MDS) [45], Independent Component Analysis (ICA) [39], and their variations. Whenever multivariate observations can be presented graphically in two or three dimensions, visual inspection can greatly aid interpretations.

Since all the features measured in Chapter 3 are numeric and represent distance

measures, a k-means based clustering algorithm is preferred in our study (see Figure 4.1).

4.2 Strains

The alleles and predicted products of the genes used in these experiments are 8 mutant types as follows: *unc-38* (x20); nicotinic acetylcholine receptor alpha-subunit (null allele); *unc-29* (x29), nicotinic acetylcholine receptor non-alpha-subunit (null allele); *goa-1* (n1134), G-protein-alpha-subunit (strong loss-of-function allele); *unc-36* (e251), voltage-gated calcium channel alpha-2-subunit (strong loss-of-function allele); *unc-2* (mu74), N-type voltage-gated calcium channel alpha-1-subunit (null allele); *egl-19* (n582), L-type voltage-gated calcium channel alpha-1-subunit (partial loss-of-function allele); *nic-1* (lj22), type 1 glycosyltransferase (partial loss-of-function allele).

4.3 Normalization of Feature Data

Standardizing inputs on a set of carefully selected features plays an important role in pattern recognition. Since our features are measured in different units, it is necessary to normalize them on a common scale to avoid one feature dominating others. The outliers introduced by noise and errors during the feature extraction process tend to give false clusters in clustering analysis; thus, the scaling method also needs to be carefully selected to suppress outliers. We evaluate three standard normalization methods: Min-max (linear transformation of the original input range into $[-1, 1]$), Zscore (defined as Equation 4.1),

$$x = \frac{f - mean(f)}{stdev(f)} \tag{4.1}$$

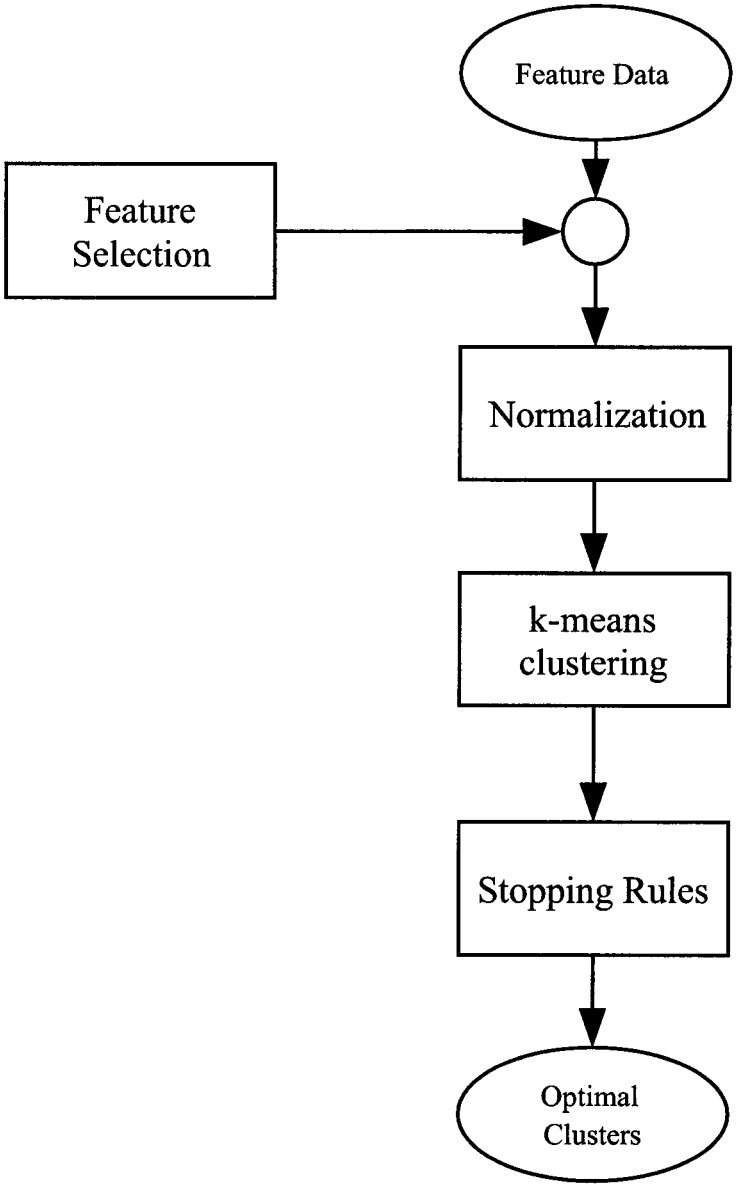


Figure 4.1: Clustering flow chart.

where f is the original input feature, and sigmoidal method ([34]). The Sigmoidal method is defined as in Equation 4.2,

$$y = \frac{1 - e^{-x}}{1 + e^{-x}} \tag{4.2}$$

where x is the output of Zscore scaling. Figure 4.2 shows a comparison among different scaling methods and feature subsets. The blue, red, and magenta curves represent the 1 Nearest Neighbor (1-NN) classification error rate using Min-Max, Sigmoidal, and Zscore scaling, respectively. The error is an average of 50 trials of 10-fold cross-validation result for each method. The features are selected from the first few Principal Components (PCs) of the entire 253 input features. All three scaling methods achieve similar performance, with the sigmoidal and Min-Max methods slightly outperforming the Zscore. The fact that the error curves level off indicates most of the useful information for classification is heavily concentrated in the very first few PCs. The black curve shows the same cross-validation test but with a subset of features selected by a backward elimination method (see Section 4.5). The black curve also shows the adverse effect of increasing error rate with more features added. The Sigmoidal method is chosen because it obtains a better balance of limiting outliers and equalizing feature variance on our dataset given our goal of natural clustering.

4.4 Representation of Phenotypic Patterns in Multidimensional Feature Space

To visualize the phenotypic patterns as defined by the selected parameters, we use principal component analysis (PCA) to obtain a two-dimensional projection of our 253-

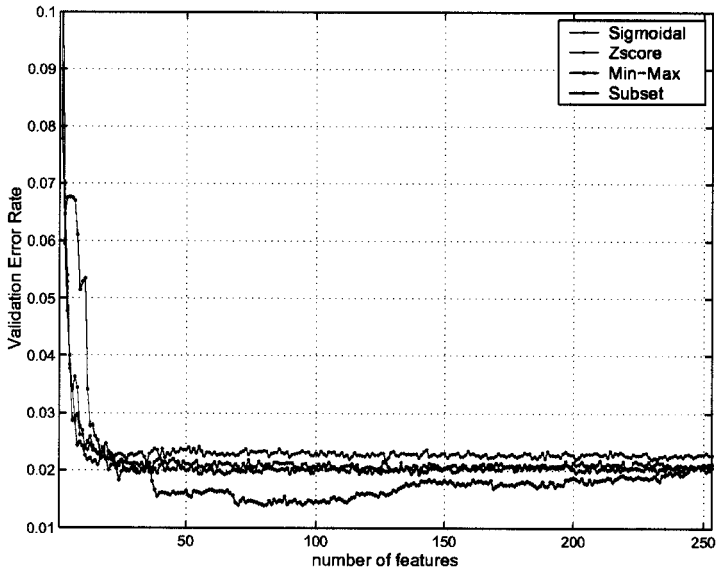


Figure 4.2: Comparison of three scaling methods and feature subset.

dimensional data. We observe in Figure 4.4 that the data points for each mutant type form a data cloud that occupies a specific region of feature space. To investigate the distribution of these clouds, we compute the centroid for each mutant type (i.e., the center of the data cloud as measured by Euclidean distance), and consider this to be the prototype for that mutant type as shown in Table 4.1.

Consistent with our expectation, the majority of the worm samples for each type are closer to its respective prototype than are samples from other mutant types as shown in Table 4.2. Interestingly, the distances between the centers (see Table 4.1) of the mutant data clouds also show a strong correspondence to the similarities between the described mutant phenotypes. For example, the clouds for the 4 mutants (*unc-2*, *unc-36*, *unc-29*, and *unc-38*) described in the literature as "kinkers" map close together in feature space, whereas the wild-type, *goa-1*, *nic-1* and *egl-19* clouds are more widely

Table 4.1: Euclidean distance between prototype centers (cluster centers) measured in 253-dimension feature space. Wild-type–*nic-1* are the furthest; *unc-29*–*unc-38* and *unc-2*–*unc-36* are among the closest.

	Wild type	*goa-1*	*nic-1*	*unc-36*	*unc-38*	*unc-29*	*egl-19*	*unc-2*
Wild type	-	6.5	**11.0**	8.4	7.0	5.7	5.9	8.7
goa-1		-	9.0	6.6	6.9	5.8	8.5	7.1
nic-1			-	6.6	5.6	8.0	10.6	6.6
unc-36				-	5.2	5.1	6.1	**3.6**
unc-38					-	**3.5**	6.8	4.1
unc-29						-	5.2	4.2
egl-19							-	7.1
unc-2								-

Table 4.2: 10-fold cross-validated classification result using 1-Nearest Neighbor classifier. The percentage number shows the probability the mutant type specified in the row is classified as being the mutant type specified in the column by this classifier. A subset of 39 features achieves a similar performance to the full set.

	Wild type	*goa-1*	*nic-1*	*unc-36*	*unc-38*	*unc-29*	*egl-19*	*unc-2*
			Using	253	features			
Wild type	**1.00**	0	0	0	0	0	0	0
goa-1	0.01	**0.94**	0	0.01	0.02	0.01	0	0
nic-1	0	0	**0.99**	0	0	0	0	0.01
unc-36	0	0	0	**0.84**	0.05	0	0	0.11
unc-38	0	0	0.01	0	**0.80**	0.19	0	0
unc-29	0	0	0.01	0	0.37	**0.60**	0	0.02
egl-19	0	0	0	0.03	0.01	0.01	**0.95**	0
unc-2	0	0	0	0.08	0.04	0	0.01	**0.87**
			Using	39	features			
Wild type	**1.00**	0	0	0	0	0	0	0
goa-1	0.01	**0.95**	0	0.01	0.02	0.01	0	0
nic-1	0	0	**0.99**	0	0	0	0	0.01
unc-36	0	0	0	**0.87**	0.03	0	0	0.09
unc-38	0	0	0.02	0	**0.78**	0.20	0	0
unc-29	0	0	0.01	0	0.36	**0.62**	0	0.01
egl-19	0	0	0	0.03	0.01	0	**0.95**	0
unc-2	0	0	0	0.09	0.04	0	0.01	**0.86**

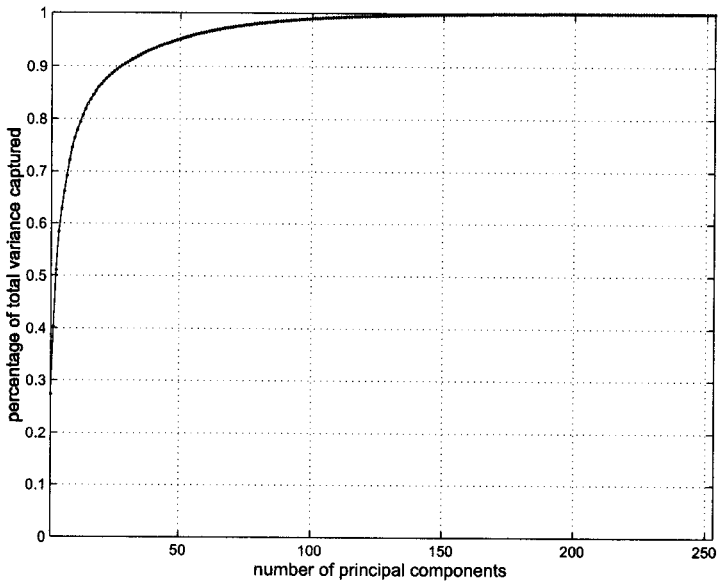


Figure 4.3: Percentage of the total variance captured by the first few principal components (PCs) shows the evidence that feature data may be represented in lower-dimensional space. The top 43 PCs capture over 94% of the total variance.

separated from the other types and from each other. Moreover, the closest two clusters are *unc-29* and *unc-38* (distance=3.5); these encode nicotinic receptor subunits with overlapping functional expression. *unc-2* and *unc-36* (distance=3.6), the next closest clusters, respectively encode a-1 and a-2 voltage-gated calcium channel subunits with nearly coincident expression patterns. This indicates that a simple Euclidean distance in feature space can be used to quantify the relative similarity between different mutant types.

4.5 Feature Selection and Classification of Phenotypes

Since one of our main objectives is to identify parameters that define particular mutant types, we wish to identify a small number of features that provide discriminative information. A variance plot (Figure 4.3) shows that the top 43 principal components (17% of total PCs) capture over 94% of the total variance. This gives a strong indication that a few carefully selected features would represent the data well.

To identify best features for distinguishing any two worm types, we screen the entire feature set using a backward elimination process based on the linear Lagrangian Support Vector Machine classifier [58][64]. The support vector machine classifier is used because it generalizes well. The process starts from the full feature set. In each iteration, one feature is eliminated from the remaining feature set by evaluating all the possible subsets (n subsets, each containing $n-1$ features) and selecting the subset that achieves the smallest training error as our next feature set. We use a low training error as an approximation of the importance of that feature. All the features can thus be ranked according to when they are eliminated from the backward elimination process. We repeat this process for all 8 mutant types in a pairwise fashion and generate 28 sequences of ranked features.

Feature subsets that are effective to distinguish all worm types are then selected progressively by choosing the most frequently features that appear on the top of all 28 sequences. For example, the first feature is selected as the feature that appeared most frequent as the No. 1 feature in all 28 sequences. The second feature is selected as the feature that appears most frequently as the No. 1 or No. 2 feature in all 28 sequences besides the feature that is already in the subset. A simple 1-nearest neighbor (1-NN)

classifier with 10-fold cross-validation [20] is used to evaluate subset performance. To avoid over-fitting, a 10-fold cross validation technique is used. For each feature subset in each trial, we divide data from each worm type randomly into 10 sections. One section (80 worms) is held out for testing and the other 9 sections (720 worms) are used as training data. In subsequent steps in the trial, different testing and training sections are chosen. The classification error is calculated as the average of the 10 iterations for each of the 28 class pairs. For each subset, 50 trials are performed to give an aggregated classification error rate for that subset. We also compare the classification error of the first few principal components using the three scaling methods (Figure 4.2).

A small set of features can be readily identified to approximate the dataset by following the cross-validation error curve. Table 4.2 shows the classification results by using all 253 and a subset of 39 features. This subset is chosen by the backward elimination process when the error rate first drops below 0.02. The data are well represented using a subset of 39 features for discriminating phenotypes. These features include several measurements of speed and reversals averaged over different time periods, and worm head and tail width and brightness information (Table 4.3).

Table 4.3: Features used in mutant characterization.

Feature	Description
CNTMVAVG	avg centroid movement
CNTMVMAX	max centroid movement
LNECRAVG	avg length/eccentricity
LNECRMIN	min length/eccentricity
LNMFRMAX	max length/MER
ANCHRMAX	max angle change
ANCHSMAX	max angle change std
RV20MAX	max reversal rate in 20s
RV20AVG	avg reversal rate in 20s
Continued on next page	

Table 4.3 – continued from previous page

Feature	Description
RV40MAX	max reversal rate in 40s
RV60MAX	max reversal rate in 60s
RV80MAX	max reversal rate in 80s
RV100MAX	max reversal rate in 100s
RV120MAX	max reversal rate in 120s
TOTRV	total reversal
TOTMOVE	dist moved in 5 min
PRP50MAX	max displacement in 25 sec
PRP40MAX	max displacement in 20 sec
PRP30MAX	max displacement in 15 sec
PRP20MAX	max displacement in 10 sec
PRP10MAX	max displacement in 5 sec
MVHLFAVG	avg speed in 0.5 sec
MVHLFMAX	max speed in 0.5 sec
LNGTHAVG	avg length
LNGTHMAX	max length
LNGTHMIN	min length
CNLNRAVG	avg center width/length
CNLNRMAX	max center width/length
CNLNRMIN	min center width/length
HCTHRMAX	max head to center thickness ratio
HEADBRAVG	avg head brightness
TAILBRMIN	min tail brightness
TAILBRMAX	max tail brightness
HTBRRMAX	max head/tail brightness
HANGCRMAX	max head angle change
HDMVHFAVG	avg head movement in 0.5s
HTMVRAVG	avg head/tail movement ratio
HDHFTOTMV	head movement in 5min
TLHFTOTMV	tail movement in 5min

4.6 Natural Clustering of Phenotypic Data

To further investigate the clustering of the data points, we apply the k-means clustering algorithm to find the natural clusters in the behavioral data. For this analysis, each data point is treated individually without regard to mutant type. The k-means algorithm is an elementary but very popular clustering method. It enjoys the benefits of making no assumptions about the underlying data probability distributions, and is thus applicable to many problems. Suppose there are to be k clusters with respective centers $C = c_1, ..., c_k$ and their corresponding non-overlapping divisions of feature space are defined as $D = D_1, ..., D_k$. Let $\|.\|^2$ denote squared Euclidean distance. Our data are $x_i : i = 1, 2, ..., 797$. We would like to choose $C = c_1, ..., c_k$ so that $C = argmin \sum_{j=1}^{k} \sum_{x_i \in D_k} \|x_i - c_j\|^2$.

While there is no closed form solution to the minimization, Lloyd [57] demonstrated that an alternating descent algorithm will always converge. The Lloyd algorithm for k-means clustering is an iterative descent algorithm. Starting with an initial set of k representative points, all the points in the data set are assigned to whichever of the k points is closest according to some distance measure, usually Euclidean distance. Next, each of the k representative points is relocated to be the centroid of the data points which just got assigned to it. At this point, we have a new set of k representative points, and can go back to the assignment step. The algorithm iterates between these steps of data point assignment and cluster centroid calculation, until convergence is reached. The final convergence, in general, depends on the initial choice of k representative points. The algorithm does not necessarily find the global optimum, and so often many random initialization seeds are used. We generate sufficiently many (10,000) random initializations for each k and track the error at the convergence to be reasonably confident that

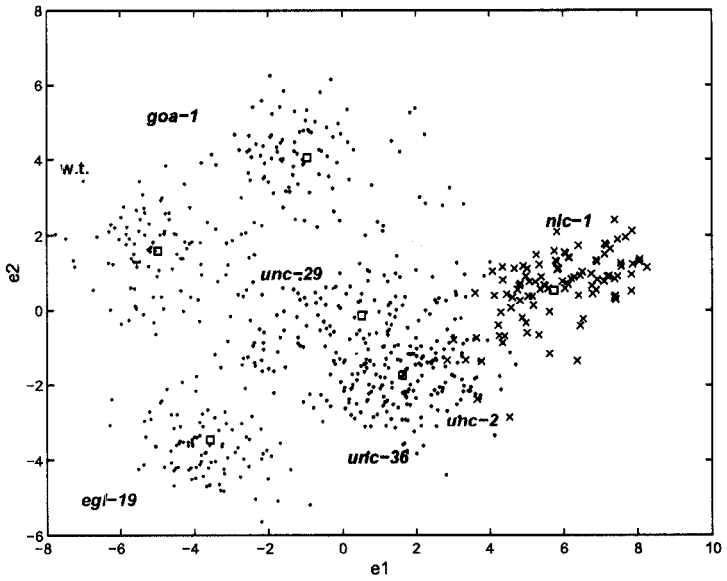


Figure 4.4: Cluster centers found by the k-means algorithm, $k = 6$. The data are presented in the first 2 principal component directions of the 253 features.

the global minimum is found.

Figures 4.4 & 4.5 show the cluster centers identified by the k-means algorithm; for each case, the centers are marked by black squares. Although the actual k-means clustering is done using all 253 selected features, the data are visualized by showing the first two principal components.

A key issue in k-means clustering is to determine the optimal number of clusters for the data set. We use two algorithms to determine the optimal cluster number for our behavioral data: the gap statistic [84] and the information theoretic method [78][77].

The idea of the Gap Statistic is to standardize the graph of $log(W_k)$ by comparing it to its expectation under an appropriate null reference distribution of the data. W_k is the

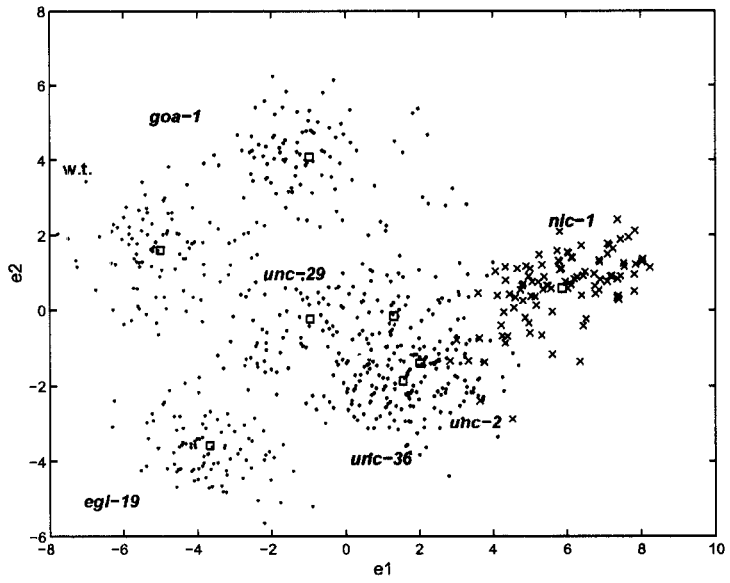


Figure 4.5: Cluster centers found by the k-means algorithm, $k = 8$. The data are presented in the first 2 principal component directions of the 253 features.

total within-cluster sum of squares around the cluster centers, when there are k clusters. Since we have 797 points in our data set, the null reference distribution is generated by drawing 797 samples from a distribution that is uniform along each feature data dimension. This is repeated B times. The expectation of the null reference $E[log(W_k b^*)]$ can be estimated as $\frac{\sum_{k=1}^{B} log(W_k b^*)}{B}$, where $W_k b^*$ is the within-cluster sum of squares of the reference dataset, and B is the number of reference datasets. The distance between these two curves is defined as the Gap (Equation 4.3),

$$Gap(k) = \sum_{b=1}^{B} log(W_k b^*) - log(W_k), k = 1, ..., K \tag{4.3}$$

where K is the maximum number of clusters defined by the user according to the expected range of clusters. We use a maximum of 10 centers ($K = 10$) and 5 reference datasets ($B = 5$). The sampling distribution can be measured by $s_k = sd_k\sqrt{1 \pm \frac{1}{B}}$,

where sd_k is the standard deviation of the reference null distribution. The formula to calculate the optimal number of clusters k_{opt} can be obtained as the first location where the gap curve starts to drop or level off. That is the first k that satisfies $gap(k) \geq gap(k+1) - \alpha s_{k+1}$, where α is a multiplier adjusted to reject null mode. Here is α set to 3.

The Information Theoretic approach tries to find the optimal number of clusters by fitting the within-cluster sum of squares curve (distortion curve) with two hyperbolic curves breaking at the location of the optimal k. When applying a negative power to the hyperbolic curves, they are transformed into two straight lines and the location of the break can be identified. The magnitude of the power is controlled by the dimensionality of the data . Here it is set to 7. The transformed distortion curve usually can be approximated reasonably well by a piecewise linear function consisting of two straight lines with a break, or elbow, at the location of the optimal k. The optimal number of clusters can be easily obtained by finding the biggest jump, which is the difference between the successive points on the transformed distortion curve. The paper [78] provides theoretic justification and points out that this method can also provide suboptimal solutions by finding smaller jumps in the curve. This is particularly appealing given our objective of exploring the substructure of the data.

As shown in Figures 4.6 & 4.7, both methods identify 6 clusters as the optimal number as shown in Table 4.4. In this optimal classification, the calcium channel mutants *unc-36* and *unc-2* are grouped into a single cluster and the nicotinic receptor mutants *unc-29* and *unc-38* into another cluster. In addition, the information theoretic approach identifies an additional suboptimal solution of 8 clusters with each cluster composed primarily of a single mutant type (Figure 4.4 and Table 4.5). Together, these results demonstrate that

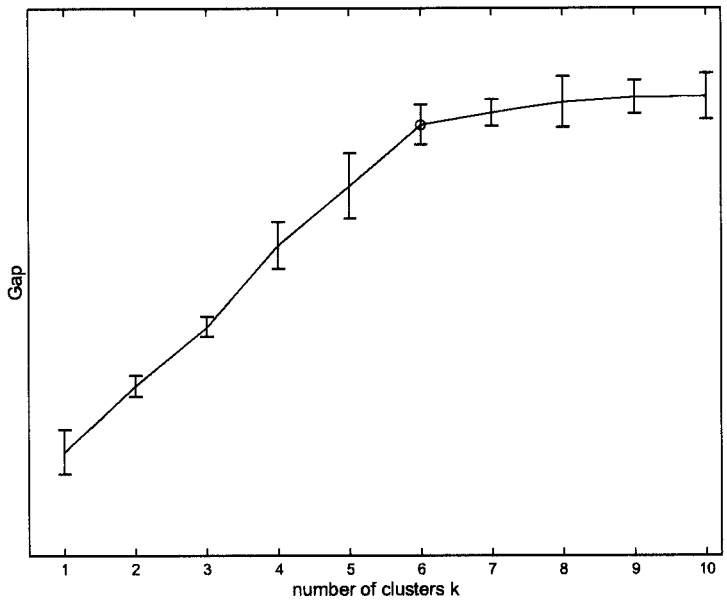


Figure 4.6: Gap plot by the gap statistic method. The optimal number of clusters, marked by a red circle, is identified as the gap curve first started to level off.

worms of the same mutant type tend to exhibit similar behavioral patterns and further show that cluster analysis can be used to assess phenotypic similarities between different mutant classes.

4.7 Summary

In this chapter, we showed that quantitative morphological and locomotion features obtained from digital video recordings can be used to distinguish the behavioral phenotypes of *C. elegans* mutants. As shown in Table 4.2, a reduced set of 39 features is sufficient to identify visibly dissimilar mutant types with very high reliability. Furthermore, these features can often be used to distinguish between types with highly similar phenotypes (e.g. *unc-2* and *unc-36*) that can not be reliably identified even by an experienced human observer. Thus, the parameters in the reduced feature set are likely

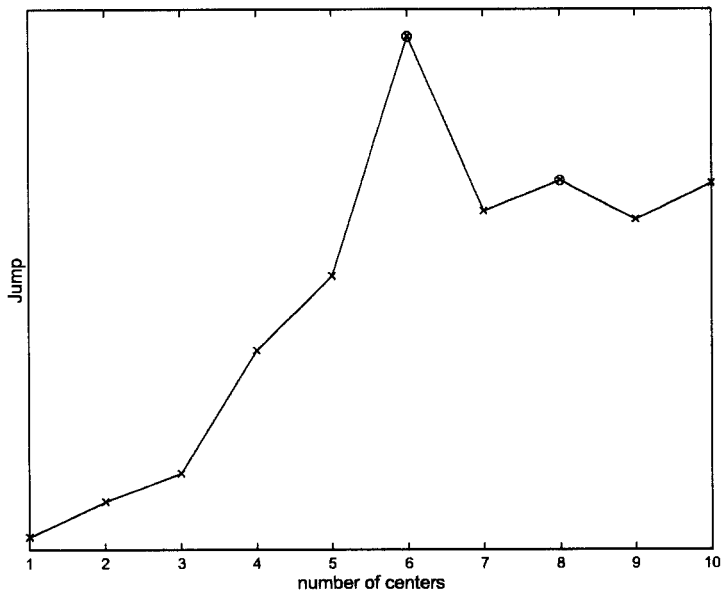


Figure 4.7: Jump plot by the information theoretic method. The optimal and suboptimal number of clusters, marked by red circles, are identified as the most and second most significant peaks.

Table 4.4: Data points are classified into 6 clusters (optimal number of clusters) based on their shortest distance to the cluster centers identified by the k-means algorithm. Note *unc-38* and *unc-29* are clustered together, as are *unc-2* and *unc-36*.

Center	1	2	3	4	5	6
Wild type	**97**	2	0	0	1	0
goa-1	2	**94**	0	3	1	0
nic-1	0	0	**100**	0	0	0
unc-36	0	0	0	**90**	10	0
unc-38	0	0	2	7	**91**	0
unc-29	1	0	1	9	**82**	5
egl-19	0	0	0	1	0	**99**
unc-2	0	0	2	**74**	22	1

Table 4.5: Data points are classified into 8 clusters (suboptimal number of clusters) based on their shortest distance to the cluster centers identified by the k-means algorithm. For the 8-cluster result, the majority of the samples belong to the correct clusters.

Center	1	2	3	4	5	6	7	8
Wild type	**97**	2	0	0	1	0	0	0
goa-1	2	**93**	0	4	1	0	0	0
nic-1	0	0	**97**	1	2	0	0	0
unc-36	0	0	0	**70**	5	2	0	23
unc-38	0	0	1	4	**69**	24	0	2
unc-29	0	0	0	5	26	**64**	1	2
egl-19	0	0	0	2	0	1	**97**	0
unc-2	0	0	1	15	15	1	1	**66**

to have great utility in assessing subtle or modest abnormalities in behavior caused by hypomorphic mutant alleles or by incompletely penetrant dsRNA inhibition.

These studies also provide insight into the nature of specific mutant phenotypes. For example, *unc-36*, *unc-29*, *unc-38* and *unc-2* have all been categorized as "weak kinkers", a term that has been difficult to define precisely. From Tables 4.3 & A.1, it is apparent that these mutants share many common effects on the variables used in our classification; in particular, all have a substantially higher angle change rate and substantially lower centroid movement and global speed parameters than wild-type. This combination of characters (increased body bending and a decreased rate of movement) thus provides an operational definition of the "kinker" phenotype. Likewise, the combination of increased centroid movement and increased angle change rate provides a functional definition of *goa-1*'s "hyperactive loopy" phenotype, while increased length and length/eccentricity and decreased angle change rate and speed define the "long, slow and floppy" phenotype of *egl-19*. In some cases, significant phenotypic differences are identified that are unno-

ticed (or unreported) in previous observer-based studies. For example, both *goa-1* and *unc-36* mutants show particularly large reductions in the ratio of head-to-tail movement, an abnormality whose neural basis could be investigated in future studies. Thus, it has been possible not only to obtain precise quantitative descriptions of phenotypic classes whose definitions have previously been subjective and qualitative, but also to resolve subtle differences within broad classes such as kinker Uncs.

The application of machine-based pattern recognition methods also allows us to probe the similarities between different behavioral patterns based on their clustering in multi-dimensional feature space. In general, the pattern of phenotypic clustering mirrors the known similarities in molecular function and cellular site of action of the mutant gene products. For example, *unc-29* and *unc-38*, which respectively encode α and β nicotinic receptor subunits with overlapping expression patterns, form a single cluster in the optimal clustering and have centers that are the closest together by Euclidean distance (Figure 4.4). Likewise, *unc-2* and *unc-36* mutants, which are defective in the a-1 and a-2 subunits respectively of the neuronal N-type calcium channel, form a single cluster in the optimal k-means clustering, and the centers of these two types' data clouds are relatively close in feature space. In fact, the centers for all four of these types (which have all been designated as kinker Uncs and all encode excitatory ion channels whose focus of action is primarily at body muscle neuromuscular junctions) are closer to one another than to the other Unc mutants or to wild-type. Thus, the quantitative phenotypic signature obtained through behavioral tracking appears to correspond well to the underlying functional defects of the mutants we analyze.

We anticipate that this type of comprehensive quantification of mutant behavioral phenotypes will have powerful applications in functional genomic studies. Clustering and

pattern recognition analysis of microarray-derived gene expression profiles has provided important information about the likely functions of novel gene products in *C. elegans* and other organisms [47]. In principle, a behavioral phenotype represents a similarly complex quantitative signature whose direct linkage to nervous system activity makes it particularly useful for classifying genes that function in excitable cells. In several genome-wide deletion and RNAi-based knockout surveys undertaken in *C. elegans*, the identification and classification of behavioral and other non-lethal phenotypes has been a crucial limiting factor [26][96]. Using the machine-based phenotyping approaches described here, it should be possible to record the behavior of an uncharacterized knockout strain, compare its phenotypic pattern to a database of known mutants, and make an informed initial hypothesis about the molecular pathways in which the mutant gene product participates.

Part of this chapter has appeared in the following publications.

- W. Geng, P. Cosman, J-H Baek, C. Berry, and W.R. Schafer, "Quantitative Classification and Natural Clustering of *C. elegans* Behavioral Phenotypes", *Genetics*, vol. 165, pp. 1117–1136, 2003.
- W. Geng, P. Cosman, J.-H. Baek, C. Berry and W.R. Schafer, "Feature Extraction and Natural Clustering of Worm Body Shapes and Motion Characteristics", *IASTED International Conference on Signal and Image Processing (SIP 2003)*, August 13-15, 2003, Honolulu, Hawaii.

I was the primary researcher and the co-author. Dr. Pamela C. Cosman and Dr. William R. Schafer directed and supervised the research which forms the basis for this chapter.

Chapter 5

Classification of Large Numbers of *C. elegans* Phenotypes

Hundreds of genes have been identified in *C. elegans* that affect behavior and morphology in specific ways. Our long-term aim is to collect data on large numbers of mutant types and effectively classify them according to their phenotypic similarity. With an increasing data set, it becomes progressively more challenging to identify features that effectively classify and distinguish the large variety of worm types.

To identify a proper identification and classification system, we present the results from different classifiers in this chapter. We start with an overview of the common classification procedures. Then we describe results from CART and Random Forests classifiers. The comparisons to human observers are discussed in Section 5.4, and the classifier comparison is discussed in Section 5.5, and we conclude with a summary section.

5.1 Classification Overview

A classification task is a supervised learning process. From a statistical learning point of view, a classification is a process that can be formulized as follows: Given a set of n training observations and the class label pairs $(x_1, c_1), ..., (x_n, c_k), x_i \in x_1, ..., x_n, c_j \in c_1, ..., c_k$ from k different classes. The goal of learning is to find a function $g_{opt} = minE[l(g(x), y)]$ that minimizes the expected losses caused by using the classification function $g(x)$, and l is a loss function. When all classification errors are assumed equally costly, a "0-1 loss function" such as:

$$l(g(x), y) = \begin{cases} 1 & \text{if } g(x) \neq y \\ 0 & \text{otherwise} \end{cases}.$$

is used. In this case, Bayesian Decision Theory states the minimum probability of error is achieved when $g(x) = \arg\min_j P_{c|x}(j|x)$ is chosen, where $P_{c|x}(j|x)$ is the conditional probability of class label c is true given sample x is present. The optimal (minimum probability of error) solution is to choose the class that the observation x is most likely from given the observations. In practice, however, it is difficult to achieve the minimum probability of error because the underlying conditional probability density function $P(x|c)$ is unknown. By making certain assumptions about $P(x|c)$ such as Gaussianity and independence, a series of linear classifiers can achieve good results [72].

Instead of estimating probability density function, non-parametric methods such as k-nearest neighbor(KNN) [20] and Classification and Regression Trees (CART) [9], try to build models from the data directly. Neural networks approaches(NN) [4] try to extract linear combinations of the inputs as derived features, and then derive the outputs as a

nonlinear function of these features. Support Vector Machines (SVM) [86][13] produce nonlinear boundaries by constructing a linear boundary in a large, transformed version of the feature space.

Other methods take advantage of ensemble learning by combining different classifiers or using the same classifier with different input data to procedure an optimal result. Such methods include Boosting, Bagging, and Stacking [93].

As the training (model building) process is established using only the training data, an important issue in the learning process is overfitting (bias and variance tradeoff). That is, the training error alone is not a good estimate of the test error because the training error consistently decreases with increasing model complexity. However, a model with zero training error is overfit to the training data and will typically generalize poorly. Therefore, it is important to understand how to estimate the test error when designing the experiments. Bootstrapping or cross-validation methods [63] are commonly used to estimate the test error.

5.2 Classification and Regression Trees (CART)

The CART algorithm for designing classification and regression trees has its origins in a 1984 monograph by Breiman, Friedman, Olshen, and Stone ([9]). Briefly, the CART approach involves recording a set of examples of each worm type (i.e., wild-type or a specific mutant), and measuring features that might in principle be used to distinguish different types. From these measurements, a training vector is generated for each recording consisting of an identifier of worm type along with the values for each feature

measurement. Using this learning sample (which in our case consisted of 1,596 data points - approximately 100 for each of the 16 strains), CART produces a binary classification tree in which each binary split of the data involves a splitting question of the form "Is $x_m \leq c$?" where x_m is one of the measurements, and c is a threshold. The root node of the tree contains all the training cases; the worm types are equally mixed together in this node. The goal of CART is to successively subdivide the training set using binary splits in such a way that the data associated with the terminal nodes of the tree do not have a mix of worm types; rather each node should be as pure as possible. We use the Gini index of diversity to measure the impurity of a set of data. A class assignment rule assigns a class to every terminal node. A simple rule is to assign the most popular class for each terminal node; this is called the plurality rule, and is what we use. When two different classes are tied for the most popular class in the node, we arbitrarily choose the lower numbered class as the class for that node.

A CART tree applied on a small set of six mutants is shown in Figure 5.1 to illustrate a typical tree process. CART has some distinct benefits over other classifiers. These benefits include that there is no need to scale the inputs; it can deal with a mixture of ordinal and numerical variables; and the ease of interpretability. For a large and complex dataset such as ours, however, CART performance is less satisfactory as shown in Table 5.2. There are more than 40 nodes in the 16-class CART tree, it is difficult to explain the complex structure.

5.3 Random Forests

Almost 20 years after being a co-inventor of CART, Breiman [5] [6][7][8] recently proposed a much improved classification and learning method called Random Forests.

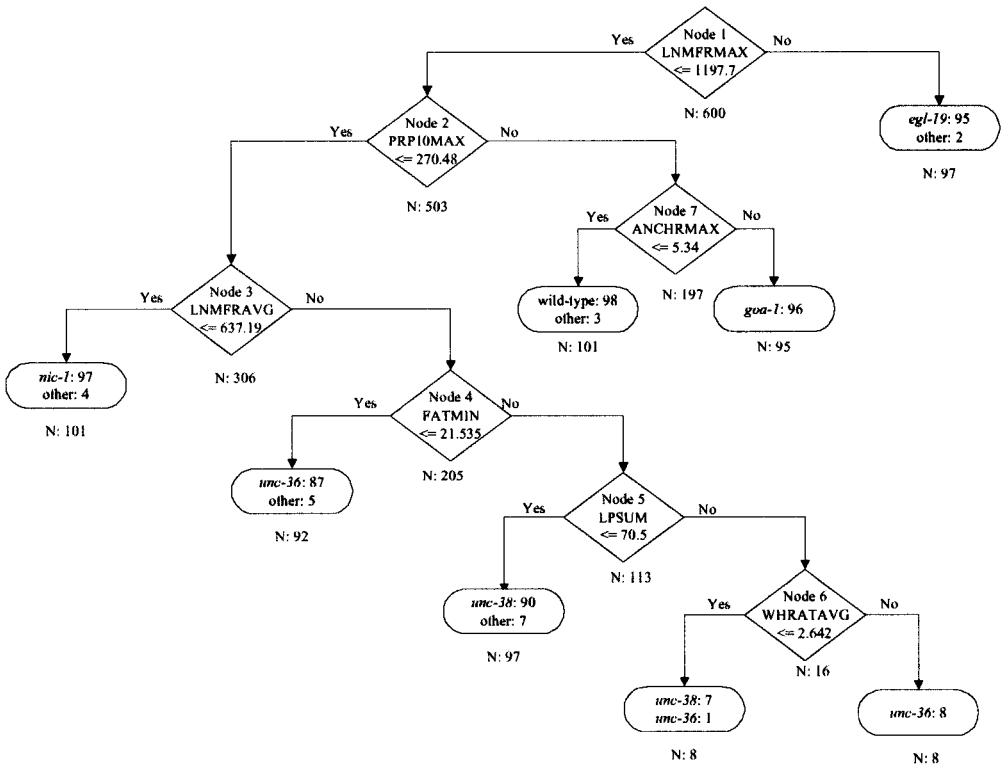


Figure 5.1: Optimal classification tree for 6 mutant types. The tree is constructed using the CART algorithm as described. The number of total animals in each node (N:) is indicated below the respective node; the number of animals of a particular type in that node is indicated within the node. For each type, the predominant terminal node is indicated in yellow.

Table 5.1: Classification result using CART with 253 features. The optimal tree is determined using 10-fold cross validation. The average classification success percentage is around 79%. Some of the mutant names have been abbreviated to fit into the table.

worm type	w.t.	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*	*tph1*	*uc63*	*dgk1*	*uc43*	*dop1*	*flp1*	*eat4*	*cat2*
w.t.	**85**				1				3			1	2	4	1	3
goa1		**86**			1			1	3	2	4			1	2	
nic1			**90**		2			3		4				1		
uc36			1	**90**		1		6	1	1						
uc38	1		2	1	**66**	21		4		2	1	1		1		
uc29		3	1	1	15	**65**	2	1		3		3	1	1	1	
eg19		1		1		1	**89**			1		2		2	1	2
uc2			2	7		1		**84**	1	1		1		2	1	
tph1	3	1	1	1	2			2	**77**	3		1	4	1	5	1
uc63		2	5	1	4	2				**76**	3	1	1		2	
dgk1		2			1				3	2	**82**		3	1	6	
uc43	1				2	2	1		1	1	2	**91**				
dop1	2	1			2	2			1	3	1	3	**73**	3	10	
flp1	4	1	1		2		1	2	1	1	7		2	**69**	3	4
eat4	1	6	2	1	1	2		2		6	2	8	8	2	**59**	
cat2	3								2				4	4	2	**87**

Random Forests utilizes an ensemble learning scheme. Instead of generating a single classification tree, many trees (to make up the forests) are generated independently by bootstrapping from the original data. A simple majority vote is taken for prediction. In addition to constructing each tree with a different bootstrap sample of the data, Random Forests adds an additional layer of randomness by splitting at each node using a random subset of predictors instead of using the best split among all features as is done in CART [9].

By adding these two layers of randomness, Breiman [5] provided both empirical and theoretic evidence that the two levels of randomness minimize the correlation (dependence) among trees while maintaining strength (accuracy of each individual tree classifier). Thus the Random Forests method performs very well compared to CART and many other classifiers including discriminant analysis, support vector machines and neural networks. The method is also robust against overfitting [5]. An estimate of the error

rate can be obtained by predicting using "Out-of-bag" (OOB[8]) data, which are the data (around 36% of the data) that are not used in each bootstrap sample. The classification error rate is thus defined as the aggregated OOB prediction error rate. Given enough trees being grown, the OOB error rate is quite accurate [14]. There are only two free parameters (the number of trees in the forest and the number of random features considered at each split).

Random Forests also provides four measures of feature importance that can be used for model reduction. One of these measures of feature importance, defined as the average lowering of the margin across all samples when this feature is randomly permuted, is used because it is more robust against noise (Personal Communications from Liaw and Breiman & [56]). For each sample, the margin is defined as the proportion of votes for its true class minus the maximum of the proportion of votes for each of the other classes.

For the classification of 16 mutant types, the forest is made up by $5,000$ trees ($n_{trees} = 5,000$). At each split, 15 features are randomly selected to be considered for splitting ($m_{try} = 15$), which is approximately the square root of the total 253 features used. The confusion matrix, represented by OOB errors, is shown in Table 5.3. The classification success rates are listed along the shaded main diagonal while the off-diagonal entries represent the misclassification error rates. The average success percentage is 90.9%, showing a high degree of success at identifying the correct mutant type even if presented with a single example recording.

The important features identified by Random Forests are shown in Table 5.3. If we run the classification procedure using only the top 25 features (10% of the total features) identified by Random Forests, we achieved 85.9% classification accuracy.

Table 5.2: Classification result using Random Forests with 253 features, 5,000 trees and 15 random features to split on at each node. The OOB estimate of success percentage is 90.09%. Some of the mutant names have been abbreviated to fit into the table.

worm type	w.t.	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*	*tph1*	*uc63*	*dgk1*	*uc43*	*dop1*	*flp1*	*eat4*	*cat2*
w.t.	**93**								1		2		1			3
goa1	1	**93**								2	1	1			2	
nic1			**98**	1						1						
uc36				**97**		1		2								
uc38			3	2	**83**	11				1						
uc29		1			21	**74**		1		1						
eg19				2			**97**									
uc2			1	1				**97**								
tph1			1					3	**89**	1			2	1	3	
uc63			6	1	1	3				**87**			1		1	
dgk1	1	1								3	**91**				3	
uc43							1					**98**		1		
dop1					1		1		2				**88**	1	7	
flp1							3		1	1	5	1	1	**85**	2	1
eat4			4	1					4	1					**90**	
cat2	4									1	1		1	2		**91**

Figure 5.2A shows the effect of the number of features selected at each split on the error rate with 5000 trees constructed. The errors are stable between 10 and 100 features ($error = [0.088, 0.095]$) and trend upward slightly afterwards. Figure 5.2B shows the effect of the number of trees used when 15 features are selected at each split. The error converges quickly after 800 trees ($error = [0.090, 0.096]$) are constructed. Both plots indicate that the results are not sensitive to the selection of these two parameters.

5.4 Comparison to Human Observers

It is interesting to compare our system with human experts. A preliminary comparison with one human expert reveals that our automatic system outperforms the human dramatically. For example, in one experiment we conduct, an experienced observer is

Table 5.3: Important features identified by Random Forests. The OOB estimate of success percentage by using these 25 features is 85.9%.

Feature	**Description**
MVHLFMAX	Maximal centroid movement in 0.5 sec
HTMVRAVG	Local head and tail movement ratio
PRP10MAX	Maximal centroid movement in 5 seconds
TOTMOVE	Total centroid movement in 5 minutes
TAILBRMAX	Maximal tail brightness
TAILBRAVG	Average tail brightness
PRP20MAX	Maximal centroid movement in 10 seconds
HTBRRMIN	Head and tail brightness ratio
HDHFTOTMV	Sum of head movements in 0.5 sec
LNECRMIN	Length/eccentricity max
ANCHRMAX	Max angle change rate
TLHFTOTMV	Sum of tail movements in 0.5 sec
LNGTHAVG	Average Length
TLMVHFMAX	Maximal tail movements in 0.5 sec
MVHLFAVG	Average centroid movement in 0.5 sec
TAILBRMIN	Minimal tail brightness
HTBRRAVG	Average head to tail brightness ratio
LNGTHMIN	Minimal length
HDMVHFAVG	Average head movements in 0.5 sec
HDMVHFMAX	Maximal head movements in 0.5 sec
HTBRRMAX	Maximal head to tail brightness ratio
LNGTHMAX	Maximal length
AVGBRMAX	Maximal body brightness
ANCHSMAX	Maximal standard deviation of angle change rate
TLTHKMIN	Minimal tail thickness

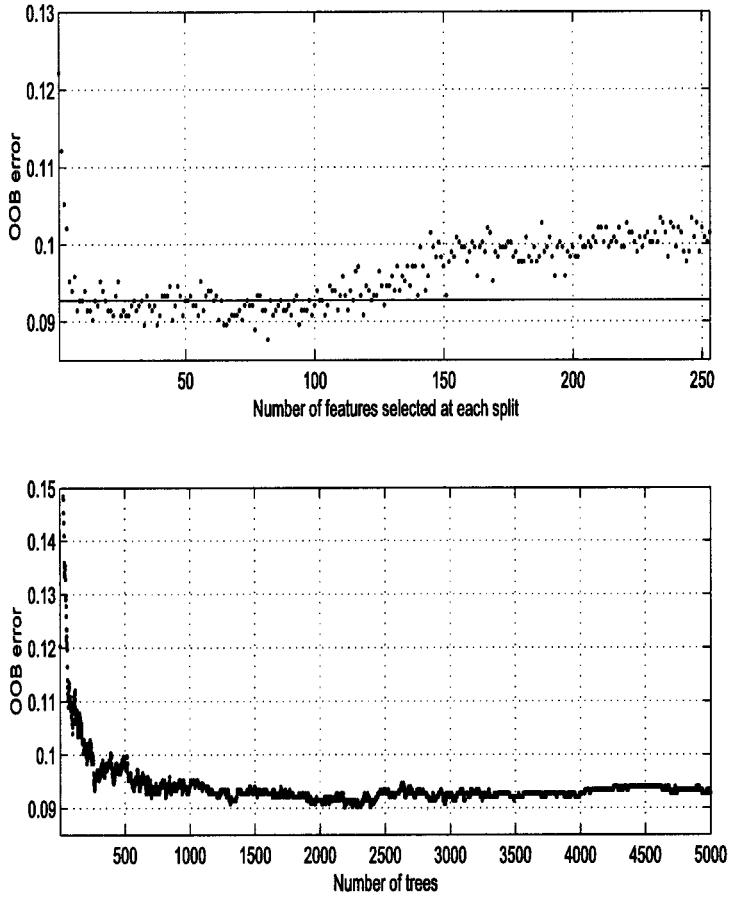


Figure 5.2: (A) Effect of number of features selected at each split on OOB error rate. 5,000 trees are used. The horizontal line represents the error rate using 15 features. (B) Effect of number of trees used to construct Random Forests on OOB error rate. 15 features are used at each split.

presented with 100 1-minute videos of *unc-36* and *unc-2* (50 of each), and identified only 50% of *unc-36* and 90% of *unc-2* videos correctly. Running the same experiment on *dgk-1* and *goa-1*, the observer identified 84% of *goa-1* and 52% of *dgk-1* correctly. Using 1-minute videos for the human observer reduces the experiment time by 80% compared to 5-minute videos, and in any case, the human observer typically makes his decision within the first 70 frames (35 seconds). The human observer in our experiment is a *C. elegans* expert with more than twenty years of experience working in this field. The system produces over 93% correctness for each of these types against 15 other types combined. Furthermore, from the features extracted by the system, the head and tail brightness difference and total movement and reversals are the top features distinguishing *unc-2*/*unc-36* and *goa-1*/*dgk-1* pairs respectively. These features are hard to quantify by eye.

5.5 Comparison to Other Classifiers

To investigate other popular classifiers and also compare the cross-validation and OOB methods, we compare the performance of several well-known classification methods for our *C. elegans* data. These methods include: k-nearest neighbor classifier (KNN), support vector machines (SVM), CART, Random Forests using bootstrap samples, and Random Forests using cross-validation method.

KNN classifiers are based on finding the k nearest examples in some reference set, and taking a majority vote among the classes of these k examples, or, equivalently, estimating the posterior probability $p(c|x)$ by the proportions of the classes among the k examples. The nearness is measured by Euclidean distance. Here we consider using $k = 1, 3$.

Support Vector Machines classifiers first map input feature vectors to a higher dimensional space by a function ϕ. Using a radial basis kernel function, $K(x_i, x_j) = exp(-\gamma||x_i - x_j||^2), \gamma > 0$, the mapping function can be expressed as $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$. Then SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. The maximal margin is desired to avoid overfitting. Vapnik [86][13] show that the mapping and maximal margin can be found by solving a dual optimization problem defined in the following Equations 5.1 and 5.2:

Define (x_i, y_i), $i = 1, \ldots, l$, where $x_i \in R^n$ and $y \in \{1, -1\}^l$,

$$\arg\min_{\mathrm{w,b},\xi} \; 1/2 w^T w + C \sum_{t=1}^{l} \xi_i \tag{5.1}$$

$$\text{subject to } y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \; \xi_i \geq 0. \tag{5.2}$$

where w, b defines the linear separating hyperplane in the transformed feature space. ξ_i is the distance between the point on the wrong side of the hyperplane and the hyperplane. C is the penalty parameter of the error term, which defines a band that are C units away from the hyperplane.

All the implementation is done using statistical computing software R [69]. In each cross-validation method, a 10-fold stratified cross-validation [63] is applied. For each classification method, 100 trials are constructed. For the Random Forests method, a bootstrap validation is also performed. The parameters are tuned for each method before the final experiments. The result is shown in Figure 5.3.

The Random Forests methods show the best results. Notice both bootstrap and cross-

validation experiments give consistent results. The variability of Random Forests experiments are among the smallest. SVM also shows good result. KNN ($k = 1, 3$) have higher classification error rates. CART has the worst performance.

Overall, we have found that the Random Forests approach both leads to an overall lower misclassification rate as well as to a more stable assessment of classification errors. Combined with the ability to identify the important features and no need to scale the input features, our preliminary analyses suggest that Random Forests method is ideal for *C. elegans* phenotype classification.

5.6 Summary

In this chapter, we evaluate some popular classification methods on a large set of *C. elegans* behavioral feature data extracted using the methods presented in Chapters 2 & 3. The results show that the features describe the phenotypes well. The machine vision based classification system outperforms a human expert. Among all the classifiers evaluated, Random Forests is best suited for *C. elegans* classification.

Part of the chapter has appeared in the following publications.

- W. Geng, P. Cosman, C. Huang, and W. R. Schafer. "Automated Worm Tracking and Classification." *Proc. of the 37th IEEE Asilomar Conference on Signals, Systems and Computers*, pp. 2063-2068, Pacific Grove, CA, November 2003.
- W. Geng, P. Cosman, C. Berry, Z. Feng and W.R. Schafer, "Automatic Tracking,

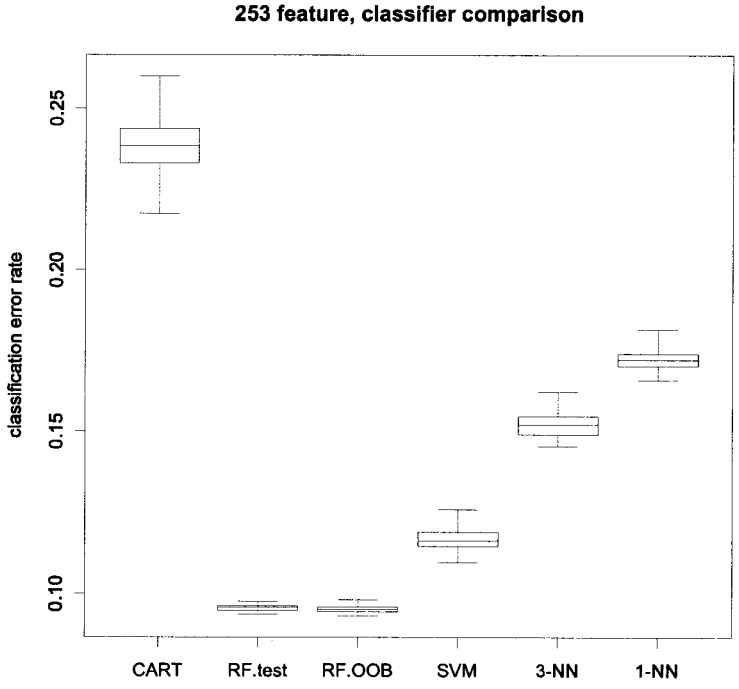


Figure 5.3: Summary of errors for classifications performed on the same set of data (1,596 worms from 16 mutant types with 253 features). In all cases, the box extends from the first quartile (25th percentiles) to the third quartile (75th percentiles), and the horizontal line within the box indicates the median. The lower and upper error bars indicate 10th and 90th percentiles respectively.

Feature Extraction and Classification of *C. elegans* Phenotypes", *IEEE Transactions on Biomedical Engineering*, in press, 2004.

I was the primary researcher and the co-author. Dr. Pamela C. Cosman and Dr. William R. Schafer directed and supervised the research which forms the basis for this chapter.

Chapter 6

Egg-laying Behavior Study

We have shown in this dissertation thus far that a machine vision system (including data acquisition, segmentation, tracking, and feature extraction) can be used to define phenotypes (classification and clustering). Another way of studying *C. elegans* behaviors using this system is to study specific behaviors. In this chapter, we describe a method of studying egg-laying using the machine vision system illustrated in Chapters 2 and 3. We begin with a brief overiew on egg-laying (Section 6.1). Then the algorithm for attached egg detection is introduced in Section 6.2-3. Egg onset detection and the behavioral changes surrounding egg-laying events are studied in Section 6.4-5. We conclude this chapter with a summary section.

6.1 Egg-laying Overview

One of the most important behaviors for the analysis of neuronal signal transduction mechanisms is egg-laying. Egg-laying in *C. elegans* occurs when embryos are expelled from the uterus through the contraction of 16 vulval and uterine muscles [92]. In the

presence of abundant food, wild-type animals lay eggs in a specific temporal pattern: egg-laying events tend to be clustered in short bursts, or active phases, which are separated by longer inactive phases during which eggs are retained. This egg-laying pattern can be accurately modeled as a three-parameter probabilistic process, in which animals fluctuate between discrete inactive, active, and egg-laying states [88]. Egg-laying has also been shown to be coordinated with locomotion: specifically, animals undergo a transient increase in global speed immediately before each egg-laying event [35]. Many neurotransmitters and neuronal signal transduction pathways have been shown to have specific effects on egg-laying behavior; thus it has become an important behavioral assay for the analysis of many neurobiological problems in *C. elegans*.

Because egg-laying is infrequent, it is well suited for analysis by automated imaging methods. In previous egg-laying studies [35][89] [95], individual worm movements were videotaped and the centroid location and time information were saved at 1s intervals during recording. The entire videos were later played back and each video frame was examined by expert observers to look for egg and egg onset frames (the frames in which the egg first appears).

Automated egg-laying detection is a difficult problem. There is a variety of scenarios due to a low occurence of egg-laying events and their bursty nature. For example, there could be more than one egg being laid at the same time. The eggs tend to stay attached to the worm body for a period of time (ranging from a few seconds to a few minutes) after egg-laying onsets. In the meanwhile, other subsequent eggs could be laid. Adding to the complexity of the problem, the worm sometimes could crawl back to the previous eggs. We break the onset detection problem into two problems. We will solve the above issues in the egg onset detection section. In following section, we will

only focus on deciding whether or not there is an egg or eggs attached to the worm body in a given frame.

6.2 Model-based Attached Egg Detection

6.2.1 Image Analysis

To find the possible egg locations and limit the search area for deformable template matching, we developed a series of morphological image analysis algorithms to limit our search area to around 2% of a typical region that a worm body covers. The search is greatly expedited and match accuracy is improved by effectively eliminating potential false alarms. The flowchart of attached egg detection is shown in Figure 6.1.

For each input video frame, the worm body is first segmented from the background and the skeleton (medial axis) is obtained by algorithms described in Chapter 2. The laying of an egg changes the shape of the binarized worm body (Figure 6.2.1), which can be captured by examining the width profile in the middle part of the worm body in the following way. For each pixel in the skeleton pixel list, a straight line traversing the worm body that passes through that skeleton pixel is calculated. 71 additional lines are also calculated at 5-degree intervals to cover a 360 degree radius. The worm body width at that skeleton pixel is the shortest of the 72 lines, which has the shortest distance traversing the binary image through the skeleton pixel. We next test for an abnormal width, by which we mean a difference greater than 7.5 pixels between median and peak width in the middle part of the body, indicating a potential egg event. The fixed width of 7.5 pixels is chosen over a proportional width (threshold proportional to the worm's body size) because most eggs have very similar size regardless of the worm's size. In

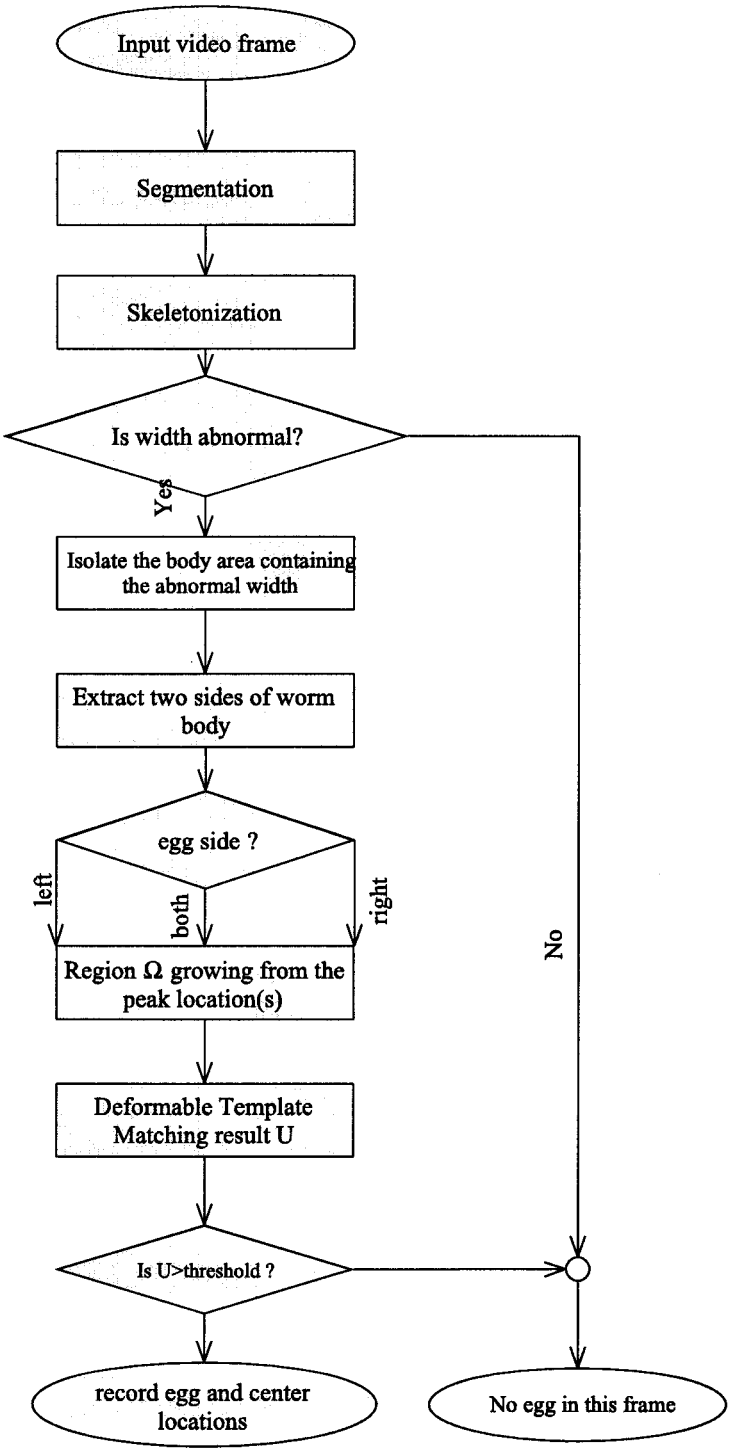


Figure 6.1: Egg detection process flow chart.

the case where the abnormal width is caused by an attached egg, one of the two end point locations on the shortest-distance line is enclosed by that egg.

Figure 6.2A shows the frame immediately prior to an egg-laying event. Figure 6.2B shows the egg-laying frame. The corresponding width profiles are shown in Figure 6.2C and Figure 6.2D respectively. The solid curves show the width measured along the worm skeletons. The horizontal dotted lines in Figure 6.2C and Figure 6.2D show the median width for the middle part of the worm body. A second horizontal line in Figure 6.2D shows the threshold (7.5 pixels above the median width value) that defines abnormal width. The width profile curves are normalized to 300 pixels for comparison. Since egg laying is a rare event, over 90% of the frames are quickly passed through and not subject to further analysis.

Since the abnormal width measure can not tell us which side the egg is on (which end point the egg encloses), we extract the boundaries from both sides of the worm body and consider the side that has higher k-curvature values to be the egg side. This way, the search area is constrained to only one side of the worm body and half of the search area is effectively eliminated. The process starts with isolating the body area containing the abnormal width by cutting off the worm body area that is 25 pixels before and after using the minimal-distance straight lines passing through the skeleton pixels. This cutoff area is 51-pixels in medial axis and has four boundaries. Two of the boundaries are the straight cutoff lines, and the other two are the two sides of the worm body (Figure 6.3B). A boundary following algorithm similar to [75] is then used to extract the two boundaries along the sides of the worm body (Figure 6.3C). The k-curvature [40] of these two boundaries is calculated, and the boundary that has higher (for all 5 k-curvature measurements) values is designated as the egg side. If neither boundary has

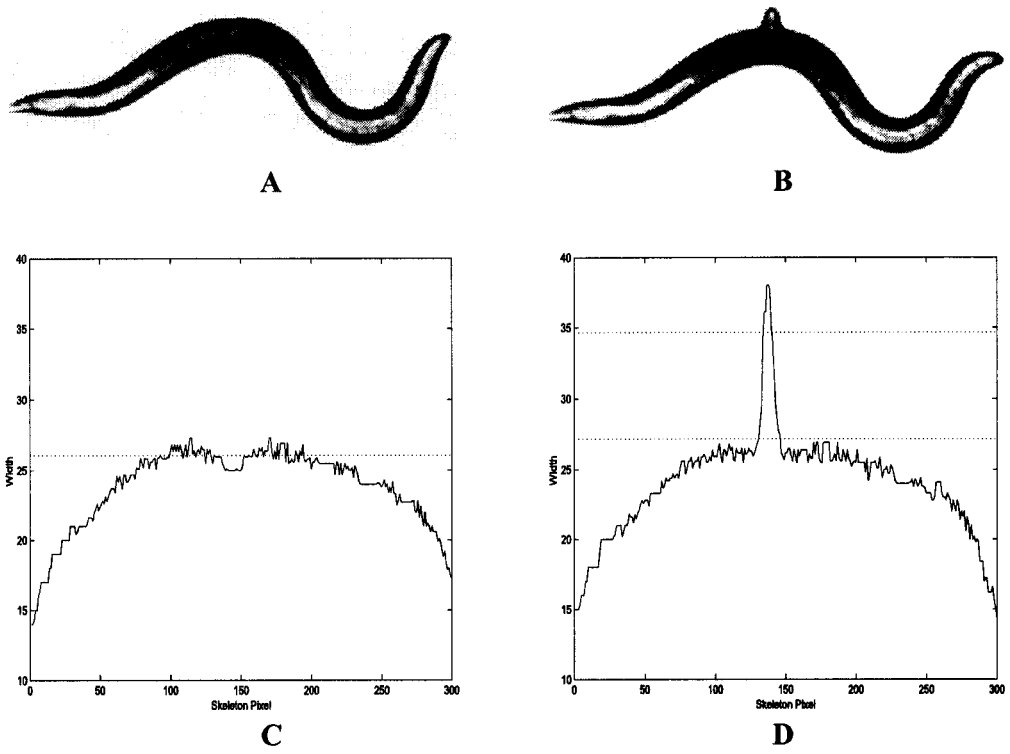


Figure 6.2: Width profile change on egg onsets. (A) Gray level image right before an egg onset. (B) Gray level image right after an egg onset. (C) Width profile of (A). The dotted line is the median value of the middle part of the width profile. (D) Width profile of (B). The lower dotted line is the median value of the middle part of the width profile. The upper dotted line is 7.5 pixels above the lower dotted line.

all 5 measurements higher, both sides are checked for eggs. The k-curvature is defined in Equation 3.1, where $(x_i, y_i), (x_{i+1}, y_{i+1}), ...$ are the locations of consecutive points that are k pixels apart along the worm side boundaries.

Once the location of the maximal peak is decided, the search region W can be obtained by region growing out of the egg side end point to enclose the egg center. A directional dilation algorithm such as [17] can be used for this purpose. Here we once again take advantage of the worm skeleton. The directional dilation is achieved by ap-

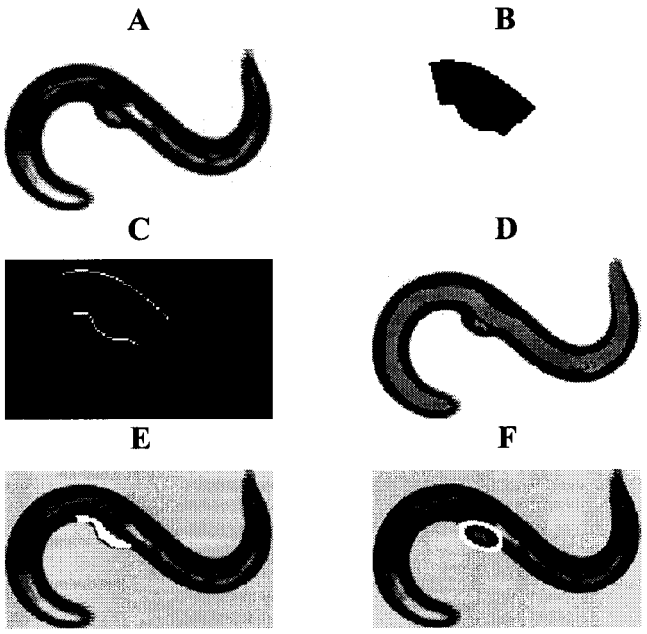


Figure 6.3: Illustration of egg detection image analysis. (A) Gray level image. (B) The cutoff portion containing egg. (C) Two boundaries. (D) The highlighted area shows the skeleton dilation region that will not be searched. (E) The highlighted area shows the final search region. (F) best-fit ellipse.

plying two constraints in the dilation process: (1) dilation starts from the end point and should remain inside the binary worm body; (2) dilation remains outside skeleton area (dilated 4 times from skeleton) (Figure 6.3D). The dilation process stops when more than 200 pixels are inside the region. The directional dilation forces the search area to be inside the worm body close to the side boundaries rather than close to the skeleton. The final search region W (Figure 6.3E) typically contains between 200 and 250 pixels for each frame.

6.2.2 Deformable Template Matching

Deformable template matching models have been applied to a variety of image recognition and analysis applications with success [61][42][41] [22][23][24][43]. They enjoy not only the flexibility of a parameterized model, but also can be explained in a Bayesian framework. Even though the attached eggs could be partially obscured by shadows, by the worm body, and/or partially laid, they share many common characteristics. They tend to have oval shapes, and are generally brighter in the middle and darker around the boundary. The eggs are similar in size. These characteristics make them ideal for the elliptic deformable templates.

In an ideal case, the shape of the attached eggs can be modeled by an elliptic model such as the one shown in Figure 6.4 with 7 parameters $v = (x, y, a, b, \theta, \rho_1, \rho_2)$, where (x, y) are the coordinates of the center, a and b are the semi axes and θ is the rotation angle. Together, these 5 parameters control the geometric shape and location of the inner ellipse that captures the bright center part of the egg. ρ_1 equals the ratio between the area of the middle band and the inner ellipse, ρ_2 equals the ratio between the area of the outer band and the middle ellipse. The middle band encloses the dark

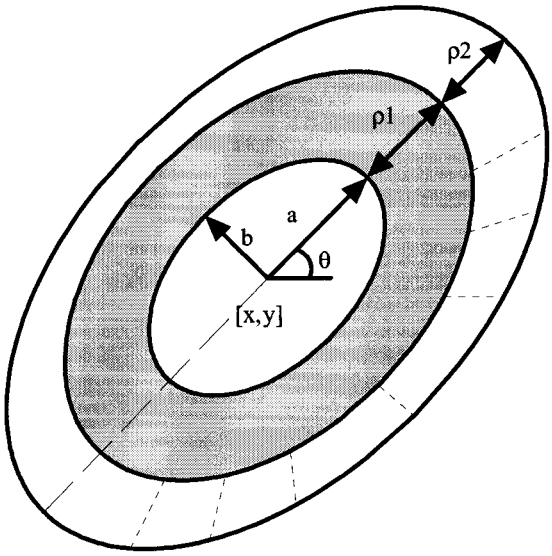


Figure 6.4: Ellipse egg model.

exterior part of the egg. The outer band covers part of the worm body and part of the background. By studying the homogeneity of the pixels enclosed, the outer band can be used to suppress noise and find the best location for the egg. For example, if (x, y) is mistakenly inside the worm body, then the outer band will have similar brightness to the worm body (dark). If (x, y) is in the background area, the outer band has similar brightness to the background (light). Half worm body and half background inside the outer band indicate a perfect attached egg location. To reduce model complexity, we opt to use a simplified model (Figure 6.5) that does not have the outer band, and use image analysis described in the previous subsection to restrain the search area. The outer band in Figure 6.4 is only used (in Figure 6.7) to mark the locations of the best-fit ellipses. There are 6 parameters characterizing the shape of the simplified elliptic model $v = (x, y, a, b, \theta, \rho)$.

From a Bayesian framework, we have $p(v|E) = \frac{p(v)p(E|v)}{p(E)}$, where E is the event

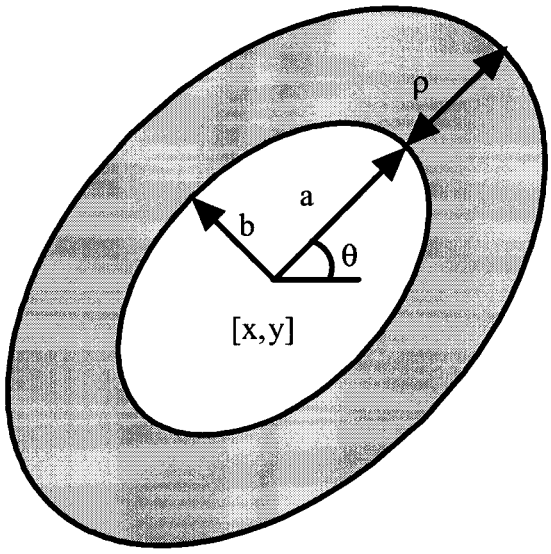


Figure 6.5: Simplified ellipse egg model.

that the image contains an egg, and $p(v|E)$ is the probability density function of parameter configuration given that an egg is present. There are many ways to define the likelihood function. We propose the following model:

$$p(E|v) = \frac{1}{z} exp-(\alpha u_{in}(v) + \beta u_{out}(v)) \tag{6.1}$$

where $u_{in}(v)$ is the mean pixel value inside the inner ellipse, $u_{out}(v)$ is the mean pixel value in the band around the inner ellipse (Figure 6.5), and α,β are the weights to be selected to give a proper weight for inside and outside areas. For calculating the mean values, the pixel intensities are linearly rescaled to go from -1 to $+1$. z is a normalization constant to ensure that $p(E|v)$ is a proper statistical distribution.

The egg finding problem can then be modeled as finding the most likely parameter configuration v_{opt} given that there is an egg in the image. Using a maximum a posteriori

(MAP) estimator, we have

$$v_{\text{opt}} = \arg\max_{v} p(v|E) = \arg\max_{v} \frac{p(v)p(E|v)}{p(E)} \tag{6.2}$$

Since the egg can occur in any orientation and location in the search space, it is reasonable to assume a uniform prior. For simplicity, we also assume a and b are uniformly distributed in a narrow range. So Equation 6.2 is identical to

$$v_{\text{opt}} = \arg\max_{v} p(E|v) = \arg\max_{v} \frac{1}{z} exp{-(\alpha u_{in}(v) + \beta u_{out}(v))} \tag{6.3}$$

Furthermore, because z is a constant, Equation 6.3 is identical to

$$v_{\text{opt}} = \arg\max_{v} (\alpha u_{in}(v) + \beta u_{out}(v)) \tag{6.4}$$

The optimal parameter configuration is the parameter v that maximizes the function

$$U(v) = \alpha u_{in}(v) + \beta u_{out}(v) \tag{6.5}$$

We choose $\alpha = 0.5$, $\beta = -1$, and $\rho = 8$ by feeding a small set of training samples of egg and non-egg values of u_{in},u_{out} into the Classification and Regression Trees (CART) algorithm [9]. The final model for locating eggs is as follows:

For a specific search space Ω in the image, find

$$v_{\text{opt}} = (x_{opt}, y_{opt}, a_{opt}, b_{opt}, \theta_{opt}) = \arg\max_{v} U(v) \tag{6.6}$$

where $U = 0.5u_{in}(v) - u_{out}(v)$. Notice $U \in [-1.5, 1.5]$.

For every pixel (x_c, y_c) inside the search region Ω, U is calculated for each configura-

tion with a range ($a = [3.4, 3.6], b = [1.9, 2.1], \theta = [0, 180]$). If U_{opt} is greater than a threshold value t, the location (x_{opt}, y_{opt}) is marked as the egg location and an egg is declared found.

6.3 Experimental Results

The egg detection algorithm was tested on 1,600 5-minute video sequences from 16 different mutant types (100 videos for each type) and on five 20-minute video sequences of wild type animals treated with serotonin, which causes an increase in egg laying. The data were collected over a 3-year period by different individuals. A laborious manual check found 9,000 frames containing 200 different eggs. These eggs cover a wide variety of recording conditions and mutant types. 100,000 non-egg frames (also verified manually) are randomly selected from the rest of the 800,000 frames as non-egg cases. By applying the above algorithm with the decision threshold t varying from -1.5 to 1.5, the performance result is shown as a ROC curve [62] in Figure 6.6 and Table 6.1. The True Positive fraction is over 98% when the False Positive fraction is 1% . Figure 6.7 shows some examples of the locations and best-fit ellipses identified by the algorithm.

6.4 Egg Onset Detection

Egg detection algorithms can be readily incorporated into a broader scheme for egg event onset detection. Figure 6.8 shows one algorithm to accomplish it. The main functions of the egg onset detection routine are to use the single frame egg detection result for a sequence. First, we decide whether the current egg is a newly laid or a previ-

Table 6.1: The true positive (Rate of egg frames detected as egg frames), true negative (Rate of non-egg frames detected as non-egg frames), false positive (Rate of non-egg frames detected as egg frames), and false negative (Rate of gg frames detected as non-egg frames) values for part of the ROC curve. The highlighted row is the final threshold used in the egg onset detection.

False positive	True Positive	False Negative	True Negative	Threshold t
0.0967	0.9985	0.0015	0.9033	0.35
0.0947	0.9983	0.0017	0.9053	0.36
0.0924	0.998	0.002	0.9076	0.37
0.0893	0.9977	0.0023	0.9106	0.38
0.0857	0.9972	0.0028	0.9143	0.39
0.0814	0.9964	0.0036	0.9186	0.4
0.0769	0.9961	0.0039	0.9231	0.41
0.072	0.9955	0.0045	0.928	0.42
0.0663	0.9946	0.0054	0.9337	0.43
0.0597	0.9927	0.0073	0.9403	0.44
0.0524	0.9915	0.0085	0.9476	0.45
0.044	0.9902	0.0098	0.956	0.46
0.0354	0.9893	0.0107	0.9646	0.47
0.027	0.9883	0.0117	0.973	0.48
0.0194	0.9865	0.0135	0.9806	0.49
0.0131	**0.9851**	**0.0149**	**0.9869**	**0.5**
0.0101	0.9826	0.0174	0.9899	0.51
0.0082	0.9785	0.0215	0.9918	0.52
0.0065	0.9729	0.0271	0.9935	0.53
0.0052	0.9658	0.0342	0.9948	0.54
0.0042	0.9531	0.0469	0.9959	0.55

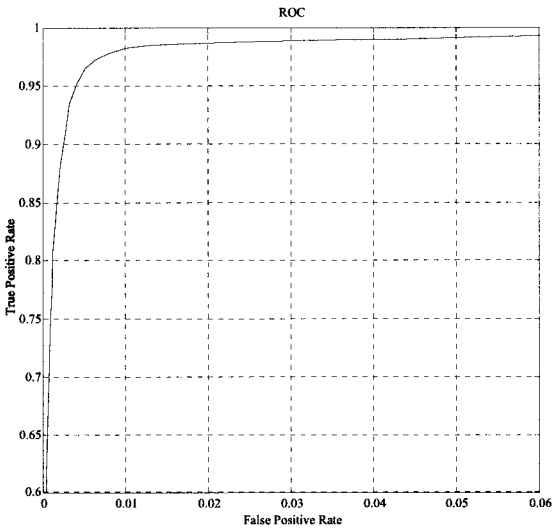


Figure 6.6: A plot of the receiver operating characteristic (ROC) curve with threshold t varying from -1.5 to 1.5.

ously laid egg (worms sometimes crawl back to previous eggs). This is accomplished by maintaining a list of all existing locations of eggs. When the new location is not on the list, an egg onset event is detected. Secondly, there are occasions when multiple eggs are laid at the same time. The egg onset detection routine runs the single frame egg detection routine repeatedly in the search regions after the detected egg area (outer ellipse in the template model) is removed from the image in each run. This way, clusters of eggs can be detected. The egg onset detection routine also runs the abnormal width detection routine repeatedly to find out new search regions to detect all the eggs attached to the worm body.

By setting the thresholds conservatively (t=0.5), our algorithm is able to identify all 88 egg onsets in the 135 hour videos. There are 131 false alarm onset frames. The false alarm onsets are easily eliminated by inspecting each onset frame visually. Among the 88 onsets detected, there are 6 onsets that are delayed from true onsets by 1, 2, 3, 4,

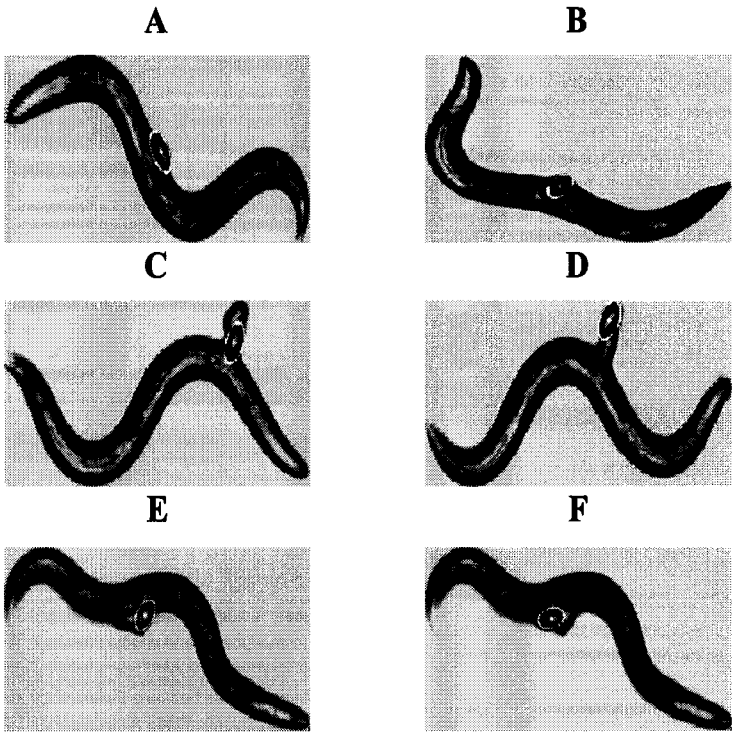


Figure 6.7: Some best-fit results of the deformable template matching. (A) A fully laid egg in perfect condition. (B) A half laid egg. (C-D) Stacked eggs, identified by repeating the search. (E-F) Two eggs laid together in close distance.

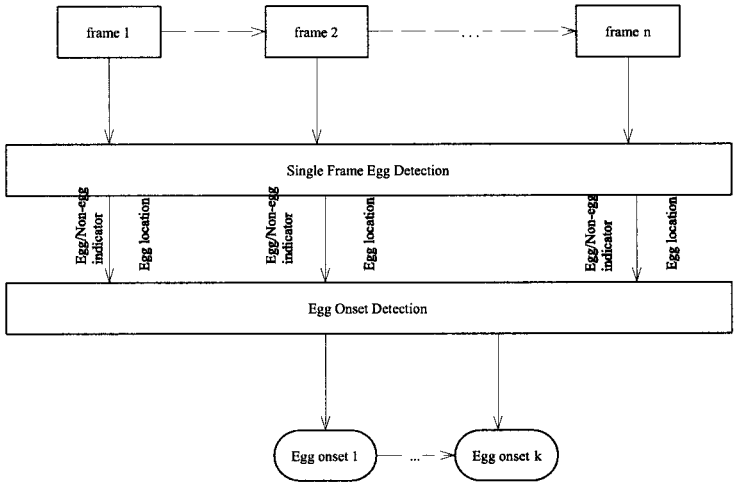


Figure 6.8: Egg event onset detection flow chart.

10,and 18 frames respectively. Assuming 40 frames (20 frames before and 20 after) surrounding each identified egg onset need to be manually verified to eliminate false alarms, the egg onset detection algorithm represents a saving of more than 99% of the time human observers spend on examining all 972000 frames in the 135-hour video.

6.5 Behavior Study

Previous study [35] indicates significantly increasing locomotion activity prior to egg onset. We study the behavior changes before and after 55 wild type egg onsets (a fresh 10-hour recording) detected by our onset detection algorithm. The behavioral characteristics can be summarized by extracting features proposed by the feature extraction system [1][28][29][27][30]. For each feature, we looked for a significant difference in that feature before and after the onset frame by using the non-parametric rank sum test [49] on paired data. For each of the 55 eggs, we pair the data from 40 seconds before the onset frame with 40 seconds of data after the onset frame. Out of the 253 features (see

Chapter 3), 14 are found to be significant at the .01 significance level. We also consider the possibility that some features may be significantly different both before and after egg laying compared to the values for a worm that is not near an egg-laying time. So we also look at the paired data where the values from 40 seconds before an egg-laying onset are paired with the values from an equal number of frames starting from a randomly selected non-egg frame, and similarly where the values from after an egg-laying onset are paired with the values from an equal number of frames starting from a randomly selected non-egg frame. There are 32 comparisons that are significant at the .01 significance level for before, and 32 for after. We notice that, by random chance alone, out of 253 comparisons, we would expect to see 2.5 features to show a significant difference at the .01 significance level.

Most of the features found to be significantly different are related to speed, confirming earlier results that were determined manually. In particular, we find that the global centroid movement, as well as the local movement of the tail and head, are all significantly larger before the onset compared to after (see Figure 6.9). Previous results only considered global movement. Local head movement is often related to foraging behavior. We also find some differences in brightness parameters. Due to the multiplicity of comparisons being made, these remain to be verified when further data are collected.

6.6 Conclusion

We have presented a computer analysis method for attached egg detection and egg onset event detection. The testing results of egg detection on 800,000 frames and 200 eggs from a variety of mutant types and recording conditions illustrate the effectiveness of

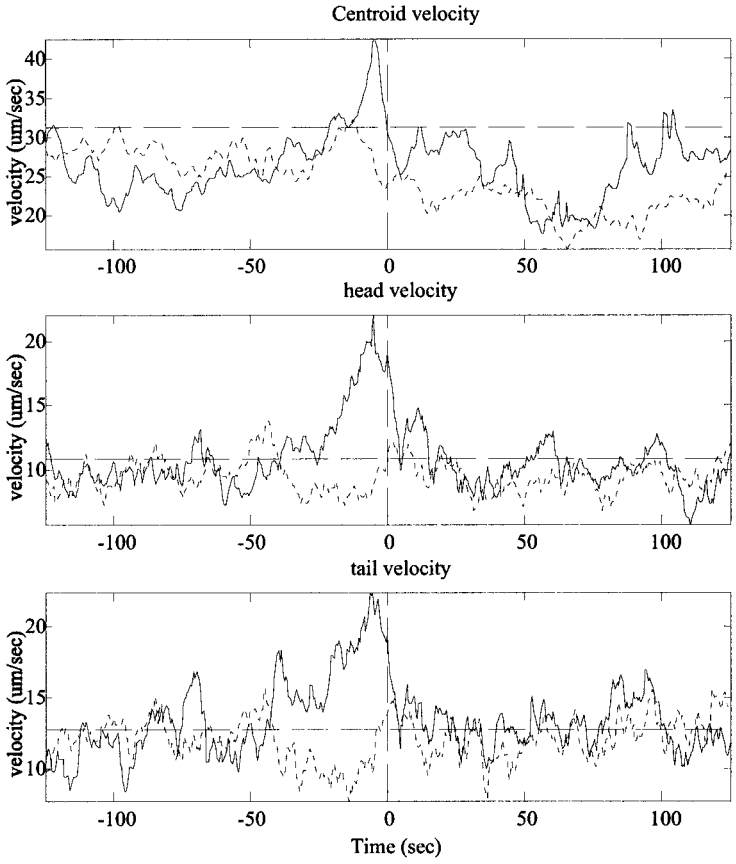


Figure 6.9: Velocity change 125s before and after egg onsets. The velocity is a moving average of 10s interval. The solid curves are the averaged moving velocities surrounding egg onsets. The dotted curves are the averaged moving velocities randomly chosen as reference. The dashed vertical lines mark the egg onsets. The dashed horizontal lines represent the average velocity of the 250s period surrounding egg onsets. (A) Centroid velocity. (B) Head velocity. (C) Tail velocity.

our proposed algorithm. The behavior study of egg onsets confirms the results from previous studies.

Furthermore, it has demonstrated that the machine vision system presented in Chapters 2 and 3 can be used effectively to study specific behaviors. With more accurate and complex computer vision systems [1][28][29][27][30] being developed, we anticipate that many more behavior features will be discovered. Therefore, we will be able to combine the automatic egg onset detection and behavior studies together and explore the temporal correlation between egg-laying and other behavioral characteristics more effectively. Moreover, the ability to automatically detect egg-laying events will make it possible to use these correlations between other behaviors and egg-laying, which previously could only be assayed through time-consuming human analysis of videotapes [35], as automatically-evaluated features for use in phenotype classification and clustering studies [28].

More generally, egg-laying has historically been an extremely useful assay for genetic analysis of diverse aspects of neuromuscular function. For example, egg-laying has provided a behavioral measure for the activity of the Go/Gq signaling network in neurons and muscle cells [3] and for neuromodulation by serotonin, acetylcholine, and neuropeptides [85][91][89]. The egg-laying assays typically used in genetic studies are generally indirect measures of overall egg-laying rate, and consequently allow limited inference about the functions of specific mutant genes in the behavior. Quantitative assays of the temporal pattern of egg-laying can in principle make it possible to distinguish effects on different egg-laying signal transduction pathways [88][89]. The automated methods for egg-detection described here should greatly facilitate these more detailed behavioral analyses.

Part of the chapter will appear in the following publications.

- W. Geng, P. Cosman, and W. R. Schafer. "Egg Onset Detection Using Deformable Template Matching." *IASTED International Conference on Computer Graphics and Image Processing(CGIM2004)*, Kauai, Hawaii, August 2004, to appear.
- W. Geng, P. Cosman, M. Palm, and W. R. Schafer. "*C. elegans* Egg-laying Detection and Behavior Study Using Image Analysis." submitted to *EURASIP Journal on Applied Signal and Image Processing*, January 2004.

I was the primary researcher and the co-author. Dr. Pamela C. Cosman and Dr. William R. Schafer directed and supervised the research which forms the basis for this chapter.

Chapter 7

Summary

This dissertation starts with presenting the idea of using machine vision and statistical learning techniques to study *C. elegans* behavioral phenotypes, therefore gaining insights into their corresponding genotypes. Then the dissertation addresses individual system blocks (segmentation, tracking, feature extraction, statistical learning for natural clustering, classification, and *C. elegans* egg detection). Our contributions in these aspects are summarized next.

7.1 Contributions

- Developed a content based segmentation scheme to improve the segmentation results.
- Created a tracking algorithm to track worm movement.
- Created a comprehensive set of 253 features that measure worm morphological, locomotion, behavior, and texture information.

- Developed a natural clustering scheme to visualize the mutant samples in a low dimensional feature space and also correlate the phenotypes and their underlying genotypes.
- Evaluated and identified that Random Forests classification method is best among a set of classifiers in distinguishing the phenotypes. The important features can also be identified in the process.
- Using the above machine vision system, we have developed automated algorithms and procedures to identify egg-laying events and the behavior changes surrounding the events.

7.2 Application of the System to Genetic Study

7.2.1 Quantitative Definition of Behavioral Mutant Phenotypes

In this dissertation, we have shown that quantitative morphological, locomotion, and texture features obtained from digital video recordings can be used to distinguish the behavioral phenotypes of *C. elegans* mutants. As shown in Table 5.3, a reduced set of approximately 25 features is sufficient to identify visibly dissimilar mutant types with very high reliability. Furthermore, these features can often be used to distinguish between types with highly similar phenotypes (e.g. *unc-2* and *unc-36*) that can not be reliably identified even by an experienced human observer. Thus, the parameters in the reduced feature set are likely to have great utility in assessing subtle or modest abnormalities in behavior caused by hypomorphic mutant alleles or by incompletely penetrant dsRNA inhibition.

These studies also provide insight into the nature of specific mutant phenotypes.

For example, *unc-36*, *unc-29*, *unc-38* and *unc-2* have all been categorized as "weak kinkers", a term that has been difficult to define precisely. From Tables 4.3 and A.1, it is apparent that these mutants share many common effects on the variables used in our classification; in particular, all have a substantially higher angle change rate and substantially lower centroid movement and global speed parameters than wild-type. This combination of characteristics (increased body bending and a decreased rate of movement) thus provides an operational definition of the "kinker" phenotype. Likewise, the combination of increased centroid movement and increased angle change rate provides a functional definition of *goa-1*'s "hyperactive loopy" phenotype, while increased length and length/eccentricity and decreased angle change rate and speed define the "long, slow and floppy" phenotype of *egl-19*. In some cases, significant phenotypic differences are identified that are unnoticed (or unreported) in previous observer-based studies. For example, both *goa-1* and *unc-36* mutants showed particularly large reductions in the ratio of head-to-tail movement, an abnormality whose neural basis could be investigated in future studies. Thus, it has been possible not only to obtain precise quantitative descriptions of phenotypic classes whose definitions have previously been subjective and qualitative, but also to resolve subtle differences within broad classes such as kinker Uncs.

With the collection of larger data sets, it should be possible to use this approach to define and subdivide other widely-cited phenotypic classes of *C. elegans*. For example, it should be possible to obtain precise definitions for other classes of uncoordinated mutants, such as coilers, shrinkers, and loopy mutants. In addition, although we have focused here on the analysis of phenotypes associated with abnormal locomotion, the image parameters we have used in this study could also be used to categorize other classes of behavioral or developmental mutants that involve alterations in body mor-

phology. Such studies would provide valuable insight into the nature of these additional phenotypic types; in addition, it would be interesting from an informatics perspective to learn how the inclusion of genes whose focus of action is outside the neuromuscular system would impact the importance of features used in classification.

7.2.2 Prospects for Using Behavioral Phenotypes for Bioinformatic Analysis

The application of machine-based pattern recognition methods also allows us to probe the similarities between different behavioral patterns based on their clustering in multi-dimensional feature space. In general, the pattern of phenotypic clustering mirrors the known similarities in molecular function and cellular site of action of the mutant gene products. For example, *unc-29* and *unc-38*, which respectively encode α and β nicotinic receptor subunits with overlapping expression patterns, formed a single cluster in the optimal clustering and have centers that are the closest together by Euclidean distance (Figure 4.4). Likewise, *unc-2* and *unc-36* mutants, which are defective in the $\alpha - 1$ and $\alpha - 2$ subunits respectively of the neuronal N-type calcium channel, form a single cluster in the optimal k-means clustering, and the centers of these two types' data clouds are relatively close in feature space. In fact, the centers for all four of these types (which have all been designated as kinker Uncs and all encode excitatory ion channels whose focus of action is primarily at body muscle neuromuscular junctions) are closer to one another than to the other Unc mutants or to wild-type. Thus, the quantitative phenotypic signature obtained through behavioral tracking appears to correspond well to the underlying functional defects of the mutants we analyzed.

We anticipate that this type of comprehensive quantification of mutant behavioral phenotypes will have powerful applications in functional genomic studies. Clustering and pattern recognition analysis of microarray-derived gene expression profiles has provided important information about the likely functions of novel gene products in *C. elegans* and other organisms [47]. In principle, a behavioral phenotype represents a similarly complex quantitative signature whose direct linkage to nervous system activity makes it particularly useful for classifying genes that function in excitable cells. In several genome-wide deletion and RNAi-based knockout surveys undertaken in *C. elegans*, the identification and classification of behavioral and other non-lethal phenotypes has been a crucial limiting factor ([26][96]). Using the machine-based phenotyping approaches described here, it should be possible to record the behavior of an uncharacterized knockout strain, compare its phenotypic pattern to a database of known mutants, and make an informed initial hypothesis about the molecular pathways in which the mutant gene product participates.

7.2.3 Applications for Computer Vision-based Quantification of Mutant Phenotypes

Hundreds of genes have been identified in *C. elegans* that affect behavior and morphology in specific ways. Our long-term aim is to collect data on large numbers of mutant types and effectively classify them according to their phenotypic similarity. With an increasing set of mutant types, it becomes progressively more challenging to identify features that effectively classify and distinguish the large variety of worm types. The image processing methods developed here, including new features that require accurately identifying the head and tail regions of the animal, allow us to achieve high classification

accuracy even for a data set involving 16 different mutant types with subtle distinctions that are hard to classify by eye.

7.2.4 Application of Computer Vision System to Specific Behavior Study

The image processing and computer vision methods developed in this dissertation are basic building blocks for studying specific *C. elegans* behaviors. For example, Chapter 6 shows the methods to automatically detect egg-laying events which allow the study of egg-laying events on a large scale. We anticipate that similar studies can be readily conducted using similar principles. For example, an interesting study could be automated reversal detection and the behavioral changes surrounding the reversal events. Future studies with multi-animal behaviors such as mating, social feeding, etc, can also take advantage of the methods developed. The algorithms developed here for tracking, head/tail recognition and feature extraction will be an essential part of a completely automated *C. elegans* tracking and identification system.

Appendix A

FEATURE DESCRPTIONS

Table A.1: 253 features statistics and descriptions.

		worm type							
Variable	**Stat.**	**wild**	***goa1***	***nic1***	***uc36***	***uc38***	***uc29***	***eg19***	***uc2***
AREAMIN	Mean	7867	6054	5669	6377	7152	7318	7755	7040
(min area of worm)	Std	596.3	550.7	735.5	558.5	564.9	628.7	661.4	494.8
AREAMAX	Mean	8440	6501	5925	6744	7560	7741	8222	7486
(max area of worm)	Std	715.9	537.4	750.0	573.0	614.9	666.0	652.8	630.4
AREAAVG	Mean	8132	6272	5798	6557	7347	7527	7990	7266
(avg area of worm)	Std	636.8	532.5	742.6	563.2	579.6	635.0	622.3	567.8
HGHTMIN	Mean	97.9	91.1	111.4	100.5	103.2	104.5	116.3	114.7
(min height of MER)	Std	21.5	13.6	48.2	29.6	28.2	24.5	31.8	32.4
HGHTMAX	Mean	282.8	227.5	167.8	240.6	231.2	241.2	304.5	233.4
(maxheight of MER)	Std	21.6	22.9	50.4	31.3	32.3	26.9	28.6	31.7
HGHTAVG	Mean	195.4	158.9	139.1	171.3	167.6	172.7	216.7	173.7
(avg height of MER)	Std	24.6	19.4	49.7	29.5	29.8	25.2	32.8	30.8
WDTHMIN	Mean	101.3	92.7	121.2	100.5	109.1	107.8	107.9	109.4
(min width of MER)	Std	22.8	15.3	51.4	26.1	28.7	22.7	26.1	27.2
WDTHMAX	Mean	284.6	226.8	178.4	236.7	234.8	242.7	296.4	227.1
(max width of MER)	Std	16.5	19.7	53.8	31.5	27.8	30.9	30.6	34.8

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
WDTHAVG	Mean	199.9	160.9	151.5	171.0	173.4	175.0	202.8	168.5
(avg width of MER)	Std	22.5	19.0	52.5	28.7	28.3	25.0	30.5	30.4
LNGTHMIN	Mean	277.8	232.0	197.7	255.5	251.6	263.0	307.9	264.1
(min length)	Std	13.1	12.6	18.7	10.0	10.9	14.1	17.7	13.1
LNGTHMAX	Mean	299.1	254.5	217.4	276.2	273.9	288.6	331.2	288.0
(max length)	Std	13.7	13.3	20.4	10.9	12.4	13.7	14.5	11.0
LNGTHAVG	Mean	288.9	243.4	207.6	266.0	262.9	276.2	320.1	276.0
(avg length)	Std	13.3	12.8	19.6	10.2	11.6	13.5	14.1	10.7
WHRATMIN	Mean	0.4	0.4	0.9	0.5	0.5	0.5	0.4	0.5
(min width-to-height ratio of MER)	Std	0.1	0.2	0.7	0.2	0.3	0.2	0.2	0.2
WHRATMAX	Mean	3.0	2.5	2.1	2.5	2.4	2.4	2.7	2.1
(max width-to-height ratio of MER)	Std	0.7	0.5	1.5	0.9	0.7	0.7	0.9	0.7
WHRATAVG	Mean	1.4	1.3	1.5	1.3	1.3	1.3	1.3	1.2
(avg width-to-height ratio of MER)	Std	0.3	0.3	1.0	0.5	0.4	0.3	0.4	0.4
MERFLMIN	Mean	0.2	0.2	0.3	0.2	0.2	0.2	0.2	0.2
(min ratio of worm area to MER area)	Std	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
MERFLMAX	Mean	0.4	0.4	0.4	0.4	0.4	0.4	0.3	0.4
(max ratio of worm area to MER area)	Std	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.1
MERFLAVG	Mean	0.3	0.3	0.3	0.3	0.3	0.3	0.2	0.3
(avg ratio of worm area to MER area)	Std	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
MAJORMIN	Mean	274.2	191.9	195.1	216.4	215.7	210.3	279.1	204.9
(min length of best-fit ellipse's major axis)	Std	19.1	22.0	35.5	24.0	27.9	25.4	41.0	31.0
MAJORMAX	Mean	326.9	268.2	240.0	290.6	283.8	294.1	350.3	296.3
(max length of best-fit ellipse's major axis)	Std	15.4	16.6	26.0	14.1	18.0	17.0	19.8	20.6
MAJORAVG	Mean	304.8	237.8	219.9	258.3	252.5	255.8	320.0	253.2
(avg length of best-fit ellipse's major axis)	Std	14.9	16.7	29.0	16.0	21.0	19.5	26.7	23.3
MINORMIN	Mean	52.2	54.9	42.9	55.6	57.2	61.3	60.4	60.2
(min length of best-fit ellipse's minor axis)	Std	3.5	4.6	9.4	6.0	8.6	7.6	8.2	7.5
MINORMAX	Mean	93.6	94.9	72.7	94.1	98.7	109.2	110.8	104.6
(max length of best-fit ellipse's minor axis)	Std	10.5	7.2	17.2	13.7	12.4	11.6	15.8	14.8
MINORAVG	Mean	70.0	73.3	56.4	72.8	76.5	84.1	82.9	81.1
(avg length of best-fit ellipse's minor axis)	Std	5.7	5.4	12.4	8.1	9.7	9.1	10.6	9.8
ECCTYMIN	Mean	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
(min best-fit ellipse's eccentricity)	Std	0.0	0.0	0.1	0.1	0.1	0.1	0.1	0.1

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
ECCTYMAX	Mean	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
(max best-fit ellipse's eccentricity)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
ECCTYAVG	Mean	1.0	0.9	1.0	1.0	0.9	0.9	1.0	0.9
(avg best-fit ellipse's eccentricity)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MVHLFMIN	Mean	0.2	0.3	0.0	0.0	0.0	0.0	0.1	0.0
(min distance moved in 0.5 seconds)	Std	0.5	0.7	0.0	0.1	0.0	0.1	0.2	0.1
MVHLFMAX	Mean	59.6	53.1	5.7	18.1	19.7	25.7	35.7	12.2
(max distance moved in 0.5 seconds)	Std	8.4	8.5	3.4	6.6	5.9	7.9	33.2	3.4
MVHLFAVG	Mean	24.3	24.5	1.0	6.4	6.1	8.3	11.0	3.7
(avg distance moved in 0.5 seconds)	Std	5.6	7.3	0.4	2.3	2.5	3.6	3.2	1.8
PRP10MIN	Mean	13.3	7.3	0.3	3.7	4.4	6.1	12.5	2.6
(min distance moved in 5 sec or 10 frames)	Std	11.7	10.4	0.6	4.1	6.9	9.0	13.5	2.8
PRP10MAX	Mean	412.3	376.4	15.5	118.3	120.1	160.6	198.7	81.6
(max distance moved in 5 sec or 10 frames)	Std	74.4	55.7	10.4	33.9	39.3	64.6	46.4	27.7
PRP10AVG	Mean	198.5	165.6	4.9	57.1	53.8	69.6	100.6	33.4
(avg distance moved in 5 sec or 10 frames)	Std	53.6	58.3	3.2	23.5	24.4	35.7	32.0	18.6
PRP20MIN	Mean	44.8	28.4	0.9	8.6	9.2	19.4	35.7	7.2
(min distance moved in 10 sec or 20 frames)	Std	41.4	40.4	1.1	9.7	9.8	21.7	37.8	8.0
PRP20MAX	Mean	740.8	673.4	20.5	211.7	214.1	273.0	340.7	143.2
(max distance moved in 10 sec or 20 frames)	Std	146.3	124.4	16.4	72.5	79.4	116.2	86.8	56.7
PRP20AVG	Mean	370.0	284.9	7.1	98.4	98.5	128.5	183.6	62.8
(avg distance moved in 10 sec or 20 frames)	Std	106.1	104.7	6.0	47.8	47.6	67.2	64.9	37.8
PRP30MIN	Mean	91.0	64.3	1.6	16.9	21.7	37.0	59.5	12.6
(min distance moved in 15 sec or 30 frames)	Std	80.6	86.5	2.7	18.8	29.5	46.1	67.6	14.8
PRP30MAX	Mean	1028.8	864.9	24.9	291.3	294.3	364.6	466.4	196.7
(max distance moved in 15 sec or 30 frames)	Std	217.1	218.1	23.2	114.6	115.9	155.9	133.2	85.9
PRP30AVG	Mean	535.5	394.4	9.0	133.8	139.9	180.3	258.8	88.0
(avg distance moved in 15 sec or 30 frames)	Std	156.8	148.9	8.9	72.5	71.1	94.9	98.8	55.1
PRP40MIN	Mean	139.8	89.7	2.5	23.8	31.2	58.9	89.8	21.6
(min distance moved in 20 sec or 40 frames)	Std	113.5	118.4	4.0	31.7	38.9	62.9	95.1	32.1
PRP40MAX	Mean	1275.9	1038.6	27.2	358.1	371.4	439.1	593.5	235.6
(max distance moved in 20 sec or 40 frames)	Std	290.1	285.3	26.0	156.1	151.6	199.8	176.5	109.6
PRP40AVG	Mean	679.6	482.7	10.9	167.4	179.9	225.5	331.2	112.1
(avg distance moved in 20 sec or 40 frames)	Std	202.8	192.1	11.8	94.5	92.7	115.1	132.7	74.0

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
PRP50MIN	Mean	217.8	142.1	3.1	33.5	55.6	88.5	123.3	42.1
(min distance moved in 25 sec or 50 frames)	Std	173.7	172.3	6.5	40.0	71.5	85.9	123.7	61.9
PRP50MAX	Mean	1476.9	1140.7	30.0	415.1	437.9	507.9	700.2	269.3
(max distance moved in 25 sec or 50 frames)	Std	364.6	332.9	30.5	194.8	187.2	239.8	229.4	136.0
PRP50AVG	Mean	818.6	568.9	12.4	200.2	223.8	273.0	403.5	139.6
(avg distance moved in 25 sec or 50 frames)	Std	251.7	239.0	14.3	119.7	118.3	141.9	162.9	98.0
PRP60MIN	Mean	315.6	175.7	3.5	46.0	64.9	122.6	166.7	57.1
(min distance moved in 30 sec or 60 frames)	Std	237.2	182.4	6.1	49.1	77.5	126.2	165.2	93.9
PRP60MAX	Mean	1648.4	1216.9	30.8	466.2	487.4	573.7	802.1	291.3
(max distance moved in 30 sec or 60 frames)	Std	422.4	380.0	31.6	242.3	215.4	279.9	277.7	164.7
PRP60AVG	Mean	953.6	642.1	14.0	226.2	255.5	313.3	465.6	157.8
(avg distance moved in 30 sec or 60 frames)	Std	296.4	263.2	16.9	143.3	137.8	167.5	198.4	119.3
TOTMOVE	Mean	13645	13595	576	3134	2807	3360	5854	1415
(total amount of movement in 5 minutes)	Std	3195	4142	206	1313	1197	1517	1884	796
RV20MAX	Mean	4.8	5.0	0.6	2.3	2.3	3.1	2.3	1.4
(max number of reversals in 10 sec)	Std	1.2	1.1	0.7	0.5	0.8	1.0	0.9	0.6
RV20AVG	Mean	1.2	1.7	0.1	0.5	0.4	0.6	0.4	0.3
(avg number of reversals in 10 sec)	Std	0.5	0.5	0.1	0.2	0.2	0.3	0.3	0.1
RV40MIN	Mean	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0
(min number of reversals in 20 sec)	Std	0.2	0.6	0.0	0.1	0.0	0.0	0.0	0.0
RV40MAX	Mean	7.1	7.4	0.7	3.2	3.2	4.2	3.1	1.9
(max number of reversals in 20 sec)	Std	2.0	1.8	0.8	0.8	1.2	1.4	1.5	0.7
RV40AVG	Mean	2.4	3.3	0.1	1.1	0.8	1.1	0.8	0.5
(avg number of reversals in 20 sec)	Std	1.1	1.0	0.1	0.4	0.4	0.5	0.6	0.3
RV60MIN	Mean	0.3	1.3	0.0	0.1	0.0	0.0	0.0	0.0
(min number of reversals in 30 sec)	Std	0.8	1.3	0.0	0.4	0.1	0.1	0.1	0.0
RV60MAX	Mean	8.9	9.4	0.7	4.0	3.8	5.0	3.9	2.3
(max number of reversals in 30 sec)	Std	2.7	2.4	0.9	1.1	1.6	1.8	2.0	0.9
RV60AVG	Mean	3.6	5.0	0.1	1.6	1.2	1.7	1.3	0.8
(avg number of reversals in 30 sec)	Std	1.7	1.5	0.2	0.7	0.7	0.8	0.9	0.4
RV80MIN	Mean	0.9	2.5	0.0	0.3	0.1	0.1	0.1	0.0
(min number of reversals in 40 sec)	Std	1.4	1.8	0.0	0.5	0.3	0.3	0.4	0.2
RV80MAX	Mean	10.5	11.3	0.8	4.7	4.3	5.8	4.5	2.6
(max number of reversals in 40 sec)	Std	3.4	3.0	1.0	1.5	1.9	2.2	2.5	1.1

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
RV80AVG	Mean	4.9	6.6	0.2	2.1	1.7	2.3	1.7	1.1
(avg number of reversals in 40 sec)	Std	2.2	2.0	0.3	0.9	0.9	1.1	1.3	0.6
RV100MIN	Mean	1.7	3.8	0.0	0.6	0.2	0.2	0.2	0.2
(min number of reversals in 50 sec)	Std	2.0	2.4	0.0	0.8	0.6	0.6	0.7	0.4
RV100MAX	Mean	11.8	13.1	0.8	5.3	4.8	6.2	5.0	2.9
(max number of reversals in 50 sec)	Std	4.2	3.5	1.1	1.7	2.2	2.5	2.8	1.3
RV100AVG	Mean	6.1	8.2	0.2	2.7	2.1	2.8	2.1	1.4
(avg number of reversals in 50 sec)	Std	2.8	2.6	0.4	1.2	1.2	1.4	1.6	0.7
RV120MIN	Mean	2.5	5.3	0.0	0.9	0.4	0.5	0.4	0.3
(min number of reversals in 60 sec)	Std	2.7	3.0	0.1	1.1	0.9	0.9	0.9	0.5
RV120MAX	Mean	12.9	14.8	0.9	5.9	5.2	6.9	5.4	3.2
(max number of reversals in 60 sec)	Std	4.9	4.0	1.1	1.9	2.4	2.8	3.2	1.4
RV120AVG	Mean	7.3	9.9	0.3	3.2	2.5	3.4	2.5	1.6
(avg number of reversals in 60 sec)	Std	3.4	3.1	0.5	1.4	1.5	1.7	2.0	0.9
TOTRV	Mean	29.2	39.8	1.1	12.5	9.3	13.3	9.8	6.3
(total number of reversals in 5 minutes)	Std	12.6	12.3	1.6	5.0	4.9	6.0	7.1	3.1
HDTHKMIN	Mean	9.4	9.2	9.5	8.8	9.6	9.6	8.7	10.0
(min head width)	Std	0.9	1.5	1.1	1.4	1.0	0.9	0.8	1.0
HDTHKMAX	Mean	12.6	13.1	13.3	12.8	12.8	12.6	11.3	14.2
(max head width)	Std	0.7	0.7	0.9	1.3	0.8	0.9	1.6	1.2
HDTHKAVG	Mean	11.0	11.2	11.4	10.8	11.3	11.1	9.9	12.2
(avg head width)	Std	0.7	0.9	0.7	1.1	0.8	0.7	0.6	0.9
TLTHKMIN	Mean	8.2	8.2	8.9	7.7	7.1	6.4	8.1	7.7
(min tail width)	Std	1.1	1.3	1.6	1.3	0.8	1.0	0.8	1.1
TLTHKMAX	Mean	12.9	12.8	12.8	12.1	13.1	12.6	11.3	12.3
(max tail width)	Std	1.7	1.2	1.4	1.5	1.4	2.4	1.3	1.5
TLTHKAVG	Mean	10.6	10.6	10.9	9.9	10.1	9.5	9.7	10.2
(avg tail width)	Std	1.2	1.1	1.2	1.0	0.9	1.3	0.8	1.0
CNTHKMIN	Mean	25.7	23.0	25.3	22.4	25.5	24.7	24.0	23.8
(min center width)	Std	1.3	1.4	2.1	1.5	1.6	1.8	1.5	1.4
CNTHKMAX	Mean	28.1	25.0	27.7	24.5	27.7	26.8	26.0	26.4
(max center width)	Std	1.9	1.4	2.5	1.9	1.8	2.2	1.8	2.7
CNTHKAVG	Mean	26.8	24.0	26.4	23.4	26.5	25.8	25.0	25.1
(avg center width)	Std	1.4	1.4	2.2	1.6	1.6	1.9	1.5	2.0

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
HDLNRMIN	Mean	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
(min head width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HDLNRMAX	Mean	0.0	0.1	0.1	0.1	0.1	0.1	0.0	0.1
(max head width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HDLNRAVG	Mean	0.0	0.1	0.1	0.0	0.0	0.0	0.0	0.0
(avg head width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TLLNRMIN	Mean	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
(min tail width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TLLNRMAX	Mean	0.1	0.1	0.1	0.1	0.1	0.1	0.0	0.1
(max tail width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TLLNRAVG	Mean	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
(avg tail width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CNLNRMIN	Mean	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
(min center width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CNLNRMAX	Mean	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
(max center width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CNLNRAVG	Mean	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
(avg center width to length ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
HDTLTRMIN	Mean	0.8	0.8	0.8	0.8	0.8	0.9	0.9	0.9
(min head to tail width ratio)	Std	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
HDTLTRMAX	Mean	1.4	1.4	1.4	1.5	1.6	1.7	1.3	1.6
(max head to tail width ratio)	Std	0.2	0.2	0.3	0.4	0.2	0.3	0.1	0.2
HDTLTRAVG	Mean	1.1	1.1	1.1	1.1	1.2	1.3	1.1	1.2
(avg head to tail width ratio)	Std	0.1	0.1	0.2	0.2	0.1	0.2	0.1	0.1
HCTHRMIN	Mean	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
(min head to center width ratio)	Std	0.0	0.1	0.1	0.1	0.0	0.0	0.0	0.0
HCTHRMAX	Mean	0.5	0.6	0.5	0.6	0.5	0.5	0.5	0.6
(max head to center width ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.0
HCTHRAVG	Mean	0.4	0.5	0.5	0.5	0.5	0.5	0.4	0.5
(avg head to center width ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TCTHRMIN	Mean	0.3	0.4	0.4	0.4	0.3	0.3	0.4	0.3
(min tail to center width ratio)	Std	0.0	0.1	0.1	0.1	0.0	0.0	0.0	0.1
TCTHRMAX	Mean	0.5	0.6	0.5	0.5	0.5	0.5	0.5	0.5
(max tail to center width ratio)	Std	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
TCTHRAVG	Mean	0.4	0.5	0.4	0.5	0.4	0.4	0.4	0.4
(avg tail to center width ratio)	Std	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
AMPMIN	Mean	37.4	40.4	26.5	42.1	42.6	48.0	46.8	47.9
(min amplitude of worm skeleton wave)	Std	3.4	4.6	9.4	6.0	7.0	7.5	8.9	6.7
AMPMAX	Mean	79.0	82.2	57.8	78.9	83.4	92.5	95.6	89.4
(max amplitude of worm skeleton wave)	Std	9.9	7.4	14.6	12.6	11.6	10.5	16.2	12.2
AMPAVG	Mean	55.4	59.3	41.1	58.9	61.9	69.8	69.4	67.5
(avg amplitude of worm skeleton wave)	Std	5.8	5.5	11.3	8.3	9.1	8.8	10.8	8.7
AMPRMIN	Mean	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
(min amplitude ratio)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
AMPRMAX	Mean	0.8	0.8	0.6	0.7	0.8	0.8	0.7	0.7
(max amplitude ratio)	Std	0.0	0.0	0.2	0.1	0.1	0.1	0.1	0.1
AMPRAVG	Mean	0.3	0.4	0.2	0.3	0.3	0.3	0.3	0.3
(avg amplitude ratio)	Std	0.0	0.1	0.1	0.1	0.1	0.1	0.1	0.1
ANCHRMIN	Mean	2.8	4.1	4.5	3.9	4.0	3.7	2.3	4.0
(min angle change rate)	Std	0.3	0.6	1.1	0.5	0.6	0.5	0.4	0.6
ANCHRMAX	Mean	3.9	6.7	7.4	6.4	6.0	5.9	3.5	6.9
(max angle change rate)	Std	0.4	0.9	1.8	0.8	0.8	0.9	0.6	1.0
ANCHRAVG	Mean	3.3	5.2	5.9	5.0	5.0	4.7	2.8	5.4
(avg angle change rate)	Std	0.3	0.7	1.4	0.6	0.7	0.7	0.5	0.8
ANCHSMIN	Mean	1.8	2.7	3.0	2.4	2.6	2.4	1.6	2.6
(min standard deviation of angle change)	Std	0.2	0.3	0.7	0.3	0.3	0.3	0.2	0.3
ANCHSMAX	Mean	2.7	4.5	5.2	3.9	4.0	3.9	2.4	4.2
(max standard deviation of angle change)	Std	0.2	0.6	1.1	0.4	0.5	0.6	0.4	0.5
ANCHSAVG	Mean	2.2	3.4	4.0	3.1	3.2	3.1	2.0	3.4
(avg standard deviation of angle change)	Std	0.2	0.4	0.9	0.3	0.4	0.4	0.3	0.4
LNMFRMIN	Mean	791.8	627.6	528.1	737.7	683.0	722.6	1047.2	730.4
(min ratio of worm length to MER fill)	Std	77.0	72.0	148.9	87.1	80.5	88.7	118.6	114.5
LNMFRMAX	Mean	1634	1206	807	1452	1231	1347	2078	1384
(max ratio of worm length to MER fill)	Std	141.0	131.7	199.9	147.7	137.5	178.3	215.7	208.1
LNMFRAVG	Mean	1225	906	667	1072	948	1021	1558	1037
(avg ratio of worm length to MER fill)	Std	113.7	94.6	175.1	109.2	110.2	129.8	175.6	160.7
LNECRMIN	Mean	285.6	243.6	205.5	265.7	262.9	277.0	317.8	278.3
(min ratio of worm length to eccentricity)	Std	13.7	12.9	19.6	10.9	12.2	15.9	18.3	11.5
LNECRMAX	Mean	313.1	282.0	237.3	301.9	304.9	328.1	358.9	330.5
(max ratio of worm length to eccentricity)	Std	15.7	18.4	34.4	24.6	30.5	28.2	24.2	38.3

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
LNECRAVG	Mean	299.9	262.8	220.3	283.4	282.8	301.2	337.5	301.3
(avg ratio of worm length to eccentricity)	Std	14.2	14.2	22.4	15.0	18.0	18.4	16.6	17.2
FATMIN	Mean	26.8	24.5	26.7	23.6	26.7	26.0	23.9	25.2
(min fatness = length/area)	Std	1.3	1.4	1.7	1.5	1.5	1.8	1.1	1.5
FATMAX	Mean	29.7	27.1	29.2	25.8	29.3	28.7	26.0	27.6
(max fatness = length/area)	Std	1.9	1.4	1.9	1.8	1.8	2.3	1.3	2.3
FATAVG	Mean	28.2	25.8	27.9	24.7	28.0	27.3	25.0	26.4
(avg fatness = length/area)	Std	1.5	1.4	1.8	1.6	1.6	2.0	1.2	1.8
LNWDRMIN	Mean	1.0	1.1	1.3	1.1	1.1	1.2	1.1	1.2
(min of worm length to MER width ratio)	Std	0.1	0.1	0.5	0.2	0.2	0.2	0.1	0.2
LNWDRMAX	Mean	3.0	2.6	2.1	2.8	2.5	2.7	3.1	2.6
(max of worm length to MER width ratio)	Std	0.6	0.4	1.0	0.6	0.6	0.6	0.6	0.6
LNWDRAVG	Mean	1.7	1.7	1.6	1.8	1.7	1.8	1.9	1.8
(avg of worm length to MER width ratio)	Std	0.3	0.2	0.7	0.4	0.3	0.4	0.3	0.4
CNTMVMIN	Mean	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
(min of normalized centroid movement)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CNTMVMAX	Mean	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.1
(max of normalized centroid movement)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
CNTMVAVG	Mean	0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.0
(avg of normalized centroid movement)	Std	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
NUMOMEGA	Mean	75.0	55.1	125.8	97.8	96.1	92.5	95.8	87.0
(total number of frames the worm has omega shape)	Std	16.1	20.9	70.3	50.7	48.5	45.0	33.7	39.1
OMEGACHG	Mean	50.0	38.1	37.2	26.7	29.1	28.0	32.0	22.5
(number of times the worm changes from non-omega shape to omega shape)	Std	9.1	12.6	13.2	8.0	8.7	8.6	7.4	7.4
HEADBRMIN	Mean	67.1	68.6	70.8	72.5	70.0	70.8	78.5	71.8
(min head brightness)	Std	4.9	7.2	7.2	8.1	5.7	6.0	5.7	6.9
HEADBRMAX	Mean	91.1	94.6	91.0	95.3	93.3	94.1	99.1	95.9
(max head brightness)	Std	5.4	8.0	7.6	8.4	6.2	6.9	7.0	7.9
HEADBRAVG	Mean	79.5	82.0	81.1	84.1	81.8	82.5	88.9	84.4
(avg head brightness)	Std	5.0	7.6	7.2	8.1	5.9	6.3	5.9	7.3
TAILBRMIN	Mean	49.8	49.1	58.1	71.2	53.3	51.8	65.6	68.3
(min tail brightness)	Std	3.6	4.8	6.9	7.8	3.8	3.6	5.7	7.7
TAILBRMAX	Mean	67.3	64.2	73.2	92.5	70.8	68.1	85.5	87.9
(max tail brightness)	Std	4.9	6.8	7.4	8.6	5.3	5.5	7.4	8.8

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
TAILBRAVG	Mean	58.9	56.7	65.7	82.1	62.0	59.9	75.5	78.4
(avg tail brightness)	Std	4.1	5.7	7.1	8.2	4.5	4.3	6.2	8.1
CNTBRMIN	Mean	39.0	46.3	47.3	48.7	41.4	43.1	45.8	49.1
(min center brightness)	Std	2.3	6.6	5.1	5.7	2.8	2.7	3.2	4.0
CNTBRMAX	Mean	48.9	57.3	54.5	59.4	49.6	51.4	57.6	58.4
(max center brightness)	Std	3.0	7.1	6.1	6.2	3.4	3.4	7.5	4.4
CNTBRAVG	Mean	44.0	52.1	51.0	54.3	45.6	47.4	51.5	53.9
(avg center brightness)	Std	2.6	6.9	5.6	5.9	3.0	2.9	3.8	4.1
AVGBRMIN	Mean	45.6	51.3	53.3	56.7	48.5	49.7	53.7	56.8
(min whole body brightness)	Std	2.4	5.9	4.8	6.0	2.9	2.8	3.4	4.5
AVGBRMAX	Mean	55.6	61.8	60.6	67.5	56.6	57.8	65.7	66.1
(max whole body brightness)	Std	2.9	6.2	5.7	6.2	3.3	3.5	7.0	4.9
AVGBRAVG	Mean	50.8	56.9	57.0	62.3	52.7	53.9	59.6	61.6
(avg whole body brightness)	Std	2.6	6.1	5.2	6.0	3.0	3.1	3.9	4.6
HTBRRMIN	Mean	1.1	1.2	1.1	0.9	1.1	1.1	1.0	0.9
(min head/tail brightness)	Std	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
HTBRRMAX	Mean	1.7	1.8	1.5	1.2	1.6	1.7	1.4	1.3
(max head/tail brightness)	Std	0.1	0.2	0.2	0.1	0.1	0.1	0.1	0.1
HTBRRAVG	Mean	1.4	1.5	1.3	1.0	1.3	1.4	1.2	1.1
(avg head/tail brightness)	Std	0.1	0.1	0.2	0.1	0.1	0.1	0.1	0.1
HTBRDMIN	Mean	24.4	17.2	20.4	19.0	24.7	23.9	28.6	18.4
(min head brightness - tail brightness)	Std	4.2	6.1	6.2	5.0	5.3	5.4	4.8	5.2
HTBRDMAX	Mean	46.1	41.7	39.3	40.1	47.4	46.1	45.8	41.6
(max head brightness - tail brightness)	Std	4.2	6.5	6.0	5.2	5.2	5.6	5.0	6.2
HTBRDAVG	Mean	35.5	29.8	30.1	29.8	36.2	35.1	37.4	30.5
(avg head brightness - tail brightness)	Std	4.2	6.2	5.7	5.0	5.2	5.3	4.6	5.5
HEADWDMIN	Mean	17.7	16.6	17.1	16.1	17.8	17.6	15.2	17.6
(min of head area width average)	Std	1.3	1.5	1.8	1.5	1.5	1.3	1.1	1.2
HEADWDMAX	Mean	21.6	21.1	21.6	19.9	21.2	20.9	17.9	21.0
(max of head area width average)	Std	1.5	1.2	2.1	1.7	1.6	2.0	1.4	1.6
HEADWDAVG	Mean	19.6	18.8	19.2	18.0	19.4	19.2	16.6	19.3
(avg of head area width average)	Std	1.2	1.2	1.5	1.4	1.4	1.5	1.0	1.2
TAILWDMIN	Mean	18.6	16.9	18.6	15.8	18.2	17.5	16.1	17.0
(min of tail area width average)	Std	1.4	1.1	1.5	1.3	1.4	1.6	1.0	1.3

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
TAILWDMAX	Mean	22.8	20.5	22.3	19.4	22.9	22.4	19.2	21.0
(max of tail area width average)	Std	1.9	1.4	2.1	1.7	1.9	2.4	1.2	2.2
TAILWDAVG	Mean	20.7	18.8	20.4	17.6	20.6	20.0	17.7	19.0
(avg of tail area width average)	Std	1.5	1.2	1.6	1.4	1.5	1.8	1.0	1.6
CNTWDMIN	Mean	25.0	22.5	25.0	21.5	25.0	24.3	22.4	22.9
(min of center area width average)	Std	1.2	1.4	1.9	1.4	1.5	1.8	1.1	1.4
CNTWDMAX	Mean	26.8	23.9	26.2	22.9	26.6	25.8	23.7	24.6
(max of center area width average)	Std	1.9	1.3	2.0	1.7	1.8	2.2	1.4	2.1
CNTWDAVG	Mean	25.8	23.2	25.6	22.2	25.7	25.0	23.0	23.8
(avg of center area width average)	Std	1.4	1.3	1.9	1.5	1.5	1.9	1.2	1.7
AVGWDMIN	Mean	23.0	21.0	23.0	20.0	23.1	22.4	20.5	21.4
(min of whole body width average)	Std	1.2	1.3	1.6	1.3	1.4	1.6	1.0	1.3
AVGWDMAX	Mean	25.0	22.6	24.4	21.5	24.7	24.1	21.7	23.0
(max of whole body width average)	Std	1.7	1.2	1.7	1.6	1.6	2.0	1.3	1.8
AVGWDAVG	Mean	23.9	21.7	23.7	20.7	23.8	23.2	21.1	22.2
(avg of whole body width average)	Std	1.3	1.2	1.6	1.4	1.4	1.8	1.1	1.5
HTWDRMIN	Mean	0.8	0.8	0.8	0.9	0.8	0.8	0.8	0.9
(min head/tail width ratio)	Std	0.0	0.1	0.1	0.1	0.1	0.1	0.1	0.1
HTWDRMAX	Mean	1.1	1.2	1.1	1.2	1.1	1.1	1.1	1.2
(max head/tail width ratio)	Std	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
HTWDRAVG	Mean	1.0	1.0	1.0	1.0	1.0	1.0	0.9	1.0
(avg head/tail width ratio)	Std	0.0	0.1	0.1	0.1	0.1	0.1	0.1	0.1
HBBDRMIN	Mean	0.8	0.8	0.7	0.8	0.8	0.8	0.7	0.8
(min head/whole body brightness)	Std	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
HBBDRMAX	Mean	0.9	1.0	0.9	0.9	0.9	0.9	0.8	0.9
(max head/whole body brightness)	Std	0.0	0.0	0.1	0.0	0.0	0.0	0.1	0.0
HBBDRAVG	Mean	0.8	0.9	0.8	0.9	0.8	0.8	0.8	0.9
(avg head/whole body brightness)	Std	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
HANGCRMIN	Mean	4.6	4.2	4.4	4.7	4.5	4.3	3.9	5.3
(min head area angle change rate)	Std	0.3	0.5	0.6	0.6	0.4	0.4	0.3	0.7
HANGCRMAX	Mean	11.1	12.0	14.2	12.4	11.3	10.6	10.0	14.0
(max head area angle change rate)	Std	0.7	0.9	2.1	1.2	0.8	0.9	0.4	1.6
HANGCRAVG	Mean	7.8	8.0	9.1	8.4	7.9	7.4	6.9	9.5
(avg head area angle change rate)	Std	0.4	0.6	1.1	0.8	0.6	0.5	0.3	1.1

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
TANGCRMIN	Mean	4.2	5.4	5.1	5.3	5.7	5.7	4.4	5.7
(min tail area angle change rate)	Std	0.3	0.5	1.2	0.7	0.8	0.8	0.5	0.8
TANGCRMAX	Mean	10.3	13.9	13.5	12.7	13.6	13.8	10.5	13.6
(max tail area angle change rate)	Std	0.5	1.5	1.9	1.2	1.6	1.7	1.0	1.8
TANGCRAVG	Mean	7.1	9.5	9.1	8.9	9.5	9.6	7.4	9.5
(avg tail area angle change rate)	Std	0.3	0.8	1.4	0.8	1.2	1.1	0.7	1.2
CANGCRMIN	Mean	5.9	6.7	6.4	6.9	6.9	6.9	5.7	7.3
(min center area angle change rate)	Std	0.2	0.5	0.8	0.5	0.5	0.5	0.4	0.7
CANGCRMAX	Mean	8.7	10.1	10.1	10.3	10.1	10.1	8.5	11.0
(max center area angle change rate)	Std	0.3	0.7	1.1	0.6	0.7	0.7	0.4	0.8
CANGCRAVG	Mean	7.3	8.3	8.2	8.6	8.5	8.5	7.1	9.1
(avg center area angle change rate)	Std	0.2	0.5	0.9	0.5	0.5	0.5	0.4	0.7
BANGCRMIN	Mean	8.1	9.4	9.3	9.0	9.1	8.9	7.7	9.1
(min whole body area angle change rate)	Std	0.3	0.5	0.6	0.4	0.5	0.5	0.3	0.6
BANGCRMAX	Mean	11.6	13.6	14.0	13.0	13.1	12.9	11.0	13.5
(max whole body area angle change rate)	Std	0.4	0.7	1.1	0.6	0.6	0.7	0.5	0.8
BANGCRAVG	Mean	9.7	11.4	11.4	10.9	11.0	10.8	9.2	11.2
(avg whole body area angle change rate)	Std	0.3	0.5	0.7	0.5	0.5	0.6	0.4	0.7
HDAREAMIN	Mean	975.2	777.8	677.1	826.3	884.6	914.9	925.3	943.5
(min head area)	Std	79.0	87.9	98.3	86.9	82.8	89.1	80.8	72.7
HDAREAMAX	Mean	1232	1021	876	1062	1109	1152	1163	1170
(max head area)	Std	99.4	84.6	120.2	105.4	106.5	109.2	94.2	100.2
HDAREAAVG	Mean	1098	895	775	941	994	1031	1040	1058
(avg head area)	Std	86.1	81.5	105.2	91.9	90.6	95.9	84.0	80.4
TLAREAMIN	Mean	990	762	686	785	860	880	959	868
(min tail area)	Std	96.8	70.7	81.6	73.7	75.1	72.9	82.4	69.1
TLAREAMAX	Mean	1309	997	912	1023	1169	1187	1234	1137
(max tail area)	Std	144.9	108.6	118.8	112.8	123.2	167.8	111.8	124.9
TLAREAAVG	Mean	1145	879	798	902	1011	1031	1093	1003
(avg tail area)	Std	117.6	88.2	93.5	89.3	96.0	96.9	93.4	94.9
CNAREAMIN	Mean	5600	4253	4016	4487	5090	5201	5587	4944
(min center area)	Std	423.0	383.0	560.3	387.0	393.6	463.0	517.0	358.7
CNAREAMAX	Mean	6112	4667	4343	4869	5511	5643	6041	5397
(max center area)	Std	504.9	395.1	586.4	421.9	440.5	490.2	491.4	501.8
CNAREAAVG	Mean	5844	4458	4182	4675	5298	5423	5819	5166
(avg center area)	Std	451.3	383.1	572.0	400.7	410.6	467.8	465.5	418.8

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
HDAMPMIN	Mean	-19	-15	-13	-17	-18	-18	-21	-20
(min head area amplitude)	Std	2.1	2.3	3.2	4.1	3.2	4.3	4.1	3.2
HDAMPMAX	Mean	18.8	14.8	12.7	16.5	17.1	18.8	20.2	18.8
(max head area amplitude)	Std	1.9	2.0	4.6	4.1	3.2	3.7	4.2	4.8
HDAMPAVG	Mean	-0.3	-0.2	-0.2	-0.6	-0.3	0.6	-0.4	-1.1
(avg center area amplitude)	Std	1.5	1.8	4.3	4.3	3.6	4.1	4.0	4.4
TLAMPMIN	Mean	-16	-17	-10	-17	-17	-19	-20	-18
(min tail area amplitude)	Std	2.4	1.9	4.9	4.3	4.4	4.7	4.8	4.6
TLAMPMAX	Mean	15.5	16.3	9.1	16.3	16.7	19.8	19.0	16.8
(max tail area amplitude)	Std	2.3	2.4	5.8	3.7	5.2	3.8	5.6	5.5
TLAMPAVG	Mean	-0.3	-0.3	0.0	-0.6	0.1	1.0	-0.4	-1.0
(avg tail area amplitude)	Std	1.8	2.3	5.1	4.2	5.2	5.2	5.1	5.6
CNTAMPMIN	Mean	-34	-29	-23	-36	-36	-38	-41	-43
(min center area amplitude)	Std	7.6	9.4	18.0	16.3	14.1	15.6	17.0	17.3
CNTAMPMAX	Mean	31.8	26.9	22.4	32.9	35.0	42.8	38.6	37.3
(max center area amplitude)	Std	7.9	9.4	21.3	17.6	14.3	15.0	16.8	18.1
CNTAMPAVG	Mean	-0.8	-0.8	0.1	-1.5	-0.7	2.4	-0.9	-2.9
(avg center area amplitude)	Std	5.4	6.2	19.4	13.4	13.4	14.2	13.9	16.4
AVGAMPMIN	Mean	-26	-22	-18	-28	-28	-29	-32	-33
(min of whole body amplitude average)	Std	5.6	6.8	13.2	12.1	10.4	11.8	12.5	12.8
AVGAMPMAX	Mean	25.1	20.5	17.6	26.0	27.0	32.9	30.0	28.6
(max of whole body amplitude average)	Std	5.7	6.9	15.7	13.1	10.7	11.4	12.4	13.6
AVGAMPAVG	Mean	-0.6	-0.6	0.1	-1.2	-0.5	1.9	-0.7	-2.3
(avg of whole body amplitude average)	Std	4.1	4.6	14.4	10.2	10.2	10.8	10.8	12.4
HDCNTDMIN	Mean	120.9	85.7	76.6	86.2	96.2	95.4	124.1	77.2
(min head to centroid distance)	Std	8.9	9.9	21.2	20.0	13.5	12.1	24.0	18.7
HDCNTDMAX	Mean	149.1	124.1	110.7	133.2	132.2	136.7	165.9	132.8
(max head to centroid distance)	Std	7.3	7.8	11.7	6.6	7.6	7.2	9.5	9.5
HDCNTDAVG	Mean	136.6	108.1	96.0	113.6	116.0	118.5	148.0	108.5
(avg head to centroid distance)	Std	7.1	7.5	14.6	9.5	9.1	8.4	13.4	12.5
TLCNTDMIN	Mean	122.2	79.0	85.4	90.5	88.6	81.0	120.8	82.1
(min tail to centroid distance)	Std	9.2	11.5	19.0	15.7	18.0	14.9	25.8	22.1

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
TLCNTDMAX	Mean	149.4	119.1	107.2	131.7	127.5	133.1	159.5	133.3
(max tail to centroid distance)	Std	7.1	7.7	16.4	6.7	8.7	8.7	9.8	11.1
TLCNTDAVG	Mean	137.4	103.0	97.5	114.4	110.3	110.1	143.2	110.9
(avg tail to centroid distance)	Std	6.7	8.1	16.7	8.0	11.2	10.2	14.4	13.0
HDTLANMIN	Mean	146.4	142.8	129.7	125.3	132.6	125.6	134.3	117.9
(min head to centroid vs. tail to centroid angle)	Std	7.3	10.5	29.1	24.1	15.5	12.8	21.3	24.9
HDTLANMAX	Mean	177.7	177.4	173.2	175.8	175.9	175.8	176.8	174.1
(max head to centroid vs. tail to centroid angle)	Std	0.5	0.8	13.8	4.2	3.5	1.7	1.8	5.7
HDTLANAVG	Mean	163.8	162.4	154.0	154.9	156.8	154.0	159.1	149.9
(avg head to centroid vs. tail to centroid angle)	Std	3.0	3.9	19.2	11.2	7.9	6.1	8.3	12.4
HDTLDMIN	Mean	48.9	37.9	36.4	43.4	41.3	42.9	52.0	44.0
(min head to tail distance)	Std	6.9	6.0	6.7	5.9	4.9	5.6	6.7	6.8
HDTLDMAX	Mean	278.4	221.7	198.9	239.3	234.1	238.7	298.9	232.5
(max head to tail distance)	Std	13.7	15.2	27.8	15.5	17.4	16.9	22.7	22.5
HDTLDAVG	Mean	179.8	138.8	126.9	149.8	148.1	149.0	190.3	144.2
(avg head to tail distance)	Std	10.3	11.1	19.1	11.1	12.9	11.5	18.5	15.7
HDANGMIN	Mean	-131	-131	-37	-110	-99	-120	-126	-109
(min absolute head to centroid angle)	Std	35.1	38.5	96.1	54.9	68.6	56.6	41.7	62.6
HDANGMAX	Mean	130.4	129.5	64.7	122.3	104.4	113.1	122.2	106.5
(max absolute head to centroid angle)	Std	35.1	30.2	90.7	47.7	67.3	56.4	50.4	62.2
HDANGAVG	Mean	-3.1	0.4	13.7	8.7	0.3	-1.7	-1.9	-3.0
(avg absolute head to centroid angle)	Std	29.9	30.5	79.6	41.0	58.9	49.4	42.2	52.2
TLANGMIN	Mean	-131	-132	-49	-117	-102	-115	-126	-112
(min absolute tail to centroid angle)	Std	27.9	33.0	91.4	53.8	68.6	51.5	47.9	57.0
TLANGMAX	Mean	131.7	124.9	-5.3	109.2	99.4	114.1	125.3	101.4
(max absolute tail to centroid angle)	Std	36.5	41.2	97.0	56.6	69.7	53.0	43.3	63.2
TLANGAVG	Mean	4.8	-3.2	-29	-10	-2.7	1.9	1.8	-5.8
(avg absolute tail to centroid angle)	Std	30.8	33.9	86.5	47.3	55.8	45.1	36.2	52.6
REVSALD	Mean	0.2	0.3	0.6	0.2	0.2	0.2	0.1	0.2
(total reversal distance)	Std	0.1	0.1	0.2	0.1	0.1	0.1	0.1	0.1

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
REVSALTIM	Mean	0.2	0.4	0.7	0.3	0.3	0.3	0.2	0.3
(total percentage of time stay in reversal)	Std	0.1	0.2	0.1	0.1	0.2	0.2	0.1	0.2
HDMVHFMIN	Mean	0.9	0.4	0.2	0.2	0.5	0.5	0.4	0.2
(min local head move in 0.5 sec or 1 frame)	Std	0.7	0.6	0.2	0.1	0.3	0.4	0.4	0.2
HDMVHFMAX	Mean	87.9	79.6	31.2	30.8	38.1	50.7	51.2	31.0
(max head move in 0.5 sec or 1 frame)	Std	30.1	13.6	9.7	11.0	10.4	20.4	15.0	10.4
HDMVHFAVG	Mean	23.5	23.1	7.5	7.1	10.3	12.9	11.6	7.3
(avg local head move in 0.5 sec or 1 frame)	Std	4.9	5.9	2.4	1.8	2.6	4.2	3.5	2.0
HDMV10MIN	Mean	4.5	3.5	2.1	3.7	4.4	6.4	4.8	5.1
(min local head move in 10 seconds)	Std	2.5	4.0	1.7	2.5	3.2	5.5	3.6	3.9
HDMV10MAX	Mean	220.2	185.6	77.1	123.3	119.1	144.1	161.0	111.7
(max local head move in 10 seconds)	Std	48.1	33.8	29.0	49.0	36.0	38.1	41.3	44.1
HDMV10AVG	Mean	59.9	60.9	24.5	37.0	40.0	51.8	52.1	36.6
(avg local head move in 10 seconds)	Std	13.9	16.7	7.5	9.9	10.8	16.1	12.2	9.3
HDMV20MIN	Mean	8.1	7.4	3.5	7.2	7.9	10.3	9.5	10.4
(min local head move in 20 seconds)	Std	4.7	7.2	2.5	5.0	6.1	7.6	6.7	9.3
HDMV20MAX	Mean	236.2	199.6	82.1	147.9	148.1	167.9	202.3	130.8
(max local head move in 20 seconds)	Std	42.7	36.6	31.6	47.4	41.7	47.1	50.0	43.5
HDMV20AVG	Mean	83.9	83.3	29.0	51.3	55.1	71.5	73.8	50.8
(avg local head move in 20 seconds)	Std	19.8	23.6	8.8	13.6	16.7	24.6	19.9	15.1
HDMV30MIN	Mean	12.1	13.0	5.1	10.8	11.6	17.2	15.2	13.7
(min local head move in 30 seconds)	Std	7.7	14.7	3.4	7.1	8.7	13.1	11.0	9.7
HDMV30MAX	Mean	244.5	209.0	84.9	171.4	164.4	185.0	223.9	137.7
(max local head move in 30 seconds)	Std	41.0	35.1	33.4	46.4	44.4	46.3	50.9	51.2
HDMV30AVG	Mean	101.4	99.1	32.2	64.4	66.9	85.9	91.9	59.7
(avg local head move in 30 seconds)	Std	24.8	29.4	9.7	19.2	21.1	28.7	24.9	20.7
HDMV40MIN	Mean	18.8	16.7	6.4	15.6	14.2	26.9	18.8	20.6
(min local head move in 40 seconds)	Std	14.6	19.4	3.7	11.2	13.1	23.4	16.1	17.8
HDMV40MAX	Mean	253.4	208.8	84.3	177.7	170.6	192.6	229.5	147.2
(max local head move in 40 seconds)	Std	39.0	34.1	34.3	48.8	47.4	45.7	52.0	49.7
HDMV40AVG	Mean	118.4	107.7	33.6	74.4	75.2	99.1	107.6	71.2
(avg local head move in 40 seconds)	Std	27.4	31.4	11.7	21.6	24.7	33.6	31.4	25.2
HDMV50MIN	Mean	24.1	25.2	7.6	21.0	21.1	28.8	26.6	25.0
(min local head move in 50 seconds)	Std	19.2	20.9	4.6	16.8	15.0	23.7	27.0	20.3

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
HDMV50MAX	Mean	254.2	206.8	84.7	182.1	174.3	197.7	239.0	160.8
(max local head move in 50 seconds)	Std	36.4	36.1	34.5	47.8	49.9	49.9	55.5	52.7
HDMV50AVG	Mean	128.5	116.5	36.5	84.1	84.0	105.0	117.9	79.8
(avg local head move in 50 seconds)	Std	31.3	32.8	12.2	26.0	29.7	32.7	37.9	30.1
TLMVHFMIN	Mean	0.4	0.3	0.0	0.2	0.1	0.2	0.2	0.1
(min local tail move in 0.5 sec or 1 frame)	Std	0.3	0.4	0.0	0.1	0.1	0.2	0.2	0.1
TLMVHFMAX	Mean	79.3	81.0	18.8	30.8	31.8	40.1	45.7	25.5
(max tail move in 0.5 sec or 1 frame)	Std	25.5	11.0	13.0	9.8	9.1	14.8	14.4	10.5
TLMVHFAVG	Mean	14.9	21.2	3.0	6.3	6.6	9.1	9.1	4.7
(avg local tail move in 0.5 sec or 1 frame)	Std	3.6	5.8	1.0	1.6	2.3	3.7	3.1	1.4
TLMV10MIN	Mean	3.4	3.4	0.8	2.4	2.9	5.3	3.5	2.8
(min local tail move in 10 seconds)	Std	1.8	3.8	0.6	1.6	2.6	4.6	2.4	2.3
TLMV10MAX	Mean	218.1	178.5	37.6	129.1	110.8	131.4	160.9	115.9
(max local tail move in 10 seconds)	Std	53.9	31.5	24.3	46.6	38.4	38.9	44.2	44.7
TLMV10AVG	Mean	53.1	57.8	9.9	33.5	32.6	44.9	46.6	29.2
(avg local tail move in 10 seconds)	Std	14.4	17.0	3.6	8.0	12.1	15.7	13.5	11.1
TLMV20MIN	Mean	6.5	7.4	1.6	6.1	5.7	11.9	8.6	6.1
(min local tail move in 20 seconds)	Std	4.5	7.9	1.4	4.3	4.4	11.3	6.6	5.3
TLMV20MAX	Mean	234.7	194.7	40.7	150.7	132.1	157.9	187.7	133.3
(max local tail move in 20 seconds)	Std	48.1	33.8	27.3	45.3	42.2	43.6	47.2	43.6
TLMV20AVG	Mean	78.9	78.8	12.4	48.4	46.4	65.3	68.6	44.4
(avg local tail move in 20 seconds)	Std	21.0	24.9	4.5	13.1	18.0	25.2	20.8	16.8
TLMV30MIN	Mean	10.9	13.8	2.2	9.6	8.6	17.5	11.5	9.2
(min local tail move in 30 seconds)	Std	9.7	17.3	1.5	7.4	6.4	16.3	9.3	8.6
TLMV30MAX	Mean	243.9	201.6	41.3	172.1	146.5	171.9	209.7	144.6
(max local tail move in 30 seconds)	Std	42.5	34.1	30.6	46.4	45.4	45.2	56.2	46.8
TLMV30AVG	Mean	98.2	94.0	13.4	62.2	57.1	78.0	85.8	56.2
(avg local tail move in 30 seconds)	Std	26.5	29.2	5.1	18.8	21.5	30.0	26.0	21.8
TLMV40MIN	Mean	16.8	18.3	2.7	12.9	13.0	25.1	19.3	17.2
(min local tail move in 40 seconds)	Std	10.7	20.5	1.8	12.0	11.8	23.1	16.2	18.4
TLMV40MAX	Mean	253.5	201.6	42.7	181.3	159.3	177.4	219.8	153.8
(max local tail move in 40 seconds)	Std	39.9	33.4	31.2	44.8	47.2	51.2	56.5	49.5
TLMV40AVG	Mean	115.0	103.5	14.8	74.0	67.2	90.2	100.4	66.8
(avg local tail move in 40 seconds)	Std	28.3	31.6	6.2	23.8	24.9	34.5	30.4	27.3

Continued on next page

Table A.1 – continued from previous page

Variable	Stat.	wild	*goa1*	*nic1*	*uc36*	*uc38*	*uc29*	*eg19*	*uc2*
TLMV50MIN	Mean	24.7	24.8	3.7	17.8	16.4	30.9	27.6	25.4
(min local tail move in 50 seconds)	Std	18.8	24.3	2.5	16.0	11.6	21.3	24.7	24.1
TLMV50MAX	Mean	256.3	199.9	42.2	184.6	159.6	179.5	226.4	147.4
(max local tail move in 50 seconds)	Std	38.2	33.1	33.4	44.2	48.0	52.4	52.2	49.4
TLMV50AVG	Mean	126.3	110.1	16.0	82.6	74.3	96.8	111.9	75.5
(avg local tail move in 50 seconds)	Std	31.8	32.8	7.4	25.4	26.1	34.1	37.3	32.6
HTMVRMIN	Mean	0.1	0.1	0.2	0.1	0.1	0.1	0.1	0.2
(min total local head move/tail movement)	Std	0.1	0.0	0.1	0.0	0.0	0.1	0.0	0.0
HTMVRMAX	Mean	23.4	18.9	14.6	11.1	14.2	14.9	13.0	12.8
(max total local head move/tail movement)	Std	6.6	7.7	4.2	4.0	3.1	4.3	3.9	4.0
HTMVRAVG	Mean	2.5	1.6	2.8	1.5	2.2	2.0	1.7	2.0
(avg total local head move/tail movement)	Std	0.3	0.2	0.4	0.2	0.4	0.3	0.2	0.3
HDHFTOTMV	Mean	13219	12844	4326	3389	4676	5148	6049	2756
(total local head move)	Std	2845	3295	1525	837	1170	1607	1631	850
TLHFTOTMV	Mean	8379	11730	1738	3024	2971	3652	4747	1752
(total local tail move)	Std	1931	3174	621	817	1016	1349	1338	556

Bibliography

[1] J. Baek, P. Cosman, Z. Feng, J. Silver, and W. R. Schafer. Using machine vision to analyze and classify *C. elegans* behavioral phenotypes quantitatively. *Journal Neurosci. Meth.*, vol. 118, pp. 9-21, 2002.

[2] R. Bainton, L. Tsai, C. Singh, M. Moore, W. Neckameyer, and U. Heberlein. Dopamine modulates acute responses to cocaine, nicotine and ethanol in Drosophila. *Current Biology*, vol. 10, pp. 187–194, 2000.

[3] C. A.. Bastiani, C.A., Gharib, S., Simon, M. I., P.W. Sternberg. *Caenorhabditis elegans* Gq regulates egg-laying behavior via a PLC-independent and serotonin-dependent signaling pathway and likely functions both in the nervous system and in muscle. *Genetics*, vol. 165, pp. 1805–1822, 2003.

[4] C Bishop. *Neural Networks for Pattern Recognition*, Oxford University Press, 1996.

[5] L. Breiman. Random forests. *Machine Learning*, vol. 45(1), pp. 5–32, 2001.

[6] L. Breiman. Manual on setting up, using and understanding random forests v3.1. *online material*,2002.

[7] L. Breiman. Bagging predictors. *Machine Learning*, vol. 24(2), pp. 123–140, 1996.

[8] L. Breiman. Out-of-bag estimation. *online material*, ftp.stat.berkeley.edu/pub/users/breiman/OOBestimation.ps, 1996.

[9] L. Breiman, J. Friedman, R. Olshen, C. Stone. *Classification and Regression Trees*, Wadsworth, Belmont, CA, 1984.

[10] S. Brenner. The genetics of *Caenorhabditis elegans*. *Genetics*, vol. 77, pp. 77–94, 1974.

[11] P. J. Brockie, J. E. Mellem, T. Hills, D. M. Madsen, and A. V. Maricq. The *C. elegans* glutamate receptor subunit NMR-1 is required for slow NMDA-activated currents that regulate reversal frequency during locomotion. *Neron*, vol. 31, pp. 617–630, 2001.

[12] L. Brundage, L. Avery, A. Katz, U.-J. Kim, J. E. Mendel, P. W. Sternberg, and M. I. Simon. Mutations in a *C. elegans* Gqa gene disrupt movement, egg laying and viability. *Neuron*, vol. 16, pp. 999–1009, 1996.

[13] C. J. Burges A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, vol. 2(2), pp. 121–167, 1998.

[14] T. Bylander. Estimating generalization error on two-class datasets using out-of-bag estimates. *Machine Learning*, vol. 48, pp. 287–297, 2002.

[15] R. K. M. Choy and J. H. Thomas. Fluoxetine resistant mutants in *C. elegans* define a novel family of transmembrane proteins. *Molecular Cell*, vol. 4, pp. 143–152, 1999.

[16] R. Dhawan, D. B. Dusenbery, and P. L.Williams. Comparison of lethality, reproduction, and behavior as toxicological endpoints in the nematode *Caenorhabditis elegans*. *J. Toxicology and Environmental Health*, Part A, vol. 58, pp. 451–462, 1999.

[17] G. Borgefors. Distance Transformations in Digital Images. *Academic Press*, pp. 344–371, 1986.

[18] M. de Bono, C. I. Bargmann . Natural Variation in a Neuropeptide Y Receptor Homolog Modifies. Social Behavior and Food Response in *C. elegans*. *Cell*, vol. 94, pp. 679–689, 1998.

[19] A. Dempster, N.Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm (with discussion). *J.R.Statist. Soc. B*, vol. 39, pp. 1–38, 1977.

[20] R. Duda, P. Hart, and D. Stork. *Pattern Classification*, Second Edition, Wiley, New York, Wiley, 2002.

[21] H. M. Ellis and H. R. Horvitz. Genetic control of programmed cell death in the nematode *C. elegans*. *Cell*, vol. 44, pp. 817–829, 1986.

[22] F. Escolano, M. Cazorla, D. Gallardo, and R. Rizo. Deformable Templates for Tracking and Analysis of Intravascular Ultrasound Sequences. *Proc. Of EMM-CVPR97*, vol. 1223, 1997.

[23] R. Fisker, J.M. Carstensen, M.F. Hansen, F. Bodeker, S. Morup. Estimation of Nanoparticle Size Distributions by Image Analysis. *Journal of Nanoparticle Research*, vol. 2(3), pp. 267–277, 2000.

[24] R. Fisker. *Making Deformable Template Models Operational*, PhD Thesis, Technical University of Denmark, 2000.

[25] J. T. Fleming, M. D. Squire, T. M. Barnes, C. Tornoe, K. Matsuda, J. Ahnn, A. Fire, J. E. Sulston, E. A. Barnard, D. B. Sattelle, and J. A. Lewis. *Caenorhabditis elegans* Levamisole Resistance Genes *lev-1*, *unc-29*, and *unc-38* Encode Functional Nicotinic Acetylcholine Receptor Subunits. *J. Neurosci.*, vol. 17, pp. 5843–5857, 1997.

[26] A. Fraser, et al. Functional genomic analysis of *C. elegans* chromosome I by systematic RNA interference. *Nature*, vol. 408, pp. 325–330, 2001.

[27] W. Geng, P. Cosman, C. Huang, and W. R. Schafer. Automated Worm Tracking and Classification. *Proc. of the 37th IEEE Asilomar Conference on Signals, Systems and Computers*, pp. 2063-2068, Pacific Grove, CA, November 2003.

[28] W. Geng, P. Cosman, J-H Baek, C. Berry, and W.R. Schafer. Quantitative Classification and Natural Clustering of *C. elegans* Behavioral Phenotypes. *Genetics*, vol. 165, pp. 1117–1136, 2003.

[29] W. Geng, P. Cosman, J.-H. Baek, C. Berry and W.R. Schafer. Feature Extraction and Natural Clustering of Worm Body Shapes and Motion Characteristics. *IASTED International Conference on Signal and Image Processing (SIP 2003)*, August 13–15, 2003, Honolulu, Hawai.

[30] W. Geng, P. Cosman, C. Berry, Z. Feng and W.R. Schafer. Automatic Tracking, Feature Extraction and Classification of *C. elegans* Phenotypes. *IEEE Transactions on Biomedical Engineering*, in press, 2004.

[31] W. Geng, P. Cosman, and W.R. Schafer. Egg Onset Detection Using Deformable Template Matching. to appear in *IASTED International Conference on Computer Graphics and Image Processing(CGIM2004)*, Kauai, Hawaii, August 2004.

[32] W. Geng, P. Cosman, M. Palm, and W.R. Schafer. *C. elegans* Egg-laying Detection and Behavior Study Using Image Analysis. *Submitted to EUROSIP Journal on Applied Signal Processing.*

[33] R. Gonzalez and R. Woods. *Digital Image Processing*, Second Edition, Prentice Hall, New Jersey, 2002.

[34] D. Grossman. Short Course in Data Warehousing and Data Mining. *online material*, http://www.ir.iit.edu/ dagr/DataMiningCourse/Spring2001/Notes/Data_Preprocessing.pdf

[35] L. A. Hardaker, L. A. Singer, E. Kerr, G. T. Zhou, and W. R. Schafer. Serotonin modulates locomotory behavior and coordinates egg laying and movement in Caenorhabdits elegans. *Journal of Neurobiology* , vol. 49, pp. 303–313, 2001.

[36] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*, Springer, New York,2002.

[37] H.R. Horvitz. Genetic control of programmed cell death in the nematode *Caenorhabditis elegans*. *Cancer Research* vol. 59, pp. 1701–1706, 1999.

[38] J. Hodgkin. Male phenotypes and mating efficiency in *Caenorhabditis elegans*. *Genetics*, vol. 103, pp. 43–64, 1983.

[39] A. Hyvarinen and E. Oja. Independent component analysis: Algorithms and applications. *Neural Computing*, vol. 3, pp. 411–430, 2000.

[40] R. Jain, K. Rangachar, and B. Schunck. *Machine Vision*, New York, McGraw-Hill, 1995.

[41] R. Jain, Y. Zhong, and M.P. Dubuisson-Jolly. Deofrmable Template Models: A Review. *Signal Processing*, vol. 71, pp. 109–129, 1998.

[42] R. Jain, Y. Zhong, and S. Lakshmanan. Object Matching Using Deformable Templates. *Signal Processing*, vol. 18, No. 3, pp. 267–277, 1996.

[43] V .M. Kanti, W. Qian, D. Shah, and K. de Souza. Deformable Template Recognition of Multiple Occluded Objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, pp. 1035–1042, 1997.

[44] R. Kohavi and G. John. Wrappers for feature subset selection. *Artificial Intelligence Journal, special issue on relevance*, vol. 97, pp. 273–324, 1997.

[45] R. Johnson and D.W. Wichern. *Applied Multivariate Statistical Analysis*, Prentice-Hall, New Jersey,2002.

[46] L. Kaufman and P. Rousseeuw. *Finding Groups in Data: An Introduction to Clustering Analysis*, Wiley, New York, 1990.

[47] S. Kim, D. Poole, L. Waggoner, A. Kempf, D. Ramirez, A. Treschow, W. R. Schafer. Genes affecting the activity of nicotinic receptors involved in *C. elegans* egg-laying behavior. *Genetics*, vol. 157, pp. 1599–1610, 2001.

[48] T. Kohonen. *The self-organizing map*, Springer-Verlag, Berlin, 1990.

[49] R. j. Larsen and M. L. Marx *An Introduction to Mathematical Statistics and Its Applications*, Third Edition, Prentice-Hall, New Jersey, 2001.

[50] R. Y. N. Lee, L. Lobel, M. Hengartner, H. R. Horvitz, and L. Avery. Mutations in the alpha1 subunit of an L-type voltage-activated Ca2+ channel cause myotonia in *Caenorhabditis elegans*. *EMBO J.* vol. 16, pp. 6066–6076, 1997.

[51] R. C. Lee and V. Ambros. An Extensive Class of Small RNAs in *Caenorhabditis elegans*. *Science* vol. 294, pp. 862–854, 2001.

[52] C. Leung-Hagesteijn, A. M. Spence, B. D. Stern, Y. Zhou, M. W. Su, E. M. Hedgecock, and J. G. Culotti. *unc-5*, a transmembrane protein with immunoglobulin and thrombospondin type 1 domains, guides cell and pioneer axon migrations in *C. elegans*. *Cell* vol. 77, pp. 289–299, 1992.

[53] C. M. Loer and C. J. Kenyon. Serotonin-deficient Mutants and Male Mating Behavior in the Nematode *Caenorhabditis elegans*. *J. Neurosci.*, vol. 13, pp. 5407–5417, 1993.

[54] D. Levitan and I. Greenwald. Facilitation of lin-12-mediated signalling by sel-12, a *Caenorhabditis elegans* S182 Alzheimer's disease gene. *Nature*, vol. 377, pp. 351–254, 1995.

[55] J. A. Lewis, C. H. Wu, J. H. Levine, H. Berg, Levamisoleresistant mutants of the nematode *Caenorhabditis elegans* appear to lack pharmacological acetylcholine receptors. *Neuroscience*, vol. 5, pp. 967-989, 1980.

[56] A. Liaw and Wiener. Classification and Regression by randomForest. *R newsletter*, Vol. 2/3. http://www.r-project.org, 2002.

[57] S. P. Lloyd. Least Squares Quantization in PCM. *Institute of Mathematical Statistics Meeting*, September, 1957.

[58] O. Mangasarian and R. Musicant. Lagrangian Support Vector Machines. *J. Machine Learning Research*, vol. 1, pp. 161–177, 2001.

[59] J. E. Mendel, H. C. Korswagen, K. S. Liu, Y. M. Hadju-Cronin, M. I. Simon, R. H. Plasterk, and P. W. Sternberg. Participation of the protein Go in multiple aspects of behavior in *C. elegans*. *Science*, vol. 267, pp. 1652–1655, 1995.

[60] C. McClung, and J. Hirsh. Stereotypic behavioral responses to free-base cocaine and the development of behavioral sensitization in Drosophila. *Current Biology*, vol. 8, pp. 109–112, 1998.

[61] T. McInernery and D. Terzopoulos. Deformable Models in Medical Image Analysis: A Survey. *Medical Image Analysis*, vol. 1(2), pp. 91–108, 1996.

[62] C.E. Metz. Basic Principles of ROC Analysis. *Seminars in Nuclear Medicine*, vol. VIII, No. 4, pp. 283–298, 1978.

[63] T. Mitchell. *Machine Learning*, McGraw-Hill, New York,1997.

[64] F. Model, P. Adorjan, A. Olek, and C. Piepenbrock. Feature Selection for DNA methylation based cancer classification. *Bioinformatics*, vol. 17, Suppl. S1, pp. 57–64, 2001.

[65] L. S. Nelson, M. L. Rosoff, and C. Li. Disruption of a neuropeptide gene, *flp-1*, causes multiple behavioral defects in *C. elegans*. *Science*, vol. 281, pp. 1686–1690, 1998.

[66] M. L. Nonet, K. Grundahl, B. J. Meyer, and J. B. Rand. Synaptic function is impaired but not eliminated in *C. elegans* mutants lacking synaptotagmin. *Cell*, vol. 73, pp. 1291–1305, 1993.

[67] S. Nurrish, L. Segalat, and J. .M. Kaplan, 1999. Serotonin inhibition of synaptic transmission: Gao decreases the abundance of UNC-13 at release sites. UNC-13 at release sites. *Neuron*, vol. 24, pp. 231-242, 1999.

[68] J. Pierce-Shimomura, T. Morse, and S. Lockery. The fundamental role of pirouettes in *C. elegans* chemotaxis. *Journal of Neuroscience*, vol. 19(21), pp. 9557–9569, 1999.

[69] The R Project for Statistical Computing. *http://www.r-project.org*.

[70] D. J. Reiner, E. M. Newton, H. Tian, and J. .H. Thomas, Diverse behavioural defects caused by mutations in *Caenorhabditis elegans unc-43* CaM kinase II. *Nature*, vol. 402, pp. 199203, 1999.

[71] D. L. Riddle, T. Blumenthal, B. J. Meyer, and J. R. Priess. *C. elegans II*, Cold Spring Harbor Laboratory Press: Cold Spring Harbor, NY, 1997.

[72] B. D. Riply. *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, 1996.

[73] S. Suparna, R. F. Wintle1, K. Katie, W. M. Nuttley, R. Arvan1, P. Fitzmaurice1, E. Bigras, D. C. Merz, T. E. Hbert, D. Kooy, W. R. Schafer, J. Culotti, and H. Van Tol Dopamine modulates the plasticity of mechanosensory responses in *Caenorhabditis elegans*. *EMBO Journal*, vol. 23, No. 2, pp. 473–482, 2004.

[74] L. Segalat, D. A. Elkes, and J. M. Kaplan. Modulation of serotonin-controlled behaviors by Go in *Caenorhabditis elegans*. *Science*, vol. 267, pp. 1648–1651, 1995.

[75] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*, Second Edition, Brooks/Cole Publishing Company, 1999.

[76] W. R. Schafer and C. J. Kenyon. A calcium channel homologue required for adaptation to dopamine and serotonin in Cenorhabditis elegans. *Nature*, vol. 375, pp. 73-78, 1995.

[77] C. Sugar. Techniques for Clustering and Classification with Applications to Medical Problems. *Stanford University PhD Thesis*, 1998.

[78] C. Sugar and G. James. Finding the number of clusters in a data set: An information theoretic approach *J. American Statistical Assoc*, vol. 98, pp. 750–763, 2003.

[79] J. Sulston, M. Dew, and S. Brenner. Dopaminergic neurons in the nematode Caenarhabditis elegans. *J. Comp. Neur.*, vol. 163, pp. 215–226, 1975.

[80] J. Sulston and H. Horvitz. Post-embryonic cell lineages of the nematode *Caenorhabditis elegans*. *Developmental Biology*, vol. 56, pp. 110–156, 1977.

[81] J. Sulston, E. Schierenberg, J. White, and J. Thomson. The embryonic cell lineage of the nematode *Caenorhabditis elegans*. *Developmental Biology*, vol. 100, pp. 64–119, 1983.

[82] B. Van Swinderen, O. Saifee, L. Shebester, R. Roberson, M. L. Nonet, and C. M. Crowder. A neomorphic syntaxin mutation blocks volatile-anesthetic action in *Caenorhabditis elegans*. *Proc. Natl. Acad. Sci. USA*, vol. 96, pp. 2479–2484, 1999.

[83] J. Y. Sze, M. Victor, C. Loer, Y. Shi, and G. Ruvkun, References Food and metabolic signalling defects in a *Caenorhabditis elegans* serotonin-synthesis mutant. *Nature*, vol. 403, pp. 560-564, 2000.

[84] R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a dataset via the Gap statistic. *J. Royal Statistical Society Series B*, vol. 63, pp. 411–423, 2001.

[85] C. Trent, N. Tsung, and H. R. Horvitz. Egg-laying defective mutants of the nematode *Caenorhabditis elegans*. *Genetics*, vol. 104, pp. 619–647, 1983.

[86] V. Vapnik *Statistical Learning Theory*, Wiley, New York, 1998.

[87] W. N. Venables and B. D. Ripley. *Modern Applied Statistics With S*, Fourth Edition, Springer, 2002.

[88] L. E. Waggoner, G. T. Zhou, R. W. Schafer, W. R. Schafer. Control of alternative behavioral states by serotonin in *Caenorhabditis elegans*. *Neuron* vol. 21, pp. 203–214, 1998.

[89] L. E. Waggoner, K. A. Dickinson, D. S. Poole, Y. Tabuse, J. Miwa, and W. R. Schafer. Long-term nicotine adaptation in *Caenorhabditis elegans* involves. PKC-dependent changes in nicotine receptor abundance. *J. Neurosci.*, vol. 20, pp. 8802–8811, 2000.

[90] D. Weinshenker, G. Garriga, and J.H. Thomas. Genetic and pharmacological analysis of neurotransmitters controlling egg-laying in *C. elegans*. *J. Neurosci.* vol. 15, pp. 6975–6985, 1995.

[91] D. Weinshenker, A. Wei, L. Salkoff, and J. H. Thomas. An eag K+ channel links cell excitation and antidepressant action in *C. elegans*. *J. Neurosci.*, vol. 19, pp. 9831–9840, 1999.

[92] J. E. White, E. Southgate, N. Thomson, and S. Brenner. The structure of the *Caenorhabditis elegans* nervous system. *Philos. Trans. R. Soc. Lond. (Biol.)* vol. 314, pp. 1–340, 1986.

[93] I. Witten and E. Frank. *Data Mining*, Morgan Kaufmann, 2000.

[94] T. Y. Zhang and C.Y. Suen. A Fast Parallel Algorithm for Thinning Digital Patterns. *Comm. ACM*, vol. 27, no.3, pp. 236–239, 1984.

[95] G. T. Zhou, W. R. Schafer, R. W. Schafer. A three-state biological point process model and its parameter estimation. *IEEE Trans. Signal Processing*, vol. 46, pp. 2698–2707, 1998.

[96] P. Zipperlen, A. Fraser, R. Kamath, M. Martinez-Campos and J. Ahringer. Roles for 147 embryonic lethal genes on *C. elegans* chromosome I identified by RNA interference and video microscopy. *EMBO J.*, vol. 20, pp. 3984–3992, 2001.

[91] D. Weinshenker, A. Wei, L. Salkoff, and J. H. Thomas. An eag K+ channel links cell excitation and antidepressant action in *C. elegans*. *J. Neurosci.*, vol. 19, pp. 9831–9840, 1999.

[92] J. E. White, E. Southgate, N. Thomson, and S. Brenner. The structure of the *Caenorhabditis elegans* nervous system. *Philos. Trans. R. Soc. Lond. (Biol.)* vol. 314, pp. 1–340, 1986.

[93] I. Witten and E. Frank. *Data Mining*, Morgan Kaufmann, 2000.

[94] T. Y. Zhang and C.Y. Suen. A Fast Parallel Algorithm for Thinning Digital Patterns. *Comm. ACM*, vol. 27, no.3, pp. 236–239, 1984.

[95] G. T. Zhou, W. R. Schafer, R. W. Schafer. A three-state biological point process model and its parameter estimation. *IEEE Trans. Signal Processing*, vol. 46, pp. 2698–2707, 1998.

[96] P. Zipperlen, A. Fraser, R. Kamath, M. Martinez-Campos and J. Ahringer. Roles for 147 embryonic lethal genes on *C. elegans* chromosome I identified by RNA interference and video microscopy. *EMBO J.*, vol. 20, pp. 3984–3992, 2001.