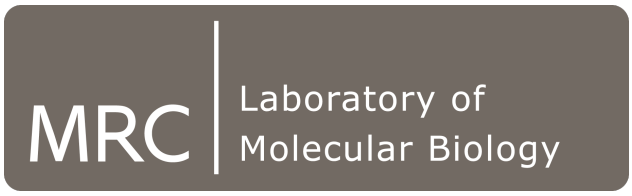
High-throughput, single-worm tracking and analysis in *Caenorhabditis elegans*



This dissertation is submitted for the degree of Doctor of Philosophy

Eviatar Yemini

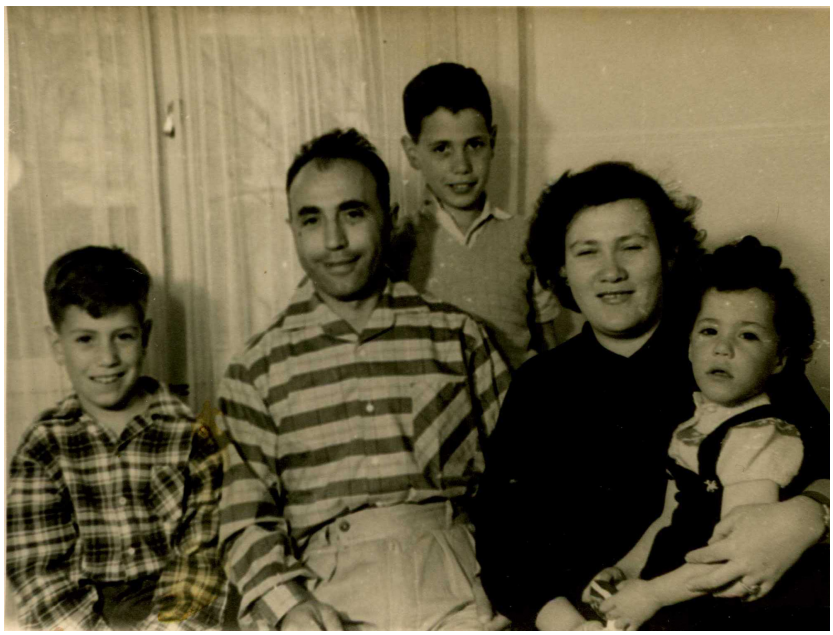
Emmanuel College, Cambridge

MRC Laboratory of Molecular Biology

November 2011

This work is dedicated to my grandmother Rachel Alexander and in loving memory of my grandparents Zvi Alexander, Rifka Yemini, and Israel Yemini.

Their unconditional love and support, as well as that of my family and close friends, are the things I treasure most and hold dearest among all the blessings I have enjoyed.



Preface

This dissertation is the result of my own work and includes nothing which is the outcome of work done in collaboration except where specifically indicated in the text and Acknowledgements.

Eviatar Yemini

Summary

Caenorhabditis elegans, a millimeter-sized, soil-dwelling nematode, is a model organism for biology research. Its whole genome has been sequenced. The lineage and fate, for each one of the cells in wild-type (N2) worms, is known. The connectivity, for all 302 neurons of wild-type hermaphrodites, has been mapped. Many of its genes have homologs within other organisms, including humans. *C. elegans* have a well-defined repertoire of observed behaviors. For these reasons, and due to a wealth of experimental data, *C. elegans* is a well-suited organism for mapping genetics to phenotype. This thesis details a system for relating genetics to phenotype. I present a methodology for semi-automated, high-throughput, high-resolution investigation of gene effects on behavior and morphology using *C. elegans*.

In the first section beyond the introduction, Chapter 2, I describe a new single-worm tracking system (hardware and software), titled Worm Tracker 2.0 (WT2), which was used to collect videos of worm behavior with high throughput. While multi-worm tracking systems exist, including ones that enable higher experimental throughput by recording multiple worms at once, their videos have insufficient resolution to resolve worm bodies well and these systems have been limited to only simple measurements. While other single-worm tracking systems also exist, they present, among other limitations, significant costs precluding high experimental throughput. I designed and built the hardware and software for a less expensive unit, which is approximately 1/4 the cost of previous single-worm trackers. This enabled us to purchase eight such units for high-throughput of experimentation. Other novelty for our system includes the ability to track worms at all larval stages and the ability to follow single-worms swimming.

In Chapter 3, I describe a novel automated analysis for the worm videos collected using the aforementioned single-worm tracker. While analysis exists for other single-worm tracking systems, several limitations precluded adaptation. Our worm videos are on food and the worms are of variable size. Several previous algorithms attempted to deal with worms on food but, for our purposes, suffer from

poor resolution at the head and tail, areas necessary to obtain significant phenotypic information. The analysis I built uses a novel algorithm driven by a need to obtain high-accuracy and precise worm contours (and their consequent skeletons) in our difficult conditions (e.g., on food and swimming environments) with invariance to worm size (bounded by a minimal limit of resolution). This accuracy was necessary due to the sheer size of the data set collected, roughly 1/3 of a billion frames, which precludes manual verification.

In the final section, Chapter 4, I describe the results from my analysis of our collected data. Using our trackers we collected more than 12,000 videos, each 15 minutes in length, at 640x480 20-30Hz resolution, representing over 300 mutant strains matched to wild-type controls. This large set was filtered to obtain high-quality data and remove strains specific to private data sets (prepared for future publications). The filtered analysis covers 330 worm groups compiled from 300 mutant strains, 2 wild isolates, three descendants of N2, along with our N2 controls divided into hourly, daily, and monthly groups. A subset of 79 strains, representing 76 genes with no previously characterized phenotype, show significant measures in my analysis. Further sensitivity of the analysis is explored through measures of habituation, small morphological changes due to growth, and a phenotypic comparison of the three descendants from the ancestral, wild-type N2. With the sensitivity explored, I present an N2 phenotypic reference compiled from 1,218 worms, recorded over three years. Statistics of this set define a reference measure of the N2 phenotype (specific to the Schafer Lab wild type) with broad implications for performing and controlling *C. elegans* experiments. Three genes, implicated in mechanosensation as a result of genetic sequence but lacking any observed phenotypic support, reveal locomotory phenotypes in our analysis. This prompts a large clustering of all 330 groups, to assess the predictive capabilities of our system. The N2 groups cluster together in a large exclusive aggregate. Further support for the predictive capabilities of the clustering emerge among multiple published pathways that also form exclusive clusters. I end by discussing a set of genes, predicted to be acetylcholine receptors through genetic sequence and functional heterologous expression, which now receive further support through strong aggregation within their own exclusive phenotypic cluster.

Acknowledgements

This thesis would not have been possible without the support and guidance of my supervisor, Bill Schafer. His mentoring gave me the freedom to try my own ideas, understand their strengths and weaknesses, and when I felt lost, he led me back on course to complete the work herein. There is so much more I would love to mention, regarding my thanks for his instruction, but there is insufficient space and I would not even know where to begin. I am forever indebted for my experience with Bill.

I knew how lucky I was to have Sarah Teichmann as my second supervisor when my father mentioned he was enthralled by her publications and asked me if I knew of her. Sarah always had time to help me with my questions, she personally introduced me to several computer vision professors instrumental to my work, and my great regret is that I should have spent much more time learning from her insight.

Laura Grundy has tirelessly spent the last three years recording over 12,000 videos using the worm tracking hardware and software I built, all through gradual, incremental refinements of the design. An exhausting feat I cannot even imagine performing and for which I am forever in her debt. Additional thanks go to James Cregg and Victoria Butler who were also instrumental in providing feedback on early versions of the worm tracking hardware and software.

Andre Brown has been an invaluable mentor. Among other general help, his initial work facilitated the stage movement detection and head detection mentioned in Chapter 3 as well as the clustering and several features mentioned in Chapter 4. Well beyond this, he has been a constant source of enthusiastic support in my work and scientific career.

Chris Cronin happily provided insights into programming Ludl-brand stages and the initial elements to create a graphical user interface (GUI) for the analysis mentioned in Chapter 3. Tadas Jucikas undertook the work necessary to adapt Chris's GUI to our own functions. Furthermore, Tadas created web pages that provide access

to the WT2 software and the collective results from my analysis, described in Chapter 4. Apart from the work, Tadas heartily organized many unforgettable holidays and social events that have tempered my research with a great deal of fun. James Scott-Brown is a fantastic summer student who built a tool (detailed in Chapter 3) to correct automated head-tail assignments in our worm videos and remove any poorly segmented video frames, all prior to phenotypic assessment.

Richard Samworth, of the Mathematical Statistics Department, answered all of my statistics questions diligently. Joan Lasenby and Nick Kingsbury, of the Signal Processing Group, gave me early guidance with computer vision. Andrew Deonarine adapted my clustering to R wherein we could produce vector graphics along with a statistical bootstrap. Guilhem Chalancon and Sreenivas Chavali imparted considerable understanding of the statistical methods available for analyzing big data sets such as our own.

Regarding my mother, father, and sister, I could fill several volumes on their personal support during my graduate work. I would especially like to thank my father for taking some of the most complex mathematics and simplifying them to provide an intuition for difficult computational techniques. Despite all the literature I read on principal component analysis, clustering, linear algebra, functional analysis, and other such methods, I never understood the concepts as well as when my father patiently took the time to explain them to me.

Finally, the Gates Trust and the Medical Research Council have generously funded my journey here.

Abbreviations

AChR	Acetylcholine Receptor
CDF	Cumulative Distribution Function
CGC	Caenorhabditis Genetics Center
FDR	False-Discovery Rate
GUI	Graphical User Interface
LDA	Linear Discriminant Analysis
MER	Minimum Enclosing Rectangle
MWT	Multi-Worm Tracker
NBRP	National BioResource Project
NGM	Nematode Growth Medium
PCA	Principal Component Analysis
PDF	Probability Density Function
RGS	Regulator of G-protein Signaling
RNAi	RNA Interference
SEM	Standard Error of the Mean
SWT	Single-Worm Tracker
TRP	Transient Receptor Potential channel
WT1	Worm Tracker 1.0
WT2	Worm Tracker 2.0

Content

List of Figures and Tables

Chapter 1 : Introduction

1.1 *Caenorhabditis elegans*

Caenorhabditis elegans is a soil-dwelling nematode that nowadays, often inhabits laboratories as a model biological organism. Progeny of an isolate from Bristol, labeled N2, provide the exemplary wild-type control [1]. *C. elegans*, for the most part, are hermaphrodite with low incidence of male births (approximately 0.2%, although the number can be varied through several manipulations in maternal environment and breeding conditions) [2]. Hermaphroditism is the result of two X chromosomes, whereas males only have one sex chromosome. Self-fertilizing hermaphrodites provide stability in the genetic lineage although 2.1×10^{-8} mutations (per site, per generation) have been measured to occur [3]. Even the N2 controls, cultivated by many labs, are thought to have diverged slightly from their genetic ancestry. An adult N2 usually provides a brood of 330 offspring and this, along with general hardiness to environmental conditions, makes worm cultivation within a laboratory setting quite easy. *C. elegans* hatch from their egg and undergo four molts to reach adulthood within approximately 3 days [4]. First larval stage N2 worms measure approximately 250 microns and adults reach just over 1 millimeter. The N2 life span lasts nearly 2-3 weeks depending on environmental factors [5]. Within the laboratory, *C. elegans* are often reared on Nematode Growth Medium (NGM) agar plates with a lawn of OP50 strain *Escherichia coli* providing their food source.

Ease of cultivation as well as a wealth of experimental techniques and publications have transformed *C. elegans* into a popular organism for multiple avenues of biology research. Moreover, several significant discoveries, methods, and repositories of information have established this worm as a model organism. The genome is completely sequenced [6]. Genetic manipulation can take as short as 3 days from fertilization and gestation to adulthood [7]. The developmental fates for all its cells are deterministic and known [8]. The hermaphrodite nervous system is a simple network of 302 neurons whereas the male nervous system incorporates 381. The neuronal

wiring, for wild-type hermaphrodites, has been fully mapped [9]. Furthermore, a wealth of research has helped elucidate the behavioral roles for many sensory, command and inter neurons within this network.

A host of genetic manipulations altering the *C. elegans* genome, alongside several unique wild-type isolates, are available for experimental investigation. WormBase, a collaborative database housed at Caltech, serves as a single, diverse repository for worm experiments, genomics, phenotype, and related information [10]. WormBase version WS228 lists the Caenorhabditis Genetics Center (CGC) [11] as having 546,796 strains available for order (all WormBase information has been generously provided by Dr. Karen Yook, Science Curator for WormBase, as a personal communication). These CGC strains represent mutations in 5,652 independent genes among 20,484 protein-coding genes, including 5,034 unique gene knockouts provided by the *C. elegans* Knockout Consortium. Another 3,949 gene knockouts are available from the Japanese National BioResource Project (NBRP). In addition to the availability of these genetic mutants, RNAi feeding libraries provide another method to obtain quick genetic manipulation. These libraries are capable of selectively knocking down expression for about 86% of the *C. elegans* genome, albeit with modifications necessary for efficacy in neurons[12-14].

1.2 Behavior and Phenotype

C. elegans have a simple external morphology and display a set of well-characterized behaviors. A long annulated cuticle encases the worm and contains orifices at the mouth, anus, and, for hermaphrodites, at the vulva as well. Males have special mating adaptations at their tail. At low magnification, as would be expected, *C. elegans* have a simple worm-like morphology and the tail has a slightly sharper taper than the head. Physical limitations of cuticle flexibility and muscular innervation limit the range of shapes a worm can display. Specifically, it has been shown that four dimensions are sufficient to describe 95% of the variance in wild-type worm shapes (as defined by bending angles along the worm body) [15].

The most salient and commonly explored behavior in *C. elegans* is locomotion. Locomotion is powered by 95 muscles separated into four longitudinal quadrants [16]. These quadrants represent left and right subdivisions of the dorsal and ventral sides. Worms move by means of opposing dorsal ventral contractions. Greater innervation in the head and neck, permit better control and more freedom in their movement. Depending on environment, worms move by crawling or swimming. There is some debate as to whether crawling and swimming represent two ends on the continuum of locomotion or whether, at a higher level, they are driven by different neural circuits [17-20]. Generally, crawling takes place on surfaces and in high-viscosity liquids. It involves smooth transitions through sinusoidal shapes at wavelengths less than the worm length. Swimming, on the other hand, takes place in low-viscosity liquid environments and involves higher-frequency transitions between bowed shapes.

Worms crawl on either their left or right sides. Rolling from one side to another is infrequent and rarely observed outside of a small subset of mutants [21]. Crawling can take place either forwards or backwards and is punctuated by pauses and re-orienting events [22]. Omega-bend reorientation involves a deep bend, during which the worm forms a shape characteristic of the Greek letter Ω, to turn nearly opposite its former direction of motion. Pirouettes, on the other hand, involve a brief reversal to achieve a smaller magnitude of reorientation. Independent of locomotion, the head swings from side to side and performs small extensions and retractions to explore the local environment and direct orientation. Several other behaviors are less

salient, due to the spatial scale, temporal scale, and/or frequency at which they occur, but still represent highly active areas of research. Among these behaviors are dynamics associated with coiled shapes, egg laying, defecation, and pharyngeal pumping.

Sydney Brenner's seminal paper on *C. elegans* genetics identified approximately 300 mutations with salient behavioral and morphological effects [1]. These mutations were categorized into seven classes that broadly illustrate the identified differences from wild type worms. Five morphology classes are used to label long, small, dumpy (shorter than wild type but with a normal diameter), blistered, and variable abnormal mutants (as a catchall for worms excluded from the four other categories). Two behavioral classes are used to label uncoordinated and dorsal-ventral rolling worms. The initial characterization of these anomalous worms cast light on their underlying genetics, and for many of these mutants, has eventually resulted in uncovering the biological role of their mutated gene. Nearly a decade later, Jonathan Hodgkin attempted a finer scale classification of these uncoordinated worms, partitioning 94 *uncs* into 16 categories, to better distinguish their phenotype. These categories combine terms such as kinker, coiler, twitcher/shaker, backward, forward, slow, and sluggish to describe locomotory behavior [23]. Over the last four decades, many additional mutants have been characterized based on morphological differences (e.g., *mab* as an acronym for male morphological abnormalities and *pvl* labeling hermaphrodites with a protruding vulva), their response to stimuli (e.g., *che* for chemotaxis mutants, *ttx* for anomalous thermotaxis, *osm* labeling abnormal osmotic avoidance, and *mec* indicating mechanosensory mutants), egg laying defects (labeled *egl*), atypical defecation, and numerous other observations as well as behavioral assays [24].

1.3 Experiments Guided by Phenotype

A central topic in biology research encompasses relating functional roles of genetics to their phenotypic effects. Within *C. elegans*, many experiments have been guided by initial phenotypic characterizations, which later led to focused investigation of the underlying mutated genes and eventual elucidation of their function. Numerous examples come as a result of Sydney Benner's classic work. One particularly early example is the discovery of a gene function for *unc-54* in encoding the myosin heavy chain which composes the major fraction of nematode myosin. In his classic paper, Brenner described *unc-54(e190)* worms as paralyzed, with a defect in their body muscle cells [1]. Follow up work by Epstein et al. and later, by MacLeod et al., uncovered the role of *unc-54* in encoding a myosin heavy chain necessary for bodywall muscle construction, further identifying the specific mutation present within the mutagenized allele created and discovered by Brenner [25,26].

More recently, many biological tools have been constructed with the purpose of facilitating this investigation into the relationship between genetics, biological function, and phenotype. Among those tools in silico are several large databases of genomic sequence data accompanied by experimental annotations, tools for genomic and proteomic functional predictions, and a host of recent databases linking genetics to phenotype [27]. Alongside these tools, and specific to nematodes, worm trackers provide a precise method to automatically measure behavior and morphology, and thereby acquire data sets linking worm genetics with their phenotype. Use of worm trackers in exploring new phenotypes has been sparse. But, in one recent example, Li et al. used a single-worm tracker to investigate a predicted mechanosensory mutant (based on TRP channel homology), uncovering a functional role for *trp-4* as a *C. elegans* stretch receptor gene [28]. Careful measurements of velocity and body bends helped guide this experiment as well as contribute the evidence necessary to elucidate the role of *trp-4*. Several other high-profile experiments have employed worm tracking to similar ends, including one within the same year to explore nicotine-dependency in *C. elegans* and mediation of this dependency through TRP and AChR proteins [29].

1.4 Worm Tracking

Automated worm tracking appears to date back to 1985. David Dusenbery initiated the effort, constructing a multi-worm tracker to measure behavior in response to CO_2 concentration [30]. This tracker used 10-second windows to automatically count binary events of motion and turns independently. A 2001 study by Hardaker et al. introduced what may be considered the first single-worm tracker. Their hybrid tracking system coupled automated measurements of locomotion with other measures, such as egg-laying quantitation, done by eye from recorded video [31]. This single-worm system advanced automation to measurements of velocity and direction with a precision of 1Hz. In 2002, Baek et al. contributed a significant advance to automated single-worm tracking, constructing a system capable of measuring 94 features at 2Hz precision [32]. With this system they profiled five worms, representing well-known gene mutations. Since then, the Baek tracker has seen continuous improvements and now supports 161 measured features at 10Hz [33].

Contemporary worm trackers represent substantial trade offs between resolution and throughput. Single-worm trackers tend to offer higher-resolution measurements with more features, whereas multi-worm trackers offer greater experimental throughput. Specifically, single-worm trackers can measure morphology, distinct body positions during rest and locomotion, and discriminate between the head and tail as well as their associated fine-scale movements [15,33-36]. Present limitations, in instrument fabrication and processing bandwidth, constrain total camera resolution and the frame rates available at high resolutions. These constraints limit the measurement details available to multi-worm trackers. At the requisite magnification necessary to measure fine details, coupled with the highest camera resolutions available, multi-worm trackers would frequently lose tracked worms from their field of view. For this reason, multi-worm trackers are often restricted to low magnification and, consequently, simply measuring velocity and path. On the other hand, multi-worm trackers permit significantly higher throughput by assaying many worms in parallel [37-39]. Achieving equivalent throughput with single worm trackers is, for most laboratories, cost and space prohibitive.

1.5 Thesis Aims

The benefits of single-worm tracking, in terms of precise high-resolution phenotypic quantification and analysis, and its contributions to experimental investigation, motivate my thesis.

In Chapter 2, I introduce a new single-worm tracker, titled Worm Tracker 2.0 (WT2) whose software and hardware I built to address several limitations of existing units. Among other benefits, this tracker removes several environmental restrictions, permitting high-resolution worm tracking off food, on food (as is necessary to keep worms from fleeing their plate), and within liquid environments which assay worm swimming behaviors. Two choices of inexpensive, magnifying cameras (with 200x and 500x zoom) permit recordings of all stages of worm development, post hatching. The entire assembly is approximately 4-fold less expensive than our previous units and, therefore, permits purchasing more units to increase experimental throughput. Several other design decisions maintain the worm in a relatively stimulus-free environment keeping the worm stable, undisturbed by stage motions, as well as using long wavelengths for illumination so as not to trigger avoidance behaviors associated with shorter wavelengths.

In Chapter 3, I introduce a collection of new algorithms I built to extract worms from video and measure their morphology and behavior. These algorithms are primarily motivated by the new WT2 hardware. They tackle such needs as scale-invariance (requisite for segmenting the variety of wild-type and mutant proportions present at different life stages) as well as adaptations to extract worms from the difficult environments permitted by our assembly. Several complications that troubled previously published algorithms are addressed, thus ultimately leading to more accurate and precise measurements of the worm skeleton (especially at the head and tail which carry significant phenotypic information) in our videos. Given our data set which, at about 1/3 of a billion frames, is several orders of magnitude larger than previous sets, these improvements yield high accuracy and precision for a collection of measurements that would otherwise take a substantial amount of time to correct. Due to the size of our data set and the necessity of manual checks, I created a new method of video overlays, which permit quick manual review of our worm

measurements. Additionally, I present a set of features and viewing tools to analyze plots, histograms, and on-plate dynamics for worm phenotypic measurements.

In the final section, Chapter 4, I explore the results of our data collection and analysis. Initially, the sensitivity of the WT2 system is profiled through measures of growth and habituation among 25 wild-type worms, over 2 hours. More complex measures of sensitivity are then explored by comparing our laboratory stock of N2 to the one available from the CGC and, thereafter, to the ancestral relative of both these wild types, LSJ1. LSJ1 is an N2 descendant, preserved in axenic liquid culture, which retained the wild-isolate variants of the *npr-1* and *glb-5* genes [40-42]. With the sensitivity for WT2 profiled, I analyze the results from 1,218 N2 controls, collected over 3 years. The findings show that hourly differences in young-adult age and monthly changes (most likely a reflection of temperature) can significantly impact worm phenotype. An assessment of the statistical β indicates that, for popular behavioral features, roughly 10 worms discriminate a mean difference of two standard deviations from wild-type behavior with well over 90% power and, 20 worms discriminate one standard deviation with over 80% power. The analysis elucidates phenotypes for 76 previously uncharacterized genes. I further explore our new phenotypes through examples of locomotion behavior for several genes with multiple strain representations in our data set. Bioinformatic clustering is then performed for a collective set of 305 strains, a group of male N2s, and 24 N2 subgroups divided by hour, day, and month (several subgroups are not represented due to insufficient sampling for inclusion). The strength of the clustering is validated through nearly complete wild-type aggregation within its own cluster. Multiple clusters of published pathways provide further validation of the analysis. Thereafter, a cluster of acetylcholine receptors with published predictions from genomic sequencing data receives further predictive support through our phenotypic clustering. Finally, I detail a large online resource, for our entire collection of nearly 10,000 experiments, that provides visual assessments of our phenotypic measures and statistics (per strain) along with the raw data (for each experiment and, collectively, per strain) in multiple popular formats.

1.6 Bibliography

1. Brenner, S. The genetics of Caenorhabditis elegans. *Genetics* **77**, 71–94 (1974).
2. Hodgkin, J., Horvitz, H. R. & Brenner, S. Nondisjunction Mutants of the Nematode Caenorhabditis Elegans. *Genetics* **91**, 67–94 (1979).
3. Denver, D. R., Morris, K., Lynch, M. & Thomas, W. K. High mutation rate and predominance of insertions in the Caenorhabditis elegans nuclear genome. *Nature* **430**, 679–682 (2004).
4. Cassada, R. C. & Russell, R. L. The dauerlarva, a post-embryonic developmental variant of the nematode Caenorhabditis elegans. *Developmental Biology* **46**, 326–342 (1975).
5. Huang, C., Xiong, C. & Kornfeld, K. Measurements of age-related changes of physiological processes that predict lifespan of Caenorhabditis elegans. *Proc Natl Acad Sci USA* **101**, 8084–8089 (2004).
6. C. elegans Sequencing Consortium Genome sequence of the nematode C. elegans: a platform for investigating biology. *Science* **282**, 2012–2018 (1998).
7. Mello, C. C., Kramer, J. M., Stinchcomb, D. & Ambros, V. Efficient gene transfer in C.elegans: extrachromosomal maintenance and integration of transforming sequences. *EMBO J.* **10**, 3959–3970 (1991).
8. Sulston, J. E., Schierenberg, E., White, J. G. & Thomson, J. N. The embryonic cell lineage of the nematode Caenorhabditis elegans. *Developmental Biology* **100**, 64–119 (1983).
9. White, J. G., Southgate, E., Thomson, J. N. & Brenner, S. The structure of the nervous system of the nematode Caenorhabditis elegans. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.* **314**, 1–340 (1986).
10. Stein, L., Sternberg, P., Durbin, R., Thierry-Mieg, J. & Spieth, J. WormBase: network access to the genome and biology of Caenorhabditis elegans. *Nucleic Acids Res.* **29**, 82–86 (2001).
11. Caenorhabditis genetics center. *Nature* **284**, 513 (1980).
12. Kamath, R. S. *et al.* Systematic functional analysis of the Caenorhabditis elegans genome using RNAi. *Nature* **421**, 231–237 (2003).
13. Esposito, G., Di Schiavi, E., Bergamasco, C. & Bazzicalupo, P. Efficient and cell specific knock-down of gene function in targeted C. elegans neurons. *Gene* **395**, 170–176 (2007).
14. Calixto, A., Chelur, D., Topalidou, I., Chen, X. & Chalfie, M. Enhanced neuronal RNAi in C. elegans using SID-1. *Nat Meth* **7**, 554–559 (2010).
15. Stephens, G. J., Johnson-Kerner, B., Bialek, W. & Ryu, W. S. Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* **4**, e1000028 (2008).
16. Sulston, J. E. & Horvitz, H. R. Post-embryonic cell lineages of the nematode, Caenorhabditis elegans. *Developmental Biology* **56**, 110–156 (1977).
17. Park, S. *et al.* Enhanced Caenorhabditis elegans Locomotion in a Structured Microfluidic Environment. *PLoS ONE* **3**, e2550 (2008).
18. Berri, S., Boyle, J. H., Tassieri, M., Hope, I. A. & Cohen, N. Forward locomotion of the nematode C. elegans is achieved through modulation of a single gait. *HFSP J* **3**, 186–193 (2009).
19. Fang-Yen, C. *et al.* Biomechanical analysis of gait adaptation in the nematode Caenorhabditis elegans. *Proc. Natl. Acad. Sci. U.S.A.* **107**, 20323–20328 (2010).

20. Vidal-Gadea, A. *et al.* Caenorhabditis elegans selects distinct crawling and swimming gaits via dopamine and serotonin. *Proc. Natl. Acad. Sci. U.S.A.* **108**, 17504–17509 (2011).
21. Cox, G. N., Laufer, J. S., Kusch, M. & Edgar, R. S. Genetic and Phenotypic Characterization of Roller Mutants of Caenorhabditis Elegans. *Genetics* **95**, 317–339 (1980).
22. Croll, N. A. Components and patterns in the behaviour of the nematode Caenorhabditis elegans. *Journal of zoology* **176**, 159–176 (1975).
23. Hodgkin, J. Male Phenotypes and Mating Efficiency in CAENORHABDITIS ELEGANS. *Genetics* **103**, 43–64 (1983).
24. Hart, A. C. Behavior. *WormBook* 1–67 (2006).doi:10.1895/wormbook.1.87.1
25. Epstein, H. F., Waterston, R. H. & Brenner, S. A mutant affecting the heavy chain of myosin in Caenorhabditis elegans. *Journal of Molecular Biology* **90**, 291–300 (1974).
26. MacLeod, A. R., Waterston, R. H., Fishpool, R. M. & Brenner, S. Identification of the structural gene for a myosin heavy-chain in Caenorhabditis elegans. *Journal of Molecular Biology* **114**, 133–140 (1977).
27. Houle, D., Govindaraju, D. R. & Omholt, S. Phenomics: the next challenge. *Nature Publishing Group* **11**, 855–866 (2010).
28. Li, W., Feng, Z., Sternberg, P. W. & Xu, X. Z. S. A C. elegans stretch receptor neuron revealed by a mechanosensitive TRP channel homologue. *Nature* **440**, 684–687 (2006).
29. Feng, Z. *et al.* A C. elegans Model of Nicotine-Dependent Behavior: Regulation by TRP-Family Channels. *Cell* **127**, 621–633 (2006).
30. Dusenbery, D. B. Video camera-computer tracking of nematode Caenorhabditis elegans to record behavioral responses. *Journal of chemical ecology* **11**, 1239–1247 (1985).
31. Hardaker, L. A., Singer, E., Kerr, R., Zhou, G. & Schafer, W. R. Serotonin modulates locomotory behavior and coordinates egg-laying and movement inCaenorhabditis elegans. *J. Neurobiol.* **49**, 303–313 (2001).
32. Baek, J.-H., Cosman, P., Feng, Z., Silver, J. & Schafer, W. R. Using machine vision to analyze and classify Caenorhabditis elegans behavioral phenotypes quantitatively. *Journal of Neuroscience Methods* **118**, 9–21 (2002).
33. Zhang, S. *et al.* Profiling a Caenorhabditis elegans behavioral parametric dataset with a supervised K-means clustering algorithm identifies genetic networks regulating locomotion. *Journal of Neuroscience Methods* **197**, 315–323 (2011).
34. Leifer, A. M., Fang-Yen, C., Gershow, M., Alkema, M. J. & Samuel, A. D. T. Optogenetic manipulation of neural activity in freely moving Caenorhabditis elegans. *Nat Meth* **8**, 147–152 (2011).
35. Hoshi, K. & Shingai, R. Computer-driven automatic identification of locomotion states in Caenorhabditis elegans. *Journal of Neuroscience Methods* **157**, 355–363 (2006).
36. Wang, W., Sun, Y., Dixon, S. J., Alexander, M. & Roy, P. J. An Automated Micropositioning System for Investigating C. elegans Locomotive Behavior. *Journal of the Association for Laboratory Automation* **14**, 269–276 (2009).
37. Ramot, D., Johnson, B. E., Berry, T. L., Carnell, L. & Goodman, M. B. The Parallel Worm Tracker: a platform for measuring average speed and drug-induced paralysis in nematodes. *PLoS ONE* **3**, e2208 (2008).
38. Swierczek, N. A., Giles, A. C., Rankin, C. H. & Kerr, R. A. High-throughput

behavioral analysis in C. elegans. *Nat Meth* **8**, 592–598 (2011).
39. Tsechpenakis, G., Bianchi, L., Metaxas, D. & Driscoll, M. A novel computational approach for simultaneous tracking and feature extraction of C. elegans populations in fluid environments. *IEEE Trans. Biomed. Eng.* **55**, 1539–1549 (2008).
40. de Bono, M. & Bargmann, C. I. Natural variation in a neuropeptide Y receptor homolog modifies social behavior and food response in C. elegans. *Cell* **94**, 679–689 (1998).
41. Persson, A. *et al.* Natural variation in a neural globin tunes oxygen sensing in wild Caenorhabditis elegans. *Nature* **458**, 1030–1033 (2009).
42. Weber, K. P. *et al.* Whole Genome Sequencing Highlights Genetic Changes Associated with Laboratory Domestication of C. elegans. *PLoS ONE* **5**, e13922 (2010).

Chapter 2 : Worm Tracker 2.0

2.1 Abstract

Automatic quantification of *Caenorhabditis elegans* morphology and behavior, using worm trackers, provides a means for accurate phenotypic characterization, divorced of human bias. Traditionally, this work has been divided into a dichotomy of single-worm and multi-worm trackers, with a respective tradeoff between high-precision quantification of single worms and high-throughput experiments of multiple worms. Furthermore, worm environment and age, in single-worm trackers, have been restricted for the purposes of image quality. I here report the construction of a new single-worm tracker, titled Worm Tracker 2.0 (WT2) with approximately 4-fold reduction in cost over previous units as well as reduction in size. These cost and space reductions permit the purchase of multiple units to achieve higher throughput. Several other improvements permit tracking in multiple environments (including deep food lawns and liquid assays), at all life stages of *C. elegans* (from L1 to adulthood), and software that does not restrict the spatial and temporal resolutions for long durations of recording (the only restrictions stem from hardware choices). The standard hardware and software setup are described here.

2.2 Introduction

Experiments linking worm behavior to genetics are frequently done by eye, although, the work is often tedious and may be prone to experimenter bias. Moreover, fine scale measurements of body positions, morphology, and movement are nearly impossible to do by hand. The advent of worm trackers has transferred a significant portion of this work to the domain of computers [1]. Worm trackers use a camera to record video of behaving worms. Single-worm trackers often record at higher magnification and use a stage to keep a moving worm within video. These trackers come in multiple configurations (one example of which is illustrated by the Worm Tracker 2.0 system in Figure 2.2). The videos can then be reviewed by eye or, as is more often the case, a computer-vision algorithm locates the worm in each video frame and automatically computes a set of measurements. In automating measurements, worm trackers achieve significantly higher throughput and greater accuracy than can be performed by researchers alone.

Surprisingly, the first automated worm tracking publication dates back to 1985 [2]. David Dusenbery used a microcomputer and dark-field illumination to track as many as 25 worms simultaneously. The algorithm, while somewhat crude by today's standards, is capable of automatically tracking movement and direction changes in real time; and, similar methods are still in use today, among the highest throughput multi-worm trackers, which measure the frequency of swimming movements in 96 worms simultaneously [3]. The publication of a high-resolution single-worm tracker, by Baek et al. in 2002 [4], introduced a much finer-scale analysis that extracted the worm's body shape including its head and tail. In 2004, Feng et al. combined this single-worm tracker with a similar effort by Paul Sternberg's laboratory [5] and Geng et al. adapted the analysis to automatically assign head and tail to the two worm endpoints [6]. Further extensions in 2005 and 2006 contributed detection of egg laying, omega bends, and coiled shapes [7,8]. We consider the end product of these efforts from both the Schafer and Sternberg labs to be Worm Tracker 1.0 (WT1). In 2007, Clark et al. combined single-worm tracking with calcium imaging to simultaneously record neural activity and behavior in freely moving worms [9]. Most recently, in 2011, concurrent publications by Leifer et al. and Stirman et al., combined single-worm tracking with

spatially-targeted illumination and optogenetics to record worm behavior in real-time while, synchronously, exciting and inhibiting neural activity in response to behavioral patterns [10-12].

Choosing between single and multi-worm trackers represents substantial trade offs. Single-worm trackers offer more extensive and higher-resolution measurements whereas multi-worm trackers offer much higher experimental throughput (approximately 30-50 worms in parallel [13,14]). Specifically, single-worm trackers can measure morphology, distinct body positions during rest and locomotion, and discriminate between head and tail as well as their associated fine-scale movements [10,15-17]. Present limitations, in instrument fabrication and processing bandwidth, constrain total camera resolution and the frame rates available at high resolutions. These constraints limit the measurement details available to multi-worm trackers. At the requisite magnification necessary to measure fine details, coupled with the highest camera resolutions available, multi-worm trackers would frequently lose tracked worms from their field of view. For this reason, multi-worm trackers are often restricted to low magnification and, consequently, simply measuring velocity and path. But, multi-worm trackers present significantly higher throughput by assaying many worms in parallel [13,14,18]. Achieving equivalent throughput with single worm trackers is cost and space prohibitive.

In addition to limitations in throughput, existing single-worm trackers suffer several other confounds. The accuracy at which computer-vision algorithms segment a worm from a video frame is directly correlated to the quality of the image. For this reason, existing trackers use a microscope and its accompanying illumination to obtain images which clearly distinguish the worm from its background. Unfortunately, the weight of the microscope and its illumination preclude moving it on a stage. Therefore, the microscope is kept stable while the worm is moved to maintain it in view. Worms are highly sensitive to the small vibrations that may happen when a stage changes acceleration and it is unclear whether they are responsive to the acceleration itself. Moreover, swimming assays require a liquid environment where acceleration changes will induce waves and centrifugal effects on the worm. To aid in image quality, existing single-worm trackers use white light for illumination. Unfortunately, this lighting includes short wavelengths that have been

shown to trigger strong avoidance behaviors [19]. To provide clear image boundaries between the worm and its background, most existing single-worm trackers require the worm to be off food. Not only does this limit the range of available experimental conditions but, in the preferred experimental environment of agar Petri plates, worms will often crawl onto the sidewalls of the plate and permanently leave the viewable area. On commonly used 5cm plates, this often occurs within the first few minutes of recording, thereby limiting the length of continuous experiments. The alternative, a copper ring that acts as a chemorepellent [20], biases the worm's behavior to copper avoidance.

There is no community standard among worm preparation and tracking protocols. Freely behaving worms exhibit significant variability that is notoriously difficult to control. Worm behavior is known to be affected by temperature, environmental moisture, food, epigenetic effects associated with ancestral conditions, the presence of other worms, and many other factors. Moreover, environmental disturbances can result in long-term behavioral responses which take considerable time to return to their basal state. For example, locomotion significantly increases when a worm is relocated as is the case when moving to a tracking plate[14]. For these reasons, it would benefit the community to adopt a standard with strong controls that reduce behavioral variability.

2.3 Results

To address limitations of single-worm trackers, I have constructed new system entitled Worm Tracker 2.0 (WT2), described below and available at www.mrc-lmb.cam.ac.uk/wormtracker. Several modifications significantly reduce both cost and size. For example, our former published system, WT1 [21],(Figure 2.1) is representative of the least expensive units [15-17,20]. It costs approximately $20,000 whereas WT2 amounts to less than $5,000 and occupies approximately half as much space (Figure 2.2). The high costs of WT1, and similar systems, are primarily associated with their use of a standard microscope and specialized hardware. In the case of WT1, the costs are a combination of expenses that include a dissecting microscope (roughly $10,000), a National Instruments (NI) compatible camera to connect with this microscope (roughly $2,000), NI-compatible motorized stage hardware (roughly $10,000), and an NI Data Acquisition board to interface with the hardware (roughly $1,000); furthermore, programming the NI hardware requires a license for the LabVIEW closed-source developing environment (roughly $1,000). The cost and space reduction of WT2 have permitted us to purchase and run eight units, in parallel, significantly increasing the throughput that can be achieved. This section provides an overview of the WT2 system. Detailed protocols for construction and use of the WT2 system are presented in section 2.7.

The creation of WT2 was directly motivated by issues of throughput, specifically cost and space considerations. While a full review of single-worm trackers is beyond the scope of this text, a brief overview is presented in Table 2.1. Moreover, several issues, aside from throughput, precluded the use of existing combinations of hardware and software. Our previous system, WT1 [21], often crashed during experiments and eventually fell into disuse due to frustration. Several labs purchased the equipment and followed instructions to build WT1 but abandoned the project without success. In short, the Feng lab (Case Western Reserve University) was left as the sole location wherein WT1 could be made to work. Furthermore, WT1 suffered a hardware and software limitation of less than 2500 frames per video (a combination of the National Instruments data acquisition board used, and programming choices in response to this board's limitation). This restriction appears to remain in place. The most recent tracking publication from the Feng lab uses a

protocol of 4-minute long videos at 10Hz (2,400 frames) [15]. The Hoshi tracker software is not publicly available and its hardware is considerably out of date, requiring a video-cassette recorder [16]. The remaining single-worm trackers were not yet published when we began developing WT2 and, for the less expensive options [17,20], the software has yet to see a public release.

Tracker	WT2	WT1	Hoshi 2006	Stephens 2008	Wang 2009	Leifer 2011	Stirman 2011	Faumont 2011
Approximate Cost	$5,000	$20,000	NA (hardware unavailable)	$20,000	$20,000	> $20,000	> $20,000	> $20,000
Approximate Space	1x1 foot	1x2 feet	> 1x2 feet	> 1x2 feet	> 1x2 feet	> 1x2 feet	> 1x2 feet	> 1x2 feet
Phenotypic Measurements	702 measurements	161 measurements	4 motion states	4 eigenworms	2 (amplitude and speed)	2 (curvature and wave speed)	1 (speed)	2 (bending angle and speed)
Environmental Restrictions	none (relative to other trackers)	no swimming	off food, no swimming	off food, no swimming	off food, no swimming	off food, no swimming	off food, no swimming	off food, no swimming
Other Benefits	none	none	none	none	none	targetted optogenetics	targetted optogenetics	optogenetics and neural imaging

Table 2.1: Single-Worm Tracker Comparison

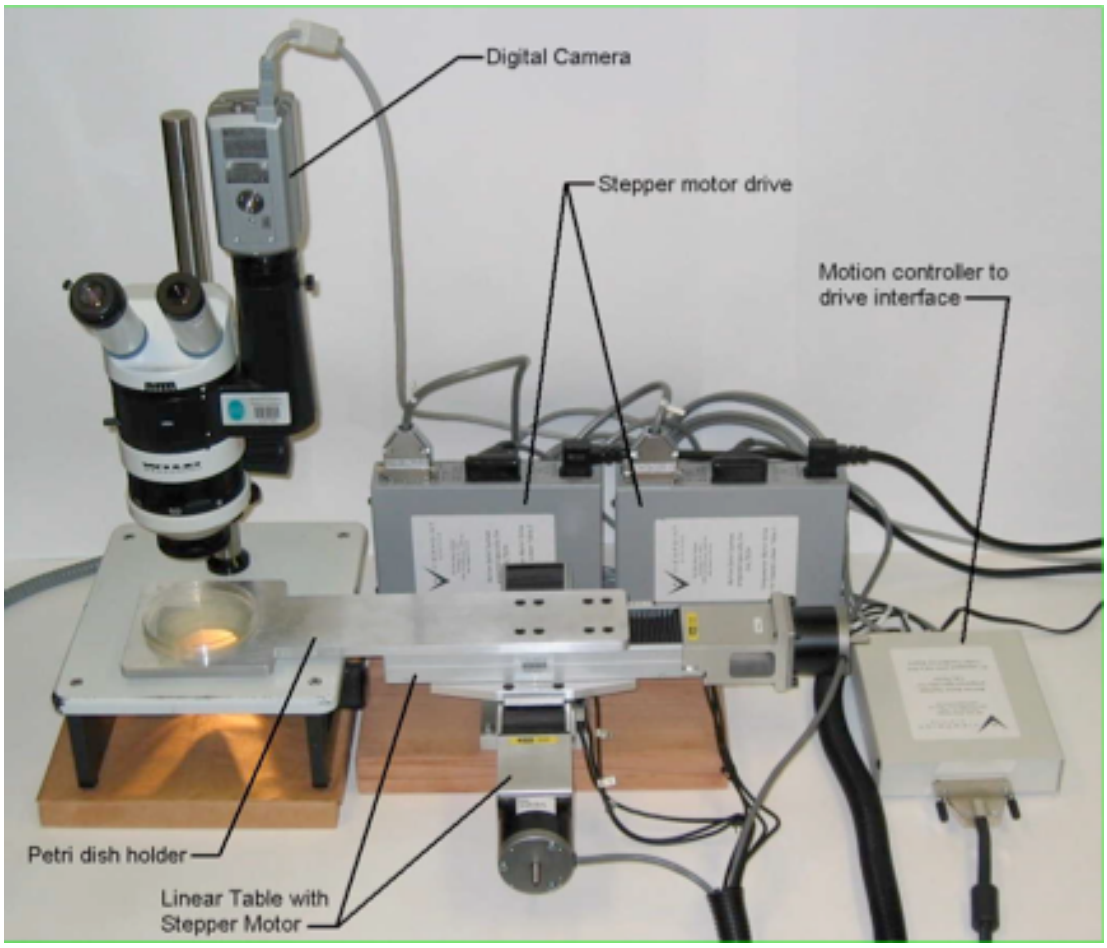


Figure 2.1: WT1 Tracking Hardware

A picture of the WT1 hardware, reprinted from "Automated imaging of C. elegans behavior" [21]. A digital camera is mounted onto a dissecting microscope to record videos of a single worm. A motorized stage, composed of two linear tables with stepper motors, stepper motor drives, and a motion controller to drive the interface, all feed into a National Instruments Data Acquisition board (NI-DAQ) within the tracking computer (not pictured). WT1 software uses a closed loop wherein video images of the worm are used to guide the motorized stage such that the worm is kept centered within the camera's field of view.

WT2 is the first and, currently, only single-worm tracker to use a small, inexpensive camera, with high magnification, in place of a microscope and camera unit. The microscope illumination is replaced with a compact red LED that emits a tight range of long wavelengths (620.5-645nm with a dominant wavelength of 627nm) which are not known to trigger avoidance behaviors, such as those present at blue through ultraviolet wavelengths [19]. The reduced weight and size of the camera and light permit them to be mounted on a motorized stage (the full weight of the stage-mounted chassis, with camera, lighting, and structural support, is 1.5kg). The stage moves the camera to keep the worm in view. The worm's enclosure is kept still, and separate from movement, to avoid accelerations and any consequent vibrations that can affect worm behavior (a limitation of all other current single-worm trackers). This setup further permits swimming assays. The tracking and analysis software use an adaptive thresholding algorithm to identify the worm, thereby enabling long experiments on food and under other difficult imaging conditions. The details of WT1 provided a basic understanding of the challenges involved in building the next generation of single-worm tracker. While several WT2 algorithms are derived from those published for WT1 (specifically an automated calibration for the worm's pixel-threshold intensity, shown in Figure 2.12F, and initial elements of extracting worms from video, discussed in Chapter 3), most of WT2's functionality is entirely novel since it deals with a new set of challenges stemming from flexibility in both hardware choices and recording environments. I am grateful to James Cregg and Victoria Butler who tested early versions of the WT2 system and provided feedback on the design.

The tracking protocol (2.7.3.5 Video Collection) for phenotypic characterization uses approximately 5 hours of behavioral recording, at 30Hz, from 20 representative worms raised in a strictly controlled environment, to ensure a dense behavioral sampling set. Worm preparation embraces a strict protocol (2.7.3.4 Worm and Plate Preparation) for controlling environmental effects on behavior. Plates for worm propagation, rearing, and recording have strict rules regarding composition and age. As plates age, their composition is affected through moisture evaporation and other changes. Throughout the process temperature is maintained as stably as possible, food is carefully measured then applied within strict time limits, at least one generation is passaged to achieve stability in ancestral conditions (e.g., to control the effects of maternal food availability on their progeny [22]), and worm numbers are

restricted to regulate crowding and starvation. Prior to tracking, worms are separated to remove all social cues. Upon being moved to the tracking plate, worms are placed centrally within their food lawn. A half hour habituation period is then observed, prior to recording, to permit a return to the basal behavioral state.

2.3.1 Tracking Hardware

Figure 2.2 shows the WT2 tracking hardware.

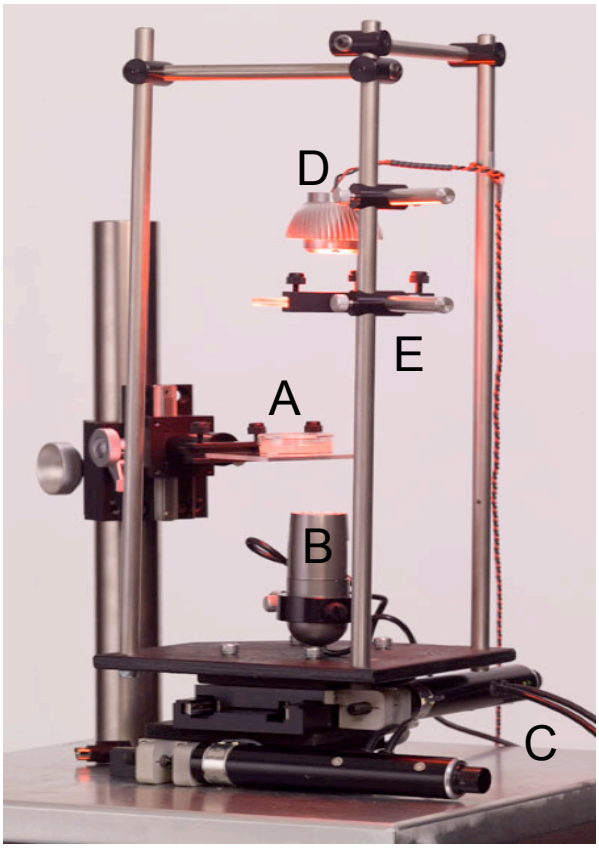


Figure 2.2: WT2 Tracking Hardware

A. A worm plate is kept still, on a separate platform, shielding it from stage movement and permitting swimming assays. B. An inexpensive DinoLite camera replaces a costly microscope and camera. The DinoLite's small size and low weight enable mounting it directly onto the stage. C. An inexpensive Zaber motorized stage replaces more costly Ludl and Prior models. D. A red LED provides strong illumination with little weight, enabling it to be mounted directly onto the stage. The red wavelength obviates *C. elegans* avoidance behavior associated with shorter wavelengths. The LED's extended distance from the sample protects against heating the worm. E. An opal diffuser diffuses the LED to create uniform illumination within the camera's field of view.

2.3.1.1 Camera

WT2 substitutes a DinoLite camera in place of the standard microscope and attached camera (Figure 2.2B). The DinoLite AM413T achieves equivalent imaging results to other published single-worm trackers. But, this substitution can represent significant cost reductions. For example, the DinoLite, which costs under $500 replaces the nearly $10,000 cost associated with the microscope and camera used in our former system.

Moreover, the DinoLite is small when compared to the alternative hardware. This size reduction has enabled us to mount the camera on the stage, moving the optics while keeping the worm's enclosure still (Figure 2.2A). In addition to shielding the worm far away from the effects of stage acceleration and vibration, this change permits swimming assays in fluid environments that would otherwise suffer waves and centrifugal, displacing forces.

The WT2 software uses Windows DirectShow to achieve compatibility with most scientific and consumer grade cameras. This choice was made to permit simple adaptation of WT2 to new applications. For example, the DinoLite AM413T5 has a much higher magnification than the AM413T and we have used it to record and analyze early larval stage worms from the L1 stage at hatching to the L4 stage that precedes adulthood.

2.3.1.2 Motorized Stage

The WT2 software has been programmed to communicate with any model of Ludl, Prior, and Zaber motorized stage (Figure 2.2C). Ludl and Prior stages represent two of the most popular scientific choices. Zaber stages can offer substantial cost savings with equivalent accuracy. For example, our former tracking assembly required a nearly $10,000 Prior stage whereas WT2 uses an approximately $3,300 Zaber unit with similar capabilities. I am grateful to Christopher Cronin who provided instruction to achieve compatibility with Ludl brand stages.

2.3.1.3 Illumination

WT2 uses a small, lightweight, red LED with a range of 620.5-645nm and a dominant wavelength of 627nm (Figure 2.2D). The wavelength is sufficiently far from those known to trigger avoidance behaviors, which present at wavelengths ranging from 500nm through to, at least, the ultraviolet spectrum [19]. To maintain room temperature near the worm, sources of heat must remain far away. While the camera and stage generate very little temperature change, the light source can generate substantial heat. Therefore the light is positioned far from the worm and, thereby, is less likely to heat it (although prolonged exposure may raise the temperature, especially in environments with less air circulation and consequent heat convection). Weight on the stage contributes to momentum and resulting vibration from acceleration. The further this weight is from the stage actuators, the more structural support it requires to avoid these vibrations. These considerations motivate the choice of a lightweight LED. Save for an improperly constructed WT2 at the beginning of our experiments, multiple examinations of our data collection have failed to show perceptible vibrations within video. The light source can be decoupled from the stage entirely but, to achieve the image quality requisite for computer-vision algorithms, the illumination must remain uniform in the presence of camera relocation. Large, bright, uniform sources of red illumination are significantly more expensive than an LED.

2.3.1.4 Computer

All contemporary computers running recent versions of the Windows operating system are capable of running WT2 to track and record video with 640x480 resolution at 30 frames/second. Therefore, the hardware protocol (2.7.3.2 Hardware Assembly) does not specify a particular choice of computer (although we do provide the details of our own setup). We do, however, recommend using RAID as well as other backup schemes to ensure that disk failure does not result in catastrophic data loss.

2.3.2 Tracking Protocol

Insufficient sampling can result in a sparse and biased phenotypic characterization. To combat this, for each strain, the tracking protocol (2.7.3.5 Video Collection) uses a data set compromised of roughly 20 worms recorded, in aggregate, for 5 hours at 30Hz. Individual worms can suffer biases due to a host of environmental and genetic effects. Therefore, roughly 20 worms are used to provide a robust sampling set (Chapter 4 presents data detailing the relationship between statistical power and the number of worms sampled). Due to the volume of our data set, in practice, we often record more worms as insurance against videos lost to worm escape and any potential equipment malfunctions. Foraging activity represents a lower bound for temporal scale. Huang et al. had shown that foraging movements can reach just over 10Hz [23]. This implies a Nyquist limit of at least 20Hz to measure foraging. Consequently, we use a 30Hz sampling rate to capture the fine details associated with these movements. Conversely, at present our analysis software does not assay any behavioral features that extend beyond the order of seconds. Therefore, we use 15-minute videos to adequately sample behavioral sequences. For the vast majority of tracked worms, 5 hours of video appears likely to provide a robust sample of the assayed behavioral space although rigorous experimental investigation of this hypothesis has not yet been explored.

C. elegans are highly sensitive to their environment. Their behavior is known to be affected by temperature, environmental moisture, food, the presence of other worms, epigenetic effects associated with ancestral conditions, vibration, and many other factors. The worm preparation protocol (2.7.3.4 Worm and Plate Preparation) employs a host of procedures to minimize and control for behavioral variability within sampled populations. Every worm is matched to a wild-type, N2 control raised, prepared, and tracked at the same time under identical protocol. We strongly recommend keeping the temperature to within 2°C of 22°C, for the entire life cycle of the worm, leading up until tracking. Unfortunately, a green policy implemented by our building management appears to have allowed slightly lower temperatures at night for our own collection, acting as a cautionary tale in favor of using incubators. Our worm preparation protocol of picking L4 worms the evening prior to tracking helped ensure that all worms were near the same stage when tracked. That stated, data

presented in Chapter 4 will show significant monthly effects on phenotype that appear related to changes in overnight temperature. Beyond stabilizing temperature, maintenance and growth plates are used within one week of pouring to ensure similar levels of moisture. Food quantity is carefully measured and its age, on the plate, is strictly controlled. Each plate has a strict number of worms to control for crowding and starvation. At least one parent generation is passaged under these conditions to regulate the ancestral environment (e.g., effects of parental starvation on their progeny). Chapter 4 provides data indicating that 30 minutes is sufficient to bypass the steep part of the habituation curve for wild-type speed. Given our focus on locomotory behaviors, we use changes in speed as a proxy for worm habituation. Therefore, prior to tracking, worms are placed within the center of their food lawn then given 30 minutes to habituate to the tracking environment. During tracking, the worm plate is kept still to prevent acceleration and vibration.

2.3.3 Tracking Software

2.3.3.1 Overview of the Tracking Program

The WT2 tracking software provides a simple interface to automate worm tracking along with several customizable features. For the tracking protocol herein (2.7.3.5 Video Collection), the software is used to locate the worm on the plate, track it, and record video of the behavior into a file. The stage position is concurrently logged to provide a translation to absolute coordinates. Three preference screens provide adjustments to routine functional settings as well as adaptations for special applications.

2.3.3.2 Standard Tracking

WT2 automates several features to simplify the standard tracking protocol. The software offers image correction for camera vignetting and dirt that may confound worm tracking. Calibration is performed, automatically, to determine the conversion from onscreen pixels to absolute coordinates (2.7.3.3.3 Software Calibration and Figure 2.13). Worms may be tracked using a manual threshold, an

adaptive threshold that can accommodate more complicated and changing image conditions, or a motion threshold which follows movement (please refer to section 2.3.3.4 which discusses these thresholds in greater detail).

2.3.3.3 Specialized Tracking

WT2 provides several adaptations for specialized tracking applications beyond the specified protocols. The software is compatible with all DirectShow cameras (a superset of USB cameras in addition to other cameras with more specialized connectivity), irrespective of whether they record color or grayscale video (please refer to section 2.7.3.3.2 and Figure 2.10 for more details). Cameras with these drivers represent the vast majority of commercial and consumer-grade equipment. Through this compatibility WT2 can offer different resolution, magnification, and cost savings beyond the standard configuration. The software is compatible with all Ludl, Prior, and Zaber model stages. Through this compatibility WT2 can accommodate stages with different resolution, maximum travel distance, maximum cargo weight, and cost savings beyond the standard configuration.

WT2 provides absolute coordinates onscreen. Behavioral responses to fixed-position stimuli, as in chemotaxis assays, are one application of onscreen absolute coordinates. The software provides an interface to invert thresholding and follow bright objects as may be required for dark-field and fluorescent tracking. The software can also be configured for semi-automatic tracking. Semi-automatic tracking uses manual control, such as a joystick, to correct automatic tracking failures when the tracked object moves too fast and escapes the field of view. High-resolution tracking of fluorescent neurons is an example of an application that requires both inverted thresholding and semi-manual control.

2.3.3.4 Overview of the Worm Identification and Tracking Algorithms

WT2 uses a standard algorithm, originally published by Baek et al. [4] to identify and track worms (Figure 2.3D):

1) Incoming video images are converted to 8-bit grayscale (Figure 2.3A-B).
2) A threshold is used to separate pixels into binary categories of foreground and background.
3) A fast connected-components algorithm [24] is used to discover the largest 8-connected foreground pixel set. Two pixels are considered 8-connected if they touch each other at any of the eight possible vertices. The largest connected component is considered the worm (Figure 2.3C).
4) The worm's centroid is computed as the pixel mean in the x and y axes. When the centroid leaves a user-defined boundary, the centroid distance from the image center is converted from pixels to stage steps, and the stage is directed to re-center the worm (Figure 2.3D).

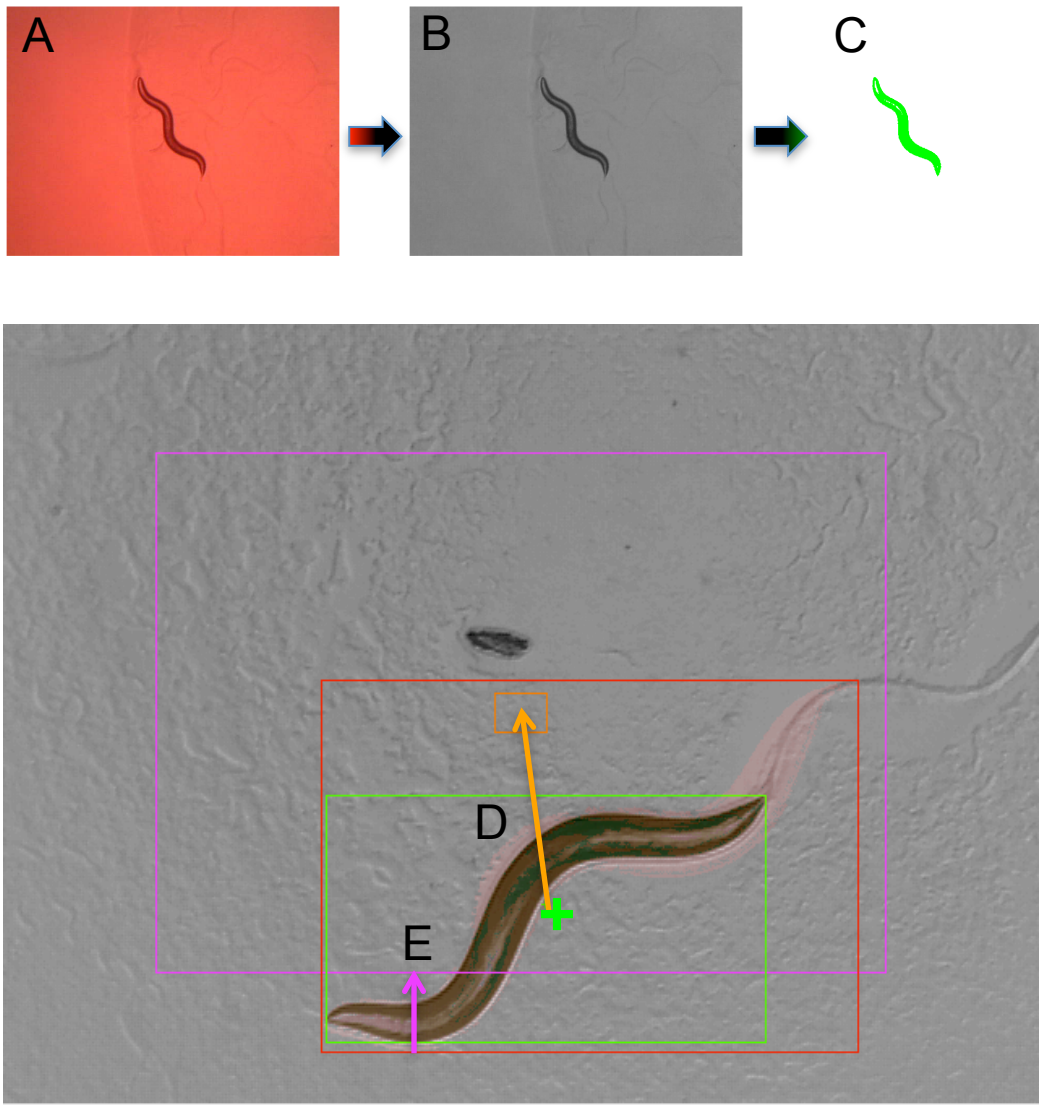


Figure 2.3: Worm Tracking Algorithm

A. An incoming video frame. B. The incoming video frame is converted to grayscale. The grayscale formula is defined by the user; the default formula uses the standard values of 30% of the red channel, 59% of the green channel, and 11% of the blue channel. C. The grayscale image is thresholded to identify the foreground pixels. The 8-connected components are labeled and the largest component is considered the worm. D. When tracking by centroid, the worm's centroid (labeled by the green cross) is re-centered (labeled by the orange arrow) whenever it escapes it bounding box (labeled by the orange rectangle). E. When tracking by motion, the worm's motion and minimum enclosing rectangle (MER -- labeled in red) are moved (labeled by the magenta arrow) to just within the bounding box (labeled the magenta rectangle), whenever the motion and MER escape the bounding box.

Adaptive thresholding assumes the onscreen pixels form a bimodal distribution, essentially two peaks, representing worm and background. WT2 uses the Otsu method [25] to automatically compute a threshold which separates both modalities, essentially a location to split both peaks, thereby identifying the worm (Figure 2.4). The Otsu method is a fast algorithm. It calculates the optimal threshold to minimize the variance within modalities and maximize the variance between modalities. In auto-thresholding mode, the software uses the Otsu method to identify the foreground mean and its standard deviation. A user-defined scaling factor determines the continuously adaptive threshold as a multiple of standard deviations from the foreground mean.

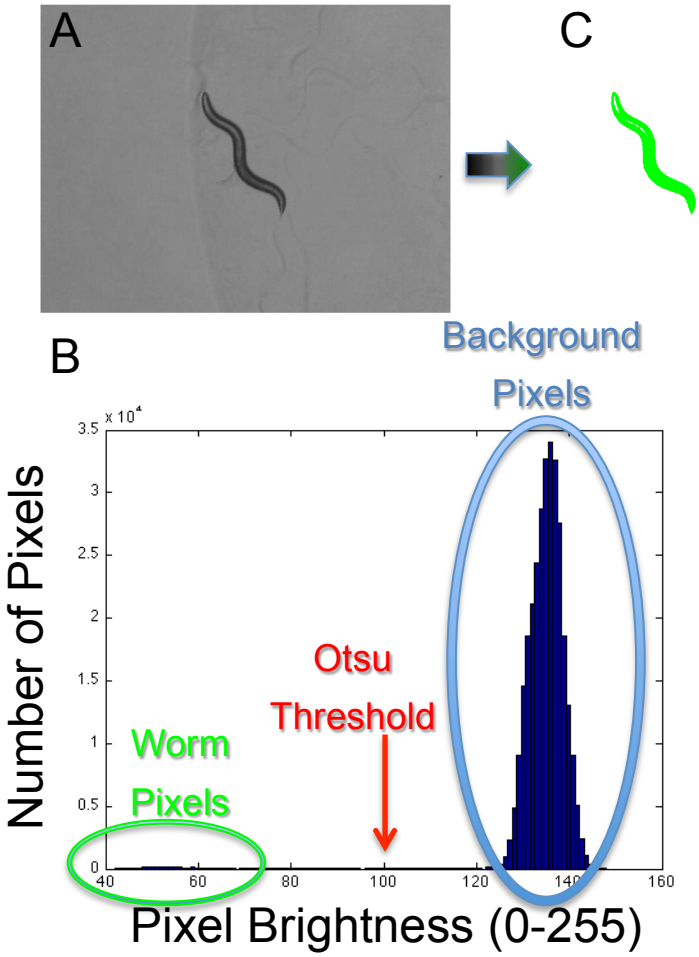


Figure 2.4: Adaptive Thresholding Using the Otsu Method

A. A grayscale video frame. B. The histogram of the grayscale video frame is assumed to be bimodal. The dark modality represents the worm pixels (labeled by the green ellipse). The bright modality represents the background pixels (labeled by the blue ellipse). The Otsu method calculates a threshold (labeled by the red arrow) that optimally splits both modalities. The threshold minimizes the variance within modalities and maximizes the variance between modalities. C. The grayscale image is then thresholded using the Otsu value to identify the foreground pixels. The 8-connected components are labeled and the largest component is considered the worm.

Motion thresholding (Figure 2.3E) is performed using a minor adaptation of the standard thresholding algorithm defined above:

1) Subsequent video frames are differentiated (i.e. subtracted from each other). The user defines the spacing (frame count) between these differentiated frames. When a stage movement occurs between differentiated frames, the two frames are cropped to remove the image areas lost and gained due to movement. The cropped frames are then aligned for differentiation.
2) The magnitude of the differentiated image is defined as the difference image.
3) The difference image is thresholded, using a user-defined value, to identify pixels with large grayscale changes.
4) The standard connected-components algorithm is run on the thresholded difference image.
5) A user-defined threshold is used to identify big, connected components comprised of large pixel changes. These components are considered the moving parts of the worm. The minimum enclosing rectangle (MER) for movement is computed as the minimum and maximum pixel coordinates, in the x and y axes, among all big, connected components. If the MER exceeds the movement boundary on opposing sides, the stage is directed to move so as to balance the MER equally on either side (Figure 2.3E). The MER cannot be centered as it represents only a partial view of the worm.

2.3.3.5 Tracking Software Code

The WT2 tracking software is open source and written in Java (version 6). The source code, available at http://sourceforge.net/projects/worm-tracking/, is accompanied by Javadoc documentation providing details to understand and modify the code.

2.4 Comparison with other Systems

WT2 presents a cost effective means to approach the throughput of multi-worm trackers while preserving the resolution afforded in single-worm tracking. The hardware represents a 4-fold reduction in price from our previously published unit [5]. This previous unit is a fair representation of the hardware costs involved in many other published single-worm trackers. Moreover, the current software presents a convenient interface for running multiple trackers at once. In our lab, we now employ eight units in parallel, supervised by one person, to collect videos characterizing mutant phenotypes. An additional four units are employed by individual lab members and often provided on loan to other labs. In total we have collected and analyzed approximately 12,000 videos, representing 3,000 hours at 30Hz, of over 300 worm strains with N2-matched controls. This volume and sampling resolution of data exceeds those of any previously published systems.

Existing trackers provide a host of specializations for specific experimental needs. Their options can be difficult to find in aggregate in any one published unit and may be one of the driving forces behind the multitude of published, single-worm tracker designs. WT2 is programmed to use Windows DirectShow compatible cameras. These cameras represent the majority of consumer and scientific grade cameras. Through this compatibility, WT2 is able to handle the high frame rates, large image resolution, and long recordings which continue to grow with improvements in camera and computer technology. The standard WT2 hardware design keeps the worm platform stable while moving the camera and illumination to perform tracking. This adaptation, along with a high-frame rate camera, has enabled us to use the software for tracking and analyzing swimming behavior. Moreover, using the standard setup we regularly record and analyze long videos of over 10 hours an ability absent from many of the single-worm tracking alternatives.

Many labs already own much of the equipment necessary for single-worm tracking. For example, Ludl and Prior stage are used in a host of applications unrelated to worm tracking. Microscopes with camera attachments and worm-scale magnification are nearly ubiquitous in worm research labs. WT2's compatibility with this equipment provides an even less expensive and quick alternative to purchasing

the standard hardware setup; conversely, strict hardware requirements associated with many other single-worm trackers precludes this option.

2.5 Limitations

Despite its cost and ability to generate and analyze large data sets, WT2 cannot compete with the experimental throughput of multi-worm trackers which can handle approximately 30-50 worms in parallel [13,14]. Also, due to its single-worm restriction, WT2 cannot measure parameters related to social behavior. On the other hand, WT2's camera compatibility as well as the high-resolution of the new DinoLite cameras, mean that a WT2 unit can be configured for use with a multi-worm tracker as well. Moreover, due to WT2's ability to handle small worms, extending the analysis for multi-worm tracking would simply require a collision detection algorithm or size thresholds to identify single-worm objects while avoiding touching worms and background artifacts (please refer to Chapter 3 section 3.5.2.2 for further discussion). That stated there are still arguments as to why hiring a person may be more cost effective and less error prone than purchasing and constructing any type of worm tracker. Primarily, labs often have access to cost-free volunteer labor. Moreover, some behaviors, such as defecation, are not yet amenable to automated quantification and would therefore require a potentially inaccurate proxy for assessment (e.g., using changes in worm length as a proxy for defecation). There are two strong counter arguments to this non-automated approach. First, as we discuss in Chapter 4, the WT2 analysis automatically provides, per group, 702 measures representing summaries of over 50 top-level features, a task well beyond human capabilities. Second, the automated, algorithmic nature of WT2 provide far more accurate and precise measures than humans are capable of given finite time resources and the difficulties associated with tracing worm shapes in, for example, just the 27,000 frames representing a single 15-minute video at 30Hz.

Similar to almost all single worm-trackers, WT2 chooses to ignore frames marred by stage movement. Fortunately, these frames represent approximately 10%, for the most active worms, and frequently far less of the 5 hours collected per strain (please refer to Chapter 3 for more details). This result implies an approximate lower bound, with respect to stage movements, of 4.5 hours of data, at 30Hz, per worm strain. But, since stage movements occur to offset worm motion, removal of these frames may bias the data set marginally towards samples of non-moving behaviors and remove a small, mostly random subset of moving shapes. Differentiation (e.g., as

is used in calculating velocity) partially corrects for this bias by incorporating the scale of the computation to interpolate over the lost frames. Moreover, similar to almost all single worm-trackers, WT2 does not segment coiled shapes and these are not represented within the data. This results in a further loss of data.

Computer cameras continue to advance and the recent shift to USB 3.0 connectivity has meant wider bandwidth from computer peripherals to the processor. Consequently, while formerly limiting users to combinations such as 640x480 at 30fps or 1280x1024 at 5fps, the improved bandwidth is likely to sustain much larger video resolutions at very high frame rates. It is unclear how computer processors will scale to this task and, therefore, how fast WT2 will be able to track when burdened with a combination of very high resolution and frame rate. Moreover, WT2 currently relies on DirectShow for its generic camera compatibility. Should this standard be abandoned, the tracking software will require a significant update.

Finally, while an improvement over previous worm preparation and tracking protocols, our protocols still suffer issues confounding the strength of the experimental results. We attempt to roughly synchronize age by choosing L4 worms the evening prior to tracking but, as we will see in Chapter 4, the hour when tracking is performed can still have a significant effect on phenotype. Moreover, we compare mutant strains of *C. elegans* to wild types although, aside from any noted mutation(s), the two strains may well have a very different genetic background and, therefore, the observed phenotype may relate to background differences as opposed to the gene of interest. Therefore, further experiments are necessary to attribute phenotypic traits to genetics. For example, a stronger experimental design would compare a mutant strain to a rescue for the gene of interest and even, perhaps, a mock rescue to control for the effects of the rescue technique. Unfortunately, these additional experiments can significantly impact the high-throughput element of the work. As will be discussed in Chapter 4, we chose the alternative of looking into the phenotypic intersection between two or more strains sharing mutations within the same gene; in other words, shared phenotypes among multiple alleles.

2.6 Future Directions

2.6.1 Screening New Mutants for Phenotypes

At current we have screened over 300 mutant strains and identified a number of novel behavioral phenotypes (see Chapter 4). The CGC and NBRP provide access to over half a million more strains representing more than 7,600 gene knockouts, including many with uncharacterized phenotypes. There exist RNAi feeding libraries capable of knocking down expression for about 86% of the worm genome [26]. This wealth of mutants, many of which have subtle alterations within the same genes, permits multiple viewpoints relating genes to their phenotypic effects. Investigating multiple alleles, for an individual gene, enables a careful assessment of the consequences from loss and gain of function as well as dominant negative mutations. The spectrum of allelic effects on phenotype may help characterize the role of various protein domains and identify a purpose for conserved sequences. Furthermore, mutations with similar phenotypes among distinct genes, may indicate genetic products within the same pathway such as ligand-receptor pairs, proteins involved in building specific morphological structures, enzymes and their substrates, shared promoters, and similar connectivity. We continue to screen mutant strains and are attempting to recruit other labs to this task in order to build a repository mapping genetics to phenotype. In the near future, I would like to extend WT2 to segment and analyze coiled shapes and egg laying, thereby, adding key phenotypic descriptions of worm behavior.

2.6.2 Combined Behavior, Fluorescence Imaging, and Optical Stimulation

The methods by which sensory information is input, processed, and used to direct output and the underlying neural coding are a significant area of research. There are a plethora of techniques to observe and control neurons in awake, behaving animals. *C. elegans* is particularly suited to this endeavor due to its small nervous system and ease of genetic manipulation. Current techniques employ genetically encoded reporters of neuron and muscle activity as well as light-activated channels to activate and inactivate neurons. As mentioned earlier, the WT2 hardware has been extended to record behavior in synchrony with neural and muscle activity. We would

like to further augment this hardware to provide illumination at the wavelengths necessary to stimulate optically-gated neural control, similar to the Leifer and Stirman trackers [10,11] but without targeting body-specific locations. Moreover, we would like to explore options to reduce the size and cost of the new hardware prototype to promote high-throughput neurobehavioral recording with optic neural control.

2.7 Procedure

2.7.1 Experimental Design

The following points should be considered before beginning a collection of experiments.

i. Determine the phenotypes to quantify. Obtain and maintain the requisite strains.
ii. Phenotype can only be measured relative to a control. In the absence of specific hypothesis testing, each *C. elegans* experiment requires, at least, a wild-type, N2 control. Further rescue experiments as well as tracking multiple alleles for the gene(s) of interest, will significantly strengthen the experiment.
iii. We recommend a specific environment for experiments. Ensure that the environment is entirely as specified. Deviations will affect the phenotypic analysis. If you adapt the environmental conditions from our protocols, you must maintain your specified conditions identically across experiments. Worms behave differently depending on their age, agar recipe of their inhabited plate, age of the agar preparation, food type, food thickness, the illuminating wavelength, the presence of other worms, temperature, vibrations, and the environment during their and their parents' upbringing.
iv. Early larval stage worms require higher magnification. For these experiments, the DinoLite AM413T5 must be used in place of the DinoLite AM413T (now replaced by the models AD4113T5X and AD4113T, respectively).
v. Our protocols can be adapted to use multiple cameras (e.g., to record neural activity during behavior). These experiments require a method to synchronize concurrent videos. Ensure that hardware provides this synchrony or that you are using a distinguishable marker, in video, to align the time-series of these videos.
vi. Our protocols can be adapted to analyze behavioral response to location-specific stimuli. For these experiments, the stimuli must have fixed locations and these locations must be recorded for each experiment.
vii. Our protocols can be adapted to track any object distinguishable by dark or light coloration and/or size. Objects may require camera substitution to achieve appropriate magnification. High-speed objects may require low magnification

and/or motorized stage substitution to achieve appropriate tracking speeds. This adaptation will produce videos but the analysis software only works for worms.

2.7.2 Materials

2.7.2.1 Reagents

- Wild-type and mutant *C. elegans* may be obtained from the Caenorhabditis Genetics Center (CGC) and Japanese National BioResource Project (NBRP)
- Bacto Agar (Scientific Laboratory Supplies)
- Bacto Peptone (Scientific Laboratory Supplies)
- Cholesterol 5mg/ml in ethanol (Sigma-Aldrich)
- *E. coli* OP50 Strain (lab stock)
- $CaCl_2$ (Sigma-Aldrich)
- KPO_4 buffer pH 6.0 (Sigma-Aldrich)
- $MgSO_4$ (Sigma-Aldrich)
- NaCl (Sigma-Aldrich)
- 35mm Petri plates (VWR International)
- 50mm Petri plates or larger (VWR International)

2.7.2.2 Hardware

- Computer running Windows XP, Vista, or Windows 7 operating system; we use the Dell Precision T5400 with dual Intel Xeon E5405 quad core chipsets (2.00GHZ, 1333Mhz), 4096MB RAM (4x1024MB, 667MHz DDR2 Quad Channel FBD), three 500GB hard drives (7200RPM, Serial ATA II, RAID 5), and a 256MB NVIDIA Quadro NVS 290, running the Windows XP operating system
- Matlab version 7.10 (R2010a) or later (www.matlab.com)
- Java software version 6 or later (http://www.java.com)
- Worm Tracker 2.0 software (http://www.mrc-lmb.cam.ac.uk/wormtracker/)
- DinoCapture software version 2 or later (www.dino-lite.com)

- DinoLite AM413T and/or AM413T5 (for early larval stages) Dino-Lite Pro handheld digital microscope (Dino-Lite), both models weigh 100g; while these models are still available in limited circulation, they have been replaced by the AD4113T and AD4113T5X, respectively; we use the manufacturer setting of 200x, at a resolution of 640x480, to achieve roughly 4µm/pixels with a field of view nearly 2.5x2mm (as noted by the calibration measurements)
- Zaber KT-NA08A50 linear actuator, 50mm travel, 0.047625µm resolution, 0.22-8000µm speed, RS232 (serial port) control, kit (Zaber Technologies Inc.) -- ensure that the kit's power supply and electrical plug match your location
- Zaber T-NA08A50 linear actuator, 50mm travel, 0.047625µm resolution, 0.22-8000µm speed, RS232 (serial port) control (Zaber Technologies Inc.)
- 2 x Zaber TSB60M translation stages, 60mm travel, M6 thread (Zaber Technologies Inc.)
- Philips Lumileds LXHL-FD3C, LUXEON III Star LED, side-emitting, red, with a dominant wavelength of 627nm across a 621-645nm range, 90lm (Future Electronics); while this model is still available in limited circulation, it has been replaced by the LUXEON Rebel LXML-PD01 Red 40 Lumen LED
- Carclo Optics 10140, Luxeon 20mm wide beam frosted optic (Future Electronics)
- Carclo Optics 10043, Luxeon Star 20mm black holder (Future Electronics)
- Aavid Thermalloy KL-100-B, TO220 high power heatsink, 3.5°C/W (RS Components)
- Heatsink glue
- Power supply for 1400mA at 2.95V
- Comar 203 RM 01, 203x38.1mm rod (Comar Optics Inc.)
- Comar 20 RM 01, basic carrier for 38.1mm diameter rods (Comar Optics Inc.)
- Comar 30 XT 40, rack and pinion stage with 75mm travel (Comar Optics Inc.)
- 3 x Comar 16 BR 10, 50x10mm stainless steel post (Comar Optics Inc.)
- 2 x Comar 36 BR 10, 100x10mm stainless steel post (Comar Optics Inc.)
- 2 x Comar 66 BR 10, 150x10mm stainless steel post (Comar Optics Inc.)
- 3 x Comar BR 93, 300x10mm stainless steel post (Comar Optics Inc.)
- 6 x Comar 105 BR 10, cross clamp for two 10mm diameter posts (Comar Optics Inc.)
- 1 x Comar BR 94, cross clamp for three 10mm posts (Comar Optics Inc.)

- Comar 45 BH 10, 45mm post holder for 10mm posts (Comar Optics Inc.)
- Comar BR 92 252 BR 00, tube holder, bored out to 33mm diameter (Comar Optics Inc.)
- 3 x Comar 427 BR 00, 80mm edge clamp (Comar Optics Inc.)
- Comar 50 FQ 50, 64x64mm 50mm Fresnel lens (Comar Optics Inc.)
- Comar DO 90, 65x65mm opal diffuser (Comar Optics Inc.)
- 60x60x2mm transparent PMMA (acrylic glass) square
- 163x163x10mm wood square, stage platform
- 100x75x7.5mm wood rectangle, stage platform adaptor
- 350x350x30mm metal or wood square, base plate
- M6 machine screws and 40mm washer.
- Adhesive foam pads.

2.7.3 Methods

2.7.3.1 Reagent Setup

2.7.3.1.1 NGM (Nematode Growth Medium) Plates

NGM plates are prepared according to the standard protocol published by Sydney Brenner in 1974 [27].

2.7.3.1.2 Low-Peptone NGM Plates

Low-Peptone NGM plates are prepared similar to the standard protocol for NGM plates found in 2.7.3.1.1 with two modifications. For 1L of NGM, use 20g agar (instead of 17g) and use 0.13g of peptone (instead of 2.5g).

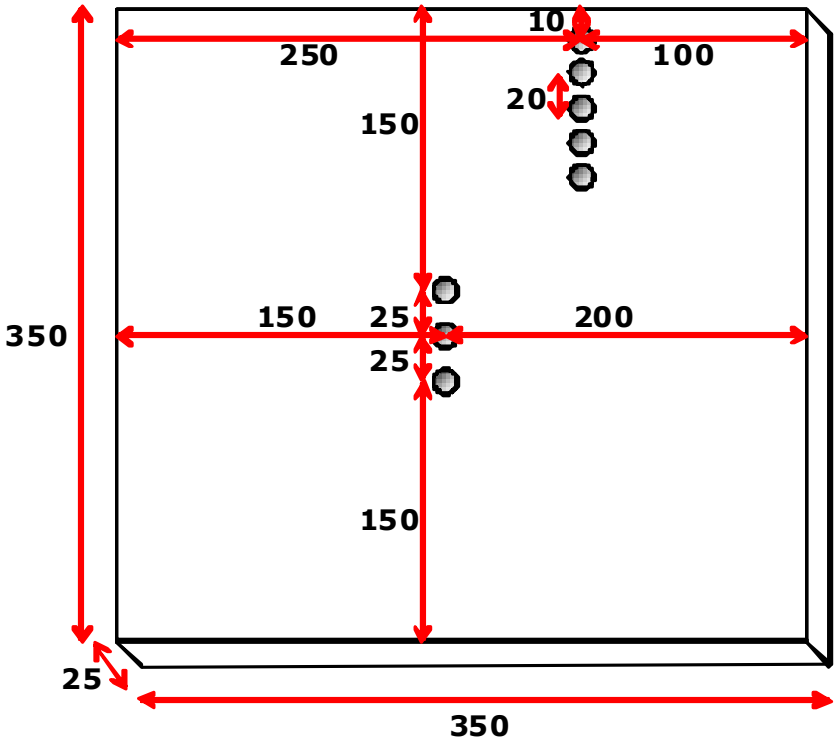
2.7.3.2 Hardware Assembly

2.7.3.2.1 Fabrication

Timing: 1 hour

Figure 2.5 through Figure 2.7 display the hardware fabrication. Hardware fabrication requires both a drill and a saw. Drill bits are required to bore a 4.2cm hole as well as M6 machine screw holes.

1) Drill the eight holes, shown in Figure 2.5, into the 350x350x30mm base plate.
2) Line the edges of the base plate's underside with adhesive foam padding.
3) Drill the eight holes, shown in Figure 2.6, into the 163x163x10mm stage platform. Then saw off the corner as shown in the figure.
4) Drill the five holes, shown in Figure 2.7, into the 100x75x7.5mm stage platform adaptor.



= hole large enough for M6 machine screw to fit loosely through (Note: for Zaber stages, scale hole size so that the screws to your stage model fit loosely through).

All measurements in mm.

Figure 2.5: WT2 Base Plate

The WT2 base plate serves as the base of the WT2 hardware. The plate may be made of metal or wood. Diagrammed here are the dimensions of the plate as well as eight M6-sized holes which require drilling.

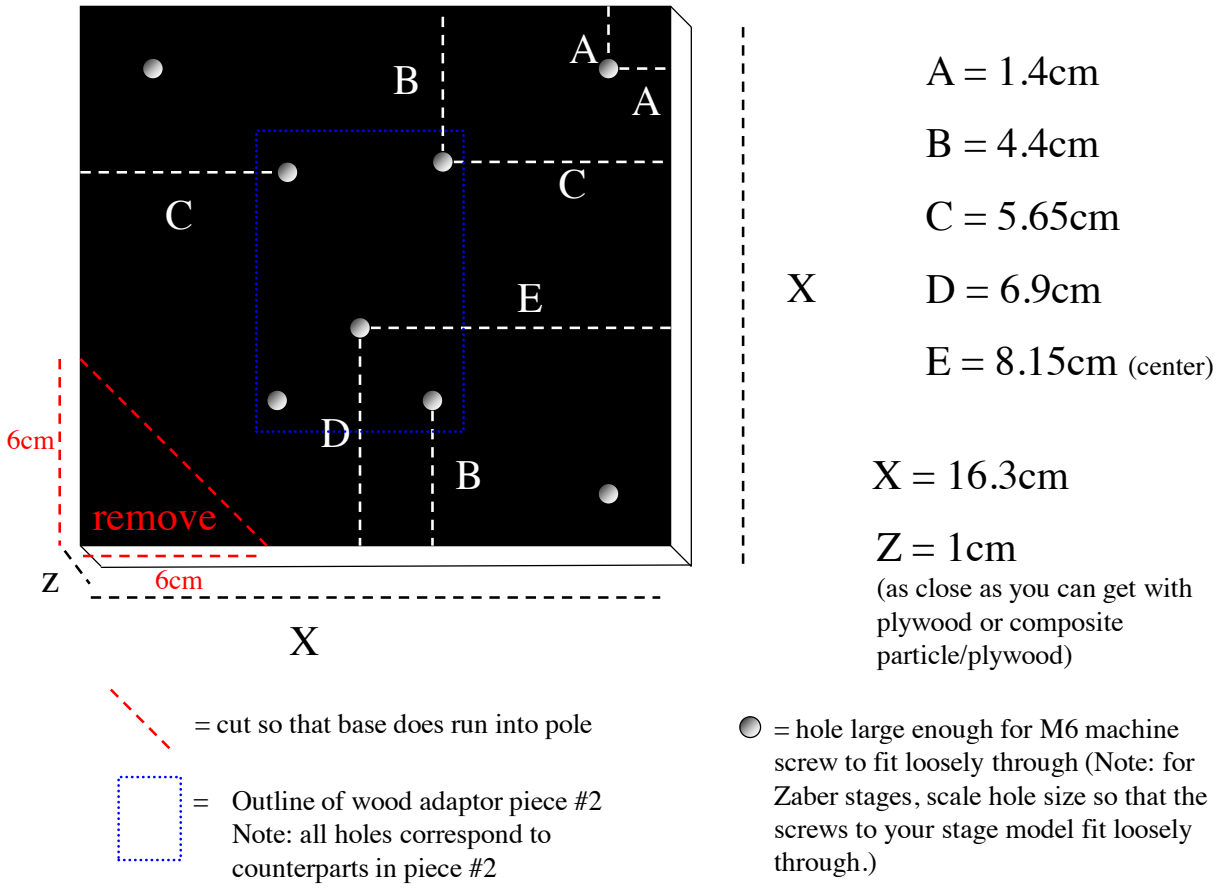
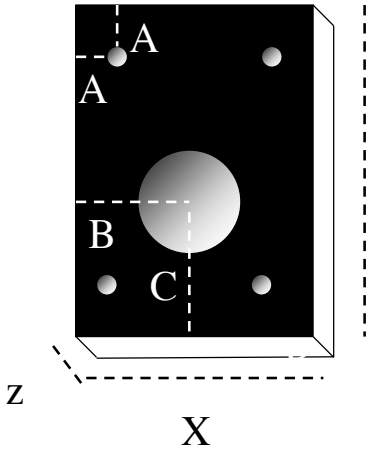


Figure 2.6: WT2 Stage Platform (Wood Adaptor #1)

The stage platform (wood adaptor #1) serves as the platform for the camera-light cage. This platform will support both the camera and light. Diagrammed here are the dimensions of the platform, eight M6-sized holes which require drilling, and a corner which must be removed.



A = 1.25cm

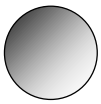
B = 3.75cm (center)

C = 3.75cm

X = 7.5cm

Y = 10cm

Z = <or=0.75cm
(must be high enough to allow for washer and screw head to be recessed within--this is large enough for an M6 machine screw (0.5cm) and a standard 4cm wood washer(0.15cm))



= 4.2cm
(Note: must be larger than the washer used to dampen the stage to camera vibrations -- 4cm)

= hole large enough for M6 machine screw to fit loosely through (Note: for Zaber stages, scale hole size so that the screws to your stage model fit loosely through)

Figure 2.7 WT2 Stage Platform Adaptor (Wood Adaptor #2)

The stage platform adaptor (wood adaptor #2) serves as the adaptor interface between the motorized stage and the stage platform (wood adaptor #1). Diagrammed here are the dimensions of the adaptor as well as four M6-sized holes and a washer-sized 4.2cm hole which require drilling.

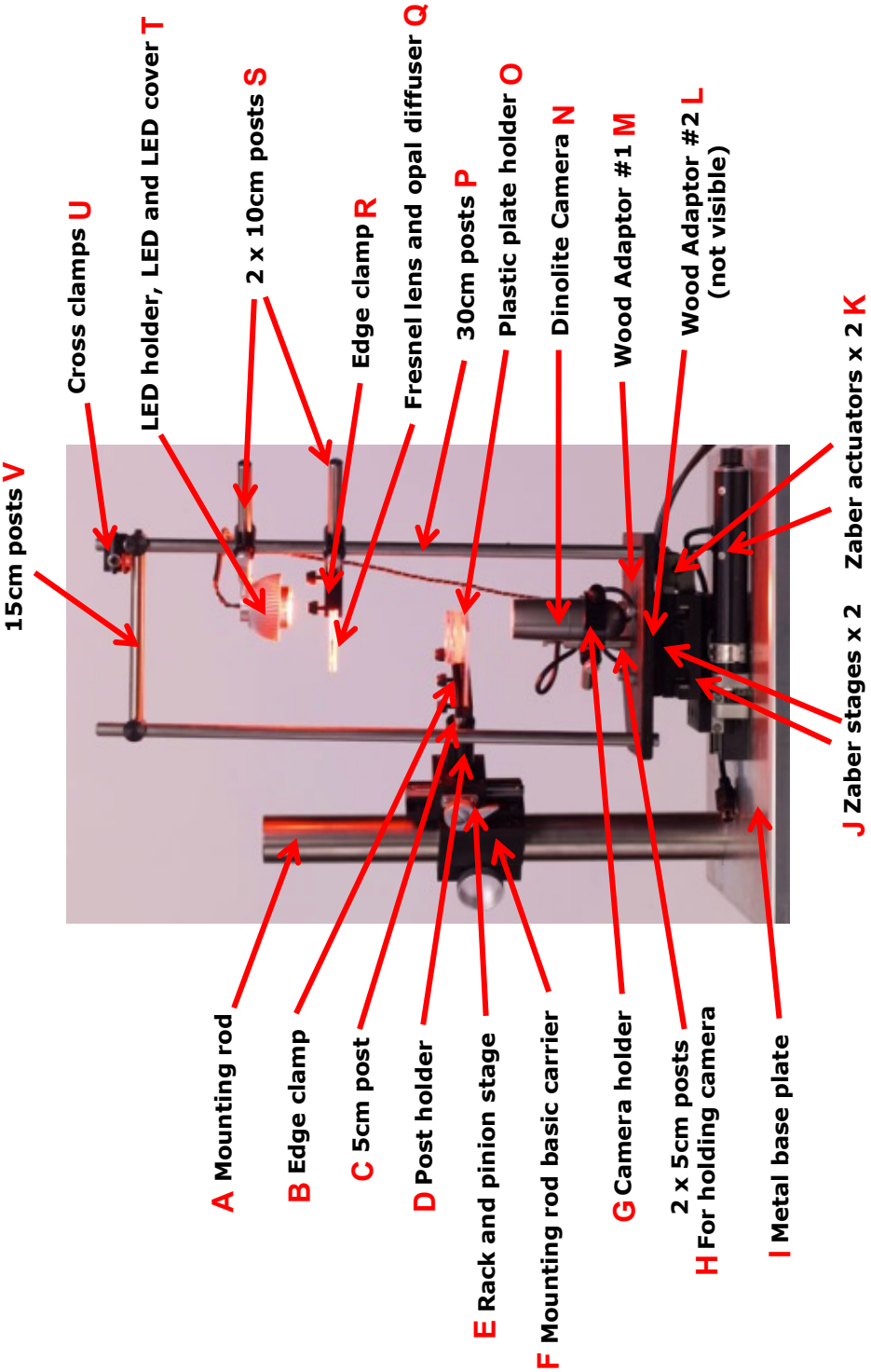


Figure 2.8: WT2 Hardware Assembly

The assembled WT2 hardware is shown here with every part labeled.

A. Comar 203 RM 01, 203x38.1mm rod

B. Comar 427 BR 00, 80mm edge clamp

C. Comar 16 BR 10, 50x10mm stainless steel post

D. Comar 45 BH 10, 45mm post holder for 10mm posts

E. Comar 30 XT 40, rack and pinion stage with 75mm travel

F. Comar 20 RM 01, basic carrier for 38.1mm diameter rods

G. Comar BR 92 252 BR 00, tube holder, bored out to 33mm diameter

H.1 Comar 16 BR 10, 50x10mm stainless steel post

H.2 Comar 105 BR 10, cross clamp for two 10mm diameter posts

I. 350x350x30mm metal or wood square, base plate

J. Zaber TSB60M translation stages, 60mm travel, M6 thread

K. Zaber KT-NA08A50 linear actuator, 50mm travel, RS232 control, kit

L. 100x75x7.5mm wood rectangle, stage platform adaptor

M. 163x163x10mm wood square, stage platform

N. DinoLite AM413T(5) Dino-Lite Pro handheld digital microscope

O. 60x60x2mm transparent PMMA (acrylic glass) square

P. Comar BR 93, 300x10mm stainless steel post

Q.1 Comar DO 90, 65x65mm opal diffuser

Q.2 Comar 50 FQ 50, 64x64mm 50mm Fresnel lens

R. Comar 427 BR 00, 80mm edge clamp

S.1 Comar 36 BR 10, 100x10mm stainless steel post

S.2 Comar 105 BR 10, cross clamp for two 10mm diameter posts

T.1 Philips Lumileds LXHL-FD3C, LUXEON III Star LED, side-emitting, red

T.2 Carclo Optics 10043, Luxeon Star 20mm black holder

T.3 Carclo Optics 10140, Luxeon 20mm wide beam frosted optic

T.4 Aavid Thermalloy KL-100-B, TO220 high power heatsink, 3.5°C/W

T.5 Comar 427 BR 00, 80mm edge clamp

U.1 Comar 105 BR 10, cross clamp for two 10mm diameter posts

U.2 Comar BR 94, cross clamp for three 10mm posts

V. Comar 66 BR 10, 150x10mm stainless steel posts

2.7.3.2.2 Assembly

Figure 2.8 displays the hardware assembly. Hardware assembly requires both screwdrivers and pliers.

2.7.3.2.2.1 Assembling the Motorized Stage

Timing: 30 minutes

Caution must be observed, when handling the Zaber T-NA08A50 linear actuators so as not to damage their lead screw. Caution must be observed, when handling the Zaber TSB60M translation stages so as not to damage their retraction spring.

1) Attach each Zaber T-NA08A50 linear actuator to a Zaber TSB60M translation stage (Figure 2.8J-K).
2) Mount a Zaber actuator-stage unit onto the three central holes in the base plate (Figure 2.8J-K and Figure 2.5).
3) Mount the remaining Zaber actuator-stage unit, perpendicularly, onto the Zaber actuator-stage (Figure 2.8J-K) from step 2. The two actuator-stage units should now form an x-y pair, with dimensional travel, mounted to the base plate.

2.7.3.2.2.2 Assembling the Worm Plate Platform

Timing: 30 minutes

1) Attach the Comar 30 XT 40 rack and pinion stage to the Comar 20 RM 01 rod carrier (Figure 2.8E-F).
2) Comar 45 BH 10 post holder to the Comar 30 XT 40 rack and pinion stage (Figure 2.8D-E) from step 1.
3) Insert the Comar 16 BR 10 (50mm) post, midway into the Comar 45 BH 10 post holder (Figure 2.8C-D) from step 2.
4) Attach a Comar 427 BR 00 edge clamp to the Comar 16 BR 10 (50mm) post (Figure 2.8B-C) from step 3.

5) Clamp the transparent PMMA square with the Comar 427 BR 00 edge clamp (Figure 2.8B,O) from step 4.
6) Mount the Comar 20 RM 01 rod carrier, from step 1, midway onto the Comar 203 RM 01 (203mm) rod (Figure 2.8A,F).
7) Mount the Comar 203 RM 01 (203mm) rod onto the base plate (Figure 2.5 and Figure 2.8A,I). Choose a hole, among the five near the base plate edge, such that the transparent PMMA square, from step 5, has its center positioned approximately over the center of the motorized stage's travel. Please note, the motorized stage has 50mm of travel in each axis. Therefore, the motorized stage's center is approximately 25mm, in each axis, from its retracted position.

2.7.3.2.2.3 Assembling the Red Light Source

Timing: 30-60 minutes

Figure 2.8T displays the assembled red light source. Caution must be observed so that the Phillips Lumileds LXHL-FD3C LED does not overheat.

1) Attach the back of the Phillips Lumileds LXHL-FD3C LED to center of the largest, flat surface on the Aavid Thermalloy KL-100-B heatsink using heatsink glue. The LED must face outward from the heatsink.
2) Snap the Carclo Optics 10140 frosted lens into the Carclo Optics 10043 lens holder.
3) Attach the Carclo Optics 10043 lens holder onto the Phillips Lumileds LXHL-FD3C LED from step 2. The lens holder should snap into place. Heatsink glue may be used for a more secure fit.
4) Connect the power supply to the Phillips Lumileds LXHL-FD3C LED from step 3. The LED must be driven at its maximum, 1400ma at 2.95V. The power supply must have at least a 600mm cord separating it from the LED; otherwise, the motorized stage may not be able to travel its full length without reaching the cord's maximum extension.

2.7.3.2.2.4 Assembling the Camera-Light Cage

Timing: 2 hours

Caution must be observed so as not to damage the DinoLite AM413T camera lens.

1) Attach the three Comar BR 93 (300mm) posts to the three holes at the corners of the stage platform (Figure 2.6 and Figure 2.8P).
2) Mount two Comar 105 BR 10 double cross clamps onto the central Comar BR 93 (300mm) post (Figure 2.8P,S.2) from step 1.
3) Insert a Comar 36 BR 10 (100mm) post into each of the two Comar 105 BR 10 double cross clamps from step 2 (Figure 2.8S).
4) Attach a Comar 427 BR 00 edge clamp to each of the two Comar 36 BR 10 (100mm) posts (Figure 2.8R,S.1,T.5) from step 3.
5) Place the Comar DO 90 opal diffuser on top of the Comar 50 FQ 50 Fresnel lens and insert them into the lower Comar 427 BR 00 edge clamp (Figure 2.8R,Q) from step 4. The opal diffuser must be above the Fresnel lens. The opal diffuser will face the light source and diffuse it. The Fresnel lens will face the camera and re-collimate the diffuse light into a uniform beam, through the sample, and onto the camera lens.
6) Insert the Aavid Thermalloy KL-100-B heatsink, from the assembled red light source, with LED facing down, into the upper Comar 427 BR 00 edge clamp (Figure 2.8T) from step 4. The Phillips Lumileds LXHL-FD3C LED must face the Comar DO 90 opal diffuser from step 5. Make sure the power supply cord does not obstruct the light.
7) Mount the Comar BR 94 triple cross clamp at the top of the central Comar BR 93 (300mm) post (Figure 2.8P,U.2) from step 1.
8) Mount a Comar 105 BR 10 double cross clamp at the top of each of the two side (non-central) Comar BR 93 (300mm) posts (Figure 2.8P,U.1) from step 1.
9) Insert a Comar 66 BR 10 (150mm) post into each of the two Comar 105 BR 10 double cross clamps (Figure 2.8U.1,V) from step 8. Then insert the opposite end into the Comar BR 94 triple cross clamp (Figure 2.8U.2) from step 7. Make sure the three Comar BR 93 (300mm) posts, from step 1, remain vertical. Secure this cage structure tightly; otherwise, motorized stage motion will vibrate the cage.
10) Mount the stage platform cage onto the stage adaptor (Figure 2.6, Figure 2.7, and Figure 2.8L,M). The stage platform's central-most hole must be aligned with the

stage adaptor's central 42mm hole. The stage adaptor must be on opposite side to the three Comar BR 93 (300mm) posts, from step 1.

11) Mount a Comar 16 BR 10 (50mm) post onto the stage platform's central-most hole using a 40mm washer on the opposing, stage adaptor, side (Figure 2.7 and Figure 2.8H.1).
12) Mount a Comar 105 BR 10 double cross clamp onto the Comar 16 BR 10 (50mm) post (Figure 2.8H) from step 11.
13) Use pliers to remove the DinoLite AM413T camera's transparent plastic lens guard. The guard extends the camera's gray-plastic body, opposite the cord. This guard obstructs placing samples at the camera's focal length when using high magnification.
14) Insert the DinoLite AM413T camera into the Comar BR 92 252 BR 00 tube holder (Figure 2.8G,N). Secure the camera at its base, near its cord, within the tube holder. The DinoLite AM413T5 camera, which achieves higher magnification, is of identical size and may be substituted in order to record early larval stages (L1 to L3) of *C. elegans*.
15) Attach the Comar BR 92 252 BR 00 tube holder from step 14 to a Comar 16 BR 10 (50mm) post (Figure 2.8G,H.1).
16) Insert the Comar 16 BR 10 (50mm) post, from step 15, into the Comar 105 BR 10 double cross clamp (Figure 2.8H). Make sure the camera lens is centered beneath, and facing, the Fresnel lens.
17) Mount the stage adaptor onto the motorized stage with the stage platform's cut corner facing the Comar 203 RM 01 (203mm) rod (Figure 2.6, Figure 2.7, and Figure 2.8A,J,L,M). Make sure the power supply and camera cords do not obstruct the motorized stage's travel.

2.7.3.2.2.5 Assembling the Worm Tracker

Timing: 1 hour

Caution must be observed to retract the Zaber T-NA08A50 linear actuators when not in use. Caution must be observed to turn off the Phillips Lumileds LXHL-FD3C LED, Zaber T-NA08A50 linear actuators, and computer when not in use.

Caution must be observed to protect the DinoLite AM413T camera's lens from dirt and damage when not in use.

1) Secure all loose fittings.
2) Connect the DinoLite AM413T camera to the computer.
3) Plug in the Phillips Lumileds LXHL-FD3C LED power supply to an electrical supply. Turn it on.
4) Connect the top Zaber T-NA08A50 linear actuator to the bottom one. Then connect the bottom Zaber T-NA08A50 linear actuator to the computer.
5) Plug in the bottom Zaber T-NA08A50 linear actuator to an electrical supply. Turn it on.
6) Adjust both Zaber T-NA08A50 linear actuator knobs to their neutral position.
7) Install the DinoLite AM413T camera software. The software includes Windows camera drivers and the DinoCapture software.
8) Download Java from http://www.java.com. Install the Java software.
9) Purchase Matlab from http://www.matlab.com. Install the Matlab software.
10) Download the Worm Tracker 2.0 software from http://www.mrc-lmb.cam.ac.uk/wormtracker/. Install the Worm Tracker 2.0 software.

2.7.3.3 Software Setup

The software must be setup prior to calibration and use. The supplemental software guide provides further clarification, options, and troubleshooting.

2.7.3.3.1 Camera Setup

Timing: 5 minutes

1) Run the DinoCapture software.
2) Set the camera to 640x480 resolution at 30 frames/second.
3) Navigate to the real time image settings:
4) Set the camera to grayscale (black and white) mode.
5) Turn off the camera's auxiliary white LEDs.
6) Save the DinoLite's configuration.

7) Position the light source centered over the camera.
8) Position the Fresnel lens and diffuser centered over the camera.
9) Quit the DinoCapture software.

2.7.3.3.2 Worm Tracker 2.0 Setup

Timing: 10 minutes

1) Turn on the red light source and motorized stage.
2) Run the Worm Tracker 2.0 software.
3) On the main screen (Figure 2.9C,F):
    a) Set the software to record, by time length, for 15 minutes.
    b) Make sure you are logging the stage coordinates.
    c) Make sure you are logging the stage position when recording.
4) Navigate to the display preferences (Figure 2.10A-B):
    a) Turn on grayscale conversion.
    b) Set the grayscale formula to use 100% of the red channel and 0% of the green and blue channels.
5) Navigate to the recording preferences (Figure 2.11A):
    a) Restore the default file name date format.
    b) If you have more than one tracker, append a machine identifier onto the filename date format. For example, for our third tracker, we use a file name date format of "_yyyy_MM_dd__HH_mm_ss___3___".
6) Navigate to the tracking preferences (Figure 2.12A,C-F):
    a) Set the tracking rate to 1 frame.
    b) Set the tracking delay to 330ms.
    c) Set the stage type to Zaber.
    d) Set the communication port correctly so that the stage responds to software commands.
    e) Set the software to move the stage using absolute coordinates.
    f) Set the software to check for stage responses with a timeout 5000ms.
    g) Set the software unit conversion to 20.997 steps/microns in both axes.
    h) Set the stage speed and acceleration to the highest values that do not cause stalling or vibration; alternatively, you may try setting the speed to 6000

(Zaber uses an arbitrary unit of measurement, roughly 0.45 microns/seconds) and the acceleration to 100 (Zaber uses an arbitrary unit of measurement, roughly 536 microns/seconds^2) to test whether these values work.

i) Set the stage's home location to its center at 25,000 microns in both axes.
j) Set the manual stage movement size to 250 microns in both axes.
k) Set the stage's rolling speed to 1000 microns/second in both axes.
l) Set the software to track by centroid.
m) Use a centroid tracking boundary of 200 microns in both axes.
n) Set the manual threshold to 95 (unsigned 8-bit pixel intensity).
o) Make sure the continuous auto threshold is turned off.
p) Make sure tracking is NOT inverted.

7) On the main screen, save the software configuration as your default.

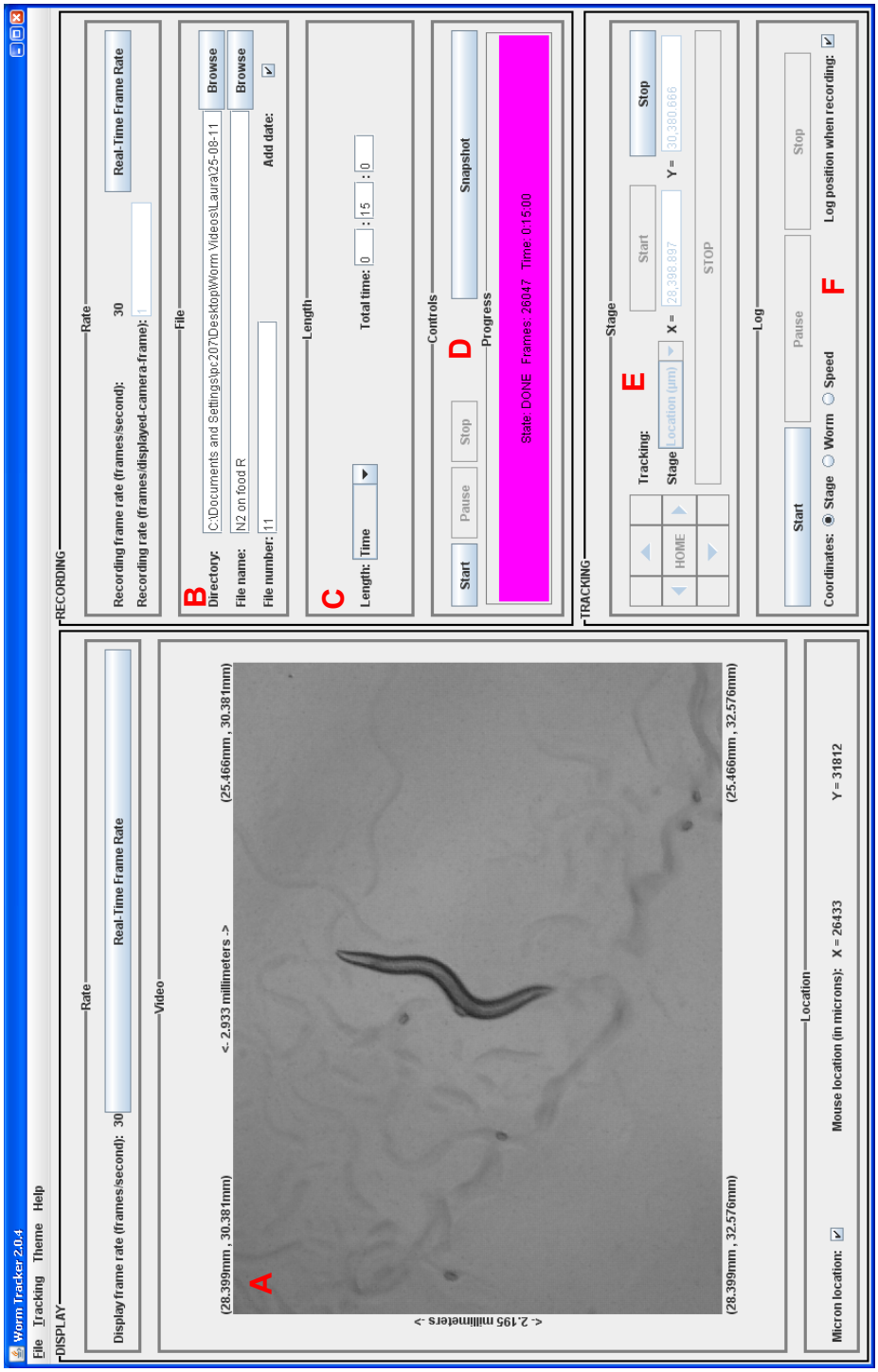


Figure 2.9 WT2 Main Screen

A. Incoming video is displayed alongside absolute coordinates and measurements. B. Experiment filename information. C. Recording length information. D. Recording controls. E. Motorized stage and tracking controls. F. Stage-movement logging controls.

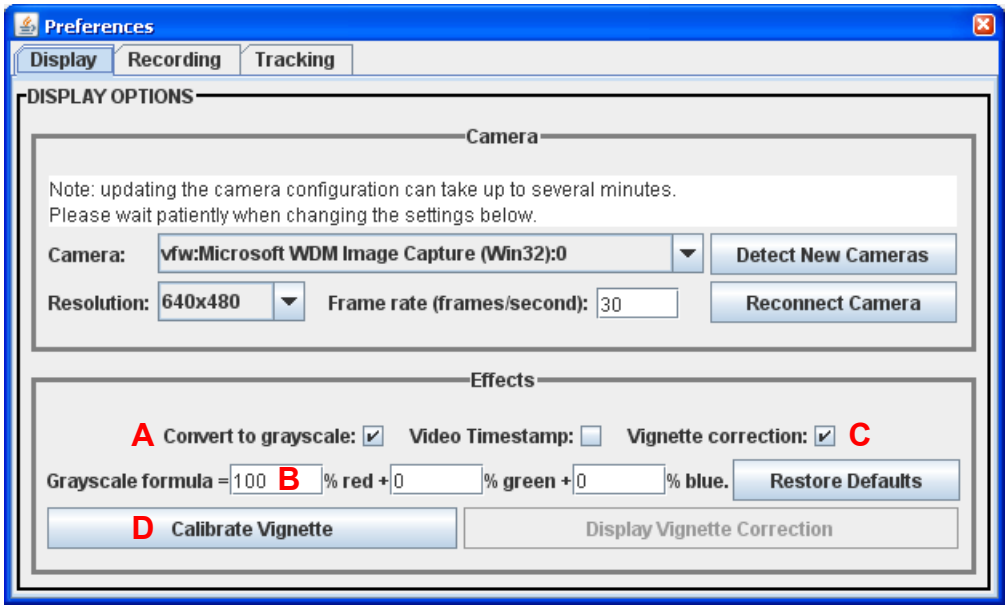


Figure 2.10: WT2 Display Preferences

A. Convert the video to grayscale. B. The grayscale formula for converting the red, green, and blue channel in a color video to a single grayscale image. C. Correct camera vignetting and/or dirt on the lens. D. Calibrate the vignette correction using the current video image.

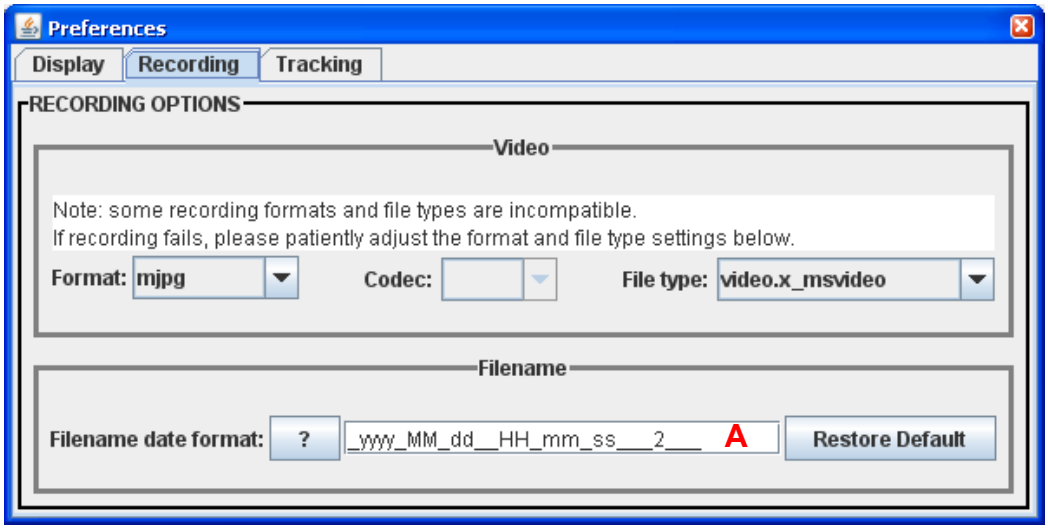


Figure 2.11: WT2 Recording Preferences

A. The date and time format appended to experiment file names.

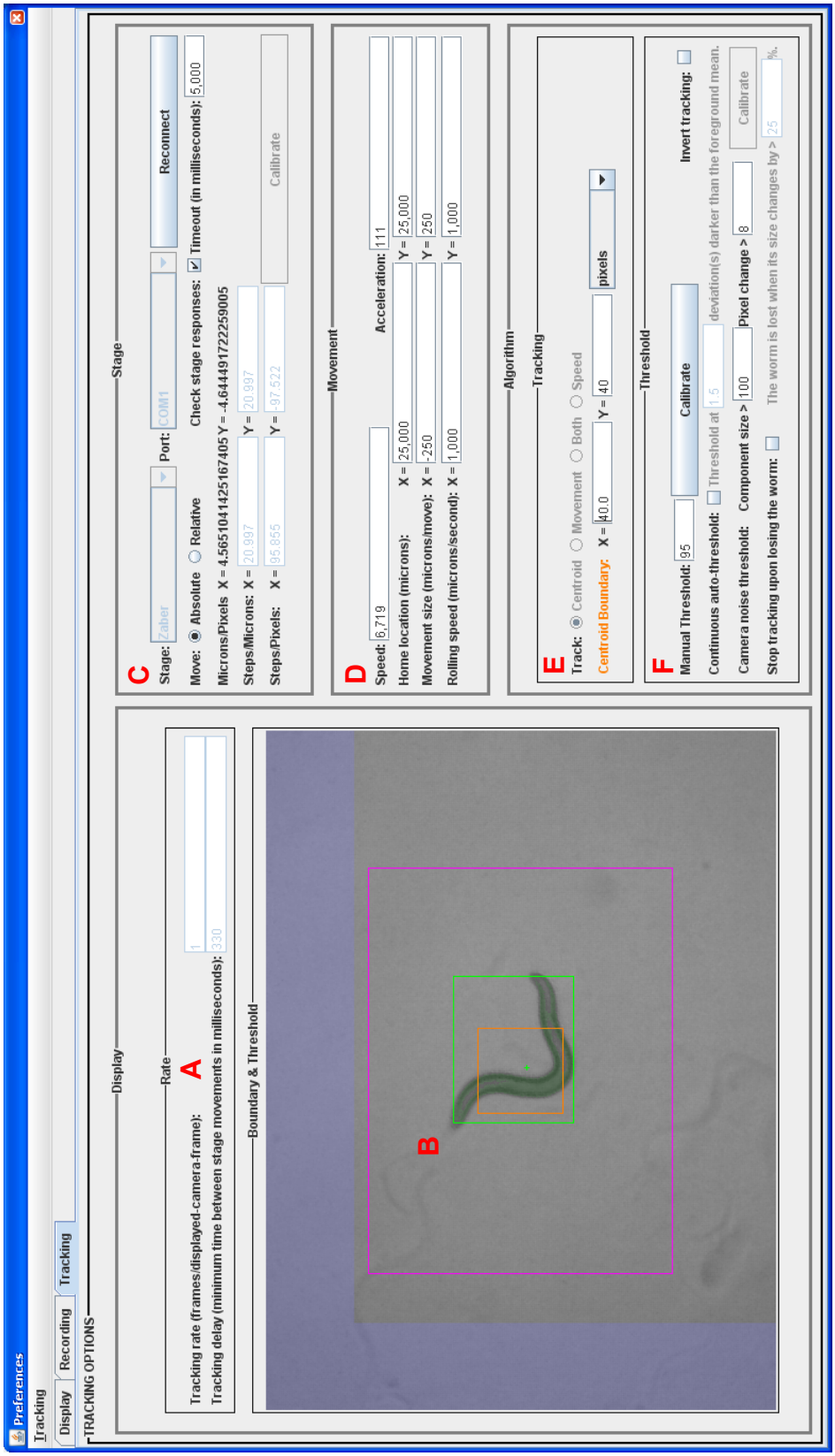


Figure 2.12: WT2 Tracking Preferences

A. The rate and delay for executing tracking stage movements. B. The tracked worm, centroid, and MER (painted green). The centroid (painted orange) and movement (painted magenta) boundaries. Newly tracked in areas (painted lavender). C. Motorized stage information. D. Stage movement settings. E. Tracking type (centroid and/or motion) information. F. Tracking threshold settings.

2.7.3.3.3 Software Calibration

The hardware and software must be calibrated before use. We recommend re-calibrating the equipment once weekly during periods of heavy use. The supplemental software guide provides further clarification, options, and troubleshooting.

2.7.3.3.3.1 Camera-Magnification Calibration

Timing: 5 minutes

1) Follow the "Worm and Plate Preparation" procedure to prepare several *C. elegans* young adults on one low-peptone NGM plate.
2) Place the plate from step 1 onto the worm platform.
3) Turn on the red light source and motorized stage.
4) Run the Worm Tracker 2.0 software.
5) Locate a worm on camera by adjusting the location of the worm plate and the height of the worm platform.
6) Adjust the camera magnification so that the worm's length is approximately ¾ the camera's width (Figure 2.9A).
7) Focus the worm by adjusting the worm platform height (Figure 2.8E,O).

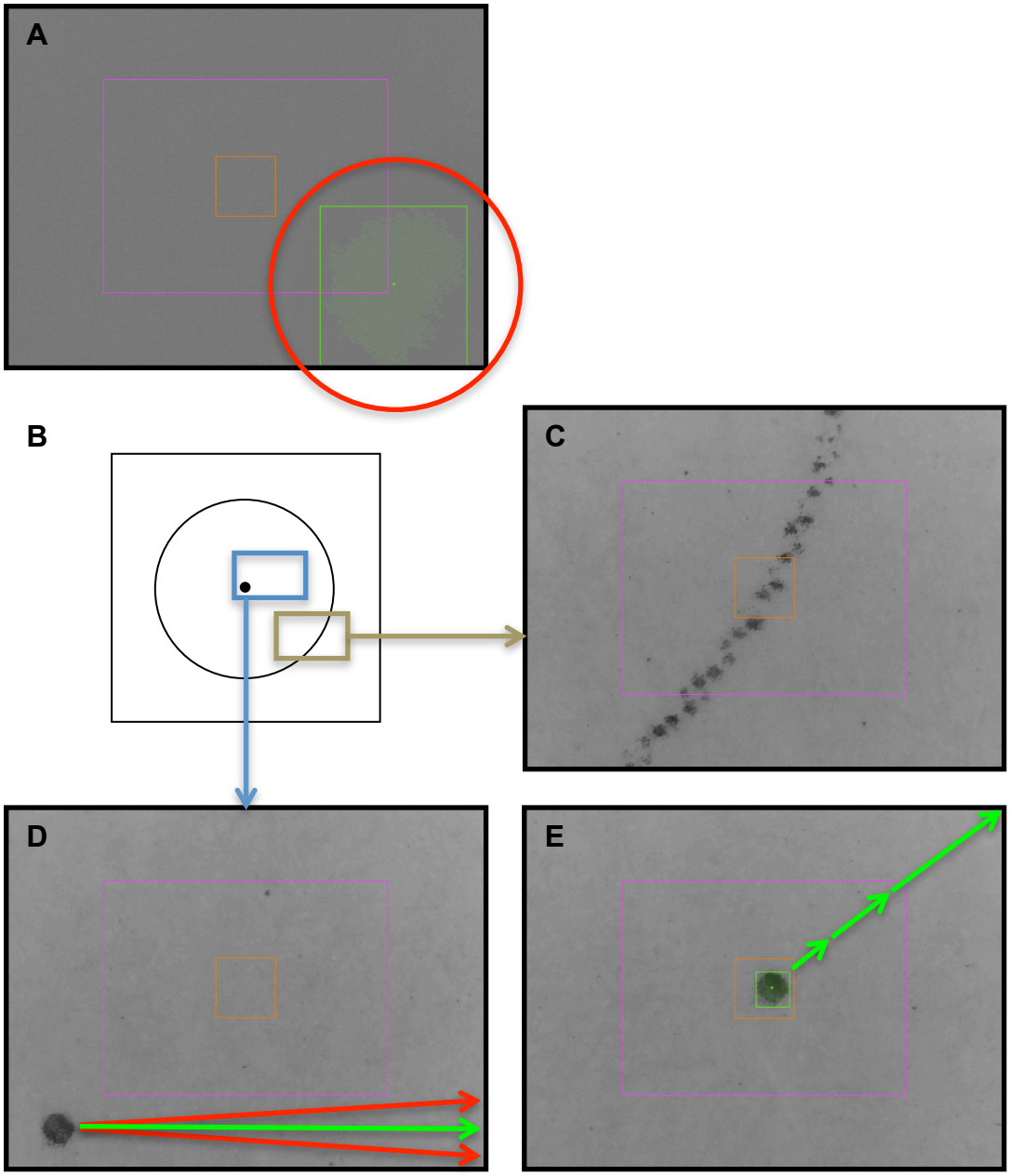


Figure 2.13: WT2 Calibration Screenshots (on the Tracking Preferences Screen)

A. The illumination, thresholded in green, must be centered. B. WT2 comes with a PDF of squares, containing circles, containing dots. A print out of the PDF is used for fine-scale calibration. C. The circle's edge, in the print out, serves as a guide to locate the small, central dot. D. The camera and stage must be aligned axially. X-axis stage movements must result in precise onscreen, axial dot movements (an example of axial dot movement is labeled green; the red vectors are not confined to a single axis and, therefore, indicate miscalibration). E. WT2 auto-calibrates the conversion between the stage's unit of steps and the camera's unit of pixels (an example of the auto-calibrating movements is shown in green). This conversion solves the conversion between onscreen pixels and real-world, micron coordinates.

2.7.3.3.3.2 Camera-Illumination Calibration

Timing: 10 minutes

1) Adjust the height of the Fresnel lens and opal diffuser so that the surface of the Fresnel lens is 90mm from the camera lens (Figure 2.8N,Q).
2) Adjust the height of the light source so that the surface of the LED lens is 140mm from the camera lens (Figure 2.8N,T).
3) Turn on the red light source and motorized stage.
4) Swivel the Fresnel lens and opal diffuser out of the way of the light source so that the light shines directly into the camera lens. Make sure there is nothing on the worm platform.
5) Run the Worm Tracker 2.0 software.
6) Navigate to the tracking preferences.
7) Make sure automatic thresholding is turned off. Adjust the manual threshold until the camera's vignette is painted green whereas the light source remains grayscale (Figure 2.13A).
8) Center the light source in the image. The vignette should be approximately equal at all four image edges. You may need to adjust the manual threshold, to see the vignette, throughout this procedure. Make sure the camera lens and light source are level and parallel to each other.
9) Swivel the Fresnel lens and opal diffuser back to center it underneath the light source. Make sure the Fresnel and opal diffuser are level and parallel to the camera lens and light source.
10) Make sure you have maintained the distances from steps 1 and 2.
11) Make sure the camera is not focused on anything. In the Worm Tracker 2.0 display preferences, turn on the vignette correction and calibrate the vignette.
12) Save the software configuration as your default.

2.7.3.3.3.3 Camera-Stage Axial Calibration

Timing: 10 minutes

1) Print the file "dots.pdf", located in your "Program Files\Worm Tracker\Documents" directory.

2) Cut out a square (Figure 2.13B) from the print out in step 1.
3) Place the square, from step 2, face down onto a 35mm Petri plate. Then place this plate onto the worm platform.
4) Turn on the red light source and motorized stage.
5) Run the Worm Tracker 2.0 software.
6) On the main screen, locate and focus on an edge on the outer circle of your print out (Figure 2.13C). If there is insufficient illumination, you may swivel the Fresnel lens and diffuser out of the way of the light source and/or turn on the camera's auxiliary white LEDs. Be sure to revert these changes when done.
7) Use the circle's outer edge to locate the central dot in your print out.
8) Use the software to move the dot. Twist the camera, in its holder, till the onscreen x and y axes match the software's manual, stage-movement commands.
9) Use the software to position the dot at one of the four onscreen corners.
10) Use the software to move the dot, solely along the x axis, to the opposite corner. If the dot changes its position in y axis, gently twist the camera, in its holder, to precisely align the camera and stage axes. Repeat this step until the software's manual, x-axis, stage-movement commands move the dot solely within the x axis (Figure 2.13D).

2.7.3.3.3.4 Microns/Pixels Calibration

Timing: 10 minutes

1) Continue from or repeat until step 6 of Camera-Stage Axial Calibration.
2) Navigate to the tracking preferences.
3) Adjust the manual, tracking threshold until the entire dot, and only the dot, is painted in green. Fine tune the manual threshold so that the dot's centroid position, indicated as a green "+", remains as stable as possible.
4) Use the software to position the dot in the center of the video image (Figure 2.13E).
5) Press the button to calibrate the steps/pixels and wait for the calibration to complete.

6) If the difference between the steps/pixels in both axes is more than 1, repeat the Camera-Stage Axial Calibration until the steps/pixels is sufficiently precise (i.e., the axial difference in steps/pixels is less than 1).
7) Repeat from step 4 until the steps/pixels are stable (i.e., they do not change by more than one from their previously calibrated values).
8) Save the software configuration as your default.

2.7.3.3.3.5 Worm Tracking Calibration

Timing: 5 minutes

1) Follow the "Worm and Plate Preparation" procedure to prepare several *C. elegans* young adults on individual low-peptone NGM plates.
2) Place a plate from step 1 onto the worm platform.
3) Turn on the red light source and motorized stage.
4) Run the Worm Tracker 2.0 software.
5) Locate a worm on camera by adjusting the location of the worm plate and the height of the worm platform.
6) Focus the worm by adjusting the worm platform height (Figure 2.8E,O).
7) Navigate to the tracking preferences.
8) Adjust the manual, tracking threshold until the worm, and only the worm, is painted in green (Figure 2.12B). Fine tune the manual threshold to paint as much of the worm as possible, without bleeding into the surrounding background of worm tracks, food, and agar.
9) You may attempt to lower the tracking delay from 330ms to achieve faster tracking. But, when tracking, if you experience chaotic stage movements during which the stage continuously fails to re-center the worm, you will need to raise the delay value.
10) Save the software configuration as your default.

2.7.3.4 Worm and Plate Preparation

All mutant strains require a wild-type, N2 control. Please maintain N2 worms as experimental controls. Social cues and environmental disturbances can

significantly affect *C. elegans* behavior. Therefore, great care is taken to control these effects.

20 or more worm videos, 15 minutes apiece, are required for each analyzed strain. To obviate localized environmental biases (e.g., temperature changes due to the time of day or positioning in the room), we recommend splitting strains across several days and interleaving experiment and control videos, throughout the day, on each tracker.

2.7.3.4.1 Maintaining Strains for Tracking

Timing: preparation 6 days, work 15 minutes

1) Use freshly poured NGM plates of 5mm or larger. Leave the plates to dry, for 2 days, with the lid on. You will need at least one plate per strain.
2) Place approximately 150µl of fresh *E. coli* OP50 into the center of each plate. Leave the plates to dry and culture, for 2 days, with the lid on.
3) Label the plates. Then transfer six adult *C. elegans*, of the appropriate strain, to each plate. We recommend using, at most, 1-week old plates (aged from the time of pouring). Older plates can provide a significantly different growth environment.
4) Maintain the plates at 22°C. Large temperature fluctuations can affect behavior. If you cannot control the temperature to within 2°C of the prescribed value, please use a 22°C incubator to maintain your strains.
5) Repeat from step 1, at least once and every 3 days, to maintain your strains and obviate social cues from densely populated plates.

2.7.3.4.2 Preparing L4 Worms for Tracking

Timing: preparation 3 days, work 15 minutes

1) Ensure that you have maintained the appropriate strain(s) following the "Maintaining Strains for Tracking" procedure (2.7.3.4.1). At least one generation of worms must be passaged, from birth to adulthood, according to this protocol.
2) 3 days prior to tracking, prepare freshly poured NGM plates of 5mm or larger. Leave the plates to dry, for 2 days, with the lid on. You will need one plate per 10

worms. We recommend preparing more plates than you need in the event that you experience problems with a portion of the plates and/or worms.

3) Place approximately 150µl of fresh *E. coli* OP50 into the center of each plate. Leave the plates to dry and culture, for 2 days, with the lid on.
4) Label the plates. Then, approximately 16-18 hours prior to tracking, transfer 10 L4 stage worms, from your maintained strains, onto a labeled plate. This controls for social cues beyond the final molt into adulthood. We recommend using, at most, 1-week old plates (aged from the time of pouring). Older plates can provide a significantly different environment.

2.7.3.4.3 Preparing Low-Peptone NGM Plates for Tracking

Timing: preparation 2 days, work 15 minutes

1) 2 days prior to tracking, prepare freshly poured 35mm Low-Peptone NGM plates. Leave the plates to dry, for 2 days, with the lid on. These Low-Peptone NGM plates produce thinner food lawns that result in clearer worm videos. You will need at least one plate per tracked worm. We recommend preparing more plates than you need in the event that you experience problems with a portion of the plates and/or worms.
2) Approximately 1 hour prior to recording, place exactly 20µl of fresh *E. coli* OP50 into the center of each plate. Leave the plates to dry, for 30 minutes, with the lid on. We recommend using only 2-day old plates (aged from the time of pouring). Older plates can provide a significantly different environment. We recommend using plates within, at most, 3 hours of food transfer to control for bacterial growth and ensure thin lawns.

2.7.3.4.4 Preparing Young Adult Worms for Recording

Timing: 15 minutes

1) Turn on the Worm Tracker 2.0 hardware and start up its software.
2) The 10 L4 worms, maintained on the NGM plate, should now have grown into young adults.

3) Approximately 30 minutes prior to recording, label the Low-Peptone NGM plates on the sidewall. The plates are labeled on their sidewall so as not to obfuscate worm image on camera. Transfer one young adult onto each individual, labeled, Low-Peptone NGM plate. Place the worm approximately within the center of its food lawn. We recommend using only plates seeded 30 minutes prior with *E. coli*. Older seeding will lead to thick food lawns and poor imaging. Younger seeding will lead to wet food lawns and a significantly different environment.
4) If you cannot resolve the worm's vulval side on camera, use a microscope to make this determination.

2.7.3.5 Video Collection

All mutant strains require a wild-type, N2 control. Caution must be observed to match every mutant video with an N2 control video performed in similar conditions. Social cues and environmental disturbances can significantly affect *C. elegans* behavior.

20 worm or more videos, 15 minutes apiece, are required for each analyzed strain. To obviate localized environmental biases (e.g., temperature changes due to the time of day or positioning in the room), we recommend splitting strains across several days and interleaving experiment and control videos, throughout the day, on each tracker.

2.7.3.5.1 Recording a Worm

Timing: preparation 1 hour, work 15 minutes

Caution must be observed to maintain the worm plate at a temperature of 22°.

1) The young adult must habituate to the recent environmental disturbances. Therefore, wait 30 minutes from the point at which the worm was moved onto its plate before moving the plate onto the tracker and beginning the recording.
2) During the wait, ensure that the Worm Tracker 2.0 software is set up correctly. Check that the recording time is set to 15 minutes. Check that the software is set

to log stage coordinates and that logging is set to synchronize with recording. Check that you are recording video to the desired disk location. Check that the timestamp is being appended to the video file name. Finally, enter the video file name for the worm using the following convention, "*gene strain food vulva*", where:

a) *gene* = "N2" for wild-type worms or the mutant gene name (e.g., "unc-1"); if the worm has more than one mutation, the mutant gene names are appended with a ";" separating them (e.g., "unc-1;cat-1").
b) *strain* = empty for wild-type worms; otherwise, the worm strain.
c) *food* = "on food" or "off food" to identify whether the worm plate was seeded with food.
d) *vulva* = "R" if the vulva is clockwise from the worm's head, "L" if anti-clockwise, and "X" if the vulval side cannot be identified or if it flipped during the experiment.

3) After 30 minutes, when the wait for habituation is over, move the worm plate to the tracker. Locate the worm and begin tracking. Then, begin recording.
4) After 15 minutes, when the recording is complete, check the worm's vulval side. If the vulval side has flipped, please amend the recorded video file name.
5) Repeat from step 1 with a wild-type, N2 control.

2.7.3.6 Tracking Troubleshooting

2.7.3.6.1 Video Problems

2.7.3.6.1.1 Problem: the video is not displayed.

- ***Possible reason***: the camera is disconnected.
  Solution: connect the camera to the computer.

- ***Possible reason***: the camera software is incorrectly installed.
  Solution: re-install the camera software and ensure the DinoCapture software displays video.

- ***Possible reason***: the Java software was updated.
  Solution: locate "Program Files\Worm Tracker\Prerequisites\JMF\jmf-2_1_1e-windows-i586.exe". Run it to uninstall then reinstall the Java Media Framework (JMF).

2.7.3.6.1.2 Problem: the worm cannot be found.

- ***Possible reason***: the agar surface is not in focus.
  Solution: adjust the worm platform height to locate the agar surface. When the worm is located, focus on the worm. If the agar surface cannot be focused, you may need to flip the worm plate so that it faces the camera, bringing the camera closer to the worm.

- ***Possible reason***: the worm is far from the onscreen location.
  Solution: find the worm's tracks and follow them to the worm. Or, remove the plate and use a microscope or locate the worm by eye. Note the location. Place the plate back onto the center of the platform and navigate the camera to the worm.

- ***Possible reason***: the worm has crawled to an edge or off the plate.
  Solution: prepare another worm plate or use a spare one.

2.7.3.6.1.3 Problem: the video is blurry.

- ***Possible reason***: the camera lens is dirty.
  Solution: use lens paper to clean the camera lens.

- ***Possible reason***: the worm platform is dirty or scratched.
  Solution: clean the worm platform or replace it.

- ***Possible reason***: the agar is not transparent.

Solution: review the low-peptone NGM plate recipe. If the agar remains cloudy, you may need to flip your worm plates so that their agar sides face the camera.

2.7.3.6.1.4 Problem: the video has poor contrast.

- ***Possible reason***: the worm is out of focus.
  Solution: adjust the worm platform height to focus the worm.

- ***Possible reason***: the food is too thick.
  Solution: follow the instructions in the Worm and Plate Preparation procedure. If the food is on the plate for too long and/or if there is too much peptone in the agar, the food will grow too thick.

- ***Possible reason***: the camera's auxiliary white LEDs are on.
  Solution: use the DinoCapture software to turn off the auxiliary white LEDs. Then save the configuration. Quit the DinoCapture software.

- ***Possible reason***: the red light source is off or too weak.
  Solution: make sure the red light source is on and being driven at its maximum power, 1400ma at 2.95V.

- ***Possible reason***: the light source and/or Fresnel lens and diffuser are incorrectly placed.
  Solution: follow the instructions in the Camera-Illumination Calibration procedure. You may need to adjust the height of the light source and/or Fresnel lens and diffuser to achieve better contrast.

2.7.3.6.1.5 Problem: the recorded video has a low frame rate and/or drops frames.

- ***Possible reason***: the computer's processor is overload.
  Solution: quit all applications but Worm Tracker 2.0.

- ***Possible reason***: the camera is using a fixed exposure.
  Solution: use the DinoCapture software's real time image settings to turn on auto exposure. Then save the configuration.

- ***Possible reason***: grayscale conversion is too processor intensive.
  Solution: turn off grayscale conversion in the Worm Tracker 2.0 display preferences. Then save the configuration.

- ***Possible reason***: the light source is too weak.
  Solution: make sure the light source is being driven at its maximum power, 1400ma at 2.95V.

2.7.3.6.2 Stage Problems

2.7.3.6.2.1 Problem: the stage does not respond to the software.

- ***Possible reason***: the stage was disconnected and/or off.
  Solution: connect the stage to the computer and turn the stage on. Then, reconnect the stage in the Worm Tracker 2.0 tracking preferences.

- ***Possible reason***: the stage's communication port is incorrect.
  Solution: set the stage's communication port, correctly, in the Worm Tracker 2.0 tracking preferences.

- ***Possible reason***: one or both of the stage's actuator knobs in the non-neutral position (the LED on the knob will flash red).
  Solution: center the stage's actuator knobs in the neutral position (the LED on the knob will turn green). Then, reconnect the stage in the Worm Tracker 2.0 tracking preferences.

2.7.3.6.2.2 Problem: the stage cannot travel to its minimum and/or maximum extension.

- ***Possible reason***: the stage is blocked.
  Solution: free the stage and the equipment above it from any cords and surrounding lab materials. Ensure the worm platform is not in the way.

- ***Possible reason***: the stage is broken.
  Solution: contact the manufacturer to arrange for a fix or replacement.

2.7.3.6.2.3 Problem: the stage reached its minimum and/or maximum extension.

- ***Possible reason***: the worm plate is larger than 35mm.
  Solution: follow the instructions in the Worm and Plate Preparation procedure.

- ***Possible reason***: the worm platform and/or plate are not centered.
  Solution: center the worm platform at the stage's center of travel. Center the worm plate on the worm platform.

2.7.3.6.2.4 Problem: the stage vibrates.

- ***Possible reason***: the stage speed and/or acceleration are too high.
  Solution: lower the stage speed and/or acceleration in the Worm Tracker 2.0 tracking preferences.

- ***Possible reason***: the hardware is loose.
  Solution: tighten the hardware screws.

- ***Possible reason***: nearby equipment is vibrating the Worm Tracker hardware.
  Solution: move the equipment further away and/or add more dampening underneath the Worm Tracker hardware.

2.7.3.6.3 Tracking Problems

2.7.3.6.3.1 Problem: the tracker does not follow the worm.

- ***Possible reason***: the manual, tracking threshold is incorrect.
  Solution: follow the instructions in the Worm Tracking Calibration procedure.

- ***Possible reason***: the tracking delay is too high.
  Solution: lower the tracking delay in the Worm Tracker 2.0 tracking preferences.

- ***Possible reason***: the stage speed and/or acceleration are too low.
  Solution: raise the stage speed and/or acceleration in the Worm Tracker 2.0 tracking preferences.

- ***Possible reason***: the camera magnification is too high.
  Solution: lower the camera magnification. Then follow the instructions in the Microns/Pixels Calibration procedure.

- ***Possible reason***: the worm has crawled to an edge or off the plate.
  Solution: prepare another worm plate or use a spare one.

- ***Possible reason***: there is more than one worm on the plate.
  Solution: follow the instructions in the Worm and Plate Preparation procedure.

2.7.3.6.3.2 Problem: the tracker is moving chaotically.

- ***Possible reason***: the tracking delay is too low.
  Solution: raise the tracking delay in the Worm Tracker 2.0 tracking preferences.

- ***Possible reason***: the software is confusing the vignette with the worm.
  Solution: correct the vignette in the Worm Tracker 2.0 display preferences. Then follow the instructions in the Worm Tracking Calibration procedure.

2.7.3.7 Performing an Analysis

All analysis functions run within Matlab.

2.7.3.7.1 Segment a Video

Timing: 30 minutes

1) Locate the video for segmentation.
2) Run video2Diff to differentiate the video for stage movement identification.
3) Run saveWormFrames to segment the video (extract the worm contour from each video frame – please refer to Chapter 3 for details) and save each frame's worm information.

2.7.3.7.2 Correct the Head, Tail, and Vulva Location

Timing: 1-15 minutes

1) Use the Chunk Tool to review the video and correct any errors in the head, tail, and vulva location.

2.7.3.7.3 Behavioral Analysis and Results

Timing: 1 minute

1) Load the analysis file from saveWormFrames. The analysis file contains basic worm information, for each frame, including the contour, skeleton, angles, length, widths, body areas, coloring, and orientation. Matlab provides a wealth of functions to further analyze and display this data.
2) Use multiScaleWorm to differentiate the basic worm information at multiple scales (e.g., to compute velocity and foraging).

2.7.3.8 Analysis Troubleshooting

2.7.3.8.1 Problem: findStageMovement fails to identify the stage movements.

- ***Possible reason***: the video has a recording error.
  Solution: correct the video. Then rerun video2Diff and saveWormFrames.

- ***Possible reason***: findStageMovement missed a stage movement.
  Solution: load the video2Diff output file. Use findStageMovement to locate the missing stage movement frame(s). Set the frame(s) differential to the maximum value of all the differentials.

- ***Possible reason***: findStageMovement mistook a worm movement or stage vibration for a stage movement.
  Solution: load the video2Diff output file. Use findStageMovement to locate the mistaken stage movement frame(s). Set the frame(s) differential to the minimum value of all the differentials.

- ***Possible reason***: findStageMovement mistook two or more stage movements for one stage movement.
  Solution: load the video2Diff output file. Use findStageMovement to locate the mistaken stage movement frame(s). Set the frame(s) differential, at the intervals between the stage movements, to the minimum value of all the differentials.

- ***Possible reason***: the worm was lost.
  Solution: remove the end portion of the video and the stage movement log at the point where the worm was lost.

- ***Possible reason***: the tracker moved chaotically.
  Solution: follow the instruction in the Tracking Troubleshooting guide to correct the tracker problem. Then follow the Video Collection procedure to re-record the video.

2.7.3.8.2 Problem: segWorm fails to segment a large portion of the video

- ***Possible reason***: the video's vignette correction is miscalibrated or missing.
  Solution: follow the instruction in Camera-Illumination Calibration procedure to calibrate the vignette for future video collection. Run the video2Vignette analysis function to compute the vignette correction for your problematic video. Then rerun saveWormFrames.

- ***Possible reason***: the food is too thick.
  Solution: follow the instructions in the Worm and Plate Preparation procedure. If the food is on the plate for too long and/or if there is too much peptone in the agar, the food will grow too thick.

- ***Possible reason***: the worm was lost.
  Solution: remove the end portion of the video and the stage movement log at the point where the worm was lost.

- ***Possible reason***: the worm is coiled or touching another object.
  Solution: the analysis software cannot segment under these conditions. If you require more data, follow the Video Collection procedure to collect more videos.

2.8 Bibliography

1. Buckingham, S. D. & Sattelle, D. B. Strategies for automated analysis of C. elegans locomotion. *Invert Neurosci* **8**, 121–131 (2008).
2. Dusenbery, D. B. Video camera-computer tracking of nematode Caenorhabditis elegans to record behavioral responses. *Journal of chemical ecology* **11**, 1239–1247 (1985).
3. Simonetta, S. H. & Golombek, D. A. An automated tracking system for Caenorhabditis elegans locomotor behavior and circadian studies application. *Journal of Neuroscience Methods* **161**, 273–280 (2007).
4. Baek, J.-H., Cosman, P., Feng, Z., Silver, J. & Schafer, W. R. Using machine vision to analyze and classify Caenorhabditis elegans behavioral phenotypes quantitatively. *Journal of Neuroscience Methods* **118**, 9–21 (2002).
5. Feng, Z., Cronin, C. J., Wittig, J. H., Sternberg, P. W. & Schafer, W. R. An imaging system for standardized quantitative analysis of C. elegans behavior. *BMC Bioinformatics* **5**, 115 (2004).
6. Geng, W., Cosman, P., Berry, C. C., Feng, Z. & Schafer, W. R. Automatic tracking, feature extraction and classification of C elegans phenotypes. *IEEE Trans. Biomed. Eng.* **51**, 1811–1820 (2004).
7. Geng, W., Cosman, P., Palm, M. & Schafer, W. R. C. elegans egg-laying detection and behavior study using image analysis. *EUROSIP J. Appl. Signal Processing* **2005**, 2229–2240
8. Huang, K.-M., Cosman, P. & Schafer, W. R. Machine vision based detection of omega bends and reversals in C. elegans. *Journal of Neuroscience Methods* **158**, 323–336 (2006).
9. Clark, D. A., Gabel, C. V., Gabel, H. & Samuel, A. D. T. Temporal activity patterns in thermosensory neurons of freely moving Caenorhabditis elegans encode spatial thermal gradients. *Journal of Neuroscience* **27**, 6083–6090 (2007).
10. Leifer, A. M., Fang-Yen, C., Gershow, M., Alkema, M. J. & Samuel, A. D. T. Optogenetic manipulation of neural activity in freely moving Caenorhabditis elegans. *Nat Meth* **8**, 147–152 (2011).
11. Stirman, J. N. *et al.* Real-time multimodal optical control of neurons and muscles in freely behaving Caenorhabditis elegans. *Nat Meth* **8**, 153–158 (2011).
12. Faumont, S. *et al.* An image-free opto-mechanical system for creating virtual environments and imaging neuronal activity in freely moving Caenorhabditis elegans. *PLoS ONE* **6**, e24666 (2011).
13. Ramot, D., Johnson, B. E., Berry, T. L., Carnell, L. & Goodman, M. B. The Parallel Worm Tracker: a platform for measuring average speed and drug-induced paralysis in nematodes. *PLoS ONE* **3**, e2208 (2008).
14. Swierczek, N. A., Giles, A. C., Rankin, C. H. & Kerr, R. A. High-throughput behavioral analysis in C. elegans. *Nat Meth* **8**, 592–598 (2011).
15. Zhang, S. *et al.* Profiling a Caenorhabditis elegans behavioral parametric dataset with a supervised K-means clustering algorithm identifies genetic networks regulating locomotion. *Journal of Neuroscience Methods* **197**, 315–323 (2011).
16. Hoshi, K. & Shingai, R. Computer-driven automatic identification of locomotion states in Caenorhabditis elegans. *Journal of Neuroscience Methods*

157, 355–363 (2006).
17. Wang, W., Sun, Y., Dixon, S. J., Alexander, M. & Roy, P. J. An Automated Micropositioning System for Investigating C. elegans Locomotive Behavior. *Journal of the Association for Laboratory Automation* **14**, 269–276 (2009).
18. Tsechpenakis, G., Bianchi, L., Metaxas, D. & Driscoll, M. A novel computational approach for simultaneous tracking and feature extraction of C. elegans populations in fluid environments. *IEEE Trans. Biomed. Eng.* **55**, 1539–1549 (2008).
19. Edwards, S. L. *et al.* A novel molecular solution for ultraviolet light detection in Caenorhabditis elegans. *Plos Biol* **6**, e198 (2008).
20. Stephens, G. J., Johnson-Kerner, B., Bialek, W. & Ryu, W. S. Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* **4**, e1000028 (2008).
21. Cronin, C. J., Feng, Z. & Schafer, W. R. Automated imaging of C. elegans behavior. *Methods Mol. Biol.* **351**, 241–251 (2006).
22. Harvey, S. C. & Orbidans, H. E. All Eggs Are Not Equal: The Maternal Environment Affects Progeny Reproduction and Developmental Fate in Caenorhabditis elegans. *PLoS ONE* **6**, e25840 (2011).
23. Huang, K.-M., Cosman, P. & Schafer, W. R. Automated detection and analysis of foraging behavior in Caenorhabditis elegans. *Journal of Neuroscience Methods* **171**, 153–164 (2008).
24. Wu, K., Otoo, E. & Suzuki, K. Two strategies to speed up connected component labeling algorithms. (2008).
25. Otsu, N. A threshold selection method from gray-level histograms. *Automatica* **11**, 23–27 (1975).
26. Kamath, R. S. *et al.* Systematic functional analysis of the Caenorhabditis elegans genome using RNAi. *Nature* **421**, 231–237 (2003).
27. Brenner, S. The genetics of Caenorhabditis elegans. *Genetics* **77**, 71–94 (1974).

Chapter 3 : Worm Analysis

3.1 Abstract

In the previous chapter, I introduced a new single-worm tracker that permits recording videos within difficult environments for all sizes and stages of *Caenorhabditis elegans* development (from hatching to adulthood). The removal of these environmental and size restrictions presents challenges to existing algorithms for video extraction and analysis of recorded worms. Moreover, our aim to perform high-throughput, single-worm tracking necessitates a robust, reliable, user-friendly analysis package to accompany the tracking software. I here report a suite of new algorithms to extract and analyze worms from these difficult videos. Among other refinements, my new algorithms are scale-invariant, handling worms of all sizes (as long as they can be resolved within the image). I use adaptive techniques to locate worms within difficult images and temper them with checks to eliminate inaccurate worm shapes. I end with a discussion of new tools that permit users to evaluate the quality of worm segmentation and features measurements. These tools provide a method to automate worm analysis then verify its fidelity. The Worm Analysis Toolbox presents a user-friendly GUI (designed by Tadas Jucikas) to access all these tools within a single program.

3.2 Introduction

One of the main benefits of single-worm tracking is high-resolution, video images that permit detailed phenotypic analysis. Automated measurement of worm shape and features, from these high-resolution videos, obviates the errors and substantial time requirements associated with analogous manual measurements. Moreover, for large videos, the sheer size of the data set may preclude human measurement of individual video frames. Therefore, considerable effort has been made to automate measurement of worm tracking videos.

Traditionally, automated measurement is split into several consecutive steps shown in Figure 3.1. In each video frame, the worm is extracted from its background in a process called segmentation. The worm's body is reduced to a skeleton representation, often accompanied by the contour as well. Thereafter, some algorithms identify the worm's head and tail although, this step is not always performed. Finally, the sequence of segmented worms provides a base set of measurements which are then used to compute a feature set. This feature set includes per-frame measurements, such as the amplitude and frequency associated with a static posture, as well as multi-frame, behavioral time-series descriptions like velocity and pirouettes.

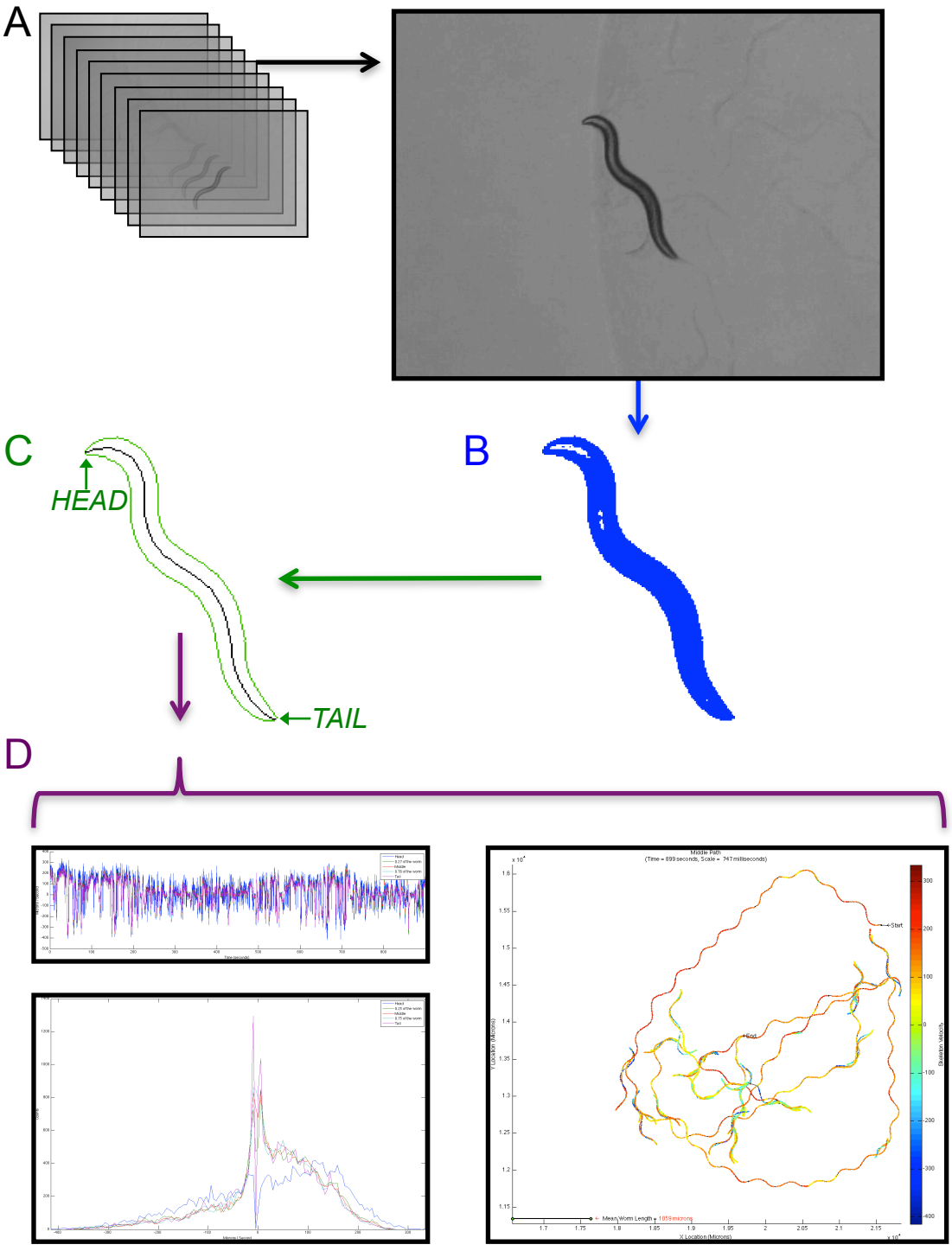


Figure 3.1: Automated Worm Measurement

Traditionally, several common steps take place to analyze a worm within a video sequence. A. The video is analyzed frame by frame. B. In each frame, the worm is segmented from its background. C. The segmented worm is reduced to a representative contour (labeled in green) and skeleton (labeled in black). Some algorithms determine the head and tail although, this step is not always performed. D. The sequence of segmented worms provides a base set of measurements which are then used to compute a feature set. The plots on left are a time series (x = time in minutes, y = speed in mm/s) above a histogram (x = speed in mm/s, y = probability). The plot on the right is a path trace (x and y = location in mm, color = speed in mm/s from backwards in blue to forwards in red).

There are many computer vision algorithms for segmentation and skeletonization. One reason for this diversity is that small changes in image features such as scene complexity, image quality, and background noise often require radically different approaches to segment and skeletonize. Moreover, many algorithms present significant trade offs between accuracy and speed. Consequently, differences in worm size, image quality, and experimental conditions have meant that worm trackers tend to be tightly coupled to their analysis software. Also, as a result of these various constraints, the majority of single-worm trackers work solely with adult worms in limited to no-food environments. There exists no single-worm tracking software to handle varying larval stages, nor one for high-resolution videos of an unconfined, swimming worm. Therefore, the primary challenge considered by this chapter is developing segmentation and skeletonization algorithms that can analyze broad scenarios of worm image quality, larval stages, worm motions and background noise (e.g., food-related image effects); specifically, in tandem with the tracking hardware detailed within the previous chapter.

The current state-of-the-art solution for phenotyping *C. elegans* is a single-worm tracker (a combination of hardware and software) originally published by Baek et al. in 2002 [1]. Since then, the system has seen several major additions and has recently been used, by Zhang et al. [2], to characterize 32 worm strains represented by 1991 4-minute videos, at 10Hz. While this volume of data represents a record for the field, it is a small amount when compared to profiling over 20,000 genes present in *C. elegans* for which there are, presently, more than half a million mutant strains available from the Caenorhabditis Genetics Center (who also supply strains from the *C. elegans* Gene Knockout Consortium), approximately 4,000 gene knockouts available from the Japanese National BioResource Project, and RNAi feeding libraries capable of knocking down expression for about 86% of the worm genome [3]. Five other, less popular, single-worm trackers have been used to collect much smaller data sets with a reduced quantity of analyzed measurements and features [4-8]. Therefore, another challenge considered by this chapter is developing worm image analysis algorithms and systems that are capable of scaling for much larger data sets, higher resolution and longer videos, and support searchable large scale databases of worm videos.

The large volume of worm genetics and available, representative, mutant strains demands a high-throughput method for phenotypic quantification. Moreover, the subtlety of behavioral differences, among many mutants, requires high precision and accuracy from this methodology. Unfortunately, current systems are not adept to these high-throughput needs. Beyond their expensive price tags, which generally preclude buying multiple units, their analysis suites suffer several major limitations. In my adaptation of published algorithms for use with our own video collection I found their techniques often produce inaccuracies at the head and tail. Among other reasons, this is due to the use of worm thinning algorithms that leave false branches at the endpoints and shorten the skeleton prematurely (before reaching the tip of the head and/or tail) or, the use of a popular alternative, which computes a spline interpolation to estimate the entire skeleton. Some examples of these inaccuracies can be seen directly within figures accompanying the original publications [9-11]. Specifically, in a paper published by Geng et al. in 2004, figure 6C displays a skeleton that ends prematurely before reaching the tip of the head and a false branch that resulted from thinning (corrected through pruning in figure 6D); later, in figure 8B, the skeleton ends prematurely at the tail and does not faithfully trace the midline of the worm midbody. In a paper published by Cronin et al. in 2005, the skeletons displayed never reach the head nor tail tips of their worms. Lastly, in a paper published by Roussel et al. in 2007, figure 2B displays a spline-fit worm wherein the skeleton is slightly off from the true midline of the contour.

Further limitations exist in automated measurements and tools for quick manual correction of large volumes of data. For example, the Baek tracker automatically determines the head and tail, within its videos, to extract some of the most sensitive phenotypic information available for worms. Tests of its algorithm show 98% correct assignment [9]. With this percentage, if the Zhang data set were to grow by one order of magnitude, from 32 to 320 worms, nearly one million video frames would have incorrect assignments. Unfortunately, the associated tool for reviewing and correcting automated measurements only displays and alters head-tail assignments, and requires that each frame be checked and dealt with individually. The only other tracker which performs automatic head and tail detection, published by Hoshi and Shingai in 2006 [5], requires VCR tapes, which are increasingly difficult to obtain and represent a cumbersome method for data collection and manual review.

Moreover, their technique offers no notable improvements over the Baek method save for slightly better head detection that may well derive from restricting the experimental environment to worms off food.

I report here several advances that permit accurate and precise measurements of worm shapes for a range of experimental conditions including off and on food as well as liquid environments (used for swimming assays), enable quick verification of automated measurement, and, therefore, scale for high-throughput worm phenotyping. I introduce a novel, scale-invariant, technique which permits us to analyze the great variety of worm mutants at all larval stages with only a minimal requirement for pixels representing the worm. I present a new skeletonization algorithm that faithfully translates the midline of a worm body. Our head and tail detection refines previously published methods by employing a classifier which can be trained, as necessary, to handle dissimilar worms in a multitude of environments (e.g., dauers off food as opposed to young adults in deep food lawns). For quick data review and verification by eye, I introduce a new technique wherein videos are overlayed with their automated measurements of contour, skeleton, head, and ventral side. A separate tool, created by James Scott-Brown, enables one to flip head-tail assignments for groups of video frames alongside the ability to reject poorly segmented worm images that evaded automated checks. Lastly, as will be discussed in Chapter 4, a feature viewer provides users with the ability to explore phenotypic measurements, per frame, coupled to the worm segmentations that generated them, all accompanied by statistical overviews of the feature data, per worm.

3.3 Materials and Methods

3.3.1 Worm Segmentation & Measurement

3.3.1.1 Scale invariance

Previous segmentation algorithms constrain worm size to a small, well-defined range in order to ease the task of segmentation. High-throughput tracking, across the full range of larval stages, requires the removal of size constraints and, therefore, an algorithm for scale-invariant segmentation. To perform scale-invariant segmentation, Worms were segmented relative to their physiological scales as shown in Figure 3.2. The worm's head and tail each occupy around 1/6 of its body from their respective contour ends (Figure 3.2D). Furthermore, *C. elegans* have 23 to 24 pairs of, almost equally-spaced, muscle segments along each side of their segmented contour (Figure 3.2A). Due to muscle overlap and coordinated innervation as well as physical constraints of the cuticle, the degrees of freedom in worm motion from muscle contractions is over fit at 24. Therefore, when sampling each side of the contour and the skeleton I often choose a fraction of these 24 muscles to provide scale invariance to our computation.

Intuitively, the magnitude of worm body bending should increase relative to the sharpness of the angle represented on its contour and skeleton. Sharp angles should have a large magnitude of bending whereas flat surfaces should be zero. Therefore, I use supplementary angles along the worm body, to measure bending. For example, the angle at a worm's head may measure 45° and therefore have a supplementary angle, for bending, of 135°. I define the worm's high-frequency bending as the supplementary angle to the one formed between two muscle segments at each point along its contour and skeleton (Figure 3.2B). This high-frequency bending angle resolves the head and tail accurately while mostly ignoring small imperfections and noise that plague the contour. I define the worm's low-frequency bending, at half the sampling rate of the high frequency, as the supplementary angle to the one formed by vertices two muscle segments away in each contour direction (Figure 3.2C). This low-frequency bending angle resolves locomotory bends along the worm body as well as the approximate locations for the head and tail.

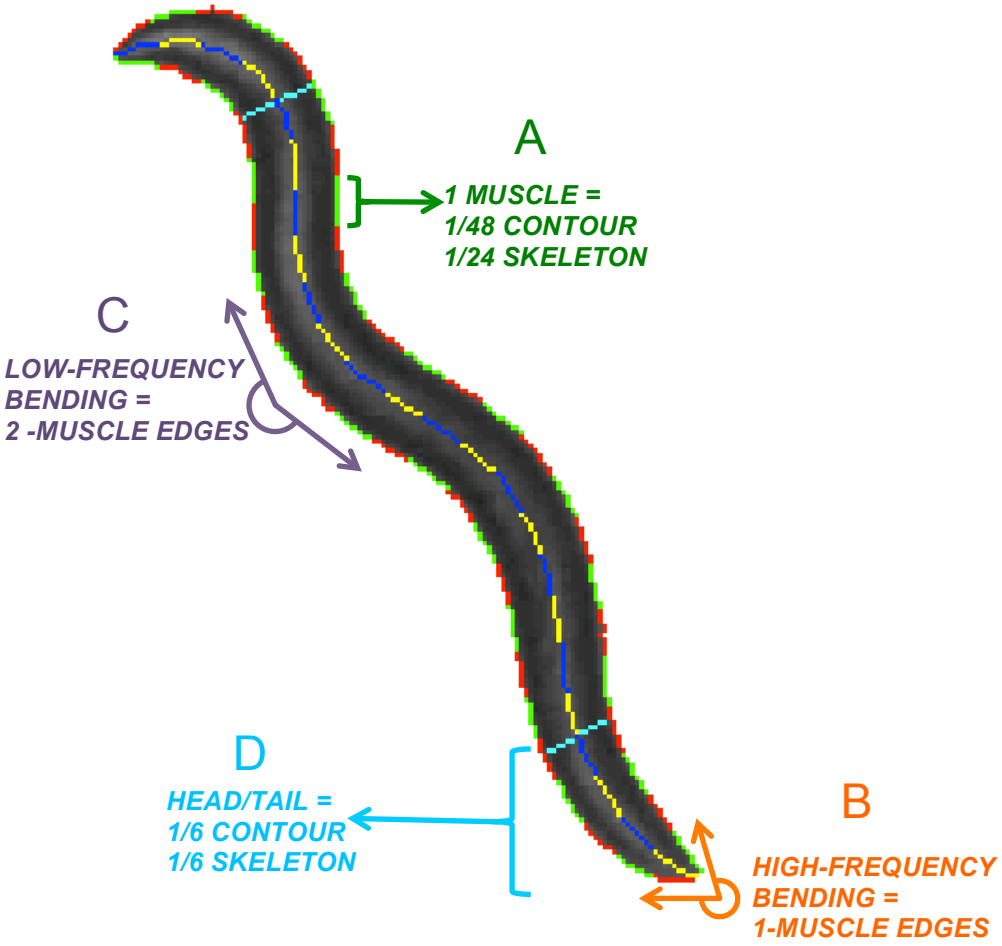


Figure 3.2: Scale-Invariant Segmentation

Scale-invariant, physiologically-based measures, permit flexible segmentation of various worm sizes and shapes. A. *C. elegans* have 23-24 pairs of, almost equally-spaced, muscle segments along each side of their segmented contour and, equivalently, along their skeleton. B. I define the worm's high-frequency bending as the supplementary angle to the one formed between two muscle segments at every point along its contour and skeleton. C. I define the worm's low-frequency bending, at half the sampling rate of the high frequency, as the supplementary angle to the one formed by vertices two muscle segments away in both contour directions. D. The worm's head and tail each occupy, approximately, 1/6 of its body from their respective contour ends. To complete the boundary, I use the skeleton point at 1/6 distance from the end, and connect it to the nearest contour point on either side.

On the contour, both bends are signed such that convex angles (e.g., the head, tail, and outer sections of a bend) are positive and concave angles (e.g., the inside sections of a bend) are negative. On the skeleton, the bending is signed so that it matches the contour's ventral side. When finding peaks within the bending, peaks are defined as the local maxima or minima within a window of, at least, one muscle segment for high-frequency bending and two muscle segments for low-frequency bending. The worm contour is a closed path and, therefore, has a bend angle at every point. The skeleton is an open path and, therefore, its bending cannot be computed at the end points where insufficient data is present to measure angle vertices.

3.3.1.2 Pixelized (Chain-Code) Length Estimation

Pixelized images have non-Euclidean geometry. Laterally and longitudinally adjacent pixels can be measured as one pixel away from each other. Diagonally adjacent pixels would therefore be $\sqrt{2}$ pixels away. Therefore, when computing the length along sections of the worm contour and skeleton, I use the aforementioned metric, known as the Freeman chain code length estimator [12] or taxicab metric, to achieve a more accurate representation for the true length of the digitized worm. Further to this point, fractional pixels are used for worm measurements to achieve subpixel resolution.

3.3.1.3 Segmentation of the Worm Contour

To maintain worm equivalence between the tracking and analysis software, I use a similar and, often, identical algorithm to segment the worm body in both programs. Chapter 2 reviews this algorithm in detail. A summary, shown in Figure 3.3, is as follows:

1) Recorded video images are converted to 8-bit grayscale.
2) If a vignette correction was saved by the tracking software, it is applied to each image.

3) Each image is assumed to be a bimodal distribution of worm and background pixels. The Otsu method [13] is used to obtain a threshold separating both modalities. This threshold is applied to identify worm pixels.
4) A fast connected-components algorithm [14] is used to discover the largest 8-connected foreground pixel set. Two pixels are considered 8-connected if they touch each other at any of the eight possible vertices. The largest connected component is considered the worm.
5) The perimeter of the worm is traced, clockwise, to extract its contour. Later, when measuring contour bending, this clockwise trace allows us to sign contour angles as concave and convex. The contour is anti aliased to remove stair-step patterns that elongate the trace.

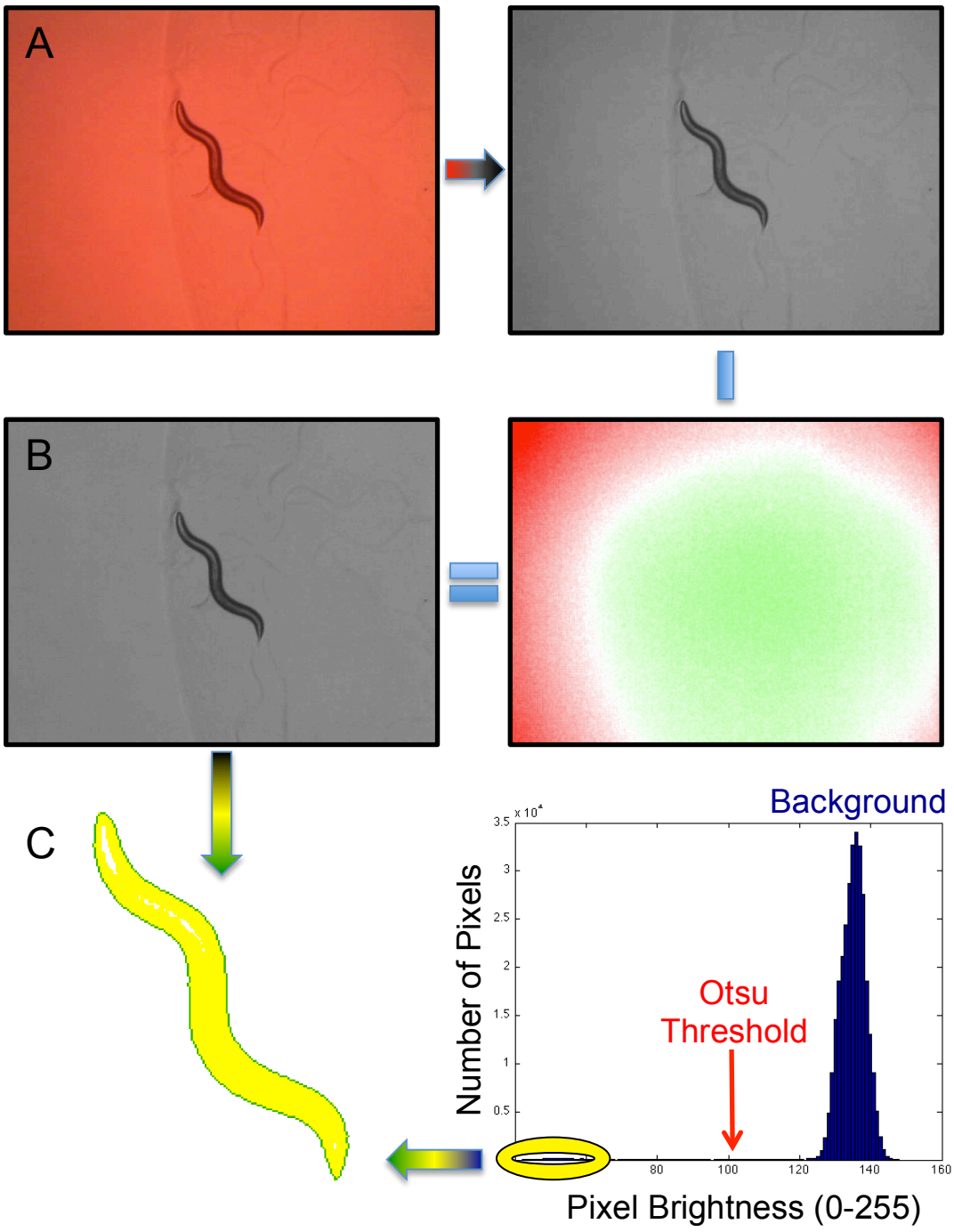


Figure 3.3: Worm Segmentation

An illustration of the worm segmentation algorithm. A. Each video frame is converted to grayscale. B. If a vignette correction exists, it is applied to each image. In the vignette graphic, correction for dark pixels is labeled in red. Correction for bright pixels is labeled in green. Color intensity corresponds to the magnitude of correction. C. Each image is assumed to be a bimodal distribution of worm (labeled in yellow) and background (labeled in blue) pixels. The Otsu method is used to separate both modalities and threshold the worm. The worm is identified as the largest 8-connected component and its perimeter is traced to extract the contour (labeled in green).

In high-resolution images, employing bright-field illumination, the worm's internal head and tail sections are often much brighter in intensity than the rest of its body. As a result the threshold may fail to identify them as foreground, worm pixels. When this happens, the worm contour forms holes and invaginations that split the head and tail. I use a novel algorithm, to extract the worm contour, that is unaffected by holes and heals split ends. The algorithm, shown in Figure 3.4, is as follows:

1) The contour's high-frequency bending is computed. At this sampling frequency the head and tail are represented by sharp convex peaks with angles much higher than 60°. Invaginations, on the other hand, are represented by sharp concave peaks with angles much lower than -90°. The remaining worm body has much shallower angle magnitudes.
2) Pairs of sharp, high-frequency convex peaks that are closer, in distance, than two muscle segments are labeled as potential splits and ordered by distance from shortest to longest.
3) While the worm has more than two sharp convex peaks (the head and tail), each potential split is checked, in ascending order of the distance between its peaks. A real split is separated by the worm contour on one side (potentially containing another split) and the invaginated contour on the other side. The worm contour is much longer than its invaginated counterpart. Moreover the invaginated contour will contain a sharp, high-frequency, concave peak. Using these criteria, each split is identified and healed, by connecting its convex peaks to remove the invaginated contour. Healing is performed until at least two sharp convex peaks (the head and tail) are left.
4) When segmented, if the contour has less than 48 representative pixels, it is below the minimum sampling scale resulting in a rejected segmentation.

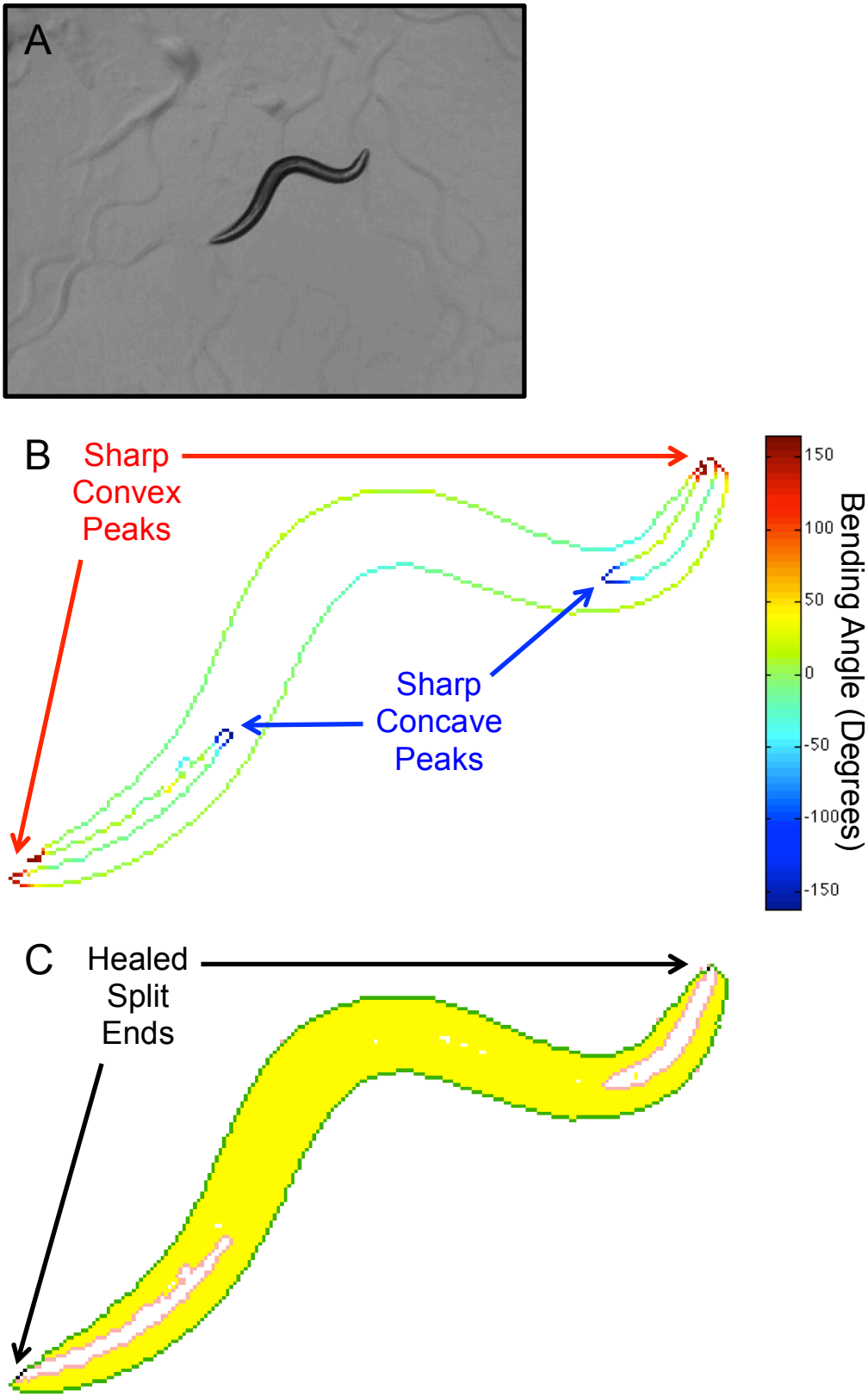


Figure 3.4: Healing Split Ends

An illustration of the algorithm to heal contour splits. A. The worm has poorly-defined edges near its bright head and tail. B. Consequently, the contour is split at both ends. Measuring the high-frequency bending identifies four sharp concave peaks and two sharp convex peaks. C. Connecting nearby convex peaks, split by a concave peak, heals both split ends.

3.3.1.4 Head and Tail Localization

With high-resolution worms in flexible environments, noise can confound localization of the head and tail. Therefore, I use a new algorithm to locate these two endpoints on the contour. Basically, I match sharp convex peaks from the low and high frequency bending, to localize the head and tail on opposing sides of the contour. The algorithm, shown in Figure 3.5, is as follows:

1) The head and tail of the worm are represented by sharp convex peaks on opposite sides of the contour. As stated, at high-frequency bending angles, the head and tail points form sharp convex peaks much higher than 60°. At low-frequency bending angles, the head and tail points form sharp convex peaks much higher than 90°.
2) To eliminate noise present in the high-frequency bending, the values at each point are smoothed. Smoothing is done by convolving each bending value with a uniform distribution of all its neighbors within a quarter muscle segment distance away. The resulting smoothing preserves most bending values while eliminating high-frequency noise.
3) If the smoothed high-frequency bending identifies less than two possible head and tail peaks or the low-frequency bending identifies more than two possible head and tail peaks, the contour is problematic resulting in a rejected segmentation. This occurs when the worm coils, lays an egg, or is otherwise obscured. The low-frequency bending can have less than 2, appropriately sized, head and tail peaks when the worm has a sharp bend, near either end, that decreases the convexity of the respective peak angle below 90°.
4) If the low-frequency bending identifies two peaks, their nearest smoothed high-frequency counterparts are chosen as accurate representations of the head and tail.
5) Otherwise, the two smoothed high-frequency peaks that are furthest from each other, on the contour, are chosen as the head and tail.

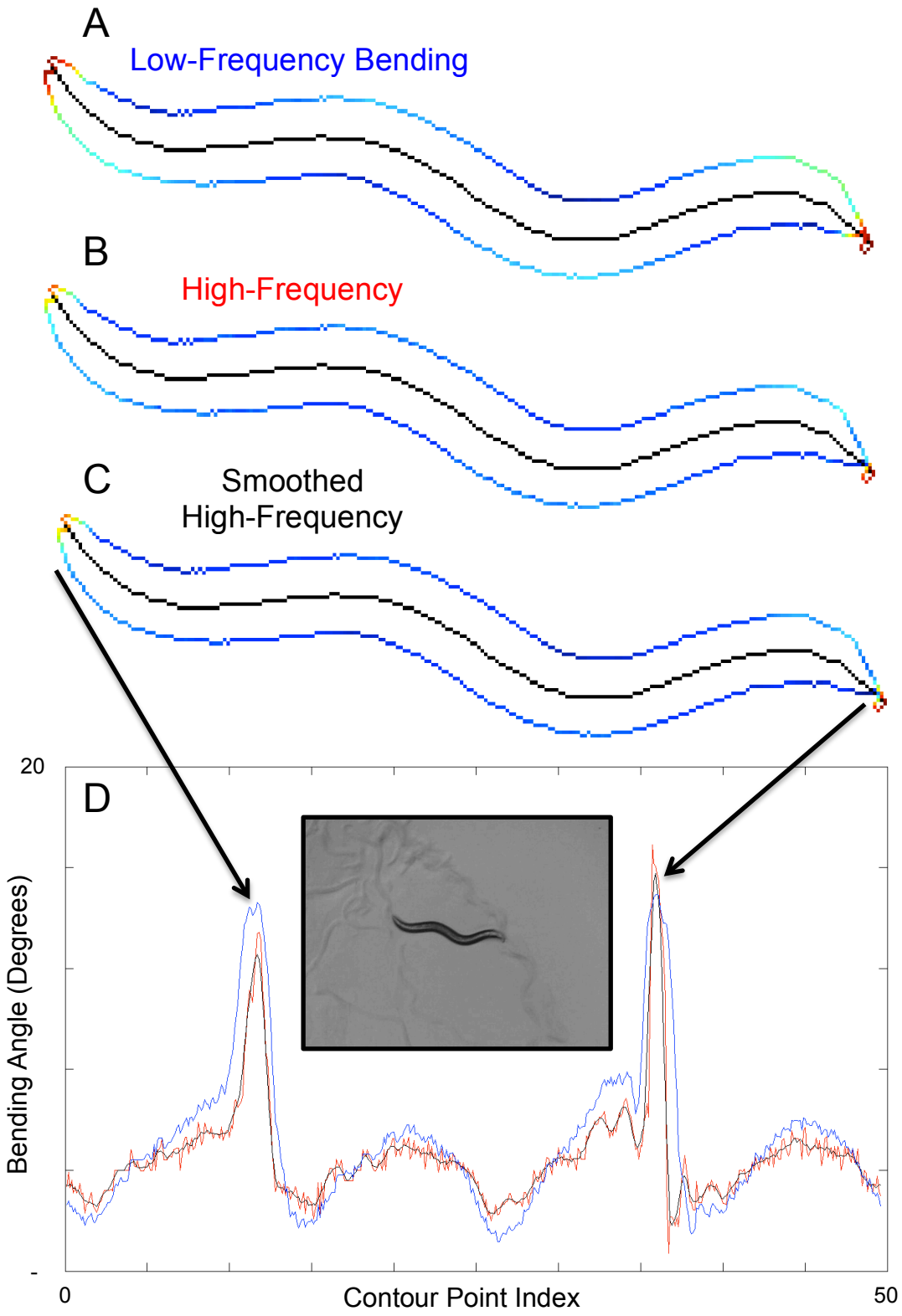


Figure 3.5: Head and Tail Localization

An illustration of the head and tail localization algorithm. The contour's bending is represented by a heat map. Small bends are colored in cool, blue tones. Conversely, large bends are colored by hot, red tones. The plot at the bottom of the figure provides a more detailed view of the bending values. A. The low-frequency bending (plotted, at the bottom, in blue) identifies the global locations for the head and tail. B. The high-frequency bending (plotted, at the bottom, in red) is more accurate but subject to noise. For example, at this scale, distortions in the contour from worm tracks, an egg, or dirt may appear as potential head and tail candidates. In this figure illustration, the head has a small concave depression. C. The smoothed, high-frequency bending (plotted, at the bottom, in black) provides a less noisy and more accurate locality for the head and tail. The chosen locations are slightly shifted from those identified by the low-frequency peaks.

3.3.1.5 Segmentation Checks

With flexible imaging and experimental conditions, occasionally, noisy and obscured video frames can result in poor worm segmentation. To eliminate these cases, I perform several checks to validate the worm shape. These checks, shown in Figure 3.6, are as follows:

1) As stated earlier, the worm contour must have only two peaks of sharp convex bending, representing the head and tail. Otherwise, the worm may be coiled, touching an egg, or otherwise obscured and, therefore, is improperly segmented.
2) The worm must not touch the image boundary. Otherwise, we may have lost a portion of its body from the image or confused it with the vignette.
3) If either side of the worm is twice as long as the opposing side, the worm is coiled or obscured and, therefore, is improperly segmented.
4) In deep food, the head and tail may be significantly obscured and collapse in their contour. Even healing split ends will not solve this issue. When the head and tail have inappropriate widths or volumes, the segmentation is discarded.

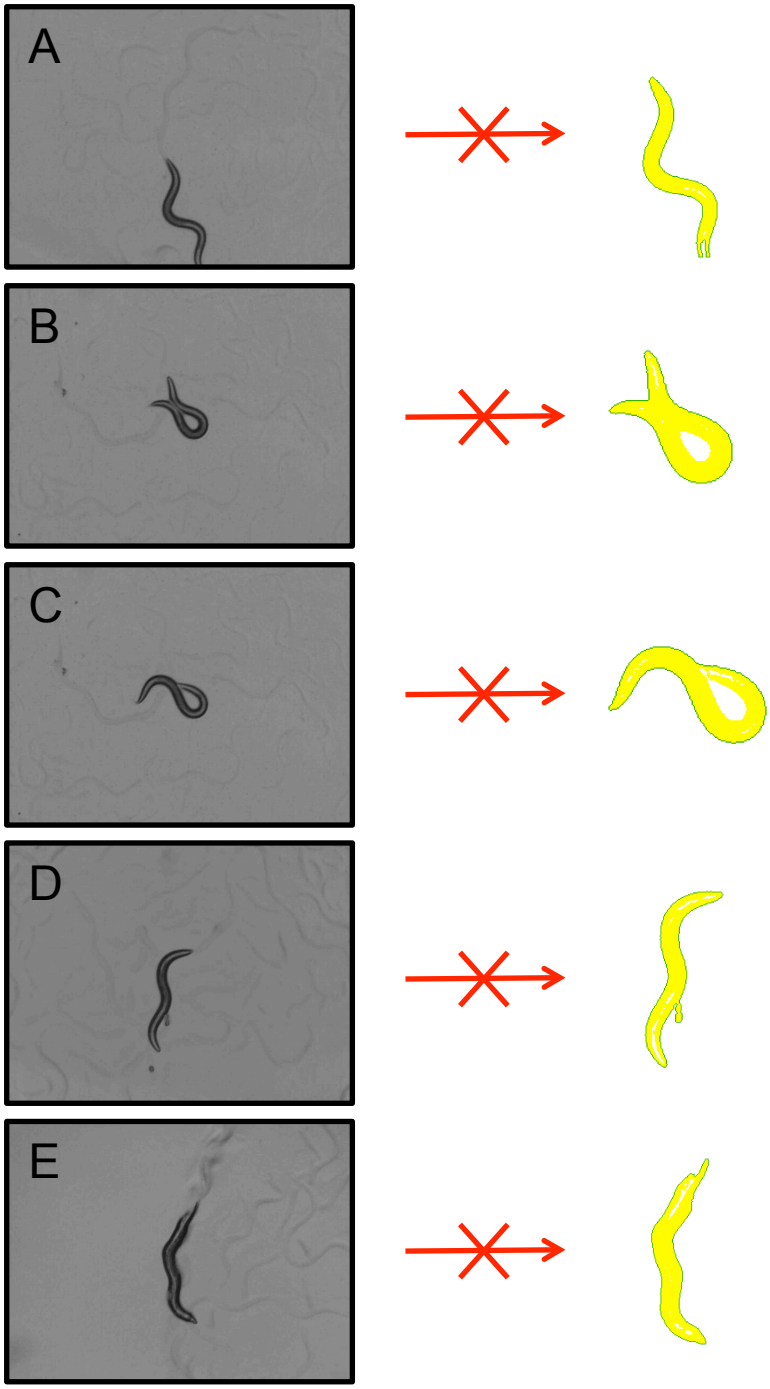


Figure 3.6: Worm Segmentation Checks

Several checks are performed to validate the worm shape. A. The worm must not touch the image boundary. B. The distance, between the head and tail on either side of the contour, must not exceed twice the distance of the opposing side. C-D. The worm contour must have exactly two peaks of sharp convex bending, representing the head and tail. E. The head and tail cannot collapse. They must have appropriate volumes relative to each other as well as the rest of the body.

3.3.1.6 Skeletonization

Skeletonization presents a simplified and more compact view of the worm body. The worm skeleton facilitates computations such as those involved in locomotion (e.g., velocity) that are otherwise difficult to calculate using the contour directly. Previous algorithms often have strict requirements for imaging conditions and may misrepresent the skeleton when confronting condition permitted by the WT2 software. To achieve precise skeletons, within our wide-ranging imaging conditions, I implement a novel approach to skeletonization. Basically, I walk along both sides of the contour, matching opposing pixels, and labeling their midpoints as the skeleton. The algorithm, shown in Figure 3.7, is as follows:

1) Worm bends can cause the skeleton to drift towards the inside of the bend. Therefore, the worm is divided into sections at its midbody and bends to avoid this distortion. These bends are identified among the low-frequency bending angles as peaks, within a corresponding low-frequency window (1/12 the worm), which exceed 20° in magnitude. Thereafter, sections are formed by connecting these bends to their nearest point on the opposing side of the contour.
2) For each section, we begin at its center on both sides of the contour. We then walk, pixel by pixel, in either direction until we hit the end of the section on opposite sides, for both directions. The midpoint, between each opposing pixel pair, is considered the skeleton and the distance between these pixel pairs is considered the width for each skeleton point.
3) Food tracks, noise, and other disturbances can form spikes on the worm contour. When no spikes are present, our walk attempts to minimize the width between opposing pairs of pixels. When a spike is present, this strategy may cause one side to get stuck in the spike while the opposing side walks. Therefore, when a spike is present, the spiked side walks while the other side remains still.
4) When done, the skeleton is ordinarily connected between each section. In the rare event that the skeleton has a small gap, the two ends are connected using linear interpolation and the gap widths are chosen as the minimum width of both ends.

5) Overlapping sections of skeleton are removed and its chain of pixels is anti aliased to achieve the shortest-path representation.

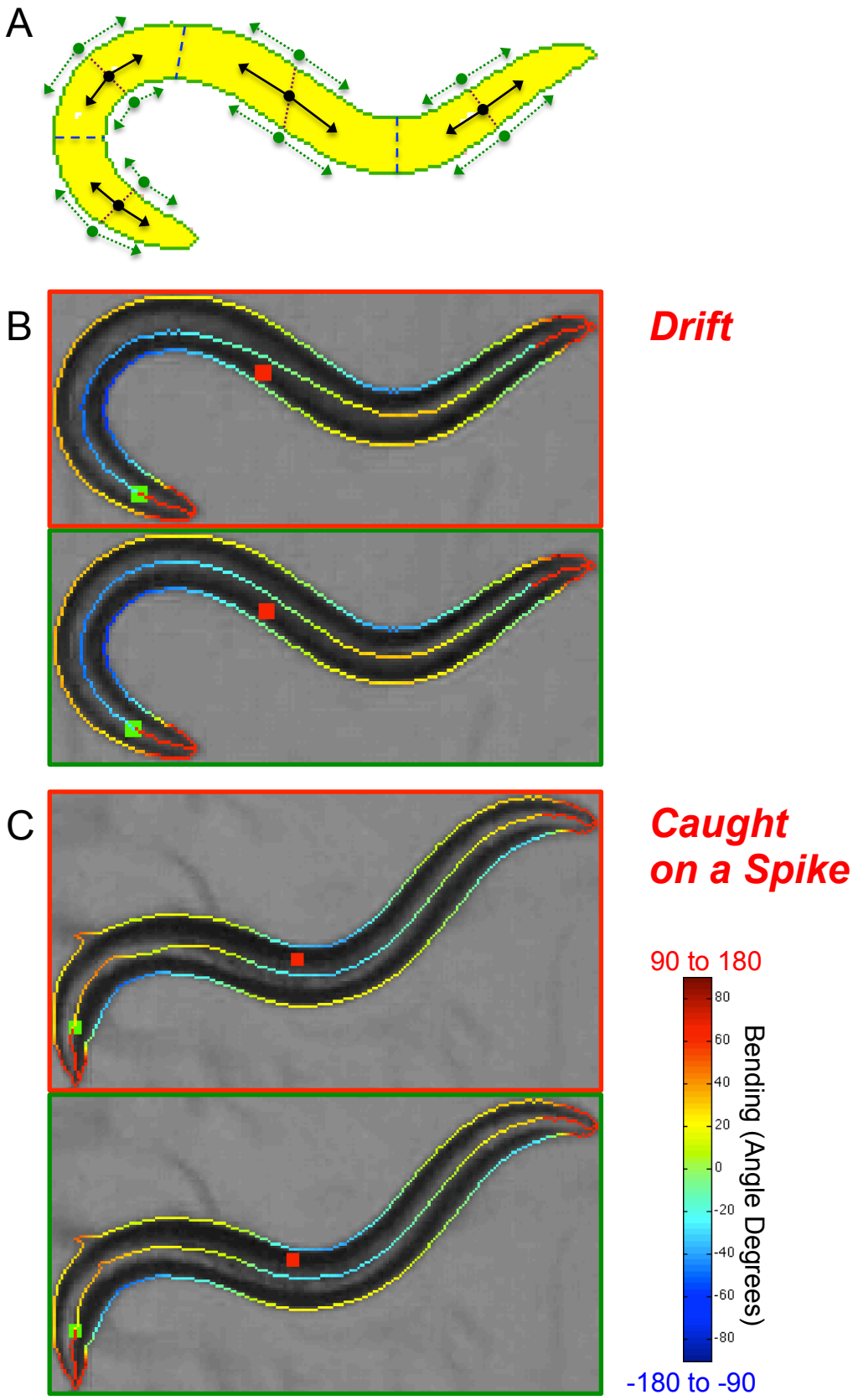


Figure 3.7: Skeletonization

An illustration of the skeletonization algorithm. A. The worm is divided into sections at its midbody and bends (labeled in blue dashed lines). The bends represent peak bending angles, exceeding 20° in magnitude, within the worm contour's low-frequency bending angles. The algorithm starts at the center of each section (labeled in purple dotted lines) on opposing sides of the contour. We then walk, pixel by pixel, in either direction (labeled by the green arrows) until we hit the end of the section on opposite sides, for both directions. The midpoint (labeled by the black arrows), between each opposing pixel pair, is considered the skeleton and the distance between these pixel pairs is considered the width for each skeleton point. The skeleton sections are joined to form a continuous skeleton. B. Tracing the skeleton, without sectioning at its bends, causes the skeleton to drift towards the inside of a bend. This outcome is shown, framed in red. The drift is a result of unequal path lengths on either side of the contour. This occurs even when scaling the length of both sides, relative to each other, since the path inequities are confined to only small sections of the contour. The result of our skeletonization algorithm, with sectioning at the worm bends, is shown below, framed in green. C. Tracing the skeleton, using width minimization to dictate which contour side should walk, causes one side to get caught on a spike while the other side continues walking. The outcome, a small depression in the skeleton, is shown framed in red. The result of our skeletonization algorithm is shown below, framed in green. The spike on our skeleton is representative of the segmented contour. Contour spikes occur infrequently. Regardless, smoothing algorithms, such as spline interpolation, can be used in post-processing to de-noise the contour and skeleton, thereby, removing these artifacts.

3.3.2 Head and Tail Detection in Video Chunks

3.3.2.1 Chunking Video and Orienting Worms Within a Chunk

Once segmentation is complete, the head and tail must be determined in each frame. A worm's head and tail can be distinguished by their pixel intensities, movement, and the convex angle of their contour [5,9]. Clearly the aggregate statistics for a video are more reliable than those per frame. In order to measure aggregate statistics, the worm shapes in each frame must be aligned so that they share the same head-to-tail orientation and, therefore, the endpoints (head and tail) match up. At 15Hz and above, worm movement is confined to small foraging motions and even smaller locomotory behaviors. Consequently, at these high frame rates, worm shapes in adjacent frames can be easily aligned to share the same head-to-tail orientation. Unfortunately, stage movements, rejected segmentations, and even dropped video frames can interrupt the continuity of worm shapes. These interruptions can lead to long gaps between segmented worms. Therefore, I subdivide each video into individual chunks wherein all worm shapes are aligned to share the same head-to-tail orientation. The algorithm, shown in Figure 3.8, is as follows:

1) The first worm in a chunk (which has a random orientation) sets the orientation for all subsequent worms within the chunk. For each pair of subsequent worms, I define worm 1 as the shape at the earlier time and worm 2 as the shape at the later time.
2) If worm 1 and worm 2 are more than 1.5 seconds away. A new chunk begins with worm 2.
3) If worm 1 and 2 are at most 0.25 seconds away, the worm cannot have moved very far. To orient worm 2 relative to worm 1 I use the following algorithm:
    a. 6 points are chosen at identical locations on both worm skeletons. These points are, also, symmetrically spaced from the worms' head and tail.
    b. The distance between corresponding points, on both worms, is measured with worm 2's segmented orientation and, once again, flipping worm 2's head-to-tail orientation.

   c. If more point-to-point distances are minimized in the flipped orientation, worm 2 is flipped.

4) If worm 1 and 2 are at most 0.25 seconds away and a stage movement intervened, I align the worms at their centroids, to remove the displacement, before proceeding with step 3.
5) If worm 1 and worm 2 are between 0.25 and 1.5 seconds away, an omega bend may have intervened and flipped the worm's head-to-tail orientation. I use a novel algorithm to distinguish this case:
   a. An omega bend requires a sharp, concave bend along the worm contour. The high-frequency angle of such bends is much lower than -75°.
   b. If both worms have high-frequency sharp, concave bends, I choose the nearest skeleton point, on each worm, to represent their bends. I define these as their bend points.
   c. If both worms' bend points are less than two muscle segments distance from each other, the worm must have followed a path through this bend.
   d. I compute the worm's path through the bend as the tangent to the bend point. The tangent is formed by points, on the skeleton, one muscle segment in either direction of the bend point.
   e. If both worms have nearly orthogonal tangent vectors, conservatively estimated as a difference of more than 67.5°, the worm's path through the bend cannot have been an omega turn. Under these circumstances, the simple algorithm in step 3 will orient worm 2.
   f. If worm 2 shares the same head-to-tail tangent vector as worm 1, its orientation is preserved. Otherwise, worm 2's head-to-tail orientation is flipped.
   g. The algorithm is not restricted to omega bends. Any worm locomotion, within the specified time period, that preserves a sharp bend and triggers this orientation procedure, will result in a correctly oriented worm.
6) If worm 1 and worm 2 are between 0.25 and 1.5 seconds away, and a stage movement intervened, I convert the worms to absolute coordinates, to remove the displacement, before checking for an omega bend in step 5.

7) If worm 1 and worm 2 are more than 0.5 seconds away, and an omega bend did not intervene, I act conservatively and begin a new chunk with worm 2.
8) Otherwise, if worm 1 and worm 2 are more than 0.25 seconds away, and an omega bend did not intervene, I use step 3 to orient worm 2.

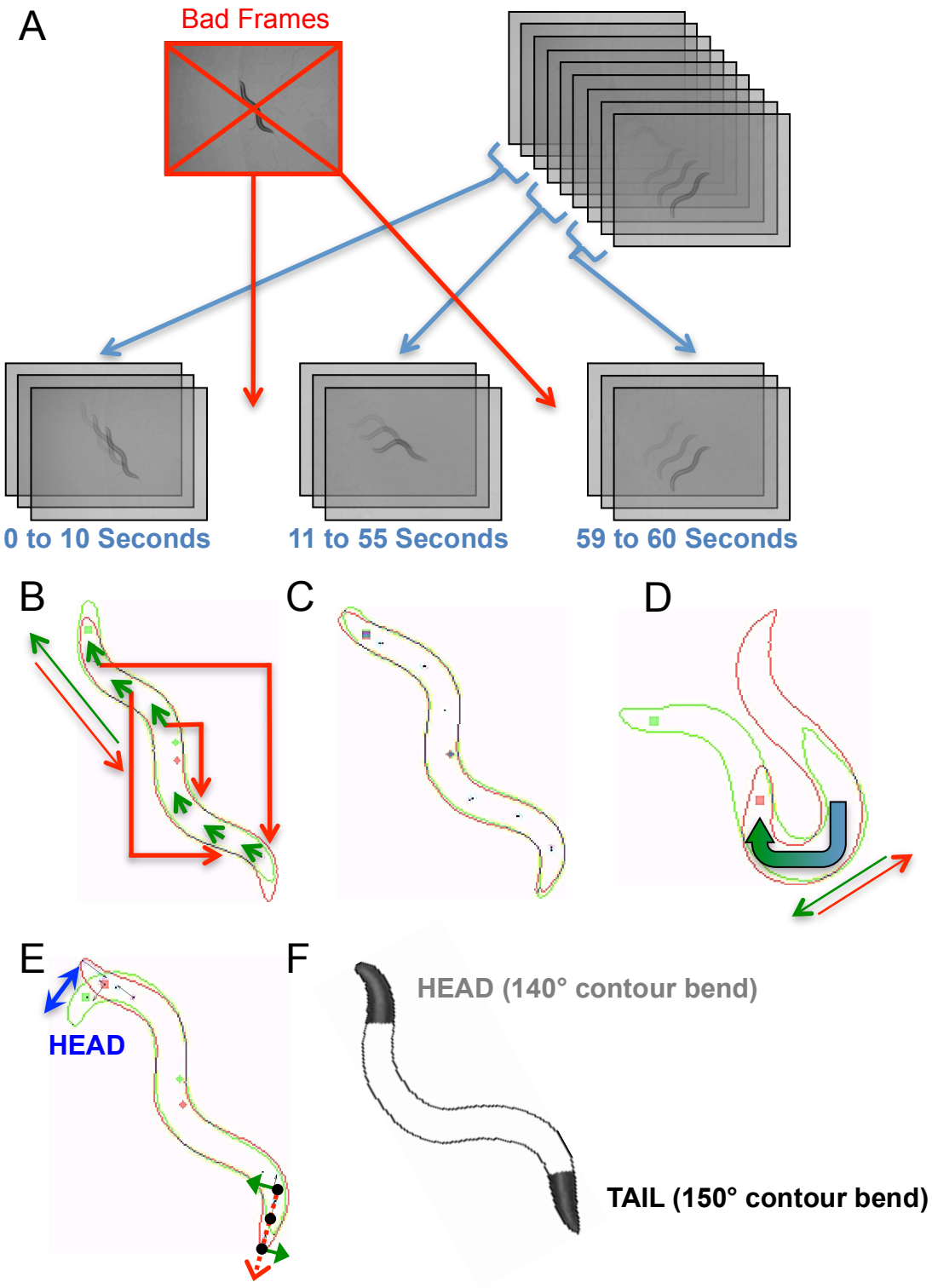


Figure 3.8: Chunking Video and Head-Tail Detection

Head and tail detection proceeds by organizing continuous worm sequences into chunks. Worm sequences, in each chunk, are organized to share the same orientation. The orientation simply matches up the two ends of subsequent worms since their head-tail identity is not yet known. The aggregate head foraging movements or pixel intensities then decide the head and tail assignment in each chunk. A. A video is segmented, frame-by-frame, into a sequence of worms. Subsequent worms are aggregated into a chunk. Subsequent worms may be interrupted by dropped frames, discarded stage movement frames, and rejected segmentations. When an interruption is sufficiently long, the chunk is terminated and a new chunk begins. B. Within a chunk, each worm sets the orientation for its subsequent neighbor. Skeleton distances are measured to the subsequent worm in its originally-segmented and flipped orientations. The orientation with minimal distances (labeled by green arrows) is chosen as the correct one. A chunk therefore contains a set of worms with identical head-to-tail orientation. C. When a stage movement interrupts subsequent worms, the worms are aligned by their centroid (labeled by a green cross) before measuring the distance between their skeleton points and choosing the correct orientation. D. When an omega bend interrupts subsequent worms, the worms are oriented by following the path through the bend. The head-to-tail vectors (labeled by the green and red arrows), tangent to the bend, must point in the same direction for both worm shapes. E. Head foraging activity (labeled by the blue arrow) helps to determine the head and tail assignments for a chunk. To compute the foraging movements, three points (labeled by black dots) are chosen along each worm end. The mean vector (labeled by the red dotted arrow) through these three points is computed for the initial worm shape. The subsequent worm shape is used to measure the movement for each point. Head foraging is measured as the mean of these point movements' orthogonal vector components (labeled by green arrows), relative to the red dotted arrow (the mean vector through the initial worm's endpoints). F. When there is insufficient data to measure foraging, pixel intensities (which are brighter and have more variance at the head) and convex bend at the tip of their contour (which is smaller at the head) alone determine the head and tail assignments.

3.3.2.2 Head and Tail Assignment

With the video subdivided into chunks, and each chunk's worm shapes sharing identical head-to-tail orientation, the head and tail can be assigned. The assignment is performed, per chunk, using each chunk's aggregate statistics. Previous methods have used movement, pixel intensities, and the angle of the contour to distinguish between the head and tail [5,9]. I use similar methods with several refinements.

Head foraging swings can be resolved within 1/4 to 1/6 of a second. These movements are distinguished by large vectors, orthogonal to the head-to-tail axis, which occur only at the head. Forward motion, on the other hand, is an unreliable indicator of the head-to-tail axis. Wild-type worms often crawl backwards for long periods of time and several mutants show an even greater preference for backward locomotion [15,16]. Therefore, when sufficient data is present within a chunk, I use orthogonal movements to help identify the head. The algorithm, shown in Figure 3.8E, is as follows:

1) I define worm A and worm B as subsequent worm shapes, within the chunk, at least 1/6 of a second apart and up to 1/4 second apart if smaller time intervals are unavailable. Worm A is the shape at the earlier time and worm B is the shape at the later time. For each frame where worm A and B are available, these worm shapes are compared to identify head swings.
2) The skeleton is an approximation of the worm. Therefore, for redundancy, three skeleton points are used to represent each endpoint (the head and tail). The first point is at the skeleton's end. The last point is at the, approximate, head and tail boundary (1/6 from the end). The midpoint is centered, on the skeleton, between both the first and last point. I define these as point 1, 3, and 2 respectively.
3) I use the redundancy in step 2 to ensure a good approximation of the direction in which the endpoint faces. To do so, the mean of the vectors, between points 1 and 2 and points 2 and 3, is defined as the endpoint's direction.
4) I measure the motion vectors, from points 1-3 (at both endpoints) on worm A, to their counterparts on worm B.

5) Potential head swings are computed as the component of this motion vector that is orthogonal to the endpoint's direction.
6) Head swings generate orthogonal motion in the same direction for all three points. Locomotion, on the other hand, generates orthogonal motion of much smaller magnitude and, often, in opposing directions for all three points. Therefore, the mean of the orthogonal motion, at all three points, is computed to accentuate the difference between swings and locomotion. This mean is defined as the orthogonal motion for its respective endpoint.
7) The orthogonal motion is summed for each endpoint across the chunk. The endpoint with the greatest sum is more likely to be the head.

Generally, head foraging swings are likely to be more reliable than head-tail pixel intensities and angularity -- in our large data set, we have never observed a worm that did not exhibit foraging but it is not uncommon to see the head and tail segment with unexpected pixel intensities and angles. Unexpected measures for pixel intensities and angles are especially common in permissive experimental conditions which have a consequent variety in image characteristics. Unfortunately, in some cases, chunks may be too short or fragmented to identify head foraging swings. In these circumstances, I rely on pixel-intensity and angular statistics to distinguish the head and tail. For most experimental conditions, the worm's head appears brighter than the tail and has more variance due to the contrast between its bright pharynx and dark cuticle edges. Moreover, the head is less angular than the tail. I exploit these pixel-intensity and angular characteristics for our head and tail choice. The algorithm, shown in Figure 3.8F, is as follows:

1) For each worm shape within the chunk, I define the area of both endpoints (the head and tail) as 1/6 of the worm from their respective ends. This area is bounded by the contour and two lines emanating from 1/6 of the skeleton to the nearest point on either side of the contour.
2) For both endpoints I compute the 25%, 50%, and 75% quantiles from the cumulative distribution function (CDF) as well as the standard deviation of the bounded pixels' grayscale values. The interquartile range, defined as the difference between the 75% and 25% quantiles, is also computed as another measure of variance.

3) The angle, for each endpoint, is defined by the low-frequency bending at their respective contour end points (the actual locations of the head and tail, detailed in 13.3.1.4 Head and Tail Localization).

With the statistics aggregated across each chunk, we use a classifier to detect the head. This work was performed by Andre Brown. Linear discriminant analysis (LDA) was used on the combination of lateral motion, pixel intensities, and angularity measurements from worm endpoints for a training set of 68 randomly-chosen videos. Angular measurements were found to have low weighting, relative to lateral motion and pixel intensities, and were therefore removed from the classification. The resulting classifier was then used for our entire data set to automatically detect the head for each worm, within each chunk.

3.3.3 Mapping Logged Stage Movements in Video

To reduce costs, our WT2 hardware foregoes using a synchronized clock for its camera and motorized stage. As a result, logged stage movements (which re-centered the worm in video) must be mapped to their corresponding video frames and, these movement frames must be discarded from segmentation. This mapping is also required to convert the worm's video pixels into real-world coordinates. I use an intricate algorithm to match the log and video. Fundamentally, each logged stage movement is matched to a corresponding large change in variance between subsequent frames. The beginning and end of each stage movement is then delineated using local thresholds to separate worm motion from stage motion. The algorithm, shown in Figure 3.9, is as follows:

1) The video is converted to grayscale.
2) Each video frame is subtracted, pixel-by-pixel, from its subsequent frame.
3) The variance of the pixel differences is recorded for each pair of subtracted frames. This variance captures worm and background movement while ignoring automatic adjustments to the camera's exposure.
4) The Otsu method is used locally, as a rough estimate, to separate the stage movement modality from the opposing non-stage-movement modality.

5) Each stage movement in the log is paired with its video frame(s) counterpart by searching for a corresponding, Otsu-identified, peak in variance near the time at which it occurred.
6) A small threshold is used to identify the end points of stage motion on either side of the corresponding peak. This small threshold is computed, per side, from the local, small-variance values between the current stage-movement peak, up to the next boundary for stage motion.
7) I use multiple checks to handle clock uncertainty as well as ambiguities in variance between small stage movements as opposed to large worm movements. If any of the stage movements lack video counterparts or any of the mappings are ambiguous, the algorithm fails and reports the error.

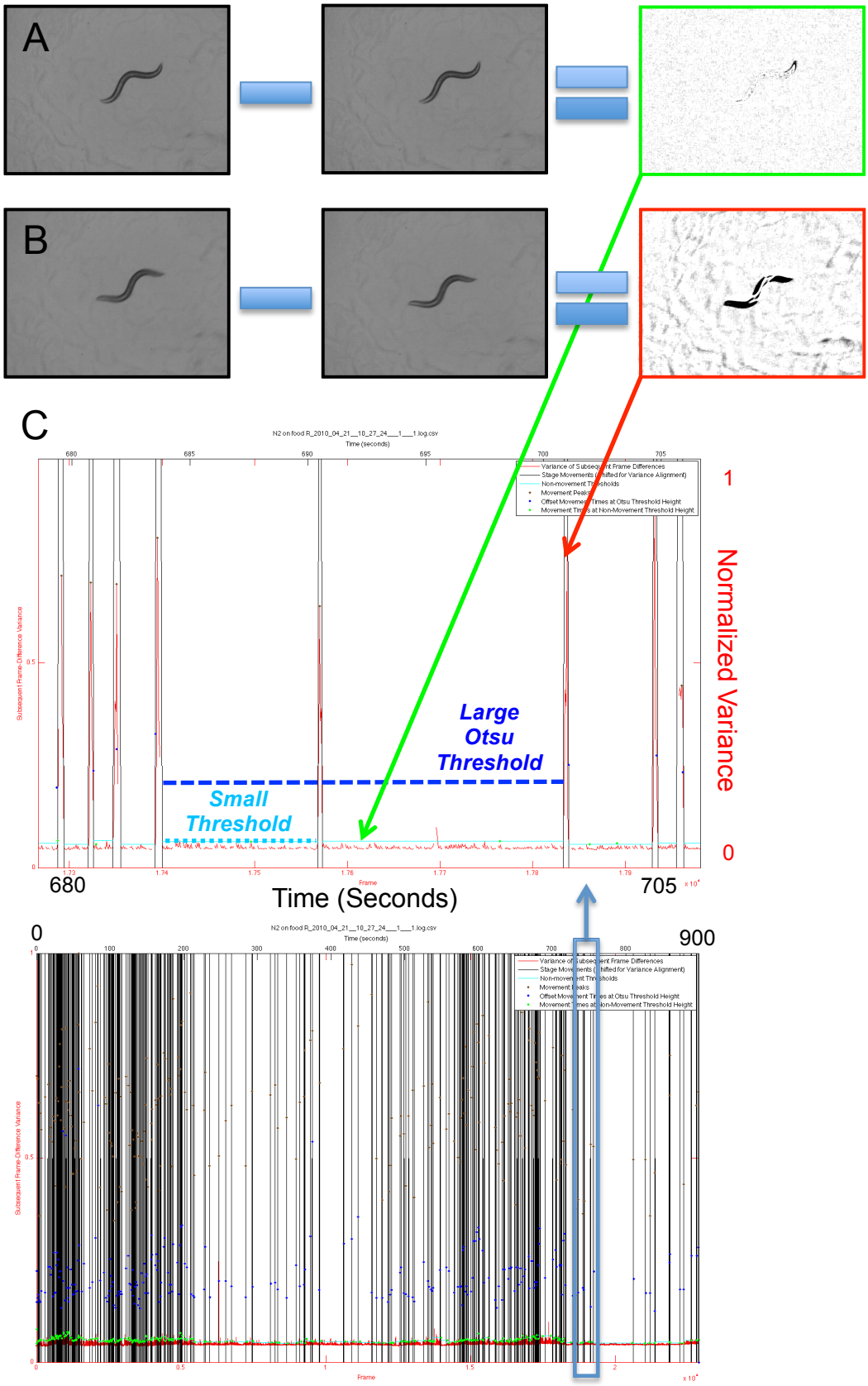


Figure 3.9: Stage Movement Mapping (WT2 Screenshot)

An illustration of mapping logged stage movements to their corresponding video frames. A. Subsequent video frames, containing only worm motion, are subtracted from each other. The result shows small changes due to worm motion and pixel noise with a correspondingly small variance (observed among the magnitudes of the pixel differences). B. Subsequent video frames, containing stage motion, are subtracted from each other. The result shows big changes due to stage motion with a correspondingly large variance. C. Below, a WT2 screenshot highlights high-variance, stage motions in vertical black lines. Above, a magnified portion of the WT2 screenshot details the variance (labeled by the red plot) associated with each video frame. The non-movement frames, from part A (labeled by the green arrow), have low variance values. The stage-movement frames, from part B (labeled by the red arrow), have much higher variance values and have been enclosed in vertical black lines to indicate the start and end of movement. An Otsu threshold (labeled by the blue dashed line) is computed locally assuming two modalities, non-movement (low variance) and stage-movement (high variance). This threshold separates the non-movement frames from those containing stage motion. The Otsu threshold identifies a peak, corresponding to stage movement, near the logged time. A small threshold (labeled by the cyan dotted line) is used to identify the end points of stage motion on either side of its corresponding peak.

3.3.4 Automatic Vignette Recovery

When recording video with a vignette correction, the WT2 tracking software does not correct the recorded video images. The video and vignette correction are saved separately so that the correction can be applied during segmentation. Occasionally, the vignette is miscalibrated or lost. I can recover the vignette correction from video. Basically, the worm is removed from stage movement frames and these frames are then used to compute the vignette correction. The algorithm, shown in Figure 3.10, is as follows:

1) The video is converted to grayscale.
2) The algorithm in section 13.3.3, Mapping Logged Stage Movements in Video, is used to identify stage movement frames. Stage movement frames have several advantages in vignette computation. Their worm is off-center in a, somewhat, random location. Therefore, with sufficient frames I can sample most of the vignette, as opposed to losing the central image pixels, which the worm normally occludes. Also, stage motion causes the image of the plate, and especially the worm tracks, to blur. This blurring of the plate leaves behind a clearer representation of the vignette and any dirt obscuring the lens.
3) In each stage movement frame, the worm is identified using the algorithm in section 13.3.1.3, Segmentation of the Worm Contour; except, no vignette correction is applied.
4) If the worm touches the image boundary, I assume it has been confused with the vignette and, therefore, discard the frame.
5) A mask is applied to remove the worm from the image. Thereafter, the mean of the remaining image pixels is subtracted from these pixels. The result is a vignette correction for the current image, albeit one confounded by stage-motion-blurred worm tracks.
6) The final vignette correction is computed as the mean of the vignette corrections across all images from step 5. With only a few frames, the stage-motion-blurred worm tracks are smoothed out, resulting in an accurate representation of the vignette correction.
7) In the event that this automatic vignette correction is unsatisfactory, the user can manipulate the automated computation to achieve better results.

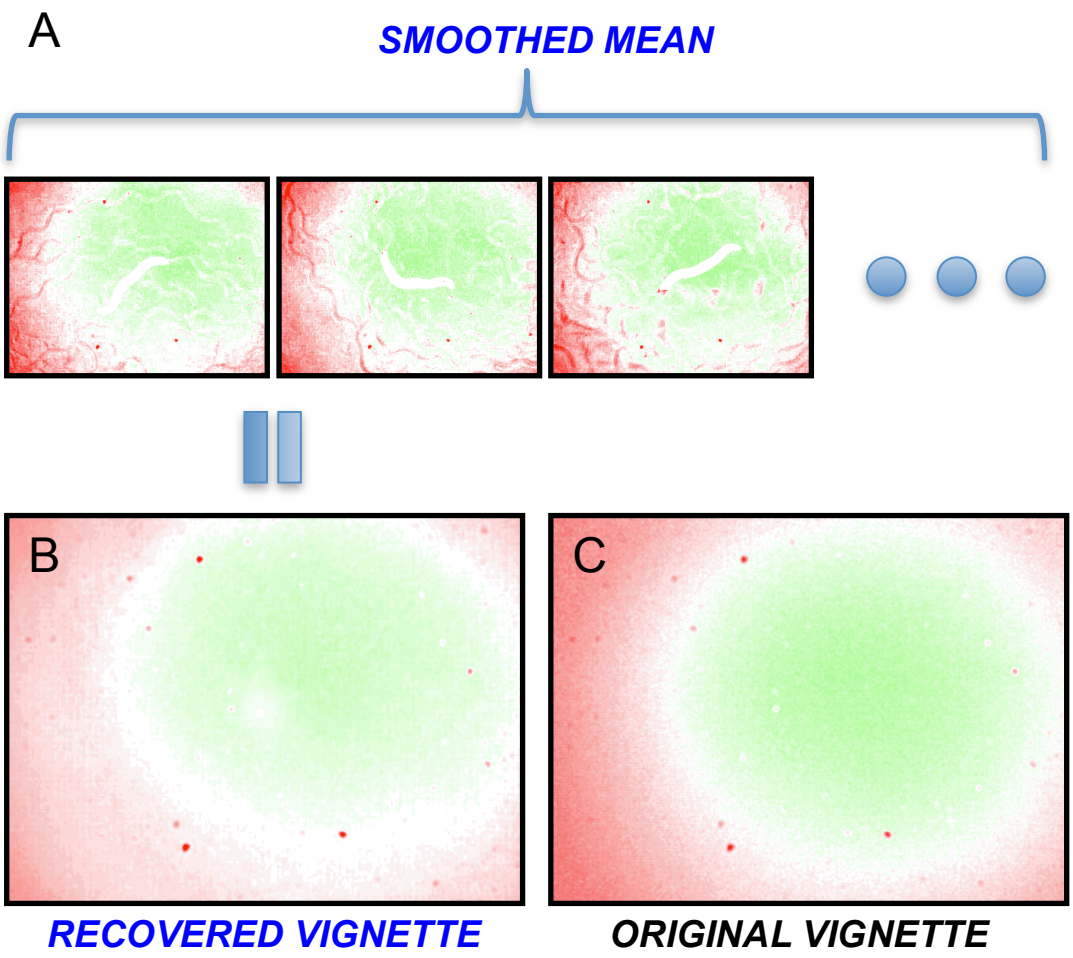


Figure 3.10: Automatic Vignette Recovery

An illustration of the automated vignette recovery algorithm. Red pixels are corrected for being too dark whereas green ones are corrected for being too bright. A. Stage-motion frames are chosen due to the off-center location of their worm, as well as, their blurred background which leaves a clearer representation of the camera vignette and lens occlusions. The worm is removed from each frame using Otsu thresholding. B. The smoothed mean of these frames is used to compute the recovered vignette correction. C. The original vignette, computed by the tracking software, is shown for comparison.

3.4 Results

3.4.1 Stage Movements and Segmentation

Large data sets present particular obstacles to reviewing segmentation and head-tail assignments. For example, we currently have 12,000 videos collected using the WT2 protocol. This collection represents approximately 1/3 of a billion frames, a considerable amount to review. We examined the accuracy of our analysis algorithms. Our earlier videos suffered greater intermittent frame loss due to dropped frames and issues in image quality. These problems have since been corrected. Present videos suffer no dropped frames and rejected segmentation is primarily due to coiled shapes which are not yet handled by the software.

Over 100 early videos, representing a broad range of mutants as well as wild types, composed of roughly 2.5 million frames, and illustrative of some of the worst cases for imaging conditions, were reviewed for their segmentation and stage movement detection, by eye, using three experts, Andre Brown, Tadas Jucikas, and myself. The stage movement detection was judged to be accurate and precise, identifying only frames containing stage movement. The segmentation was judged to be conservative, eliminating cases where the resulting skeleton would have been considered correct, as a consequence of automated checks for accuracy. Those skeletons and contours which passed automated checks, were judged to be accurate by the three reviewers, with minor contentions at the head and tail, when the image had insufficient clarity for both the algorithm and the reviewer to choose an accurate outline for these locations (e.g., within deep food or when the worm dove within in the agar or lifted its head off the plate). Under normal imaging conditions, reviewers judged the resulting contour and skeleton to be both accurate and precise. It is worth noting that removal of stage movement frames may slightly bias measurements away from locomotory shapes. Similarly, worms that dwell in occluding food (e.g., as is present at the edge of food lawns), dive into the agar, or lift their head off the plate, may experience more rejected segmentations due to our checks for accuracy, resulting in a measurement bias away from their corresponding body shapes during these events.

To further assess the statistics for frame loss and segmentation checks, we selected roughly one quarter of our growing data set, to sample a broad range of video conditions and worm strains (Figure 3.11A-B). This subset includes 2,958 videos, about 740 hours or 30 continuous days, representing approximately 80 million total frames. Our segmentation algorithm provides a report for every video frame that did not segment. This report is directly available in our database for experiments. Within the video subset, 9% of the frames produced worm shapes that failed automated checks for accuracy, 6% of the frames were lost to stage movements, and 4% of the frames were lost to dropped frames, a problem I have since fixed. Therefore, 81% of the frames produced worm shapes for further analysis. Using our protocol of 20 15-minute-long videos per worm strain, I expect, on average, 4 hours of segmented data to characterize each strain. Chapter 4 will further discuss these statistics for a collection of videos, filtered to ensure quality, representing 305 of our recorded strains.

We next explored the segmentations rejected by our automated checks. These checks eliminate inaccurate worm shapes. Among the 9% of frames rejected by our automated checks, 31% produced worm shapes that were removed due to an insufficient number of worm ends (i.e., a worm must have both a head an tail); such shapes often occur when the worm coils or touches itself, hiding either the head, tail, or both ends. Conversely, 27% of shapes were removed due to too many worm ends; this tends to occur when the worm touches its eggs or is obscured by dark tracks, left behind from locomotion, within the food lawn. Another 28% of worm shapes were removed due to having a disproportionate body shape; this can happen when the worm touches its eggs, when the head and tail are obscured in food, when the worm lifts its head above the plate, or when it dives beneath the agar. With far less representation, 11% of the shapes were removed due to significantly disproportionate lengths on either side of the contour (i.e., both sides of the contour should roughly be the same length); this is a strong indication of the worm touching itself in an omega or wreath-like shape. The remaining 3% of worm shapes were removed due to the worm touching the image boundary; while this scenario sounds like the worm nearly escaped tracking, more often, it indicates that the worm reached a dark location on the plate (e.g., as may be caused by shadows cast from condensation on the plate's cover) or even minor vignetting at the very edge of an image – in detail, the aforementioned

dark location, at the image boundary, is thresholded as part of, or instead of, the worm thereby causing it to appear as if the worm touches the image boundary.

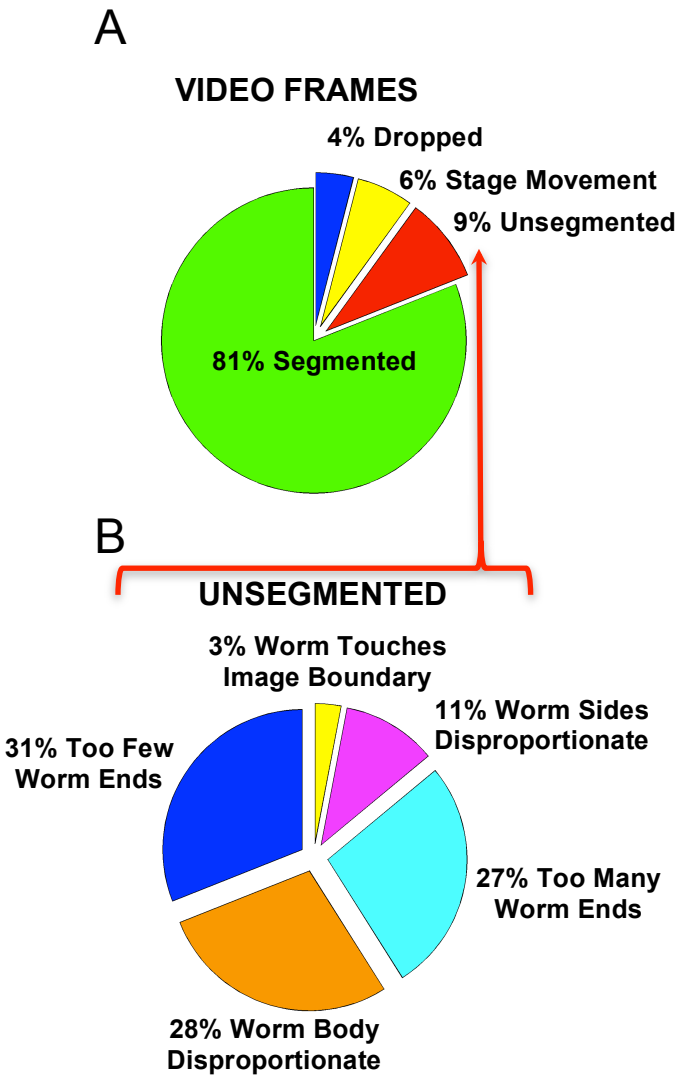


Figure 3.11: Segmentation Statistics

A. The statistics for frame loss and segmentation of 2958 videos (approximately 740 hours or 80 million total frames) using our algorithm. B. A detailed assessment, displaying the automated segmentation checks, which led to rejection of 9% of the video frames in part A.

3.4.2 Head-Tail Detection

Andre Brown and I examined the accuracy of our chunking and head-tail assignment algorithms. Andre examined 62 of our early videos. These videos represent 1,084,368 segmented frames split among 1,047 chunks. Some of them are of such poor quality that many of their chunks are composed of less than 1 second worth of frames. The poor quality of these videos is reflected in their low count for segmented frames, which amount on average to less than 70% of the videos. In the resulting segmentation, all 1047 chunks preserved correct endpoint (head-to-tail) orientation across their sequence of frames. There was not a single failure present, even within videos composed of one continuous chunk. Manual labeling of the head in these chunks was used to train our classifier. After automated head detection by our classifier, I examined 133 random videos (roughly 1% of our data, 2.25 million segmented frames), representing a broad range of mutants (including several nearly motionless UNCs). I found that the head was correctly labeled with a mean and standard deviation of 94.39 ± 17.54% across individual videos and 95.6% of the video frames collectively.

3.4.3 Primary Worm Measurements

Traditionally, extracting worm features from a video is split into two steps, primary and secondary computations. Primary worm measurements are quantified, from the segmented worm, for each individual frame. Secondary behavioral features are computed from the primary measurements. Primary measurements include the worm's contour, skeleton, length, a subset of widths, and a subset of pixel-intensity details. I extract a superset of these primary measurements and translate them to their real-world coordinates and values. Many secondary feature computations require a stable worm representation with a fixed number of representative points. For example, to compute neck motion, the points representing the neck must correspond to similar worm coordinates from one frame to the next; if the worm elongates or shrinks the number of points can fluctuate, complicating the algorithm. For this reason, I provide an additional normalized worm representation. This normalized representation transforms the worm skeleton and both sides of its contour to 49 points each so as to

sample at, approximately, twice the rate of the musculature (Figure 3.12**Error! Reference source not found.**). Consequently, WT2 can be extended to work with, or compute, the multitude of published, secondary feature extractions. Chapter 4 presents a secondary set of features computed using the primary set discussed herein.

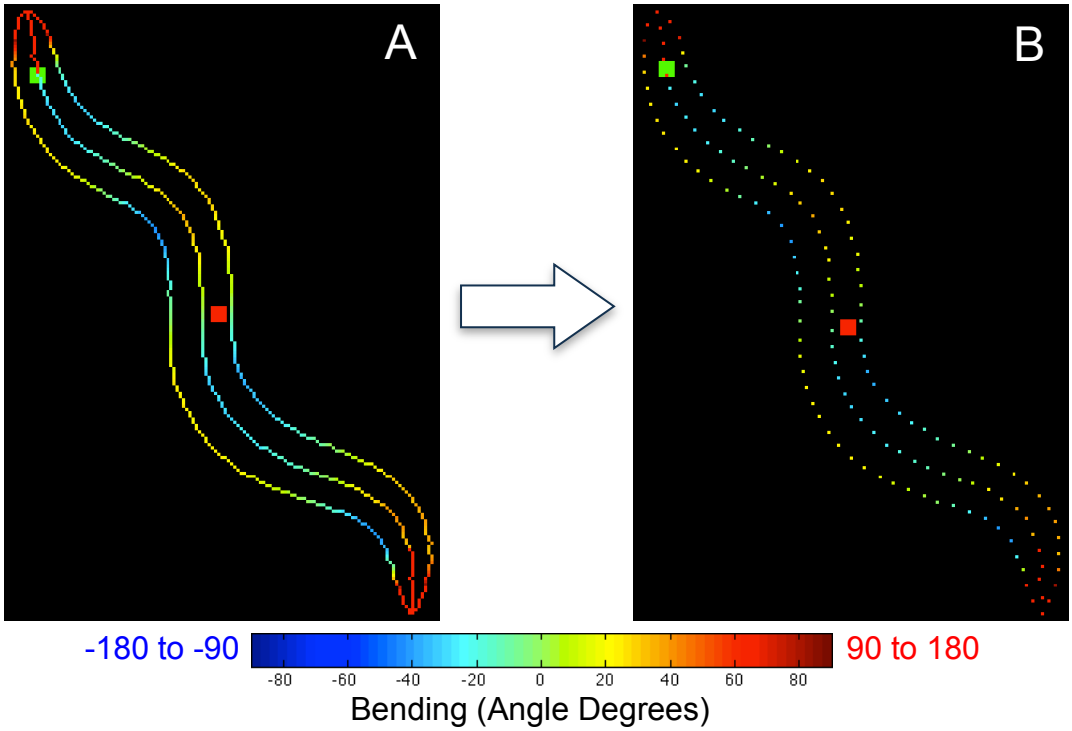


Figure 3.12: Worm Normalization to 49 Points

A. A segmented worm. The worm is normalized to aid in secondary feature computation. The bending of the worm's contour and skeleton are represented as a heat map (ranging from blue at a concavity of -90° and below, to red at a convexity of 90° and above). The head is labeled with a green square. The ventral side is labeled with a red square. B. The segmented worm is normalized to 49 equally-spaced points on the skeleton and both sides of the contour. The points are fractional absolute coordinates and have been scaled and rounded to accommodate the pixel grid.

Our primary measurements, shown in WT2's corresponding onscreen image in Figure 3.13, include:

1) The worm's contour, split into dorsal and ventral sides.
2) The contour's low-bending at each point. The bending is defined as the angle formed, at each contour point, between edge vertices two muscle segments away. A muscle segment is defined to be 1/48 of the contour's length. Worm muscle segments overlap and have grouped innervation. Therefore, the contour bending presents high signal to noise at this spatial sampling.
3) The worm's skeleton. The skeleton is defined as the midline of the contour from head to tail.
4) The skeleton's bending at each point. The bending is defined as the angle formed, at each skeleton point, between edge vertices two muscle segments away. A muscle segment is defined to be 1/24 of the skeleton's length. Bending for points less than two muscle segments from the head and tail cannot be computed and is therefore undefined.
5) The worm's head and tail point defined as the locations of the maximum convex bending angle on opposing sides of the contour (detailed in 3.3.1.4, Head and Tail Localization).
6) The worm's length.
7) The worm's width at each skeleton point.
8) The worm's head, tail, dorsal side, and ventral side area boundaries on both the contour and skeleton. The head and tail areas are bounded at 1/6 distance, along the contour and skeleton, from their respective points. The dorsal and ventral side areas are bounded by the ends of the head and tail and split by the skeleton.
9) The worm's head, tail, dorsal side, and ventral side area. These areas correspond to the number of pixels within their boundaries.
10) The worm's head, tail, dorsal side, and ventral side pixel intensities. These intensities are evaluated from the pixels within their boundaries. The pixel-intensity measurements include the 2.5%, 25%, 50%, 75%, and 97.5% quantiles from the cumulative distribution function as well as the standard deviation of the bounded pixels' grayscale values.

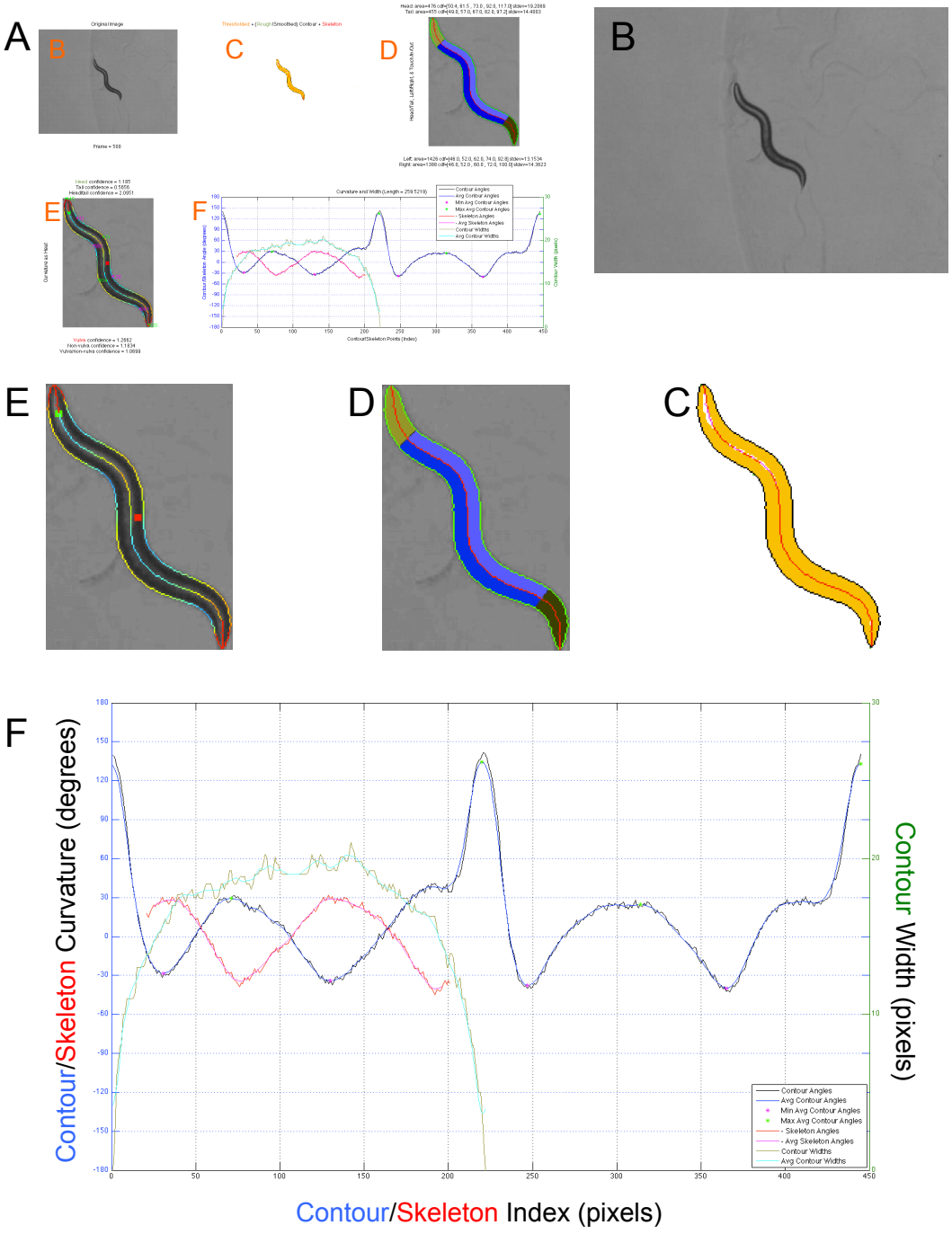


Figure 3.13: Primary Worm Measurements (WT2 Screenshot)

An illustration of our primary worm measurements. A. A WT2 screenshot of worm measurements. The individual panels have been labeled and magnified in parts B-F. B. The video frame image. C. The thresholded image of the worm (labeled in yellow), its contour (labeled in black), and its skeleton (labeled in red). D. The worm's head area (labeled in yellow), tail area (labeled in green), dorsal side (labeled in blue), and ventral side (labeled in lavender). The WT2 screen provides further information. The size of the areas (in pixels) is presented along with pixel-intensity information (in grayscale values ranging from black at 0 to white at 255). The pixel-intensity information includes 2.5%, 25%, 50%, 75%, and 97.5% quantiles from the cumulative distribution function of the bounded pixels' values as well as their standard deviation. E. The bending of the worm's contour and skeleton represented as a heat map (ranging from blue at a concavity of -90° and below, to red at a convexity of 90° and above). The head is labeled with a green square. The ventral side is labeled with a red square. The WT2 screen provides further information. The local minimas (labeled in magenta) and local maximas (labeled in green) of bending are marked to correspond with the graph in part F. F. The bending of the worm's contour and skeleton are graphed alongside the contour width (in pixels) at each skeleton point. The WT2 screen provides further information. The local minima (labeled in magenta) and local maxima (labeled in green) of bending are labeled to correspond with the heat map in panel E. The worm's length (in pixels) is provided in the graph's title.

3.5 Discussion

3.5.1 Comparison to Related Work

The push for larger data volume, and higher-throughput, in single-worm tracking places considerably larger demands on current tracking systems, which were developed to work with much smaller data sets. To date, the largest published collection of tracked worms amounts to 32 strains represented by, nearly, five million frames as reported by Zhang et al. [2]. The associated technique restricts both the imaging conditions and worm age. Moreover, were they to grow their set by one order of magnitude, to 320 strains, the result would be 50 million frames and, as will be discussed within the next paragraph, with a 98% success rate for head detection, 100,000 frames would be misclassified (marring any consequent phenotypic measurements) without quick recourse to fix these frames. Not to mention that their associated tracking unit costs nearly four times the price of a WT2 unit. Our current collection is several orders of magnitude larger and amounts to over 300 strains. Every strain is represented by, at least, 20 worms recorded for 15 minutes, at 640x480 pixels and 30Hz resolution. All mutant worms are matched to an N2 control, prepared and recorded in the same conditions, at the same time. This amounts to, nearly, 300 million frames. Furthermore, our WT2 system permits flexible experimental conditions and the full range of larval stages.

Clearly, manual checks for even a small fraction of such a large data set would require a significant amount of time. Therefore, this large data volume and experimental flexibility demand robust algorithms to automate the vast majority of feature extraction and expedite manual review when necessary. The aforementioned Zhang collection uses a tracking system that achieves 98% correct head and tail classification [9]. Hoshi and Shingai report values of 99.8% correct for their head and tail detection algorithm [5]. However, their method uses similar classification information to ours and requires a data set of worms off food, an even greater imaging restriction than that required by the Zhang protocol and one which makes their conditions at odds with the normal laboratory state of worms. We achieve approximately 95% correct on a much broader data set. Given that we use similar parameters for distinguishing the head, presumably our percentage reflects a bound

associated with using the same classification formula for a broad range of mutations in permissive experimental conditions. Hence, training separate classifiers for distinct subsets of mutations (e.g., specifically on a set of UNCs, EGLs, and nearly wild-type mutants) may restore higher rates for correct head classification.

The segmentation algorithm evolved by testing previously published methods and then addressing any significant complications found when segmenting our own video frames. Several of these adaptations are described in the following paragraphs. Grayscale pixel-intensity characteristics of the head and tail are frequently used to determine their assignment. A consequence of the brighter pixel intensities found at the head and tail is that thresholding often fails to capture the full details of their shape, leaving behind holes and split ends. A common approach, published by Baek et al. [1], uses an imaging operation termed closing to dilate a thresholded worm in order to close holes and split ends. Subsequent to dilation, the thresholded worm is eroded back to its original size while preserving closures. Unfortunately, in our permissive data set we found this closing operation required fine tuning for worm size and had such side effects as smoothing the contour, filling in tight bends, and uniting unconnected regions of a worm, leading to a misrepresentation of the worm (Figure 3.14B). Alternative algorithms obviate this step either by using dark worm pictures, which lack distinguishable differences in head and tail intensity, or by discarding of contours altogether in favor of measuring just worm skeletons. Our algorithm segments a high-fidelity contour, from our worm images, preserving distinguishably contrasting intensities at the head and tail. Holes are not an issue since our contour traces the worm perimeter. The algorithm heals any split ends, removing them from the contour.

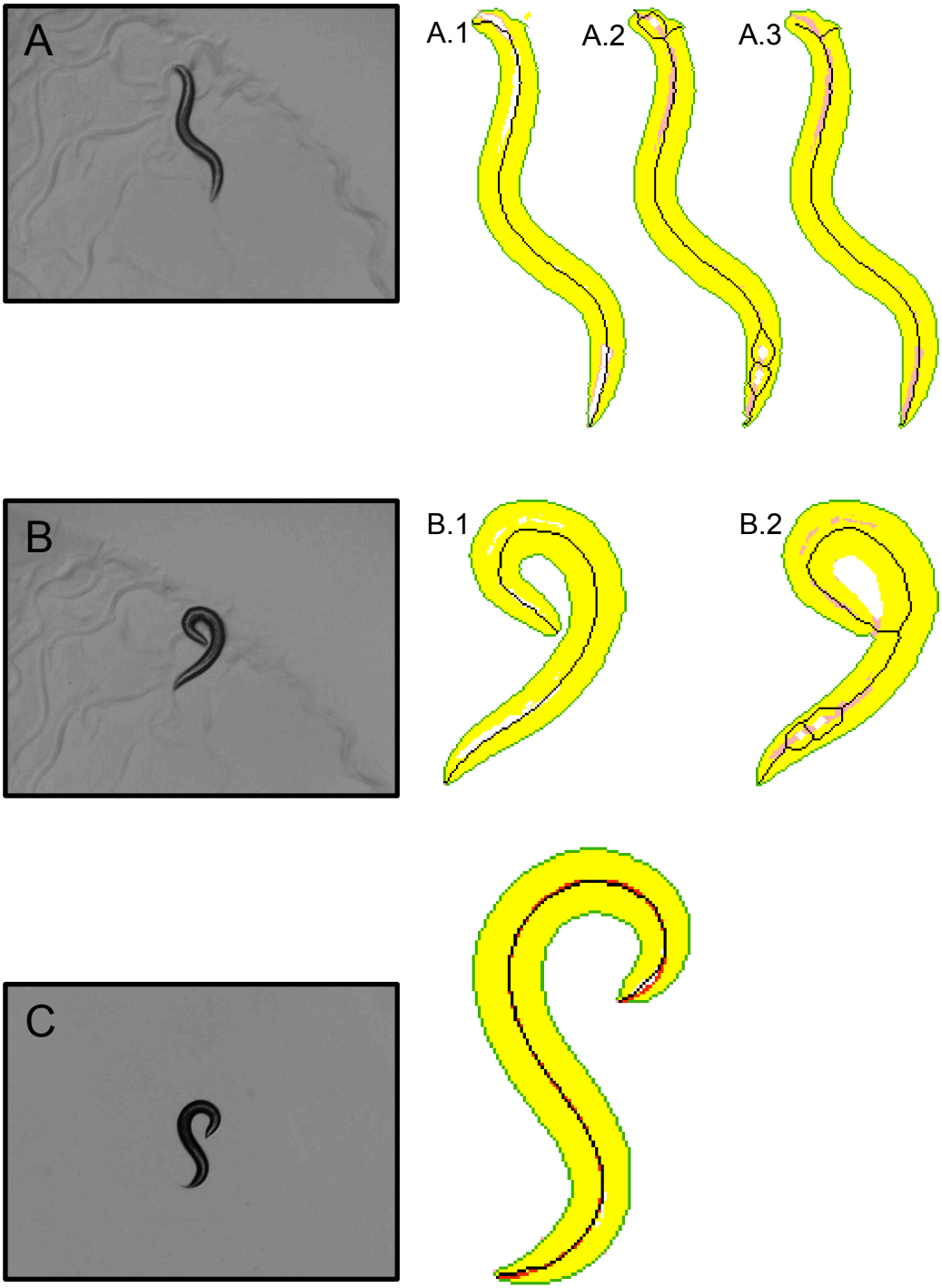


Figure 3.14: Comparing Segmentation and Skeletonization Algorithms

A series of three video images from our data set are used to illustrate the outcome of my own segmentation and skeletonization algorithm in comparison to two popular alternatives (which guided my technique): closing followed by thinning and, spline interpolation. Closing operations required fine tuning to the worm size. Too little closing can leave behind holes. Too much closing can smooth the contour, fill in tight bends, and unite unconnected regions of worm. Spline interpolation, on the other hand, estimates the worm skeleton and, thereby, may misrepresent the correct fit for the midline of the contour. A. The result our algorithm is shown in A.1. A.2 shows the outcome of insufficient closing followed by thinning. The skeleton bifurcates to encircle holes. A.3 shows the outcome of sufficient closing followed by thinning. The skeleton branches, extending to both the head and a nearby bump. Additional closing continues to yield this branched outcome. B. The result of my algorithm is shown in B.1. B.2 illustrates a common case where closing followed by thinning is incapable of segmenting and skeletonizing the worm. The closing operation connects regions that should remain disjoint before it can fill all the holes within the worm contour. There is no equilibrium point between too little and too much closing. C. The result of my skeletonization is shown in black. A cubic spline is interpolated using 12, equally spaced, points along our skeleton. The interpolated skeleton, shown in red, has a poor fit at the head where it diverges from the central path. Spline interpolation is commonly performed using the results of closure and is, therefore, subject to many of the complications shown in parts A and B.

Difficulties with head and tail skeletonization restricted previous algorithms from careful measurements at these locations within our data set. A popular thinning technique, published by Baek et al. [1], led to branching at the head and tail, often misrepresenting the skeleton at these locations (Figure 3.14A). Another popular alternative, published by Roussel et al. [11], which now has several variants, requires splines to interpolate an estimate of the skeleton, potentially misrepresenting the head and tail (Figure 3.14C); however, the smoothing aspects of spline interpolation can be useful in post-processing, for example, to clean measurements from noisy images. A higher-accuracy algorithm, published by Berri et, al. [17], requires distinguishable edges, in terms of pixel intensity, to draw its skeleton. Our flexible imaging conditions are not compatible as they often lead to only small intensity gradients, without a well-defined edge, at the head and tail. I, therefore, adapted a new algorithm to resolve a precise skeleton for the head and tail without branching, interpolation, or the need for distinguishably intense edges.

To achieve ease in segmentation and measurement, other single-worm trackers constrain the size of their worms as well as the permitted experimental conditions. By using adaptive thresholding and scaling algorithms, I remove many of these constraints. Specifically, we have used the WT2 system to analyze worms at all larval stages and different sizes within video (Figure 3.15 and Figure 3.18). Moreover, we have used WT2 to analyze worms off food, on food, and within liquid environments (Figure 3.16). The WT2 code is available publicly. Its flexibility permits it to be used on old data sets as well as in new applications without much modification. Beyond these benefits, my underlying segmentation and measurement algorithms may provide a more accurate skeleton at the head and tail and, prospectively, greater precision overall as a result of chain-code lengths and subpixel interpolation.

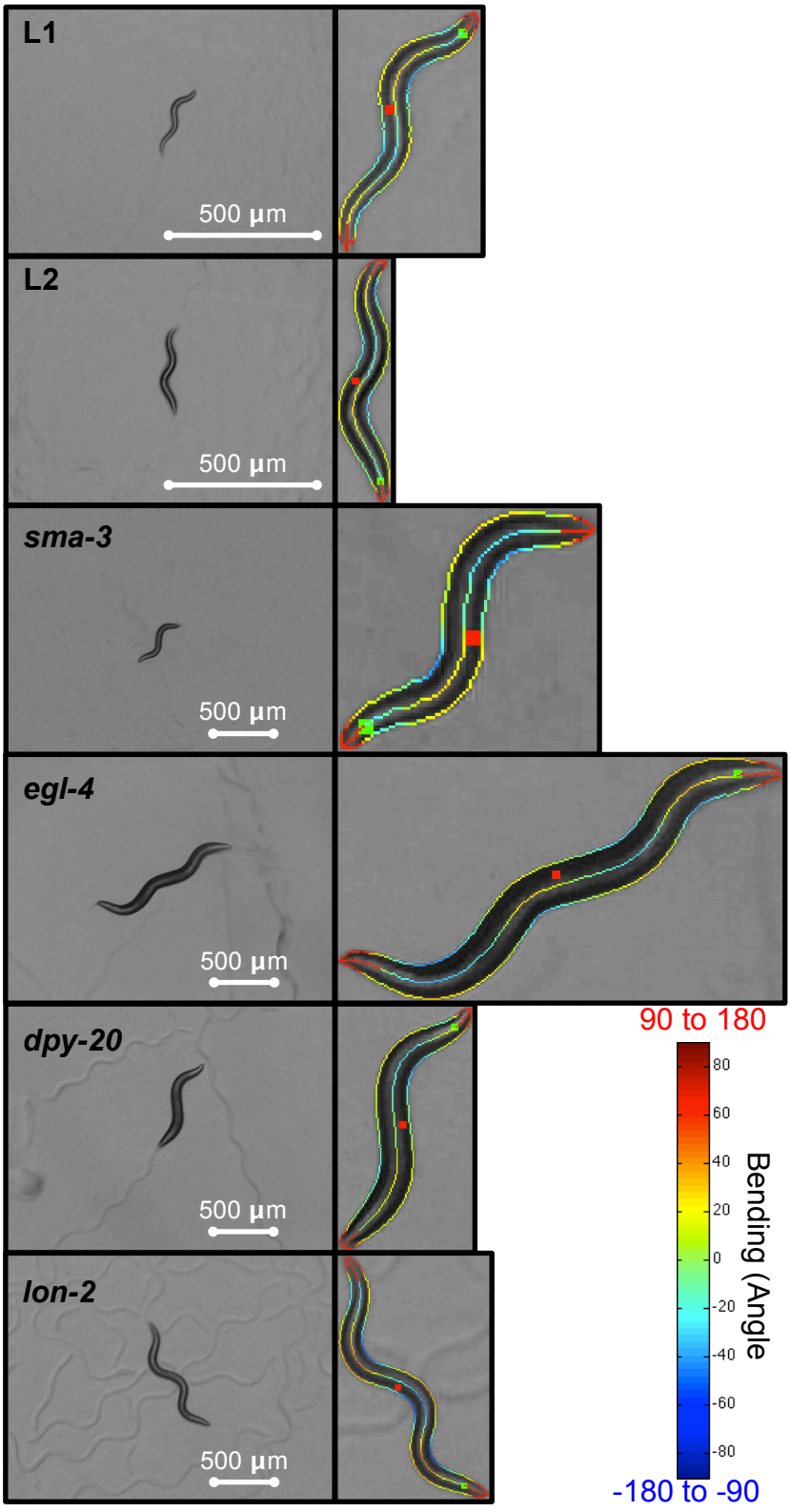


Figure 3.15: Analyzing Early Larval Stages and Various Worm Sizes (WT2 Screenshots)

WT2 screenshots illustrating scale-invariant analysis. The contour and skeleton are drawn using a heat map to indicate their bending. Skeleton bending near the endpoints cannot be calculated due to insufficient data; these data points are colored in red for visibility. Automated head and ventral-side detection choices are labeled, respectively, by green and red squares, at representative locations. Ventral-side detection is not discussed; this assignment is usually specified by the user, at the tracking stage, when performing their experiment. The first two high-magnification, video images show analysis of the first two larval stages of *C. elegans*. Upon hatching, wild-type L1 worms measure approximately 250µm long. After the next molt, wild-type L2 worms measure nearly 400µm long. The remaining images show analysis for a variety of mutant worms with varying body shapes. Several of these illustrate classic body-shape phenotypes designated by Sydney Brenner in 1974 [18]. *sma-3* worms have a mutation that results in a "small" phenotype, a short and thin body relative to the wild type. *egl-4* worms have a mutation that leads to a longer body than wild type [19]. *dpy-20* worms have a mutation that results in a "dumpy" phenotype, shorter than wild type but with the same diameter. *lon-2* worms have a mutation that results in a "long" phenotype, a long and thin body relative to the wild type.

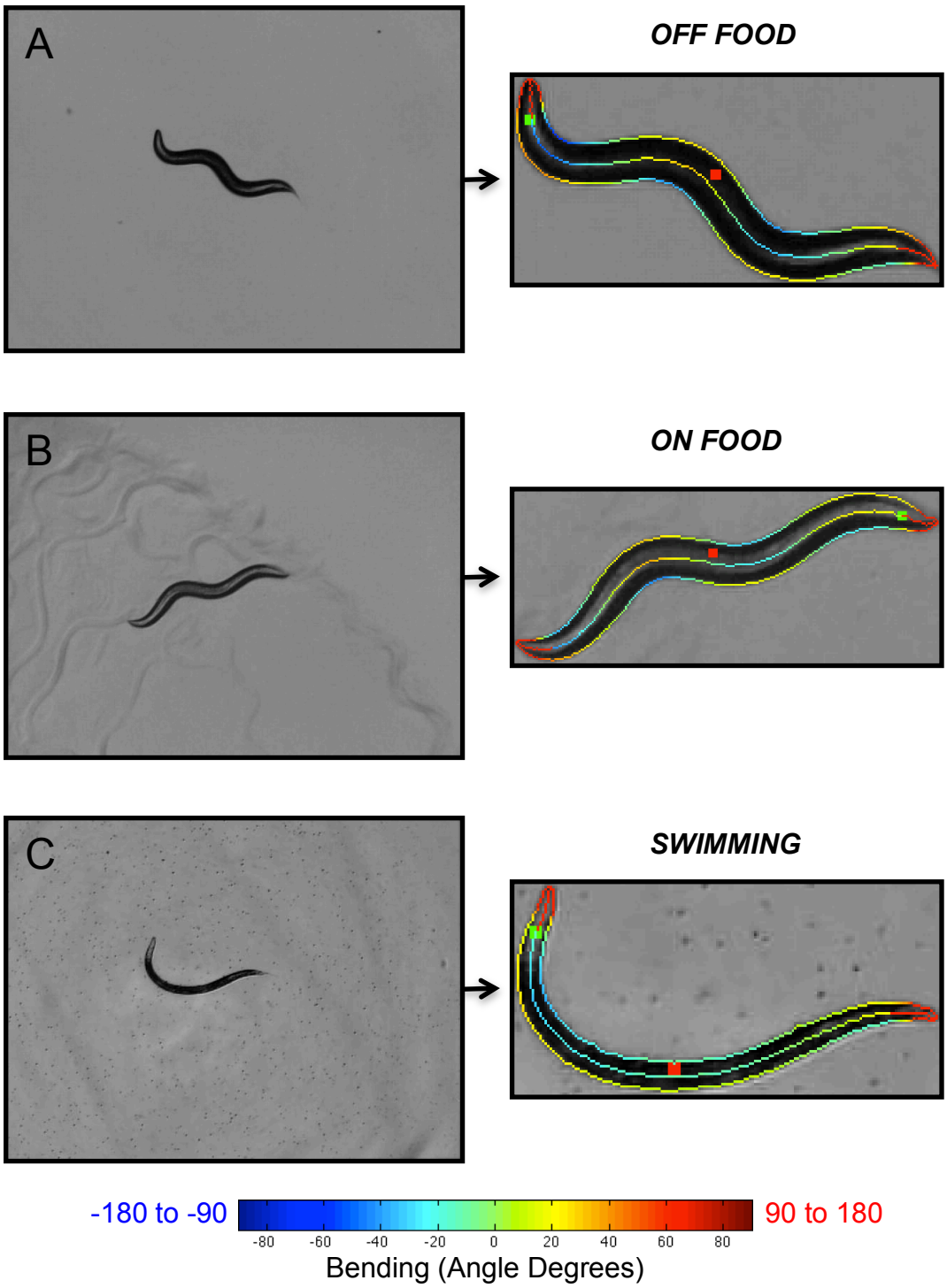


Figure 3.16: Analyzing Worms in Various Environments (WT2 Screenshots)

WT2 screenshots illustrating analysis in different environmental conditions. As mentioned earlier, the contour and skeleton are drawn using a heat map to indicate their bending. Skeleton bending near the endpoints cannot be calculated and is colored in red for visibility. The head and ventral-side are labeled, respectively, by green and red squares. A. Off-food conditions. B. On-food conditions. C. Liquid conditions which induce swimming behavior (as opposed to crawling on agar).

Large data volumes can cause a bottleneck if the speed of analysis is insufficiently fast to keep up with the speed of data collection. Moreover, despite automation, extracted data must permit quick manual checks for verification of the calculations. Information on the analysis speed and any tools available for manual review is hard to come by in publications of previous trackers. On our own computers (detailed in Chapter 2) and laptops, the WT2 software completes analysis for 15-minute-long color videos within 25 minutes (grayscale videos take a much shorter time), an upper bound of 1.7 times real time. This permits us to combine 8-hours of experiments with an overnight analysis on the same computer, without any bottleneck. Furthermore, for each experiment, I produce an analogous video output of the original recording combined with the extracted contour, skeleton, bending, head choice, ventral side, and discarded frame status (i.e., stage movement, rejected segmentation, dropped frame) overlayed onto each corresponding frame. This video, illustrated in Figure 3.17, can be watched (and optionally sped up) to manually check every frame. We also provide a tool, created by James Scott-Brown, for rapidly reviewing chunks to flip their head-to-tail and dorsal-ventral orientation, when necessary as well as to remove poorly segmented worm frames from the data set.

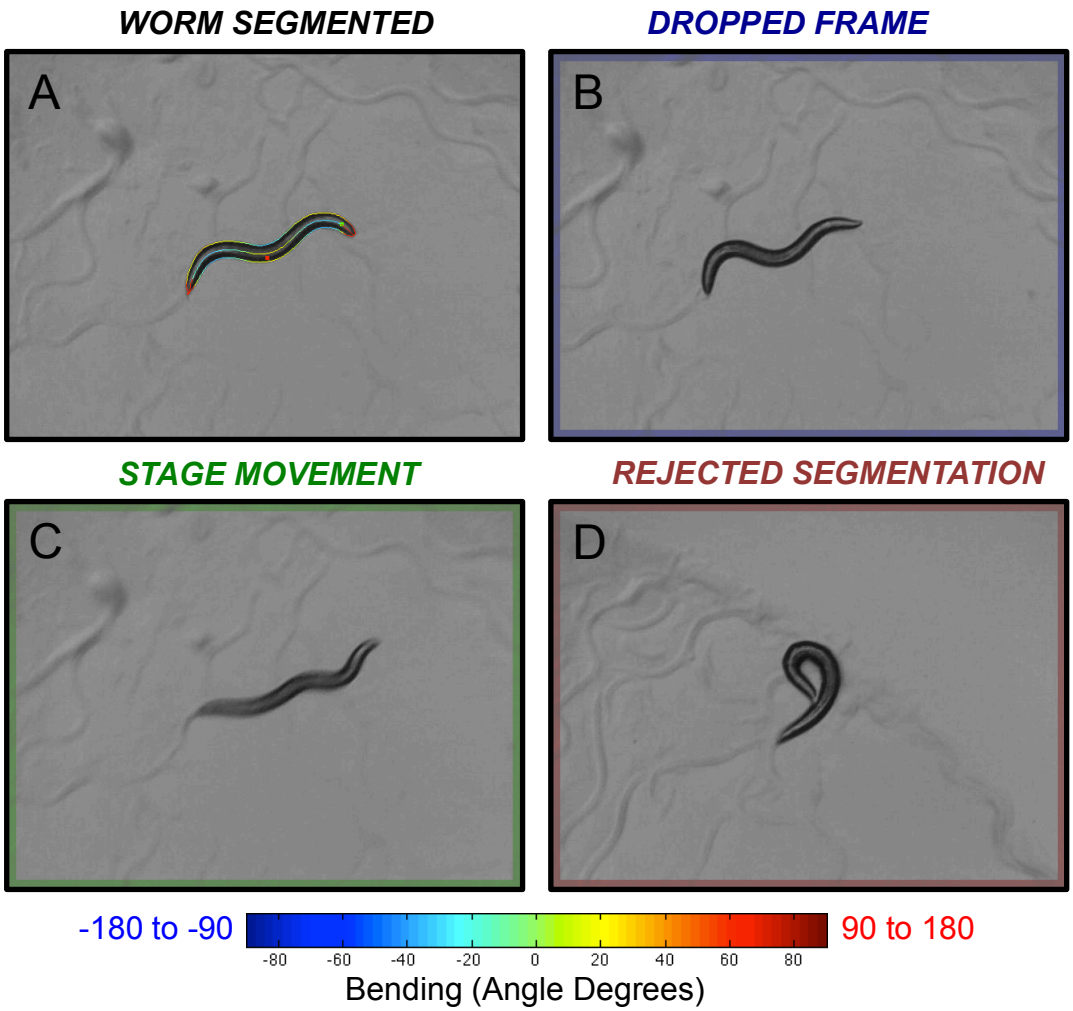


Figure 3.17: Video Review of Analysis (WT2 Screenshots)

WT2 video screenshots illustrating a method for quick manual review and verification of our automated worm measurements and analysis. A. A segmented video frame overlayed with the worm contour, skeleton, bending, head choice, and ventral side. As mentioned earlier, the contour and skeleton are drawn using a heat map to indicate their bending. Skeleton bending near the endpoints cannot be calculated and is colored in red for visibility. The head and ventral-side are labeled, respectively, by green and red squares. B. Dropped frames are outlined with blue borders for rapid identification during video playback. In place of a blank image, I use the last good video frame so as not to disrupt the movie. C. Stage movement frames are outlined with green borders for rapid identification during video playback. D. Frames for which segmentation was rejected are outlined with red borders for rapid identification during video playback.

3.5.2 Future Directions

3.5.2.1 Online, Searchable Database of Mutant Phenotypes and Individual Experiments

The experimental throughput of our WT2 system, its precision, and the volume of screened phenotypes facilitate the adoption of a central repository for worm behavior. We have developed an initial database to manage the growing collection of phenotypic information as well as offer an online method to search and access individual experiments along with statistical summaries of mutant sets (detailed in Chapter 4). The goal is to permit in-silico investigation of phenotypic data similar to that performed for collections of genomic sequence and expression data. This database uses WormBase to connect our phenotypic analysis with their extensive compendium of available genomic information. The database would significantly benefit from a rich tool set to manipulate the data it houses so as to explore a variety of scientific inquiry. We hope our own questions, along with user demand, will drive the development of this toolset. Furthermore, as new labs adopt WT2, their controlled experiments can also be absorbed into the database in order to rapidly grow the publicly-available strain collection in tandem with our own tracking efforts.

3.5.2.2 High-Precision, Multi-Worm Tracking

We are currently exploring an automated, multi-worm tracking extension to the software. My scale-invariant segmentation facilitates quick extension to measure multiple worms. With a protocol of controls, this data could be absorbed into the aforementioned database, albeit with slightly fewer and noisier measurements per worm. Figure 3.18A shows our automated segmentation function applied to a dyadic progression of identical worm images, at decreasing resolutions. Clearly, the image at 1/4 the resolution of our standard 640x480 protocol maintains much of the precision found in the original image's measurements. Figure 3.18B shows the segmentation function applied directly, without any modification, to small patches from a 640x480 resolution, multi-worm image (provided by Emanuel Busch). An automated multi-worm tracker involves minor modifications to the existing code. First, the segmentation algorithm requires a small adjustment to segment multiple thresholded

components, representing worms, as opposed to just a single worm component, as is done now. Second, a collision detection algorithm, of which several worm-specific ones have been published, must be employed to avoid confusing multiple, touching worms as one.

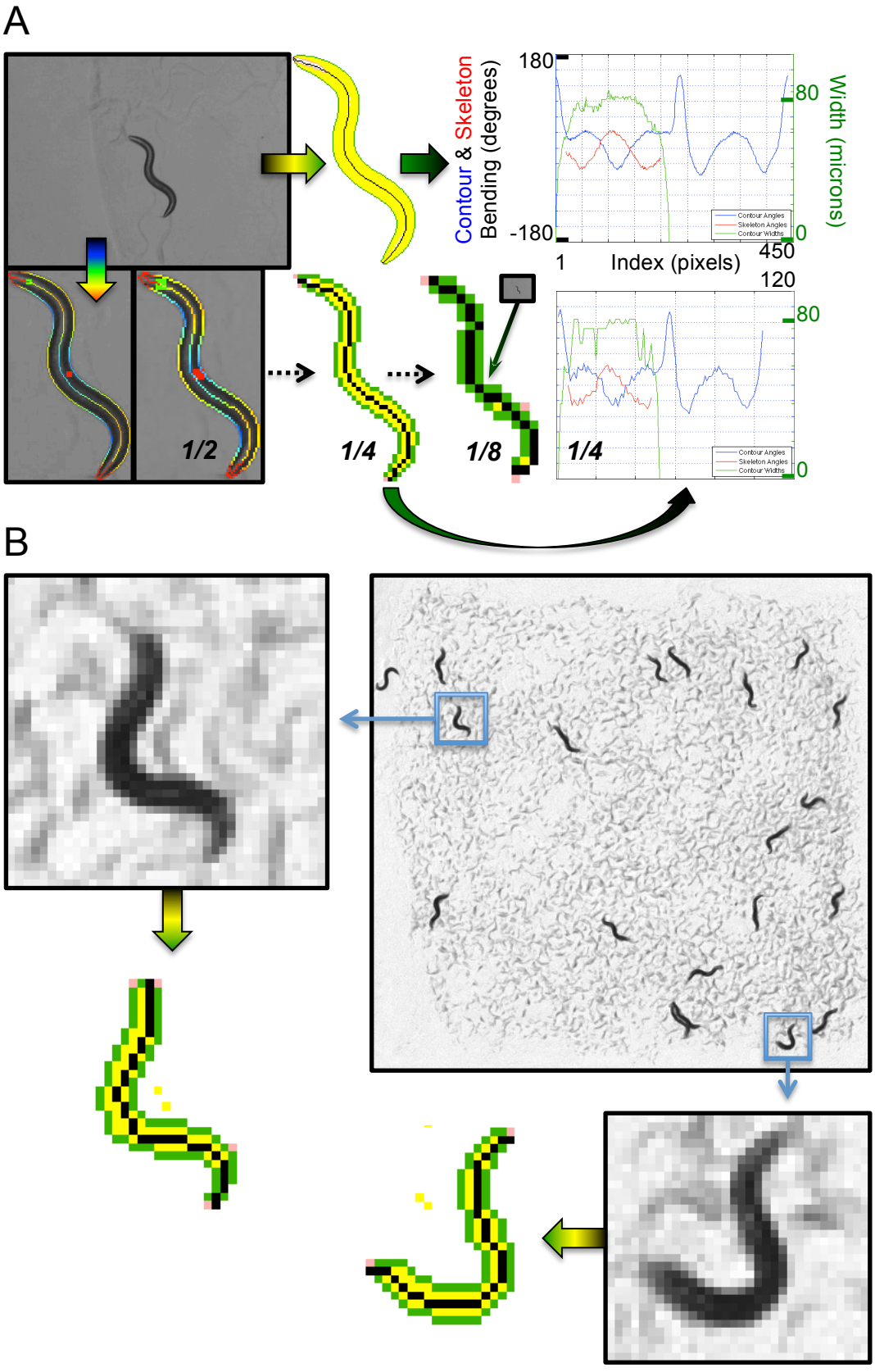


Figure 3.18: Low-Resolution Segmentation and Multi-Worm Tracking

Results from our automated segmentation and skeletonization algorithms at progressively lower resolutions and, their application in multi-worm tracking, at 640x480 resolution, with 19 worms in view. A. A 640x480 video image, from our regular data set, is shown alongside our corresponding segmentation and skeletonization. The image to the right displays the contour (labeled in green) and the skeleton (labeled in black). Further right, a plot shows the contour and skeleton bending (plotted in blue and red, respectively) as well as the contour widths at each skeleton point (plotted in green). The image below displays a WT2 screenshot. As mentioned earlier, the contour and skeleton are drawn using a heat map to indicate their bending. Skeleton bending near the endpoints cannot be calculated and is colored in red for visibility. The head and ventral-side are labeled, respectively, by green and red squares. To the right of the WT2 screenshot is the analysis progression when cutting the image and worm to 1/2 (320x240), 1/4 (160x120), and 1/8 (80x60) its original resolution. Beyond 1/2 resolution, the colorful heat map bending is distracting and, therefore, I switch to representing the thresholded worm in yellow, its contour in green, and its skeleton in black. The pink pixels where eliminated from the contour due to anti aliasing. A plot for the 1/4 resolution worm is shown below the full-resolution plot. Clearly, the algorithm is capable of recovering similar measurements at low resolutions. An image of the 1/8 worm is shown, to scale, relative to the original image. The 1/8 worm has merely enough pixels to independently identify the contour and skeleton. Its plot is noisy and, therefore, not shown. B. Our automated worm segmentation and skeletonization algorithms are applied to two patches of a 640x480 multi-worm image containing 19 worms. Even at this low resolution, the algorithm performs well in identifying the contour and skeletons for both worms.

3.5.2.3 Further Improvements

Our analysis provides a superset of the standard primary worm measurements and Chapter 4 details a secondary set of phenotypic features. There are plans to adapt published algorithms for coiled shapes [20] and egg laying [21] for use with WT2. With the contour sides' pixel-intensity statistics present, there may be an opportunity to detect the worm's dorsal-ventral orientation from video, rather than by experimenter as the current protocol dictates.

Improvements to the automated accuracy of head and tail detection would reduce the burden of manual data checks. Therefore, we are exploring using separate classifiers for short and long chunks, multiple classifiers to distinguish mutants with shared classification criteria, and switching to non-linear classifiers. Short chunks suffer a lack of discriminating head motions (which require multiple frames at a given spacing, as detailed in section 3.4.2 Head-Tail Detection) and therefore must rely on pixel intensities and, potentially, head-tail angularity to detect the head. Long chunks tend to have more statistics from head motion and are likely to benefit by lending more weight to this parameter. Separate classifiers for short and long chunks might exploit their differences to yield better head detection. Similarly, severely-paralyzed mutants and ones with morphological disturbances, such as small and long worms, may benefit from a different classifier than those resembling the wild type. Lastly, linear classifiers are by definition bound to their linear restriction whereas a non-linear classifier may yield better tuning when weighting criteria such as pixel intensities versus head motion. Beyond these particulars, we may find additional benefits in using the statistical confidence scores, assigned by our classifier, to make more refined decisions as to which data most urgently requires manual review.

Adaptation of new hardware and published computer vision techniques may address some of the limitations of WT2 (and similar worm tracking systems) as well as extend its capabilities into more complex phenotypic assessment. At present, we do not segment worms during stage movement. Despite interpolation for features such as velocity, the removal of these shapes is likely to bias phenotypic measurement towards less dynamic worm shapes. A camera with a global shutter, in combination with an algorithm to determine the motorized-stage location during stage movement,

would recover these lost worm shapes. This, in tandem with coiled-shape and egg-laying segmentation (discussed above, in this section), would likely recover a significant portion of the nearly 20% of unsegmented video frames. Beyond this, computer-vision techniques to detect the food lawn and/or lab techniques to grow lawns with a specified size and location, would provide the ability to discriminate worm phenotype with respect to the presence of food (specifically on food, off food, and at the food border). At present, the location of food is not discriminated and dependent behaviors likely show specific modalities with respect to food proximity. Lastly, extension of computer vision techniques and high-resolution optics to segment worm internals, such as the pharynx, gut, and internal sex-related structures (e.g., the gonads and, for hermaphrodites, developing oocytes within), would enable a new set of "deep features" that describe internal worm phenotypes available now only through human assessment of these structures and their dynamics.

3.6 Analysis Software Code

The WT2 analysis software is open source and written in Matlab (version 2010a). Comments and help within the functions provide details to understand and modify the code. The source code is accessible at http://sourceforge.net/projects/worm-tracking/ and receives periodic updates. Unless otherwise noted, the image processing functions were available through standard Matlab toolboxes (version 2010a) or created as needed. The videoIO toolbox by Gerald Dalley, available at the Matlab Central File Exchange (http://www.mathworks.com/matlabcentral/fileexchange/), significantly facilitated video manipulation within Matlab.

3.7 Bibliography

1. Baek, J.-H., Cosman, P., Feng, Z., Silver, J. & Schafer, W. R. Using machine vision to analyze and classify Caenorhabditis elegans behavioral phenotypes quantitatively. *Journal of Neuroscience Methods* **118**, 9–21 (2002).
2. Zhang, S. *et al.* Profiling a Caenorhabditis elegans behavioral parametric dataset with a supervised K-means clustering algorithm identifies genetic networks regulating locomotion. *Journal of Neuroscience Methods* **197**, 315–323 (2011).
3. Kamath, R. S. *et al.* Systematic functional analysis of the Caenorhabditis elegans genome using RNAi. *Nature* **421**, 231–237 (2003).
4. Pierce-Shimomura, J. T., Morse, T. M. & Lockery, S. R. The fundamental role of pirouettes in Caenorhabditis elegans chemotaxis. *Journal of Neuroscience* **19**, 9557–9569 (1999).
5. Hoshi, K. & Shingai, R. Computer-driven automatic identification of locomotion states in Caenorhabditis elegans. *Journal of Neuroscience Methods* **157**, 355–363 (2006).
6. Stephens, G. J., Johnson-Kerner, B., Bialek, W. & Ryu, W. S. Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* **4**, e1000028 (2008).
7. Leifer, A. M., Fang-Yen, C., Gershow, M., Alkema, M. J. & Samuel, A. D. T. Optogenetic manipulation of neural activity in freely moving Caenorhabditis elegans. *Nat Meth* **8**, 147–152 (2011).
8. Wang, W., Sun, Y., Dixon, S. J., Alexander, M. & Roy, P. J. An Automated Micropositioning System for Investigating C. elegans Locomotive Behavior. *Journal of the Association for Laboratory Automation* **14**, 269–276 (2009).
9. Geng, W., Cosman, P., Berry, C. C., Feng, Z. & Schafer, W. R. Automatic tracking, feature extraction and classification of C elegans phenotypes. *IEEE Trans. Biomed. Eng.* **51**, 1811–1820 (2004).
10. Cronin, C. J. *et al.* An automated system for measuring parameters of nematode sinusoidal movement. *BMC Genet.* **6**, 5 (2005).
11. Roussel, N., Morton, C. A., Finger, F. P. & Roysam, B. A computational model for C. elegans locomotory behavior: application to multiworm tracking. *IEEE Trans. Biomed. Eng.* **54**, 1786–1797 (2007).
12. Freeman, H. On the encoding of arbitrary geometric configurations. *Electronic Computers, IRE Transactions on* 260–268 (1961).
13. Otsu, N. A Tlreshold Selection Method from Gray-Level Histograms. *Automatica* **11**, 23–27 (1975).
14. Wu, K., Otoo, E. & Suzuki, K. Two strategies to speed up connected component labeling algorithms. (2008).
15. van Swinderen, B. *et al.* Goalpha regulates volatile anesthetic action in Caenorhabditis elegans. *Genetics* **158**, 643–655 (2001).
16. Zhao, B., Khare, P., Feldman, L. & Dent, J. A. Reversal frequency in Caenorhabditis elegans represents an integrated response to the state of the animal and its environment. *Journal of Neuroscience* **23**, 5319–5328 (2003).
17. Berri, S., Boyle, J. H., Tassieri, M., Hope, I. A. & Cohen, N. Forward locomotion of the nematode C. elegans is achieved through modulation of a single gait. *HFSP J* **3**, 186–193 (2009).
18. Brenner, S. The genetics of Caenorhabditis elegans. *Genetics* **77**, 71–94 (1974).

19. Daniels, S. A., Ailion, M., Thomas, J. H. & Sengupta, P. egl-4 acts through a transforming growth factor-ß/SMAD pathway in Caenorhabditis elegans to regulate multiple neuronal circuits in response to sensory cues. *Genetics* **156**, 123–141 (2000).
20. Huang, K.-M., Cosman, P. & Schafer, W. R. Machine vision based detection of omega bends and reversals in C. elegans. *Journal of Neuroscience Methods* **158**, 323–336 (2006).
21. Geng, W., Cosman, P., Palm, M. & Schafer, W. R. C. elegans egglaying detection and behavior study using image analysis. *EUROSIP J. Appl. Signal Processing* **2005**, 2229–2240

Chapter 4 : Worm Phenotypes

4.1 Abstract

The primary output of the nervous system is behavior; consequently, genetic screens that identify mutants with behavioral phenotypes have been invaluable in identifying genes underlying critical nervous system functions. With recent advances in machine vision and automation, high-content phenotyping of uncharacterized mutant strains has begun to provide a complementary route to link genes with behavior. In the previous chapters I introduced the Worm Tracker 2.0 (WT2) hardware and software. In this chapter I explore our results using WT2 to examine worm phenotypes. Using WT2, we have generated a database of behavioral data for 305 *C. elegans* strains (a subset of our larger data set), including 76 mutants with no previously described phenotype. This database consists of 9,203 short videos that have been segmented to extract behavior and morphology features. The database also includes summary statistics for 702 phenotypic measures with statistical comparisons to wild-type controls. Hierarchical clustering analysis reveals robust grouping of mutant strains known to affect common molecular pathways, suggesting that the clustering patterns of uncharacterized strains may provide insight into their function.

4.2 Introduction

This chapter is a modified version of the draft manuscript "A database of *C. elegans* behavioral phenotypes" authored by Eviatar Yemini, Laura Grundy, Tadas Jucikas, Andre Brown, and William Schafer. The initial abstract and discussion were written by William Schafer whilst the initial discussion was written by Andre Brown. The results and analysis are my own work, except when stated otherwise.

The principal output of the nervous system is motor behavior. Therefore, a variety of neural perturbations ultimately manifest as changes in motion. This fact has made locomotion a useful phenotype for neurogenetics in model organisms [1-6]. For the nematode *C. elegans*, the study of mutations that cause visible defects in spontaneous crawling on agar have given insight into neural functions as diverse as synaptic transmission, axon guidance, neuromodulation, and proprioception [7-11]. The affected genes code for a variety of proteins, including, for example, enzymes required for monoamine biosynthesis, several families of ion channels, neuropeptides, innexins (components of invertebrate gap junctions), and netrin receptors to name just a few.

Despite these successes, extending behavioral phenotyping to large-scale screens remains a challenge. Perturbing many genes, one at a time, does not necessarily result in phenotypic effects evident through quick manual observation. In fact, 85% of *C. elegans* genes have no reported phenotypic effect when knocked down using RNA interference [12]. This could be due to redundancy, knocking down a single gene often has no effect because other genes compensate; or because the targeted genes are required in specific environments which have not yet been assayed. Both of these mechanisms are likely to play a role in the small number of reported phenotypes. However, by measuring progeny number over several generations, rather than looking for obvious visible defects, it was found that most genes are required for wild-type fitness [12]. This suggests that the main contributing factor to missing phenotypes is in fact a *phenotyping gap*: knocking down most genes has an effect, it is just too subtle to see by manual observation.

To close this phenotyping gap, it is desirable to have a system that is capable of measuring phenotypes both extensively and intensively [13]. For behavior, extensive phenotyping requires measuring many parameters that quantify motion, posture, and path as well as the frequencies and intervals between relevant behaviors such as reversals and sharp turns. Achieving intensive sampling requires following individual worms with high temporal resolution over extended intervals. These conditions can be met using single-worm trackers that follow freely-behaving worms, using a motorized stage to keep the worm within the camera's field of view. Previously described single-worm trackers are relatively bulky and expensive, limiting their potential for large-scale tracking.

Here we report extensive and intensive behavioral phenotypes for 9,203 individuals, representing 305 strains of *C. elegans* (see Appendix A for the list of strains), collected using multiple inexpensive single-worm trackers running in parallel. Within our database are 76 mutants with no previously characterized phenotype, 15 genes with multiple allelic representation, and 13 double or triple mutant combinations (the majority of which are accompanied by single mutant representation as well). The videos are easily accessible online at the video-sharing website YouTube. The data are available online as well, with various degrees of processing, from the skeleton and outline coordinates to the time series of extracted features, their histograms, and an in-depth view of their summary statistics. For computational researchers, the database is a rich source of processed measures and raw data to develop new algorithms for segmentation, behavioral quantification, and bioinformatic approaches which link complex phenotypes with genetic perturbations. For neurogeneticists, the summary statistics and visualizations make it possible to identify behavioral phenotypes in mutants of interest. Among these we present phenotypes for 76 genes with no previously reported differences from wild type.

4.3 Methods

4.3.1 Tracking, Ventral Side Annotation, and Head Detection

Our worm features necessitate dorsal-ventral and head-tail distinctions. The worm's ventral side was annotated for each video by eye. We did not profile rolling mutants and therefore expected worms to maintain their dorsal-ventral orientation. Nevertheless, 126 random videos were examined and the worms therein found never to flip sides. Head-tail orientation was annotated automatically by software. 133 random videos (roughly 1% of our data, 2.25 million segmented frames) were examined, a collection of 100 from a quality-filtered set and 33 rejected by this filter (see section 4.3.4, for information regarding the filter's purpose and methods), representing a broad range of mutants (including several nearly motionless UNCs). Many of these include early videos which suffered multiple dropped frames and poor imaging conditions that were later improved. The head was correctly labeled with a mean and standard deviation of 94.39 ± 17.54% across individual videos and 95.6% of the video frames collectively. The results for the quality-filtered set are discussed separately in section 4.3.4.1.

4.3.2 Phenotypic Features

4.3.2.1 Phenotypic Features Overview

All feature formulas are computed from the worm's segmented contour and skeleton, using chain-code pixel lengths and absolute coordinates (detailed throughout Chapter 3). The skeleton and each side of the contour are scaled down to 49 points for feature computation (see Chapter 3 section 3.4.3). Wild-type worms have four quadrants of longitudinal, staggered bodywall muscles [14]. Each quadrant contains 24 such muscles with the exception of the ventral-left quadrant, which has 23. With a sampling of 49 points, the skeleton and contour sides have a well-defined midpoint. Moreover, since the worm is confined to two dimensions, its bodywall muscles present roughly 24 degrees of freedom (although in practice it seems to be far less [15]). With 49 points there are 2 samples per degree of freedom and, therefore, we expect to be sampling above the Nyquist rate for worm posture.

A common notation is used to define the body parts (Figure 4.1A). The head is controlled by the first four bodywall muscles, per quadrant -- approximately 1/6 the length of the worm [16]. Similarly, the neck is controlled by the next four bodywall muscles, per quadrant -- also approximately 1/6 the length of the worm. For this reason, I define the head as the first 1/6 of the worm and the neck as the next 1/6 of the worm (skeleton points 1-8 and 9-16, respectively). For symmetry, I define the tail and "hips", in a similar manner, on the opposite end of the worm. The tail is the last 1/6 of the worm and the hips are defined as the next 1/6 (skeleton points 42-49 and 34-41, respectively). The midbody is defined as the remaining middle 1/3 of the worm (skeleton points 17-33). For some features, the head and tail are further subdivided to extract their tips, the first and last 1/12 of the worm (skeleton points 1-4 and 46-49), respectively.

Frame-by-frame features are represented by top-level histograms and statistics as well as subdivisions exploring their values during forward, backward, and paused states. This is to measure behaviors that depend on the state of motion such as foraging amplitude, which is reduced during reversals in wild-type worms [17]. Many features are signed to reflect dorsal-ventral orientation, forward-backward trajectory, and other special cases (e.g., eigenworm projection) to capture any asymmetry. Finally, event-style features (coiling, turning, and motion states) are summarized using global and local measures. Gobal measures include the event frequency, the ratio of time spent within the event to the total experiment time, and a similar measure for the ratio of the distance covered within the event to the total distance traveled by the worm (when available). Local measures include the time spent in every individual event, the distance covered in each event (when available), and both the time and distance covered between each pair of successive events.

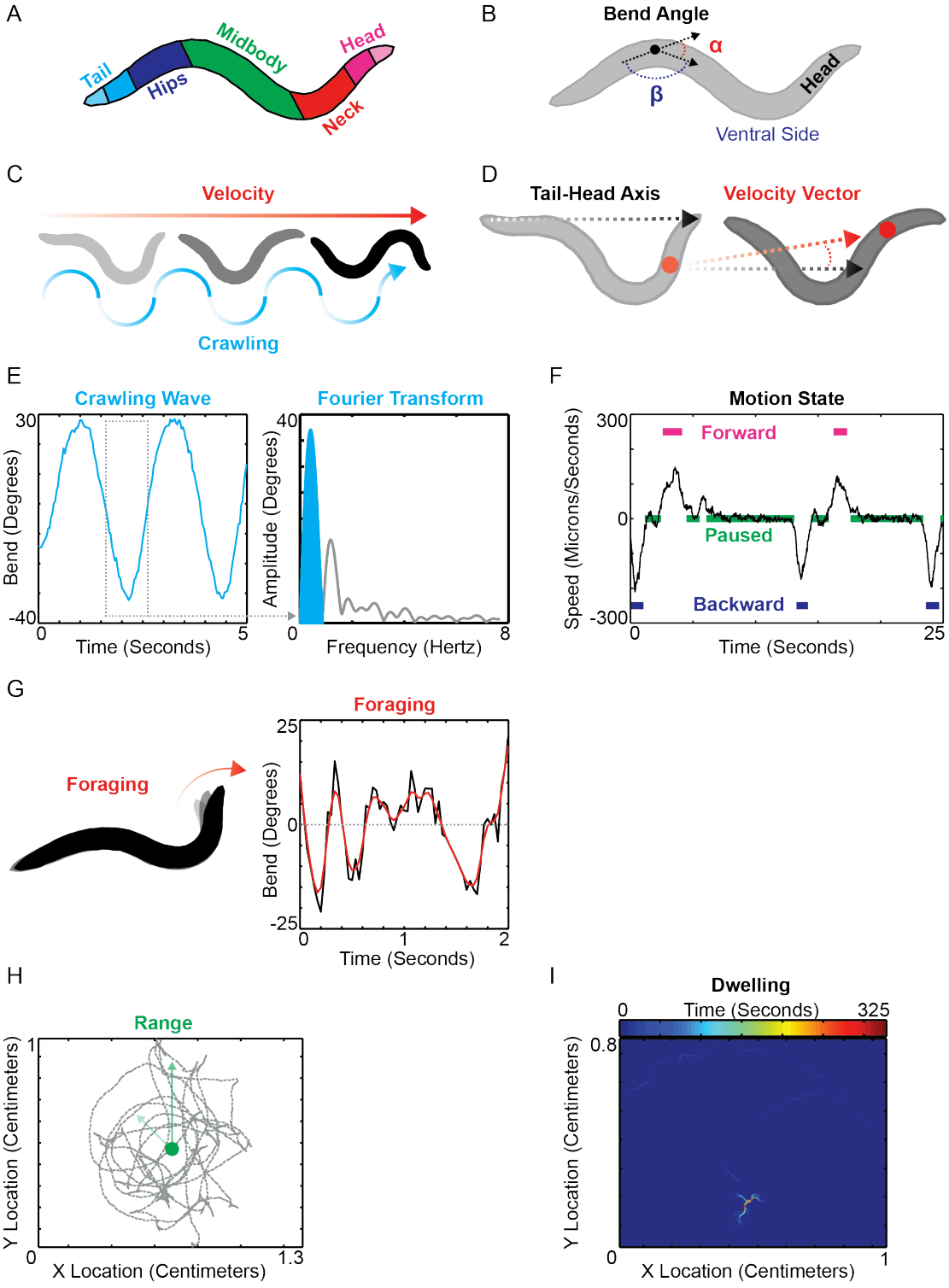


Figure 4.1: Phenotypic Feature Computation

Brief synopsis for several of the new feature algorithms. A. The locations of the body parts used for feature computation: head, neck, midbody, hips, and tail. Each body part encompasses 1/6 of the worm (measured along the contour and skeleton, independently), save for the midbody which is 1/3. The head and tail

are further split in half to create the head and tail tips, each 1/12 long. B. The bend angle (α) is the difference in tangent angles at each point; or, alternatively phrased, the supplementary angle (α) with respect to the angle formed by any three consecutive points (β). The bend angle is signed negatively whenever the ventral side is concave within the bend (as is the case for the bend shown). C-F. Methods of quantifying worm motion. C. A simple diagram represents worm velocity and the crawling wave. D. The velocity vector of a body part is measured relative to the head-tail axis. The velocity, per body part, is the vector of its respective centroid. E A crawling wave is represented as the sinusoidal wave of the bend angle at its associated body part. Note the dorsal-ventral asymmetry both in the amplitude and in the wavelength itself. The dotted window encloses a waveform used to measure an instantaneous crawling wave. The Fourier transform of the waveform reveals a strong peak. This peak defines the instantaneous crawling amplitude and frequency. F. A 25 second window of worm motion reveals two forward, three backward, and roughly five paused events. A short, small peak (between the second and third pauses) that may have been forward motion, remains unclassified due to ambiguity. G. Worm foraging is measured from the bend angle between the two sections of the head (panel A). The noisy foraging signal (black) is smoothed (red) by convolving with a Gaussian. The foraging amplitude is defined as the largest foraging bend angle measured, prior to crossing 0°. Foraging speed is simply the angular speed. Ventral and dorsal foraging is present within the first 1 second of the trace. The latter half of the trace displays the difficulties associated with measuring signal above the noise. At nearly 1 second in, the nose appears to quickly cross ventrally before rebounding dorsally. Smoothing eliminated the associated sign change in amplitude; although, a small, nearly 0° ventral amplitude may well be considered noise. H. The range is defined, per frame, as the distance of the worm's midbody from its final path centroid. The central dot displays the final path centroid. The two arrows display the range at early and late times within the experiment. I. The locations of worm dwelling are shown as a heatmap. A single location of dwelling dominates faint traces of the worm's path during motion.

4.3.2.2 Morphology Features

The morphology features are defined below.

1. Length. Worm length is computed from the segmented skeleton by converting the chain-code pixel length to microns.

2. Widths. Worm width is computed from the segmented skeleton. The head, midbody, and tail widths are measured as the mean of the widths associated with the skeleton points covering their respective sections. These widths are converted to microns.

3. Area. The worm area is computed from the number of pixels within the segmented contour. The sum of the pixels is converted to microns^2.

4. Area/Length.

5. Midbody Width/Length.

4.3.2.3 Posture Features

The posture features are defined below.

1. Bends. Worm bending is measured using the supplementary angles to the bends formed along the skeleton, with each skeleton point serving as the vertex to its respective bend (Figure 4.1B). The supplementary angle can also be expressed as the difference in tangent angles at the skeleton point. The supplementary angle provides an intuitive measurement. Straight, unbent worms have an angle of 0°. Right angles are 90°. And the largest angle theoretically possible, a worm bending back on itself, would measure 180°. The supplementary angle is determined, per skeleton point, using edges 1/12 the skeleton's chain-code length, in opposing

directions, along the skeleton. When insufficient skeleton points are present, the angle remains undefined (i.e., the first and last 1/12 of the skeleton have no bending angle defined). The mean and standard deviation are measured for each body segment. The angle is signed to provide the bend's dorsal-ventral orientation. When the worm has its ventral side internal to the bend, the bending angle is signed negatively.

2. Bend Count. The bend count is a rough measure of the number of bends along the worm. The supplementary skeleton angles are measured during segmentation and signed to reflect their dorsal-ventral orientation. These angles are convolved with a Gaussian filter, 1/12 the length of the skeleton, with a width defined by the Matlab "gausswin" function's default α of 2.5 and normalized such that the filter integrates to 1, to smooth out any high-frequency changes. The angles are then sequentially checked from head to tail. Every time the angle changes sign or hits 0°, the end of a bend has been found and the count is incremented. Bends found at the start and end of the worm must reflect a segment at least 1/12 the skeleton length in order to be counted. This ignores small bends at the tip of the head and tail.

3. Eccentricity. The eccentricity of the worm's posture is measured using the eccentricity of an equivalent ellipse to the worm's filled contour. The orientation of the major axis for the equivalent ellipse is used in computing the amplitude, wavelength, and track length (described below). This feature was coded by Andre Brown.

4. Amplitude. Worm amplitude is expressed in two forms: a) the maximum amplitude found along the worm body and, b) the ratio of the maximum amplitudes found on opposing sides of the worm body (wherein the smaller of these two amplitudes is used as the numerator). The formula and code originate from the publication "An automated system for measuring parameters of nematode sinusoidal movement" [18].

   The worm skeleton is rotated to the horizontal axis using the orientation of the equivalent ellipse and the skeleton's centroid is positioned at the origin. The

maximum amplitude is defined as the maximum y coordinate minus the minimum y coordinate. The amplitude ratio is defined as the maximum positive y coordinate divided by the absolute value of the minimum negative y coordinate. If the amplitude ratio is greater than 1, its reciprocal is used instead.

5. Wavelength. The worm's primary and secondary wavelength are computed by treating the worm's skeleton as a periodic signal. The formula and code originate from the publication "An automated system for measuring parameters of nematode sinusoidal movement" [18].

   The worm's skeleton is rotated as described above for the amplitude. If there are any overlapping skeleton points (the skeleton's x coordinates are not monotonically increasing or decreasing in sequence -- e.g., the worm is in an S shape) then the shape is rejected, otherwise the Fourier transform computed. The primary wavelength is the wavelength associated with the largest peak in the transformed data. The secondary wavelength is computed as the wavelength associated with the second largest amplitude (as long as it exceeds half the amplitude of the primary wavelength). The wavelength is capped at twice the value of the worm's length. In other words, a worm can never achieve a wavelength more than double its size.

6. Track Length. The worm's track length is the range of the skeleton's horizontal projection (as opposed to the skeleton's arc length) after rotating the worm to align it with the horizontal axis. The formula and code originate from the publication "An automated system for measuring parameters of nematode sinusoidal movement" [18].

7. Coils. Worm coiling (touching) events are found by scanning the video frame annotations. During segmentation, every frame that cannot be segmented is annotated with a cause for failure (see Chapter 3 section 3.3.1.5). Two of these annotations reflect coiling events. First, if there are fewer than two sharp ends on the contour (reflecting the head and tail) then the head and/or tail are obscured in a coiling event. Second, if the length between the head and tail on one side of the contour is more than double that of the other side, the worm has either assumed an

omega bend or is crossed like a wreath. Empirically, less than 1/5 of a second is a very fast touch and not usually reflective of coiling. Therefore, when a period of unsegmented video frames exceeds 1/5 of a second, and either of the coiling annotations are found, the event is labeled coiling.

8. Eigen Projections. The eigenworm amplitudes are a measure of worm posture. They are the projections onto the first six eigenworms which together account for 97% of the variance in posture. The eigenworms were computed from 15 N2 videos (roughly 3 hours of video, 1/3 of a million frames). The method originates from the publication "Dimensionality and dynamics in the behavior of *C. elegans*" [15] and was coded by Andre Brown.

   Briefly, 48 tangent angles are calculated along the skeleton and rotated to have a mean angle of zero. Principal components analysis is performed on the pooled angle data and we keep the 6 principal components (or eigenworms) that capture the most variance. The first eigenworm roughly corresponds to body curvature. The next two eigenworms are akin to sine and cosine waves encoding the travelling wave during crawling. The fourth eigenworm captures most of the remaining variance at the head and tail. Projected amplitudes are calculated from the posture in each frame. Even for the mutants, the data is always projected onto the N2-derived eigenworms.

9. Orientation. The worm's orientation is measured overall (from tail to head) as well as for the head and tail individually. The overall orientation is measured as the angular direction from the tail to the head centroid. The head and tail centroids are computed as the mean of their respective skeleton points.

   The head and tail direction are computed by splitting these regions in two, then computing the centroid of each half. The head direction is measured as the angular direction from the its second half (the centroid of points 5-8) to its first half (the centroid of points 1-4). The tail direction is measured as the angular direction from the its second half (the centroid of points 42-45) to its first half (the centroid of points 46-49).

4.3.2.4 Motion Features

The motion features are defined below.

1. Velocity. The worm's velocity is measured at the tip of the head and tail, at the head and tail themselves, and at the midbody. The velocity (Figure 4.1C-D) is composed of two parts, speed and direction (expressed as an angular speed). The velocity is signed negatively whenever the respective body part moves towards the tail (as opposed to the head).

   The head and tail tips' instantaneous velocity is measured at each frame using a 1/4 second up to a 1/2 second window. For each frame, I search for a start frame 1/4 of a second before and an end frame 1/4 second after to delineate the worm's instantaneous path. If the worm's location is not known within either the start or end frame, I extend the search for a known location up to 1/2 second in either direction. If the worm's location is still missing at either the start or end, the velocity is marked unknown at this point. The speed is defined as the distance between the centroids of the start and end frames (for the respective body parts) divided by the time between both frames. The direction is defined as the angle (between centroids) from the start to the end frame, relative to the worm's overall body angle, divided by the time between both frames. The worm's overall body angle is defined as the mean orientation of the angles, in the tail-to-head direction, between subsequent midbody skeleton points. The body angle is used to sign the velocity. If the head or tail tip's start-to-end angle exceeds 90°, clockwise or anti-clockwise, relative to the overall worm body angle, the motion is towards the tail. In this case both the speed and direction are negatively signed. The head, midbody, and tail velocity are computed identically except they use a 1/2 second up to a 1 second window for choosing their start and end frames.

2. Motion States. The worm's forward, backward, and paused motion states attempt to differentiate these event states unambiguously (Figure 4.1F). Therefore, ambiguous motion has no associated state.

The motion states are computed from the worm's velocity and length (described in section 4.3.2.2). Missing lengths are linearly interpolated between segmented frames. The following filtering criteria were chosen based on human labeling of events within a variety of N2 and mutant videos. The worm is defined in a state of forward motion when a period, more than half a second long, is observed wherein: a) the worm travels at least 5% of its mean length over the entire period; and, b) the worm's speed is at least 5% of its length, per second, in each frame. The worm must maintain this speed almost continuously with permissible interruptions of, at most, a quarter second (this permits quick contradictory movements such as head withdrawal, body contractions, and segmentation noise). The criteria for backward motion is identical except the worm must be moving backwards (the midbody speed must be negatively signed). The worm is defined in a paused state when a period, more than half a second long, is observed wherein the worm's forward and backward speed do not exceed 2.5% of its length, per second, in each frame. The worm must observe these speed limits almost continuously with permissible interruptions of, at most, a quarter second (once again, this permits quick contradictory movements).

3. Crawling. Worm crawling is expressed as both an amplitude and frequency (Figure 4.1C, E). These features are measured instantaneously at the head, midbody, and tail. The amplitude and frequency are signed negatively whenever the worm's ventral side is contained within the concave portion of its instantaneous bend.

   Crawling is only measured during forward and backward motion states. The worm bend mean angles (described in section 4.3.2.3) show a roughly periodic signal as the crawling wave travels along the worm's body. This wave can be asymmetric due to differences in dorsal-ventral flexibility or simply because the worm is executing a turn. Moreover the wave dynamics can change abruptly to speed up or slow down. Therefore, the signal is only roughly periodic and solely measured for its nearly instantaneous properties.

Worm bends are linearly interpolated across unsegmented frames. The motion states criteria (described earlier in this section) guarantee that interpolation is no more than 1/4 of a second long. For each frame, I search both backwards and forwards for a zero crossing in the bend angle mean – the location where the measured body part (head, midbody, or tail) must have hit a flat posture (a supplementary bend angle of 0°). This guarantees that we are observing half a cycle for the waveform. Crawling is bounded between 1/30Hz (a very slow wave that would not resemble crawling) and 1Hz (an impossibly fast wave on agar). If the window between zero crossings is too small, the nearest zero crossing is assumed to be noise and I search for the next available zero crossing in its respective direction. If the window is too big, crawling is marked undefined at the frame. Once an appropriate window has been found, the window is extended in order to center the frame and measure instantaneous crawling by ensuring that the distance on either side to respective zero crossings is identical. If the distances are not identical, the distance of the larger side is used in place of the zero-crossing distance of the smaller side in order to expand the small side and achieve a symmetric window, centered at the frame of interest.

I use a Fourier transform to measure the amplitude and frequency within the window described above. The largest peak within the transform is chosen for the crawling amplitude and frequency. If the troughs on either side of the peak exceed 1/2 its height, the peak is rejected for being unclear and crawling is marked as undefined at the frame. Similarly, if the integral between the troughs is less than half the total integral, the peak is rejected for being weak.

4. Foraging. Worm foraging is expressed as both an amplitude and an angular speed (Figure 4.1G). Foraging is signed negatively whenever it is oriented towards the ventral side. In other words, if the nose is bent ventrally, the amplitude is signed negatively. Similarly, if the nose is moving ventrally, the angular speed is signed negatively. As a result, the amplitude and angular speed share the same sign roughly only half the time. Foraging is an ambiguous term in previous literature, encompassing both fine movements of the nose as well as larger swings associated with the head. Empirically we have observed that the nose movements are aperiodic while the head swings have periodicity. Therefore, I measure the

aperiodic nose movements and term these foraging whereas the head swings are referred to as measures of head crawling (described earlier in this section).

Foraging movements can exceed 6Hz [19] and, at 20-30fps, our video frame rates are just high enough to resolve the fastest movements. By contrast, the slowest foraging movements are simply a continuation of the crawling wave and present similar bounds on their dynamics. Therefore, I bound foraging between 1/30Hz (the lower bound used for crawling) and 10Hz.

To measure foraging, the head is split in two (skeleton points 1-4 and 5-8) in order to measure the angle between these sections. To do so, the mean of the angle is measured between subsequent skeleton points along each section, in the tail-to-head direction. The foraging angle is the difference between the mean of the angles of both sections. In other words, the foraging angle is simply the bend at the head. Missing frames are linearly interpolated, per each skeleton point, for fragments up to 0.2 seconds long (4-6 frames at 20-30fps -- twice the upper foraging bound). When larger fragments are missing, foraging is marked undefined. Segmentation of the head at very small time scales can be noisy. Therefore, the foraging angles are smoothed by convolving with a Gaussian filter 1/5 of a second long (for similar reasons to those mentioned in frame interpolation), with a width defined by the Matlab "gausswin" function's default α of 2.5 and normalized such that the filter integrates to 1.

The foraging amplitude is defined as the largest foraging angle measured, prior to crossing 0°. In other words, the largest nose bend prior to returning to a straight, unbent position. Therefore, the foraging amplitude time series follows a discrete, stair-step pattern. The amplitude is signed negatively whenever the nose points towards the worm's ventral side. The foraging angular speed is measured as the foraging angle difference between subsequent frames divided by the time between these frames. To center the foraging angular speed at the frame of interest and eliminate noise, each frame is assigned the mean of the angular speed computed between the previous frame and itself and between itself and the next frame. The angular speed is signed negatively whenever its vector points towards the worm's ventral side.

5. Turns. Omega and upsilon turn events are computed similarly to a previously described method [20] but using skeleton bends instead of a single head-midbody-tail angle. Omega and upsilon turns are signed negatively whenever the worm's ventral side is sheltered within the concavity of its midbody bend. This feature was coded by Andre Brown.

   The worm bends (described in section 4.3.2.3) are used to find a contiguous sequence of frames (interruptible by coiling and other segmentation failures) wherein a large bend travels from the worm's head, through its midbody, to its tail. The worm's body is separated into three equal parts from its head to its tail. The mean supplementary angle is measured along each third. For omega turns, this angle must initially exceed 30° at the first but not the last third of the body (the head but not the tail). The middle third must then exceed 30°. And finally, the last but not the first third of the body must exceed 30° (the tail but not the head). This sequence of a 30° mean supplementary angle, passing continuously along the worm from head to tail, is labeled an omega turn event. Upsilon turns are computed nearly identically but they capture all events that escaped being labeled omega turns, wherein the mean supplementary angle exceeded 15° on one side of the worm (the first or last third of the body) while not exceeding 30° on the opposite end.

4.3.2.5 Path Features

The path features are defined below.

1. Range. The centroid of the worm's entire path is computed. The range is defined as the distance of the worm's midbody from this overall centroid, in each frame (Figure 4.1H).

2. Dwelling. The worm dwelling is computed for the head, midbody, tail, and the entire worm (Figure 4.1I). The worm's width is assumed to be the mean of its head, midbody, and tail widths across all frames. The skeleton's minimum and

maximum location, for the x and y axes, is used to create a rectangular boundary. This boundary is subdivided into a grid wherein each grid square has a diagonal the same length as the worm's width. When skeleton points are present on a grid square, their corresponding body part is computed as dwelling within that square. The dwelling for each grid square is integrated to define the dwelling distribution for each body part. For each body part, untouched grid squares are ignored.

3. Curvature. The path curvature is defined as the angle, in radians, of the worm's path divided by the distance it traveled in microns. The curvature is signed to provide the path's dorsal-ventral orientation. When the worm's path curves in the direction of its ventral side, the curvature is signed negatively.

   The worm's location is defined as the centroid of its body, with the head and tail removed (points 9-41). The head and tail are removed because their movement can cause large displacements in the worm's centroid. For each frame wherein the worm's location is known, I search for a start frame 1/4 of a second before and an end frame 1/4 second after to delineate the worm's instantaneous path. If the worm's location is not known within either the start or end frame, I extend the search for a known location up to 1/2 second in either direction. If the worm's location is still missing at either the start or end, the path curvature is marked unknown at this point.

   With three usable frames, we have an approximation of the start, middle, and end for the worm's instantaneous path curvature. The difference in tangent angles between the middle to the end and between the start to the middle are used to delineate the path direction. The path distance is measured as the integral of the distance traveled, per frame, between the start and end frames. When a frame is missing, the distance is interpolated using the next available segmented frame. The instantaneous path curvature is then computed as the angle divided by the distance. This path curvature is signed negatively if the angle curves in the direction of the worm's ventral side.

4.3.3 Phenotypic Ontology

The phenotypic ontology (available online within the PDF files -- see section 4.3.4.2) attempts to find significant features and reduce our large set of statistical measures to several simple terms. Each ontological term has a prefix indicating whether all significant measurements agree that the feature is greater (+), less (-), or different (Δ) than the control. A feature is said to be different than its control whenever the magnitude has no direct meaning (e.g., asymmetry does not translate to a clear description of the measurement being less nor greater than the control) or its measures do not express a simple magnitude (e.g., the strain pauses with greater frequency but spends less time in each paused event). Each term also has a suffix indicating the minimum q-value (significance) found for the term's defining measures (* when $q \leq 0.05$; ** when $q \leq 0.01$; *** when $q \leq 0.001$; and, **** when $q \leq 0.0001$). The q-value is a p-value replacement that corrects for multiple testing [21]. The ontology terms are as follows:

1. Length. The worm's length.

2. Width. The worm's head, midbody, and/or tail width.

3. Area. The worm's area if neither the "Length" nor "Width" were found significant.

4. Proportion. The worm's area/length and/or width/length if neither the "Length", "Width", nor "Area" were found significant.

5. Head Bends. The worm's head bend mean and/or standard deviation.

6. Tail Bends. The worm's tail bend mean and/or standard deviation.

7. Posture Amplitude. The worm's maximum amplitude and/or amplitude ratio.

8. Posture Wavelength. The worm's primary and/or secondary wavelength.

9. Posture Wave. The worm's track length if neither the "Posture Amplitude" nor the "Posture Wavelength" were found significant.

10. Body Bends. The worm's eccentricity, its number of bends, and/or its neck/midbody/hips bend mean and/or standard deviation; only if neither the "Posture Amplitude", "Posture Wavelength", nor "Posture Wave" were found significant.

11. Pose. The worm's eigenworm projections if neither the "Head Bends", "Body Bends", "Tail Bends", "Posture Amplitude", "Posture Wavelength", nor "Posture Wave" were found significant.

12. Coils. The worm's coiling event details.

13. Foraging. The worm's foraging amplitude and/or angular speed.

14. Forward Velocity. The worm's forward (positive) velocity vector.

15. Backward Velocity. The worm's backward (negative) velocity vector.

16. Velocity. The worm's velocity vector magnitude and/or asymmetry if neither the "Forward Velocity" nor "Backward Velocity" were found significant.

17. Head Motion. The worm's head-tip and/or head velocity vectors if neither the "Foraging", "Forward Velocity", nor "Backward Velocity" were found significant.

18. Tail Motion. The worm's tail-tip and/or tail velocity vectors if neither the "Forward Velocity" nor "Backward Velocity" were found significant.

19. Forward Motion. The worm's forward motion event details.

20. Pausing. The worm's pausing event details.

21. Backward Motion. The worm's backward motion event details.

22. Crawling Amplitude. The worm's crawling amplitude.

23. Crawling Frequency. The worm's crawling frequency.

24. Turns. The worm's omega and/or upsilon event details.

25. Path Range. The worm's path range.

26. Path Curvature. The worm's path curvature.

27. Dwelling. The worm's dwelling if its "Pausing" was not found significant.

4.3.4 Feature Files

4.3.4.1 Feature Files Overview

The features are presented within four types of files available online at: http://wormbehavior.mrc-lmb.cam.ac.uk/ (the website was created by Tadas Jucikas, while the content originates from my analysis of Laura Grundy's worm tracking). PDF files provide a visual summary of the data, per strain. CSV files provide a spreadsheet of the data, per strain. And, three types of MAT files are provided to access the strain data and statistics as well as the skeleton, contour, and feature data for each individual experiment, per frame.

The MAT files, per worm, are available for every experiment. To ensure high-quality experimental data, strain collections of experiments and controls were filtered and only include worm videos of at least 20fps, 14-15 minutes long, wherein at least 20% of the frames were segmented. We only include data collected Monday through Saturday, from 8am to 6pm. This resulted in a mean of 24 worms per strain with a minimum of 12 and a standard deviation of 14. Controls were chosen from the filtered N2 data collection by matching the strain collections to controls performed within the same week. This resulted in a mean of 63 controls, per strain collection, with a

minimum of 18 and a standard deviation of 29. I examined 100 videos (roughly 2 million frames) from our filtered collection and found that the head was correctly labeled with a mean and standard deviation of 95.17 ± 17.5% across individual videos and 95.69% of the frames collectively.

Outliers can compress visual details in their corresponding histograms. For this reason, the strain collections underwent one more filtering step prior to inclusion in the PDF files. Experiments were discarded wherein any of the worm data exceeded reasonable bounds of 250 to 2000 microns for length, 25 to 250 microns for width, and/or -1000 to 1000 microns/seconds for the midbody speed. Outliers were seldom found. Overall, 49 non-control worms were lost from a collection of 7,529 experiments. No strain collection lost more than 2 worms. The N2 collection of controls lost 5 worms from its total of 1,218 experiments. The CSV files and MAT statistical-significance files are available for both the primary quality-filtered data sets and the secondary, outlier-filtered data sets.

Shapiro-Wilk testing (performed using the “swtest” function by Ahmed Ben Saïda -- see section 4.3.7) of each feature measure (with corrections for multiple comparisons) showed a maximum q-value of 0.0095 over our collective N2 data set, indicating that, in aggregate, none of the measures are normally distributed. Further testing across all strain collections (which have far lower sampling than the N2 collective) and their controls, indicated a roughly 2:1 ratio of normal to non-normal distributions, rejecting the null hypothesis of normality at a q-value of 0.05. Therefore, I chose to test strain measurements against their controls by using the non-parametric Wilcoxon rank-sum test (with the null hypothesis that both sets of mean values were drawn from the same distribution). In four strains, at least one measure was detected exclusively in either the strain or its control, meaning the measurement was always observed within one set and never in the other (e.g., some strains never perform reversals). When this occurred, a Fisher’s exact test (performed using the “fexact” function by Michael Boedigheimer -- see section 4.3.7) was used to measure the probability that our sets were drawn from the same distribution of observed and unobserved events. Occasionally, features measurements had insufficient values for testing due to low sampling (e.g., omega-turn events), these measures were ignored and their p-value marked as undefined. In total, 702 measurements (see Appendix B

for the list of feature measurements) were obtained for each of 305 strains in addition to collections of our lab-stock N2 worms by hour (9am-4pm, with 8am and 5pm discarded due to very low sampling), weekday (Tuesday-Friday, with Monday and Saturday discarded due to very low sampling), and month (January-December). I used False-Discovery Rate (FDR) to correct for nearly 702 measures by 329 groups and transform the p-values to their q-value equivalents [21].

The unfiltered histograms, presented within individual MAT files, were constructed by choosing standard bin resolutions (widths and centers) that resulted in roughly 10^3 bins, per feature, for our lab-stock N2 data. When plotting histograms, a common formula is used to downsample the bins. The square root of the total number of data samples contributing to the collective histogram is computed. If this value is less than the number of bins available, the histogram is downsampled to reduce the number of bins to the nearest integer at or below the computed square root. When multiple histograms are plotted together, the smallest common bin size is used to downsample all the histograms to the same bin width and centers.

4.3.4.2 PDF Files

The PDF (portable document format) files include five sections: a) a table of contents and overview of the results, b) a short summary of the most important features, c) the details for every feature, d) traces of the worm paths, and e) a reference with the experimental methods. Each page uses a color scheme to provide quick visual summaries of its results. All pages display tabs, on the right side, that explain their color scheme. The initial summary page of histograms (page 2) displays an example histogram that acts as a guide to understanding histogram plots and the statistics displayed in their titles. The page formats are as follows:

1. The table of contents details the layout of the PDF file. All feature measures are shown alongside their minimum q-value and a page number for details. The table of contents page also shows an overview with the experiment annotation and its phenotypic ontology (see section 4.3.3).

2. There are three summary pages. These pages show important feature histograms, with the collective experiments in color and their controls in gray. The background color, for the histogram plots, indicates the minimum q-value significance for the plotted feature. The title of each plot provides several statistical measures for the experiment and control collections. An example histogram, at the beginning of the first summary page, provides a reference to interpret the aforementioned statistical measures. Significant measures, with $q \leq 0.05$, are marked in bold font within the plot title.

   The crawling frequency, worm velocity, foraging speed, all event features, path range, and dwelling are shown on a pseudo log-value scale to improve readability within their small summary histograms. This pseudo log-value scale is achieved by taking the magnitude of the data values (to avoid complex numbers resulting from the logarithms of any negative numbers), translating the magnitude by 1 (to avoid the logarithms of any values less than 1, which would invert the sign of the data), taking the logarithm, then re-signing the formerly negative data values.

3. The detail pages present a detailed view of the histograms for every feature. They follow a similar format to the summary pages except that they never use a log scale for feature values. The title of each plot provides a large set of statistical measures. The control values are shown between square brackets. The statistical values include: a) the number of worms providing measurements (“WORMS”); b) the number of measurements sampled for the collection of worms (“SAMPLES”); c) the mean of the data (“ALL”) alongside the SEM and, when the data is signed, the means for the absolute data values (ABS), positive data values only (“POS”), and negative data values only (“NEG”) alongside their SEMs as well; d) the p-value results using Wilcoxon rank-sum testing and q-value results using False Discovery Rate correction (for multiple tests across 329 strain collections by 702 feature measurements), both labeled accordingly (respectively “p” and “q”); e) event features also display their mean frequency (“FREQ”), the mean percentage of time spent in the event relative to the total experiment time (“TIME”), and, when available, the mean percentage of distance traveled during the event relative to the total distance covered during the experiment (“DIST”).

Features that have motion-state subdivisions are shown with an additional view wherein all motion-state histograms, and their integral histogram, are shown on the same plot. This allows one to quickly distinguish behaviors dependent on the motion state. Event features have an additional view wherein event and inter-event measures are plotted on a log-probability scale to make outlying events more visible.

4. The path trace pages display the paths for the worms' head, midbody, and tail and heatmaps for the midbody speed and foraging amplitude. Pages with the head, midbody, and tail include a tab, on the right side, to interpret the color associated with each body part. Pages with heatmaps include a tab, on the right side, to interpret the color gradient. On the path trace plots, the start and end of each path is denoted by a gray and black worm, respectively. Moreover, on each plot, the locations for coiling events are marked by a "+" and those for omega turns are marked by an "x". Body part plots use transparency to roughly indicate dwelling through color opacity.

   The first page of each path trace shows a collection of up to 24 worms (when available) overlayed for both the experiment and control collections, at the same scale. These overlays provide a quick view of features such as relative path sizes, food leaving behaviors, and the relative locations for coiling events and omega turns. When more than 24 worms are available we sort the worms by date, then choose 24 from the first to the last experiment at regular intervals. The paths are rotated to align their longest axis vertically, and then centered using the minimum and maximum x and y path values, per worm.

   The next page of the path traces shows each collection of 24 paths on the same plot, ordered roughly from largest to smallest, spaced out to avoid any overlay. The experiments and their controls use independent scales. This ordered plot provides a quick view to distinguish salient characteristics of experiment versus control paths (e.g., bordering at the edge of the food lawn).

   The subsequent pages for each path trace show the 24 individual worm paths, for the experiments and their controls, without rotation, sorted by date.

5. The method pages provide a reference for the details of our methodology.

4.3.4.3 CSV Files

The CSV (comma separated value) files are compatible with popular spreadsheet programs (e.g., Microsoft Excel, Apple iWork Numbers, OpenOffice, etc.). Each experimental collection is accompanied by four CSV files presenting the data and statistics for all morphology (<filename>.morphology.csv), posture (<filename>.posture.csv), motion (<filename>.motion.csv), and path features (<filename>.path.csv). The CSV files present the strain, genotype, and date for the experimental strain and control worms. The mean and standard deviation are presented for each feature measure, per worm and for the collection of experiments and controls. The p and q-values are presented for the strain as a whole (the null hypothesis is that experiment and control worms are drawn from the same distribution) and for each feature measure individually. These p and q values are shown for both the non-parametric Wilcoxon rank-sum test and the normal-distribution Student's *t*-test (unpaired samples with unequal variance). The Shapiro-Wilk test for normality (with associated p and q values) is also shown for each measure. Correction for multiple testing (the q-values) was performed over our entire set of 329 groups of strain collections by 702 measures. For the Shapiro-Wilk normality test, correction for multiple comparisons included an additional 329 group-specific controls by 702 measures.

4.3.4.4 MAT Files

Each experiment is represented in a MAT, HDF5-formatted file (Hierarchical Data Format Version 5 -- an open, portable, file format with significant software support). HDF5 files are supported by most popular programming languages including Matlab, Octave (a free alternative to Matlab), R, Java, C/C++, Python, and many other environments. These experiment files contain the time-series feature data for an individual worm. Additionally, each strain collection of experiments and their

collection of controls are also represented in a single HDF5, MAT file. These strain files contain histogram representations and summary statistics (but not significance) for the collective experiments. Finally, the statistical significance, for our entire collection of mutants, is presented in a single HDF5, MAT file.

The first two MAT file types, individual experiments and strain collections, share a similar format. The individual experiment files present the feature data as a time series. They also include the full skeleton and the centroid of the contour, per frame, permitting novel feature computations. The strain collections present the data in summary and in histograms. The format for both file types is two top-level structs, "info" ("wormInfo" for the strain collections) and "worm", which contain the experimental annotation and data, respectively.

The "info" struct contains the experimental annotation. For the strain collections, the "info" from each experiment is collected into an array of structs called "wormInfo". Both variables share the same format with the following subfields:

1. wt2. The Worm Tracker 2.0 version information.

2. video. The video information. The video "length" is presented as both "frames" and "time". The video "resolution" is in "fps" (frames/seconds), pixel "height" and "width", the ratio of "micronsPerPixel", and the codec's "fourcc" identifier. The video frame "annotations" are presented for all "frames" with a "reference" specifying the annotation's numerical "id", the "function" it originated from, and a "message" describing the meaning of the annotation.

3. experiment. The experiment information. The "worm" information is presented for its "genotype", "gene", "allele", "strain", "chromosome", "sex", "age", the "habituation" time prior to recording, the location of its "ventralSide" in the video (clockwise or anti-clockwise from the head), the "agarSide" of its body (the body side touching the agar), and any other worm "annotations". The "environment" information is presented for the experiment conditions including the "timestamp" when the experiment was performed, the "arena" used to contain the worm (always a low-peptone NGM plate for the data presented here), the "food" used

(e.g., OP50 *E. coli*), the "temperature", the peak wavelength of the "illumination", any "chemicals" used, the "tracker" on which the experiment was performed (a numerical ID from 1 to 8), and any other environmental "annotations".

4. files. The name and location for the analyzed files. Each experiment is represented in a "video" file, "vignette" file (a correction for video vignetting), "info" file (with tracking information, e.g., the microns/pixels), a file with the log of "stage" movements, and the "computer" and "directory" where these files can be found.

5. lab. The lab information where the experiment was performed. The lab is represented by its "name", the "address" of the lab, the "experimenter" who performed the experiment, and any other lab-related "annotations".

The "worm" struct contains experimental data. The individual experiments contain the full time series of data along with the worm's skeleton and the centroid of its contour, per frame. The strain collections contain summary data and histograms in place of the time-series data. Both files share a similar initial format with the following subfields:

1. morphology. The morphology features. The morphology is represented by the worm's "length", its "width" at various body locations, the "area" within its contour, the "widthPerLength", and the "areaPerLength".

2. posture. The posture features. The worm's posture is represented by its bend count in "kinks", measures of the "bends" at various body locations (computed as both a "mean" and standard deviation, "stdDev"), its "max" "amplitude" and its "ratio" on either side, its "primary" and "secondary" "wavelength", its "trackLength", its "eccentricity", its "coils", the orientation "directions" of various body parts, and its six "eigenProjections". Individual experiment files also contain the "skeleton" "x" and "y" coordinates, per frame.

3. locomotion. The motion features. Worm motion states are represented by "forward", "backward", and "paused" events, the "speed" and angular "direction" of the "velocity" for various body parts, the "amplitude" and "frequency" of the

crawling “bends” for various body parts, as well as the “foraging” “bends” which are measured in an “amplitude” and “angleSpeed”, and the “turns” associated with “omega” and “upsilon” events. Individual experiment files also contain a “motion” state “mode” with values distinguishing forward (1), backward (-1), and paused (0) states, per frame.

4. path. The path features. The path is represented by its “range”, “curvature”, and the dwelling “duration” for various body parts. Individual experiment files also contain the “x” and “y” “coordinates” of the contour’s centroid. Moreover, the individual experiment files present the “duration” as an “arena” with a “height”, “width”, and the “min” and “max” values for the “x” and “y” axes of the arena. The arena can be transformed to a matrix using the given height and width. The duration of the worm and body parts are represented as an array of “times” spent at the “indices” of the arena matrix.

All events are represented by their “frequency” and either their “timeRatio” (the ratio of time in the event type to the total experiment time) or, if the worm can travel during the event, the “ratio.time” (equivalent to “timeRatio”) and “ratio.distance” (the ratio of the distance covered in the event type to the total distance traveled during the experiment). The individual experiment files represent each event as “frames” with a “start” frame, “end” frame, the “time” spent in this event instance, the “distance” traveled during this event instance (when available), the “interTime” till the next event, and the “interDistance” traveled till the next event. The strain collection files summarize these fields, excluding the individual “frames” and their “start” and “end”.

The strain collection files present the data for each feature within a “histogram” (as opposed to the individual experiment files which simply use a time-series array of values). Furthermore, when a feature can be subdivided by motion state, sub histograms are included for the “forward”, “backward”, and “paused” states. All histograms contain the “PDF” (probability distribution function -- for the special case of this MAT file, as opposed the acronym used herein this chapter) for each of their “bins” (centered at the associated feature’s values). All histograms also contain the “resolution” (width) of their bins, whether or not there “isZeroBin” (would one of the

bins be centered at 0?), and whether or not the feature "isSigned" (can the feature values be negative?).

Finally, the strain collection files present their data in three types of fields: a) individually as the "data" per experiment, b) summarized over the "sets" of experiments and, c) aggregated in "allData" as if we ran one giant experiment instead of our sets. In other words, "sets" weights each experiment identically whereas "allData" weights every frame, across all experiments, identically. The data is always represented as both a "mean" and "stdDev" (standard deviation). The mean and standard deviation are always computed for "all" the data. When the data is signed, the mean and standard deviation are also computed for the data's "abs" (absolute value), "pos" (only the positive values), and "neg" (only the negative values). The format for the three types of data is as follows:

1. data. The individual data for every experiment is presented in arrays (in the same order as the "wormInfo" experiment annotations). The array data presents each experiment's individual "mean", "stdDev", the number of "samples" measured, and the experiment's data "counts" for each one of the histogram's "bins".

2. sets. The data for the set of experiments is presented as the "mean", "stdDev", and "samples" (the number of experiments) of the collected set.

3. allData. The aggregate of all data measurements, as if the collection of videos were instead one long, giant video, is presented as a "mean", "stdDev", the total "samples" (the total number of frames wherein the data was measured), and the aggregate of "counts" for each one of the histogram's bins.

4.3.4.5 Statistical Significance MAT File

The statistical significance for all strains is collected into a single MAT file. This file contains three top-level structs with information for both the "worm" and "control" collections as well as the "dataInfo" necessary to interpret the included

matrices of data. The matrices are organized as rows of strains and columns of feature measures. The "worm" struct has the following subfields:

1. info. The worm information for each strain collection presented as their "strain", "genotype", "gene", and "allele".

2. stats. The statistics for each strain collection presented, for every feature measure, as their "mean", "stdDev" (standard deviation), "samples" (the number of worms providing a measurement for the feature – e.g., not all worms execute omega turns), and "zScore" relative to the control (a simple normalization to the control -- note that the collection of N2 controls has no zScore). Measurements exclusively found in the experimental group have a zScore of infinity and those found exclusively found in the control are -infinity. Furthermore, we include Shapiro-Wilk tests of data normality, per measure, in "pNormal" and correction for multiple testing, using their False-Discovery rate q-value replacements, in "qNormal". The q-values are computed across all measures per "strain" and their associated controls (roughly 1404 tests) and across "all" strain and control measures collectively (roughly 329 by 1404 tests).

3. sig. The statistical significance for each strain collection is presented, for every feature measure, as their "pTValue" (Student's *t*-test p-value, unpaired samples with unequal variance) and "pWValue" (Wilcoxon rank-sum test p-value). The "qTValue" and "qWValue" represent the False-Discovery rate q-value replacements for the "pTValue" and "pWValue" respectively. The q-values are computed across all measures per "strain" (approximately 702 tests) and across "all" strains and measures collectively (approximately 329 by 702 tests). The collection of N2s has no associated significance.

The "control" struct contains the control "stats" in an identical format to the "worm" struct "stats", but without the "zScores".

The "dataInfo" provides information for each column of the feature measure matrices used in the "worm" and "control" structs. Each feature measure has a "name", a "unit" of measurement, titles for three possible subdivisions ("title1",

"title2", and "title3" – the title of the feature itself, its motion state, and its signed subdivision), helpful indexed offsets for these titles ("title1I", "title2I", and "title3I"), an associated struct "field" to locate the feature in our other MAT files, the corresponding "index" for the struct field (e.g., the six eigenworm projections are represented in a field, as a 6-element array), "isMain" (is this the main feature as opposed to a subdivision of a main feature?), the feature "category" (morphology "m", posture "s", motion "l", path "p"), the feature "type" (simple data "s", motion data "m", event summary data "d", event data "e", inter-event data "i"), the feature "subtype" (none "n", forward motion state "f", backward motion state "b", paused state "p", event-time data "t", event-distance data "d", event-frequency data "h"), and information regarding the feature's "sign" (the feature is signed "s", unsigned "u", is the absolute value of the data "a", contains only positive data values "p", contains only negative data values "n").

4.3.5 Testing strain and group significance

Wilcoxon rank-sum was used to test the differences between each group's features and the wild type (329 groups of strains and time-based N2 groups by 702 measures), and Fisher's exact test was used for measurements found exclusively in the experiment or control groups. False-discovery rate was controlled across these comparisons, converting p-values to their q equivalents [21]. Each group was assigned its minimum q-value as a measure of group significance. Under this paradigm, every group was found to be significantly different from the wild type at $q \leq 0.05$. To ensure this was not an artifact of our methodology we chose a second method to verify our result. Since sparse sampling led to a difficulty in assessing normality, and since dimensionality outnumbered observations for all but the lab N2 data set, we measured the Hotelling T^2 statistic using a shrinkage estimation of the covariance matrix and a permutation test to determine p-values [22] -- translated to Matlab code by Andre Brown. Four strains had exclusive measures that would be penalized as a result of this method due to the absent values. For these groups, the minimum p-value associated with their Fisher's exact test was used in place of computing the MANOVA. For all other groups, we ran 10,001 permutations. The p-values were corrected, controlling the false-discovery rate, to their q equivalents. Once again, every group was found to be

significantly different from the wild type at $q \leq 0.05$, bearing out the results of our primary methodology (see Appendix C for a list of strain and group significance).

4.3.6 Clustering

Each strain's measures were collapsed to the mean of their observations and normalized to their wild-type controls (by subtracting the control mean and scaling by its variance). We also added the hourly, daily, and monthly lab N2 groups to the clustering as a control. Each of these groups was normalized against all lab N2s not present within their group. For example, the 9AM group was controlled against all lab N2s recorded from 10AM onwards. The clustering results, which bring most the lab N2s together in an exclusive cluster as well as multiple published pathways with similar exclusivity (see section 4.3.6), bear out this choice of normalization. In four strains, measures were exclusively found either in the tested strain or its control. When a measure was always observed in one set but not the other, its normalization (and consequent z-score calculation) could not be computed. We reasoned that this was such an extreme case that it should be reflected in the representative z-score. Therefore, when a measurement was present in a strain but not its control, its z-score was imputed to be double the population maximum. Conversely, when a measurement was present in the control but excluded from the strain, the z-score was imputed to be double the population minimum (the minimum z-score was always negative). Finally, given the choice of normalization, the collective lab N2 group was represented using 0 for all of its measurements, prior to z-scoring the population of groups.

4.3.7 Computer Code

Unless otherwise noted, the image processing, statistics, and bioinformatics functions were available through common Java libraries (version 1.6), standard Matlab toolboxes (version 2010a), standard R packages (version 2.15), or created as needed. Several publicly available Mathworks packages significantly facilitated the analysis and figures within this chapter: the videoIO toolbox by Gerald Dalley, swtest function by Ahmed Ben Saïda, fexact function by Michael Boedigheimer, export_fig

function by Oliver Woodford, notBoxPlot function by Rob Campbell, and rdir function by Gus Brown.

4.4 Results

4.4.1 Behavioral phenotyping of 305 *C. elegans* strains

Eight WT2 units were used in tandem to collect the video data. In order to analyze behavioral phenotypes, automated algorithms were used to measure relevant parameters from video data. The choice of features was informed by common observer-based behavioral assays [23] in addition to automated methods developed for use in previously-described worm trackers [15,18,19,24,25]. Furthermore, several new algorithms for feature extraction were developed to quantify parameters related to motion state, crawling, foraging, dwelling, and exploratory range. At the top level, the extracted features include measures of morphology, posture, locomotion, and path dynamics (see section 4.3.2). These primary features were also evaluated in different contexts to give more complex parameterization; for example, mean speed was measured over the entire movie, as well as independently for periods when the animal was moving either forward or backward. Likewise, dorsal and ventral bending were measured over the entire body, and in specific regions such as the head, tail, and midbody. Finally, specific behavioral events such as reversals or omega turns were used to generate multiple parameters; for example, the frequency, time spent in execution, and the distance covered during the event. Altogether, the permutation of these measures yielded 702 distinct feature measurements (see Appendix B for the list of phenotypic measurements).

The data from these experiments have been compiled, and are available publicly at http://wormbehavior.mrc-lmb.cam.ac.uk. We provide the data in multiple resources alongside methods to check fidelity. Each video is available as a time series of feature computations. The WT2 feature viewer (a combined effort by Tadas Jucikas and myself) provides a visual assessment of these feature time series, frame-by-frame, accompanied by the worm's contour, skeleton, and its location on plate (Figure 4.2A-D). Each video is also provided, via YouTube, with an overlay of the skeleton, contour, head, ventral side, and annotations for unsegmented frames (Figure

4.2A). These two distinct views provide complementary methods to verify the fidelity of the features. The data set of 9,203 worm videos, presented herein, has a mean and standard deviation of 79.32 ± 18.49% segmented frames, 8.05 ± 8.94% dropped frames, 5.62 ± 10.24% stage movement frames, and 7.01 ± 14.22 % failed extraction frames. Further aggregate phenotypic measurements and statistics, per strain, are available directly on our website pages (see section 4.3.4), in a visual format in .PDF files, in HDF5-formatted .MAT files (an open-source format with widespread support including Matlab, Octave, R, Java, C/C++, and many other languages), and as spreadsheet compatible .CSV files. Inclusion of the skeleton in the .MAT files permits new features to be rapidly computed from the existing data set.

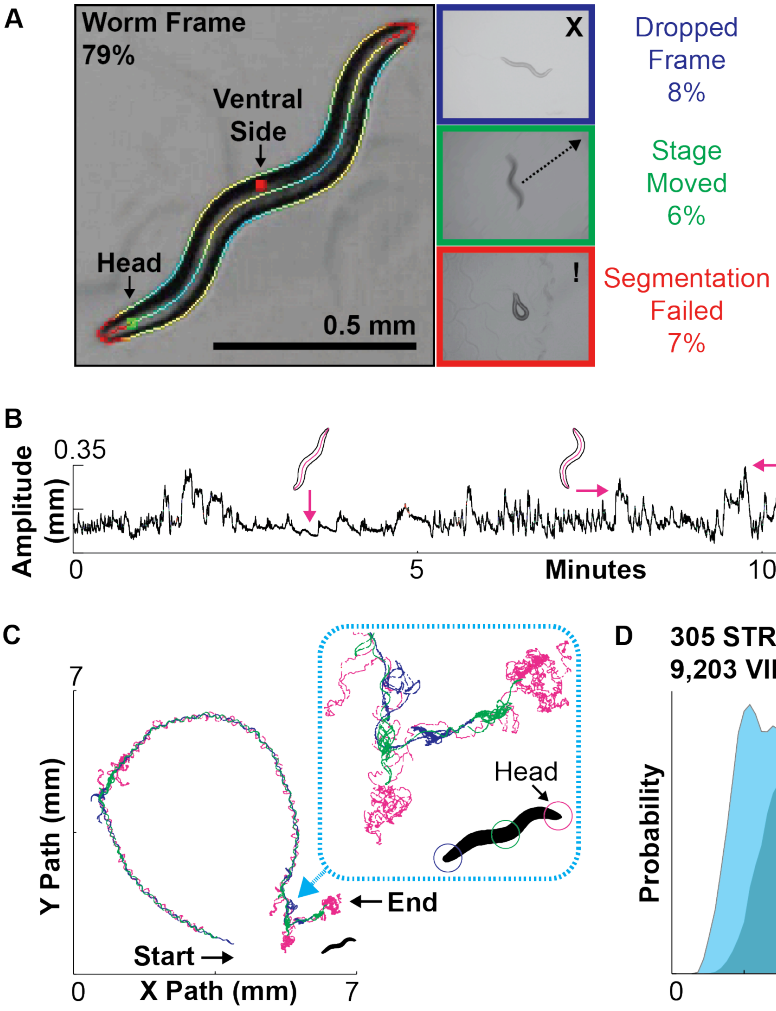


Figure 4.2: WT2 Online Data

Our full collection of data is available online at http://wormbehavior.mrc-lmb.cam.ac.uk. Individual experiments and grouped sets are accessible in multiple formats for statistical analysis (as well as for the purpose of validating segmentation and feature computations). A. Every analyzed video is available with an overlay indicating the head, ventral side, skeleton, and contour (see Chapter 3 section 3.5.1). Frames from which the worm could not be extracted are annotated with a color-coded border. The border indicates whether the frame was dropped (due to the processor's inability to keep up with incoming video frames), a stage movement occurred (blurring the worm image), or worm extraction failed (due to an unrecognized shape). Numbers indicate percentage in each category (mean ± standard deviation) for the present data set. Our data set of 9,203 worm videos, presented herein, has a mean and standard deviation of 79.32 ± 18.49% segmented frames, 8.05 ± 8.94% dropped frames, 5.62 ± 10.24% stage movement frames, and 7.01 ± 14.22 % failed extraction frames. B-D. The Worm Toolbox Feature Viewer provides path traces and frame-by-frame skeletons for every experiment, in addition to histograms and time-series plots for each feature of every experiment. B. An example of a single-worm behavioral time series measuring maximum amplitude. Several shapes are shown with their corresponding amplitude. C. An example of a single-worm path (the same worm as in panel B). D. An example of a feature histogram measuring maximum amplitude (for the strain presented in panels C and D).

4.4.2 Reproducibility and sensitivity of measurements

We performed a number of tests to assess the sensitivity and reproducibility of the measurements in our database. Variability in behavioral data would be expected from a number of sources. For example, the behavior of an individual animal is expected to fluctuate over time as a result of the stochastic nature of underlying commanding processes. In addition, animals of the same genotype might be expected to vary in their behavior due to slight differences in factors such as age or environmental conditions. Finally, different lab stocks of a particular strain might diverge genetically over time and thus give rise to phenotypic differences. We therefore assessed in turn each of these potential sources of variability.

4.4.2.1 Variability within an individual animal.

Many worm behaviors are stochastic and therefore vary in the course of even a short recording. Moreover, it is known that some behaviors change over time, in particular following transfer to a new environment. To investigate these issues, we tracked 25 young-adult wild-type hermaphrodites for 2 hours. Recording began immediately after transferring each worm to its tracking plate, without the usual habituation period we perform for our regular assays. Consistent with previous reports [26,27], the worm speed (titled "Absolute Midbody Speed" in our measurements) was well-fit by an exponential decay, with a time constant τ = 19 minutes (R^2 = 0.96; Figure 4.3B). Since speed might be expected to correlate with both crawling amplitude and frequency, we investigated whether these two measures might show similar habituation. In fact, crawling frequency (titled "Absolute Midbody Frequency" in our measurements) was best fit with an exponential decay of τ = 19 minutes (R^2 = 0.71; Fig. 2b); whereas amplitude (titled "Absolute Midbody Amplitude" in our measurements) decayed with a longer time constant of τ = 48 minutes (R^2 = 0.59; Figure 4.3C-D). Thus, the time necessary for locomotion features to adapt to steady state appears to differ, even among features affecting similar aspects of behavior.

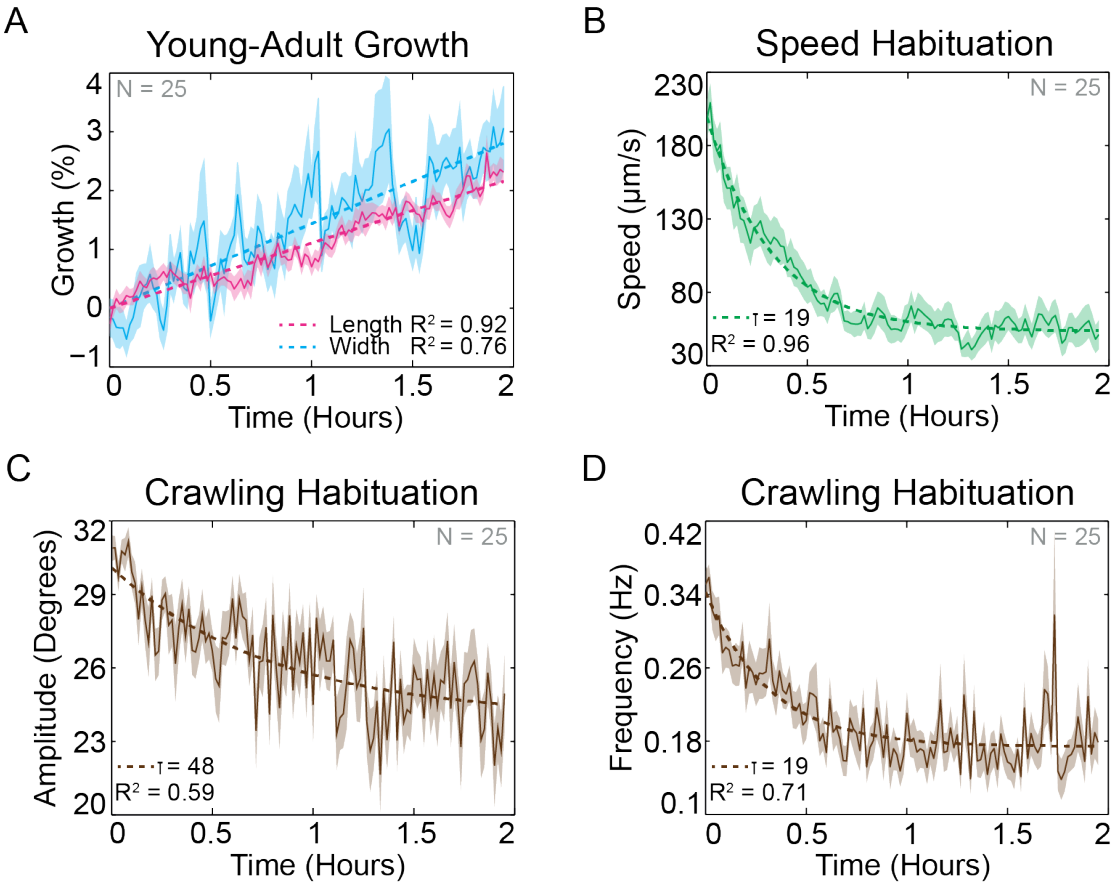


Figure 4.3: Growth and Habituation

To analyze growth and habituation, 25 lab-stock N2s were recorded immediately, upon being transferred to a new plate. In order to reduce noise, all features were averaged, per worm, into 1-minute bins. The mean across all worms, per binned minute, was then used to represent the analyzed feature values. A. Both length and width are fit with a 1% linear growth per hour (R^2 = 0.92 and 0.76 respectively). B. Worm speed habituation was fit to an exponential decay, with an estimated time constant τ = 19 minutes (R^2 = 0.96). C. The habituation of the crawling-wave amplitude is fit to an exponential with time constant τ = 48 minutes (R^2 = 0.59). D. The habituation of the crawling-wave frequency is fit with the exponential τ = 19 minutes (R^2 = 0.71) similar to that of the worm's speed.

4.4.2.2 Variability between animals of the same genetic stock

We next assessed the between-animal variability of worms from the same N2 wild-type stock. Individuals of a given N2 stock, after generations of self-fertilization, would be expected to be nearly genetically identical; however, small differences in age as well as difficult-to-control environmental factors might be expected to affect behavioral data. After filtering for quality (see section 4.3.4.1), we recorded a total of 1,218 N2 young-adult hermaphrodites over the course of three years, from 2009 to 2012. Animals were tracked from January to December, from Tuesday through Friday, and from 9am to 5pm (several time-based groups were excluded due to very low sampling -- see section 4.3.4.1). We chose six representative measures to analyze variability: length, forward speed (titled "Positive Midbody Speed" in our measurements), foraging amplitude (titled "Absolute Foraging Amplitude" in our measurements), reversals (titled "Backward Motion Frequency" in our measurements), coiling frequency, and exploratory range (Figure 4.4A). To assess the influence of these factors (hour, day, and month), one-way ANOVAs were performed with Bonferroni correction for 18 tests and checked for significance at $p \leq 0.05$. The tested groups show a mixture of normality and non-normality when using a Shapiro-Wilk test with correction for multiple comparisons. Therefore, to avoid the assumption of normality, Kruskal-Wallis tests were also performed with the same correction and α, resulting in identical significance. Since all tracked animals were identified as fourth-stage larvae the night before, we reasoned that animals tracked later in the day should be slightly older. Indeed, there was a small but significant difference in length (and other features) between animals tracked throughout the day. We measured changes in length and midbody width, over 2 hours, among the previously mentioned group of 25 young-adult, lab-stock N2s and found them well-fit with a 1% linear growth per hour (Figure 4.3A), $R^2 = 0.92$ and 0.76 respectively. This may well explain the observed hourly differences between animals. Likewise, we reasoned that although we control our lab to maintain 22°C, the temperature may still vary slightly with the season whereas the day of the week should have no consistent trend. Indeed, the day of the week had no significant effect on any of the tested parameters, whereas the month of the year had a slight but significant effect on all parameters but coiling (Figure 4.4A).

To account for seasonal effects, all data were controlled by wild-types collected within a three-week window centered around the experiments.

With the variability in mind, we computed the number of worms required to achieve statistical power in discriminating phenotypic differences. We bootstrapped Wilcoxon rank-sum tests, comparing multiple group sizes. The tested groups were chosen from our three-year collection of wild-type lab stock and compared to ones chosen from an identical collection, mean-shifted by the discriminatory amount. The findings indicate that 10 worms provide well over 90% power to discriminate two standard deviations of mean difference. Moreover, 20 worms discriminate a single standard deviation at over 80% power (Figure 4.4B). For example, when testing forward speed, 20 worms discriminate a mean difference of at least 53μm/s with 90% power (assuming the compared groups share an identically-shaped distribution to our lab-stock N2).

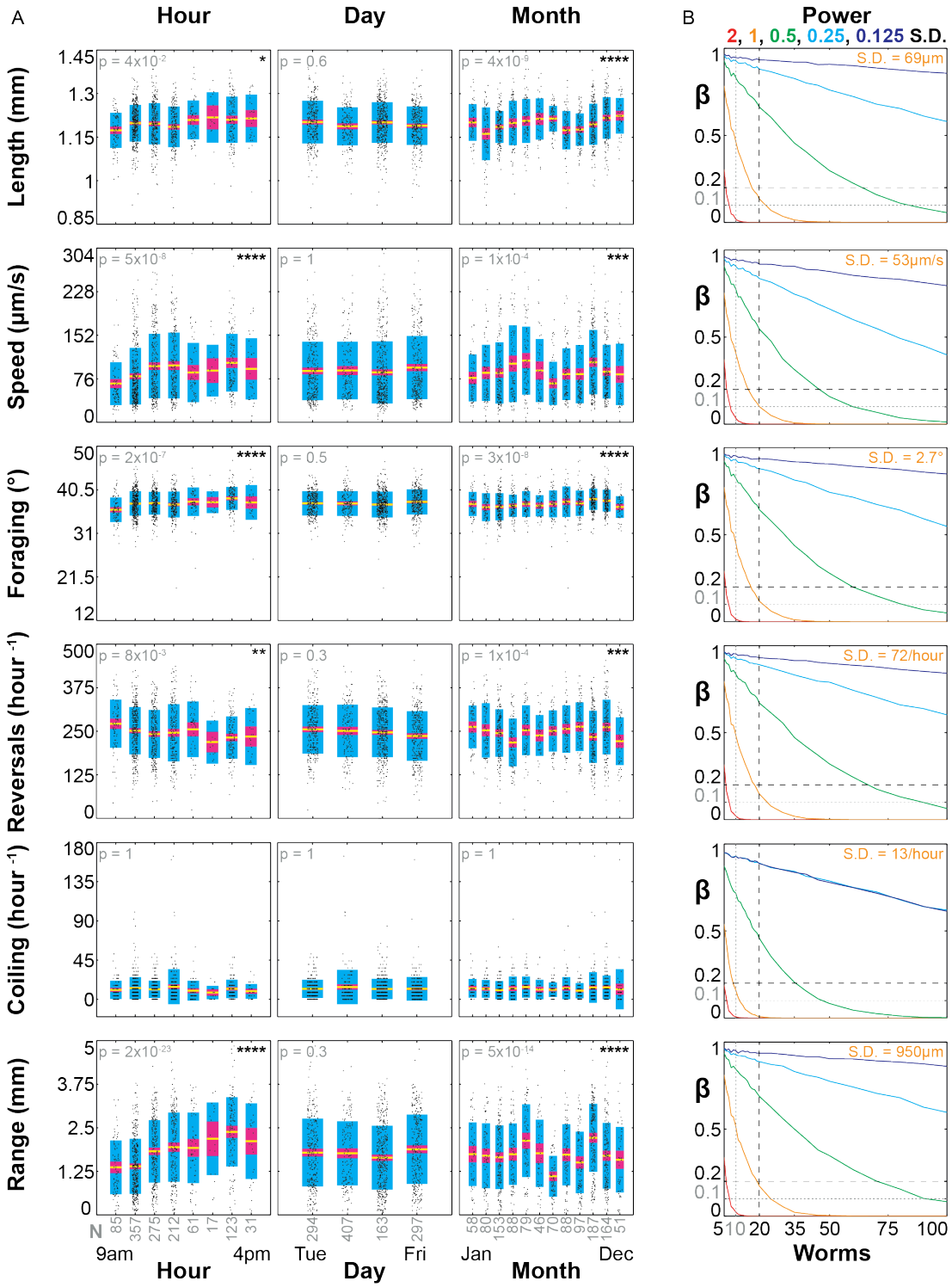


Figure 4.4: Wild-Type Variability and Statistical Power

Phenotypic variability and statistical power were analyzed using 1,218 young-adult N2 hermaphrodites, our lab-stock wild type, recorded over 3 years. A. Our wild type show significant differences in hourly and monthly measures at $p \leq 0.05$, but none daily (p and N values are shown on the plots). Kruskal-Wallis, one-way analysis of variance, tests were performed with Bonferroni correction for 18 tests of 3 groups (hour, day, and month) by six common measures: length, forward speed, foraging amplitude (ignoring the dorsal-ventral orientation), reversal frequency, coiling frequency, and exploratory range. Each experimental mean is plotted as a black dot and the mean of means is shown in yellow, SEM in magenta, and standard deviation in cyan. B. The number of worms necessary to discriminate a mean displacement of 2 (red), 1 (orange), 0.5 (green), 0.25 (cyan), and 0.125 (blue) standard deviations (SD) from the collective lab stock. Approximately, 10 worms discriminate a displacement of 2 SD with over 90% power (gray, dotted lines) and 20 worms discriminate 1 SD at over 80% power (black, dashed lines). Corresponding feature labels are presented on the far left, in panel A. Each plotted value was computed using 10,000 bootstrapped Wilcoxon rank-sum tests. The value for each SD is shown in orange on its corresponding plot.

4.4.2.3 Variation between different wild-type stocks

Self-fertilizing *C. elegans* experience mutation rates on the order of 10^{-8} per site, per generation [28]. To estimate genetically-derived variability between different lab stocks of the same strain, we compared the behavior of our lab N2 stock to one obtained from the Caenorhabditis Genetics Center. We chose four common measures (Figure 4.5A) to assess potential differences: length, speed (titled "Midbody Speed" in our measurements), foraging amplitude, and range. Speed and foraging were further subdivided into forward-backward and dorsal-ventral statistics, yielding a total of six statistical comparisons. Shapiro-Wilk tests, with $\alpha = 0.05$, indicated a mixture of normal and non-normal distributions for our 21 CGC and 27 lab N2 measures. Therefore, we compared both groups using Wilcoxon rank-sum tests with Bonferroni correction. Of all these comparisons, only dorsal-oriented foraging ($p = 6x10^{-3}$) showed a significant difference between the two wild-type stocks although visually the histograms for other features show some apparent differences and further testing, using subdivided feature measurements, uncovers related significant measures (available online in our database). Nonetheless, many features appear relatively consistent between the two wild-type stocks despite likely genetic divergence.

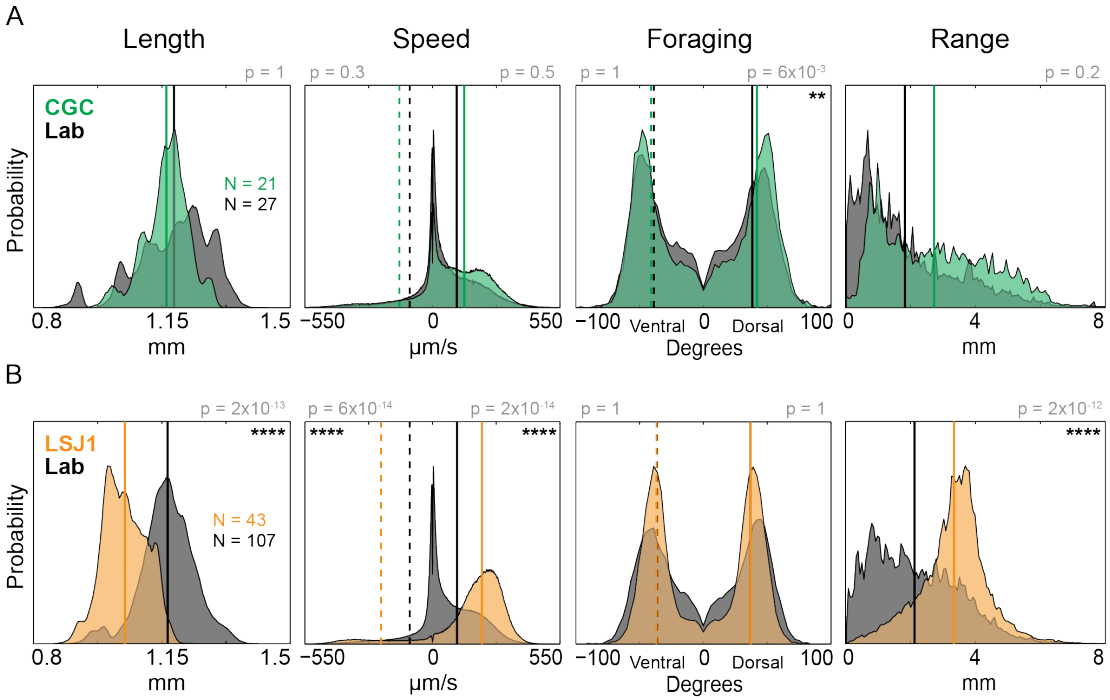


Figure 4.5: N2 Descendants

A comparison of N2 descendants. The means are shown as vertical lines for positive (solid line) and negative (dashed line) portions of the histogram. Wilcoxon rank-sum tests were performed with Bonferroni correction for the 12 tests across both panels (p and N values are shown on each histogram). A. The CGC-stock N2 (green) is compared to our lab-stock N2 (gray). B. LSJ1 (orange) is compared to our lab-stock N2 (gray).

We also analyzed LSJ1, a more distant relative of N2. This strain is descended from the same wild isolate as N2, but the two strains have diverged following years of laboratory cultivation [29]. In particular, mutations in at least two genes, *npr-1* and *glb-5*, have arisen in the N2 strain that promote solitary feeding under lab conditions and lead to lower speeds and decreased preference for the border of the food lawn [30-32]. We compared 43 LSJ1 with 107 of our lab N2 for the same six features described above (Figure 4.5B). As expected, the LSJ1 showed significantly higher speed (backward $p = 2x10^{-14}$; forward $p = 6x10^{-14}$) and a significantly larger range of exploration ($p = 2x10^{-12}$), correlating with the bordering phenotype of LSJ1. Additionally, LSJ1 was slightly (0.1 mm) but significantly shorter ($p = 2x10^{-13}$) than N2. In comparison, the summary statistics for foraging are nearly overlapping ($p = 1$ for both sides) with a dorsal mean and SEM of 37±0.5° and 36.9±0.4°, for the LSJ1 and lab N2 respectively, and ventral values of 36.5±0.3° for both strains.

4.4.3 Locomotion phenotypes

Having assessed the sensitivity and variability of our system in wild-type strains, we turned our attention to identifying new phenotypes in mutants. Our database contains 305 strains carrying mutations in genes known or likely to affect nervous system function, 76 of which had no previously characterized phenotype [33]. We compared each of these strains to controls of lab-stock wild type, with respect to 702 measures. In four strains, at least one measure was detected exclusively in either the strain or its control, meaning the measurement was always observed within one set and never in the other (for example, some strains never perform reversals). In such cases, we used a Fisher's exact test to measure the probability that these two sets were drawn from the same distribution of observed and unobserved events. For the remaining strains, we performed statistical tests of significance using the Wilcoxon rank-sum test, controlling the false-discovery rate with a q-value in place of p [21]. All strains were found significantly different than the wild type at $q \leq 0.05$, including those representing 76 genes with no previously characterized phenotype. These results

are summarized in Appendix C. A detailed set of 330 descriptions (including a wild-type reference, using our 1,218, lab-stock N2) is available online at our database.

At present, most of the mutant genes in our database are represented by single alleles only, so the identified new phenotypes must be regarded as provisional. Nonetheless, some genes are represented by multiple alleles, including the previously uncharacterized *trpa-2*. *trpa-2* is a member of the TRPA family of cation channels, which function in nociception and thermosensation in many animals including *C. elegans* [34]. However, TRPA-2 is an unusual member of the TRPA family as it contains no predicted ankyrin repeats. TRPA-2 is expressed in multiple neurons including the ASEs which sense taste [35]. We tracked three *trpa-2* deletion mutants (*ok3189*, *tm3085*, and *tm3092*). From our analysis of tracking videos, we determined that all three mutant strains showed a similar significantly altered posture during reversals (Figure 4.6A), corresponding to a greater number of body bends and a decrease in a measure related to sinusoidal locomotion. These results suggest a potential proprioceptive role for *trpa-2*, a hypothesis that can be investigated further in the future.

In addition, we also detected new locomotion phenotypes for several genes previously implicated in other processes. For example, OCR-4 is a TRPV channel expressed in the four OLQ nose touch neurons. We observed that two *ocr-4* mutants, *tm2173* and *vs137*, had decreased amplitude when crawling and more tail motion when paused (Figure 4.6C). Likewise, *trp-2* encodes a TRPC channel expressed widely in the nervous system that has been previously implicated in responses to nicotine. We observed that two *trp-2* mutants, *gk298* and *sy691*, showed similar alterations in waveform and foraging, specifically when reversing (Figure 4.6B). Moreover, both alleles spend less time in omega turns, which are already infrequently observed in our 15-minute wild-type videos (on average 2 per video). Together, these results indicate that automated tracking data can robustly detect previously unnoticed locomotion phenotypes.

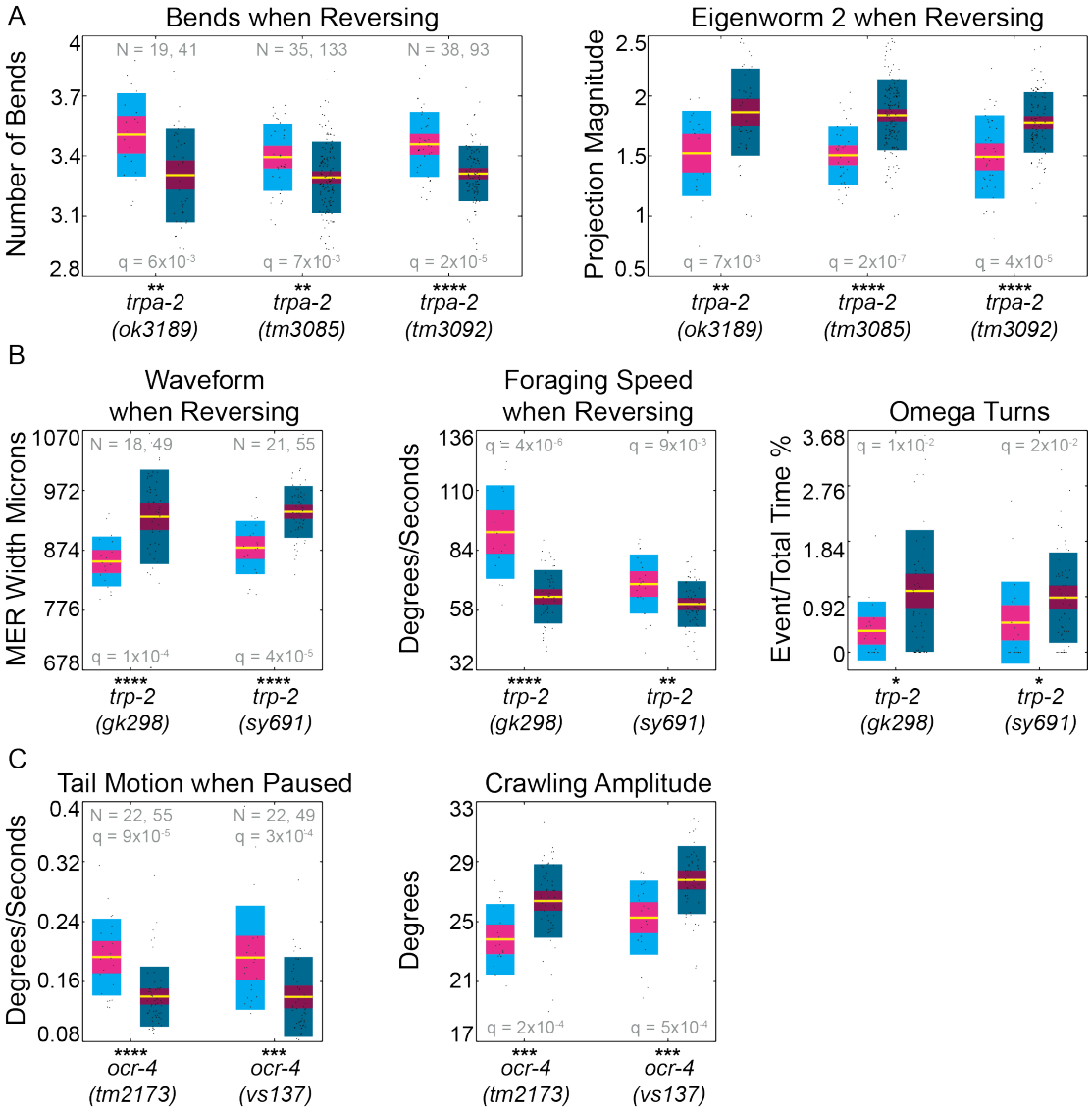


Figure 4.6: New Locomotion Phenotypes

New locomotion phenotypes reported for three TRP channels. Both *trpa-2* and *ocr-4* have no characterized phenotypes despite previous experimental publications. Each experimental mean is plotted as a black dot and the mean of means is shown in yellow, SEM in magenta, and standard deviation in cyan. For each allele, the experiments are brighter and shown on left, whereas the controls are darker and shown on right. N and false-discovery rate q-values are displayed on the plots. A. All three *trpa-2* alleles show significantly more bends when reversing and a decrease in their second eigenworm projection (a feature measurement related to sinusoidal locomotion) when reversing. B. Both *trp-2* alleles display significantly altered activity during reversals as well. Their waveform, when reversing, is more compact. Their foraging speed, when reversing, is faster. The two *trp-2* strains spend even less time in omega turns than their wild-type controls. C. The *ocr-4* alleles also display significant differences when subdividing by motion state. Both alleles move their tail more when paused. Furthermore, both *ocr-4* alleles display smaller amplitudes when crawling.

4.4.4 Phenotype clustering

Clustering analysis can identify phenotypic similarities and thereby predict possible gene functions. We therefore performed hierarchical clustering using the mean values of each of the 702 measures described above. Each group's measures were normalized to their wild-type controls and converted to z-scores based on the population of 330 groups as a whole. Some measurements were sparsely sampled, missing entirely from either the experiment or its control, yet not exclusive to either. These missing measurements were imputed to be the population mean, 0. We expected a high correlation between many of our measurements. This might lead to some measures being weighted more than others when clustering. Therefore, we used principal components analysis (PCA) to transform our feature space into an orthogonal space of independent eigenfeatures. We kept the first 127 eigenfeatures, which accounted for 99% of the total variance, for our clustering (Figure 4.7A). We reasoned that similar groups, with subtle differences in genetic background unrelated to the mutation of interest, could end up separated due to small deviations in their eigenfeature values. For this reason, we focused on sizable group similarities, choosing uncentered correlation to compute their distance and single linkage for their hierarchical clustering [36]. We ran 10,000 bootstraps with multiscale resampling and generated approximately unbiased (AU) p-values, measuring the confidence of tree selection for each branch of the clustering [37] (see Appendix D for the full clustering figure).

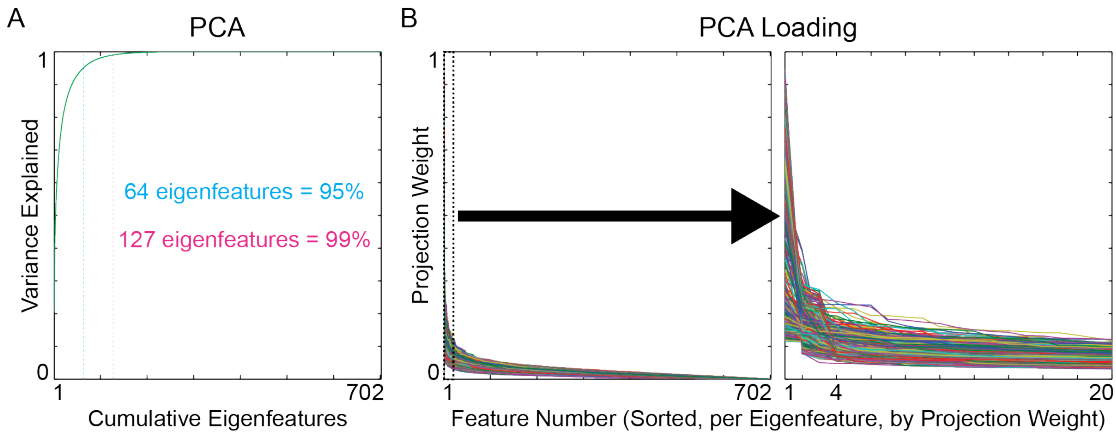


Figure 4.7: Eigenfeatures

To achieve an independent weighting of features when clustering strains and groups, our correlated feature measurements were transformed into orthogonal eigenfeatures by computing a Principal Component Analysis (PCA) over 330 groups of 702 feature measurements (normalized and z-scored to the set of groups). A. 127 eigenfeatures capture 99% of the variance. B. The PCA loading, for each one of the 702 eigenfeatures, was sorted by weight (highest to lowest coefficient magnitude). After roughly the top 4 weights, the remaining coefficients appear to contribute little to each eigenfeature projection. Therefore, most of the eigenfeature projections are primarily the result of a few features. Unfortunately, despite this simplicity, the combination of features that contribute are not obviously related and so do not provide an intuitive phenotypic perspective.

As a control for the clustering results, we first examined the outcome of the wild-type recordings. To understand the range of clustering values, the maximum pairwise correlation in the entire clustering was 0.95, between the group of lab-stock N2s and those tracked on Tuesday. In contrast, the minimum was 0.34, found when comparing *unc-1*(e1598) to all other groups. Similarly, while the top-most branch has an AU of 1 (as is expected for hierarchical clustering), the next greatest AU was 99.98% for the branch connecting the two alleles of *unc-108*. In contrast, 165 of the 329 branches had an AU of 0%. The majority of our lab N2 groups clustered together exclusively (meaning no other groups were included in their cluster), with a correlation of 0.75 and an AU of 98.56% (Figure 4.8A). Wild isolates (Figure 4.8B) from Hawaii and Freiburg, strains CB4856 and RC301, clustered together exclusively at 0.88 (AU = 97.97%). Both these strains carry an allele of the *npr-1* neuropeptide receptor gene that confers social feeding behavior and has lower activity than the allele found in N2 [30]. Interestingly, these strains clustered with an N2 mutant strain carrying an *npr-1(ad609)* mutation at 0.81 (AU = 97.10%), as well as LSJ1, the relative of N2 carrying the *npr-1(215V)* allele [31], at 0.8 (AU = 96.22%). N2 mutants for several other neuropeptide-related genes were present in this putative *npr-1*-related cluster; these may represent candidates for other molecules involved in NPR-1 signaling.

Many other known pathways clustered with high correlation. For example, a loss-of-function mutation in the G_q RGS gene *eat-16* clustered exclusively with an activating mutation in the G_q alpha-subunit *egl-30* at 0.89 correlation (AU = 99.71%) and were joined exclusively by a loss-of-function mutation in the Go alpha-subunit *goa-1* [38] at 0.78 and 90.8% AU (Figure 4.8C). EAT-16 acts antagonistically to GOA-1, suppressing the loss-of-function mutant. Experimental evidence indicates that EAT-16 acts downstream or in parallel with GOA-1, negatively regulating EGL-30 signaling. *unc-38* and *unc-63*, which encode α-subunits of the same acetylcholine receptor (AChR) [39], clustered exclusively together at 0.76 (AU = 97.3%). *unc-79* and *unc-80*, subunits of the NALCN sodium channel [40], clustered exclusively at 0.84 (AU = 99.33%). Both alleles of *unc-4* clustered together exclusively with *unc-37*, which is required for their activity in specifying the fate of A-class motor neurons [41], at a correlation of 0.68 (AU = 98.19%). Moreover, both alleles of *unc-108* clustered together exclusively at 0.79 (AU = 99.98%) as did both alleles of *egl-21* at 0.62 (AU

= 99.94%). *bas-1*, required for serotonin synthesis [42], clustered exclusively with the three G protein-coupled serotonin receptors *ser-1*, *ser-4* and *ser-7* [43-45] as well as the octopamine receptor *ser-2* [46] at 0.74 (AU = %98.36). Finally, *unc-7* and *unc-9*, innexin genes whose products are thought to multimerize in gap junctions [47], clustered together exclusively at 0.66 (AU = 94.36%).

Of genes with unknown functions, a mixed set of seven predicted and verified AChRs clustered together (Figure 4.8D) at 0.82 correlation (AU = 99.53%): *acr-2*, *acr-3*, *acr-6*, *acr-10*, *acr-11*, *acr-18*, and *acr-21* [48]. These strains showed many shared significant differences from the wild-type control, including increased locomotion, foraging, exploratory range, and less path curvature (Table 4.1). *acr-2* and *acr-3* are known to form functional channels when separately co-expressed with *unc-38* in Xenopus oocytes and have been implicated in the control of locomotion [49-51]; however, the functions of the other genes in this cluster are unknown. Since AChRs are pentameric, and the clustered collection contains a mix of five α and two non-α subunits [52], these may combine combinatorially to generate several related receptors involved in the control of foraging and body movement.

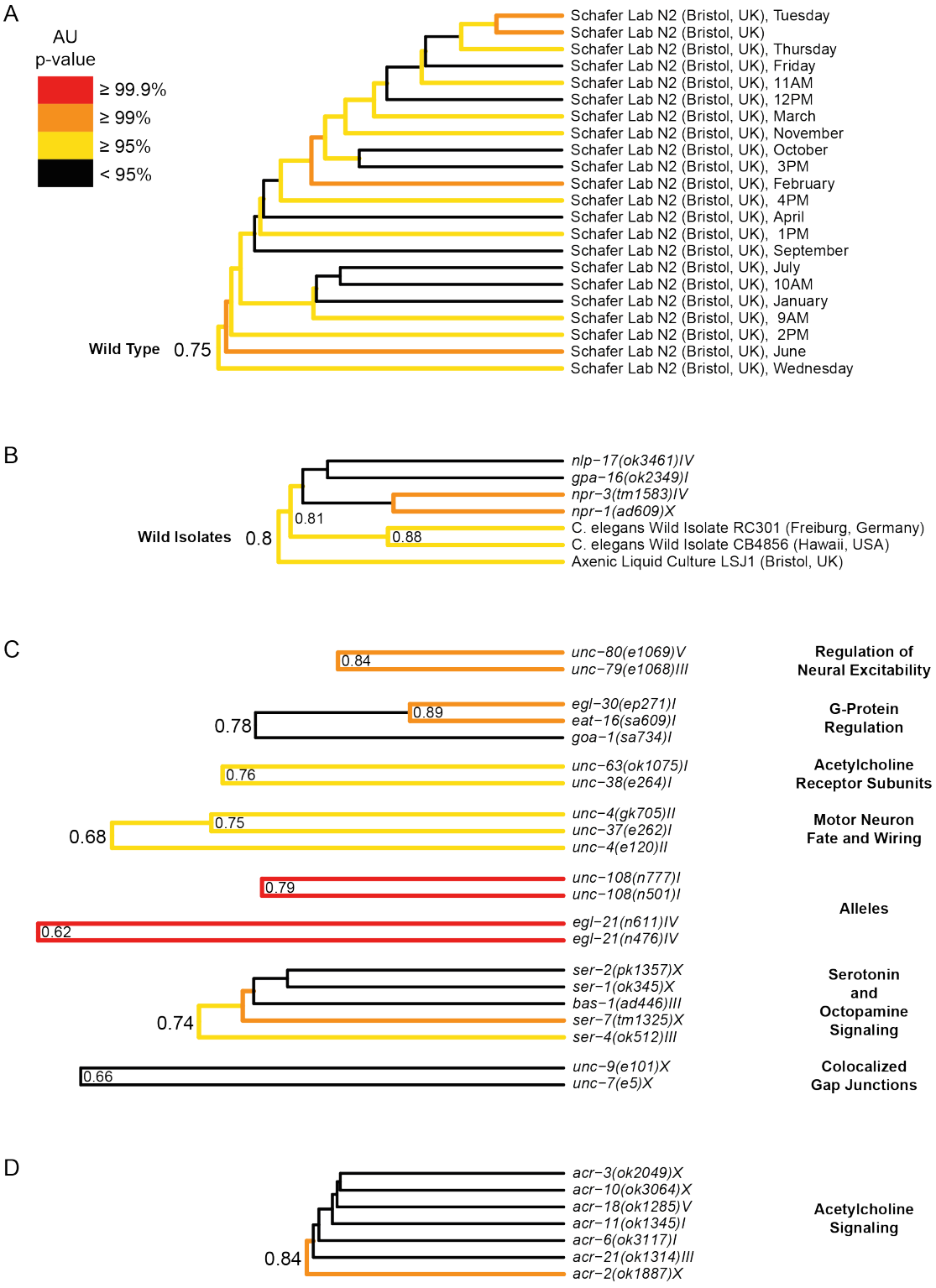


Figure 4.8: Phenotypic Clustering

A subset of the phenotypic clustering results. The full clustering figure, for all analyzed strains and groups, is available in Appendix D. Correlation values are shown in black at their respective branches. The branches themselves are color coded to show the approximately unbiased probability (AU) of the clusters below them. AU ≥ 99.9% is colored red, AU ≥ 99% orange, AU ≥ 95% yellow, and all else is black. A. Most of our lab-stock N2 (as a full group of 1,218 and separated into hours, days, and months) clustered together exclusively. B. The wild isolates CB4856 (Hawaii, USA) and RC301 (Freiburg, Germany), which share identical mutations in their *npr-1* and *glb-5* alleles, clustered exclusively with very high correlation and AU. Their cluster was closely joined by LSJ1 (which shares their variant *npr-1* and *glb-5* alleles) and *npr-1*(ad609), an NPR-1 mutant in an N2 background. C. Multiple published pathways are shown to cluster together exclusively with both high correlation and high AU values (see text for further details). D. A set of seven AChR mutant strains, five α and two non- α subunits, cluster together exclusively with high correlation and AU values relative to the other controls displayed above.

Genotype	Strain	Phenotypic Ontology
acr-2(ok1887)X	RB1559	+Length (*), +Width (*), +Head Bends (****), +Tail Bends (*), +Posture Amplitude (*), +Posture Wavelength (*), +Coils (*), +Foraging (**), +Forward Velocity (**), +Backward Velocity (**), +Forward Motion (**), -Pausing (**), ΔBackward Motion (*), +Crawling Amplitude (***), ΔCrawling Frequency (*), ΔTurns (*), +Path Range (***), -Path Curvature (**)
acr-3(ok2049)X	RB1659	+Head Bends (****), +Tail Bends (**), ΔPosture Amplitude (**), -Posture Wavelength (*), +Coils (*), +Foraging (**), +Forward Velocity (***), +Backward Velocity (**), +Forward Motion (***), -Pausing (***), ΔBackward Motion (**), ΔCrawling Amplitude (****), ΔCrawling Frequency (**), ΔTurns (**), +Path Range (***), -Path Curvature (***)
acr-6(ok3117)I	RB2294	+Width (*), +Head Bends (****), +Tail Bends (*), ΔPosture Amplitude (*), +Coils (*), +Foraging (**), +Forward Velocity (***), +Backward Velocity (***), +Forward Motion (***), -Pausing (****), ΔBackward Motion (**), ΔCrawling Amplitude (***), ΔCrawling Frequency (***), ΔTurns (*), +Path Range (****), ΔPath Curvature (****)
acr-10(ok3064)X	RB2262	+Width (*), ΔHead Bends (***), +Tail Bends (*), ΔPosture Amplitude (*), +Foraging (***), +Forward Velocity (****), +Backward Velocity (****), +Forward Motion (***), -Pausing (****), ΔBackward Motion (**), +Crawling Amplitude (***), ΔCrawling Frequency (**), ΔTurns (***), +Path Range (****), -Path Curvature (****)
acr-11(ok1345)I	RB1263	+Head Bends (***), ΔTail Bends (*), -Posture Wavelength (*), +Foraging (***), +Forward Velocity (****), +Backward Velocity (****), +Forward Motion (***), -Pausing (****), ΔBackward Motion (*), ΔCrawling Amplitude (***), ΔCrawling Frequency (**), ΔTurns (**), +Path Range (****), -Path Curvature (****)
acr-18(ok1285)V	RB1226	-Length (*), -Width (**), ΔHead Bends (****), +Tail Bends (**), -Posture Amplitude (*), -Posture Wavelength (*), -Coils (*), ΔForaging (***), +Forward Velocity (****), +Backward Velocity (****), +Forward Motion (****), -Pausing (****), ΔBackward Motion (***), ΔCrawling Amplitude (***), ΔCrawling Frequency (****), ΔTurns (****), +Path Range (****), -Path Curvature (****)
acr-21(ok1314)III	RB1250	+Width (*), +Head Bends (***), ΔPosture Amplitude (*), -Posture Wavelength (*), -Coils (*), +Foraging (****), +Forward Velocity (***), +Backward Velocity (***), +Forward Motion (****), -Pausing (****), ΔBackward Motion (**), ΔCrawling Amplitude (**), ΔCrawling Frequency (**), ΔTurns (***), +Path Range (***), -Path Curvature (****)

Table 4.1 : AChR Phenotypic Ontology

Seven AChR strains, that clustered together, show many shared significant differences from the wild-type control, including increased locomotion, foraging, exploratory range, and less path curvature. Phenotypic ontology is described in section 4.3.3. Briefly, "–" means the associated feature(s) measured less than the control, "+" means they measured greater, and "Δ" means they measured different (but the difference cannot be quantified as simply lesser or greater than). q-values were used to correct for multiple testing. The following convention is used: * when $q \leq 0.05$; ** when $q \leq 0.01$; * when $q \leq 0.001$; and, **** when $q \leq 0.0001$.**

4.5 Discussion

4.5.1 Phenotype Database

In summary, we present here the collection and initial analysis of behavioral data for 305 *C. elegans* strains. These data have been made available to the research community after various degrees of processing. The raw videos, with an overlay indicating the skeleton and contour, and the orientations of the head and ventral side, can be accessed publicly on YouTube. Segmented movies and their accompanying time-series data, easily reviewed within the feature viewer, can be obtained from the website http://wormbehavior.mrc-lmb.cam.ac.uk/. Also available for each genotype is a file of summary statistics, including mean, variance, sample size, and histograms of feature distributions compared to wild-type controls, for all measured parameters. Finally, we provide summary files of all statistically-significant differences between mutant genotypes and wild-type as well as hierarchical clustering information indicating phenotypic similarities between tracked strains.

These resources should prove useful to the research community in a number of ways. In particular, they provide an entry point toward understanding the functions of a significant number of genes that have been mutated by the *C. elegans* knockout consortia but not yet phenotypically characterized. For these genes, the clustering analysis provides information about phenotypically similar strains that may give insight into the molecular pathways in which the mutant gene participates. Likewise, the detailed feature measurements provide information about the behavioral processes affected by the gene of interest, which may suggest a neural basis for observed phenotypes. Finally, the availability of raw video data will make it possible for investigators to develop their own approaches to studying the behavior of particular mutant strains. In principle, these data make it possible for researchers to analyze *C. elegans* behavioral patterns, generating and testing neurobiological hypotheses without conducting any additional wet experimentation. Thus, the *C. elegans* "phenome", like its genome and connectome, could become an object of bioinformatic study for purely computational biologists.

4.5.2 Future Directions

In the future, we intend to expand our database to increase its utility to *C. elegans* researchers. An important aspect of this goal is to record strains carrying mutations in additional genes not currently represented in our data set. A number of gene families with important roles in the nervous system remain to be tracked (or are only partially tracked) including glutamate receptors, potassium channels, and G-protein coupled receptors. In addition, we plan to record mutants carrying additional alleles of genes already represented in the database. Since all mutant strains carry multiple sequence changes in addition to the identified mutation, observing behavioral phenotypes in more than one independently-derived mutant strain increases the confidence that these changes are indeed related to the gene of interest. Finally, we would like to extend the scope of our database to include strains expressing optogenetic activators or inhibitors of neural activity in defined neurons. In this way, it should be possible to use high-content phenotyping to address the link between neurons and behavior as well as genes and behavior.

Beyond adding more strains to the phenotype database, we also hope to increase the richness of the data available for each genotype. One approach would be to record each strain under a variety of different environmental conditions, probing different sensory modalities and behavioral responses. A recently published study of behavioral responses to thermal stimuli highlights the potential effectiveness of this approach [53]. In addition, it should be possible to extract additional features from the existing video data. This could involve developing new algorithms for feature measurement, or more unsupervised approaches based on time series motifs [15]. It is our hope that making these data available to the community will encourage other researchers to contribute collaboratively to the emerging science of behavioral informatics.

Of interest, several mutations that were expected to cluster together failed to do so. In particular, UNC-29, LEV-1, and LEV-8 have been shown to form functional subunits of the same levamisole-sensitive AChR with UNC-38 and UNC-63 [39]. While both *unc-38* and *unc-63* correlated well phenotypically and joined together exclusively in our clustering (Figure 4.8), *lev-1*, *lev-8*, and *unc-29* did not.

Additionally, the G-protein, beta-subunit GPB-2 has been shown to form a complex with the RGS proteins EAT-16 and EGL-10; more specifically, the *gpb-2* knockout reduces the functions of EAT-16 and EGL-10 to that of their null mutants [9]. EGL-10, in turn, negatively regulates GOA-1 [54]. While *goa-1*, *eat-16*, and *egl-30* all correlated well phenotypically and joined together in an exclusive cluster (Figure 4.8), *gpb-2* and *egl-10* did not.

One possibility as to why the aforementioned pathways failed to cluster may be phenotypic effects resulting from monthly variations (e.g., seasonal temperatures). We attempted to correct these by normalizing strain measurements to their wild-type controls, assayed within a week of each experiment (see section 4.4.2.2). Perhaps this simplistic approach was insufficient to correct for the extremes of monthly variation. As Figure 4.8 and Appendix D show, the only outliers among our lab-stock N2 time-based groups were May, August, and December, which failed to cluster exclusively with the remaining lab-stock groups. This may imply that these months represent temperature extremes with corresponding non-linear changes in phenotype. Examining the aforementioned unsuccessful clusters does not support this explanation. Our *lev-1* assays were run in September 2011 (N=16) alongside *unc-38* (also run later, in August; both months were run in 2010 where N=7,11 and 2011 where N=7,22 -- respectively in order of month), and did not cluster together, whereas *unc-63* (which correlated strongly with *unc-38*) was profiled in April 2010 (N=23). Our *gpb-2* assays were run in March 2010 (N=21) alongside *egl-30* (also run in February of the same year where N=6,13 -- respectively in order of month), and did not cluster together, whereas *eat-16* (which correlated strongly with *egl-30*) was profiled in July (N=10) and August (N=6) of 2010. These examples indicate that factors other than monthly variations likely play a strong role in the variability of phenotypic output, when considering interacting genetic pathways. Several other possibilities that might explain variability in phenotypic output include differing genetic backgrounds among the strains, undocumented background mutations accompanying the mutagenized gene of interest, differential expression patterns for gene products, and, as might be the case for the AChR subunits, the strength and relatedness of phenotypes associated with various combinatorial assemblies.

One might ask whether the dissimilarity in observed phenotypes might have an allelic basis as well. In other words, we might expect different alleles of the same gene to display significant variability in phenotype, as would be evidenced in the case of loss versus gain of function mutations. For example, EGL-30 and GOA-1 exhibit antagonistic effects. In our clustering *egl-30(ep271)*, a gain-of-function allele, correlated well with the loss-of-function allele *goa-1(sa734)*, whereas the double mutant *egl-30(n686);goa-1(n1134)*, containing only loss-of-function alleles, failed to join their cluster (Figure 4.8 and Appendix D). With this hypothesis, that allelic variations are partially responsible for dissimilarities in observed phenotypes, knockout mutants would be more likely to show correlated phenotypes. This hypothesis is not uniformly supported by our results. While several alleles clustered together exclusively as expected (Figure 4.8), surprisingly none of the *trpa-2* alleles did so, all three of which are knockouts and two of which originate from the Japanese National BioResource Project. For reference, *trpa-2(tm3092)* was assayed in March, June, and July of 2010 (N=16,8,14 -- respectively in order of month) and *trpa-2(tm3085)* was profiled at the same time in addition to April 2010 (N=7,8,7,13 -- respectively in order of month). The results discussed in this and the above two paragraphs, imply that we might benefit from exploring other methods for predicting genetic interactions. With regards to clustering techniques, we might benefit from testing various distance measures, aggregation algorithms, and transformations that achieve orthogonal eigenfeatures to see whether we can produce a more faithful clustering. Beyond these informatics techniques, and of great interest in the field of phenotypic quantification, would be a thorough investigation into the main factors contributing to variable phenotypic output, for example, whether there exist classes of mutations that significantly reduce or increase phenotypic variability.

Among the more interesting predictions in our clustering, is a branch containing two wild isolates, CB4856 and RC301, with LSJ1 (Figure 4.8). All three strains share identical variants of the GLB-5 and NPR-1 proteins [30-32]. Within this highly-correlated cluster is a targeted *npr-1* mutation in an N2 background and three mutant strains, *gpa-16*, *nlp-17*, and *npr-3*, whose functional roles in the nervous system remain relatively uncharacterized. *gpa-16* encodes a G-protein alpha-subunit involved in embryonic development with widespread neural and cellular expression [55,56]. *nlp-17* encodes a neuropeptide-like protein discovered through a screen for

corresponding genomic sequences [57]. *npr-3* encodes a neuropeptide receptor which responds to the ligand encoded by *flp-15* [58]. All strains within the cluster have at least one pairwise correlation of 0.8. For comparison, the exclusive cluster of our lab-stock N2s has an equivalent pairwise correlation of 0.75. Examining the path traces for these seven strains (Figure 4.9) shows that they all display similar phenotypes with regards to high speeds, a large exploratory range, and potential food-leaving behavior. The high speeds and large exploratory range receive statistical support in the phenotypic ontology (Table 4.2); as discussed in Chapter 3 (section 3.5.2.3), at present we cannot assay food-leaving behavior. The NPR-1 variants have already been shown to have high speeds [30,31]. Additional phenotypes associated with NPR-1 include bordering and social aggregation, neither of which are quantified by WT2. Furthermore, the GLB-5 variation has been shown to confer higher sensitivity to oxygen concentration than that of N2 [31]. If *gpa-16*, *nlp-17*, and *npr-3* are involved in pathways with *npr-1* and/or *glb-5* we might expect to see similar phenotypes for aggregation, bordering, and/or oxygen preferences. As such, it would be interesting to run these assays in order to test our clustering predictions and potentially discover new functional roles for these clustered genes.

In summary, quantification of the *C. elegans* phenome requires an intensive and extensive view of worm phenotypes. I created WT2 to remove restrictions associated with previous single-worm trackers and enhance the throughput and thoroughness of phenotypic quantification (Chapters 2 and 3). Our public database encompasses over 300 strains, an order of magnitude larger than the previous largest set [59]. Moreover, our collection presents the most extensive set of measures to date, 702 (see Chapter 2, Table 2.1), and is the sole location wherein researchers can access the underlying defining set of intensive frame-by-frame measurements for worm experiments. Each profiled strain had significant differences from its wild-type control (see section 4.3.5) implying that WT2 has the precision necessary to close the phenotyping gap and attribute significance to genetic perturbations. Conversely, my analysis indicates that this significance is not merely a byproduct of noise since our measures have the accuracy to phenotypically correlate and tightly cluster known genetic correlates and interactions (see section 4.4.4). The methods in this thesis present a framework for analyzing and presenting phenomic "big data", a common problem plaguing large data sets emerging through advances in scientific throughput.

Beyond this, the clustering of 305 strains (Appendix D) reveals multiple predictions of genetic interactions that await testing. Our large wild-type collective provides an N2 reference and with it, a meaningful quantification for what it means to be a wild-type *C. elegans* (see section 4.4.2). This reference answers a variety of questions regarding phenotypic variability among animals with nearly identical genetics, the approximate number of worms necessary for sufficient statistics and phenotypic discrimination, and factors affecting experimental control. Finally, the database itself should prove a robust computational resource in modeling biophysics, answering detailed questions of phenotypic measurement, and overall as an in-silico "sandbox" to investigate worm phenotypes and bioinformatics. Similarly, the strain statistics and measures provide a wealth of answers for biologists exploring the connection between genetics and phenotype, including phenotypic significance for 76 genes with no previously reported differences from wild type (Appendix A).

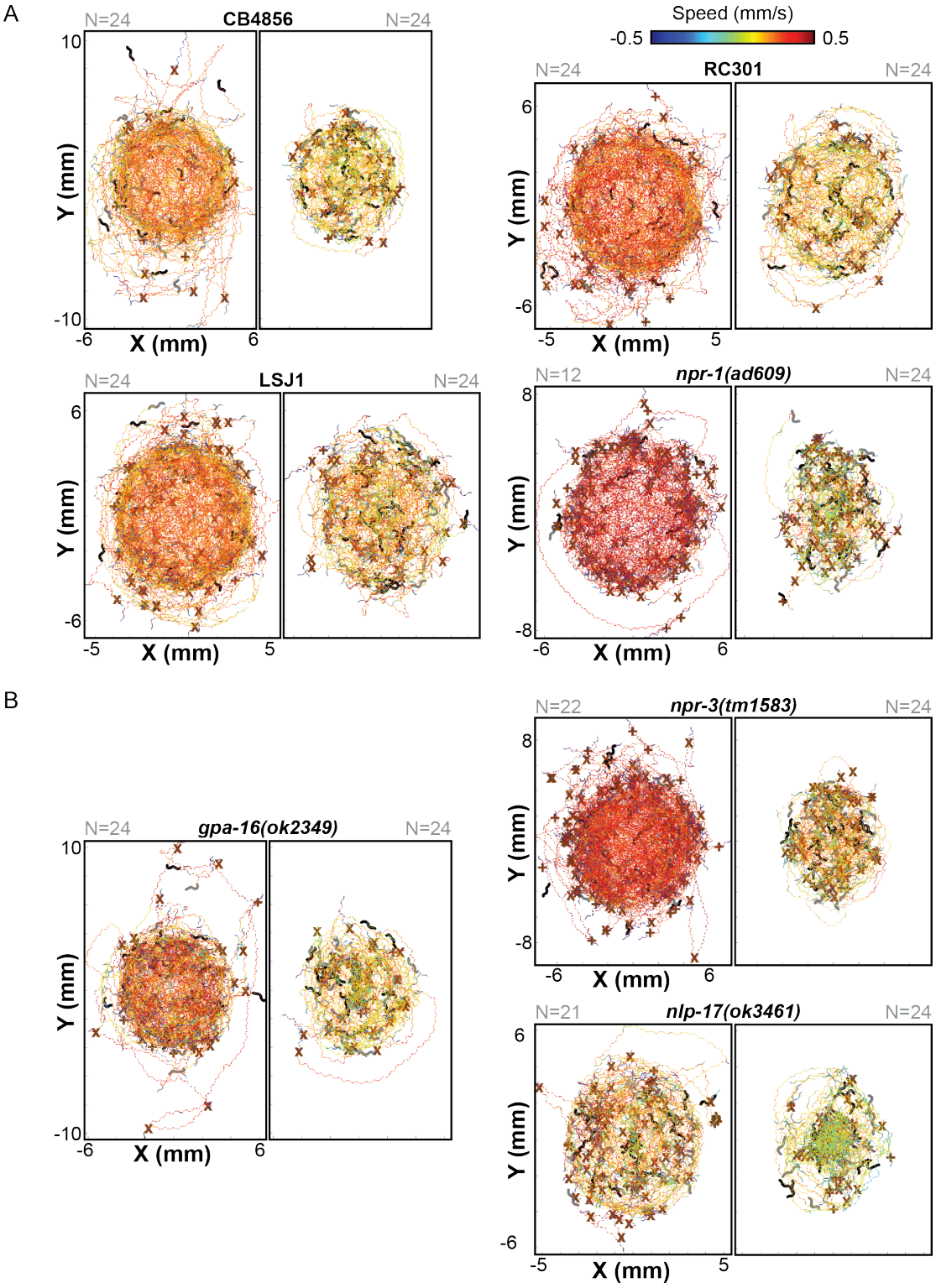


Figure 4.9: Worm Paths for GLB-5, NPR-1, and Clustering Predictions

Seven strains, that clustered together, display similar path traces showing high speeds, a large exploratory range, and potential food-leaving behavior (most evident in CB4856 and *gpa-16*). The traces originate from our online PDF files (see section 4.3.4.2). Each pair of traces shows the experimental strain on the left and its wild-type control on the right (both plots share the same scale). For clarity, at most 24 worms are shown in each trace. The number of worms (N) is displayed in gray above each plot. The start and end locations for each worm are displayed in gray and black, respectively. The midbody speed is displayed as a heatmap (shown at the top-right corner) from -0.5 (backwards) to 0.5 (forwards) mm/s. Coiling locations are marked with a "+" and omega turns with an "x". Each path has been rotated to align its longest axis vertically, then centered at 0 (see section 4.3.4.2). This rotation and translation permit a guess as to location of the food boundary and, thereby, bordering and food-leaving behaviors. A. CB4856, RC301, and LSJ1 share the same variant *glb-5* and *npr-1* alleles [30-32]. These alleles have been shown to confer high-sensitivity to oxygen concentration, promote high-speed locomotion, increased bordering at the food-lawn boundary, and social aggregation behaviors. *npr-1(ad609)* has a mutant NPR-1 protein within an N2 background. B. *gpa-16*, *npr-3*, and *nlp-17* clustered with the strains in panel A. These three strains show similar path traces to those in panel A with regards to speed and exploratory range (along with multiple other shared phenotypes -- see Table 4.2) and, therefore, their gene products are candidates to interact with GLB-5 and NPR-1.

Genotype	Strain	Terms
gpa-16(ok2349)I	RB1816	-Head Bends (****), ΔTail Bends (****), -Posture Amplitude (**), -Coils (****), ΔForaging (****), +Forward Velocity (****), +Backward Velocity (****), +Forward Motion (****), -Pausing (****), ΔBackward Motion (****), ΔCrawling Amplitude (****), ΔCrawling Frequency (****), ΔTurns (****), +Path Range (****), -Path Curvature (****)
nlp-17(ok3461)IV	RB2498	ΔProportion (*), ΔHead Bends (**), -Tail Bends (*), -Posture Amplitude (**), +Coils (***), ΔForaging (**), +Forward Velocity (***), +Backward Velocity (***), +Forward Motion (***), -Pausing (****), ΔBackward Motion (***), ΔCrawling Amplitude (*), ΔCrawling Frequency (****), ΔTurns (***), +Path Range (***), -Path Curvature (***)
npr-1(ad609)X	DA609	-Width (**), ΔHead Bends (****), ΔTail Bends (**), -Posture Amplitude (***), -Posture Wavelength (***), ΔCoils (****), ΔForaging (****), +Forward Velocity (****), +Backward Velocity (****), ΔForward Motion (****), -Pausing (****), ΔBackward Motion (****), ΔCrawling Amplitude (****), ΔCrawling Frequency (****), ΔTurns (****), +Path Range (****), -Path Curvature (****)
npr-3(tm1583)IV	FX1583	-Width (****), ΔHead Bends (****), ΔTail Bends (***), ΔPosture Amplitude (***), -Posture Wavelength (***), ΔCoils (****), ΔForaging (****), +Forward Velocity (****), +Backward Velocity (****), ΔForward Motion (****), -Pausing (****), ΔBackward Motion (****), ΔCrawling Amplitude (****), ΔCrawling Frequency (****), ΔTurns (****), +Path Range (****), -Path Curvature (****)
Wild Isolate (Hawaii, USA)	CB4856	-Length (**), -Width (***), ΔHead Bends (****), ΔTail Bends (****), -Posture Amplitude (****), -Coils (***), ΔForaging (****), +Forward Velocity (****), +Backward Velocity (****), +Forward Motion (****), -Pausing (****), ΔBackward Motion (****), ΔCrawling Amplitude (****), ΔCrawling Frequency (****), ΔTurns (****), +Path Range (****), -Path Curvature (****)
Wild Isolate (Freiburg, Germany)	RC301	-Length (**), -Width (**), ΔHead Bends (****), -Tail Bends (****), ΔPosture Amplitude (*), -Coils (**), ΔForaging (****), +Forward Velocity (****), +Backward Velocity (****), +Forward Motion (****), ΔPausing (****), ΔBackward Motion (****), ΔCrawling Amplitude (****), ΔCrawling Frequency (****), ΔTurns (****), +Path Range (****), -Path Curvature (****)
Axenic Liquid Culture (Bristol, UK)	LSJ1	-Length (****), -Width (****), ΔHead Bends (****), ΔTail Bends (****), -Posture Amplitude (****), -Posture Wavelength (****), +Coils (****), ΔForaging (****), +Forward Velocity (****), +Backward Velocity (****), ΔForward Motion (****), -Pausing (****), ΔBackward Motion (****), ΔCrawling Amplitude (****), ΔCrawling Frequency (****), ΔTurns (****), +Path Range (****), ΔPath Curvature (****)

Table 4.2: Phenotypic Ontology for GLB-5, NPR-1, and Clustering Predictions

Seven strains, that clustered together, show shared phenotypes (relative to their wild-type controls), including increased locomotion and exploratory range. Phenotypic ontology is described in section 4.3.3. Briefly, "–" means the associated feature(s) measured less than the control, "+" means they measured greater, and "Δ" means they measured different (but the difference cannot be quantified as simply lesser or greater than). q-values were used to correct for multiple testing. The following convention is used: * when $q \leq 0.05$; ** when $q \leq 0.01$; * when $q \leq 0.001$; and, **** when $q \leq 0.0001$.**

4.6 Bibliography

1. Brenner, S. The genetics of Caenorhabditis elegans. *Genetics* **77**, 71–94 (1974).
2. Sokolowski, M. B. Drosophila: genetics meets behaviour. *Nat. Rev. Genet.* **2**, 879–890 (2001).
3. Granato, M. *et al.* Genes controlling and mediating locomotion behavior of the zebrafish embryo and larva. *Development* **123**, 399–413 (1996).
4. Vitaterna, M. H. *et al.* Mutagenesis and mapping of a mouse gene, Clock, essential for circadian behavior. *Science* **264**, 719–725 (1994).
5. Hrabé de Angelis, M. H. *et al.* Genome-wide, large-scale production of mutant mice by ENU mutagenesis. *Nat. Genet.* **25**, 444–447 (2000).
6. Brown, S. D. M. *et al.* A systematic, genome-wide, phenotype-driven mutagenesis programme for gene function studies in the mouse. *Nat. Genet.* **25**, 440–443 (2000).
7. Richmond, J. Synaptic function. *WormBook* 1–14 (2005).doi:10.1895/wormbook.1.69.1
8. Jin, Y. Synaptogenesis. *WormBook* 1–11 (2005).doi:10.1895/wormbook.1.44.1
9. Chase, D. L. & Koelle, M. R. Biogenic amine neurotransmitters in C. elegans. *WormBook* 1–15 (2007).doi:10.1895/wormbook.1.132.1
10. Li, C. Neuropeptides. *WormBook* 1–36 (2008).doi:10.1895/wormbook.1.142.1
11. Goodman, M. B. Mechanosensation. *WormBook* 1–14 (2006).doi:10.1895/wormbook.1.62.1
12. Ramani, A. K. *et al.* The Majority of Animal Genes Are Required for Wild-Type Fitness. *Cell* **148**, 792–802 (2012).
13. Houle, D., Govindaraju, D. R. & Omholt, S. Phenomics: the next challenge. *Nature Publishing Group* **11**, 855–866 (2010).
14. Sulston, J. E. & Horvitz, H. R. Post-embryonic cell lineages of the nematode, Caenorhabditis elegans. *Developmental Biology* **56**, 110–156 (1977).
15. Stephens, G. J., Johnson-Kerner, B., Bialek, W. & Ryu, W. S. Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* **4**, e1000028 (2008).
16. White, J. G., Southgate, E., Thomson, J. N. & Brenner, S. The structure of the nervous system of the nematode Caenorhabditis elegans. *Philos. Trans. R. Soc. Lond., B, Biol. Sci.* **314**, 1–340 (1986).
17. Alkema, M. J., Hunter-Ensor, M., Ringstad, N. & Horvitz, H. R. Tyramine Functions Independently of Octopamine in the Caenorhabditis elegans Nervous System. *Neuron* **46**, 247–260 (2005).
18. Cronin, C. J. *et al.* An automated system for measuring parameters of nematode sinusoidal movement. *BMC Genet.* **6**, 5 (2005).
19. Huang, K.-M., Cosman, P. & Schafer, W. R. Automated detection and analysis of foraging behavior in Caenorhabditis elegans. *Journal of Neuroscience Methods* **171**, 153–164 (2008).
20. Huang, K.-M., Cosman, P. & Schafer, W. R. Machine vision based detection of omega bends and reversals in C. elegans. *Journal of Neuroscience Methods* **158**, 323–336 (2006).
21. Storey, J. D. A direct approach to false discovery rates. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **64**, 479–498 (2002).
22. Tsai, C. A. & Chen, J. J. Multivariate analysis of variance test for gene set analysis. *Bioinformatics* **25**, 897–903 (2009).

23. Hart, A. C. Behavior. *WormBook* 1–67 (2006).doi:10.1895/wormbook.1.87.1
24. Geng, W., Cosman, P., Berry, C. C., Feng, Z. & Schafer, W. R. Automatic tracking, feature extraction and classification of C elegans phenotypes. *IEEE Trans. Biomed. Eng.* **51**, 1811–1820 (2004).
25. Hoshi, K. & Shingai, R. Computer-driven automatic identification of locomotion states in Caenorhabditis elegans. *Journal of Neuroscience Methods* **157**, 355–363 (2006).
26. Zhao, B., Khare, P., Feldman, L. & Dent, J. A. Reversal frequency in Caenorhabditis elegans represents an integrated response to the state of the animal and its environment. *Journal of Neuroscience* **23**, 5319–5328 (2003).
27. Feng, Z. *et al.* A C. elegans Model of Nicotine-Dependent Behavior: Regulation by TRP-Family Channels. *Cell* **127**, 621–633 (2006).
28. Denver, D. R., Morris, K., Lynch, M. & Thomas, W. K. High mutation rate and predominance of insertions in the Caenorhabditis elegans nuclear genome. *Nature* **430**, 679–682 (2004).
29. Weber, K. P. *et al.* Whole Genome Sequencing Highlights Genetic Changes Associated with Laboratory Domestication of C. elegans. *PLoS ONE* **5**, e13922 (2010).
30. de Bono, M. & Bargmann, C. I. Natural variation in a neuropeptide Y receptor homolog modifies social behavior and food response in C. elegans. *Cell* **94**, 679–689 (1998).
31. McGrath, P. T. *et al.* Quantitative mapping of a digenic behavioral trait implicates globin variation in C. elegans sensory behaviors. *Neuron* **61**, 692–699 (2009).
32. Persson, A. *et al.* Natural variation in a neural globin tunes oxygen sensing in wild Caenorhabditis elegans. *Nature* **458**, 1030–1033 (2009).
33. WormBase web site. *http://www.wormbase.org* at <http://www.wormbase.org>
34. Kindt, K. S. *et al.* Caenorhabditis elegans TRPA-1 functions in mechanosensation. *Nat Neurosci* **10**, 568–577 (2007).
35. Etchberger, J. F. *et al.* The molecular signature and cis-regulatory architecture of a C. elegans gustatory neuron. *Genes & Development* **21**, 1653–1674 (2007).
36. Eisen, M. B., Spellman, P. T., Brown, P. O. & Botstein, D. Cluster analysis and display of genome-wide expression patterns. *pnas.org*
37. Shimodaira, H. Approximately unbiased tests of regions using multistep-multiscale bootstrap resampling. *Ann. Statist.* **32**, 2616–2641 (2004).
38. Hajdu-Cronin, Y. M., Chen, W. J., Patikoglou, G., Koelle, M. R. & Sternberg, P. W. Antagonism between G(o)alpha and G(q)alpha in Caenorhabditis elegans: the RGS protein EAT-16 is necessary for G(o)alpha signaling and regulates G(q)alpha activity. *Genes & Development* **13**, 1780–1793 (1999).
39. Boulin, T. *et al.* Eight genes are required for functional reconstitution of the Caenorhabditis elegans levamisole-sensitive acetylcholine receptor. *Proc. Natl. Acad. Sci. U.S.A.* **105**, 18590–18595 (2008).
40. Yeh, E. *et al.* A putative cation channel, NCA-1, and a novel protein, UNC-80, transmit neuronal activity in C. elegans. *Plos Biol* **6**, e55 (2008).
41. Pflugrad, A., Meir, J. Y., Barnes, T. M. & Miller, D. M. The Groucho-like transcription factor UNC-37 functions with the neural specificity gene unc-4 to govern motor neuron identity in C. elegans. *Development* **124**, 1699–1709 (1997).
42. Hare, E. E. & Loer, C. M. Function and evolution of the serotonin-synthetic bas-1 gene and other aromatic amino acid decarboxylase genes in

Caenorhabditis. *BMC Evol Biol* **4**, 24 (2004).
43. Hamdan, F. F., Ungrin, M. D., Abramovitz, M. & Ribeiro, P. Characterization of a Novel Serotonin Receptor from Caenorhabditis elegans. *Journal of Neurochemistry* **72**, 1372–1383 (2001).
44. Olde, B. & McCombie, W. R. Molecular cloning and functional expression of a serotonin receptor from Caenorhabditis elegans. *J. Mol. Neurosci.* **8**, 53–62 (1997).
45. Hobson, R. J. *et al.* SER-7, a Caenorhabditis elegans 5-HT7-like receptor, is essential for the 5-HT stimulation of pharyngeal pumping and egg laying. *Genetics* **172**, 159–169 (2006).
46. Rex, E. & Komuniecki, R. W. Characterization of a tyramine receptor from Caenorhabditis elegans. *Journal of Neurochemistry* **82**, 1352–1359 (2002).
47. Starich, T., Sheehan, M., Jadrich, J. & Shaw, J. Innexins in C. elegans. *Cell Communication & Adhesion* (2001).
48. Mongan, N. P., Jones, A. K., Smith, G. R., Sansom, M. S. P. & Sattelle, D. B. Novel alpha7-like nicotinic acetylcholine receptor subunits in the nematode Caenorhabditis elegans. *Protein Sci.* **11**, 1162–1171 (2002).
49. Squire, M. D. *et al.* Molecular cloning and functional co-expression of a Caenorhabditis elegans nicotinic acetylcholine receptor subunit (acr-2). *Recept. Channels* **3**, 107–115 (1995).
50. Baylis, H. A. *et al.* ACR-3, a Caenorhabditis elegans nicotinic acetylcholine receptor subunit. Molecular cloning and functional expression. *Recept. Channels* **5**, 149–158 (1997).
51. Stawicki, T. M., Zhou, K., Yochem, J., Chen, L. & Jin, Y. TRPM channels modulate epileptic-like convulsions via systemic ion homeostasis. *Curr. Biol.* **21**, 883–888 (2011).
52. Rand, J. B. Acetylcholine. *WormBook* 1–21 (2007).doi:10.1895/wormbook.1.131.1
53. Ghosh, R., Mohammadi, A., Kruglyak, L. & Ryu, W. S. Multiparameter behavioral profiling reveals distinct thermal response regimes in Caenorhabditis elegans. *BMC Biol.* **10**, 85 (2012).
54. van Swinderen, B. *et al.* Goalpha regulates volatile anesthetic action in Caenorhabditis elegans. *Genetics* **158**, 643–655 (2001).
55. Jansen, G. *et al.* The complete family of genes encoding G proteins of Caenorhabditis elegans. *Nat. Genet.* **21**, 414–419 (1999).
56. Gotta, M. & Ahringer, J. Distinct roles for Galpha and Gbetagamma in regulating spindle position and orientation in Caenorhabditis elegans embryos. *Nature cell biology* **3**, 297–300 (2001).
57. Nathoo, A. N., Moeller, R. A., Westlund, B. A. & Hart, A. C. Identification of neuropeptide-like protein gene families in Caenorhabditis elegans and other species. *Proceedings of the National Academy of Sciences* **98**, 14000–14005 (2001).
58. Kubiak, T. M. *et al.* Functional annotation of the putative orphan Caenorhabditis elegans G-protein-coupled receptor C10C6.2 as a FLP15 peptide receptor. *J. Biol. Chem.* **278**, 42115–42120 (2003).
59. Zhang, S. *et al.* Profiling a Caenorhabditis elegans behavioral parametric dataset with a supervised K-means clustering algorithm identifies genetic networks regulating locomotion. *Journal of Neuroscience Methods* **197**, 315–323 (2011).

Appendix A : Strains

A list of strains used in this work. Each strain is listed alongside its genotype and the number of times it has been outcrossed. Strains with no previously characterized phenotype (as of Wormbase version 233, 07/10/2012) are highlighted.

Genotype	Strain	Outcrossed
acc-4(ok2371)III	RB1832	0
acd-2(ok1237)I	RB1192	0
acd-5(ok2657)I	RB2005	0
acr-2(ok1887)X	RB1559	0
acr-3(ok2049)X	RB1659	0
acr-6(ok3117)I	RB2294	0
acr-7(tm863)II	FX863	0
acr-9(ok933)X	VC649	0
acr-10(ok3064)X	RB2262	0
acr-11(ok1345)I	RB1263	0
acr-14(ok1155)II	RB1132	0
acr-15(ok1214)V	RB1172	0
acr-18(ok1285)V	RB1226	0
acr-19(ad1674)I	DA1674	1
acr-21(ok1314)III	RB1250	0
acr-23(ok2804)V	RB2119	0
asic-1(ok415)I	RB680	0
asic-2(ok289)I	RB557	0
bas-1(ad446)III	MT7988	2
C11D2.2(ok1565)IV	RB1380	0
C24G7.1(ok1822)I	RB1523	0
C38D9.2(ok1853)V	RB1543	0
cat-2(e1112)II	CB1112	0
cat-4(e1141)V	CB1141	0
daf-3(e1376)X	CB1376	0
daf-5(e1386)II	CB1386	0
daf-7(m62)III	DR62	0
dat-1(ok157)III	RM2702	6
del-1(ok150)X	NC279	4
del-4(ok1014)I	RB1064	0
del-7(ok1187)IV	RB1156	0
del-9(ok2353)X	RB1818	0
dnc-1(or404)IV	EU1006	4
dop-1(vs100)dop-3(vs106)X	LX705	0
dop-1(vs101)X	LX636	4
dop-2(vs105)V	LX702	4
dop-2(vs105)V; dop-1(vs100)X	LX706	0
dop-2(vs105)V; dop-3(vs106)X	LX704	0
dop-2(vs105)V; dop-1(vs100)dop-3(vs106)X	LX734	0
dop-3(vs106)X	LX703	4
dop-4(tm1392)X	FG58	5

dpy-20(e1282)IV	CB1282	0
eat-16(sa609)I	JT609	3
egas-2(ok1477)V	VC975	0
egas-3(ok1522)V	RB1356	0
egg-5(ok1781)I	VC1295	0
egl-1(n487)V	MT1082	1
egl-2(n693)V	MT1444	1
egl-5(n486)III	MT1081	1
egl-6(n592)X	MT1222	0
egl-7(n575)III	MT1205	0
egl-8(n488)V	MT1083	1
egl-9(n586)V	MT1216	0
egl-10(md176)V	MT8504	6
egl-11(n587)V	MT1217	0
egl-12(n602)V	MT1232	0
egl-13(n483)X	MT1078	1
egl-14(n549)X	MT1179	0
egl-15(n484)X	MT1079	1
egl-17(e1313)X	CB1313	1
egl-18(ok290)IV	JR2370	4
egl-19(n2368)IV	MT6129	0
egl-20(mu39)IV	CF263	5
egl-21(n476)IV	KP2018	5
egl-21(n611)IV	MT1241	0
egl-23(n601)IV	MT1231	1
egl-24(n572)III	MT1202	0
egl-27(ok151)II	KS99	6
egl-28(n570)II	MT1200	1
egl-30(ep271)I	CE1047	2
egl-30(n686)goa-1(n1134)I	AQ916	0
egl-31(n472)I	MT1067	1
egl-32(n155)I	MT155	1
egl-33(n151)I	MT151	1
egl-36(n728)X	MT1540	1
egl-37(n742)II	MT1543	0
egl-40(n606)IV	MT1236	0
egl-42(n995)II	MT2068	0
egl-44(n1080)II	MT2247	0
egl-46(n1127)V	MT2316	0
egl-47(n1081)V	MT2248	0
egl-49(n1107)X	MT2293	0
egl-50(n1086)II	AQ2316	5
ets-10(gk596)X	VC1340	0
F23B2.3(ok1226)IV	RB1177	0
F54E4.4(ok2336)X	RB1802	0
flp-1(yn2)IV	NY7	0
flp-3(ok3265)X	VC2497	0
flp-6(ok3056)V	VC2324	0
flp-7(ok2625)X	RB1990	0
flp-9(ok2730)IV	RB2067	0
flp-10(ok2624)IV	RB1989	0
flp-11(tm2706)X	FX2706	0
flp-12(ok2409)X	RB1863	0
flp-13(tm2427)IV	FX2427	0
flp-16(ok3085)II	RB2275	0

flp-17(ok3587)IV	RB2575	0
flp-18(db99)X	AX1410	6
flp-19(ok2460)X	RB1902	0
flp-20(ok2964)X	RB2188	0
flp-21(ok889)V	RB982	0
flp-25(gk1016)III	VC1982	0
flp-28(gk1075)X	VC2502	0
flp-33(gk1038)I	VC2423	0
flr-1(ut11)X	JC55	0
gar-2(ok520)III	RB756	0
gld-1(op236)I	TG34	5
gly-2(gk204)I	VC335	0
goa-1(sa734)I	DG1856	6
gon-2(q362)I	EJ26	4
gpa-1(pk15)V	NL332	6
gpa-2(pk16)V	NL334	7
gpa-3(pk35)V	NL335	7
gpa-4(pk381)IV	NL790	6
gpa-5(pk376)X	NL1137	6
gpa-6(pk480)X	NL1146	6
gpa-7(pk610)IV	NL795	6
gpa-8(pk345)V	NL1142	6
gpa-9(pk438)V	NL793	6
gpa-10(pk362)V	NL1147	6
gpa-11(pk349)II	NL787	6
gpa-12(pk322)X	NL594	6
gpa-13(pk1270)V	NL2330	6
gpa-14(pk347)I	AQ495	1
gpa-15(pk477)I	NL797	6
gpa-16(ok2349)I	RB1816	0
gpa-17(ok2334)III	RB1800	0
gpb-2(sa603)I	JT603	2
gpc-1(pk298)X	NL792	6
hcf-1(ok559)IV	RB777	0
ins-3(ok2488)II	RB1915	0
ins-4(ok3534)II	RB2544	0
ins-11(tm1053)II	FX1053	5
ins-15(ok3444)II	RB2489	0
ins-16(ok2919)III	RB2159	0
ins-18(ok1672)I	VC1218	1
ins-22(ok3616)III	RB2594	0
ins-25(ok2773)I	RB2098	0
ins-27(ok2474)I	RB1911	0
ins-28(ok2722)I	RB2059	0
ins-30(ok2343)I	RB1809	0
ins-31(ok3543)II	RB2552	0
ins-35(ok3297)V	RB2412	0
jnk-1(gk7)IV	VC8	0
lev-1(x427)IV	ZZ427	4
lev-8(x15)X	ZZ15	0
lig-4(ok716)III	RB873	0
lin-39(n709)III	MT1514	1
lon-2(e678)X	CB678	0
lov-1(ok522)II	RB753	0
mec-4(u253)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ908	0

mec-7(u448)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1033	0
mec-10(e1515)X	CB1515	0
mec-10(tm1552)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1413	0
mec-10(u20)X	AQ2533	0
mec-12(e1605)III; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1031	0
mec-12(u76)III; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1037	0
mec-14(u55)III; bzIs18[pmec-4::YC2.12; lin-15(+)]	AQ1038	0
mec-18(u228)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ2649	0
mir-124(n4255)IV	MT13292	2
mod-1(ok103)V	MT9668	6
mod-5(n822)I	MT8944	2
nca-2(gk5)III	VC9	0
nhr-95(gk836)V	VC1759	0
nlp-1(ok1469)X	RB1340	0
nlp-2(tm1908)X	FX1908	0
nlp-3(ok2688)X	RB2030	0
nlp-8(ok1799)I	VC1309	0
nlp-12(ok335)IV	RB607	0
nlp-14(tm1880)X	FX1880	0
nlp-15(ok1512)I	VC1063	0
nlp-17(ok3461)IV	RB2498	0
nlp-18(ok1557)II	RB1372	0
nlp-20(ok1591)IV	RB1396	0
npr-1(ad609)X	DA609	2
npr-2(ok419)IV	AQ2056	0
npr-3(tm1583)IV	FX1583	0
npr-4(tm1782)X	AX1743	6
npr-5(ok1583)V	AX1745	6
npr-7(ok527)X	RB761	0
npr-8(tm1553)X	FX1553	0
npr-9(tm1652)X	IC683	2
npr-10(tm1568)X	FX1568	0
npr-11(ok594)X	RB799	0
npr-12(tm1498)IV	FX1498	0
npr-13(tm1504)V	AQ2153	6
npr-20(ok2575)II	RB1958	0
ocr-3(ok1559)X	RB1374	0
ocr-4(tm2173)IV	FX2173	0
ocr-4(vs137)IV	LX950	4
ocr-4(vs137)ocr-2(ak47)IV	LX981	0
ocr-4(vs137)ocr-2(ak47)IV; ocr-1(ok132)V	LX982	0
octr-1(ok371)X	VC224	0
odr-3(n2150)V	CX2205	3
osm-9(ky10)IV	CX10	0
osm-9(ky10)trpa-1(ok999)IV	AQ1422	0
pdl-1(gk157)II	VC282	0
pkc-1(nj3)V	IK130	6
pkg-1(n478)IV	MT1073	1
pmk-1(km25)IV	KU25	6
pqn-66(ok1507)II	RB1350	0
rab-3(y250)II	NM210	2
ric-19(ok833)	RB946	0
sem-4(ga82)I	EW35	3
ser-1(ok345)X	DA1814	10

ser-2(pk1357)X	OH313	4
ser-4(ok512)III	AQ866	5
ser-5(tm2654)I	AQ2197	6
ser-6(tm2146)IV	FX2146	0
ser-7(tm1325)X	DA2100	10
sma-2(e502)III	CB502	0
sma-3(e491)III	CB491	0
snf-1(ok790)I	RB919	0
snf-2(ok147)I	RB641	0
snf-4(ok496)II	RB738	0
snf-5(ok447)II	RB687	0
snf-6(eg28)III	BZ28	4
snf-7(ok482)III	RB709	0
snf-8(ok349)IV	RB648	0
snf-9(ok957)IV	RB1030	0
snf-10(hc194)V	BA1093	6
snf-11(ok156)V	RM2710	6
sng-1(ok234)X	RB503	0
spe-41(sy693)III; him-5(e1490)V	PS4330	8
srp-8(ok291)V	RB559	0
sup-9(n180)II	MT180	0
syg-1(ok3640)X	RB2615	0
syg-2(ky671)X	CX6391	0
T14B1.1(ok1702)X	VC1243	1
tbh-1(n3247)X	MT9455	8
tdc-1(n3419)II	MT13113	11
tom-1(ok285)I	VC223	0
tph-1(mg280)II	MT15434	4
trp-1(sy690)III	TQ225	7
trp-2(gk298)III	VC602	0
trp-2(sy691)III	TQ194	8
trp-4(sy695)I	TQ296	8
trpa-1(ok999)IV	RB1052	0
trpa-2(ok3189)I	RB2351	0
trpa-2(tm3085)I	FX3085	0
trpa-2(tm3092)I	FX3092	0
trpl-2(ok2433)IV	RB1883	0
tyra-2(tm1846)X	FX1846	0
tyra-3(ok325)X	VC125	0
unc-1(e1598)X	CB1598	0
unc-1(e94)X	CB94	0
unc-2(gk366)X	VC854	0
unc-2(ox106)X	EG106	0
unc-3(e151)X	CB151	0
unc-4(e120)II	CB120	0
unc-4(gk705)II	VC1528	1
unc-7(e5)X	CB5	0
unc-8(e15)IV	CB15	0
unc-8(e15lb145)IV	MP145	3
unc-8(n491n1192)IV	MT2611	0
unc-8(e15lb145)IV; del-1(ok150)X	AQ2937	0
unc-9(e101)X	CB101	0
unc-10(e102)X	CB102	0
unc-10(md1117)X	NM1657	5
unc-14(e57)I	CB57	0

unc-16(e109)III	CB109	0
unc-18(e81)X	CB81	0
unc-26(m2)IV	DR2	0
unc-29(e193)I	CB193	4
unc-30(e191)IV	CB845	0
unc-31(e169)IV	CB169	0
unc-32(e189)III	CB189	0
unc-34(e566)V	CB566	1
unc-37(e262)I	CB262	0
unc-38(e264)I	CB904	4
unc-40(n324)I	MT324	0
unc-42(e270)V	CB270	0
unc-44(e1197)IV	CB1197	0
unc-55(e402)I	CB402	0
unc-60(e723)V	CB723	0
unc-63(ok1075)I	VC731	1
unc-69(e587)III	CB587	0
unc-75(e950)I	CB950	0
unc-76(e911)V	DR96	0
unc-77(e625)IV	DR1089	0
unc-77(gk9)IV	VC12	0
unc-79(e1068)III	CB1068	0
unc-80(e1069)V	CB1069	0
unc-86(e1416)III	CB1416	0
unc-89(e1460)I	CB1460	0
unc-89(st85)I	RW85	0
unc-98(su130)X	HE130	0
unc-101(m1)I	DR1	0
unc-103(e1597)III	CB1597	0
unc-104(e1265)II	CB1265	0
unc-105(ok1432)II	RB1316	0
unc-108(n501)I	MT1093	0
unc-108(n777)I	MT1656	0
unc-115(mn481)X	SP1789	0
unc-116(e2310)III	FF41	15
unc-118(e2331)X	CB4371	0
unc-122(e2520)I	CB4870	2
unc-127(hs13)V	HH27	2
vab-7(e1562)III	CB1562	0
zyg-9(b244)II	DH244	2
C. elegans Wild Isolate CB4856 (Hawaii, USA)	CB4856	
C. elegans Wild Isolate RC301 (Freiburg, Germany)	RC301	
Axenic Liquid Culture LSJ1 (Bristol, UK)	LSJ1	
CGC N2 (Bristol, UK)	AQ2947	

Appendix B : Phenotypic Measurements

A list of all 702 phenotypic measurements. Each feature is listed with its name, up to 2 subdivisions, and the units of measure. D = dorsal, V=ventral.

Feature Number	Feature Name	Feature Subdivision 1	Feature Subdivision 2	Units
1	Length			Microns
2	Length	Forward		Microns
3	Length	Paused		Microns
4	Length	Backward		Microns
5	Head Width			Microns
6	Head Width	Forward		Microns
7	Head Width	Paused		Microns
8	Head Width	Backward		Microns
9	Midbody Width			Microns
10	Midbody Width	Forward		Microns
11	Midbody Width	Paused		Microns
12	Midbody Width	Backward		Microns
13	Tail Width			Microns
14	Tail Width	Forward		Microns
15	Tail Width	Paused		Microns
16	Tail Width	Backward		Microns
17	Area			Microns^2
18	Area	Forward		Microns^2
19	Area	Paused		Microns^2
20	Area	Backward		Microns^2
21	Area/Length			Microns
22	Area/Length	Forward		Microns
23	Area/Length	Paused		Microns
24	Area/Length	Backward		Microns
25	Width/Length			No Units
26	Width/Length	Forward		No Units
27	Width/Length	Paused		No Units
28	Width/Length	Backward		No Units
29	Head Bend Mean (+/- = D/V Inside)			Degrees
30	Head Bend Mean (+/- = D/V Inside)		Absolute	Degrees
31	Head Bend Mean (+/- = D/V Inside)		Positive	Degrees
32	Head Bend Mean (+/- = D/V Inside)		Negative	Degrees
33	Head Bend Mean (+/- = D/V Inside)	Forward		Degrees
34	Head Bend Mean (+/- = D/V Inside)	Forward	Absolute	Degrees
35	Head Bend Mean (+/- = D/V Inside)	Forward	Positive	Degrees
36	Head Bend Mean (+/- = D/V Inside)	Forward	Negative	Degrees
37	Head Bend Mean (+/- = D/V Inside)	Paused		Degrees
38	Head Bend Mean (+/- = D/V Inside)	Paused	Absolute	Degrees
39	Head Bend Mean (+/- = D/V Inside)	Paused	Positive	Degrees
40	Head Bend Mean (+/- = D/V Inside)	Paused	Negative	Degrees
41	Head Bend Mean (+/- = D/V Inside)	Backward		Degrees
42	Head Bend Mean (+/- = D/V Inside)	Backward	Absolute	Degrees
43	Head Bend Mean (+/- = D/V Inside)	Backward	Positive	Degrees
44	Head Bend Mean (+/- = D/V Inside)	Backward	Negative	Degrees
45	Neck Bend Mean (+/- = D/V Inside)			Degrees
46	Neck Bend Mean (+/- = D/V Inside)		Absolute	Degrees
47	Neck Bend Mean (+/- = D/V Inside)		Positive	Degrees
48	Neck Bend Mean (+/- = D/V Inside)		Negative	Degrees
49	Neck Bend Mean (+/- = D/V Inside)	Forward		Degrees
50	Neck Bend Mean (+/- = D/V Inside)	Forward	Absolute	Degrees

51	Neck Bend Mean (+/- = D/V Inside)	Forward	Positive	Degrees
52	Neck Bend Mean (+/- = D/V Inside)	Forward	Negative	Degrees
53	Neck Bend Mean (+/- = D/V Inside)	Paused		Degrees
54	Neck Bend Mean (+/- = D/V Inside)	Paused	Absolute	Degrees
55	Neck Bend Mean (+/- = D/V Inside)	Paused	Positive	Degrees
56	Neck Bend Mean (+/- = D/V Inside)	Paused	Negative	Degrees
57	Neck Bend Mean (+/- = D/V Inside)	Backward		Degrees
58	Neck Bend Mean (+/- = D/V Inside)	Backward	Absolute	Degrees
59	Neck Bend Mean (+/- = D/V Inside)	Backward	Positive	Degrees
60	Neck Bend Mean (+/- = D/V Inside)	Backward	Negative	Degrees
61	Midbody Bend Mean (+/- = D/V Inside)			Degrees
62	Midbody Bend Mean (+/- = D/V Inside)		Absolute	Degrees
63	Midbody Bend Mean (+/- = D/V Inside)		Positive	Degrees
64	Midbody Bend Mean (+/- = D/V Inside)		Negative	Degrees
65	Midbody Bend Mean (+/- = D/V Inside)	Forward		Degrees
66	Midbody Bend Mean (+/- = D/V Inside)	Forward	Absolute	Degrees
67	Midbody Bend Mean (+/- = D/V Inside)	Forward	Positive	Degrees
68	Midbody Bend Mean (+/- = D/V Inside)	Forward	Negative	Degrees
69	Midbody Bend Mean (+/- = D/V Inside)	Paused		Degrees
70	Midbody Bend Mean (+/- = D/V Inside)	Paused	Absolute	Degrees
71	Midbody Bend Mean (+/- = D/V Inside)	Paused	Positive	Degrees
72	Midbody Bend Mean (+/- = D/V Inside)	Paused	Negative	Degrees
73	Midbody Bend Mean (+/- = D/V Inside)	Backward		Degrees
74	Midbody Bend Mean (+/- = D/V Inside)	Backward	Absolute	Degrees
75	Midbody Bend Mean (+/- = D/V Inside)	Backward	Positive	Degrees
76	Midbody Bend Mean (+/- = D/V Inside)	Backward	Negative	Degrees
77	Hips Bend Mean (+/- = D/V Inside)			Degrees
78	Hips Bend Mean (+/- = D/V Inside)		Absolute	Degrees
79	Hips Bend Mean (+/- = D/V Inside)		Positive	Degrees
80	Hips Bend Mean (+/- = D/V Inside)		Negative	Degrees
81	Hips Bend Mean (+/- = D/V Inside)	Forward		Degrees
82	Hips Bend Mean (+/- = D/V Inside)	Forward	Absolute	Degrees
83	Hips Bend Mean (+/- = D/V Inside)	Forward	Positive	Degrees
84	Hips Bend Mean (+/- = D/V Inside)	Forward	Negative	Degrees
85	Hips Bend Mean (+/- = D/V Inside)	Paused		Degrees
86	Hips Bend Mean (+/- = D/V Inside)	Paused	Absolute	Degrees
87	Hips Bend Mean (+/- = D/V Inside)	Paused	Positive	Degrees
88	Hips Bend Mean (+/- = D/V Inside)	Paused	Negative	Degrees
89	Hips Bend Mean (+/- = D/V Inside)	Backward		Degrees
90	Hips Bend Mean (+/- = D/V Inside)	Backward	Absolute	Degrees
91	Hips Bend Mean (+/- = D/V Inside)	Backward	Positive	Degrees
92	Hips Bend Mean (+/- = D/V Inside)	Backward	Negative	Degrees
93	Tail Bend Mean (+/- = D/V Inside)			Degrees
94	Tail Bend Mean (+/- = D/V Inside)		Absolute	Degrees
95	Tail Bend Mean (+/- = D/V Inside)		Positive	Degrees
96	Tail Bend Mean (+/- = D/V Inside)		Negative	Degrees
97	Tail Bend Mean (+/- = D/V Inside)	Forward		Degrees
98	Tail Bend Mean (+/- = D/V Inside)	Forward	Absolute	Degrees
99	Tail Bend Mean (+/- = D/V Inside)	Forward	Positive	Degrees
100	Tail Bend Mean (+/- = D/V Inside)	Forward	Negative	Degrees
101	Tail Bend Mean (+/- = D/V Inside)	Paused		Degrees
102	Tail Bend Mean (+/- = D/V Inside)	Paused	Absolute	Degrees
103	Tail Bend Mean (+/- = D/V Inside)	Paused	Positive	Degrees
104	Tail Bend Mean (+/- = D/V Inside)	Paused	Negative	Degrees
105	Tail Bend Mean (+/- = D/V Inside)	Backward		Degrees
106	Tail Bend Mean (+/- = D/V Inside)	Backward	Absolute	Degrees
107	Tail Bend Mean (+/- = D/V Inside)	Backward	Positive	Degrees
108	Tail Bend Mean (+/- = D/V Inside)	Backward	Negative	Degrees
109	Head Bend S.D. (+/- = D/V Inside)			Degrees
110	Head Bend S.D. (+/- = D/V Inside)		Absolute	Degrees
111	Head Bend S.D. (+/- = D/V Inside)		Positive	Degrees
112	Head Bend S.D. (+/- = D/V Inside)		Negative	Degrees
113	Head Bend S.D. (+/- = D/V Inside)	Forward		Degrees
114	Head Bend S.D. (+/- = D/V Inside)	Forward	Absolute	Degrees
115	Head Bend S.D. (+/- = D/V Inside)	Forward	Positive	Degrees
116	Head Bend S.D. (+/- = D/V Inside)	Forward	Negative	Degrees

117	Head Bend S.D. (+/- = D/V Inside)	Paused		Degrees
118	Head Bend S.D. (+/- = D/V Inside)	Paused	Absolute	Degrees
119	Head Bend S.D. (+/- = D/V Inside)	Paused	Positive	Degrees
120	Head Bend S.D. (+/- = D/V Inside)	Paused	Negative	Degrees
121	Head Bend S.D. (+/- = D/V Inside)	Backward		Degrees
122	Head Bend S.D. (+/- = D/V Inside)	Backward	Absolute	Degrees
123	Head Bend S.D. (+/- = D/V Inside)	Backward	Positive	Degrees
124	Head Bend S.D. (+/- = D/V Inside)	Backward	Negative	Degrees
125	Neck Bend S.D. (+/- = D/V Inside)			Degrees
126	Neck Bend S.D. (+/- = D/V Inside)		Absolute	Degrees
127	Neck Bend S.D. (+/- = D/V Inside)		Positive	Degrees
128	Neck Bend S.D. (+/- = D/V Inside)		Negative	Degrees
129	Neck Bend S.D. (+/- = D/V Inside)	Forward		Degrees
130	Neck Bend S.D. (+/- = D/V Inside)	Forward	Absolute	Degrees
131	Neck Bend S.D. (+/- = D/V Inside)	Forward	Positive	Degrees
132	Neck Bend S.D. (+/- = D/V Inside)	Forward	Negative	Degrees
133	Neck Bend S.D. (+/- = D/V Inside)	Paused		Degrees
134	Neck Bend S.D. (+/- = D/V Inside)	Paused	Absolute	Degrees
135	Neck Bend S.D. (+/- = D/V Inside)	Paused	Positive	Degrees
136	Neck Bend S.D. (+/- = D/V Inside)	Paused	Negative	Degrees
137	Neck Bend S.D. (+/- = D/V Inside)	Backward		Degrees
138	Neck Bend S.D. (+/- = D/V Inside)	Backward	Absolute	Degrees
139	Neck Bend S.D. (+/- = D/V Inside)	Backward	Positive	Degrees
140	Neck Bend S.D. (+/- = D/V Inside)	Backward	Negative	Degrees
141	Midbody Bend S.D. (+/- = D/V Inside)			Degrees
142	Midbody Bend S.D. (+/- = D/V Inside)		Absolute	Degrees
143	Midbody Bend S.D. (+/- = D/V Inside)		Positive	Degrees
144	Midbody Bend S.D. (+/- = D/V Inside)		Negative	Degrees
145	Midbody Bend S.D. (+/- = D/V Inside)	Forward		Degrees
146	Midbody Bend S.D. (+/- = D/V Inside)	Forward	Absolute	Degrees
147	Midbody Bend S.D. (+/- = D/V Inside)	Forward	Positive	Degrees
148	Midbody Bend S.D. (+/- = D/V Inside)	Forward	Negative	Degrees
149	Midbody Bend S.D. (+/- = D/V Inside)	Paused		Degrees
150	Midbody Bend S.D. (+/- = D/V Inside)	Paused	Absolute	Degrees
151	Midbody Bend S.D. (+/- = D/V Inside)	Paused	Positive	Degrees
152	Midbody Bend S.D. (+/- = D/V Inside)	Paused	Negative	Degrees
153	Midbody Bend S.D. (+/- = D/V Inside)	Backward		Degrees
154	Midbody Bend S.D. (+/- = D/V Inside)	Backward	Absolute	Degrees
155	Midbody Bend S.D. (+/- = D/V Inside)	Backward	Positive	Degrees
156	Midbody Bend S.D. (+/- = D/V Inside)	Backward	Negative	Degrees
157	Hips Bend S.D. (+/- = D/V Inside)			Degrees
158	Hips Bend S.D. (+/- = D/V Inside)		Absolute	Degrees
159	Hips Bend S.D. (+/- = D/V Inside)		Positive	Degrees
160	Hips Bend S.D. (+/- = D/V Inside)		Negative	Degrees
161	Hips Bend S.D. (+/- = D/V Inside)	Forward		Degrees
162	Hips Bend S.D. (+/- = D/V Inside)	Forward	Absolute	Degrees
163	Hips Bend S.D. (+/- = D/V Inside)	Forward	Positive	Degrees
164	Hips Bend S.D. (+/- = D/V Inside)	Forward	Negative	Degrees
165	Hips Bend S.D. (+/- = D/V Inside)	Paused		Degrees
166	Hips Bend S.D. (+/- = D/V Inside)	Paused	Absolute	Degrees
167	Hips Bend S.D. (+/- = D/V Inside)	Paused	Positive	Degrees
168	Hips Bend S.D. (+/- = D/V Inside)	Paused	Negative	Degrees
169	Hips Bend S.D. (+/- = D/V Inside)	Backward		Degrees
170	Hips Bend S.D. (+/- = D/V Inside)	Backward	Absolute	Degrees
171	Hips Bend S.D. (+/- = D/V Inside)	Backward	Positive	Degrees
172	Hips Bend S.D. (+/- = D/V Inside)	Backward	Negative	Degrees
173	Tail Bend S.D. (+/- = D/V Inside)			Degrees
174	Tail Bend S.D. (+/- = D/V Inside)		Absolute	Degrees
175	Tail Bend S.D. (+/- = D/V Inside)		Positive	Degrees
176	Tail Bend S.D. (+/- = D/V Inside)		Negative	Degrees
177	Tail Bend S.D. (+/- = D/V Inside)	Forward		Degrees
178	Tail Bend S.D. (+/- = D/V Inside)	Forward	Absolute	Degrees
179	Tail Bend S.D. (+/- = D/V Inside)	Forward	Positive	Degrees
180	Tail Bend S.D. (+/- = D/V Inside)	Forward	Negative	Degrees
181	Tail Bend S.D. (+/- = D/V Inside)	Paused		Degrees
182	Tail Bend S.D. (+/- = D/V Inside)	Paused	Absolute	Degrees

183	Tail Bend S.D. (+/- = D/V Inside)	Paused	Positive	Degrees
184	Tail Bend S.D. (+/- = D/V Inside)	Paused	Negative	Degrees
185	Tail Bend S.D. (+/- = D/V Inside)	Backward		Degrees
186	Tail Bend S.D. (+/- = D/V Inside)	Backward	Absolute	Degrees
187	Tail Bend S.D. (+/- = D/V Inside)	Backward	Positive	Degrees
188	Tail Bend S.D. (+/- = D/V Inside)	Backward	Negative	Degrees
189	Max Amplitude			Microns
190	Max Amplitude	Forward		Microns
191	Max Amplitude	Paused		Microns
192	Max Amplitude	Backward		Microns
193	Amplitude Ratio			No Units
194	Amplitude Ratio	Forward		No Units
195	Amplitude Ratio	Paused		No Units
196	Amplitude Ratio	Backward		No Units
197	Primary Wavelength			Microns
198	Primary Wavelength	Forward		Microns
199	Primary Wavelength	Paused		Microns
200	Primary Wavelength	Backward		Microns
201	Secondary Wavelength			Microns
202	Secondary Wavelength	Forward		Microns
203	Secondary Wavelength	Paused		Microns
204	Secondary Wavelength	Backward		Microns
205	Track Length			Microns
206	Track Length	Forward		Microns
207	Track Length	Paused		Microns
208	Track Length	Backward		Microns
209	Eccentricity			No Units
210	Eccentricity	Forward		No Units
211	Eccentricity	Paused		No Units
212	Eccentricity	Backward		No Units
213	Bend Count			Counts
214	Bend Count	Forward		Counts
215	Bend Count	Paused		Counts
216	Bend Count	Backward		Counts
217	Coiling Events	Frequency		Hz
218	Coiling Events	Time Ratio		No Units
219	Coiling Events	Time		Seconds
220	Coiling Events	Inter Time		Seconds
221	Coiling Events	Inter Distance		Microns
222	Tail-To-Head Orientation			Degrees
223	Tail-To-Head Orientation		Absolute	Degrees
224	Tail-To-Head Orientation		Positive	Degrees
225	Tail-To-Head Orientation		Negative	Degrees
226	Tail-To-Head Orientation	Forward		Degrees
227	Tail-To-Head Orientation	Forward	Absolute	Degrees
228	Tail-To-Head Orientation	Forward	Positive	Degrees
229	Tail-To-Head Orientation	Forward	Negative	Degrees
230	Tail-To-Head Orientation	Paused		Degrees
231	Tail-To-Head Orientation	Paused	Absolute	Degrees
232	Tail-To-Head Orientation	Paused	Positive	Degrees
233	Tail-To-Head Orientation	Paused	Negative	Degrees
234	Tail-To-Head Orientation	Backward		Degrees
235	Tail-To-Head Orientation	Backward	Absolute	Degrees
236	Tail-To-Head Orientation	Backward	Positive	Degrees
237	Tail-To-Head Orientation	Backward	Negative	Degrees
238	Head Orientation			Degrees
239	Head Orientation		Absolute	Degrees
240	Head Orientation		Positive	Degrees
241	Head Orientation		Negative	Degrees
242	Head Orientation	Forward		Degrees
243	Head Orientation	Forward	Absolute	Degrees
244	Head Orientation	Forward	Positive	Degrees
245	Head Orientation	Forward	Negative	Degrees
246	Head Orientation	Paused		Degrees
247	Head Orientation	Paused	Absolute	Degrees
248	Head Orientation	Paused	Positive	Degrees

249	Head Orientation	Paused	Negative	Degrees
250	Head Orientation	Backward		Degrees
251	Head Orientation	Backward	Absolute	Degrees
252	Head Orientation	Backward	Positive	Degrees
253	Head Orientation	Backward	Negative	Degrees
254	Tail Orientation			Degrees
255	Tail Orientation		Absolute	Degrees
256	Tail Orientation		Positive	Degrees
257	Tail Orientation		Negative	Degrees
258	Tail Orientation	Forward		Degrees
259	Tail Orientation	Forward	Absolute	Degrees
260	Tail Orientation	Forward	Positive	Degrees
261	Tail Orientation	Forward	Negative	Degrees
262	Tail Orientation	Paused		Degrees
263	Tail Orientation	Paused	Absolute	Degrees
264	Tail Orientation	Paused	Positive	Degrees
265	Tail Orientation	Paused	Negative	Degrees
266	Tail Orientation	Backward		Degrees
267	Tail Orientation	Backward	Absolute	Degrees
268	Tail Orientation	Backward	Positive	Degrees
269	Tail Orientation	Backward	Negative	Degrees
270	Eigen Projection 1			No Units
271	Eigen Projection 1		Absolute	No Units
272	Eigen Projection 1		Positive	No Units
273	Eigen Projection 1		Negative	No Units
274	Eigen Projection 1	Forward		No Units
275	Eigen Projection 1	Forward	Absolute	No Units
276	Eigen Projection 1	Forward	Positive	No Units
277	Eigen Projection 1	Forward	Negative	No Units
278	Eigen Projection 1	Paused		No Units
279	Eigen Projection 1	Paused	Absolute	No Units
280	Eigen Projection 1	Paused	Positive	No Units
281	Eigen Projection 1	Paused	Negative	No Units
282	Eigen Projection 1	Backward		No Units
283	Eigen Projection 1	Backward	Absolute	No Units
284	Eigen Projection 1	Backward	Positive	No Units
285	Eigen Projection 1	Backward	Negative	No Units
286	Eigen Projection 2			No Units
287	Eigen Projection 2		Absolute	No Units
288	Eigen Projection 2		Positive	No Units
289	Eigen Projection 2		Negative	No Units
290	Eigen Projection 2	Forward		No Units
291	Eigen Projection 2	Forward	Absolute	No Units
292	Eigen Projection 2	Forward	Positive	No Units
293	Eigen Projection 2	Forward	Negative	No Units
294	Eigen Projection 2	Paused		No Units
295	Eigen Projection 2	Paused	Absolute	No Units
296	Eigen Projection 2	Paused	Positive	No Units
297	Eigen Projection 2	Paused	Negative	No Units
298	Eigen Projection 2	Backward		No Units
299	Eigen Projection 2	Backward	Absolute	No Units
300	Eigen Projection 2	Backward	Positive	No Units
301	Eigen Projection 2	Backward	Negative	No Units
302	Eigen Projection 3			No Units
303	Eigen Projection 3		Absolute	No Units
304	Eigen Projection 3		Positive	No Units
305	Eigen Projection 3		Negative	No Units
306	Eigen Projection 3	Forward		No Units
307	Eigen Projection 3	Forward	Absolute	No Units
308	Eigen Projection 3	Forward	Positive	No Units
309	Eigen Projection 3	Forward	Negative	No Units
310	Eigen Projection 3	Paused		No Units
311	Eigen Projection 3	Paused	Absolute	No Units
312	Eigen Projection 3	Paused	Positive	No Units
313	Eigen Projection 3	Paused	Negative	No Units
314	Eigen Projection 3	Backward		No Units

315	Eigen Projection 3	Backward	Absolute	No Units
316	Eigen Projection 3	Backward	Positive	No Units
317	Eigen Projection 3	Backward	Negative	No Units
318	Eigen Projection 4			No Units
319	Eigen Projection 4		Absolute	No Units
320	Eigen Projection 4		Positive	No Units
321	Eigen Projection 4		Negative	No Units
322	Eigen Projection 4	Forward		No Units
323	Eigen Projection 4	Forward	Absolute	No Units
324	Eigen Projection 4	Forward	Positive	No Units
325	Eigen Projection 4	Forward	Negative	No Units
326	Eigen Projection 4	Paused		No Units
327	Eigen Projection 4	Paused	Absolute	No Units
328	Eigen Projection 4	Paused	Positive	No Units
329	Eigen Projection 4	Paused	Negative	No Units
330	Eigen Projection 4	Backward		No Units
331	Eigen Projection 4	Backward	Absolute	No Units
332	Eigen Projection 4	Backward	Positive	No Units
333	Eigen Projection 4	Backward	Negative	No Units
334	Eigen Projection 5			No Units
335	Eigen Projection 5		Absolute	No Units
336	Eigen Projection 5		Positive	No Units
337	Eigen Projection 5		Negative	No Units
338	Eigen Projection 5	Forward		No Units
339	Eigen Projection 5	Forward	Absolute	No Units
340	Eigen Projection 5	Forward	Positive	No Units
341	Eigen Projection 5	Forward	Negative	No Units
342	Eigen Projection 5	Paused		No Units
343	Eigen Projection 5	Paused	Absolute	No Units
344	Eigen Projection 5	Paused	Positive	No Units
345	Eigen Projection 5	Paused	Negative	No Units
346	Eigen Projection 5	Backward		No Units
347	Eigen Projection 5	Backward	Absolute	No Units
348	Eigen Projection 5	Backward	Positive	No Units
349	Eigen Projection 5	Backward	Negative	No Units
350	Eigen Projection 6			No Units
351	Eigen Projection 6		Absolute	No Units
352	Eigen Projection 6		Positive	No Units
353	Eigen Projection 6		Negative	No Units
354	Eigen Projection 6	Forward		No Units
355	Eigen Projection 6	Forward	Absolute	No Units
356	Eigen Projection 6	Forward	Positive	No Units
357	Eigen Projection 6	Forward	Negative	No Units
358	Eigen Projection 6	Paused		No Units
359	Eigen Projection 6	Paused	Absolute	No Units
360	Eigen Projection 6	Paused	Positive	No Units
361	Eigen Projection 6	Paused	Negative	No Units
362	Eigen Projection 6	Backward		No Units
363	Eigen Projection 6	Backward	Absolute	No Units
364	Eigen Projection 6	Backward	Positive	No Units
365	Eigen Projection 6	Backward	Negative	No Units
366	Forward Motion	Frequency		Hz
367	Forward Motion	Time Ratio		No Units
368	Forward Motion	Distance Ratio		No Units
369	Forward Motion	Time		Seconds
370	Forward Motion	Distance		Microns
371	Forward Motion	Inter Time		Seconds
372	Forward Motion	Inter Distance		Microns
373	Paused Motion	Frequency		Hz
374	Paused Motion	Time Ratio		No Units
375	Paused Motion	Distance Ratio		No Units
376	Paused Motion	Time		Seconds
377	Paused Motion	Distance		Microns
378	Paused Motion	Inter Time		Seconds
379	Paused Motion	Inter Distance		Microns
380	Backward Motion	Frequency		Hz

381	Backward Motion	Time Ratio		No Units
382	Backward Motion	Distance Ratio		No Units
383	Backward Motion	Time		Seconds
384	Backward Motion	Distance		Microns
385	Backward Motion	Inter Time		Seconds
386	Backward Motion	Inter Distance		Microns
387	Head Tip Speed (+/- = Forward/Backward)			Microns/Seconds
388	Head Tip Speed (+/- = Forward/Backward)		Absolute	Microns/Seconds
389	Head Tip Speed (+/- = Forward/Backward)		Positive	Microns/Seconds
390	Head Tip Speed (+/- = Forward/Backward)		Negative	Microns/Seconds
391	Head Tip Speed (+/- = Forward/Backward)	Forward		Microns/Seconds
392	Head Tip Speed (+/- = Forward/Backward)	Forward	Absolute	Microns/Seconds
393	Head Tip Speed (+/- = Forward/Backward)	Forward	Positive	Microns/Seconds
394	Head Tip Speed (+/- = Forward/Backward)	Forward	Negative	Microns/Seconds
395	Head Tip Speed (+/- = Forward/Backward)	Paused		Microns/Seconds
396	Head Tip Speed (+/- = Forward/Backward)	Paused	Absolute	Microns/Seconds
397	Head Tip Speed (+/- = Forward/Backward)	Paused	Positive	Microns/Seconds
398	Head Tip Speed (+/- = Forward/Backward)	Paused	Negative	Microns/Seconds
399	Head Tip Speed (+/- = Forward/Backward)	Backward		Microns/Seconds
400	Head Tip Speed (+/- = Forward/Backward)	Backward	Absolute	Microns/Seconds
401	Head Tip Speed (+/- = Forward/Backward)	Backward	Positive	Microns/Seconds
402	Head Tip Speed (+/- = Forward/Backward)	Backward	Negative	Microns/Seconds
403	Head Speed (+/- = Forward/Backward)			Microns/Seconds
404	Head Speed (+/- = Forward/Backward)		Absolute	Microns/Seconds
405	Head Speed (+/- = Forward/Backward)		Positive	Microns/Seconds
406	Head Speed (+/- = Forward/Backward)		Negative	Microns/Seconds
407	Head Speed (+/- = Forward/Backward)	Forward		Microns/Seconds
408	Head Speed (+/- = Forward/Backward)	Forward	Absolute	Microns/Seconds
409	Head Speed (+/- = Forward/Backward)	Forward	Positive	Microns/Seconds
410	Head Speed (+/- = Forward/Backward)	Forward	Negative	Microns/Seconds
411	Head Speed (+/- = Forward/Backward)	Paused		Microns/Seconds
412	Head Speed (+/- = Forward/Backward)	Paused	Absolute	Microns/Seconds
413	Head Speed (+/- = Forward/Backward)	Paused	Positive	Microns/Seconds
414	Head Speed (+/- = Forward/Backward)	Paused	Negative	Microns/Seconds
415	Head Speed (+/- = Forward/Backward)	Backward		Microns/Seconds
416	Head Speed (+/- = Forward/Backward)	Backward	Absolute	Microns/Seconds
417	Head Speed (+/- = Forward/Backward)	Backward	Positive	Microns/Seconds
418	Head Speed (+/- = Forward/Backward)	Backward	Negative	Microns/Seconds
419	Midbody Speed (+/- = Forward/Backward)			Microns/Seconds
420	Midbody Speed (+/- = Forward/Backward)		Absolute	Microns/Seconds
421	Midbody Speed (+/- = Forward/Backward)		Positive	Microns/Seconds
422	Midbody Speed (+/- = Forward/Backward)		Negative	Microns/Seconds
423	Midbody Speed (+/- = Forward/Backward)	Forward		Microns/Seconds
424	Midbody Speed (+/- = Forward/Backward)	Forward	Absolute	Microns/Seconds
425	Midbody Speed (+/- = Forward/Backward)	Forward	Positive	Microns/Seconds
426	Midbody Speed (+/- = Forward/Backward)	Forward	Negative	Microns/Seconds
427	Midbody Speed (+/- = Forward/Backward)	Paused		Microns/Seconds
428	Midbody Speed (+/- = Forward/Backward)	Paused	Absolute	Microns/Seconds
429	Midbody Speed (+/- = Forward/Backward)	Paused	Positive	Microns/Seconds
430	Midbody Speed (+/- = Forward/Backward)	Paused	Negative	Microns/Seconds
431	Midbody Speed (+/- = Forward/Backward)	Backward		Microns/Seconds
432	Midbody Speed (+/- = Forward/Backward)	Backward	Absolute	Microns/Seconds
433	Midbody Speed (+/- = Forward/Backward)	Backward	Positive	Microns/Seconds
434	Midbody Speed (+/- = Forward/Backward)	Backward	Negative	Microns/Seconds
435	Tail Speed (+/- = Forward/Backward)			Microns/Seconds
436	Tail Speed (+/- = Forward/Backward)		Absolute	Microns/Seconds
437	Tail Speed (+/- = Forward/Backward)		Positive	Microns/Seconds
438	Tail Speed (+/- = Forward/Backward)		Negative	Microns/Seconds
439	Tail Speed (+/- = Forward/Backward)	Forward		Microns/Seconds
440	Tail Speed (+/- = Forward/Backward)	Forward	Absolute	Microns/Seconds
441	Tail Speed (+/- = Forward/Backward)	Forward	Positive	Microns/Seconds
442	Tail Speed (+/- = Forward/Backward)	Forward	Negative	Microns/Seconds
443	Tail Speed (+/- = Forward/Backward)	Paused		Microns/Seconds
444	Tail Speed (+/- = Forward/Backward)	Paused	Absolute	Microns/Seconds
445	Tail Speed (+/- = Forward/Backward)	Paused	Positive	Microns/Seconds
446	Tail Speed (+/- = Forward/Backward)	Paused	Negative	Microns/Seconds

447	Tail Speed (+/- = Forward/Backward)	Backward		Microns/Seconds
448	Tail Speed (+/- = Forward/Backward)	Backward	Absolute	Microns/Seconds
449	Tail Speed (+/- = Forward/Backward)	Backward	Positive	Microns/Seconds
450	Tail Speed (+/- = Forward/Backward)	Backward	Negative	Microns/Seconds
451	Tail Tip Speed (+/- = Forward/Backward)			Microns/Seconds
452	Tail Tip Speed (+/- = Forward/Backward)		Absolute	Microns/Seconds
453	Tail Tip Speed (+/- = Forward/Backward)		Positive	Microns/Seconds
454	Tail Tip Speed (+/- = Forward/Backward)		Negative	Microns/Seconds
455	Tail Tip Speed (+/- = Forward/Backward)	Forward		Microns/Seconds
456	Tail Tip Speed (+/- = Forward/Backward)	Forward	Absolute	Microns/Seconds
457	Tail Tip Speed (+/- = Forward/Backward)	Forward	Positive	Microns/Seconds
458	Tail Tip Speed (+/- = Forward/Backward)	Forward	Negative	Microns/Seconds
459	Tail Tip Speed (+/- = Forward/Backward)	Paused		Microns/Seconds
460	Tail Tip Speed (+/- = Forward/Backward)	Paused	Absolute	Microns/Seconds
461	Tail Tip Speed (+/- = Forward/Backward)	Paused	Positive	Microns/Seconds
462	Tail Tip Speed (+/- = Forward/Backward)	Paused	Negative	Microns/Seconds
463	Tail Tip Speed (+/- = Forward/Backward)	Backward		Microns/Seconds
464	Tail Tip Speed (+/- = Forward/Backward)	Backward	Absolute	Microns/Seconds
465	Tail Tip Speed (+/- = Forward/Backward)	Backward	Positive	Microns/Seconds
466	Tail Tip Speed (+/- = Forward/Backward)	Backward	Negative	Microns/Seconds
467	Head Tip Motion Direction (+/- = Toward D/V)			Degrees/Seconds
468	Head Tip Motion Direction (+/- = Toward D/V)		Absolute	Degrees/Seconds
469	Head Tip Motion Direction (+/- = Toward D/V)		Positive	Degrees/Seconds
470	Head Tip Motion Direction (+/- = Toward D/V)		Negative	Degrees/Seconds
471	Head Tip Motion Direction (+/- = Toward D/V)	Forward		Degrees/Seconds
472	Head Tip Motion Direction (+/- = Toward D/V)	Forward	Absolute	Degrees/Seconds
473	Head Tip Motion Direction (+/- = Toward D/V)	Forward	Positive	Degrees/Seconds
474	Head Tip Motion Direction (+/- = Toward D/V)	Forward	Negative	Degrees/Seconds
475	Head Tip Motion Direction (+/- = Toward D/V)	Paused		Degrees/Seconds
476	Head Tip Motion Direction (+/- = Toward D/V)	Paused	Absolute	Degrees/Seconds
477	Head Tip Motion Direction (+/- = Toward D/V)	Paused	Positive	Degrees/Seconds
478	Head Tip Motion Direction (+/- = Toward D/V)	Paused	Negative	Degrees/Seconds
479	Head Tip Motion Direction (+/- = Toward D/V)	Backward		Degrees/Seconds
480	Head Tip Motion Direction (+/- = Toward D/V)	Backward	Absolute	Degrees/Seconds
481	Head Tip Motion Direction (+/- = Toward D/V)	Backward	Positive	Degrees/Seconds
482	Head Tip Motion Direction (+/- = Toward D/V)	Backward	Negative	Degrees/Seconds
483	Head Motion Direction (+/- = Toward D/V)			Degrees/Seconds
484	Head Motion Direction (+/- = Toward D/V)		Absolute	Degrees/Seconds
485	Head Motion Direction (+/- = Toward D/V)		Positive	Degrees/Seconds
486	Head Motion Direction (+/- = Toward D/V)		Negative	Degrees/Seconds
487	Head Motion Direction (+/- = Toward D/V)	Forward		Degrees/Seconds
488	Head Motion Direction (+/- = Toward D/V)	Forward	Absolute	Degrees/Seconds
489	Head Motion Direction (+/- = Toward D/V)	Forward	Positive	Degrees/Seconds
490	Head Motion Direction (+/- = Toward D/V)	Forward	Negative	Degrees/Seconds
491	Head Motion Direction (+/- = Toward D/V)	Paused		Degrees/Seconds
492	Head Motion Direction (+/- = Toward D/V)	Paused	Absolute	Degrees/Seconds
493	Head Motion Direction (+/- = Toward D/V)	Paused	Positive	Degrees/Seconds
494	Head Motion Direction (+/- = Toward D/V)	Paused	Negative	Degrees/Seconds
495	Head Motion Direction (+/- = Toward D/V)	Backward		Degrees/Seconds
496	Head Motion Direction (+/- = Toward D/V)	Backward	Absolute	Degrees/Seconds
497	Head Motion Direction (+/- = Toward D/V)	Backward	Positive	Degrees/Seconds
498	Head Motion Direction (+/- = Toward D/V)	Backward	Negative	Degrees/Seconds
499	Midbody Motion Direction (+/- = Toward D/V)			Degrees/Seconds
500	Midbody Motion Direction (+/- = Toward D/V)		Absolute	Degrees/Seconds
501	Midbody Motion Direction (+/- = Toward D/V)		Positive	Degrees/Seconds
502	Midbody Motion Direction (+/- = Toward D/V)		Negative	Degrees/Seconds
503	Midbody Motion Direction (+/- = Toward D/V)	Forward		Degrees/Seconds
504	Midbody Motion Direction (+/- = Toward D/V)	Forward	Absolute	Degrees/Seconds
505	Midbody Motion Direction (+/- = Toward D/V)	Forward	Positive	Degrees/Seconds
506	Midbody Motion Direction (+/- = Toward D/V)	Forward	Negative	Degrees/Seconds
507	Midbody Motion Direction (+/- = Toward D/V)	Paused		Degrees/Seconds
508	Midbody Motion Direction (+/- = Toward D/V)	Paused	Absolute	Degrees/Seconds
509	Midbody Motion Direction (+/- = Toward D/V)	Paused	Positive	Degrees/Seconds
510	Midbody Motion Direction (+/- = Toward D/V)	Paused	Negative	Degrees/Seconds
511	Midbody Motion Direction (+/- = Toward D/V)	Backward		Degrees/Seconds
512	Midbody Motion Direction (+/- = Toward D/V)	Backward	Absolute	Degrees/Seconds

513	Midbody Motion Direction (+/- = Toward D/V)	Backward	Positive	Degrees/Seconds
514	Midbody Motion Direction (+/- = Toward D/V)	Backward	Negative	Degrees/Seconds
515	Tail Motion Direction (+/- = Toward D/V)			Degrees/Seconds
516	Tail Motion Direction (+/- = Toward D/V)		Absolute	Degrees/Seconds
517	Tail Motion Direction (+/- = Toward D/V)		Positive	Degrees/Seconds
518	Tail Motion Direction (+/- = Toward D/V)		Negative	Degrees/Seconds
519	Tail Motion Direction (+/- = Toward D/V)	Forward		Degrees/Seconds
520	Tail Motion Direction (+/- = Toward D/V)	Forward	Absolute	Degrees/Seconds
521	Tail Motion Direction (+/- = Toward D/V)	Forward	Positive	Degrees/Seconds
522	Tail Motion Direction (+/- = Toward D/V)	Forward	Negative	Degrees/Seconds
523	Tail Motion Direction (+/- = Toward D/V)	Paused		Degrees/Seconds
524	Tail Motion Direction (+/- = Toward D/V)	Paused	Absolute	Degrees/Seconds
525	Tail Motion Direction (+/- = Toward D/V)	Paused	Positive	Degrees/Seconds
526	Tail Motion Direction (+/- = Toward D/V)	Paused	Negative	Degrees/Seconds
527	Tail Motion Direction (+/- = Toward D/V)	Backward		Degrees/Seconds
528	Tail Motion Direction (+/- = Toward D/V)	Backward	Absolute	Degrees/Seconds
529	Tail Motion Direction (+/- = Toward D/V)	Backward	Positive	Degrees/Seconds
530	Tail Motion Direction (+/- = Toward D/V)	Backward	Negative	Degrees/Seconds
531	Tail Tip Motion Direction (+/- = Toward D/V)			Degrees/Seconds
532	Tail Tip Motion Direction (+/- = Toward D/V)		Absolute	Degrees/Seconds
533	Tail Tip Motion Direction (+/- = Toward D/V)		Positive	Degrees/Seconds
534	Tail Tip Motion Direction (+/- = Toward D/V)		Negative	Degrees/Seconds
535	Tail Tip Motion Direction (+/- = Toward D/V)	Forward		Degrees/Seconds
536	Tail Tip Motion Direction (+/- = Toward D/V)	Forward	Absolute	Degrees/Seconds
537	Tail Tip Motion Direction (+/- = Toward D/V)	Forward	Positive	Degrees/Seconds
538	Tail Tip Motion Direction (+/- = Toward D/V)	Forward	Negative	Degrees/Seconds
539	Tail Tip Motion Direction (+/- = Toward D/V)	Paused		Degrees/Seconds
540	Tail Tip Motion Direction (+/- = Toward D/V)	Paused	Absolute	Degrees/Seconds
541	Tail Tip Motion Direction (+/- = Toward D/V)	Paused	Positive	Degrees/Seconds
542	Tail Tip Motion Direction (+/- = Toward D/V)	Paused	Negative	Degrees/Seconds
543	Tail Tip Motion Direction (+/- = Toward D/V)	Backward		Degrees/Seconds
544	Tail Tip Motion Direction (+/- = Toward D/V)	Backward	Absolute	Degrees/Seconds
545	Tail Tip Motion Direction (+/- = Toward D/V)	Backward	Positive	Degrees/Seconds
546	Tail Tip Motion Direction (+/- = Toward D/V)	Backward	Negative	Degrees/Seconds
547	Foraging Amplitude (+/- = Toward D/V)			Microns
548	Foraging Amplitude (+/- = Toward D/V)		Absolute	Microns
549	Foraging Amplitude (+/- = Toward D/V)		Positive	Microns
550	Foraging Amplitude (+/- = Toward D/V)		Negative	Microns
551	Foraging Amplitude (+/- = Toward D/V)	Forward		Microns
552	Foraging Amplitude (+/- = Toward D/V)	Forward	Absolute	Microns
553	Foraging Amplitude (+/- = Toward D/V)	Forward	Positive	Microns
554	Foraging Amplitude (+/- = Toward D/V)	Forward	Negative	Microns
555	Foraging Amplitude (+/- = Toward D/V)	Paused		Microns
556	Foraging Amplitude (+/- = Toward D/V)	Paused	Absolute	Microns
557	Foraging Amplitude (+/- = Toward D/V)	Paused	Positive	Microns
558	Foraging Amplitude (+/- = Toward D/V)	Paused	Negative	Microns
559	Foraging Amplitude (+/- = Toward D/V)	Backward		Microns
560	Foraging Amplitude (+/- = Toward D/V)	Backward	Absolute	Microns
561	Foraging Amplitude (+/- = Toward D/V)	Backward	Positive	Microns
562	Foraging Amplitude (+/- = Toward D/V)	Backward	Negative	Microns
563	Head Crawling Amplitude (+/- = D/V Inside)			Degrees
564	Head Crawling Amplitude (+/- = D/V Inside)		Absolute	Degrees
565	Head Crawling Amplitude (+/- = D/V Inside)		Positive	Degrees
566	Head Crawling Amplitude (+/- = D/V Inside)		Negative	Degrees
567	Head Crawling Amplitude (+/- = D/V Inside)	Forward		Degrees
568	Head Crawling Amplitude (+/- = D/V Inside)	Forward	Absolute	Degrees
569	Head Crawling Amplitude (+/- = D/V Inside)	Forward	Positive	Degrees
570	Head Crawling Amplitude (+/- = D/V Inside)	Forward	Negative	Degrees
571	Head Crawling Amplitude (+/- = D/V Inside)	Backward		Degrees
572	Head Crawling Amplitude (+/- = D/V Inside)	Backward	Absolute	Degrees
573	Head Crawling Amplitude (+/- = D/V Inside)	Backward	Positive	Degrees
574	Head Crawling Amplitude (+/- = D/V Inside)	Backward	Negative	Degrees
575	Midbody Crawling Amplitude (+/- = D/V Inside)			Degrees
576	Midbody Crawling Amplitude (+/- = D/V Inside)		Absolute	Degrees
577	Midbody Crawling Amplitude (+/- = D/V Inside)		Positive	Degrees
578	Midbody Crawling Amplitude (+/- = D/V Inside)		Negative	Degrees

579	Midbody Crawling Amplitude (+/- = D/V Inside)	Forward		Degrees
580	Midbody Crawling Amplitude (+/- = D/V Inside)	Forward	Absolute	Degrees
581	Midbody Crawling Amplitude (+/- = D/V Inside)	Forward	Positive	Degrees
582	Midbody Crawling Amplitude (+/- = D/V Inside)	Forward	Negative	Degrees
583	Midbody Crawling Amplitude (+/- = D/V Inside)	Backward		Degrees
584	Midbody Crawling Amplitude (+/- = D/V Inside)	Backward	Absolute	Degrees
585	Midbody Crawling Amplitude (+/- = D/V Inside)	Backward	Positive	Degrees
586	Midbody Crawling Amplitude (+/- = D/V Inside)	Backward	Negative	Degrees
587	Tail Crawling Amplitude (+/- = D/V Inside)			Degrees
588	Tail Crawling Amplitude (+/- = D/V Inside)		Absolute	Degrees
589	Tail Crawling Amplitude (+/- = D/V Inside)		Positive	Degrees
590	Tail Crawling Amplitude (+/- = D/V Inside)		Negative	Degrees
591	Tail Crawling Amplitude (+/- = D/V Inside)	Forward		Degrees
592	Tail Crawling Amplitude (+/- = D/V Inside)	Forward	Absolute	Degrees
593	Tail Crawling Amplitude (+/- = D/V Inside)	Forward	Positive	Degrees
594	Tail Crawling Amplitude (+/- = D/V Inside)	Forward	Negative	Degrees
595	Tail Crawling Amplitude (+/- = D/V Inside)	Backward		Degrees
596	Tail Crawling Amplitude (+/- = D/V Inside)	Backward	Absolute	Degrees
597	Tail Crawling Amplitude (+/- = D/V Inside)	Backward	Positive	Degrees
598	Tail Crawling Amplitude (+/- = D/V Inside)	Backward	Negative	Degrees
599	Foraging Speed			Degrees/Seconds
600	Foraging Speed		Absolute	Degrees/Seconds
601	Foraging Speed		Positive	Degrees/Seconds
602	Foraging Speed		Negative	Degrees/Seconds
603	Foraging Speed	Forward		Degrees/Seconds
604	Foraging Speed	Forward	Absolute	Degrees/Seconds
605	Foraging Speed	Forward	Positive	Degrees/Seconds
606	Foraging Speed	Forward	Negative	Degrees/Seconds
607	Foraging Speed	Paused		Degrees/Seconds
608	Foraging Speed	Paused	Absolute	Degrees/Seconds
609	Foraging Speed	Paused	Positive	Degrees/Seconds
610	Foraging Speed	Paused	Negative	Degrees/Seconds
611	Foraging Speed	Backward		Degrees/Seconds
612	Foraging Speed	Backward	Absolute	Degrees/Seconds
613	Foraging Speed	Backward	Positive	Degrees/Seconds
614	Foraging Speed	Backward	Negative	Degrees/Seconds
615	Head Crawling Frequency (+/- = D/V Inside)			Hz
616	Head Crawling Frequency (+/- = D/V Inside)		Absolute	Hz
617	Head Crawling Frequency (+/- = D/V Inside)		Positive	Hz
618	Head Crawling Frequency (+/- = D/V Inside)		Negative	Hz
619	Head Crawling Frequency (+/- = D/V Inside)	Forward		Hz
620	Head Crawling Frequency (+/- = D/V Inside)	Forward	Absolute	Hz
621	Head Crawling Frequency (+/- = D/V Inside)	Forward	Positive	Hz
622	Head Crawling Frequency (+/- = D/V Inside)	Forward	Negative	Hz
623	Head Crawling Frequency (+/- = D/V Inside)	Backward		Hz
624	Head Crawling Frequency (+/- = D/V Inside)	Backward	Absolute	Hz
625	Head Crawling Frequency (+/- = D/V Inside)	Backward	Positive	Hz
626	Head Crawling Frequency (+/- = D/V Inside)	Backward	Negative	Hz
627	Midbody Crawling Frequency (+/- = D/V Inside)			Hz
628	Midbody Crawling Frequency (+/- = D/V Inside)		Absolute	Hz
629	Midbody Crawling Frequency (+/- = D/V Inside)		Positive	Hz
630	Midbody Crawling Frequency (+/- = D/V Inside)		Negative	Hz
631	Midbody Crawling Frequency (+/- = D/V Inside)	Forward		Hz
632	Midbody Crawling Frequency (+/- = D/V Inside)	Forward	Absolute	Hz
633	Midbody Crawling Frequency (+/- = D/V Inside)	Forward	Positive	Hz
634	Midbody Crawling Frequency (+/- = D/V Inside)	Forward	Negative	Hz
635	Midbody Crawling Frequency (+/- = D/V Inside)	Backward		Hz
636	Midbody Crawling Frequency (+/- = D/V Inside)	Backward	Absolute	Hz
637	Midbody Crawling Frequency (+/- = D/V Inside)	Backward	Positive	Hz
638	Midbody Crawling Frequency (+/- = D/V Inside)	Backward	Negative	Hz
639	Tail Crawling Frequency (+/- = D/V Inside)			Hz
640	Tail Crawling Frequency (+/- = D/V Inside)		Absolute	Hz
641	Tail Crawling Frequency (+/- = D/V Inside)		Positive	Hz
642	Tail Crawling Frequency (+/- = D/V Inside)		Negative	Hz
643	Tail Crawling Frequency (+/- = D/V Inside)	Forward		Hz
644	Tail Crawling Frequency (+/- = D/V Inside)	Forward	Absolute	Hz

645	Tail Crawling Frequency (+/- = D/V Inside)	Forward	Positive	Hz
646	Tail Crawling Frequency (+/- = D/V Inside)	Forward	Negative	Hz
647	Tail Crawling Frequency (+/- = D/V Inside)	Backward		Hz
648	Tail Crawling Frequency (+/- = D/V Inside)	Backward	Absolute	Hz
649	Tail Crawling Frequency (+/- = D/V Inside)	Backward	Positive	Hz
650	Tail Crawling Frequency (+/- = D/V Inside)	Backward	Negative	Hz
651	Omega Turn Events	Frequency		Hz
652	Omega Turn Events	Time Ratio		No Units
653	Omega Turn Events	Time		Seconds
654	Omega Turn Events	Time	Absolute	Seconds
655	Omega Turn Events	Time	Positive	Seconds
656	Omega Turn Events	Time	Negative	Seconds
657	Omega Turn Events	Inter Time		Seconds
658	Omega Turn Events	Inter Time	Absolute	Seconds
659	Omega Turn Events	Inter Time	Positive	Seconds
660	Omega Turn Events	Inter Time	Negative	Seconds
661	Omega Turn Events	Inter Distance		Microns
662	Omega Turn Events	Inter Distance	Absolute	Microns
663	Omega Turn Events	Inter Distance	Positive	Microns
664	Omega Turn Events	Inter Distance	Negative	Microns
665	Upsilon Turn Events	Frequency		Hz
666	Upsilon Turn Events	Time Ratio		No Units
667	Upsilon Turn Events	Time		Seconds
668	Upsilon Turn Events	Time	Absolute	Seconds
669	Upsilon Turn Events	Time	Positive	Seconds
670	Upsilon Turn Events	Time	Negative	Seconds
671	Upsilon Turn Events	Inter Time		Seconds
672	Upsilon Turn Events	Inter Time	Absolute	Seconds
673	Upsilon Turn Events	Inter Time	Positive	Seconds
674	Upsilon Turn Events	Inter Time	Negative	Seconds
675	Upsilon Turn Events	Inter Distance		Microns
676	Upsilon Turn Events	Inter Distance	Absolute	Microns
677	Upsilon Turn Events	Inter Distance	Positive	Microns
678	Upsilon Turn Events	Inter Distance	Negative	Microns
679	Path Range			Microns
680	Path Range	Forward		Microns
681	Path Range	Paused		Microns
682	Path Range	Backward		Microns
683	Worm Dwelling			Seconds
684	Head Dwelling			Seconds
685	Midbody Dwelling			Seconds
686	Tail Dwelling			Seconds
687	Path Curvature (+/- = D/V Inside)			Radians/Microns
688	Path Curvature (+/- = D/V Inside)		Absolute	Radians/Microns
689	Path Curvature (+/- = D/V Inside)		Positive	Radians/Microns
690	Path Curvature (+/- = D/V Inside)		Negative	Radians/Microns
691	Path Curvature (+/- = D/V Inside)	Forward		Radians/Microns
692	Path Curvature (+/- = D/V Inside)	Forward	Absolute	Radians/Microns
693	Path Curvature (+/- = D/V Inside)	Forward	Positive	Radians/Microns
694	Path Curvature (+/- = D/V Inside)	Forward	Negative	Radians/Microns
695	Path Curvature (+/- = D/V Inside)	Paused		Radians/Microns
696	Path Curvature (+/- = D/V Inside)	Paused	Absolute	Radians/Microns
697	Path Curvature (+/- = D/V Inside)	Paused	Positive	Radians/Microns
698	Path Curvature (+/- = D/V Inside)	Paused	Negative	Radians/Microns
699	Path Curvature (+/- = D/V Inside)	Backward		Radians/Microns
700	Path Curvature (+/- = D/V Inside)	Backward	Absolute	Radians/Microns
701	Path Curvature (+/- = D/V Inside)	Backward	Positive	Radians/Microns
702	Path Curvature (+/- = D/V Inside)	Backward	Negative	Radians/Microns

Appendix C : Strain and Group Significance

The list of strain and group significance using two types of statistical testing. Each strain is listed alongside its genotype. The q-value is shown, per strain and group, for the minimum of the Wilcoxon rank-sum feature measurement tests as well as the bootstrapped MANOVAs using shrinkage covariance.

Genotype	Strain	Minimum Rank-Sum q	Bootstrap MANOVA q
acc-4(ok2371)III	RB1832	6.66E-09	1.23E-06
acd-2(ok1237)I	RB1192	1.14E-05	1.23E-06
acd-5(ok2657)I	RB2005	8.41E-09	1.23E-06
acr-2(ok1887)X	RB1559	7.71E-05	1.23E-06
acr-3(ok2049)X	RB1659	7.15E-05	1.23E-06
acr-6(ok3117)I	RB2294	5.16E-06	1.23E-06
acr-7(tm863)II	FX863	3.38E-06	1.23E-06
acr-9(ok933)X	VC649	9.21E-09	1.23E-06
acr-10(ok3064)X	RB2262	9.02E-06	3.64E-06
acr-11(ok1345)I	RB1263	3.29E-05	1.23E-06
acr-14(ok1155)II	RB1132	1.57E-04	1.23E-06
acr-15(ok1214)V	RB1172	4.03E-06	1.23E-06
acr-18(ok1285)V	RB1226	8.23E-08	1.23E-06
acr-19(ad1674)I	DA1674	6.11E-06	1.23E-06
acr-21(ok1314)III	RB1250	1.58E-05	1.23E-06
acr-23(ok2804)V	RB2119	8.01E-08	1.23E-06
asic-1(ok415)I	RB680	1.92E-08	1.23E-06
asic-2(ok289)I	RB557	1.01E-06	1.23E-06
bas-1(ad446)III	MT7988	4.95E-08	1.23E-06
C11D2.2(ok1565)IV	RB1380	1.11E-06	2.68E-05
C24G7.1(ok1822)I	RB1523	3.10E-05	1.23E-06
C38D9.2(ok1853)V	RB1543	8.83E-10	1.23E-06
cat-2(e1112)II	CB1112	4.73E-31	1.23E-06
cat-4(e1141)V	CB1141	5.66E-10	1.23E-06
daf-3(e1376)X	CB1376	7.60E-06	1.23E-06
daf-5(e1386)II	CB1386	1.64E-07	1.23E-06
daf-7(m62)III	DR62	1.62E-09	1.23E-06
dat-1(ok157)III	RM2702	2.69E-09	1.23E-06
del-1(ok150)X	NC279	9.71E-07	1.23E-06
del-4(ok1014)I	RB1064	3.30E-06	1.23E-06
del-7(ok1187)IV	RB1156	2.13E-05	1.23E-06
del-9(ok2353)X	RB1818	2.07E-07	1.23E-06
dnc-1(or404)IV	EU1006	5.21E-05	1.23E-06
dop-1(vs101)X	LX636	1.49E-03	1.23E-06
dop-1(vs100)dop-3(vs106)X	LX705	1.53E-05	5.96E-06
dop-2(vs105)V	LX702	6.73E-07	1.23E-06
dop-2(vs105)V; dop-1(vs100)X	LX706	4.77E-08	1.23E-06
dop-2(vs105)V; dop-3(vs106)X	LX704	1.84E-06	1.23E-06
dop-2(vs105)V; dop-1(vs100)dop-3(vs106)X	LX734	3.99E-07	1.23E-06
dop-3(vs106)X	LX703	1.95E-07	1.23E-06
dop-4(tm1392)X	FG58	5.29E-09	1.23E-06
dpy-20(e1282)IV	CB1282	2.95E-13	1.23E-06
eat-16(sa609)I	JT609	8.31E-09	1.23E-06
egas-2(ok1477)V	VC975	1.07E-06	1.23E-06
egas-3(ok1522)V	RB1356	1.33E-06	1.23E-06
egg-5(ok1781)I	VC1295	5.97E-10	1.23E-06
egl-1(n487)V	MT1082	1.73E-11	1.23E-06

egl-2(n693)V	MT1444	2.86E-06	1.64E-05
egl-5(n486)III	MT1081	6.36E-09	1.23E-06
egl-6(n592)X	MT1222	6.36E-09	1.23E-06
egl-7(n575)III	MT1205	6.37E-09	1.23E-06
egl-8(n488)V	MT1083	7.64E-16	1.23E-06
egl-9(n586)V	MT1216	3.07E-09	1.23E-06
egl-10(md176)V	MT8504	4.44E-19	1.23E-06
egl-11(n587)V	MT1217	1.43E-13	1.23E-06
egl-12(n602)V	MT1232	9.81E-11	1.23E-06
egl-13(n483)X	MT1078	1.76E-09	1.23E-06
egl-14(n549)X	MT1179	1.80E-12	1.23E-06
egl-15(n484)X	MT1079	3.23E-09	1.23E-06
egl-17(e1313)X	CB1313	7.51E-07	1.23E-06
egl-18(ok290)IV	JR2370	1.36E-07	1.23E-06
egl-19(n2368)IV	MT6129	1.28E-08	1.23E-06
egl-20(mu39)IV	CF263	7.71E-08	1.23E-06
egl-21(n476)IV	KP2018	3.79E-12	1.23E-06
egl-21(n611)IV	MT1241	2.11E-09	1.23E-06
egl-23(n601)IV	MT1231	9.11E-12	1.23E-06
egl-24(n572)III	MT1202	1.03E-06	8.31E-06
egl-27(ok151)II	KS99	1.29E-11	1.23E-06
egl-28(n570)II	MT1200	1.63E-11	1.23E-06
egl-30(ep271)I	CE1047	5.55E-10	1.23E-06
egl-30(n686)goa-1(n1134)I	AQ916	1.90E-10	1.23E-06
egl-31(n472)I	MT1067	8.74E-14	1.23E-06
egl-32(n155)I	MT155	5.85E-09	1.23E-06
egl-33(n151)I	MT151	1.22E-08	1.23E-06
egl-36(n728)X	MT1540	3.41E-10	1.23E-06
egl-37(n742)II	MT1543	4.19E-07	1.23E-06
egl-40(n606)IV	MT1236	8.33E-13	1.23E-06
egl-42(n995)II	MT2068	7.09E-05	4.81E-06
egl-44(n1080)II	MT2247	5.69E-11	1.23E-06
egl-46(n1127)V	MT2316	1.60E-09	1.23E-06
egl-47(n1081)V	MT2248	2.11E-19	1.23E-06
egl-49(n1107)X	MT2293	1.35E-08	1.23E-06
egl-50(n1086)II	AQ2316	2.75E-09	1.23E-06
ets-10(gk596)X	VC1340	2.06E-12	1.23E-06
F23B2.3(ok1226)IV	RB1177	8.48E-09	1.23E-06
F54E4.4(ok2336)X	RB1802	2.55E-06	1.23E-06
flp-1(yn2)IV	NY7	4.45E-06	1.23E-06
flp-3(ok3265)X	VC2497	8.84E-06	1.23E-06
flp-6(ok3056)V	VC2324	4.65E-07	1.23E-06
flp-7(ok2625)X	RB1990	4.91E-04	3.07E-04
flp-9(ok2730)IV	RB2067	3.73E-05	1.23E-06
flp-10(ok2624)IV	RB1989	1.39E-05	1.23E-06
flp-11(tm2706)X	FX2706	5.13E-05	1.23E-06
flp-12(ok2409)X	RB1863	1.54E-06	1.23E-06
flp-13(tm2427)IV	FX2427	1.51E-06	1.23E-06
flp-16(ok3085)II	RB2275	2.38E-07	1.23E-06
flp-17(ok3587)IV	RB2575	1.85E-08	1.23E-06
flp-18(db99)X	AX1410	5.49E-08	1.23E-06
flp-19(ok2460)X	RB1902	3.85E-08	1.23E-06
flp-20(ok2964)X	RB2188	1.69E-10	1.23E-06
flp-21(ok889)V	RB982	6.88E-05	1.23E-06
flp-25(gk1016)III	VC1982	4.83E-07	1.23E-06
flp-28(gk1075)X	VC2502	1.06E-04	1.53E-05
flp-33(gk1038)I	VC2423	5.41E-05	1.23E-06
flr-1(ut11)X	JC55	3.06E-10	1.23E-06
gar-2(ok520)III	RB756	3.71E-08	1.23E-06
gld-1(op236)I	TG34	2.55E-04	7.50E-05
gly-2(gk204)I	VC335	2.10E-07	1.23E-06
goa-1(sa734)I	DG1856	2.36E-19	1.23E-06
gon-2(q362)I	EJ26	4.42E-06	1.23E-06
gpa-1(pk15)V	NL332	7.66E-06	1.23E-06
gpa-2(pk16)V	NL334	1.45E-08	1.23E-06
gpa-3(pk35)V	NL335	5.64E-09	1.23E-06

gpa-4(pk381)IV	NL790	2.37E-08	1.23E-06
gpa-5(pk376)X	NL1137	3.49E-08	1.23E-06
gpa-6(pk480)X	NL1146	1.24E-08	1.23E-06
gpa-7(pk610)IV	NL795	1.63E-07	1.23E-06
gpa-8(pk345)V	NL1142	5.61E-06	1.23E-06
gpa-9(pk438)V	NL793	1.75E-06	1.23E-06
gpa-10(pk362)V	NL1147	1.89E-10	1.23E-06
gpa-11(pk349)II	NL787	1.32E-07	1.23E-06
gpa-12(pk322)X	NL594	2.96E-08	1.23E-06
gpa-13(pk1270)V	NL2330	2.36E-07	1.23E-06
gpa-14(pk347)I	AQ495	1.81E-08	1.23E-06
gpa-15(pk477)I	NL797	4.62E-08	1.23E-06
gpa-16(ok2349)I	RB1816	3.03E-10	1.23E-06
gpa-17(ok2334)III	RB1800	3.28E-05	1.23E-06
gpb-2(sa603)I	JT603	3.10E-07	1.23E-06
gpc-1(pk298)X	NL792	1.07E-06	1.23E-06
hcf-1(ok559)IV	RB777	3.63E-07	1.23E-06
ins-3(ok2488)II	RB1915	6.03E-08	1.23E-06
ins-4(ok3534)II	RB2544	7.05E-07	1.23E-06
ins-11(tm1053)II	FX1053	2.01E-03	4.80E-04
ins-15(ok3444)II	RB2489	4.75E-08	1.23E-06
ins-16(ok2919)III	RB2159	1.26E-06	1.23E-06
ins-18(ok1672)I	VC1218	1.09E-10	1.23E-06
ins-22(ok3616)III	RB2594	7.61E-07	1.23E-06
ins-25(ok2773)I	RB2098	8.61E-11	1.23E-06
ins-27(ok2474)I	RB1911	3.03E-06	1.23E-06
ins-28(ok2722)I	RB2059	1.00E-10	1.23E-06
ins-30(ok2343)I	RB1809	2.69E-05	1.23E-06
ins-31(ok3543)II	RB2552	1.13E-06	1.23E-06
ins-35(ok3297)V	RB2412	1.28E-06	1.23E-06
jnk-1(gk7)IV	VC8	1.09E-12	1.23E-06
lev-1(x427)IV	ZZ427	3.44E-08	1.23E-06
lev-8(x15)X	ZZ15	6.11E-07	1.23E-06
lig-4(ok716)III	RB873	8.04E-06	1.23E-06
lin-39(n709)III	MT1514	9.56E-07	1.23E-06
lon-2(e678)X	CB678	7.14E-11	1.23E-06
lov-1(ok522)II	RB753	9.60E-07	1.23E-06
mec-4(u253)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ908	8.68E-13	1.23E-06
mec-7(u448)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1033	3.35E-18	1.23E-06
mec-10(e1515)X	CB1515	1.17E-10	1.23E-06
mec-10(tm1552)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1413	3.69E-15	1.23E-06
mec-10(u20)X	AQ2533	4.29E-09	1.23E-06
mec-12(e1605)III; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1031	4.03E-16	1.23E-06
mec-12(u76)III; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ1037	6.75E-13	1.23E-06
mec-14(u55)III; bzIs18[pmec-4::YC2.12; lin-15(+)]	AQ1038	2.09E-20	1.23E-06
mec-18(u228)X; bzIs17[pmec-4::YC2.12; lin-15(+)]	AQ2649	1.43E-12	1.23E-06
mir-124(n4255)IV	MT13292	1.09E-08	1.23E-06
mod-1(ok103)V	MT9668	3.57E-08	1.23E-06
mod-5(n822)I	MT8944	2.72E-07	1.23E-06
nca-2(gk5)III	VC9	8.26E-06	1.23E-06
nhr-95(gk836)V	VC1759	6.15E-04	1.01E-03
nlp-1(ok1469)X	RB1340	5.22E-10	1.23E-06
nlp-2(tm1908)X	FX1908	2.76E-08	1.23E-06
nlp-3(ok2688)X	RB2030	7.66E-10	1.23E-06
nlp-8(ok1799)I	VC1309	4.65E-10	1.23E-06
nlp-12(ok335)IV	RB607	1.42E-07	1.23E-06
nlp-14(tm1880)X	FX1880	2.27E-07	1.23E-06
nlp-15(ok1512)I	VC1063	2.27E-04	1.23E-06
nlp-17(ok3461)IV	RB2498	3.63E-05	9.86E-05
nlp-18(ok1557)II	RB1372	1.66E-06	1.23E-06
nlp-20(ok1591)IV	RB1396	6.24E-07	1.23E-06
npr-1(ad609)X	DA609	9.67E-07	1.23E-06
npr-2(ok419)IV	AQ2056	2.49E-05	1.23E-06
npr-3(tm1583)IV	FX1583	1.40E-10	1.23E-06
npr-4(tm1782)X	AX1743	1.30E-03	2.12E-04
npr-5(ok1583)V	AX1745	1.84E-08	1.23E-06

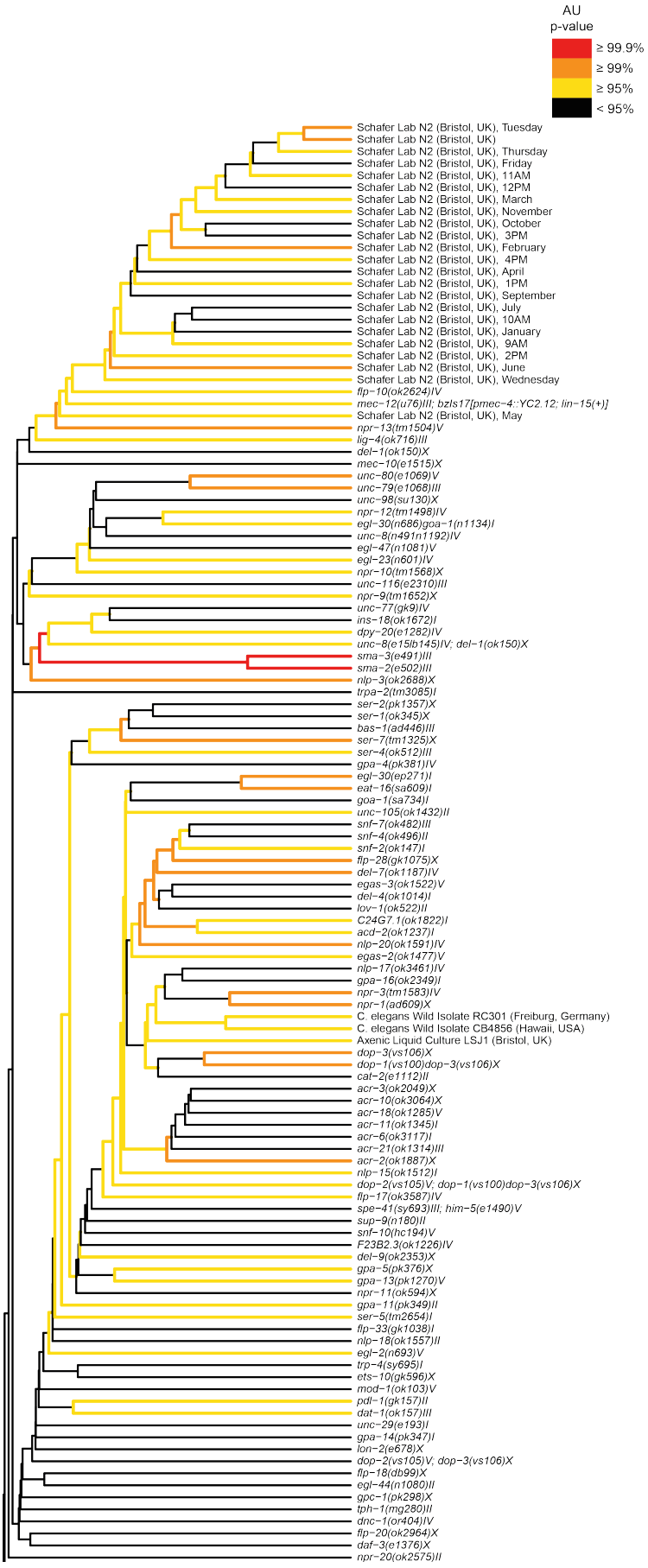
npr-7(ok527)X	RB761	1.85E-05	1.23E-06
npr-8(tm1553)X	FX1553	7.07E-06	1.23E-06
npr-9(tm1652)X	IC683	3.38E-09	1.23E-06
npr-10(tm1568)X	FX1568	2.11E-08	1.23E-06
npr-11(ok594)X	RB799	9.03E-07	1.23E-06
npr-12(tm1498)IV	FX1498	2.48E-12	1.23E-06
npr-13(tm1504)V	AQ2153	2.98E-03	2.87E-03
npr-20(ok2575)II	RB1958	1.13E-10	1.23E-06
ocr-3(ok1559)X	RB1374	1.69E-06	1.23E-06
ocr-4(tm2173)IV	FX2173	1.92E-06	1.23E-06
ocr-4(vs137)IV	LX950	4.44E-08	1.23E-06
ocr-4(vs137)ocr-2(ak47)IV	LX981	3.54E-06	1.23E-06
ocr-4(vs137)ocr-2(ak47)IV; ocr-1(ok132)V	LX982	7.88E-13	1.23E-06
octr-1(ok371)X	VC224	5.31E-04	1.23E-06
odr-3(n2150)V	CX2205	1.25E-10	1.23E-06
osm-9(ky10)IV	CX10	2.85E-06	1.23E-06
osm-9(ky10)trpa-1(ok999)IV	AQ1422	7.01E-05	1.23E-06
pdl-1(gk157)II	VC282	3.79E-06	1.23E-06
pkc-1(nj3)V	IK130	2.30E-09	1.23E-06
pkg-1(n478)IV	MT1073	3.23E-09	1.23E-06
pmk-1(km25)IV	KU25	4.31E-07	1.23E-06
pqn-66(ok1507)II	RB1350	6.19E-04	2.45E-06
rab-3(y250)II	NM210	3.90E-12	1.23E-06
ric-19(ok833)	RB946	5.28E-05	1.23E-06
sem-4(ga82)I	EW35	1.54E-10	1.23E-06
ser-1(ok345)X	DA1814	1.07E-07	1.23E-06
ser-2(pk1357)X	OH313	1.37E-07	1.23E-06
ser-4(ok512)III	AQ866	4.58E-06	1.23E-06
ser-5(tm2654)I	AQ2197	2.58E-03	1.75E-05
ser-6(tm2146)IV	FX2146	2.92E-04	1.23E-06
ser-7(tm1325)X	DA2100	8.77E-06	1.23E-06
sma-2(e502)III	CB502	1.80E-09	1.23E-06
sma-3(e491)III	CB491	2.50E-10	1.23E-06
snf-1(ok790)I	RB919	5.76E-05	1.23E-06
snf-2(ok147)I	RB641	2.41E-04	9.09E-04
snf-4(ok496)II	RB738	3.93E-04	4.81E-06
snf-5(ok447)II	RB687	6.02E-08	1.23E-06
snf-6(eg28)III	BZ28	5.29E-21	1.23E-06
snf-7(ok482)III	RB709	3.45E-06	1.23E-06
snf-8(ok349)IV	RB648	3.73E-04	4.81E-06
snf-9(ok957)IV	RB1030	4.11E-03	4.79E-04
snf-10(hc194)V	BA1093	8.84E-04	5.56E-05
snf-11(ok156)V	RM2710	1.22E-04	1.23E-06
sng-1(ok234)X	RB503	2.17E-05	1.23E-06
spe-41(sy693)III; him-5(e1490)V	PS4330	1.75E-09	1.23E-06
srp-8(ok291)V	RB559	2.40E-07	1.23E-06
sup-9(n180)II	MT180	1.62E-09	1.23E-06
syg-1(ok3640)X	RB2615	1.52E-09	1.23E-06
syg-2(ky671)X	CX6391	4.29E-07	1.23E-06
T14B1.1(ok1702)X	VC1243	1.65E-08	1.23E-06
tbh-1(n3247)X	MT9455	1.63E-05	1.23E-06
tdc-1(n3419)II	MT13113	6.57E-07	1.23E-06
tom-1(ok285)I	VC223	2.05E-11	1.23E-06
tph-1(mg280)II	MT15434	4.57E-09	1.23E-06
trp-1(sy690)III	TQ225	6.77E-08	1.23E-06
trp-2(gk298)III	VC602	4.27E-06	1.23E-06
trp-2(sy691)III	TQ194	3.92E-06	1.23E-06
trp-4(sy695)I	TQ296	1.50E-08	1.23E-06
trpa-1(ok999)IV	RB1052	5.86E-07	1.23E-06
trpa-2(ok3189)I	RB2351	4.59E-07	5.96E-06
trpa-2(tm3085)I	FX3085	1.61E-09	1.23E-06
trpa-2(tm3092)I	FX3092	8.84E-09	1.23E-06
trpl-2(ok2433)IV	RB1883	2.18E-08	1.23E-06
tyra-2(tm1846)X	FX1846	9.67E-05	2.45E-06
tyra-3(ok325)X	VC125	1.54E-04	1.23E-06
unc-1(e94)X	CB94	2.01E-10	8.63E-20

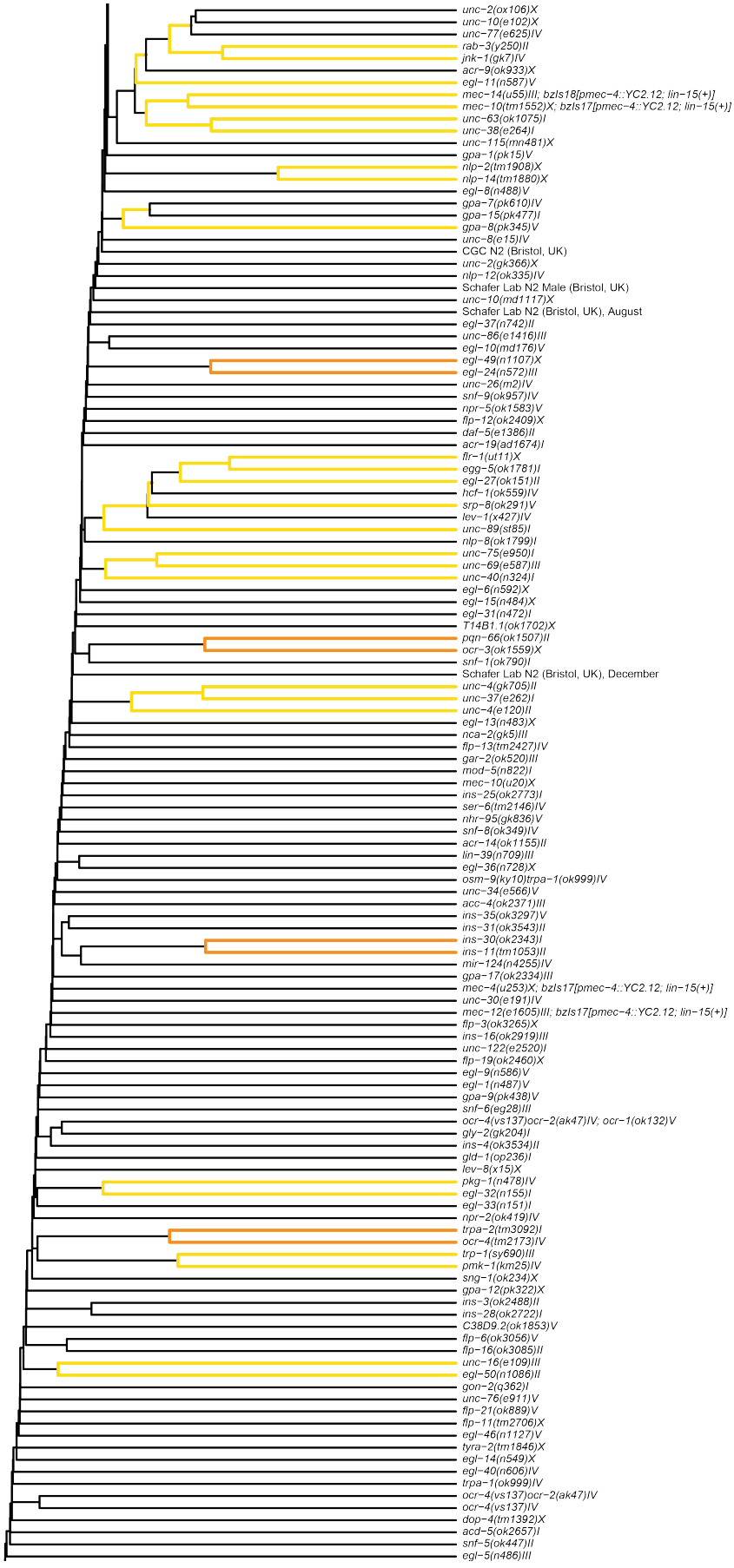
unc-1(e1598)X	CB1598	1.55E-17	1.23E-06
unc-2(gk366)X	VC854	1.75E-07	1.23E-06
unc-2(ox106)X	EG106	9.03E-10	1.23E-06
unc-3(e151)X	CB151	1.00E-09	1.23E-06
unc-4(e120)II	CB120	2.48E-22	1.23E-06
unc-4(gk705)II	VC1528	9.71E-09	1.23E-06
unc-7(e5)X	CB5	2.11E-36	1.23E-06
unc-8(e15)IV	CB15	1.58E-09	1.23E-06
unc-8(e15lb145)IV	MP145	2.08E-06	1.23E-06
unc-8(n491n1192)IV	MT2611	2.51E-12	1.23E-06
unc-8(e15lb145)IV; del-1(ok150)X	AQ2937	7.87E-07	1.23E-06
unc-9(e101)X	CB101	3.60E-10	1.23E-06
unc-10(e102)X	CB102	1.80E-08	1.23E-06
unc-10(md1117)X	NM1657	4.57E-09	1.23E-06
unc-14(e57)I	CB57	6.09E-15	4.98E-17
unc-16(e109)III	CB109	1.51E-09	1.23E-06
unc-18(e81)X	CB81	1.46E-09	1.23E-06
unc-26(m2)IV	DR2	9.14E-10	1.23E-06
unc-29(e193)I	CB193	5.21E-08	1.23E-06
unc-30(e191)IV	CB845	3.19E-10	1.23E-06
unc-31(e169)IV	CB169	2.06E-09	1.23E-06
unc-32(e189)III	CB189	1.02E-07	1.23E-06
unc-34(e566)V	CB566	9.14E-12	1.23E-06
unc-37(e262)I	CB262	5.61E-09	1.23E-06
unc-38(e264)I	CB904	1.11E-21	1.23E-06
unc-40(n324)I	MT324	4.38E-10	1.23E-06
unc-42(e270)V	CB270	3.66E-12	1.23E-06
unc-44(e1197)IV	CB1197	1.89E-19	1.07E-21
unc-55(e402)I	CB402	1.83E-10	1.23E-06
unc-60(e723)V	CB723	3.46E-09	1.23E-06
unc-63(ok1075)I	VC731	6.29E-11	1.23E-06
unc-69(e587)III	CB587	4.56E-09	1.23E-06
unc-75(e950)I	CB950	3.02E-10	1.23E-06
unc-76(e911)V	DR96	1.45E-12	1.23E-06
unc-77(e625)IV	DR1089	8.20E-11	1.23E-06
unc-77(gk9)IV	VC12	1.10E-13	1.23E-06
unc-79(e1068)III	CB1068	7.83E-18	1.23E-06
unc-80(e1069)V	CB1069	3.67E-14	1.23E-06
unc-86(e1416)III	CB1416	5.56E-15	1.23E-06
unc-89(e1460)I	CB1460	5.77E-08	1.23E-06
unc-89(st85)I	RW85	3.12E-10	1.23E-06
unc-98(su130)X	HE130	1.57E-13	1.23E-06
unc-101(m1)I	DR1	2.27E-10	1.23E-06
unc-103(e1597)III	CB1597	6.27E-14	1.23E-06
unc-104(e1265)II	CB1265	6.88E-24	4.95E-26
unc-105(ok1432)II	RB1316	4.16E-11	1.23E-06
unc-108(n501)I	MT1093	8.34E-11	1.23E-06
unc-108(n777)I	MT1656	9.50E-09	1.23E-06
unc-115(mn481)X	SP1789	3.11E-10	1.23E-06
unc-116(e2310)III	FF41	1.82E-09	1.23E-06
unc-118(e2331)X	CB4371	1.49E-09	1.23E-06
unc-122(e2520)I	CB4870	4.31E-07	1.23E-06
unc-127(hs13)V	HH27	1.34E-06	1.23E-06
vab-7(e1562)III	CB1562	3.12E-08	1.23E-06
zyg-9(b244)II	DH244	1.15E-06	1.23E-06
C. elegans Wild Isolate CB4856 (Hawaii, USA)	CB4856	6.34E-14	1.23E-06
C. elegans Wild Isolate RC301 (Freiburg, Germany)	RC301	3.45E-13	1.23E-06
Axenic Liquid Culture LSJ1 (Bristol, UK)	LSJ1	5.55E-18	1.23E-06
CGC N2 (Bristol, UK)	AQ2947	1.29E-04	1.23E-06
Schafer Lab N2 Male (Bristol, UK)	N2	3.50E-10	1.23E-06
Schafer Lab N2 (Bristol, UK), 9AM	N2	2.47E-16	1.23E-06
Schafer Lab N2 (Bristol, UK), 10AM	N2	1.06E-16	1.23E-06

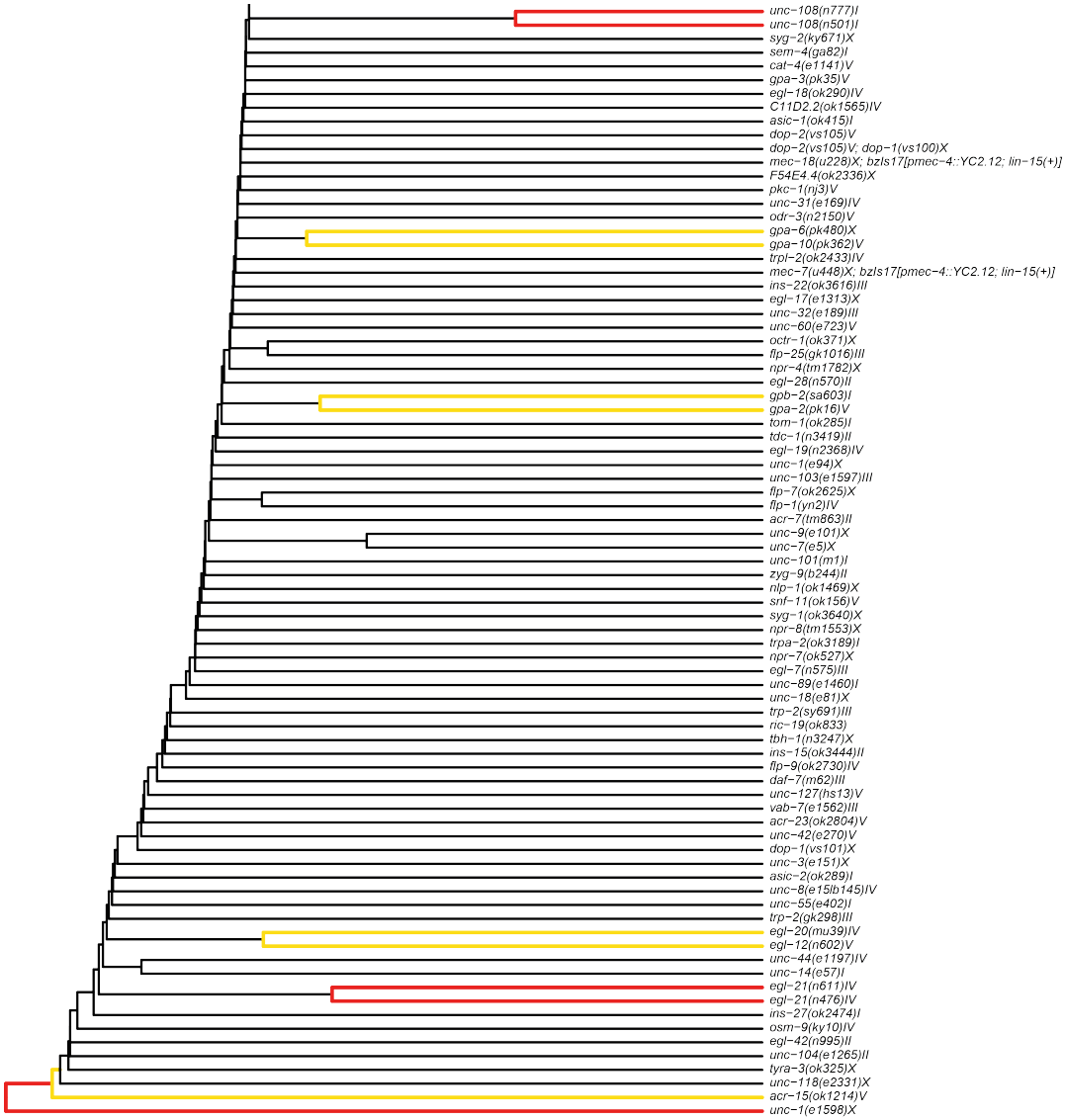
Schafer Lab N2 (Bristol, UK), 11AM	N2	4.59E-07	3.64E-06
Schafer Lab N2 (Bristol, UK), 12PM	N2	3.73E-07	1.23E-06
Schafer Lab N2 (Bristol, UK), 1PM	N2	1.77E-03	7.94E-05
Schafer Lab N2 (Bristol, UK), 2PM	N2	4.61E-05	1.23E-06
Schafer Lab N2 (Bristol, UK), 3PM	N2	4.62E-12	1.23E-06
Schafer Lab N2 (Bristol, UK), 4PM	N2	1.59E-06	6.43E-04
Schafer Lab N2 (Bristol, UK), Tuesday	N2	1.36E-02	1.58E-04
Schafer Lab N2 (Bristol, UK), Thursday	N2	1.89E-07	3.13E-05
Schafer Lab N2 (Bristol, UK), Wednesday	N2	1.87E-08	1.53E-05
Schafer Lab N2 (Bristol, UK), Friday	N2	2.78E-04	5.96E-06
Schafer Lab N2 (Bristol, UK), January	N2	1.66E-11	1.23E-06
Schafer Lab N2 (Bristol, UK), February	N2	1.47E-04	9.47E-06
Schafer Lab N2 (Bristol, UK), March	N2	7.42E-11	1.23E-06
Schafer Lab N2 (Bristol, UK), April	N2	1.82E-10	1.23E-06
Schafer Lab N2 (Bristol, UK), May	N2	2.25E-07	1.23E-06
Schafer Lab N2 (Bristol, UK), June	N2	9.97E-06	1.23E-06
Schafer Lab N2 (Bristol, UK), July	N2	6.28E-09	1.23E-06
Schafer Lab N2 (Bristol, UK), August	N2	7.69E-11	1.23E-06
Schafer Lab N2 (Bristol, UK), September	N2	1.93E-16	1.23E-06
Schafer Lab N2 (Bristol, UK), October	N2	7.27E-14	1.23E-06
Schafer Lab N2 (Bristol, UK), November	N2	1.06E-08	1.23E-06
Schafer Lab N2 (Bristol, UK), December	N2	1.78E-06	1.23E-06

Appendix D : Hierarchical Clustering

The full clustering is shown for all 330 strains and groups. Hierarchical clustering was performed using uncentered correlation and single linkage. Multiscale resampling was run with 10,000 bootstraps to generate approximately unbiased (AU) p values, measuring the confidence of tree selection for each branch. The horizontal length of each branch is scaled to its correlation value. The horizontal branches are color coded to show the AU of the clusters below them. AU ≥ 99.9% is colored red, AU ≥ 99% orange, AU ≥ 95% yellow, and all else is black. To understand the range of clustering values, the maximum pairwise correlation in the entire clustering was 0.95, between the group of lab-stock N2s and those tracked on Tuesday. In contrast, the minimum was 0.34, found when comparing *unc-1*(e1598) to all other groups. Similarly, while the top-most branch has an AU of 1 (as is expected for hierarchical clustering), the next greatest AU was 99.98% for the branch connecting the two alleles of *unc-108*. In contrast, 165 of the 329 branches had an AU of 0%.


AU
p-value
≥ 99.9%
≥ 99%
≥ 95%
< 95%
Schafer Lab N2 (Bristol, UK), Tuesday
Schafer Lab N2 (Bristol, UK)
Schafer Lab N2 (Bristol, UK), Thursday
Schafer Lab N2 (Bristol, UK), Friday
Schafer Lab N2 (Bristol, UK), 11AM
Schafer Lab N2 (Bristol, UK), 12PM
Schafer Lab N2 (Bristol, UK), March
Schafer Lab N2 (Bristol, UK), November
Schafer Lab N2 (Bristol, UK), October
Schafer Lab N2 (Bristol, UK), 3PM
Schafer Lab N2 (Bristol, UK), February
Schafer Lab N2 (Bristol, UK), 4PM
Schafer Lab N2 (Bristol, UK), April
Schafer Lab N2 (Bristol, UK), 1PM
Schafer Lab N2 (Bristol, UK), September
Schafer Lab N2 (Bristol, UK), July
Schafer Lab N2 (Bristol, UK), 10AM
Schafer Lab N2 (Bristol, UK), January
Schafer Lab N2 (Bristol, UK), 9AM
Schafer Lab N2 (Bristol, UK), 2PM
Schafer Lab N2 (Bristol, UK), June
Schafer Lab N2 (Bristol, UK), Wednesday
flp-10(ok2624)IV
mec-12(u76)III; bzIs17[pmec-4::YC2.12; lin-15(+)]
Schafer Lab N2 (Bristol, UK), May
npr-13(tm1504)V
lig-4(ok716)III
del-1(ok150)X
mec-10(e1515)X
unc-80(e1069)V
unc-79(e1068)III
unc-98(su130)X
npr-12(tm1498)IV
egl-30(n686)goa-1(n1134)I
unc-8(n491n1192)IV
egl-47(n1081)V
egl-23(n601)IV
npr-10(tm1568)X
unc-116(e2310)III
npr-9(tm1652)X
unc-77(gk9)IV
ins-18(ok1672)II
dpy-20(e1282)IV
unc-8(e15lb145)IV; del-1(ok150)X
sma-3(e491)III
sma-2(e502)III
nlp-3(ok2688)X
trpa-2(tm3085)I
ser-2(pk1357)X
ser-1(ok345)X
bas-1(ad446)III
ser-7(tm1325)X
ser-4(ok512)III
gpa-4(pk381)IV
egl-30(ep271)I
eat-16(sa609)I
goa-1(sa734)I
unc-105(ok1432)II
snf-7(ok482)III
snf-4(ok496)II
snf-2(ok147)I
flp-28(gk1075)X
del-7(ok1187)IV
egas-3(ok1522)V
del-4(ok1014)I
lov-1(ok522)III
C24G7.1(ok1822)I
acd-2(ok1237)I
nlp-20(ok1591)IV
egas-2(ok1477)V
nlp-17(ok3461)IV
gpa-16(ok2349)I
npr-3(tm1583)IV
npr-1(ad609)X
C. elegans Wild Isolate RC301 (Freiburg, Germany)
C. elegans Wild Isolate CB4856 (Hawaii, USA)
Axenic Liquid Culture LSJ1 (Bristol, UK)
dop-3(vs106)X
dop-1(vs100)dop-3(vs106)X
cat-2(e1112)II
acr-3(ok2049)X
acr-10(ok3064)X
acr-18(ok1285)V
acr-11(ok1345)I
acr-6(ok3117)I
acr-21(ok1314)III
acr-2(ok1887)X
nlp-15(ok1512)I
dop-2(vs105)V; dop-1(vs100)dop-3(vs106)X
flp-17(ok3587)IV
spe-41(sy693)III; him-5(e1490)V
sup-9(n180)II
snf-10(hc194)V
F23B2.3(ok1226)IV
del-9(ok2353)X
gpa-5(pk376)X
gpa-13(pk1270)V
npr-11(ok594)X
gpa-11(pk349)II
ser-5(tm2654)I
flp-33(gk1038)I
nlp-18(ok1557)III
egl-2(n693)V
trp-4(sy695)I
ets-10(gk596)X
mod-1(ok103)V
pdl-1(gk157)II
dat-1(ok157)III
unc-29(e193)I
gpa-14(pk347)I
lon-2(e678)X
dop-2(vs105)V; dop-3(vs106)X
flp-18(db99)X
egl-44(n1080)II
gpc-1(pk298)X
tph-1(mg280)II
dnc-1(or404)IV
flp-20(ok2964)X
daf-3(e1376)X
npr-20(ok2575)III


unc−2(ox106)X
unc−10(e102)X
unc−77(e625)IV
rab−3(y250)II
jnk−1(gk7)IV
acr−9(ok933)X
egl−11(n587)V
mec−14(u55)III; bzIs18[pmec−4::YC2.12; lin−15(+)]
mec−10(tm1552)X; bzIs17[pmec−4::YC2.12; lin−15(+)]
unc−63(ok1075)I
unc−38(e264)II
unc−115(mn481)X
gpa−1(pk15)V
nlp−2(tm1908)X
nlp−14(tm1880)X
egl−8(n488)V
gpa−7(pk610)IV
gpa−15(pk477)I
gpa−8(pk345)V
unc−8(e15)IV
CGC N2 (Bristol, UK)
unc−2(gk366)X
nlp−12(ok335)IV
Schafer Lab N2 Male (Bristol, UK)
unc−10(md1117)X
Schafer Lab N2 (Bristol, UK), August
egl−37(n742)III
unc−86(e1416)III
egl−10(md176)V
egl−49(n1107)X
egl−24(n572)III
unc−26(m2)IV
snf−9(ok957)IV
npr−5(ok1583)V
flp−12(ok2409)X
daf−5(e1386)III
acr−19(ad1674)I
flr−1(ut11)X
egg−5(ok1781)I
egl−27(ok151)II
hcf−1(ok559)IV
srp−8(ok291)V
lev−1(x427)IV
unc−89(st85)II
nlp−8(ok1799)I
unc−75(e950)II
unc−69(e587)III
unc−40(n324)I
egl−6(n592)X
egl−15(n484)X
egl−31(n472)I
T14B1.1(ok1702)X
pqn−66(ok1507)II
ocr−3(ok1559)X
snf−1(ok790)I
Schafer Lab N2 (Bristol, UK), December
unc−4(gk705)II
unc−37(e262)II
unc−4(e120)II
egl−13(n483)X
nca−2(gk5)III
flp−13(tm2427)IV
gar−2(ok520)III
mod−5(n822)II
mec−10(u20)X
ins−25(ok2773)I
ser−6(tm2146)IV
nhr−95(gk836)V
snf−8(ok349)IV
acr−14(ok1155)III
lin−39(n709)III
egl−36(n728)X
osm−9(ky10)trpa−1(ok999)IV
unc−34(e566)V
acc−4(ok2371)III
ins−35(ok3297)V
ins−31(ok3543)II
ins−30(ok2343)II
ins−11(tm1053)III
mir−124(n4255)IV
gpa−17(ok2334)III
mec−4(u253)X; bzIs17[pmec−4::YC2.12; lin−15(+)]
unc−30(e191)IV
mec−12(e1605)III; bzIs17[pmec−4::YC2.12; lin−15(+)]
flp−3(ok3265)X
ins−16(ok2919)III
unc−122(e2520)I
flp−19(ok2460)X
egl−9(n586)V
egl−1(n487)V
gpa−9(pk438)V
snf−6(eg28)III
ocr−4(vs137)ocr−2(ak47)IV; ocr−1(ok132)V
gly−2(gk204)I
ins−4(ok3534)II
gld−1(op236)I
lev−8(x15)X
pkg−1(n478)IV
egl−32(n155)I
egl−33(n151)I
npr−2(ok419)IV
trpa−2(tm3092)II
ocr−4(tm2173)IV
trp−1(sy690)III
pmk−1(km25)IV
sng−1(ok234)X
gpa−12(pk322)X
ins−3(ok2488)II
ins−28(ok2722)I
C38D9.2(ok1853)V
flp−6(ok3056)V
flp−16(ok3085)II
unc−16(e109)III
egl−50(n1086)II
gon−2(q362)I
unc−76(e911)V
flp−21(ok889)V
flp−11(tm2706)X
egl−46(n1127)V
tyra−2(tm1846)X
egl−14(n549)X
egl−40(n606)IV
trpa−1(ok999)IV
ocr−4(vs137)ocr−2(ak47)IV
ocr−4(vs137)IV
dop−4(tm1392)X
acd−5(ok2657)I
snf−5(ok447)III
egl−5(n486)III


unc-108(n777)I
unc-108(n501)I
syg-2(ky671)X
sem-4(ga82)II
cat-4(e1141)V
gpa-3(pk35)V
egl-18(ok290)IV
C11D2.2(ok1565)IV
asic-1(ok415)I
dop-2(vs105)V
dop-2(vs105)V; dop-1(vs100)X
mec-18(u228)X; bzIs17[pmec-4::YC2.12; lin-15(+)]
F54E4.4(ok2336)X
pkc-1(nj3)V
unc-31(e169)IV
odr-3(n2150)V
gpa-6(pk480)X
gpa-10(pk362)V
trpl-2(ok2433)IV
mec-7(u448)X; bzIs17[pmec-4::YC2.12; lin-15(+)]
ins-22(ok3616)III
egl-17(e1313)X
unc-32(e189)III
unc-60(e723)V
octr-1(ok371)X
flp-25(gk1016)III
npr-4(tm1782)X
egl-28(n570)III
gpb-2(sa603)I
gpa-2(pk16)V
tom-1(ok285)I
tdc-1(n3419)II
egl-19(n2368)IV
unc-1(e94)X
unc-103(e1597)III
flp-7(ok2625)X
flp-1(yn2)IV
acr-7(tm863)II
unc-9(e101)X
unc-7(e5)X
unc-101(m1)I
zyg-9(b244)II
nlp-1(ok1469)X
snf-11(ok156)V
syg-1(ok3640)X
npr-8(tm1553)X
trpa-2(ok3189)I
npr-7(ok527)X
egl-7(n575)III
unc-89(e1460)I
unc-18(e81)X
trp-2(sy691)III
ric-19(ok833)
tbh-1(n3247)X
ins-15(ok3444)II
flp-9(ok2730)IV
daf-7(m62)III
unc-127(hs13)V
vab-7(e1562)III
acr-23(ok2804)V
unc-42(e270)V
dop-1(vs101)X
unc-3(e151)X
asic-2(ok289)I
unc-8(e15lb145)IV
unc-55(e402)II
trp-2(gk298)III
egl-20(mu39)IV
egl-12(n602)V
unc-44(e1197)IV
unc-14(e57)I
egl-21(n611)IV
egl-21(n476)IV
ins-27(ok2474)I
osm-9(ky10)IV
egl-42(n995)III
unc-104(e1265)II
tyra-3(ok325)X
unc-118(e2331)X
acr-15(ok1214)V
unc-1(e1598)X