Morphology-Guided Graph Search for Untangling Objects: *C. elegans* Analysis

T. Riklin Raviv[1], V. Ljosa[2], A.L. Conery[3], F.M. Ausubel[3], A.E. Carpenter[2], P. Golland[1], and C. Wählby[2]

[1] Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, MA
[2] Imaging Platform, Broad Institute of MIT and Harvard, Cambridge, MA
[3] Dept. of Molecular Biology and Center for Computational and Integrative Biology, Mass. General Hospital, Boston, MA

Abstract. We present a novel approach for extracting cluttered objects based on their morphological properties. Specifically, we address the problem of untangling *Caenorhabditis elegans* clusters in high-throughput screening experiments. We represent the skeleton of each worm cluster by a sparse directed graph whose vertices and edges correspond to worm segments and their adjacencies, respectively. We then search for paths in the graph that are most likely to represent worms while minimizing overlap. The worm likelihood measure is defined on a low-dimensional feature space that captures different worm poses, obtained from a training set of isolated worms. We test the algorithm on 236 microscopy images, each containing 15 *C. elegans* worms, and demonstrate successful cluster untangling and high worm detection accuracy.

1 Introduction

Recent progress in robotic sample preparation, combined with automated microscopy and image analysis, enables high-throughput screening experiments for testing biological processes such as immunity, behavior, and metabolism. For example, high-throughput screening of the roundworm *Caenorhabditis elegans* is used to test tens of thousands of chemical or genetic perturbations to identify promising drugs and regulators [1,2,3]. Automatic processing of the vast amount of data obtained from a screen is therefore necessary.

Existing methods compute image-based statistics, e.g., the ratio of fluorescing-stained area in the image to the area covered by worms [4]. However, many scientific questions require measurements on individual worms, such as shape and location of reporter signals. Worms in these images often overlap and cluster into clumps, making analysis based on individual worms challenging. A few solutions have been proposed for the analysis of individual worms in low-throughput video sequences where the worms are disambiguated by tracking them over time [5,6,7]. Methods for resolving high resolution 3-D images of worms have also been demonstrated [8,9]. These methods, however, are not suitable for the comparatively low-resolution, 2-D images produced in high-throughput experiments. Extracting overlapping worms in images has been recently demonstrated by thresholding the curvature of skeleton segments of a worm cluster [10]. Yet, this

method has a limited capability to detect curvy worms or clustered worms in complex configurations.

While isolated worms can be extracted easily based on the differences of image intensities, the image data alone is not sufficient to delineate clustered worms. Moreover, edges and intensity variations within the worms often mislead conventional segmentation algorithms. Nevertheless, while the different poses of the worms introduce significant extrinsic geometrical differences, all worms share similar intrinsic geometrical properties such as length and thickness profile. The shape characteristics should therefore play a key role in a segmentation algorithm.

There exist a few algorithms that incorporate shape priors into the segmentation process for extracting multiple, possibly overlapping, objects [11], but these shape-based algorithms assume that the expected number of objects to segment is known. In practice, due to computational constraints, only a few partially occluded objects can be segmented simultaneously.

We present a conceptually novel algorithm for untangling clusters of worms based exclusively on their poses. Our formulation leads to a computationally efficient, morphology-guided graph search that relies on a probabilistic model of the worm poses, learned from training data. We model the skeleton of a worm cluster by a graph whose vertices correspond to worm segments and search over paths in the graph that are more likely to represent complete individual worms while minimizing overlap. We formulate the problem as a minimization of a cost function. To reduce computational complexity, we use a greedy search strategy to approximate the optimal exhaustive search.

The elongated, thin structure of the worms, together with the similarity in their thickness profiles, motivate the use of their medial axis transform (skeleton) as their shape descriptor [12,13,14]. Since the deformations of the worms are nearly isometric (no stretching, shrinkage, etc.), the worms' variability in appearance can be captured by a low-dimensional feature space. We apply principal component analysis (PCA) to a comprehensive set of worm shapes from the training set. Only a limited number of eigen vectors are needed to reliably represent the entire population. We use the shape space to define a probability distribution that guides the detection of the most probable worm descriptors within the graph search.

While the PCA-based representation of the worms' shape has been demonstrated before [15], the contributions of this paper include the graph representation, the graph search algorithm and the greedy approximation. To the best of our knowledge, we demonstrate the first robust, automatic method for identifying a large number of objects (worms) in cluttered clusters.

We test the algorithm on 236 microscopy images, each containing a well with 15 *C. elegans* worms, and demonstrate successful cluster untangling and high worm-detection accuracy. The quality of the results obtained with a relatively low computational complexity shows the promise of our method to remove the computational bottleneck in large-scale biological experiments.

2   Probabilistic Shape Model

In this section, we briefly review the shape representation we use in our algorithm. The details are described elsewhere [15]. A worm descriptor consists of n equally spaced

control points $\mathbf{c} \in \mathcal{R}^2$ along the worm skeleton and the average worm thickness, obtained by using the medial axis transform [12,13]. This $(2n + 1)$ dimensional vector is used for reconstruction and retrieval. The worm descriptors are aligned by similarity transformation (translation and rotation) based on their first and second moments.

We use PCA to project the descriptors of the worms in the training set onto a low-dimensional feature space. The PCA model implies a probability measure on the space spanned by the most significant eigen vectors,

$$P(\mathbf{x}) \propto \exp(-\mathbf{w}^M \Sigma_M^{-1} \mathbf{w}), \tag{1}$$

where $\Sigma_M = \mathrm{diag}(\lambda_1 \ldots \lambda_M)$ are the highest eigen values and $\mathbf{w} = \{w_1, \ldots, w_M\}$ are the weights obtained by projecting the worm descriptor $\mathbf{x}$ onto the space spanned by the M most significant eigen vectors. The dimensionality M is chosen so that the selected eigen values explain 99% of the variance.

3   Graph Representation and Search Algorithm

We now focus on the main contribution of the paper, namely performing cluster untangling using morphology-guided graph search. We assume that the worm clusters are already segmented from the background and a worm likelihood measure $P(\mathbf{x})$ is defined based on the training set, according to Eq. (1).

3.1   Graph Representation of the Worm Cluster Skeleton

Let $G_s = \{V, E\}$ denote a sparse, directed graph that represents the skeleton S of a worm cluster, where $V = \{v_i\}$ and $E = \{e_{i,j}\}$ denote the graph vertices and edges, respectively. We denote the number of vertices by $\mathcal{V} = |V|$. Fig. 1b shows the skeleton that corresponds to the worm cluster shown in Fig. 1a with the intersection points marked in red. These cluster intersection points partition the skeleton into segments that are represented by the graph vertices, as demonstrated in Fig. 1c. The length of a skeleton segment is determined by the number of pixels it contains. An edge $e_{i,j}$

(a)   (b)   (c)   (d)   (e)
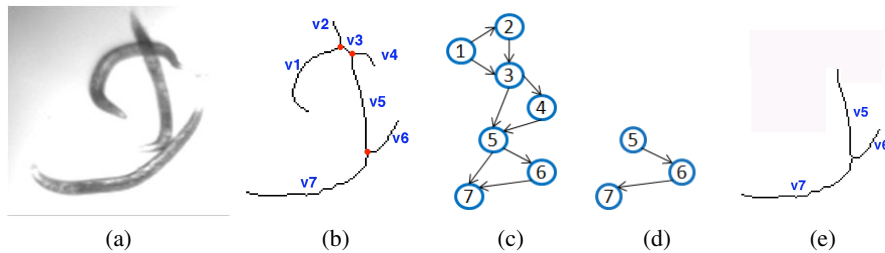
Fig. 1. Graph representation of a worm cluster. (a) Original image. (b) Cluster skeleton. Intersection points are marked in red. (c) Graph representation G_s of the skeleton shown in b. Each segment becomes a node and each intersection becomes a clique in the graph. (d) A simple path in G_s and (e) the corresponding skeleton segments. These skeleton segments form a branch which is unlikely to represent a worm skeleton.

connects two vertices $\{v_i, v_j\}$ if their corresponding segments share a common inter-section point. To avoid path duplication we impose an order on the graph nodes, making the graph directed, i.e., $e_{i,j} = 1$ and $e_{j,i} = 0$ if $i < j$. A simple path in the graph is a chain of connected vertices without cycles. Note, however, that not every simple path in the graph represents a probable worm skeleton. A path in the graph that includes, for example, a 3-clique represents 3 worm segments with a common intersection point. These segments form a branch rather than an open curve and therefore are not likely to represent a worm skeleton. See Fig. 1d-e for an example of a path that forms a branch.

We construct an $\mathcal{V} \times \mathcal{V}$ adjacency matrix representing G_s. Neighborhoods of vertices are determined by detecting the intersections associated with each skeleton segment. We use the breadth-first search (BFS) algorithm to detect the paths in G_s. Let L denote the maximal length of a path, i.e., its total number of vertices minus one. We construct length-l paths, where $l = 0 \dots L$, by visiting the end vertices of each of the existing length-$(l-1)$ paths and attaching the neighboring vertices. That is, if the last node of a length-$(l-1)$ path has two neighbors, then two additional length-l paths are detected.

A general graph search of this form (as in the traveling-salesman problem) has a combinatorial complexity of $O(\mathcal{V}!)$ when $L = \mathcal{V}$. Yet, the nature of the problem re-duces its complexity significantly. Since branches (cliques of more than two vertices) are excluded, the maximal length L of the likely paths corresponds to the number of intersection points. Let D denote the maximal size of the clique associated with an in-tersection point in the graph (i.e., the number of segments attached to it). Then the num-ber of likely paths in the graph is in on the order of $O(L^2 D^2)$. In practice we limit the length of the paths even further so that the length in pixels of the corresponding skeleton segments does not exceed the length of the longest worm seen during the training.

In the example shown in Fig. 1 there are 3 intersection points, thus $L = 3$. For all intersection points in this example $D = 3$. In this particular graph the probable paths include 7 length-0 paths, 9 length-1 paths, 6 length-2 paths and 4 length-3 paths.

3.2   Search for the Most Probable Paths in the Graph

We obtain the approximated number of worms in a given cluster, denoted here by K, from the ratio of the cluster area and the median area of the training worms. Let ρ_k denote a path in G_s that represents the skeleton of the k-th worm. We can now formalize the cluster untangling problem as a search for K paths is G_s that minimize the cost function

$$E(\rho_1 \dots \rho_K) = -\sum_{k=1}^{K} \log P(\rho_k) + \alpha \sum_{k=2}^{K} \sum_{l=1}^{k-1} |\rho_k \cap \rho_l| + \beta |\tilde{V}_K|, \qquad (2)$$

where $\tilde{V}_K = \{v | v \in G_s, v \notin [\rho_1 \cup \rho_2 \dots \cup \rho_K]\}$. Let us now take a closer look at this function. The first term (the *likelihood cost*) encourages selection of the paths that are most likely to represent complete, individual worms. The second term (the *overlap cost*) encourages minimization of the pairwise overlap between the selected paths:

$$|\rho_k \cap \rho_l| = \{v | v \in \rho_k \text{ and } v \in \rho_l\}. \qquad (3)$$

The third term (the *leftover cost*) is the number of vertices $|\tilde{V}_k|$ in G_s that are not included in the union of the selected paths $\{\rho_k\}$. This term encourages the 'coverage'

of the worm cluster skeleton S. The scalar weights α and β balance the terms in the function. In all of our experiments we set α and β to 2.

The global minimum of this optimization problem can be obtained by an exhaustive search for all subsets of K out of $\mathcal{P}$ (all) paths in G_s. This is, however, a combinatorial problem of order $\binom{\mathcal{P}}{K}$. To reduce the computational time we apply a greedy strategy. At each stage we make a locally optimized choice of a path in the graph until we select K paths. Given $\{\rho_1 \ldots \rho_{k-1}\}$ selected in the previous iterations of the search, the greedy algorithm seeks a path ρ_k that minimizes

$$E(\rho_k|\rho_1 \ldots \rho_{k-1}) = -\log P(\rho_k) + \alpha \sum_{l=1}^{k-1} |\rho_k \cap \rho_l|. \tag{4}$$

We can represent the cost of the first selected path ρ_1 by reducing Eq. (4) for $k = 1$:

$$E(\rho_1) = -\log P(\rho_1) \tag{5}$$

Once we complete $K - 1$ iterations, we also test if K is the optimal number of paths to represent the cluster. We compare the incremental cost of having only $K - 1$ paths, i.e.,

$$E(\rho_K = \emptyset|\rho_1 \ldots \rho_{K-1}) = \beta|\tilde{V}_{K-1}|, \tag{6}$$

to the cost of adding one additional path:

$$E(\rho_K|\rho_1 \ldots \rho_{K-1}) = -\log P(\rho_K) + \alpha \sum_{l=1}^{K-1} |\rho_K \cap \rho_l| + \beta|\tilde{V}_K|. \tag{7}$$

In other words, we extract K rather than $K - 1$ paths from a cluster only if the cost of having an additional path ρ_K is less than the cost of not having its contribution to the 'coverage' of the skeleton represented by G_s. This condition can be viewed as a stopping criterion of the algorithm.

The greedy strategy still attempts to minimize the original cost function in Eq. 2 since the $K - 1$ cost functions in the form of Eq. (4) and the cost function in Eq.(7) sum up to the cost functional in Eq. (2) for selecting K paths in G_s. Replacing Eq. (7) with Eq. (6), we obtain the cost function in Eq. (2) for $K - 1$ paths.

3.3 Morphology-Guided Graph Search Algorithm

The proposed algorithm can be summarized as follows:

- **Initialize**
  - Calculate K.
  - Use a breadth-first search to detect $\mathcal{P}$ paths in G_s as described in Section 3.1. This is the *path candidates* set.
  - Define an empty set of *selected paths*.
  - Calculate the *likelihood cost* $E(\rho) = -\log P(\rho)$ of each path.
- **For** $k = 1$ **to** $(K - 1)$**:**
  - Move the path ρ_k with the lowest cost $E(\rho_k|\rho_1 \ldots \rho_{k-1})$ in the *path candidates* set to the *selected paths* set.
  - For each path ρ_l in the *path candidates* set compute the *overlap cost* (Eq. (3)) due to its intersection with the path ρ_k.
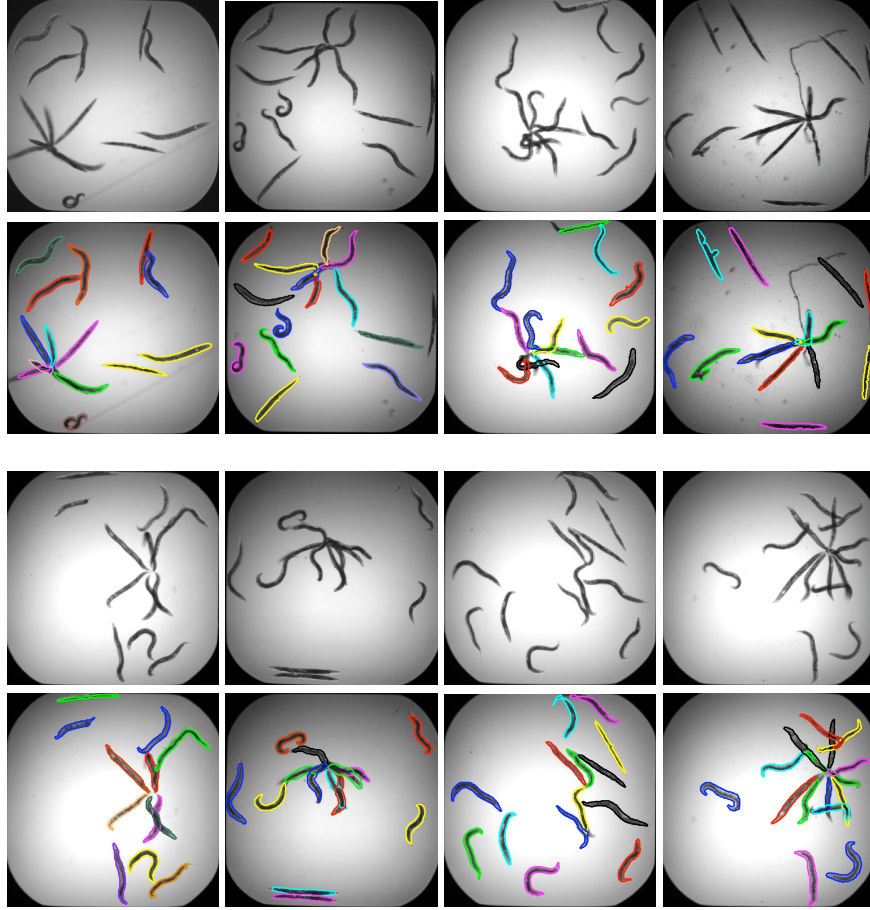
Fig. 2. Example images (odd rows) with their clusters resolved by the algorithm (even rows)

– **Determine the total number of selected paths**
  - Update the costs of the remaining paths in the *path candidates* set by adding the *leftover cost*, (Eq. 7). Let C_K denote the lowest cost. This is the cost for adding the K-th path to the *selected paths* set.
  - Calculate the *leftover cost* for not selecting an additional path (Eq. 6). Let C_{K-1} denote this cost.
  - If $C_K < C_{K-1}$ output $\{\rho_1 \ldots, \rho_K\}$, otherwise output $\{\rho_1 \ldots \rho_{K-1}\}$.

4   Experiments

We evaluated the algorithm on an image set from a viability screen of *C. elegans* that were infected by a bacterial pathogen, washed, transferred to 384-well plates containing liquid medium with the compound to be tested and incubated until the infection killed untreated worms. The worm plates were imaged within $3, 24, 48, 72, 96, 120$ and 144
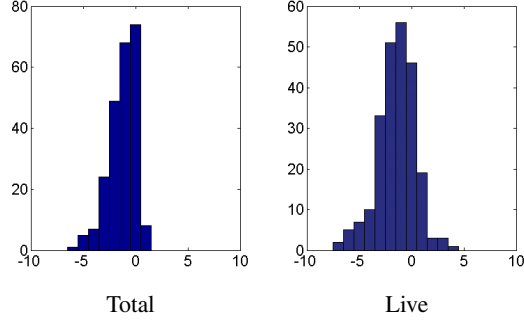
Fig. 3. Histograms of the differences in count between the automatic worm detection and the manual worm count for each of the 236 images, containing 3479 worms. Left: Total worm count differences. Right: Live worm count differences.

hours after treatment by an automated microscope with a $2\times$ magnification lens. Details of the sample preparation and image acquisition are available elsewhere [4]. The worms were segmented from the background using thresholding [15]. The training set contained 454 individual manually detected worms. Worm descriptors were generated using 21 equally-spaced control points along the worm skeletons, together with their average thickness. The feature space of the worm descriptors was spanned by the 10 most significant eigen vectors.

We applied the proposed algorithm to 236 images containing 3479 worms for which we had manual annotations by an expert. Fig. 2 presents successful untangling examples of images with complex worm clusters. To quantify the performance of the algorithm, we compared the number of worms detected by our algorithm with the manual count. In addition, we classified the detected worms as dead or alive, based on a simple classifier we trained on the training set used to construct the PCA space. Typically, dead worms are stiff and therefore straight. We then compared the number of worms classified as live to the manual live worm counts. Fig. 3 shows histograms of the count differences between the automatic worm detection and the manual worm count for each of the 236 images. Due to uneven illumination, worms near the well edges are not always detected by the preliminary segmentation. This leads to an underestimated worm counts. Overestimated total worm counts are caused by worm splitting or misclassifying debris as worms. In summary, 9% of the worms were not detected while the false detection percentage was 0.23%. Note, that underestimation in the total worm counts implies ignoring part of the data while overestimated counts (very few here) may introduce larger bias to the classification results.

5   Conclusions

This paper addresses a particulary challenging problem of extracting clustered objects. The objects are entangled and their poses vary, thus hierarchical models [16] or discriminative boosting algorithms [17] that have been proposed for image parcellation into multiple (brain) structures, are not applicable.

By exploiting the specific characteristics of our data, we developed a novel detection method based on concepts from machine learning and graph theory. Representing clustered worm segments as graph vertices, we search for paths in the graph that are more likely to represent complete, individual worms. The thin, elongated structure of

the worms motivates the use of their skeletons as a shape descriptor and allows a reduction in the computation complexity of the graph search. Yet, we believe that our morphology-guided graph search can be generalized to detect cluttered objects of different shapes with a suitable choice of descriptors.

References

1. Artal-Sanz, M., de Jong, L., Tavernarakis, N.: *Caenorhabditis elegans*: A versatile platform for drug discovery. Biotechnol. J. 1, 1405–1418 (2006)
2. Kamath, R.S., Ahringer, J.: Genome-wide RNAi screening in *Caenorhabditis elegans*. Methods 30(4), 313–321 (2003)
3. Sifri, C.D., Begun, J., Ausubel, F.M.: The worm has turned – microbial virulence modeled in *Caenorhabditis elegans*. Trends Microbiology 13, 119–127 (2005)
4. Moy, T.I., Conery, A.L., Larkins-Ford, J., Wu, G., Mazitschek, R., Casadei, G., Lewis, K., Carpenter, A.E., Ausubel, F.M.: High-throughput screen for novel antimicrobials using a whole animal infection model. ACS Chemical Biology 4(7), 527–533 (2009)
5. Ramot, D., Johnson, B., Berry, T., Carnell, L., Goodman, M.: The parallel worm tracker: A platform for measuring average speed and drug-induced paralysis in nematodes. PLoS One 3(5) (2008)
6. Geng, W., Cosman, P., Berry, C.C., Feng, Z., Schafer, W.R.: Automatic tracking, feature extraction and classification of C. elegans phenotypes. IEEE Transactions on Biomedical Engineering 51(10), 1811–1820 (2004)
7. Restif, C., Metaxas, D.: Tracking the swimming motions of C. elegans worms with applications in aging studies. In: Metaxas, D., Axel, L., Fichtinger, G., Székely, G. (eds.) MICCAI 2008, Part I. LNCS, vol. 5241, pp. 35–42. Springer, Heidelberg (2008)
8. Long, F., Peng, H., Liu, X., Kim, S.K., Myers, E.: A 3D digital atlas of C. elegans and its application to single-cell analyses. Nature Methods 6(9), 667–672 (2009)
9. Murray, J.I., Bao, Z., Boyle, T.J., Boeck, M.E., Mericle, B.L., Nicholas, T.J., Zhao, Z., Sandel, M.J., Waterston, R.H.: Automated analysis of embryonic gene expression with cellular resolution in C. elegans. Nature Methods (2008)
10. Rizvandi, N.B., Pizurica, A., Philips, W.: Machine vision detection of isolated and overlapped nematode worms using skeleton analysis. In: Proc. of ICIP, pp. 2972–2975 (2008)
11. Cremers, D., Sochen, N., Schnörr, C.: A multiphase dynamic labeling model for variational recognition-driven image segmentation. IJCV 66(1), 67–81 (2006)
12. Blum, H.: A transformation for extracting new descriptors of shape. Models for the perception of speech and visual form, 362–380 (1967)
13. Goutsias, J., Shonfeld, D.: Morphological representation of discrete and binary images. IEEE Transactions on Signal Processing 39(6), 1369–1379 (1991)
14. Stephens, G.J., Kerner, J.B., Bialek, W., Ryu, W.S.: Dimensionality and dynamics in the behavior of C. elegans. PLoS Comput. Biol. 4(4) (2008)
15. Wählby, C., Riklin-Raviv, T., Ljosa, V., Conery, A.L., Golland, P., Ausubel, F.M., Carpenter, A.E.: Resolving clustered worms via probabilistic shape models. In: Proc. of ISBI (2010)
16. Pohl, K., Bouix, S., Nakamura, M., Rohlfing, T., McCarley, R.W., Kikinis, R., Grimson, W.E.L., Shenton, M.E., Wells, W.M.: A hierarchical algorithm for MR brain image parcellation. IEEE TMI 26(9), 1201–1212 (2007)
17. Tu, Z., Narr, K., Dollár, P., Dinov, I., Thompson, P., Toga, A.: Brain anatomical structure segmentation by hybrid discriminative/generative models. IEEE TMI 27(4), 495–508 (2008)