BIOINFORMATICS

Vol. 29 ISMB/ECCB 2013, pages i18–i26
doi:10.1093/bioinformatics/btt223

Automated cellular annotation for high-resolution images of adult *Caenorhabditis elegans*

Sarah J. Aerni[1,2], Xiao Liu[3], Chuong B. Do[4], Samuel S. Gross[1], Andy Nguyen[5], Stephen D. Guo[1], Fuhui Long[6], Hanchuan Peng[7], Stuart S. Kim[5] and Serafim Batzoglou[1,*]

[1]Department of Computer Science, Stanford University, 353 Serra Mall, Stanford, CA 94305, USA, [2]Biomedical Informatics Training Program, Stanford University School of Medicine, 251 Campus Drive, Stanford, CA 94305, USA, [3]School of Life Sciences, Tsinghua University, Beijing 100084, China, [4]23andMe, Inc., 1390 Shorebird Way Mountain View, CA 94043, USA, [5]Department of Developmental Biology, Stanford University School of Medicine, 279 Campus Drive Beckman Center B300, Stanford, CA 94305-5329, USA, [6]Janelia Farm Research Campus, Howard Hughes Medical Institute, Ashburn, VA, USA and [7]Allen Institute for Brain Science, Seattle, WA, USA

ABSTRACT

Motivation: Advances in high-resolution microscopy have recently made possible the analysis of gene expression at the level of individual cells. The fixed lineage of cells in the adult worm *Caenorhabditis elegans* makes this organism an ideal model for studying complex biological processes like development and aging. However, annotating individual cells in images of adult *C.elegans* typically requires expertise and significant manual effort. Automation of this task is therefore critical to enabling high-resolution studies of a large number of genes.

Results: In this article, we describe an automated method for annotating a subset of 154 cells (including various muscle, intestinal and hypodermal cells) in high-resolution images of adult *C.elegans*. We formulate the task of labeling cells within an image as a combinatorial optimization problem, where the goal is to minimize a scoring function that compares cells in a test input image with cells from a training atlas of manually annotated worms according to various spatial and morphological characteristics. We propose an approach for solving this problem based on reduction to minimum-cost maximum-flow and apply a cross-entropy–based learning algorithm to tune the weights of our scoring function. We achieve 84% median accuracy across a set of 154 cell labels in this highly variable system. These results demonstrate the feasibility of the automatic annotation of microscopy-based images in adult *C.elegans*.

Contact: saerni@cs.stanford.edu

1 INTRODUCTION

Comprehensive gene expression profiling using high-resolution images from *in situ* hybridization or fluorescent reporter experiments has become feasible owing to advances in imaging technology and the growing availability of genomic resources. Image-based gene expression analysis is especially promising for the study of *Caenorhabditis elegans*, as the fixed developmental lineage of all 959 cells in the adult worm permits, at least in principle, direct comparison of expression values of reporter genes in analogous cells from different individuals. In practice, however, the process of identifying the cells in an image of an adult worm is usually performed manually, which is extremely tedious and time-consuming. Owing to the significant expertise required for accurate cell identification, most *in situ* analyses of gene expression in adult *C.elegans* to date have been limited to much lower regional resolution.

A crucial step in making high-resolution global gene expression analysis possible in the worm is to develop computational approaches that can extract expression data from images, thereby allowing high-throughput conversion of unstructured image data into well-structured gene expression tables suitable for computational analysis. Previous methods for single-cell gene expression analysis in model organisms have largely relied on time-series information and region markers to map the locations of individual cells (Bao *et al.*, 2006; Fowlkes *et al.*, 2008; Keränen *et al.*, 2006; Luengo Hendriks *et al.*, 2006; Murray *et al.*, 2008; Zhao *et al.*, 2008). In *C.elegans*, however, tracking cell lineages is extremely difficult after the embryonic stage owing to the amount of time required for monitoring the development of each individual worm and the large morphological changes that take place during development. Therefore, techniques that allow mapping of single cells without the assistance of time series information are needed.

For worms in the first larval stage (L1) following embryonic development, the absolute and relative spatial locations of individual cells are highly constrained. Based on this insight, a marker-guided spatially constrained bipartite matching algorithm was previously developed for labeling cells in 3D images of L1 worms (Long *et al.*, 2008). This method was shown to achieve high accuracy (86%) for annotating 357 out of the 558 cells present in the L1 developmental stage (Long *et al.*, 2009). For adult *C.elegans*, however, the cell labeling task is substantially more difficult. In addition to a near doubling of the number of somatic cells from 558 to 959, thousands of germ line cells are also present in the adult worm resulting in 2500–3500 total cells. The additional germ line cells occupy locations near somatic cells of interest throughout the trunk of the worm, which poses a substantial difficulty for annotation approaches that rely on location-based features alone. Moreover, the number of somatic cells is variable in the adult worm unlike worms at the L1 stage, further decreasing the effectiveness of spatial cues for cell identification. Methods have been proposed that combine the segmentation of cells from the 3D images and their label annotation into a single step for the L1 worm (Qu *et al.*, 2011) to improve the overall accuracy on the set of 82 muscle cells. This method requires cell-specific markers to be consistently expressed in a subset of cells and again relies on an invariable cell lineage.

*To whom correspondence should be addressed.

More concretely, although the adult *C.elegans* is post-mitotic, meaning no additional somatic cell divisions take place once development is complete, not every individual produces precisely the same number of cells. In our data, we have observed a high degree of variability in a set of four intestinal cells, which may each undergo one additional division to give rise to two daughter cells. To accurately assign labels to the cells in an individual, it is crucial to recognize whether the parent intestinal cell or the two daughter cells are present.

In this article, we formulate the problem of labeling cells in 3D images of adult *C.elegans* as a combinatorial optimization problem. Our method builds on prior work by using a rich scoring function that incorporates additional features beyond spatial location, such as cell size, intensity of a muscle-marker gene and neighborhood density. We extend our formulation to accommodate the cell number variation that arises owing to post-embryonic cell division. Finally, we show how to solve the resulting optimization problem efficiently via reduction to minimum-cost maximum-flow, and describe a straightforward cross-entropy–based algorithm for fitting parameters of the model. We test the method on a set of 25 manually curated images of day 1 adult worms. Using our algorithm, we achieve 84% median accuracy on a subset of 154 cells in the adult worm, demonstrating the feasibility of automated methods for this task.

2 METHODS

2.1 Overview of method

In this section, we present methods for automatic annotation of adult worms. Images in the adult were obtained using an experimental protocol similar to the approach described in previous work that performed automated single-cell annotation to obtain high-resolution gene expression data in the larval worm (Liu *et al.*, 2009). In these images, single cells are visualized through a combination of 4′,6-diamidino-2-phenylindole (DAPI) staining of DNA in all cells (shown in the blue channel), and green fluorescent protein (GFP) expression in a subset of nuclei (shown in the green channel). These two complementary approaches enable detection of all cells within a worm, and identification of specific marker cells to guide cell labeling. Figure 1 shows a sample image of a worm where the 3D images have been projected along the z-axis.

Previous work (Long *et al.*, 2008) attempted to solve the annotation task for worms in the first larval stage (L1) using a marker-guided two-stage bipartite matching algorithm. In this approach, unlabeled cells in an input worm image were matched with annotated cells from a reference atlas on the basis of cell location. This hierarchical strategy focuses on a small subset of GFP-marker expressing cells before considering all cells in the L1 worm. It includes a heuristics approach that through an iterative graph pruning scheme imposes relative spatial constraints on cell labelings. Because of the highly stereotyped spatial arrangement of cells within worm images (Liu *et al.*, 2009), location-based features alone were sufficient to obtain good accuracy for cellular annotation at this early stage of development.

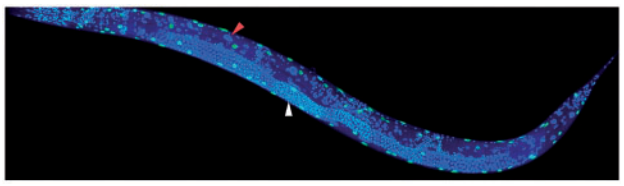

Fig. 1. This image of a day 1 adult hermaphrodite contains DAPI-stained nuclei, visible in blue, and the GFP body wall muscle reporter in a subset of cells in green. The 3D image is projected along the z-axis to create this 2D image. In the figure, the heterogeneity of the worm cell shapes is easily visible, including the elongated shape of the green muscle cells, the large number of germ line cells (white arrowhead) and the large intestinal cells (red arrowhead)

As discussed in the preceding section, however, adult worms pose a substantially greater challenge for cellular annotation than L1 worms. To achieve reasonable accuracy in adult worms, which have an order-of-magnitude more cells in total, we propose an approach that incorporates additional features into a cost function that, when used to solve the annotation task, increases accuracy compared with using location alone. We first formalize the task of label assignment as a combinatorial optimization problem, then introduce the set of features used in the cost function. We show how the optimization problem can be solved using a minimum-cost maximum-flow algorithm, and propose simple extensions that allow for the explicit incorporation of additional, variable, cell division events during late development. We finally describe the parameter estimation process used to assign weights for these features.

2.2 Formulation of cell lineage annotation as a combinatorial optimization problem

Suppose that a 3D input image contains p cells, $x = (x_1, x_2, \ldots, x_p)$, each of whose locations and boundaries have already been extracted in a preprocessing step. Let $y = (y_1, y_2, \ldots, y_p)$ denote the corresponding labels that we wish to predict for each cell, where $y_i \in \mathcal{Y}$ for some set of candidate labels $\mathcal{Y} = \{\ell_0, \ell_1, \ldots, \ell_q\}$. Here, we assume that ℓ_0 is the label used to denote cells that have no specified annotation. We refer to this label as the dud label. $\ell_1, \ell_2, \ldots, \ell_q$ correspond to the q different types of cells identified by an expert human annotator for images in our training set; in our work, for example, ℓ_1 through ℓ_{107} represent different types of muscle cells found in adult worms, including 95 body wall muscles, ℓ_{108} through ℓ_{116} represent different types of hypodermal cells and ℓ_{117} through ℓ_{142} are 26 different intestinal cells. Finally, ℓ_{143} through ℓ_{154} are a set of intestinal cells that participate in variable cell division, which we will address later.

The task of determining the appropriate label for each cell can be posed as a combinatorial optimization problem in which

(i) Each cell x_i is assigned exactly one label from $\mathcal{Y}$,
(ii) Each label ℓ_j (for $j \neq 0$) is assigned to exactly one cell and
(iii) The dud label ℓ_0 may be assigned to multiple cells (e.g. all germ line cells in the training data are given the label ℓ_0).

Let $[a_{ij}] \in \{0,1\}^{p\times(q+1)}$ be a matrix whose entries a_{ij} are set to 1 whenever cell x_i is assigned label ℓ_j, and 0 otherwise. Similarly, let $[c_{ij}] \in \mathbb{R}^{p\times(q+1)}$ be a matrix of costs for each possible assignment of cell to label. Formally, the labeling task can be written as the following integer programming problem:

$$\begin{aligned} \underset{[a_{ij}]}{\text{minimize}} & \sum_{i=1}^{p}\sum_{j=0}^{q} a_{ij}c_{ij}, \\ \text{subject to} & \sum_{j=0}^{q} a_{ij} = 1, \quad i = 1, \ldots, p \\ & \sum_{i=1}^{p} a_{ij} = 1, \quad j = 1, \ldots, q \\ & a_{ij} \in \{0,1\} \end{aligned} \tag{1}$$

where the constraints ensure that all cells are assigned exactly one label, and vice versa.

2.3 Defining cost matrices

The choice of costs $[c_{ij}]$ is the key factor in determining the quality of the predicted labelings from our bipartite matching algorithm. In this section, we describe an approach for constructing cost matrices that takes into account multiple aspects of compatibility between a cell x_i and a putative label ℓ_j:

(i) *Cell location*: A 3D vector indicating the location of a cell x_i in worm-coordinate space with each dimension standardized to have zero mean and unit variance.
(ii) *Cell size*: A scalar value indicating the size of a cell x_i as measured by the number of voxels contained in the cell object.
(iii) *GFP expression levels*: A scalar value indicating and mean green channel voxel values in the cell object, standardized within each worm to have zero mean and unit variance across all cells.
(iv) *DAPI intensity*: Two scalar values indicating the mean and the standard deviation of the blue channel voxel intensities in the cell object, standardized within each worm to have zero mean and unit variance across all cells.
(v) *Neighborhood*: Two scalar values indicating the number of cells within a certain distance of the cell's center (either a 10 voxel or 25 voxel radius).
(vi) *Cell shape*: A set of scalar values representing the percent of variance captured along each axis from the principal components analysis (PCA) of the voxel locations contained in the cell. This roughly represents the elongation along a set of axes for the cell. In addition, the value of the x-coefficient for the first eigenvector is included.

All of the above features are those typically used when an expert human annotator is presented with a new adult worm to label.

Consider a single aspect of compatibility between a cell x_i and a putative label ℓ_j. Here, we focus specifically on *cell location* (though the construction of cost matrices for other aspects of compatibility is done in the same way). We begin by assuming that we have access to a training set S of worm images, each of which have been fully annotated by a human expert. Our goal is to define a cost matrix $[c_{ij}]$ such that c_{ij} reflects the extent to which the location of a cell x_i in an input image is compatible with the location of cells that were annotated with label ℓ_j in the training set S.

Let $\langle x_i \rangle^{\text{location}}$ denote the 3D vector of coordinates for a given cell x_i in standardized worm-coordinate space. Each dimension is standardized to have zero mean and unit variance. One simple choice of cost is given by the squared *Mahalanobis distance*,

$$c_{ij}^{\text{location}} = (\langle x_i \rangle^{\text{location}} - \mu_j^{\text{location}})^T \Sigma_j^{\text{location}^{-1}} (\langle x_i \rangle^{\text{location}} - \mu_j^{\text{location}}) \quad (2)$$

where μ_j is a 3D vector giving the average location of training cells with label ℓ_j, and Σ_j is the sample covariance matrix of these locations. The Mahalanobis distance can be thought of as a variation on a weighted Euclidean distance measure that accounts for correlation between coordinates in different dimensions. Similar costs may be defined for cell size, GFP expression level and additional features. Note that in the case of scalar-valued features (e.g. cell size), the above expression reduces to

$$c_{ij}^{\text{size}} = \left(\frac{\langle x_i \rangle^{\text{size}} - \mu_j^{\text{size}}}{\sigma_j^{\text{size}}} \right)^2 \quad (3)$$

where the mean μ_j^{size} and standard deviation σ_j^{size} are estimated based on all cells from the training data with a particular label ℓ_j. Finally, given multiple separate cost matrices, we can construct a single cost matrix by taking a simple weighted sum:

$$c_{ij} = e^{w_{\text{location}}} c_{ij}^{\text{location}} + e^{w_{\text{size}}} c_{ij}^{\text{size}} + \ldots + e^{w_{\text{shape}}} c_{ij}^{\text{shape}} \quad (4)$$

where $w = (w_{\text{location}}, w_{\text{size}}, \ldots, w_{\text{shape}})$ is a vector of (log) weights.

2.4 Formulation of cell lineage annotation as a minimum-cost flow

One approach to solving the combinatorial optimization problem in one is a straightforward application of maximum weight bipartite matching (a.k.a., linear assignment) that was used in (Long *et al.*, 2008). In this approach, one constructs a bipartite graph containing p nodes in each partition. The left partition contains a single node for each input cell. The right partition contains a single node for each non-null label $\ell_1, \ldots, \ell_q$ and $p-q$ nodes for the null label ℓ_0. The cost for matching the ith node in the left partition with the jth node in the right partition is set to c_{ij}, and the minimum cost matching can be found in $O(p^3)$ time using the Hungarian algorithm.

Another alternative is to reduce 1 to an instance of the transportation problem, which eliminates the need to explicitly enumerate nodes with null labels. For general transportation tasks, the algorithm of Kleinschmidt and Schannath (Kleinschmidt and Schannath, 1995) takes $O(pq \log p(p + \log q))$ time, which is an improvement over the $O(p^3)$ afforded by the Hungarian algorithm. Here, we take an even more general approach that also avoids creating these same duplicate nodes by reduction to minimum-cost flow. We show that a simple algorithm for minimum cost flow achieves $O(pq(q + \log p))$ time as a consequence of the structure of our problem. The flexibility of the minimum-cost flow approach allows us to further extend the algorithm to handle the special case of cells that undergo variable cell division.

Construct a directed graph $G = (V, E)$ containing p nodes (denoted $x_1, \ldots, x_p$) representing cells in the input worm, $q+1$ nodes representing the possible labels for these cells (denoted $\ell_0, \ldots, \ell_q$ and which include the dud label) and two additional nodes s and t representing the sink and source for the graph. The edges of the graph consist of the following:

(i) (s, x_i): an edge from the source node to a node representing the i^{th} cell in the input worm
(ii) (x_i, ℓ_j): an edge from the i^{th} cell node in the input worm to the j^{th} label node
(iii) (ℓ_j, t): an edge from each label node to the sink

There are p total edges of the first type, $p(q+1)$ total edges of the second type and $q+1$ total edges of the third type for each input worm.

With each edge (a, b) associate a lower bound, L_{ab}, an upper bound U_{ab} and a cost C_{ab}. These constraints and costs are defined differently for each type of edge:

- $L_{s,x_i} = U_{s,x_i} = 1$ and $C_{s,x_i} = 0$, $\forall i$.
- $L_{\ell_j,t} = U_{\ell_j,t} = 1$ and $C_{\ell_j,t} = 0$, $j > 0$.
- $L_{\ell_0,t} = U_{\ell_0,t} = p - q$ and $C_{\ell_0,t} = 0$
- $L_{x_i,\ell_j} = 0$, $U_{x_i,\ell_j} = 1$ and $C_{x_i,\ell_j} = c_{ij}$ as defined in the section describing the formulation of the combinatorial optimization, $\forall i, j$

where the first two constraints ensure that all cells and non-dud labels are matched exactly once, the third constraint ensures that dud labels are provided to exactly $p-q$ cells, and the last constraint sets the costs for matching particular cells with particular labels.

The minimum-cost maximum-flow problem is stated as follows:

$$\min \sum_{(u,v)\in E} C_{uv} f_{uv}$$

$$\text{subject to} \sum_{v:(u,v)\in E} f_{uv} - \sum_{v:(v,u)\in E} f_{vu} = \beta_u \qquad \forall u \in V,$$

$$L_{uv} \le f_{uv} \le U_{uv} \qquad \forall (u,v) \in E,$$

where β_u represents the signed supply value for each node, defined as

$$\beta_u = \begin{cases} p & \text{if } u = s \\ -p & \text{if } u = t \\ 0 & \text{otherwise.} \end{cases}$$

For any solution to the minimum-cost maximum-flow problem stated above, the edges with $f_{x_i,\ell_j} = 1$ represent the annotations of cell x_i with label ℓ_j.

The computational advantage of a minimum-cost maximum-flow formulation can be more clearly seen using a slightly modified but equivalent formulation of the problem above in which the ℓ_0 node is omitted from the graph, the costs of all (ℓ_j, t) edges (for $j>0$) are adjusted by subtracting $C_{\ell_0,t}$ and the supply values are adjusted accordingly to achieve a target flow of q (rather than p). When solving the latter formulation, at most q augmentations are required to find the optimal solution using the simple minimum-cost maximum-flow algorithm suggested by Edmonds and Karp (Edmonds, 1972), giving an asymptotic runtime of $O(pq(q+\log p))$. This is a substantial improvement over the $O(p^3)$ Hungarian algorithm for weighted perfect matching, as $q \ll p$ in our setting.

In our experiments, we opted for the more straightforward formulation described here (which we extend in later sections) and used the Network Simplex function in the Lemon Library (Király and Kovács, 2010; Dezs *et al.*, 2011) for optimization. As expected, this gave a substantial practical speed-up compared to a highly efficient implementation of the Hungarian algorithm for bipartite matching.

2.5 Annotation with variable cell divisions

Although the cell lineage for adult worms is known and largely fixed, some exceptions exist. In particular, we observed in our data that the four posterior intestinal cells (two ventral and two dorsal) can each undergo an additional cell division. As an example, let's assume ℓ_j is the j^{th} cell in the *C.elegans* atlas (Altun and Hall, 2008) and is determined to be present in all adult cells. However, the data indicate that ℓ_j divides and gives rise to two additional cells, an anterior and posterior daughter cell. We will refer to these sets of cells, which undergo variable cell division as the set of labels at the indices $P = \{143, 146, 149, 152\}$. For each of the parent cells $j \in P$, let Daughter$(j) = \{j+1, j+2\}$ denote the set of daughter cells produced when j divides. Biologically, either a parent cell is present, or it has divided and given rise to the two daughter cells. In particular, a matching should never simultaneously label both a parent cell and any daughter cell; similarly, a matching that labels one of the daughter cells should also label its sister cell. These types of constraints cannot be modeled using the standard bipartite matching algorithm.

The minimum-cost flow formulation can capture some of these constraints resulting from the variability in cell divisions in the annotation process by adding nodes and edges to the graph constructed in the previous section. For each variably dividing parent cell $j \in P$, create two decision nodes, d_j^1 and d_j^2, and construct edges as follows:

- (ℓ_j, d_j^1) with constraints $L_{\ell_j,d_j^1} = 0$, $U_{\ell_j,d_j^1} = 1$ and cost $C_{\ell_j,d_j^1} = 0\ \forall j \in P$
- (ℓ_{j+1}, d_j^1) with constraints $L_{\ell_{j+1},d_j^1} = 0$, $U_{\ell_{j+1},d_j^1} = 1$ and cost $C_{\ell_{j+1},d_j^1} = 0\ \forall j \in P$
- (ℓ_{j+2}, d_j^2) with constraints $L_{\ell_{j+2},d_j^2} = 0$, $U_{\ell_{j+2},d_j^2} = 1$ and cost $C_{\ell_{j+2},d_j^2} = 0\ \forall j \in P$
- (ℓ_0, d_j^2) with constraints $L_{\ell_0,d_j^2} = 0$, $U_{\ell_0,d_j^2} = 1$ and cost $C_{\ell_0,d_j^2} = 0\ \forall j \in P$
- (d_j^k, t) with constraints $L_{d_j^k,t} = 1$, $U_{d_j^k,t} = 1$ and cost $C_{d_j^k,t} = 0\ \forall j \in P, k \in \{1,2\}$

Any previous edge directly connecting ℓ_j, ℓ_{j+1} or ℓ_{j+2} to the sink node t (for any $j \in P$) should be deleted. Finally, the amount of flow from the dud label l_0 to the sink t is reduced as a result of flow redirected to the decision nodes.

- (ℓ_0, t) with constraints $L_{\ell_0,t} = p-q-r$, $U_{\ell_0,t} = p-q-r$ and cost $c_{\ell_0,t} = 0$

where r is the number of parent cells that can variably divide; here $r=4$.

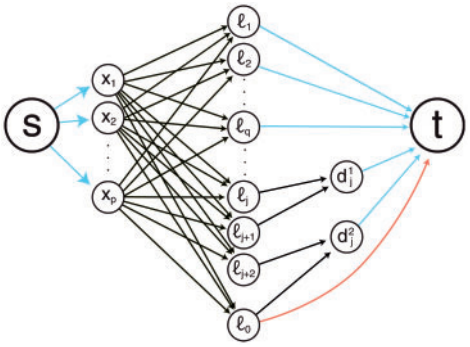

Fig. 2. The network consists of a source node (s), a sink node (t), a set of nodes for each of the p total input cells, $x_1, x_2, \ldots, x_p$, a dud label node ℓ_0 and q unique labels of which a subset participating in variable cell divisions. The label ℓ_j represents a parent and ℓ_{j+1} and ℓ_{j+2} its daughters as described in the text. d_j^1 and d_j^2 represent the decision nodes for that cell divisions. The source node, s, pushes p units of flow into the network. All blue edges have a lower and upper bound of 1. The black edges all have lower bound of 0 and an upper bound of 1. Finally, the red edge from the dud label node, ℓ_0, to the sink node, t, has a lower and upper bound equal to $p-q-r$ where r is the number of parent cells that may divide

The decision nodes are used to impose mutual exclusion constraints. For example, d_j^1 ensures that exactly one of the two labels, either the parent ℓ_j or the anterior daughter ℓ_{j+1}, will be present in the final annotation. Similarly, d_j^2 is used to determine whether the posterior daughter ℓ_{j+2} will be annotated.

Ideally, the method achieves a solution in which the pair of daughter cells (ℓ_{j+1}, ℓ_{j+2}) or only a parent cell (ℓ_j) is used by routing a unit of flow through the dud, ℓ_0. However, this construction imposes no restriction on both daughter cells being labeled simultaneously, nor does it prevent the parent being labeled alongside the posterior daughter. For this reason a heuristic post-processing step is needed after the two assignments are performed independently (Note, a graph construction exists that does account for these additional constraints. However, given the current formulation of costs, a solution cannot be obtained that properly scores these relationships. A modification to the cost formulation should be explored in future work.)

In particular, if the posterior daughter ℓ_{j+2} is labeled, then ensure that the anterior daughter ℓ_{j+1} is also labeled (reassigning from the parent ℓ_j to the anterior daughter ℓ_{j+1} as needed). Similarly, if the posterior daughter ℓ_{j+2} is unlabeled, this implies that the parent cell has not divided, and so ensures it is labeled (reassigning from the anterior daughter ℓ_{j+1} to parent ℓ_j as needed). Figure 2 presents a representation of the structure of the network.

2.6 Estimating parameter weights for improved matching results

This section focuses on the method for learning the appropriate weights used in Equation (4). Once determined, using the weights in the optimal assignment in the matching problem will yield an annotation of the p cells in an input worm.

Define $\mathcal{Y}(x)$ as the set of all possible matchings for an input worm x, and define $y' \in \mathcal{Y}(x)$ as the solution that minimizes the network flow problem for a given set of weights $w = (w_{\text{location}}, w_{\text{size}}, \ldots, w_{\text{shape}})$. The goal is to learn the appropriate weights w for combining the cost matrices as defined in Equation (4) such that for each worm $x \in S$ the solution y' is close to the true labeling, y.

Let $Q(y)$ denote the number of cells that have been assigned a label other than ℓ_0 in y (i.e. the number of cells named by the expert annotator,

$Q(y) = |\{y_i | y_i \neq l_0\}|$). Also, define $R(y, y')$ to be the number of cells with the same label in y and y', other than l_0 (i.e. $R(y, y') = |\{y_i | y_i = y'_i \wedge y_i \neq l_0\}|$).

Define the learning objective as the average percentage of cells that have the correct annotation for any given weight w.

$$h(\mathcal{X}, \mathcal{Y}, w) = \frac{1}{N} \sum_{(x,y) \in \mathcal{X}, \mathcal{Y}} \frac{R(y, y')}{Q(y)} \quad (5)$$

where N is the number of training worms. A supervised learning technique must be chosen that estimates a set of weights w for the cost function such that for the resulting predictions the difference between the predicted labels y' and the true labels y is globally minimized.

Finding a solution that minimizes not the cost of the labels for the individual cells, but rather the global matching is challenging. Various methods have been proposed to solve this parameter estimation problem (de Boer *et al.*, 2005; Caetano *et al.*, 2007, 2009; Le and Smola, 2007; Petterson *et al.*, 2009; Rubinstein and Kroese, 2004; Taskar, 2004; Taskar *et al.*, 2005; Tsochantaridis *et al.*, 2005). Some approaches may be efficient computationally, such as max-margin structured estimation and need to be explored (Taskar, 2004; Taskar *et al.*, 2005). Here, we take a sampling approach described below.

Start with a distribution over the space of weights (i.e. $\mathbb{R}^n$), where n is the number of features used in the matching problem in 1 and randomly sample from this a number of times (e.g. 100) to obtain a set of weight vectors, $\{w[1], \ldots, w[100]\}$. For each sampled weight vector, w, solve the network flow problem modeling variable cell division for each training worm, and compute the average per-worm accuracy in annotation given by Equation (5). Then take the top fraction of performing weights (the so called elite set), and use them to estimate a new distribution from which to sample the next set of weights. This is repeated until convergence as defined by a plateau in the objective function. For sampling, each of the dimensions of the weight vectors are drawn independently. In particular, each w_i is sampled from a separate distribution $\mathcal{N}(\mu_i, \sigma_i^2)$. After each iteration, the mean and standard deviation of the w_is for the top η% of samples scored are used to estimate each weight's μ_i and σ_i in the next iteration of sampling. The algorithm is run until convergence.

Intuitively this means that the matching problem is solved on the worms using a set of sampled weights. As the space of possible weights is searched, the evaluation of the performance of the sample at each step allows the algorithm to identify a distribution for weights that show good performance on the training. In essence, it is sampling the set of top performers, and removing the poor-performing set from the population. A schematic is shown in Figure 3.

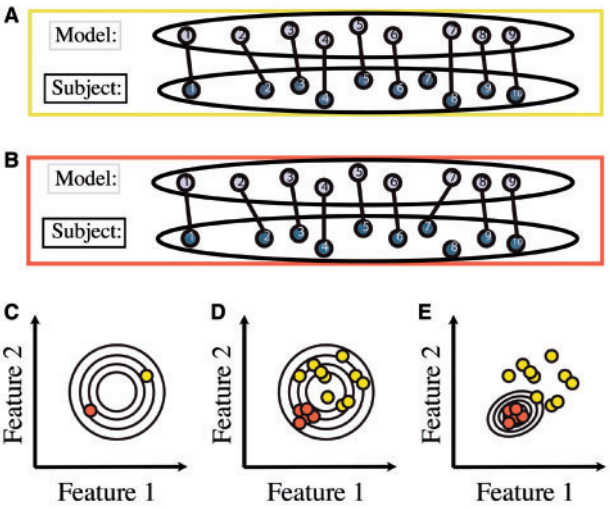

Fig. 3. This schematic of the sampling approach depicts the method used to learn optimal weights for the label assignment problem. At the top, the two toy matching examples (**A** and **B**) show different performances of model fits at sampled weights (**C**). The 'Model' shows the available cell labels. The 'Subject' is a training example with its true labels indicated, 1 through 10. The edges represent the final matching computed using the sampled weights. In the top (yellow) example, the solution labels only six cells correctly. In the lower red model, seven labels are correctly assigned. We use a red shade to represent higher accuracy in this schematic. To learn the weights for the assignment problem, the method proceeds as follows: 1. Randomly sample many weights for the features (2 features shown here) from the given distribution and solve the matching problem on the training sets, computing the average accuracy of the annotation at those weights (**D**). 2. Identify the top-scoring samples (represented by red) and use their weights to recompute a new distribution for the next iteration (shown in E). Repeat until convergence

3 RESULTS AND DISCUSSION

A set of 25 day 1 adult hermaphrodites were imaged using fluorescent confocal microscopy, producing a series of 3D image stacks. These images were processed similarly to the approaches used in the first larval stage (Liu *et al.*, 2009). Each worm was stained with DAPI, making all nuclei visible in the blue channel. In addition, the worms contained muscle-specific GFP markers, making a subset of body wall muscle cells visible in the green channel as shown in Figure 1. The cells were automatically segmented using a modified version of a gradient-based approach described in other work (Li *et al.*, 2007), which was adapted and parallelized to improve performance on the larger adult worms. The segmentation was manually corrected, and from the set of 25 expert-curated worms, we extracted features of the cells, described in full detail in Section 2.3. In short, the orientation of the worm was determined manually (head, tail, ventral). We then extracted many features of the cells including location, DAPI and GFP intensity, shape (e.g. size and elongation) and neighborhood density.

Each worm was manually annotated with 142 labels consisting of a set of intestinal, muscle and hypodermal cells. In addition, each worm was also annotated with a set of labels for cells that undergo variable division, consisting of four intestinal cells that could each divide and give rise to two daughter cells, accounting for 12 unique additional labels. Therefore, each worm was annotated with a subset of the 154 total labels. These particular cells were targeted by the expert annotator to study the biological process of aging. The muscle and intestinal tissues degenerate most during aging and are therefore cells of interest. The hypodermal cells were also included as a set that were readily recognized by the expert annotator.

3.1 Performance evaluation of an untrained matching approach using 5-fold cross-validation

We first present the performance of automatically annotating cells by combining different sets of features into a scoring function where these features are linearly combined. Table 1 summarizes the results of combining the indicated set of features with equal weight in a cost function for assigning a given label to a cell. These costs are used to solve the minimum-cost labeling of all cells using the set of available labels. This assignment problem

Table 1. Results for 5-fold cross-validation on single-cell label annotation with equally weighted features: the model is built from 20 training worms and used to label the remaining five

Name	Features	Duds scored	Median	μ	σ^2
loc	Location	−	0.38	0.36	0.012
		+	0.43	0.41	0.014
gs	Location, size, GFP	−	0.68	0.67	0.0079
		+	0.70	0.71	0.018
full	Location, size, GFP & DAPI, shape, neighborhood	−	0.66	0.65	0.0088
		+	0.74	0.73	0.011

Note: In all cases features were equally weighted. The symbol + in the column 'Dud cells scored' indicates the μ and σ^2 were estimated for unlabeled cells. These unlabeled cells were given a score of 0 otherwise. The features used are described in detail in Section 2.3. The per-worm accuracy is computed for each worm using Equation (5). The median, mean and variance are reported across all 25 worms.

and cost functions are described in full in Section 2. The table reports average accuracy as computed using Equation (5), the percentage of uniquely labeled cells in a test worm receiving the correct annotations. In each model, as only a subset of cells in a worm is assigned a unique label (at most 150), there is a large number of unlabeled cells. We present two models, one in which we ignore the unlabeled cells in the scoring of an assignment, and another in which the unlabeled cells are scored. For each combination of features, the table shows a model where unlabeled cells incur no cost in using the cost matrix described in the Equation (4) $c^k_{i0} = 0\ \forall i, k$, the i^{th} cell receiving the label of dud (a label given to cells without a unique annotation, e.g. germ line cells) and the k^{th} feature. For those models in which the dud cells were scored, the cost is computed from the estimated μ_0 and σ_0 as formulated in Equation (4). In Table 1, these models are denoted by the inclusion of the symbol + in the 'Dud cells scored' column. When dud cells are not scored, the column contains the symbol −. In all experiments in Table 1, $w_i = 0\ \forall i$.

As the result with a cost, $c_{i0} = 0\ \forall i$ should always result in the use of the model with no cell division taking place, in experiments where dud label nodes are unscored, the bipartite matching formulation is used in which all 154 cells are assigned. For + models, the minimum-cost maximum flow is solved using the LEMON open source graph template library (Dezs *et al.*, 2011) (The LEMON library uses integral cost values to solve the network flow resulting in a decrease in precision. It is noted that experiments were run to permit precision to five decimal places and had little effect on the results.)

In Table 1, the first two rows represent the model using location alone. When the cost of unlabeled cells is not included (*loc*-), the model achieves an average accuracy of 36% per worm across the 5-fold cross-validation experiment. Including a cost for unlabeled cells (*loc*+), results in an increase in accuracy to 41% per worm. These location-based models achieve the lowest accuracy across all models shown in Table 1.

Each subsequent model includes the feature of location in addition to other morphological features (see Section 2.3 for full detail). Incorporating two features of a cell, gfp intensity and size, results in a large increase in accuracy per worm to 67% (model *gs*- in Table 1), while scoring the unlabeled cells in the model shows further improvement to 71% (model *gs*+ in Table 1).

However, the highest mean accuracy score belongs to the model including a large set of features *full*+ at 73%. In the subsequent section, this set of features was used to train a more complete model in which the weights for combining features in the scoring function are learned using the sampling approach described in Section 2.6.

3.2 Performance evaluation using trained feature weights

Using the set of 25 worms, we assessed the ability to improve accuracy of the annotation by learning feature weights for the cost function defined in Equation (4). Applying the sampling technique (Section 2.6), we report results on a 5-fold cross-validation experiment using the features from the *full*+ model.

The initial distribution for each weight is set as $\mathcal{N}(0, 4)$. Each iteration performs 100 independent samplings, and uses the $\eta = 10$ top-scoring samples to compute the distribution of the weights for the subsequent iteration. In the first iteration of training, solving the label assignment problem using the LEMON Library (Dezs *et al.*, 2011; Király and Kovács, 2010) took on average 1 second (with an inter-quartile range of 0.69 to 1.29 s). Each model was trained for over 30 iterations, at which point all models converged (the point where training accuracy no longer increases). The model taken from the 30th iteration of each cross-validation run is used for testing on the held-out set of worms. Results are shown in Table 2.

As reported above, solving the matching problem using location alone resulted in the mean accuracy of only 41%, even when estimating the μ and σ of the locations of unlabeled cells. Improvements were observed by linearly combining location with additional features in the cost function, but further improvements in accuracy can be achieved by training the cost model to weight the features differently. Learning these weights led to an increase from the untrained per-worm mean accuracy of 73% to a per-worm mean accuracy of 77% on the worm model *full*+.

In addition to reporting a per-worm accuracy, a per-label accuracy is provided. This is the mean and median accuracies achieved on each label when it was present in a worm. Figure 4 shows the histogram of accuracies on a per-cell basis. In this histogram, the distribution of per-cell accuracies when performing annotation using a model that uses location alone is clearly shifted to the left. The fully trained model with learned feature weights

Table 2. Results for 5-fold cross-validation on single-cell label annotation with trained feature weights: training of feature weights was performed on 20 training worms using the set of features from the *full* model for the 154 cell labels

Accuracy measurement	Median	μ	σ^2
Per-worm accuracy	0.77	0.77	0.0083
Per-cell accuracy	0.84	0.77	0.032

Note: The model included the scoring of unlabeled cells. The per-worm accuracy is computed for each worm using Equation (5). The table reports both the per-worm and the per-cell label accuracies, including their median, mean and variance in separate columns across the 25 test worms.

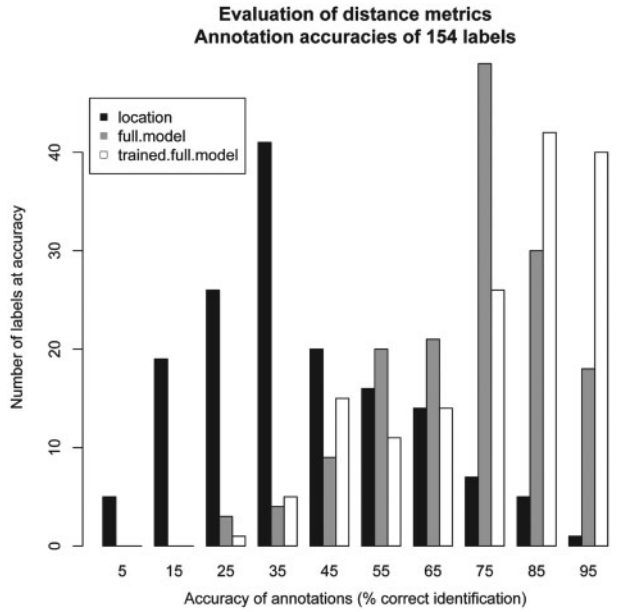

Fig. 4. Accuracy for annotation of 154 cells in adult worm images. In black, we show the accuracy using an untrained model considering location alone. The gray histogram gives the per-cell accuracy counts of the untrained model incorporating additional features. In white, we show the model with weights learned for these set of features and estimated means and variance for all cell labels, including 'other' cells

shows the strongest shift to the right. This demonstrates that the improvement in accuracy is not only on a per-worm basis, but also observed in a general improvement of individual cell label assignments. These accuracies are at a median of 35% for the model using location alone, and 84% (with a mean of 77%) for the trained model. In addition, five cells are given the correct label 100% of the time they are present in a worm. An additional 35 labels are correctly assigned in the 90th percentile.

3.3 Accuracy of cell division identification using network-flow formulation

Using the results of the cross-validated trained models described, we evaluated the accuracy of identifying cell divisions when they occur in the four intestinal cells. Among the 25 worms, 100 intestinal cells (4 per worm) are able to undergo additional cell divisions. We observed 54 events where one of these intestinal cells underwent the further division. Only 3 of the 25 worms had no additional cell divisions in all 4 of these intestinal cells. Table 3 summarizes the results for each intestinal cell. The four intestinal cells that are capable of dividing and giving rise to two daughter cells are the last two ventral cells in the intestine, Ventral9 and Ventral10, and the last two dorsal intestinal cells, Dorsal9 and Dorsal10. The posterior-most cells of this tissue in both the dorsal and ventral hemispheres are named Dorsal10 and Ventral10. The intestines just anterior to these two (Dorsal9 and Ventral 9) divide most frequently at 14 times each in the total of 25 worms.

Table 3. Results for 5-fold cross-validation identification of cell divisions of posterior intestinal cells

Parent cell name	Number of observed divisions	Correctly predicted state (%)
Ventral 9	14	72
Ventral 10	13	80
Dorsal 9	14	80
Dorsal 10	10	88

Note: The column indicated as correctly predicted state is calculated as $\frac{\text{True Positive} + \text{True Negative}}{\text{Total}}$.

Overall, the state of these cells is accurate 80% of the time. The dorsal cells receive high accuracies at 88 and 80%. The most challenging cell to predict is the ventral intestinal cell number 9 (Ventral9), which achieved 72% accuracy. It is important to note that the identification of a division is not necessarily indicative of the correct annotation. That is, although the two daughter labels are assigned within the worm, they are not necessarily assigned to the correct cells. However, use of these labels still serves an important purpose in understanding variability in the worm's development. In addition, identifying when cell divisions have not occurred prevents mis-annotation of the label to another cell when the actual cell is not present.

4 DISCUSSION

Creating automated techniques to annotate individual cells based on their unique cell labels in the organism *C.elegans* makes single-cell studies possible for non-experts and provides assistance for experts to perform analyses more rapidly. The manual curation of the automatically segmented cells can be performed in a few hours, for which the 154 cells can be rapidly annotated at high accuracy. In contrast, manual segmentation and annotation of the 154 cells in 3D for a well-trained biologist takes on the order of 2 days (X.Liu, personal communications). To truly enable widespread research of single cells in images, high-fidelity labeling of cells must be possible. This work demonstrates the potential of automatic techniques to succeed in the adult organism.

In *C.elegans*, existing approaches creating digital atlases in the developing embryo (Bao *et al.*, 2006) and larvae (L1) (Long *et al.*, 2008, 2009) proved to be poorly suited for the challenges of the adult worm as described earlier. In particular, the previous

state-of-the-art bipartite matching algorithm for labeling L1 worms failed in the adult for a number of reasons. The adult variability in the marker expression and exceptionally large number of germ line cells prevented the use of the same approach. Instead, we created a more complex model in which we learned the weights for a richer set of features, including cell characteristics of location and morphology. What is more, the bipartite matching approach used in the L1 was prohibitively slow for training a full adult model given the number of cells. As a result, learning feature weights required a new formulation using network flow, enabling the successful training of a more complex cost function.

We believe this work demonstrates the utility of using such a rich model to generate these high-confidence labels. The improvements in accuracy given in the results section provide evidence of the benefit of including morphological features in atlas-based modeling of *C.elegans*.

Future work might consider incorporating meta-features, such as posterior probabilities of a classifier that identifies tissue types, into the pipeline. Such an approach can mimic the behavior of the expert annotator who generally first identifies the tissue type of a cell (e.g. intestine) then assigns it the lineage-specific label, selecting from those available within its tissue type. Alternatively, incorporating such classifier probabilities directly into a cost function may result in a more flexible model. However, for the set of cells labeled in this work, the most salient features of the tissue were modeled directly in the scoring function (e.g. size for intestinal cells). In the future, training data will include cells from additional tissue types. At this time, inclusion of tissue classifiers might prove valuable, particularly in the case of neurons. These classifiers might distinguish small cells from over-segmented cell fragments, for example.

A confounding factor in the annotation process is the variability in the number of cells. For example, we identified a set of intestinal cells where we observed variable cell divisions. That is, at times they underwent an additional cell division, resulting in the presence of two daughter cells rather than the single parent identified in the 959 known somatic cells. Such variability cannot be properly represented by a traditional bipartite matching approach. This article presents a solution that through the construction of a special network structure for solving the annotation problem enables the selection of either the parent or the two daughters explicitly. Although achieving good accuracy, the current construction requires a post-processing step to identify the presence of a second intestinal-like cell. Future work might include developing a method that does not rely on the identification of a single additional cell in the division, but rather identifies either one large parent cell, or simultaneously both daughter cells.

We also observed an anterior intestinal cell, Ventral3, that divided just once in the 25 worms used in this work. This was not modeled owing to the infrequency of the cell division. However, this observation indicates that there is likely further variability that has not yet been observed. With increasing amounts of data, additional variability can be modeled explicitly to further improve cell annotation. What is more, it may be possible to model the co-occurrence of these cell divisions. Some weak evidence exists in this dataset indicating that two intestinal cells might be correlated in their division patterns. However, the relatively small amount of data makes it difficult to obtain statistical significance, and therefore a model that takes into consideration the co-occurring cell divisions is left for future work.

Finally, extending the annotator to include labels for more of the total 959 known cells will be most valuable. In addition to creating a more complete model of the worm, it will also improve overall annotation accuracy. We believe this work has provided evidence for such potential gains in the fidelity of automated cell labeling through the inclusion of more cell labels. In this work, we achieved an improvement in accuracy by modeling the cells that did not receive a unique cell lineage label, which we call the dud label. Yet, even the models including the duds could be further extended. There is significant variability within the set of unlabeled cells as it comprises many eggs, sperm cells, neurons, hypodermal cells and pharyngeal cells just to name a few. It is possible to create a larger set of dud labels with more homogeneous features representing the different subclasses within the unlabeled cells (e.g. the oocytes in the germ line). Therefore, the groups of duds can be mapped to their correct subtype.

In summary, we believe future work must focus on extending the annotation process by using more labels or by identifying additional subgroups to further improve accuracy. With more data, richer models can be built to account for cell division variability and inclusion of additional features. The ultimate goal is a larger model that labels a large proportion of all cells that are uniquely and reproducibly identifiable in the adult worm. This work represents the first step toward such a goal and provides a rich modeling approach capable of scaling with such extensions.

5 CONCLUSION

In this article, we present a method capable of annotating a set of single cells in images of adult *C.elegans* at a median accuracy of 84%. The work develops a novel framework for producing labels for 154 cells that is able to handle the additional challenges present in the adult worm that previous methods (created for earlier stages in the worm's development) are not able to handle. These challenges include the increase in the number of cells, and variability in cell location and cell divisions. We address these challenges through training a rich model that incorporates morphological and spatial features, constructing a special network structure and explicitly modeling cells that receive non-unique labels. By reducing the computational complexity in using a minimum-cost maximum-flow algorithm, we make feasible a cross-entropy–based learning algorithm to tune the weights of the features in our scoring function and ultimately train a more accurate model that is capable of handling the variable cell divisions. As a result, we demonstrate that the inclusion of additional features and the reformulation of the traditional approach to the label assignment make possible the training of a richer model to improve accuracy. Furthermore, we also demonstrate that inclusion of more cells, in addition to more features, leads to gains in accuracy for all cell label assignments.

ACKNOWLEDGEMENTS

The authors thank Batzoglou and Kim lab members and Eugene Myers for helpful discussions. S.J.A. drafted the manuscript, performed computational experiments, conceived of the original model and wrote the code. X.L. created, annotated and corrected

images and assisted in feature selection. C.B.D. co-wrote the manuscript and contributed code. S.J.A., C.B.D. and S.D.G. designed the model and experiments. S.J.A., X.L., A.N. and S.D.G. processed and curated image data. S.S.G. suggested the training methodology. S.D.G. assisted in image processing and selection of segmentation method and tuning parameters. F.L. and H.P. assisted in adapting L1 methods appropriately. S.S.K. and S.B. conceived of the project, participated in designing and coordinating the project and assisted with editing the manuscript. All authors read and approved the final manuscript.

Funding: This work was supported by a William R. Hewlett Stanford Graduate Fellowship, a National Science Foundation Fellowship and the Stanford Biomedical Informatics Training Grant from the National Library of Medicine (LM-07033) (to S.J.A.). This work was also funded in part by NSF grants (0640211 and 0312459) and NIH grant (GM091342-01).

Conflict of Interest: none declared.

REFERENCES

Altun,Z. and Hall,D. (2008) Handbook of C. elegans Anatomy. In: *WormAtlas.org*. http://www.wormatlas.org/hermaphrodite/hermaphroditehomepage.htm.

Bao,Z. *et al.* (2006) Automated cell lineage tracing in Caenorhabditis elegans. *Proc. Natl Acad. Sci. USA*, **103**, 2707–2712.

Caetano,T.S. *et al.* (2007) Learning graph matching. *IEEE 11th International Conference on Computer Vision 2007*, **0**, 1–8.

Caetano,T.S. *et al.* (2009) Learning Graph Matching. *IEEE Trans. Pattern Anal. Mach. Intelligence*, **31**, 1048–1058.

de Boer,P.T. *et al.* (2005) A Tutorial on the cross-entropy method. *Ann. Oper. Res.*, **134**, 19–67.

Dezs,B. *et al.* (2011) LEMON–an open source C++ graph template library. *Electron Notes Theor Comput Sci*, **264**, 23–45.

Edmonds,J. and Karp,R.M. (1972) Theoretical improvements in algorithmic efficiency for network flow problems. *J. ACM*, **19**, 248–264.

Fowlkes,C.C. *et al.* (2008) A quantitative spatiotemporal atlas of gene expression in the *Drosophila* blastoderm. *Cell*, **133**, 364–374.

Keränen,S.V.E. *et al.* (2006) Three-dimensional morphology and gene expression in the *Drosophila* blastoderm at cellular resolution II: dynamics. *Genome Biol.*, **7**, R124.

Király,Z. and Kovács,P. (2010) An experimental study of minimum cost flow algorithm. In: *8th International Conference on Applied Informatics. Eger, Hungary, 2010*.

Kleinschmidt,P. and Schannath,H. (1995) A strongly polynomial algorithm for the transportation problem. *Math. Program.*, **68**, 1–13.

Le,Q. and Smola,A. (2007) Direct optimization of ranking measures. *Mach. Learn.*, arXiv:0704.3359.

Li,G. *et al.* (2007) 3D cell nuclei segmentation based on gradient flow tracking. *BMC Cell Biol.*, **8**, 40.

Liu,X. *et al.* (2009) Analysis of cell fate from single-cell gene expression profiles in *C. elegans*. *Cell*, **139**, 623–633.

Long,F. *et al.* (2008) Automatic Recognition of Cells (ARC) for 3D images of C. elegans. In: Vingron,M. and Wong,L. (eds) *Research in Computational Molecular Biology, Volume 4955 of Lecture Notes in Computer Science*. Springer, Berlin/Heidelberg, pp. 128–139.

Long,F. *et al.* (2009) A 3D digital atlas of *C. elegans* and its application to single-cell analyses. *Nat. Meth.*, **6**, 667–672.

Luengo Hendriks,C. *et al.* (2006) Three-dimensiondal morphology and gene expression in the *Drosophila blastoderm* at cellular resolution I: data acquisition pipeline. *Genome Biol.*, **7**, R123.

Murray,J.I. *et al.* (2008) Automated analysis of embryonic gene expression with cellular resolution in *C. elegans*. *Nat. Methods.*, **5**, 703–709.

Petterson,J. *et al.* (2009) *Exponential Family Graph Matching and Ranking*. *CoRR* 2009, **abs/0904.2623**.

Qu,L. *et al.* (2011) Simultaneous recognition and segmentation of cells: application in *C. elegans*. *Bioinformatics*, **27**, 2895–2902.

Rubinstein,R. and Kroese,D.P. (2004) *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. Springer-Verlag, New York.

Taskar,B. (2004) Learning structured prediction models: a large margin approach. *PhD thesis*, Stanford University, CA.

Taskar,B. *et al.* (2005) Learning structured prediction models: a large margin approach. In: *Proceedings of the 22nd international conference on Machine learning*. ICML '05, New York, NY, USA: ACM. 896–903.

Tsochantaridis,I. *et al.* (2005) Large margin methods for structured and interdependent output variables. *J. Mach. Learning Res.*, **6**, 1453–1484.

Zhao,Z. *et al.* (2008) Comparative analysis of embryonic cell lineage between *Caenorhabditis briggsae* and *Caenorhabditis elegans*. *Dev. Biol.*, **314**, 93–99.