Graphical-Model Framework for Automated Annotation of Cell Identities in Dense Cellular Images

Authors

Shivesh Chaudhary[1], Sol Ah Lee[1], Yueyi Li[1], Dhaval S. Patel[1], and Hang Lu[1,2*]

Affiliations

[1] School of Chemical & Biomolecular Engineering, Georgia Institute of Technology, Atlanta, Georgia, United States of America

[2] Petit Institute for Bioengineering and Bioscience, Georgia Institute of Technology, Atlanta, Georgia, United States of America

* Correspondence should be addressed to HL: hang.lu@gatech.edu.

Abstract

Although identifying cell names in dense image stacks is critical in analyzing functional whole-brain data enabling comparison across experiments, unbiased identification is very difficult, and relies heavily on researchers' experiences. Here we present a probabilistic-graphical-model framework, CRF_ID, based on Conditional Random Fields, for unbiased and automated cell identification. CRF_ID focuses on maximizing intrinsic similarity between shapes. Compared to existing methods, CRF_ID achieves higher accuracy on simulated and ground-truth experimental datasets, and better robustness against challenging noise conditions common in experimental data. CRF_ID can further boost accuracy by building atlases from annotated data in highly computationally efficient manner, and by easily adding new features (e.g. from new strains). We demonstrate cell annotation in *C. elegans* images across strains, animal orientations, and tasks including gene-expression localization, multi-cellular and whole-brain functional imaging experiments. Together, these successes demonstrate that unbiased cell annotation can facilitate biological discovery, and this approach may be valuable to annotation tasks for other systems.

Introduction

Annotation of anatomical structures at cellular resolution in large image sets is a common data analysis step in many studies in *C. elegans* such as gene expression pattern analysis[1,2], lineage tracing[3], multi-cell calcium imaging and whole-brain imaging[4–7]. It is necessary for cellular resolution comparison of data across animals, trials, and experimental conditions. Particularly in whole-brain functional imaging, meaningful interpretation of population activity critically depends on cell identities as they facilitate the incorporation of existing knowledge about the system[5]. Cell identities are also needed for applying common statistical data analysis methods such as Principal Component Analysis, Tensor Component Analysis, demixed-Principal Component Analysis[8,9] *etc*. as data across experiments needs to be indexed and pooled by cell identities before applying these methods.

While accurate annotation of cell identities in images is critical, this task is difficult. Typically, the use of cell-specific markers as landmarks delivers good accuracy but has the cost of having to engineer cell specific reagents without interfering with phenotypes of interest, which is not guaranteed. Further, even with markers such as the recently developed impressive reagents in the NeuroPAL collection[10], there is a still need to automate the cell identification process. In the absence of markers, cells are identified by comparing images to a reference atlas such as WormAtlas[11] and OpenWorm[12] atlas. However, there are severe limitations from both using reference atlas and the presence of noise in data. Reference atlases assume a static and often single view of the anatomy; in contrast, anatomical features vary across individuals. Moreover, due to variations in experimental conditions during acquisition such as exact

resolution and orientation of animals, image data often do not match the static atlases, making manual cell identification extremely difficult if not infeasible. Separately, two kinds of noise are prevalent in data. First, individual-to-individual variability in cell positions compared to positions in atlas (position noise). Second, mismatch between number of cells in image and atlas (count noise). Count noise is primarily caused by variability in the expression levels of the reporter used to label cells across animals (i.e. mosaicism), incomplete coverage of promoter to label desired cells, and limits in the computational methods to detect cells. In each of these cases, fewer cells are detected in the images than cells in atlas. Empirical data have shown that in normalized coordinates, a cell's position can deviate from atlas position by more than the cell's distance to its tenth' nearest neighbor in the image[10,13]. Further, our data, as well data from other labs, have shown that 30-50% of cells in atlas may be missing from images[5–7]. As a result of large position and count noise common in data, identifying densely packed cells in head ganglion images of *C. elegans* by manually comparing images to the atlas is extremely difficult, even for experienced researchers. Further, manual annotation is labor intensive. Therefore, there is a critical need for automated methods for cell identification.

Previous computational methods for cell identification in *C. elegans* images[1,14–16] focused on identifying sparsely distributed cells with stereotypical positions in young larvae animals. Tools for identification of cells in whole-brain datasets, i.e. in dense head ganglion, do not exist. Further, previous methods[1,10,15–18] do not explicitly address the challenges imposed by the presence of position and count noise in data. All previous methods are either registration-based or formulate a linear assignment problem; objective functions in these methods minimize a first-order constraint such as the distances between cell specific features in images

76 and atlas. Thus, these methods maximize only extrinsic similarity[19] between images and atlas,
77 which is highly sensitive to count noise, position noise, and pre-alignment of spaces in which
78 the image and the atlas exist (i.e. orientations of animals in image and atlas). With the amount
79 of position and count noise commonly observed in experimental data, registration-based
80 methods produce large matching errors.

81 An alternative criterion proposed for topology invariant matching of shapes is to
82 maximize intrinsic similarity[19,20] orthogonal to extrinsic similarity. This approach has advantages
83 because noise that affects extrinsic similarity does not necessarily imply worse intrinsic
84 similarity. For instance, although cell positions in an image may deviate from their positions in
85 the atlas (large extrinsic noise), geometrical relationships among them are largely maintained
86 (low intrinsic noise). As a specific example, although absolute positions of the cell bodies of
87 AIBL and RIML in an image may deviate greatly from their atlas positions, AIBL stays anterior to
88 cell RIML. Intrinsic similarity is more robust against noises, independent of the pre-alignment of
89 spaces, and inherently captures dependencies between cell label assignments that registration
90 methods do not consider.

91 To directly optimize for intrinsic similarity and dependencies between label assignment,
92 we cast the cell annotation problem as a Structured Prediction Problem[21–24] and build a
93 Conditional Random Fields (CRF) model[25] to solve it. The model directly optimizes cell-label
94 dependencies by maximizing intrinsic and extrinsic similarities between image and atlas. One
95 major advantage, as shown using both synthetic data with realistic properties (e.g. statistics
96 from real data) and manually annotated experimental ground-truth datasets, is that CRF_ID
97 achieves higher accuracy compared to existing methods. Further, CRF_ID outperforms existing

methods in handling both position noise and count noise common in experimental data across all challenging noise levels.

To further improve accuracy, we took two approaches. First, we take advantage of spatially distributed (fluorescently labeled) landmark cells. These landmark cells act as additional constraint on the model thus aiding in optimization and help in pre as well post prediction analysis. Second, we developed a methodology to build data-driven atlases that capture the statistics of the experimentally observed data for better prediction. We provide a set of computational tools for automatic and unbiased annotation of cell identities in fluorescence image data, and efficient building of data-driven atlases using fully or partially annotated image sets. We show the utility of our approach in several contexts: determining gene expression patterns with no prior expectations, tracking activities of multiple cells during calcium imaging, and identifying cells in whole-brain imaging videos. For the whole-brain imaging experiment, our annotation framework enabled us to analyze the simultaneously recorded response of *C. elegans* head ganglion to food stimulus and identify two distinct groups of cells whose activities correlated with distinct variables – food sensation and locomotion.

Results

Cell annotation formulation using structured prediction framework

Our automated cell annotation algorithm is formulated using Conditional Random Fields (CRF). CRF is a graphical model-based framework widely used for structured/relational learning tasks in Natural Language Processing and Computer Vision community[21,22,25,26]. The goal of

118 structured learning tasks is to predict labels for structured objects such as graphs. In our
119 neuron annotation problem, we assume that our starting point is a 3D image stack of the *C.*
120 *elegans* head ganglion (Figure 1A(i)) in which neurons have been already detected (Figure
121 1A(ii)), either manually or by automated segmentation, and we want to match each neuronal
122 cell body or nucleus to an identity label (a biological name). Hence, we have N detected
123 neuronal cell bodies $\{x_1, \dots, x_N\}$ that form the set of observed variables $\mathbf{x} = \{x_i\}_1^N$, and their 3D
124 coordinates, $p_i \in \mathbb{R}^3, i \in \{1, \dots, N\}$. We also have a neuron atlas that provides a set of labels
125 $\mathcal{L} = \{l_1, \dots, l_K\}$ (biological names) of the neurons and positional relationships among them.
126 Note that the number of neurons in the atlas is greater than the number of neurons detected in
127 the image stack in all datasets, i.e. $K > N$. The goal is to annotate a label $y_j \in \mathcal{L}$ to each neuron
128 in the image stack. The problem is similar to structured labeling[22] since the labels to be assigned
129 to neurons are dependent on each other. For example, if a certain neuron is assigned label
130 AVAL, then the neurons that can be assigned label RMEL become restricted since only the cells
131 anterior to AVAL can be assigned RMEL label.

132

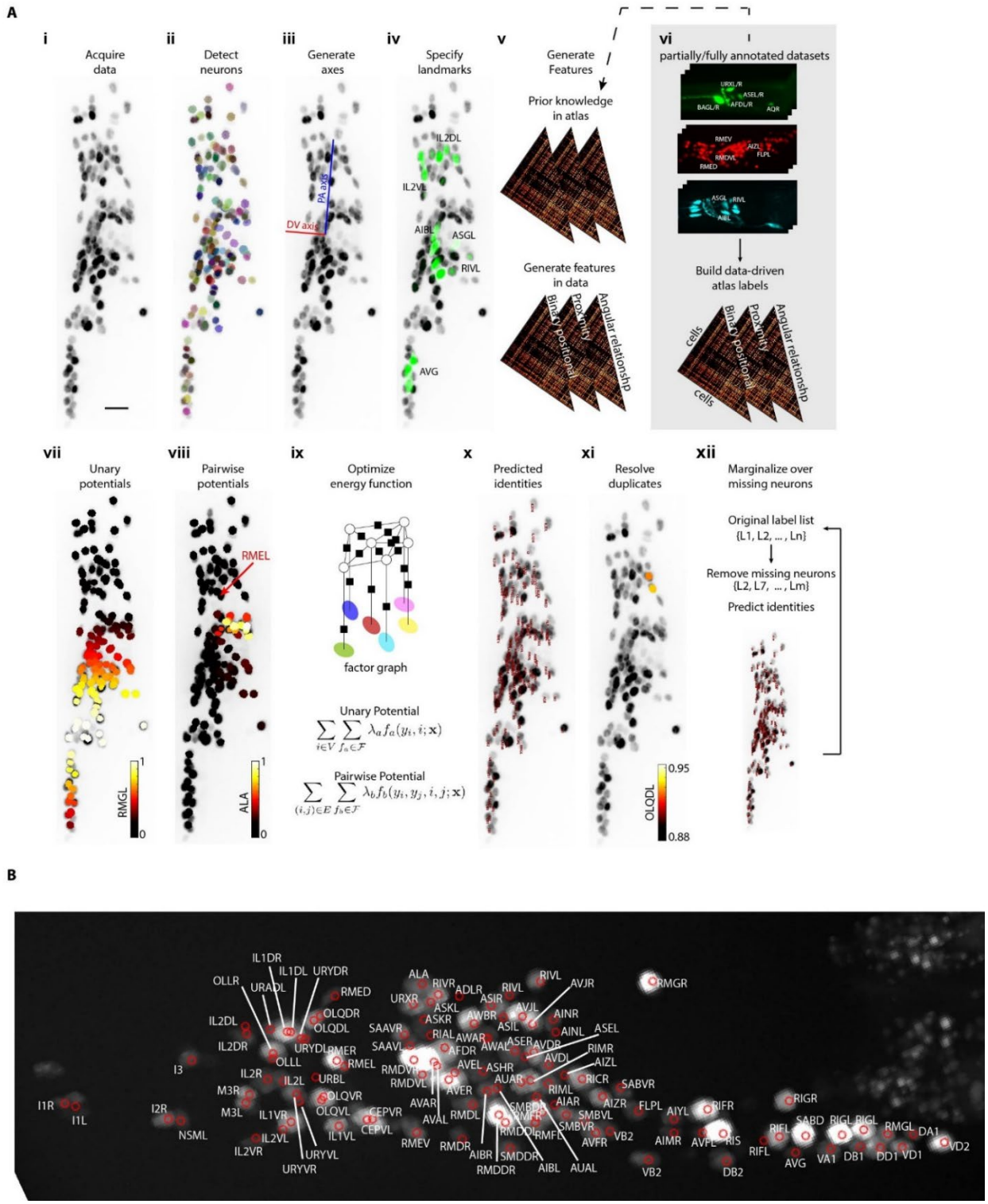


Figure 1. ***CRF_ID annotation framework automatically predicts cell identities in image stacks.***

A) Steps in CRF_ID framework applied to neuron imaging in C. elegans*. i) Max-projection of a 3D image stack showing head ganglion neurons whose biological names (identities) are to be determined. ii) automatically detected cells (Methods) shown as overlaid*

colored regions on the raw image. iii) coordinate axes are generated automatically (Note S1). iv) identities of landmark cells if available are specified. v) unary and pairwise positional relationship features are calculated in data. These features are compared against same features in atlas. vi) atlas can be easily built from fully or partially annotated dataset from various sources using the tools provided with framework. vii) an example of unary potentials showing the affinity of each cell taking the label RMGL. viii) an example of dependencies encoded by pairwise potentials, showing the affinity of each cell taking the label ALA given the arrow-pointed cell is assigned the label RMEL. ix) identities are predicted by simultaneous optimization of all potentials such that assigned labels maximally preserve the empirical knowledge available from atlases. x) predicted identities. xi) duplicate assignment of labels is handled using a label consistency score calculated for each cell (Appendix – extended methods S1). xii) the process is repeated with different combinations of missing cells to marginalize over missing cells (Note S1). Finally, top candidate label list is generated for each cell.

B) An example of automatically predicted identities (top picks) for each cell.

We use CRF-based formulation to directly optimize for such dependencies and automatically assign names to each cell. Briefly, a node v_i is associated with each observed variable x_i (i.e. segmented neuron in image data) forming the set of variables $V = \{v_i\}_1^N$ in the model. Then, CRF models a conditional joint probability distribution $P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right)$ over product space $\boldsymbol{\mathcal{Y}} = \mathcal{Y}_1 \times \ldots \times \mathcal{Y}_N$ of labels assigned to V given observations $\mathbf{x}$, where each $\mathcal{Y}_i = \mathcal{L}, i \in \{1, \ldots, N\}$ and $\mathrm{y} \in \boldsymbol{\mathcal{Y}}$ is a particular assignment of labels to V. $\boldsymbol{\mathcal{Y}}$ contains non-optimal and optimal assignments. In CRF, label dependencies among various nodes are encoded by the structure of an underlying undirected graph $\mathrm{G}(V, \mathcal{C})$ defined over nodes V, where $\mathcal{C}$ denotes the set of cliques in graph G. With the underlying graph structure, the conditional joint probability distribution $P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right)$ over full label space $\boldsymbol{\mathcal{Y}}$ factorizes over cliques in G, making it tractable to the model:

$$P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \Phi_c(y_c; \mathbf{x}) \tag{1}$$

Here Z is the normalization constant $Z = \sum_{\mathrm{y} \in \boldsymbol{\mathcal{Y}}} \prod_{c \in \mathcal{C}} \Phi_c(y_c; \mathbf{x})$ and Φ_c denotes clique potential if nodes in clique c are assigned label $y_c \in \boldsymbol{\mathcal{Y}_c} = \mathcal{Y}_1 \times \ldots \times \mathcal{Y}_c$. In our model, the fully connected graph structure considers only pairwise dependencies between every pair of neurons. Thus, the graph structure of our model becomes $\mathrm{G}(V, \mathcal{E})$, with the node set V containing nodes v_i associated with each segmented neuron and pairwise edges between all nodes that form the edge set $\mathcal{E}$. The potential functions in our model are node-potentials Φ_i and edge-potentials Φ_{ij}. These functions are parameterized with unary feature functions $f_a: \mathbf{x} \times \mathcal{Y}_i \to \mathbb{R}$ and pairwise feature functions $f_b: \mathbf{x} \times \mathcal{Y}_i \times \mathcal{Y}_j \to \mathbb{R}$ respectively:

$$\Phi_i(m; i, \mathbf{x}) = \exp\left(\sum_a \lambda_a f_a(m; i, \boldsymbol{x})\right) \tag{2}$$

$$\Phi_{ij}(m, n; i, j, \mathbf{x}) = \exp\left(\sum_b \lambda_b f_b(m, n; i, j, \boldsymbol{x})\right) \tag{3}$$

173 where $m \in \mathcal{L}$ and $n \in \mathcal{L}$ are labels in atlas. Note, there are a unary features and b pairwise
174 features with weights λ_a and λ_b respectively to define node and edge potentials.

175 While unary features account for extrinsic similarity and cell specific features, pairwise
176 features account for intrinsic similarity. To maximize accuracy, we encode pairwise dependencies
177 between all pairs of cells in the form of several geometrical relationship features (Figure 1 – figure
178 supplement 1). Optimal identities of all neurons $\mathrm{y} \in \boldsymbol{\mathcal{Y}}$ is obtained by maximizing the joint-
179 distribution $P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right)$. This is equivalent to maximizing the following energy function.

$$y = \arg max_{y \in \mathcal{Y}} \sum_{i \in V} \sum_a \lambda_a f_a(m; i, \boldsymbol{x}) + \sum_{e_{ij} \in \mathcal{E}} \sum_b \lambda_b f_b(m, n; i, j, \boldsymbol{x}) \tag{4}$$

180 Optimizing this energy function over fully connected graphs (more specifically graphs with
181 loops) is known to be an NP-hard problem[27]. However, approximate inference algorithms are
182 widely used in CRF community as they provide reasonable solutions. We implemented a
183 popular method called Loopy Belief Propagation[28] to infer the most probable labelling over all
184 cells, as well as marginal distributions of label assignments for each cell.

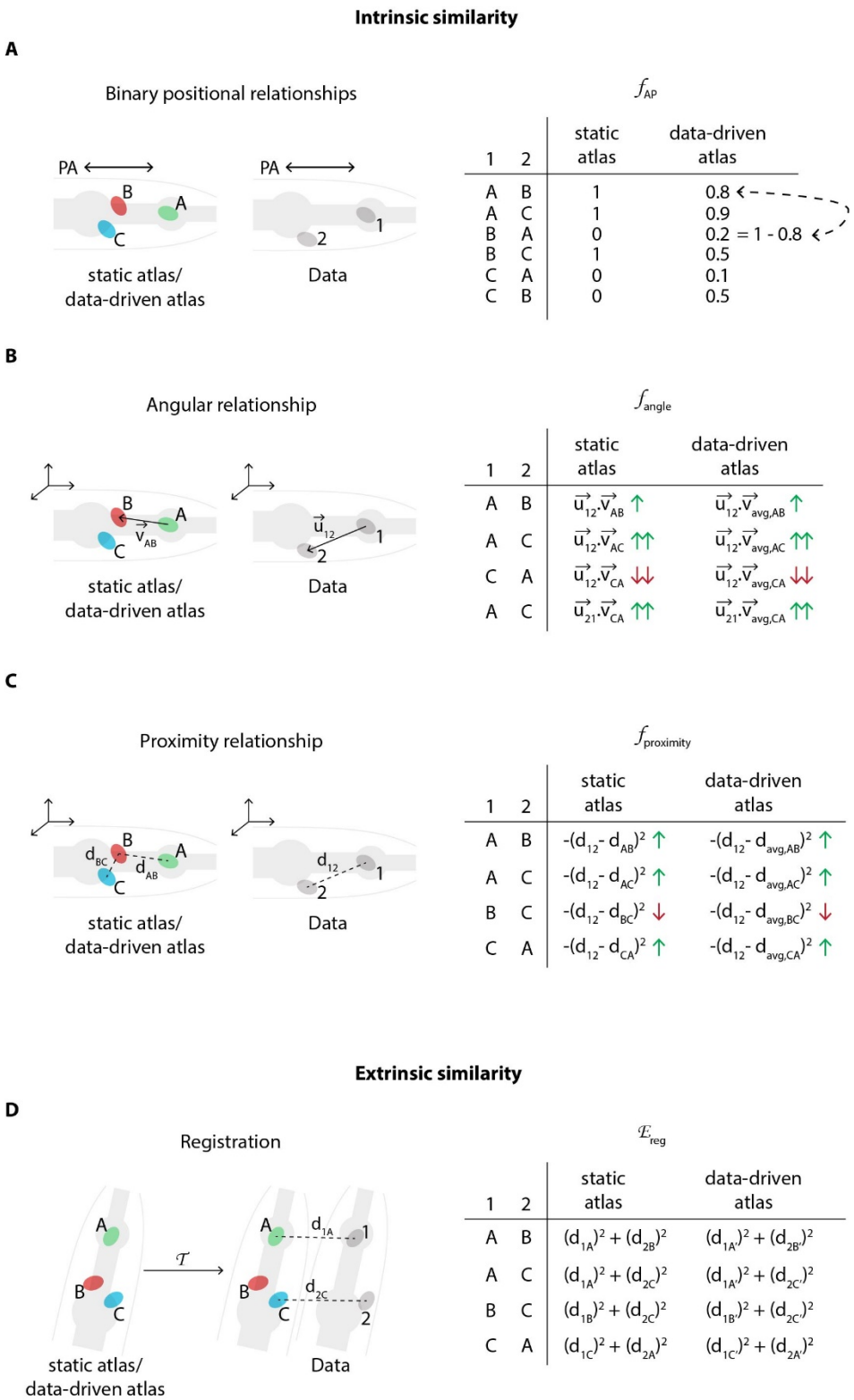


Figure 1 - figure supplement 1. ***Schematic description of various features in the CRF model that relate to intrinsic similarity and extrinsic similarity.***

A) An example of binary positional relationship feature (Appendix – extended methods S1.2.2) illustrated for positional relationships along AP axis. The table lists feature value for some exemplary assignment of labels 'A', 'B', and 'C' from the atlas to cells '1' and '2' in the image data. E.g. since cell '1' is anterior to cell '2' in image, if labels assigned to these cells are consistent with the anterior-posterior positional relationship (e.g. 'A-B', 'A-C', 'B-C'), then the feature value is high (1); else low (0). CRF_ID model assigns identities to cells in image by maximizing the feature values for each pair of cells in image over all possible label assignments. The table also illustrates the difference between using a static atlas (or single data source) and a data-driven atlas built using available annotated data. In case of static atlas, the CRF model assumes that the cell 'A' is anterior to cell 'B' with 100% probability. In contrast, in experimental data cell 'A' may be anterior to cell 'B' with 80% probability (8 out of 10 datasets) and cell 'B' may be anterior to cell 'C' with 50% probability (5 out of 10 datasets). Thus, data driven atlases relaxes the hard constraint and uses statistics from experimental data. The feature values are changed accordingly. Note, unlike registration based methods for building data-driven atlas, in CRF model data-driven atlases record only probabilistic positional relationship among cells and not probabilistic positions of cells. Thus CRF_ID does not build spatial atlas of cells.

B) An example of angular relationship feature (Appendix – extended methods S1.2.4). The table lists feature value for some exemplary assignment of labels. For example, the feature value is highest for assigning labels 'A' and 'C' to cells '1' and '2'' because the vector joining cells 'A' and 'C' in atlas ($\overrightarrow{v_{AC}}$) is most directionally similar to vector joining cells '1' and '2' in image ($\overrightarrow{u_{12}}$) as measured by dot product of vectors. For data-driven atlas, average vectors in atlas are used.

C) An example of proximity relationship feature (Appendix – extended methods S1.2.3). The table lists feature value for some exemplary assignment of labels. E.g. the feature value is low for assigning labels 'B' and 'C' to cells '1' and '2'' because the distance between cells 'B' and 'C' in atlas (d_{BC}) is least similar to distance between cells '1' and '2' in image (d_{12}). The distance metric can be Euclidean distance or geodesic distance. For data-driven atlas, average distances in atlas are used.

D) An example illustrating the cell annotation performed by maximizing extrinsic similarity in contrast to intrinsic similarity. Registration based methods maximize extrinsic similarity by minimizing registration cost function $\mathcal{E}_{reg}$. Here a transformation $\mathcal{T}$ is applied to the atlas and labels are annotated to cells in image by minimizing the assignment cost i.e. sum of distances between cell coordinates in image and transformed coordinates of cells in atlas. For data driven atlas, a spatial atlas is built using annotated data that is used for registration. Note, in contrast, CRF_ID method does not build any spatial atlas of cells because it uses intrinsic similarity features. CRF_ID only builds atlases of intrinsic similarity features shown in panels A-C.

221 The features used in the base version of the model are geometrical relationship features
222 that ensure identities assigned to cells in image are consistent with the atlas in terms of
223 satisfying pairwise geometrical relationships. These features include binary positional
224 relationship feature, proximity relationship feature, and angular relationship feature (Figure 1 –
225 figure supplement 1, Appendix – extended methods S1.2). All these features are a variant of the
226 quadratic Gromov-Wasserstein distance used in matching metric spaces[20,29] and shapes[30,31].
227 Briefly, binary positional relationship features encode that as an example, if cell i is anterior,
228 dorsal and to the right of cell j in image stack, then identities assigned to these cells should
229 satisfy these relationships in the atlas. Proximity relationship features ensure that if cell i is
230 spatially near to cell j in image stack, then identities of spatially distant cells in atlas would not
231 be assigned to these cells. Finally, angular relationships ensure that identities assigned to cells i
232 and j should satisfy fine-scale directional relationships as well, and not just simple binary
233 relationships. We show that the CRF model can be easily updated to include additional features
234 such as cells with known identities (landmark cells) and fluorescent spectral codes of cells. We
235 demonstrate this by incorporating landmark cells and spectral information of cells in the model
236 and show improvement in accuracy.

237 A critical component for the success of automated cell identification methods is data-
238 driven atlas. Static atlases such as OpenWorm atlas provide a single observation of positional
239 relationships among cells. For instance, if cell RMEL is to the left of cell AVAL in OpenWorm
240 atlas, then the model assumes that RMEL is to the left of AVAL with 100% probability. In
241 contrast, in observed experimental data RMEL may be observed to be left of AVAL with 80%
242 probability (e.g. in 8 out of 10 annotated experimental datasets). Thus, data-driven atlases relax

243 the hard-coded constraint of 100% probability imposed by static atlas and accounts for the
244 statistics that is observed experimentally for all positional relationship features (Figure 1 –
245 figure supplement 1). Note, data-driven atlas built in our framework is considerably different
246 from those built by registration-based methods. While the latter atlases store probabilistic
247 positions of cells, atlases built by our framework store only probabilistic pairwise positional
248 relationship features among cells, thus more generalizable. We show that building such data-
249 driven atlases is easy for our CRF model (Appendix – extended methods S1.7). We demonstrate
250 this by building several data-driven atlases from different data sources containing various
251 features, showing considerable improvement in accuracy. Further, building data-driven atlases
252 is computationally cheap in CRF_ID, requiring only simple averaging operations; thus, it is
253 scalable to build atlases from large-scale annotated data that may become available in future.

254 Computational workflow for automatic cell identification

255 Our annotation framework consists of four major steps (Figure 1A; Appendix – extended
256 methods S1). First, cells are automatically detected in input image channels using a Gaussian
257 Mixture-based segmentation method (see Materials and Methods – Whole-brain data analysis).
258 Cells with known identities (landmarks cells) are also detected in this step and their identities
259 are specified. We designed the framework to be flexible on several fronts: 1) easily using
260 manual segmentations of image channels or segmenting on the run; 2) integrating landmark
261 information from any number of image channels; 3) specifying identities of landmark cells on
262 the run or from existing fully or partially annotated files generated with other tools such as
263 Vaa3D[32]. In the second step, a head coordinate is generated by solving an optimization problem

with considerations of the directional consistency of axes (see Appendix – extended methods S1.3). With this coordinate system, we next define cell specific features (unary potentials) and co-dependent features (pairwise potentials) in the data (Figure 1 – figure supplement 2A, B). The base version of the model uses only pairwise relationship features for all pairs of cells, including binary positional relationships, angular relationship, and proximity relationship between cells in images (Figure 1 – figure supplement 1). However, additional unary features such as landmarks and color information can be easily added in the model. By encoding these features among all pairs of cells, our fully connected CRF model accounts for label dependencies between each cell pair to maximize accuracy. The atlas used for prediction may be a standard atlas such as the OpenWorm[12] atlas or it can be easily built from fully or partially annotated datasets from various sources using the tools provided with our framework (see Appendix – extended methods S1.7). In the third step, identities are automatically predicted for all cells by optimizing the CRF energy function consisting of unary and pairwise potentials, which in our formulation is equivalent to maximizing the intrinsic similarity between data and atlas (see Appendix – extended methods S1.4). Duplicate assignments are resolved by calculating a label-consistency score for each neuron, removing duplicate assignments with low scores (Figure 1 – figure supplement 2C, D, see Appendix – extended methods S1.5) and re-running the optimization. After the third step, the code outputs top predicted label for each cell. Next, an optional fourth step can be performed to account for missing neurons in image stack. In this step, full atlas is subsampled to remove fixed number of labels from atlas by either sampling uniformly or based on prior confidence values available on missing rate of labels in images (see Appendix – extended methods S1.6). Subsampled atlas assumes that labels

removed are missing from image and thus ensures that those labels cannot be assigned to any cell in image. The sampling procedure is repeated, and identities are predicted in each run. We perform these runs in parallel on computing clusters. Lastly, identities predicted across each run are pooled to generate top candidate identities for each cell (Figure 1B; Figure 1 – video supplement 1; see Appendix – extended methods S1.6). Thus, there are two modes of running the framework – single-run mode that outputs only top label for each cell and parallel-run mode that outputs multiple top candidate labels for each cell. We make the software suite freely available at https://github.com/shiveshc/CRF_Cell_ID.

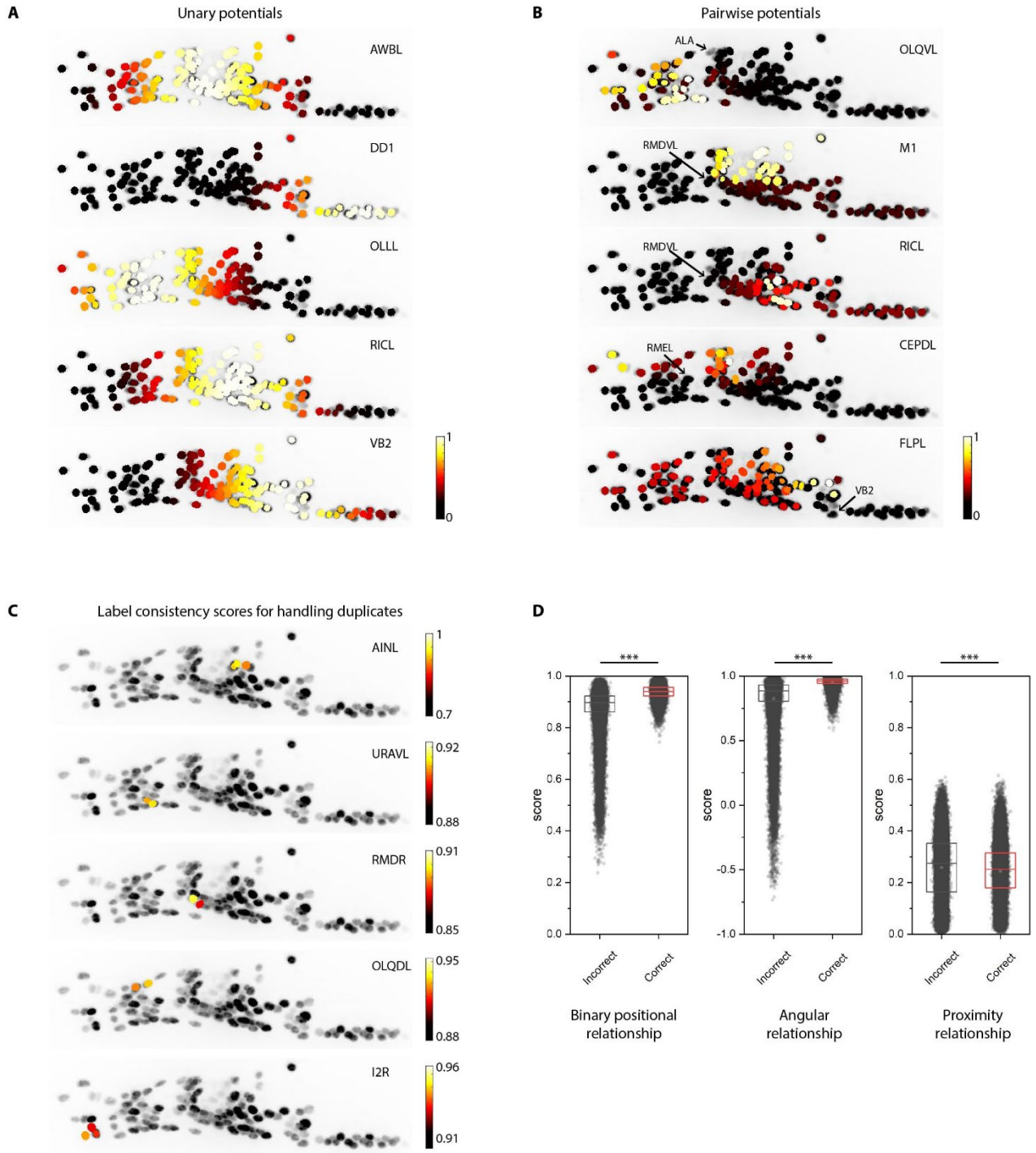


Figure 1 - figure supplement 2. ***Additional examples of unary and pairwise potentials and label consistency scores calculated for each cell.***

A) *Unary potentials encode affinities of each cell to take specific labels in atlas. Here affinities of all cells to take the label specified on the top right corner of images are shown. Randomly selected examples are shown here. In practice, unary potentials are calculated for all cells for every label.*

B) Pairwise potentials encode affinities of pair of cells in head ganglion to get two labels from atlas. Here we show the affinity of all cells taking the label specified on the top right corner of images given the cell marked by the arrow is assigned the given label. Randomly selected examples are shown here. In practice, pairwise potentials are calculated for all pairs of cells for all pairs of labels.

C) Examples of label-consistency score of cells that were assigned duplicate labels (specified on the top right corner of the image) in an intermediate step in framework. To remove duplicate assignments, only the cell with the highest consistency score is assigned the label. Optimization is run again to assign labels to all unlabeled cells while keeping the identities of labeled cells fixed.

*D) Comparison of label-consistency scores for accurately predicted cells and incorrectly predicted cells. Correctly predicted cells have a higher binary positional relationship consistency score, close to 1 angular relationship consistency score (smaller angular deviation between labels in image and atlas) and close to 0 proximity consistency score (smaller Gromov-Wasserstein discrepancy). Scores shown for all 130 predicted cells in synthetic data across ~1100 runs. Thus, $n \approx 150{,}000$. *** denotes $p < 0.001$, Bonferroni paired comparison test. Each run differed from the other in terms of random position noise and count noise applied to synthetic data to mimic real images. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.*

Prediction accuracy is bound by position and count noise in data

Given the broad utility of image annotation, we envision our workflow to apply to a variety of problems where experimental constraints and algorithm performance requirements may be diverse. For example, experimental data across different tasks inherently contains noise contributed from various sources in varying amounts that can affect annotation accuracy. These sources of noises include the following: 1) deviation between cell positions in images and positions in atlas, i.e. position noise, 2) smaller count of cells in images than number of cells in atlas due to missing cells in images, i.e. count noise, and 3) absence of cells with known identities, i.e. known landmarks. We set out to determine general principles of how these noises may affect cell identification accuracy across various tasks. We used two different kinds of data: synthetic data generated from OpenWorm 3D atlas[12] (Figure 2 – figure supplement 1A, B and Figure 2 – figure supplement 2) and experimental data generated using NeuroPAL strains[10], consisting of annotated ground-truth of 9 animals with ~100 uniquely identified neurons (Figure 2 – figure supplement 3). While experimental data enables the assessment of prediction accuracy in real scenarios, synthetic data enable us to tune the amount of noise contributed from various sources and dissect their effects on accuracy independently.

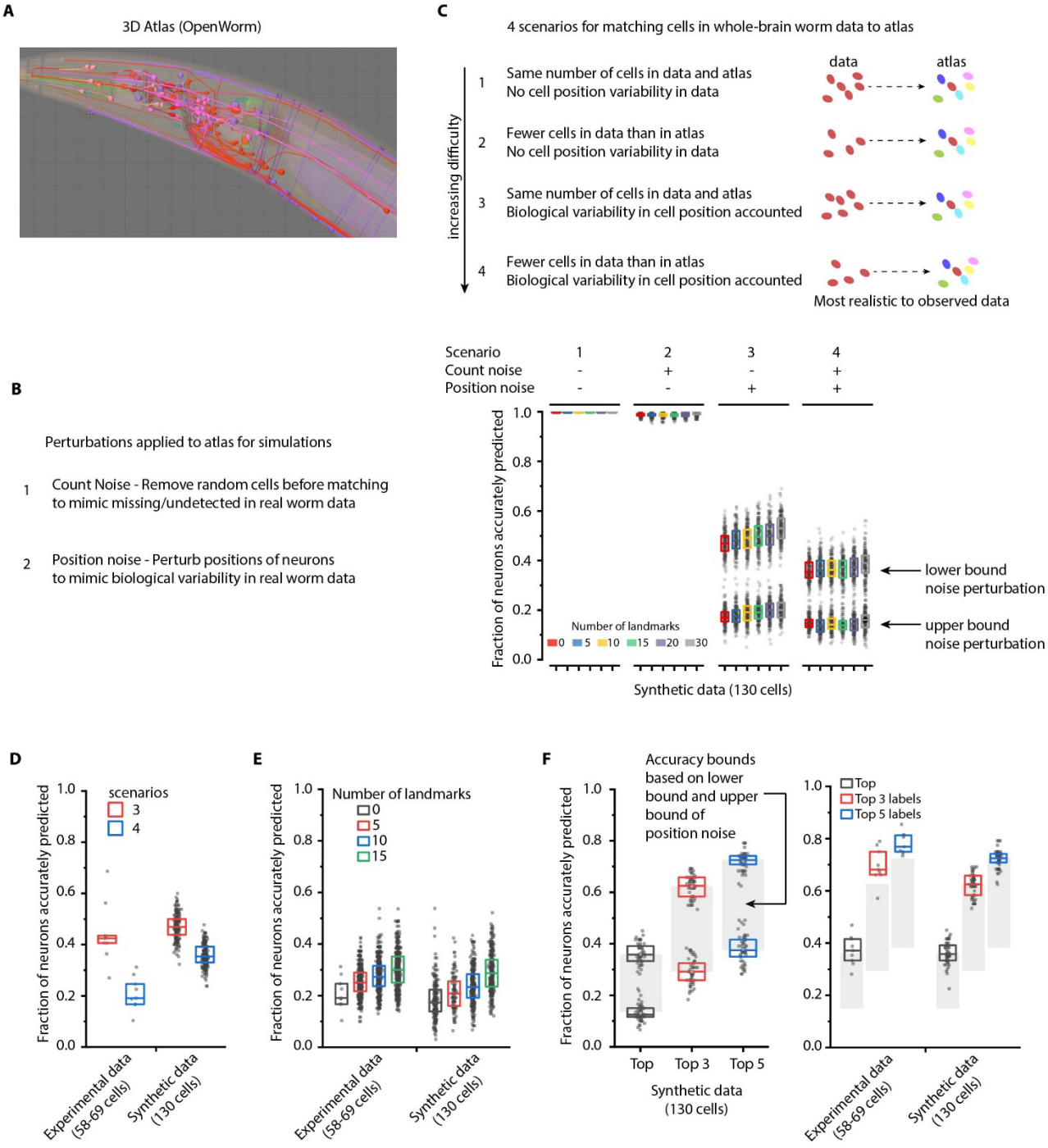


Figure 2 - figure supplement 1. ***Performance characterization using synthetic data***

A) Freely available open-source 3D atlas (OpenWorm atlas) was used to generate synthetic data.

B) 4 scenarios were simulated using atlas and prediction accuracies were quantified. These scenarios include different perturbations observed in experimental data such as count noise (discrepancy between cells in images and atlas), and position noise (variability in cell positions compared to atlas).

C) Prediction accuracy of framework for 4 simulated scenarios. For the scenario 1, in which no position noise or count noise perturbation was applied to generate simulated data, CRF_ID framework predicted identities with 99.9%

accuracy thus highlighting that identities can be predicted by using relative positional relationships only (without any information about absolute positions of cells) in CRF_ID framework. With application of position and count noise, prediction accuracy decreased (scenario 2-4). Two levels of box plots for scenario 3 and scenario 4 show prediction results for lower bound and upper bound levels of position noise applied to cells (see Figure 2 – figure supplement 2). n = 200-203 runs for each scenario and for each number of landmarks condition. Each run differed from another with respect to 1) random perturbations applied to positions of cells 2) random combination of landmarks selected in each run 3) random combination of cells set as missing. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

D) Scenarios 3 and 4 were simulated for experimental data by applying count noise to data and results were compared to results on synthetic data shown in panel C. Results show good match between synthetic data and experimental ground truth data. For synthetic data simulations, results are same as panel C for 0 landmarks condition and lower bound level of perturbation. Experimental data comes from OH15495 strain. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

E) Effect of number of landmarks on prediction accuracy using both experimental data and synthetic data show similar accuracy trends. Images of 9 animals were used as experimental ground truth datasets (strain OH15495). For no landmarks condition, n = 9. For non-zero landmarks conditions, n = 450 runs across 9 datasets were performed for each condition with randomly selected landmarks in each run. For synthetic data, n = 200 runs were performed for each condition with randomly selected landmarks in each run. For no landmarks condition in synthetic data – each run is different from another only with respect to random perturbations applied in each run. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

F) Accuracy bounds for Top, Top 3 and Top 5 labels were defined as average prediction accuracies achieved by applying lower bound and upper bound levels of position noise in synthetic data. These accuracy bounds are shown as gray regions in Figure 2. n = 50 runs for both lower and upper bounds levels of noise. Accuracies obtained for top, top 3, and top 5 labels in synthetic data are similar to accuracies in experimental data (n = 9). Experimental data comes from OH15495 strain. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

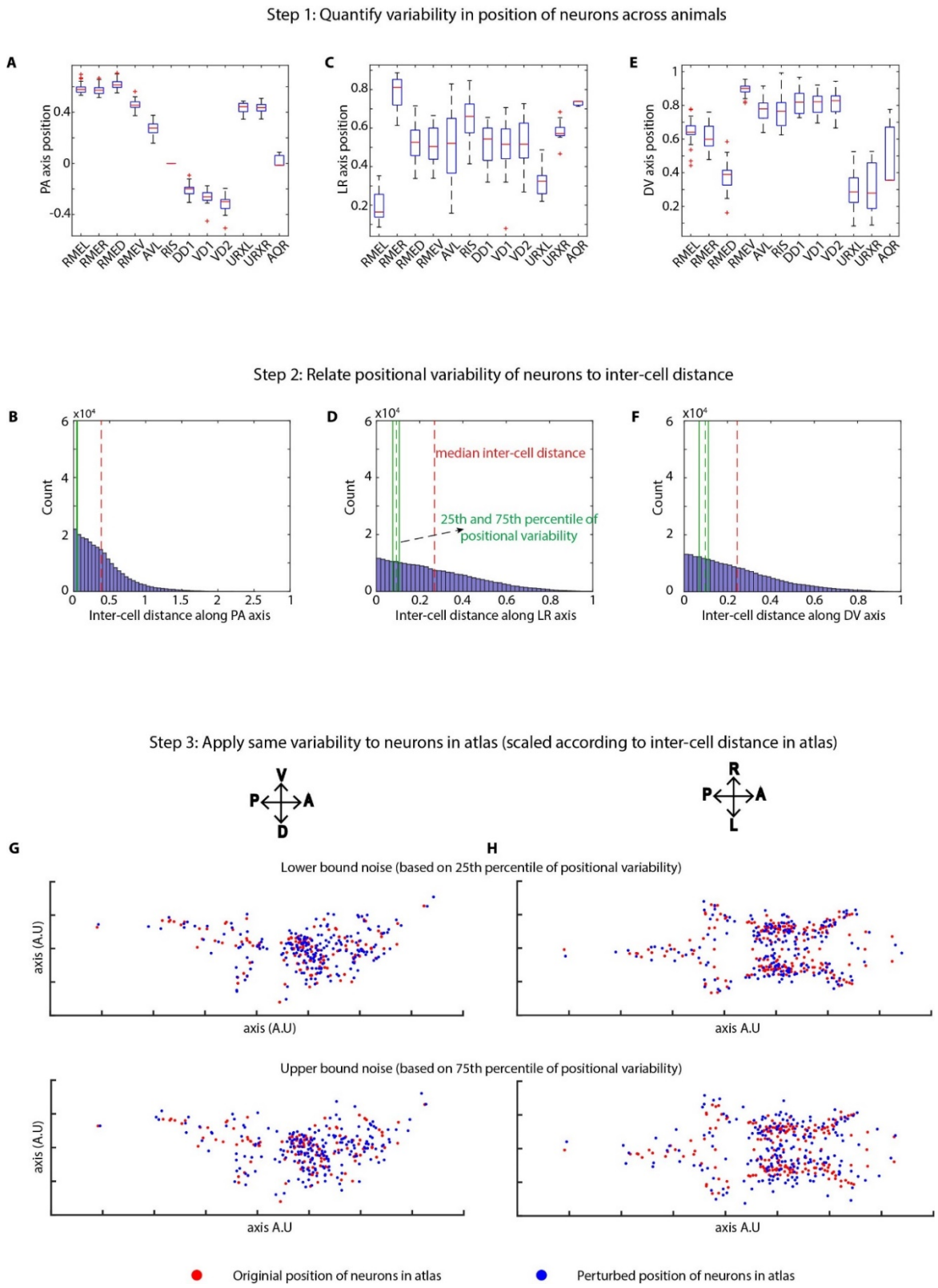


Figure 2 - figure supplement 2. ***Method of applying position noise to the atlas to generate synthetic data***

A, C, E) Variability in positions of cells were quantified in experimental data using landmark strains GT290 and GT298. Panels here show the variability of landmark cells along AP, LR, DV axes (n = 31 animals). Top and bottom lines in box plot indicate 75th percentile and 25th percentile of data, middle (red) line indicates data median, and whiskers indicate range.

B, D, F) Position variability of cells was compared to the inter-cell distances among all cells in the head ganglion. Panels show distributions of inter-cell distances between all pairs of cells in the head along AP, LR, and DV axes. 25th and 75th percentiles of cell position variability were compared to inter-cell distances to define lower bound and upper bound levels of variability in cell positions.

G, H) To calculate perturbations to be applied to cells in atlas, inter-cell distances in the atlas were calculated as well. Subsequently, the relation between position variability and inter-cell distance calculated in experimental data was used to calculate appropriately scaled perturbations based on inter-cell distance in atlas. This was done to remove the effect of different spatial scales of cell positions in experimental data vs atlas. Scaled noise perturbations were applied to each cell position. Panels show original positions of cells (red) and positions after applying perturbation (blue) in atlas for the lower bound and upper bound level of perturbations. Two views of the atlas are shown.

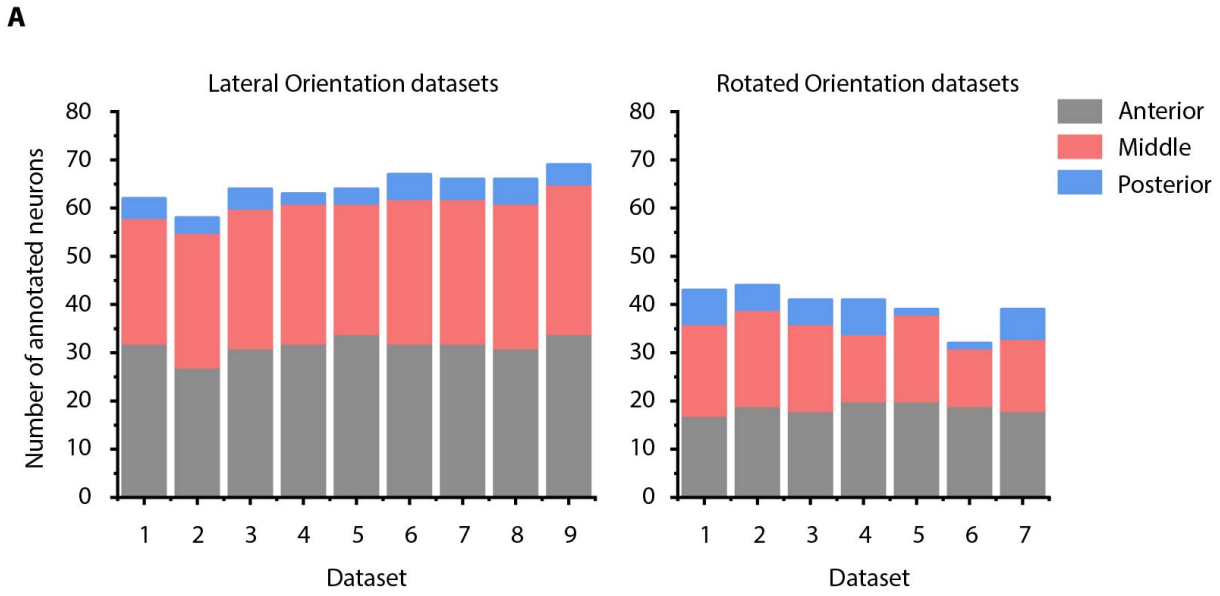


Figure 2 - figure supplement 3. ***Details of manually annotated experimental ground-truth datasets.***

A) Number of cells manually annotated in each of anterior (anterior ganglion), middle (lateral, dorsal and ventral ganglion) and posterior (retrovesicular ganglion) regions of head ganglion in two kinds of experimental ground-truth datasets: where animal is imaged in lateral orientation, i.e. LR axis is perpendicular to the image plane (left panel) and where animal is non-rigidly rotated about AP axis (right panel). See Figure 4 – figure supplement 2 for details on grouping of head ganglion in anterior, middle and posterior regions. Data was collected using NeuroPAL strains (OH15495 and OH15500).

392 To assess the effects of position noise and count noise on prediction accuracy, we simulated
393 four scenarios using the synthetic data (Figure 2 – figure supplement 1C). In the absence of any
394 noise, relative positional relationship features predicted neuron identities with perfect accuracy
395 (scenario 1 in Figure 2 – figure supplement 1C), thus demonstrating the suitability of co-
396 dependent features and CRF_ID framework for the annotation task. We found that both
397 position noise and count noise affect accuracy significantly (Figure 2 – figure supplement 1C, D)
398 with position noise having a larger effect (compare scenarios 1-2 with 3-4 in Figure 2 – figure
399 supplement 1C). As mentioned before, count noise is primarily caused by inefficiencies of either
400 the reporter used to label cells or inaccuracies of the cell detection algorithm used, thus leading
401 to fewer cells detected in the images than cells in atlas. Results on both synthetic data and real
402 data show that 10-15% improvement in prediction accuracy can be attained by simply
403 improving reagents and eliminating count noise (Figure 2 – figure supplement 1D). Next, we
404 tested the effect of landmarks (cells with known identities) on annotation accuracy (Figure 2 –
405 figure supplement 1E). We hypothesized that landmarks will improve accuracy by acting as
406 additional constraints on the optimization while the algorithm searches for the optimal
407 arrangement of labels for non-landmark cells. Indeed, we found, in both experimental data and
408 synthetic data, randomly chosen landmarks increased prediction accuracy by ~10-15%. It is
409 possible that strategic choices of landmarks could further improve accuracy.

410 Another advantage of simulations using synthetic data is that by quantifying accuracy
411 across the application of extreme-case of empirically observed noises, they can be used to
412 obtain expected accuracy bounds in real scenarios. We obtained such bounds (shown as gray
413 regions in Figure 2 – figure supplement 1F) based on observed position noise in experimental

414 data (Figure 2 – figure supplement 2). Notably, the prediction results for experimental data lay
415 close to the estimated bounds using synthetic data (Figure 2 – figure supplement 1F). Together,
416 good agreement between results obtained on synthetic and experimental data suggest that the
417 general trends uncovered using synthetic data of how various noises affect accuracy are
418 applicable to experimental data.

419 Next, with this knowledge, we tuned the features in the model, and we compared
420 prediction accuracy for several combinations of positional relationship features. Among all co-
421 dependent positional relationship features, the angular relationship feature by itself or when
422 combined with PA, LR, and DV binary position relationship features performed best (Figure 2 –
423 figure supplement 4A). To account for missing cells, we developed a method that considers
424 missing neurons as a latent state in the model (similar to hidden-state CRF[33]) and predicts
425 identities by marginalizing over latent states (see Appendix – extended methods S1.6).
426 Compared to the base case that assumes all cells are present in data, simulating missing
427 neurons significantly increased the prediction accuracy (Figure 2 – figure supplement 4B) on
428 experimental data.

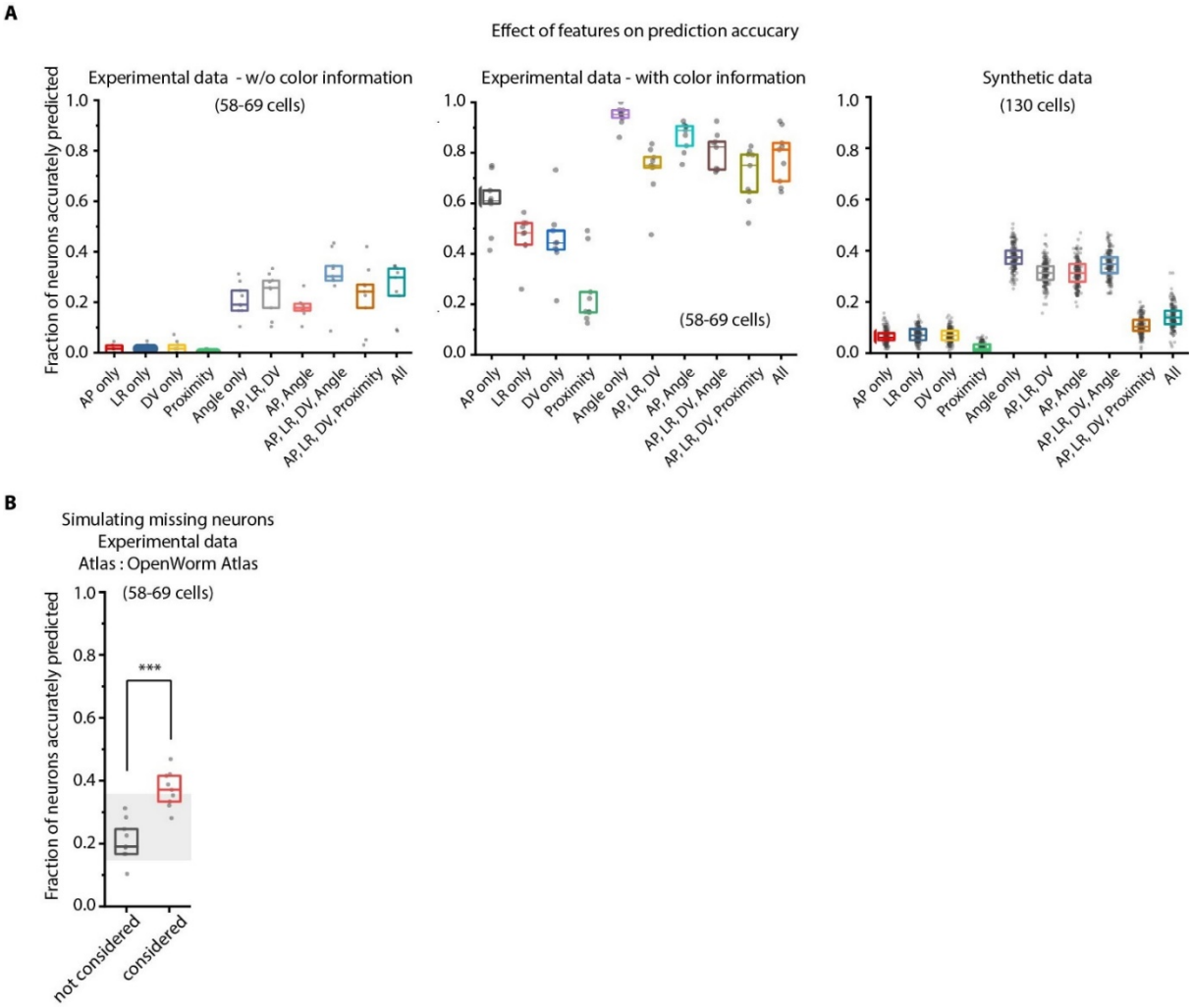


Figure 2 - figure supplement 4. ***Model tuning/characterization - features selection and simulating missing cells***

A) *Feature selection in the model was performed by keeping various feature combinations in the model and assessing prediction accuracy. Left panel – experimental data without using color information (n = 9 animals for each condition), Middle panel - experimental data using color information (n = 9 worms for each condition), Right panel – Synthetic data generated from atlas (n = 189-200 runs for each condition with random position and count noise perturbations applied in each run, see Methods). Prediction accuracy across these datasets follow a similar trend for different feature combination. Overall, the angular relationship feature by itself or combined with binary positional relationship features performs best. Experimental data comes from OH15495 strain. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.*

B) *Accounting for missing neurons improves prediction accuracy (n = 9 animals, *** $p < 0.001$, Bonferroni paired comparison test). Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.*

Identity assignment using intrinsic features in CRF_ID outperforms other methods

We next characterized the performance of our CRF_ID framework by predicting the identities of cells in manually annotated ground-truth datasets (Figure 2A). To specify prior knowledge, we built data-driven atlases combining positional information of cells from annotated ground-truth datasets and OpenWorm atlas (Appendix – extended methods S1.7). To predict cell identities in each ground-truth dataset, separate leave-one-out atlases were built keeping the test dataset held out. Building such data-driven atlas for our framework is extremely computationally efficient, requiring simple averaging operations; thus, new atlases can be built from thousands of annotated images very quickly. With data-driven atlases (of only 8 annotated set, one for each test dataset), 74% of cells were correctly identified by the top label prediction in the ground-truth data set, which exceeds the state of the art. Further, 88% and 94% of cells had true identities within the top three and the top five predicted labels respectively (Figure 2A). Note that with using only positional relationship features in the data-driven atlas, this case is equivalent to predicting identities in experimental whole-brain datasets without color information. More importantly, automated annotation is unbiased because, in principle, the framework can combine manual annotations of cell identities of several users (possibly across labs) in the form of data-driven atlases and can predict identities such that positional relationships in the atlas are maximally preserved. Thus, automated annotation removes individual biases in annotating cells. Further, it greatly supports researchers with no prior experience.

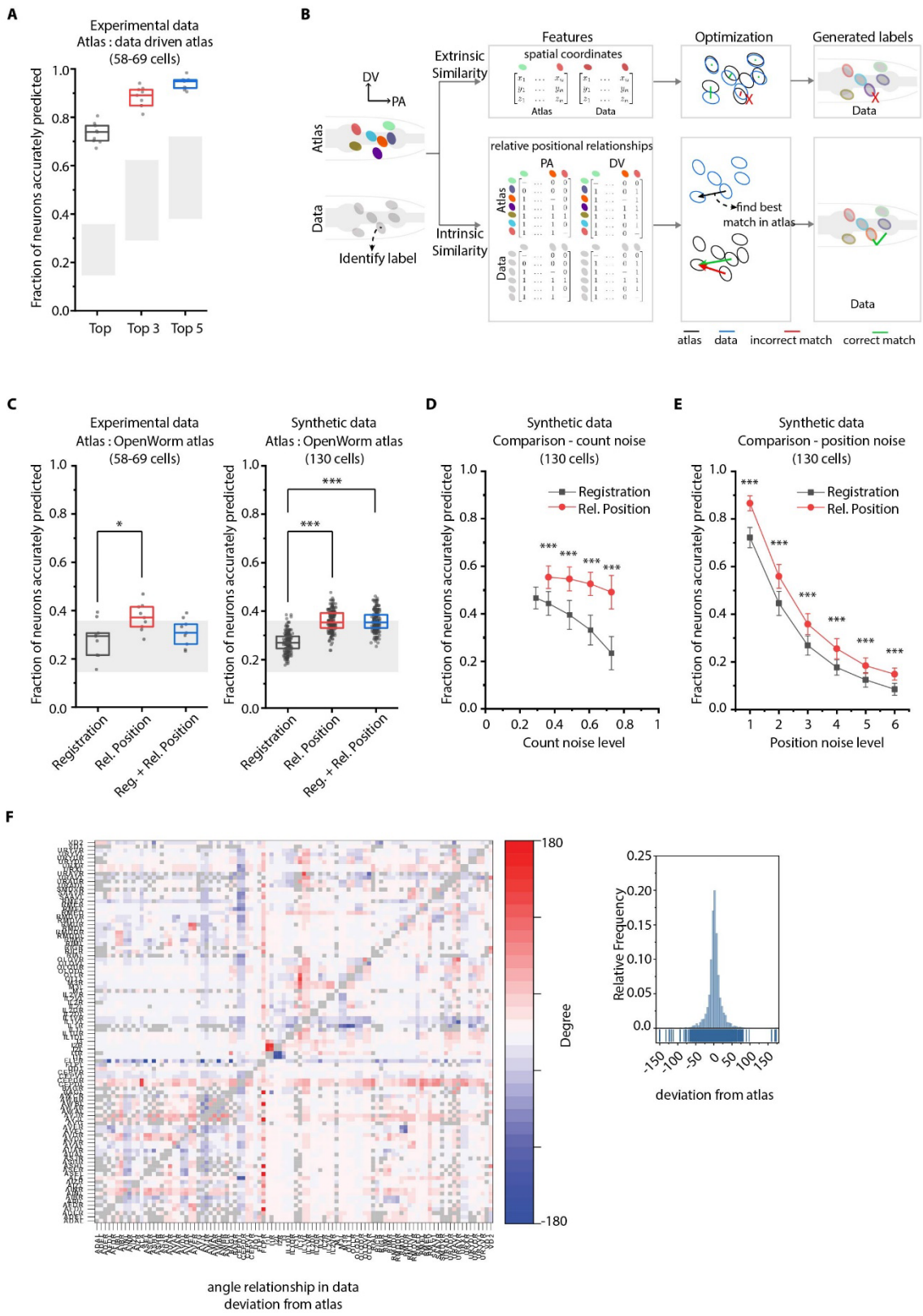


Figure 2. ***CRF_ID annotation framework outperforms other approaches.***

465 *A) CRF_ID framework achieves high prediction accuracy (average 73.5% for top labels)*
466 *using data-driven atlases without using color information. Results shown for whole-brain*
467 *experimental ground truth data (n = 9 animals). Prediction was performed using leave-*
468 *one-out data-driven atlases built with held out test dataset. Gray regions indicate*
469 *bounds on prediction accuracy obtained using simulations on synthetic data (see Figure*
470 *2 – figure supplement 1F). Experimental data comes from strain OH15495. Top, middle*
471 *and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data*
472 *respectively.*

473 *B) Schematic highlighting key difference between registration-based methods and our*
474 *CRF_ID framework.*

475 *C) Prediction accuracy comparison across methods for ground truth experimental data*
476 *(n=9, * p < 0.05, Bonferroni paired comparison test) and synthetic data (n = 190-200 runs*
477 *for each method, ***p < 0.001, Bonferroni paired comparison test). OpenWorm atlas*
478 *was used for predictions. Accuracy results shown for top predicted labels. Experimental*
479 *data comes from strain OH15495. For synthetic data, random but realistic levels of*
480 *position and count noise applied in each run. Gray regions indicate bounds on prediction*
481 *accuracy obtained using simulations on synthetic data (see Figure 2 – figure supplement*
482 *1F). Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th*
483 *percentile of data respectively.*

484 *D) Comparison of methods across count noise levels (defined as percentage of cells in atlas*
485 *that are missing from data) using synthetic data. (n = 150-200 runs for Rel. Position for*
486 *each noise level, n = ~1000 runs for Registration for each noise level, ***p < 0.001,*
487 *Bonferroni paired comparison test). OpenWorm atlas was used for prediction. Accuracy*
488 *results shown for top predicted labels. For a fixed count noise level, random cells were*
489 *set as missing in each run. Markers and error bars indicate mean ± standard deviation.*

490 *E) Comparison of methods across position noise levels using synthetic data. (n = 190-200*
491 *runs for each method for each noise level, ***p < 0.001, Bonferroni paired comparison*
492 *test). OpenWorm atlas was used for prediction. Accuracy results shown for top predicted*
493 *labels. For a fixed position noise level, random position noise was applied to cells in each*
494 *run. Different noise levels correspond to different variances of zero-mean gaussian noise*
495 *added to positions of cells (see section Methods – Generating synthetic data for*
496 *framework tuning and comparison against other methods). Noise levels 3 and 6*
497 *correspond to the lower bound and upper bound noise levels shown in Figure 2 – figure*
498 *supplement 1F. Markers and error bars indicate mean ± standard deviation.*

499 *F) Pairwise positional relationships among cells are more consistent with OpenWorm atlas*
500 *even though the absolute positions of cells vary across worms. (Left) average deviation*
501 *of angular relationship measured in ground truth data (n=9) from the angular*
502 *relationship in static atlas. (Right) distribution of all deviations in left panel (total of*
503 *8,516 relationships) is sparse and centered around 0 deviation, thus indicating angular*
504 *relationships are consistent with atlas.*

505 We next compared our method against registration-based methods popular for automatic
506 cell annotation[1,14,16–18] (see Appendix – extended methods S1.9 and S2.1). For fair comparison
507 across methods, all methods used OpenWorm atlas as reference for prediction. The major
508 difference between our framework and previous methods is the use of intrinsic similarity
509 compared to extrinsic similarities in previous methods in the annotation task (Figure 2B, Figure
510 1 – figure supplement 1). Remarkably, for both experimental and synthetic data, CRF_ID using
511 relative positional features performs the best (Figure 2C; Figure 2 – figure supplement 5A).
512 Note that the decrease in accuracy compared to Figure 2A here is due to using static
513 OpenWorm atlas, further highlighting the importance of building data-driven atlases. Notably,
514 CRF_ID outperforms registration-based method across all levels of count noise and position
515 noise in data (Figure 2D, E; Figure 2 – figure supplement 5B, C). The accuracy of registration-
516 based methods falls rapidly with increasing count noise levels, whereas CRF_ID is highly robust,
517 maintaining higher accuracy even when up to 75% of cells in atlas were missing from data. This
518 has important practical implications as the amount of count noise observed in experimental
519 data may vary significantly across reagents, imaging conditions etc. Further, neuron positions
520 being highly variable across individual animals have been shown[10], and confirmed by our
521 datasets as well(Figure 2 – figure supplement 6A). Because cell positions on average can deviate
522 from their atlas position by more than the distance to their tenth nearest neighbor (Figure 2 –
523 figure supplement 6B), we expect that this variability introduces large matching errors in
524 registration-based methods. In contrast, most pair-wise relationships are preserved despite the
525 variability of absolute positions (Figure 2F; Figure2 – figure supplement 6C, D). Interestingly, a
526 hybrid objective function that combines registration using absolute positions with relative

527 position features in CRF_ID framework corrupts the annotation performance (Figure 2 – figure
528 supplement 5A), likely due to competing effects in the objective function. This again highlights
529 the fact that higher accuracy is achieved by positional relationship features in CRF_ID method.

530 Next, to compare the computational efficiency of CRF_ID framework with that of
531 registration based methods, we compared the optimization step runtimes of the singlerun
532 mode of CRF_ID framework with that of a popular registration method (Coherent Point Drift[34])
533 (Figure 2 – figure supplement 7A). The computational speed of both methods scales with
534 number of cells to be annotated in images and number of cells in the atlas. As expected, CRF_ID
535 framework is computationally more expensive compared to CPD, because it optimizes both
536 unary and pairwise potentials. Nonetheless, the optimization runtime of CRF_ID framework for
537 multi-cell calcium imaging use-case (10-50 cells in image) is on the order of 0.1-10 seconds, on a
538 desktop computer (see Materials and Methods – Runtime comparison), when full head ganglion
539 atlas (206 cells) is used for annotation. We emphasize that using full head ganglion atlas for cell
540 identity annotation in whole-brain imaging is important because without prior knowledge of
541 which cells are missing in images, full atlas provides unbiased opportunity to cells in images to
542 take any label from the atlas. In contrast, if only a partial atlas or partially annotated data set is
543 used as atlas, the labels absent in atlas will never get assigned to any cell in images, thus
544 potentially biasing the annotation. In practice, faster runtimes can be achieved in multi-cell
545 calcium imaging and whole-brain imaging case with smaller atlases based on prior knowledge of
546 cells expected in strains. Further, the multiple-run mode of CRF_ID framework can be
547 parallelized using multiple CPU workers. Thus, higher accuracy compared to registration based
548 methods combined with reasonable speeds makes CRF_ID favorable for cell annotation tasks.

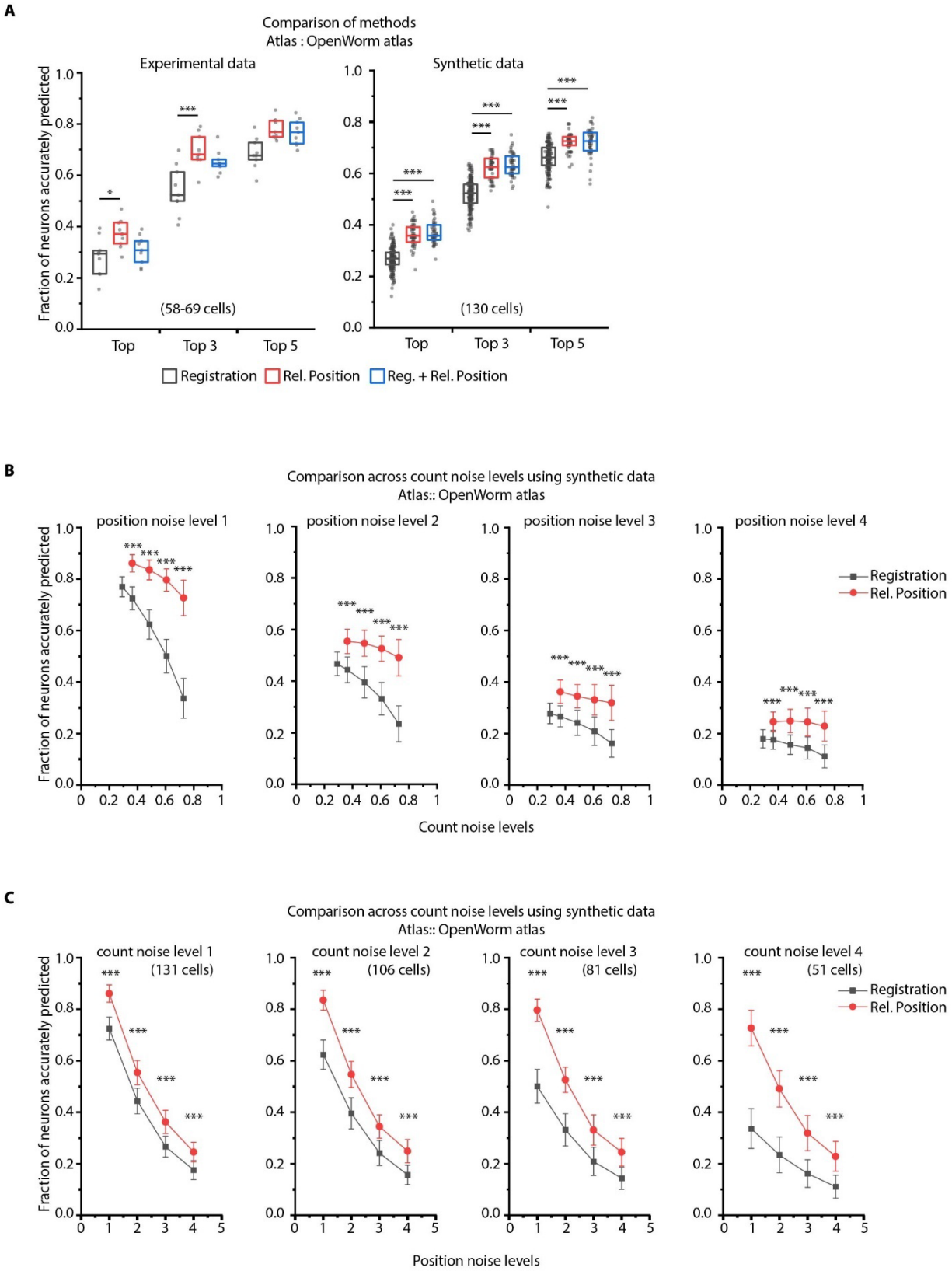


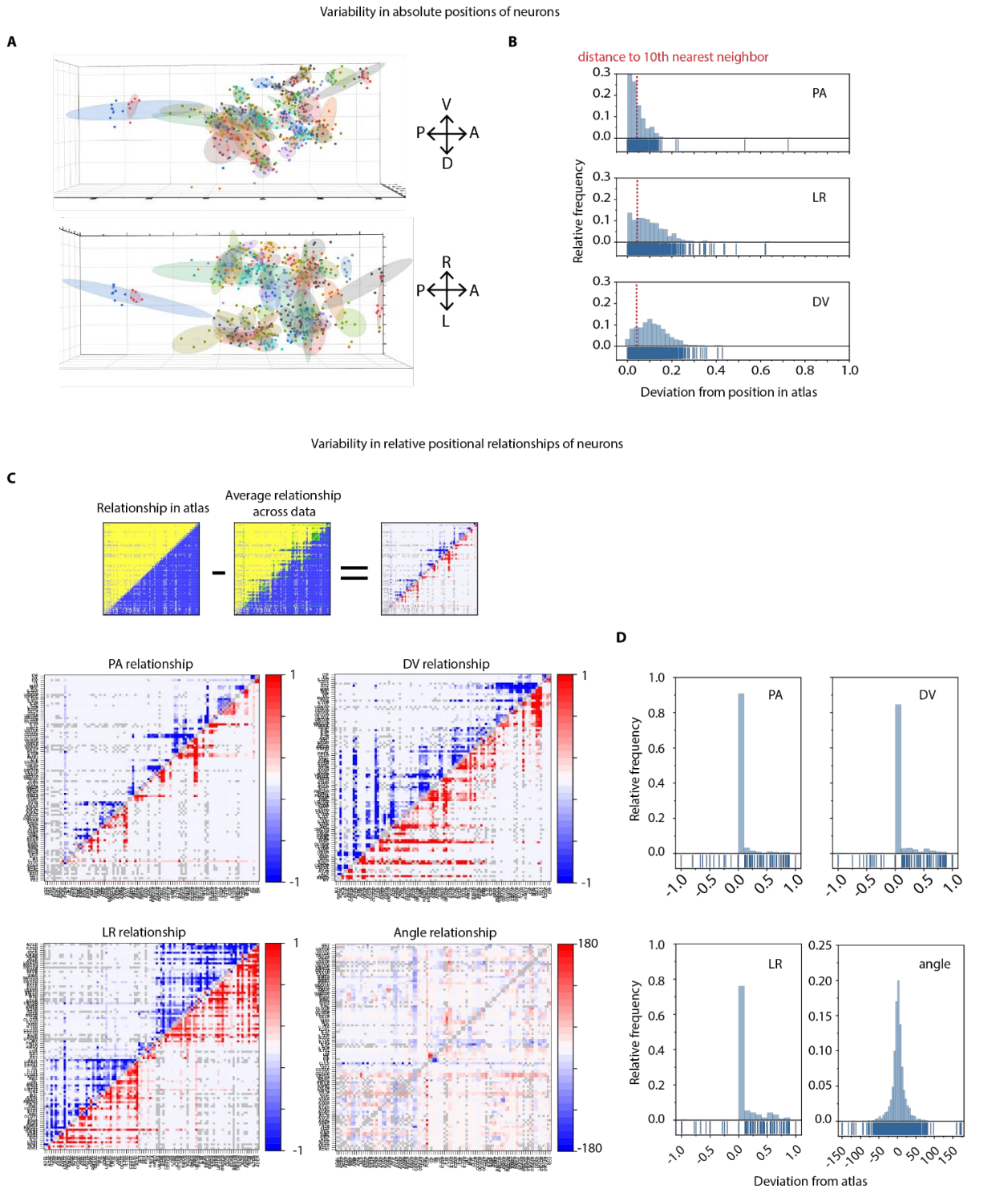
Figure 2 - figure supplement 5. ***CRF_ID framework with relative positional features outperforms registration method***

551 *A) Prediction accuracies achieved by Top, Top 3 and Top 5 labels predicted by 3 methods – Registration, CRF_ID*
552 *framework with Relative Position features and CRF_ID framework with combined features (see appendix – extended*
553 *methods S2.1-2.3). Relative position features outperform the registration method in both experimental data (left*
554 *panel) and synthetic data (right panel). For experimental data, n = 9 worm datasets, strain OH15495. For synthetic*
555 *data, n = 200 runs for registration method, n = 48-50 runs for Rel. Position method and combined method. Each run*
556 *differed from the other in terms of position noise and count noise applied. *** denotes p < 0.001, Bonferroni paired*
557 *comparison test. Part of data is re-plotted in Figure 2C. Top, middle and bottom lines in box plot indicate 75th*
558 *percentile, median and 25th percentile of data respectively.*

559 *B) Accuracy comparison across count noise levels (defined as percentage of cells in atlas that are missing from data) for*
560 *various fixed position noise levels. (n = 180-200 runs for Rel. Position for each count noise level for each condition, n =*
561 *~500 runs for Registration for each count noise level for each condition, ***p < 0.001, Bonferroni paired comparison*
562 *test). Results for position noise level 2 are shown in Figure 2D. For a fixed count noise level, random cells were*
563 *removed in each run. Markers and error bars indicate mean ± standard deviation.*

564 *C) Accuracy comparison across position noise levels for various fixed count noise levels. (n = 180-200 runs for Rel. Position*
565 *for each position noise level for each condition, n = ~500 runs for Registration for each position noise level for each*
566 *condition, ***p < 0.001, Bonferroni paired comparison test). Different noise levels correspond to different variances of*
567 *zero-mean gaussian noise added to positions of cells (see section Methods – Generating synthetic data for framework*
568 *tuning and comparison against other methods). For a fixed position noise level, random noise was applied to cell*
569 *positions in each run. Noise levels 3 corresponds to the lower bound noise level calculated in Figure 2 – figure*
570 *supplement 2. Markers and error bars indicate mean ± standard deviation.*

571



572

573 *Figure 2 - figure supplement 6.* ***Variability in absolute positions of cells and relative positional features in experimental data***
574 ***compared to the static atlas***

575 *A) DV view (top) and LR view showing positions of cells across ground truth data (n = 9 worms, strain OH15495). Each*
576 *point cloud of one color represents positions of a specific cell across datasets. Ellipsoid around the point clouds shows*
577 *50% confidence interval. Total 98 point-clouds for 98 uniquely identified cells across 9 datasets shown here.*

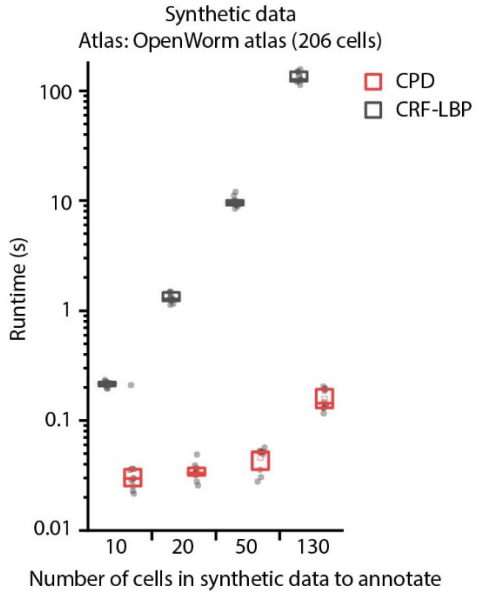
578 *B) Distributions of deviations of absolute positions of cells in experimental data (n – 9 animals, strain OH15495)*
579 *compared to corresponding positions in static OpenWorm atlas. Top, middle and bottom panels show deviation along*
580 *anterior-posterior, (AP), left-right (LR) and dorsal-ventral (DV) axes. To put the deviations into perspective, dotted red*
581 *lines show the median distance of each cell to its 10th nearest neighbor. Thus, in experimental data, cell positions can*
582 *be deviated from their atlas position by much more than their distances to their 10th nearest neighbor. As a result,*
583 *registration method often creates mismatches*

584 *C) Deviation of relative positional relationships among cells compared to the relationship in static atlas. Top panel shows*
585 *a schematic of how these deviations were calculated. E.g. each positional relationship is represented as a matrix. Each*
586 *element in the matrix records the positional relationship between pair of cells. E.g. in case of static atlas, if a column*
587 *cell such as RMEL is anterior to a row cell such as AIZR, then the corresponding elements will denote 1 otherwise 0.*
588 *Similarly, for experimental data (n = 9 worms, same as panel A, strain OH15495), each element denotes the average*
589 *number of times RMEL is observed anterior to AIZR in data. Hence the value can be between 0 and 1. Matrices are*
590 *calculated in the same manner for other features. A simple subtraction of the two matrices records the deviation of*
591 *the positional relationship feature. Below panels show deviations of AP, LR, DV and angular relationships between*
592 *static atlas and data. Closer to 0 values in PA and angular relationship indicates these relationships are most similar to*
593 *static atlas. Gray cells in matrix denote relationships that could not be measured across 9 datasets (when both cells*
594 *were never annotated in same data)*

595 *D) Distributions of deviation of relative positional relationship features between static atlas and data. Total 8,546*
596 *positional relationship features compared for 98 uniquely identified cell across data. Sparseness of the distributions*
597 *indicate that the relative positional relationships are consistent with the atlas although absolute positions of cells vary*
598 *a lot (panel a, b)*

599

A



600

601 *Figure 2 - figure supplement 7.* ***Comparison of optimization runtimes of CRF_ID framework with a registration method CPD[34].***

602 *A) Optimization runtimes of CRF method using Loopy Belief Propagation (LBP) as optimization method, and registration*
603 *method CPD across different number of cells in data to be annotated. Synthetic data was used for simulations and*
604 *annotation was performed using OpenWorm atlas with 206 head ganglion cells (n = 10 runs). Top, middle and bottom*
605 *lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.*

606

607 ## Cell annotation in gene-expression pattern analysis

608 We next demonstrate the utility of our framework for gene-expression localization analyses,
609 which is important for many problems, e.g. mapping the cellular atlas of neurotransmitters[35,36],
610 receptors[37], and neuropeptides[38]. Conventional methods e.g. screening a list of cell specific
611 marker lines that overlap in expression with the reporter are laborious and scale poorly with
612 the number of cells expressing the genes of interest and the number of new genes for which
613 expression patterns are to be determined. Our cell annotation framework can considerably
614 reduce manual efforts by generating a small list of candidate identities for each cell expressing
615 the reporter. Subsequently, researchers can easily verify or prune the candidate list. To
616 demonstrate this use case, we imaged a strain with multiple cells labeled with GFP and
617 predicted candidate identities for each cell (Figure 3A). Determining cell identities in this case is
618 difficult due to large count noise along with position noise: since the full list of labels in the atlas
619 is much bigger than few cells in the reporter strain (equivalent to scenario 4 in Figure 2 – figure
620 supplement 1C). Thus, several degenerate (equally probable) solutions are possible. To avoid
621 accuracy decrease in such cases, we directly predicted the candidate identities of all cells
622 marked with pan-neuronal red fluorescent protein (RFP) using full whole-brain atlas and
623 subsequently assessed the accuracy of only cells of interest, i.e. those marked with GFP. Our
624 framework accurately generated a candidate list for cells across all datasets (n = 21 animals);
625 85% of cells had true identities within the top 5 labels chosen by the framework. In comparison,
626 the candidate list generated by the registration method achieved only 61% accuracy (Figure
627 3B).

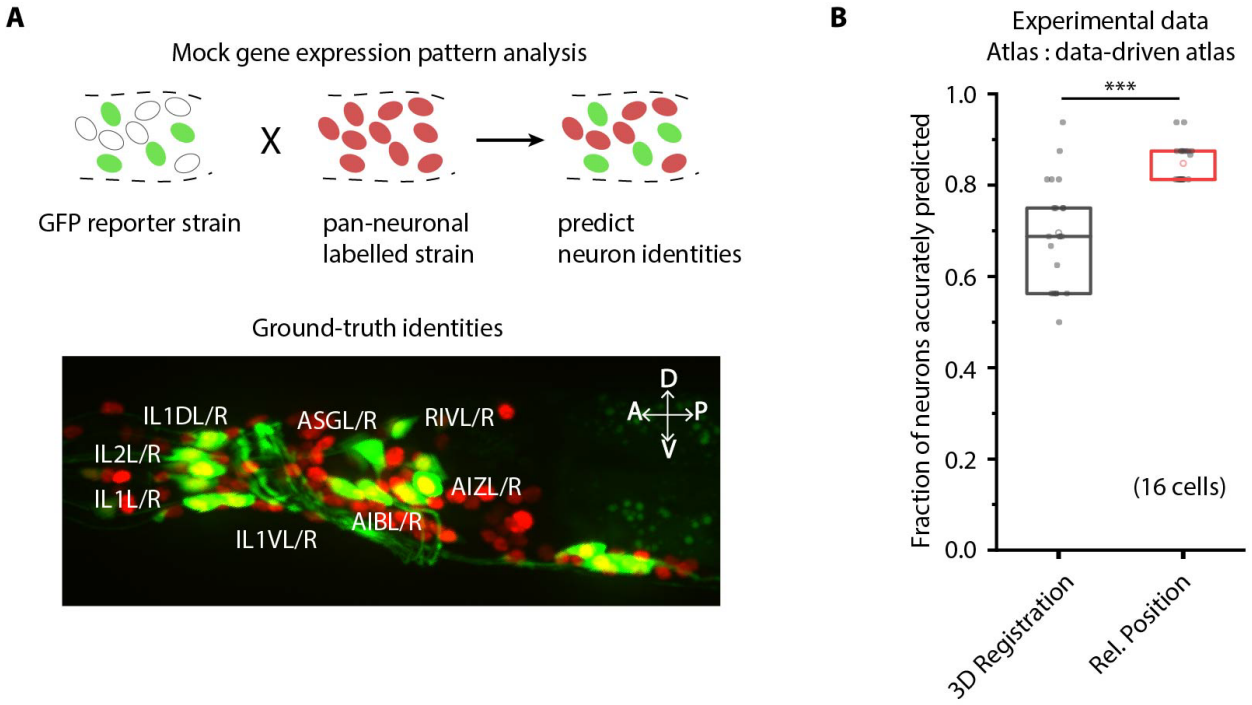


Figure 3. ***CRF_ID framework predicts identities for gene expression pattern analyses.***

A) (Top) Schematic showing a fluorescent reporter strain with GFP expressed in cells for which names need to be determined. Since no candidate labels are known a priori neurons labels are predicted for all cells marked with pan-neuronally expressed RFP using full whole-brain atlas. (Bottom)A proxy strain AML5 [rab-3p(prom1)::2xNLS::TagRFP; odr-2b::GFP] with pan-neuronal RFP and 19 cells labeled with GFP was used to assess prediction accuracy.

*B) CRF_ID framework with relative position features outperforms registration method (n = 21 animals) (***p < 0.001, Bonferroni paired comparison test). Accuracy shown for top 5 labels predicted by both methods. Experimental data comes from strain AML5. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.*

643 ## Cell annotation in multi-cell functional imaging experiments

644 We next demonstrate the utility of our algorithm in another important application - annotating
645 cell identities in multi-cell calcium functional imaging *in vivo* (Figure 4). Automation in this case
646 dramatically reduces labor associated with cell annotation for many time points, across trials,
647 animals, and experiments. We used a strain carrying GFP in multiple cells as a proxy for GCaMP-
648 labeled strains for illustration purposes (Figure 4A). Given the known candidate list of labels
649 that can be assigned (i.e. no count noise), the configurational space is small, which makes the
650 task easy (similar to scenario 3 in Figure 2 – figure supplement 1C). Indeed, our annotation
651 framework identified neurons with high accuracy (98%, n = 35 animals). In comparison, the
652 registration method predicted identities with lower accuracy (88%) even with the small label
653 assignment space (Figure 4B). In reality, some neurons may be undetected in the data due to
654 expression mosaicism or low calcium transients thus adding count noise to data (equivalent to
655 scenario 4 in Figure 2 – figure supplement 1C). We thus simulated this case by randomly
656 removing up to a third of total neurons from the images and predicting identities of remaining
657 cells using the full label list (Figure 4C; Figure 4 – figure supplement 1A). Even under these
658 conditions, the accuracy of our method remains high (88%), significantly outperforming
659 registration method (81%) (Figure 4 – video supplement 1). In practice, the performance can be
660 further compensated for by using multiple frames from each video, which we are not doing
661 here in the mock experiment.

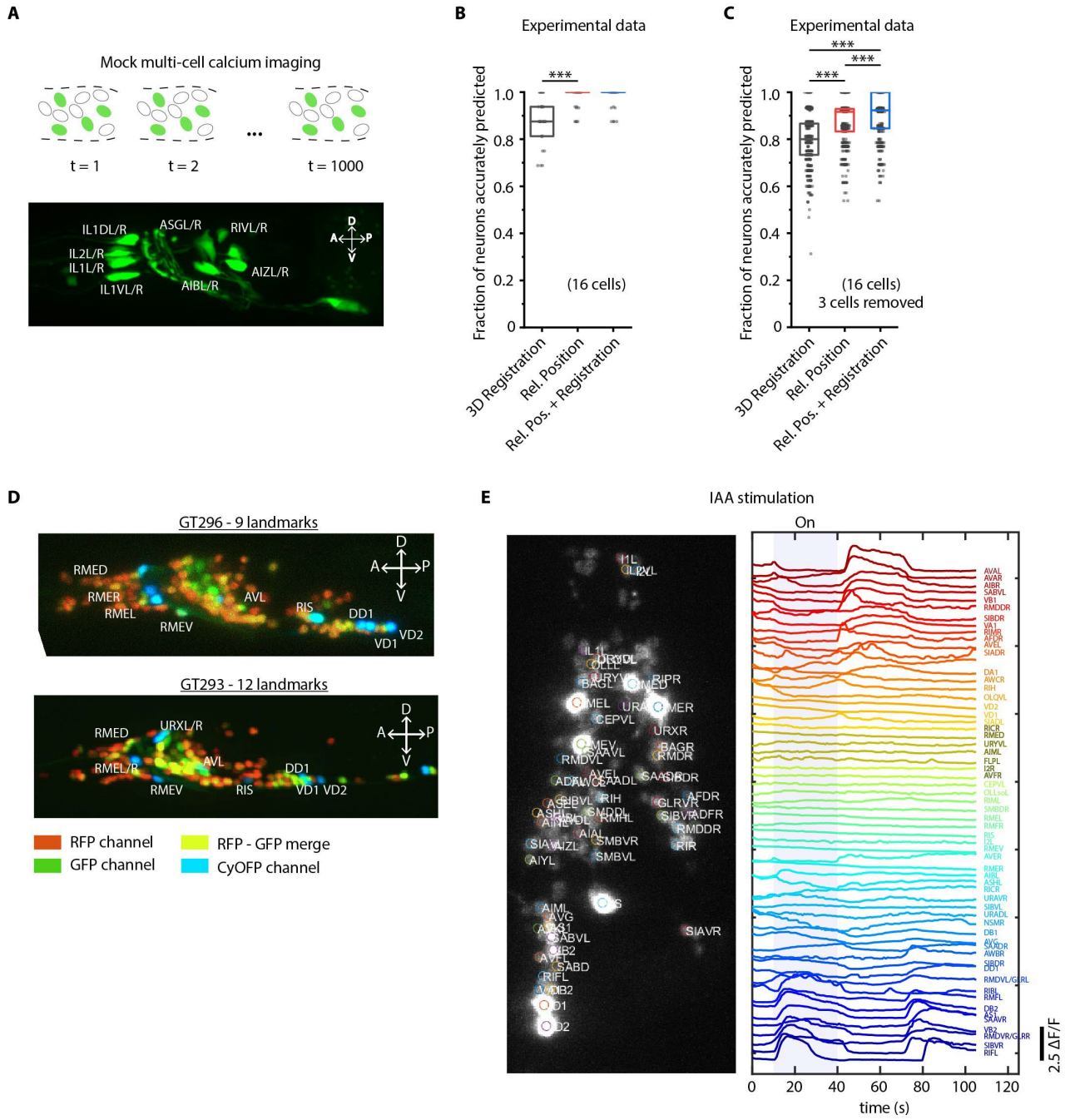


Figure 4. ***Cell identity prediction in mock multi-cell calcium imaging experiments and landmark strain.***

A) (Top) schematic showing automatic identification of cells in multi-cell calcium imaging videos for high-throughput analysis. (Bottom) A mock strain with GFP labeled cells was used as an illustration of GCaMP imaging. Only green channel of AML5 strain was used for this purpose.

670

671 *B) CRF_ID framework outperforms registration method (n = 35 animals, *** p < 0.001,*
672 *Bonferroni paired comparison test). OpenWorm atlas was used for prediction. Accuracy*
673 *results shown for top predicted labels. Experimental data comes from strain AML5 (only*
674 *green channel used). Top, middle and bottom lines in box plot indicate 75th percentile,*
675 *median and 25th percentile of data respectively.*

676 *C) Prediction accuracy comparison for the case of missing cells in images (count noise). ****
677 *p < 0.001, Bonferroni paired comparison test. Total n = 700 runs were performed across*
678 *35 animals for each method with 3 out 16 randomly selected cells removed in each run.*
679 *For fair comparison, cells removed across methods were the same. OpenWorm atlas was*
680 *used for prediction. Accuracy results shown for top predicted labels. Experimental data*
681 *comes from strain AML5 (only green channel used). Top, middle and bottom lines in box*
682 *plot indicate 75th percentile, median and 25th percentile of data respectively.*

683 *D) Max-projection of 3D image stacks showing CyOFP labeled landmark cells in head*
684 *ganglion (pseudo-colored as cyan): animals carrying [unc47p::NLS::CyOFP1::egl-13NLS]*
685 *(GT296 strain) with 9 landmarks (top), and animals carrying [unc-47p::NLS::CyOFP1::egl-*
686 *13NLS; gcy-32p::NLS::CyOFP1::egl-13NLS] with 12 landmarks (bottom).*

687 *E) (Left) max-projection of a 3D image stack from whole-brain activity recording showing*
688 *head ganglion cells and identities predicted by CRF_ID framework (Top labels). Animal is*
689 *immobilized in a microfluidic device channel and IAA stimulus is applied to the nose tip.*
690 *(Right) GCaMP6s activity traces extracted by tracking cells over time in the same 108s*
691 *recording and their corresponding identities. Blue shaded region shows IAA stimulation*
692 *period. Experimental data comes from strain GT296.*

693

A

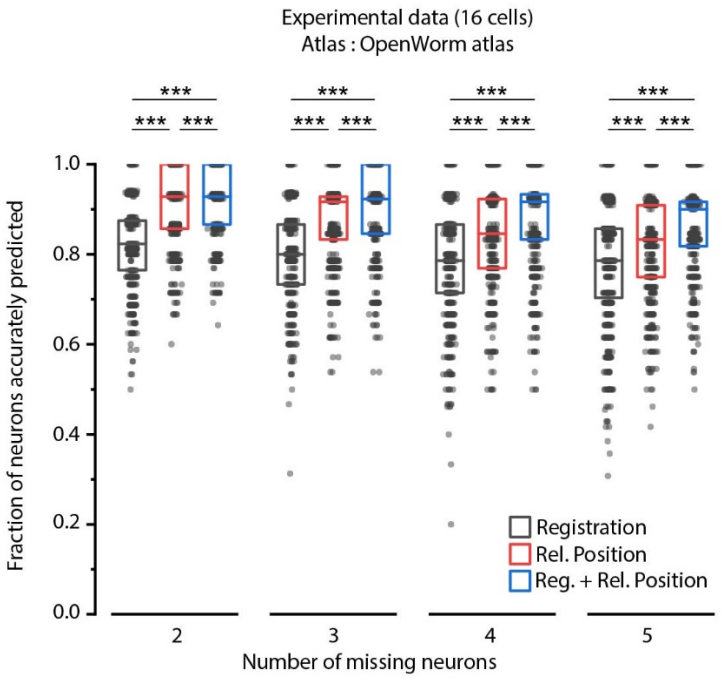


Figure 4 - figure supplement 1. ***Relative position features perform better than registration in handling missing cells in images***

*A) Comparison of prediction accuracies across three methods for different number of missing cells (out of total 16 cells) simulated in experimental data. Experimental data comes from AML5 strain (only GFP channel used). Results for 3 cells missing are shown in Figure 4C. *** $p < 0.001$, ** $p < 0.01$, all comparisons done with Bonferroni paired comparison test. n = 700 runs were performed across data from 35 animals for each number of missing cells and each method. For fair comparison across methods, the same cells were considered missing for all methods for a fixed missing cell number. Each run differed from the other with respect to random subset of cells considered missing. Part of data is re-plotted in figure 4C. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.*

705 To further facilitate annotation accuracy, we explored the utility of landmarks with known
706 identities. Landmarks can also help in establishing a coordinate system in images and guiding
707 post-prediction correction. Because the combinatorial space of potential landmarks is very
708 large (~10^{14} for 10 landmarks out of ~200 cells in the head), we asked what properties
709 landmarks should have. We found that landmarks distributed throughout the head or in lateral
710 ganglion perform better in predicting identities of neurons in all regions of the brain (Figure 4 –
711 figure supplement 2; Methods). As a test case, we developed strains with spatially distributed,
712 sparse neuronal landmarks using CyOFP (see Appendix – extended methods S3), which by itself
713 can assist researchers in manual cell identification tasks. When crossed with pan-neuronally
714 expressing GCaMP/RFP reagents, the strains can be used for whole-brain imaging (Figure 4D) by
715 using only two channels. This has two advantages: CyOFP can be imaged "for free" while
716 imaging GCaMP and RFP simultaneously, thus the landmarks providing a concurrent reference
717 in all frames; this strategy also leaves other channels open for optogenetic manipulations and
718 voltage imaging[39,40].

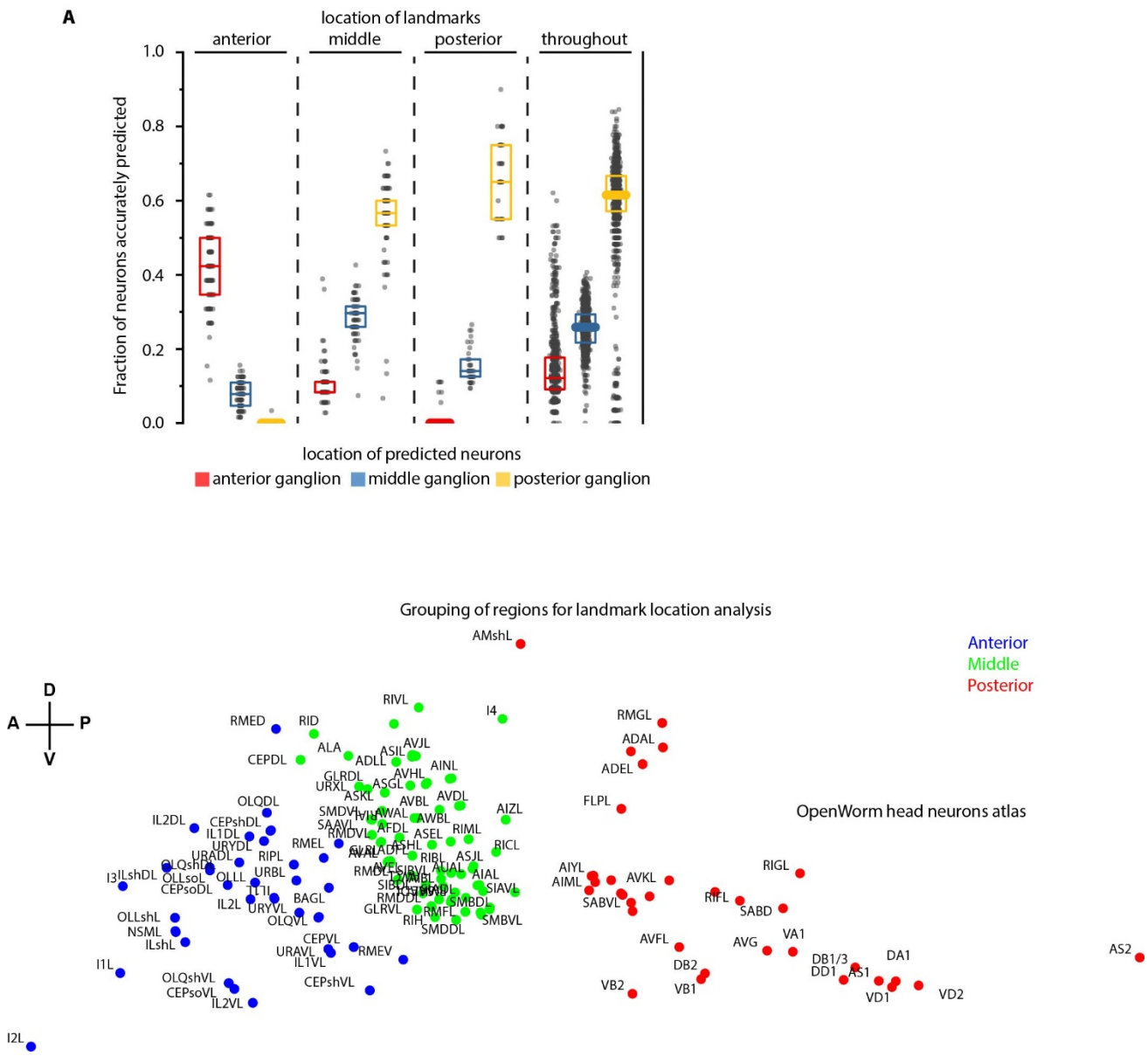


Figure 4 - figure supplement 2. ***Spatially distributed landmarks or landmarks in lateral ganglion perform best in supporting CRF_ID framework for predicting identities***

A) Top panel – Region-wise prediction accuracy achieved by our CRF_ID framework when landmarks were constrained to lie in specific regions of the head. n = 200 runs when landmarks were constrained in anterior, middle and posterior regions, n = 1000 runs when landmarks were spatially distributed throughout the head. All data is synthetic data. A random combination of 15 landmarks was selected in each run. Landmarks constrained in the anterior region perform badly in predicting identities of the posterior region, similarly landmarks constrained in the posterior region perform badly in predicting identities of the anterior region. Landmarks in the middle region or spatially distributed throughout the head show balanced accuracy for all regions. Bottom panel – shows the grouping of head regions as anterior, middle and posterior regions. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

732 We next tested this strategy in a simple whole-brain imaging experiment. Isoamyl alcohol
733 (IAA) is a well-known component of the bacterial metabolites that *C. elegans* senses and
734 responds to[41–43]. We recorded neuronal responses to a step-change in IAA concentration using
735 a microfluidic system[44] (Figure 4 – figure supplement 3). We observed both odor-specific
736 responses and spontaneous activities (Figure 4E). More importantly, neurons with algorithm-
737 assigned identities demonstrate expected behavior. For instance, we identified the sensory
738 neuron AWC, and detected an off-response to IAA, consistent with known AWC behavior. In
739 addition, the predicted interneurons (e.g. AVA, RIB, and AIB) also demonstrate previously
740 known activity patterns[5].

741 We also tested worms' responses to periodic stimuli of a more complex and naturalistic
742 input – supernatant of bacterial culture (Figure 5, Figure 5 – video supplement 1). A periodic
743 input (5 s On and 5 s Off for 8 cycles) entrains many neurons as expected, therefore allowing us
744 to better separate the odor-elicited responses from spontaneous activities (Figure 5A). We
745 generated the candidate identities for all recorded neurons (Figure 5 – figure supplement 1A).
746 Notably, several highly entrained neurons were identified as sensory neurons known to
747 respond to food stimuli[45–47] (Figure 5B), some of which responded to the onset of the stimuli
748 and some to the withdrawal of the stimuli (Figure 5D). The power spectrum of these neurons
749 showed a strong frequency component at 0.1 Hz as expected (Figure 5C).

750

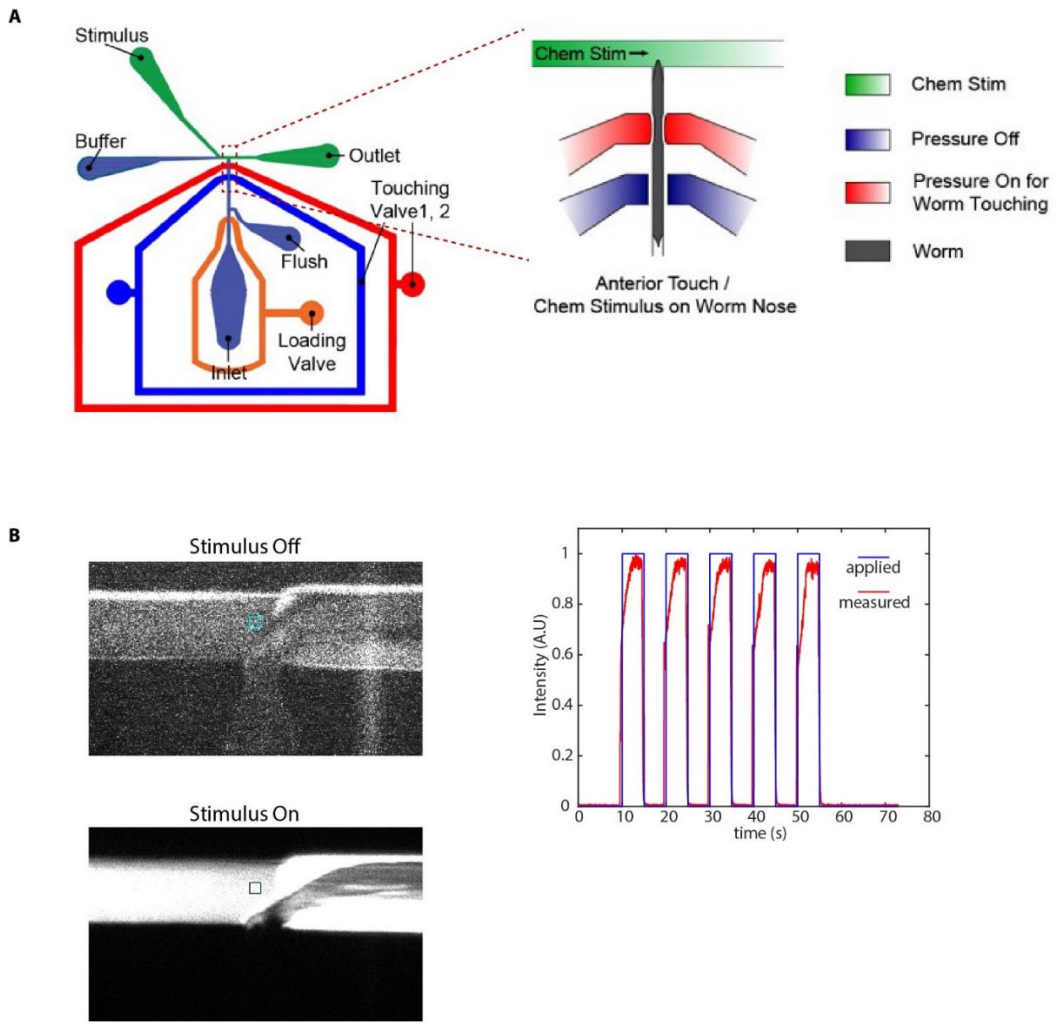


Figure 4 - figure supplement 3. ***Microfluidic device used in chemical stimulation experiments and characterization***

A) Schematic of the microfluidic device[44] used in chemical stimulation experiments. The position of nematode in the imaging channel is shown. Temporally varying stimulus is applied to the nose tip of the nematode by switching between food/IAA and buffer streams

B) Stimulus characterization was performed using FITC as stimulus and S-basal as buffer. Figures show a zoom-in of the T-junction in the device (where the nose tip of the nematode would be). Colored boxes show the regions used to characterize the stimulus profile.

C) Applied stimulus profile and measured stimulus profile for 5 sec on and 5 sec off stimulus.

761 Next, to examine the latent dynamics in the whole-brain activities during the entire
762 experiment, we used traditional Principal Component Analysis (PCA) and Sparse Principal
763 Component Analysis (sPCA)[48]. The overall dynamics are low-dimensional with top 3 traditional
764 PCs capturing 70% of the variance (Figure 5 – figure supplement 1B). In comparison, while the
765 top 3 sparse PCs (SPCs) explain 43% of the variance in the data, they enable meaningful
766 interpretation of the latent dynamics by eliminating mixing of activity profiles in PCs (Figure 5E).
767 SPC1 shows a systematic decline of the signals, presumably related to photobleaching of the
768 fluorophores; both SPC2 and SPC3 illustrate spontaneous activities with different temporal
769 dynamics. With automatic annotation, we were able to identify cell classes belonging to each
770 SPC (Figure 5 – figure supplement 1C). We then analyzed the relationship between motion and
771 neuron activities. In our microfluidic device, the animals are not fully immobilized. By tracking
772 landmarks on the body; we observed propagating waves along the body (Figure 5F; Figure 5 –
773 figure supplement 1D, Figure 5 – video supplement 2). Interestingly, cells participating in SPC2
774 showed significantly higher mutual information with motion than any other component (Figure
775 5G). Examining the connection between activities of neurons that drive SPC2 and animal
776 motion demonstrates that these neurons are indeed correlated or anti-correlated with the
777 motion we detected (Figure 5H); notably these neurons included several command
778 interneurons such as AVA, RIM, and motor neurons such as VA, DA[5] (Figure 5H). Cross-
779 correlation analysis between motion and neuron activities showed that neurons are activated
780 ahead of motion (Figure 5I); when lag is added to the neuron activities, the mutual information
781 of SPC2 neurons with motion is maximum at the same delay observed in the cross-correlation
782 (Figure 5 – figure supplement 1E). These experiments together demonstrate the power of the

783 approach, which enabled previously difficult simultaneous analyses of several sensory, inter-,

784 and motor neurons' activities to natural food stimulus. Thus, automatic identity prediction

785 enabled meaningful interpretation of the whole-brain data.

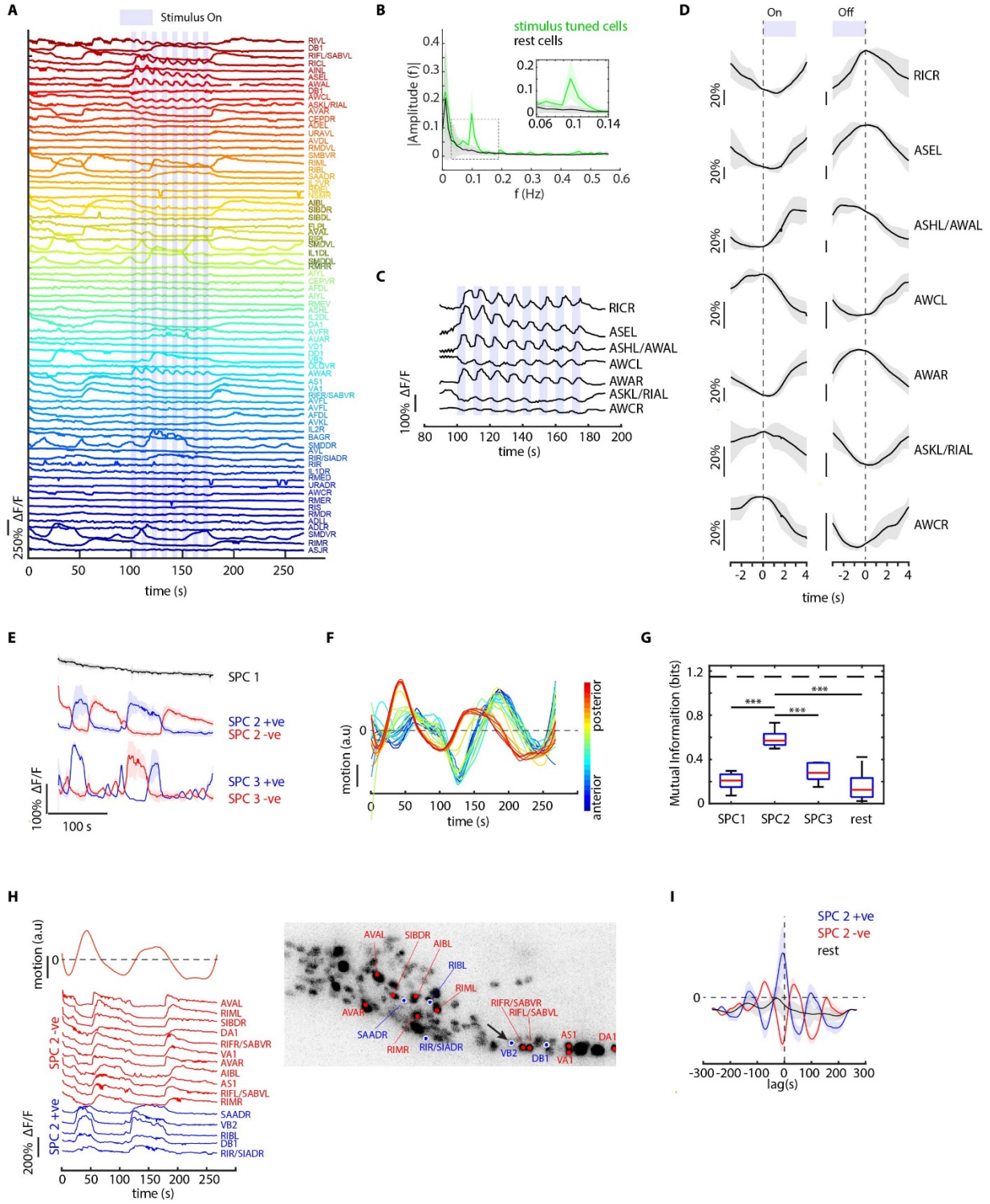


Figure 5. ***CRF_ID framework identifies neurons representing sensory and motor activities in whole-brain recording.***

A) GCaMP6s activity traces of 73 cells automatically tracked throughout a 278s long whole-brain recording and the corresponding predicted identities (top labels). Periodic stimulus

(5 sec-on – 5 sec-off) of bacteria (E. Coli OP50) supernatant was applied starting at 100 s (shaded blue regions). Experimental data comes from strain GT296.

B) Power spectrum of neuron activity traces during the stimulation period for all cells. Cells entrained by 0.1 Hz periodic stimulus show significant amplitude for 0.1 Hz frequency component (green).

C) Activity traces of cells entrained by periodic stimulus shown for the stimulation period. Blue shaded regions indicate stimulus ON, unshaded region indicate stimulus OFF. Identities predicted by the framework are labeled.

D) Average ON and OFF responses of cells entrained by periodic stimulus across trials. The black line indicates mean and gray shading indicates ± s.e.m.

E) Average activities of neurons with significant non-zeros weights in the first 3 sparse principal components (SPCs). Activities within each component are stereotypical and different components show distinct temporal dynamics. Cells with positive weights (blue) and negative weights (red) in SPC2 and SPC3 showed anti-correlated activity. Out of the 67 non-stimulus-tuned cells, 19 had non-zero weights in SPC1, 16 cells had non-zero weights in SPC2 and 5 cells had non-zero weights in SPC3. SPC1, SPC2, and SPC3 weights of cells are shown in Figure 5 – figure supplement 1. Shading indicates mean ± s.e.m of activity.

F) Velocity (motion/second) traces of cells along anterior-posterior (AP) axis (blue to red) show phase shift in velocity indicating motion in device shows signatures of wave propagation.

*G) Cells with non-zero weights in SPC2 show high mutual information with worm velocity compared to cells grouped in other SPCs (*** denotes $p < 0.001$, Bonferroni paired comparison test). Median (red line), 25th and 75th percentiles (box) and range (whiskers). Dashed line indicates entropy of velocity (maximum limit of mutual information between velocity and any random variable). Velocity of cell indicated by the black arrow in panel H right was used for mutual information analysis.*

H) Activity traces of 16 cells (with significant non-zero weights) in SPC2 and corresponding identities predicted by the framework. Red traces for cells with negative weights in SPC2, blue traces for cells with positive weights in SPC2. Worm motion/second shown on top. (Right) max projection of 3D image stack showing head ganglion neurons and cells with positive weights (blue) and negative weights (red) in SPC2. Motion/second of cell indicated with arrow is shown in left panel.

I) Cross-correlation analysis between velocity and cells with non-zero weights in SPC2 shows a strong correlation between neuron activities and velocity. In comparison, other cells show low correlation. Velocity of cell indicated by arrow in panel H right was used for cross-correlation analysis.

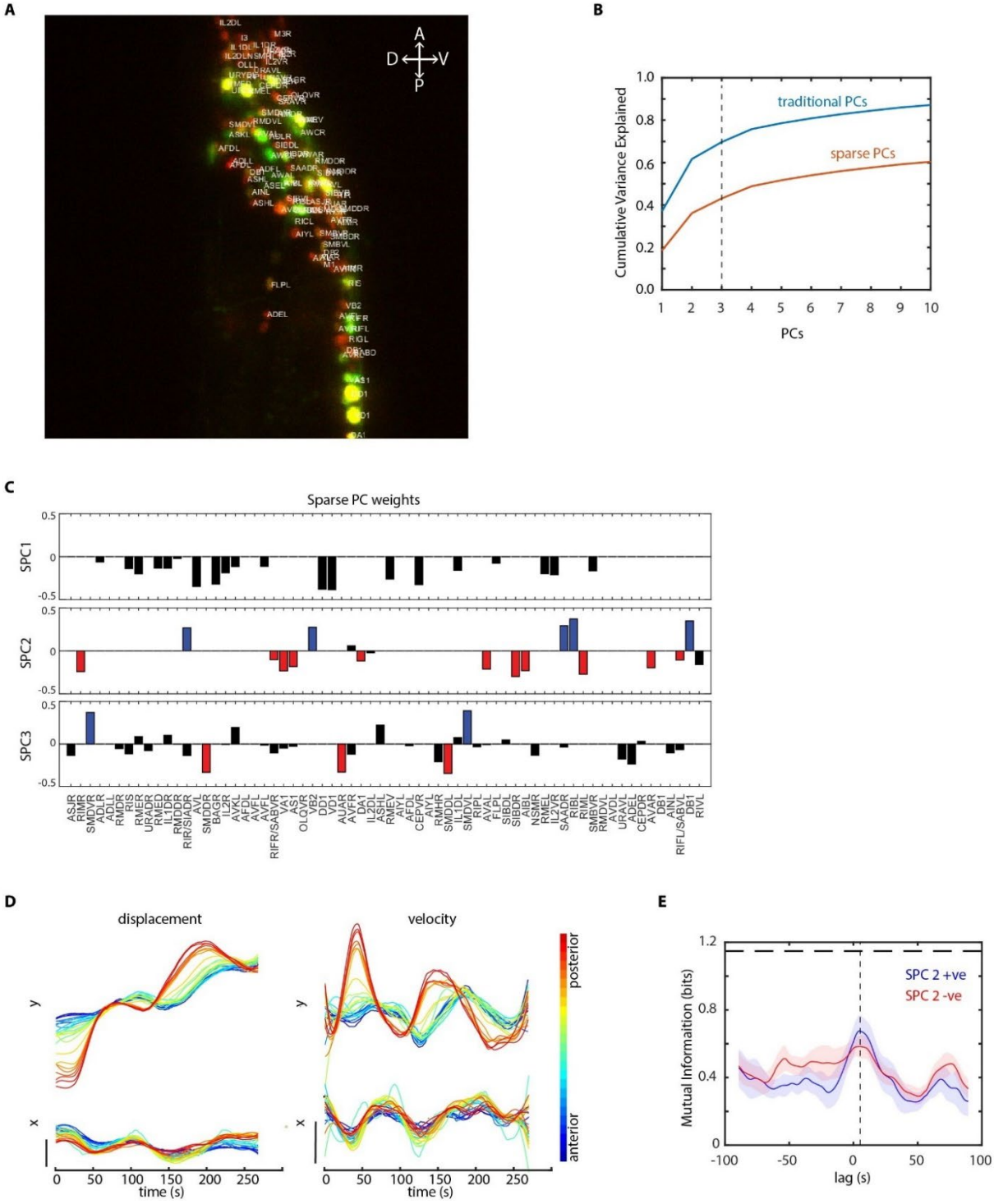


Figure 5 - figure supplement 1. ***Further analysis of data in periodic food stimulation and whole-brain imaging experiment***

A) Identities (top labels) predicted by our CRF_ID framework overlaid on the image (max-projection of image stack shown). Data comes from strain GT296.

B) Cumulative variance captured by traditional principal components (PCs) and sparse PCs. Sparse PCs capture lower variance as a tradeoff for minimizing mixing of different temporal dynamics across components to improve the interpretability of each PC.

835 *C) The weights of cells across first three sparse principal components (SPCs). Blue and red bars in SPC2 and SPC3 denote*
836 *cells with significant non-zero weights in SPC2 and SPC3. Activities of these cells are shown in Figure 5E.*

837 *D) Left panels top and bottom - Y and X displacement of randomly selected cells in the head ganglion, blue to red ordered*
838 *from anterior to posterior. Right panels top and bottom show corresponding velocities along Y and X directions.*
839 *Smooth displacement of cells and phase shift in peak velocities of cells along the AP axis show signatures of wave-*
840 *propagation in partially immobilized worm in microfluidic device.*

841 *E) Mutual information between worm velocity and lagged activities of cells grouped as SPC2 positive (blue) and SPC2*
842 *negative (red). Positive lag of neuron activities at which mutual information is maximum indicates neuron activities*
843 *precede velocity. Shading indicates mean ± sem. The horizontal dotted line indicates entropy of velocity i.e. maximum*
844 *mutual information any random variable can have with velocity.*

845

CRF framework is broadly applicable to wider conditions

Another important advantage of the CRF_ID framework is its flexibility to incorporate additional information to improve the identification accuracy, by simply adding new terms in the objective function without disturbing the weights of existing features. Here we demonstrate this idea by using the recently developed NeuroPAL[10] that provides a unique chromatic code to each neuron (Figure 6A). The chromatic code was included as a unary feature in the model (see Appendix – extended methods S2.6). Using manually curated ground-truth data, we compared different methods. These methods included different orthogonal feature combinations, as used by previous approaches, thus providing insights into which features perform best in predicting cell identities (Figure 6B, see Appendix – extended methods S2). For fair comparison across methods, static OpenWorm atlas was used across all methods. For methods that use color information, we built data-driven color atlases (Appendix – extended methods S2.4) using all datasets except the test dataset: leave-one-out color atlases. Unsurprisingly, registration performs poorly (with or without color information); color alone is not sufficient, and color combined with spatial features improves the accuracy (whether registration or relative position is used). Notably, the best performing model uses relative position features in combination with color and without registration term (Figure 6B; Figure 6 – figure supplement 1A), achieving 67.5% accuracy for the top-label prediction. Further, for 85.3% of the neurons, the true identity is within the top three labels.

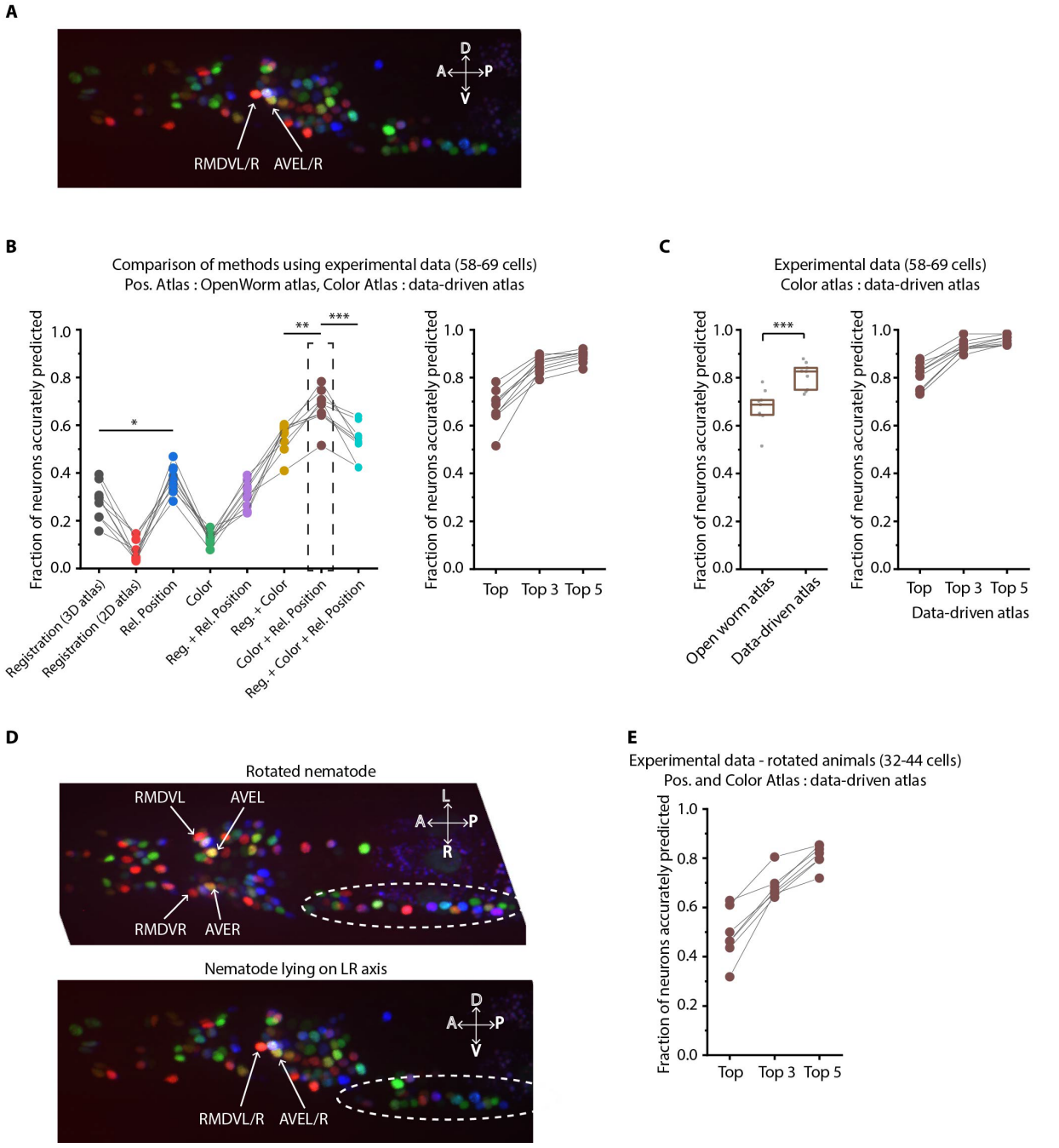


Figure 6. ***Annotation framework is generalizable and compatible with different strains and imaging scenarios.***

A) A representative image (max-projection of 3D stack) of head ganglion neurons in NeuroPAL strain OH15495.

*B) (Left) comparison of prediction accuracy for various methods that use different information. CRF_ID framework that combines relative position features along with color information performs best (n = 9 animals, * p < 0.05, *** p < 0.001, Bonferroni paired*

comparison test). (Right) the best performing method predicts cell identities with high accuracy. OpenWorm static atlas was used for all methods. Color atlas was built using experimental data with test data held out. Ensemble of color atlases that combine two different color matching methods were used for prediction. Accuracy results shown for top predicted labels. Experimental data comes from strain OH15495.

*C) (Left) annotation framework can easily incorporate information from annotated data in the form of data-driven atlas, which improves prediction accuracy (** $p < 0.01$, Bonferroni paired comparison test). Prediction was performed using leave-one-out data-driven atlases for both positional relationship features and color. Accuracy shown for top predicted labels. Ensemble of color atlases that combine two different color matching methods were used for prediction. (Right) accuracy achieved by top, top 3, and top 5 labels. Experimental data comes from strain OH15495. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.*

D) An example image of head ganglion neurons in NeuroPAL strain for rotated animal (nematode lying on DV axis). In contrast, animal lying on the LR axis is shown below. The locations of RMDVL/R, AVEL/R cells in the two images are highlighted for contrasts. Dashed ellipses indicate positions of cells in retrovesicular ganglion, showing that the rotated animal is not rigidly rotated. Experimental data comes from strain OH15495.

E) Top-label prediction accuracies for non-rigidly rotated animal. n = 7 animals. Experimental data comes from strain OH15495 and OH15500. Prediction was performed using leave-one-out data-driven atlases for both positional relationship features and color. Accuracy shown for top predicted labels. Ensemble of color atlases that combine two different color matching methods were used for prediction.

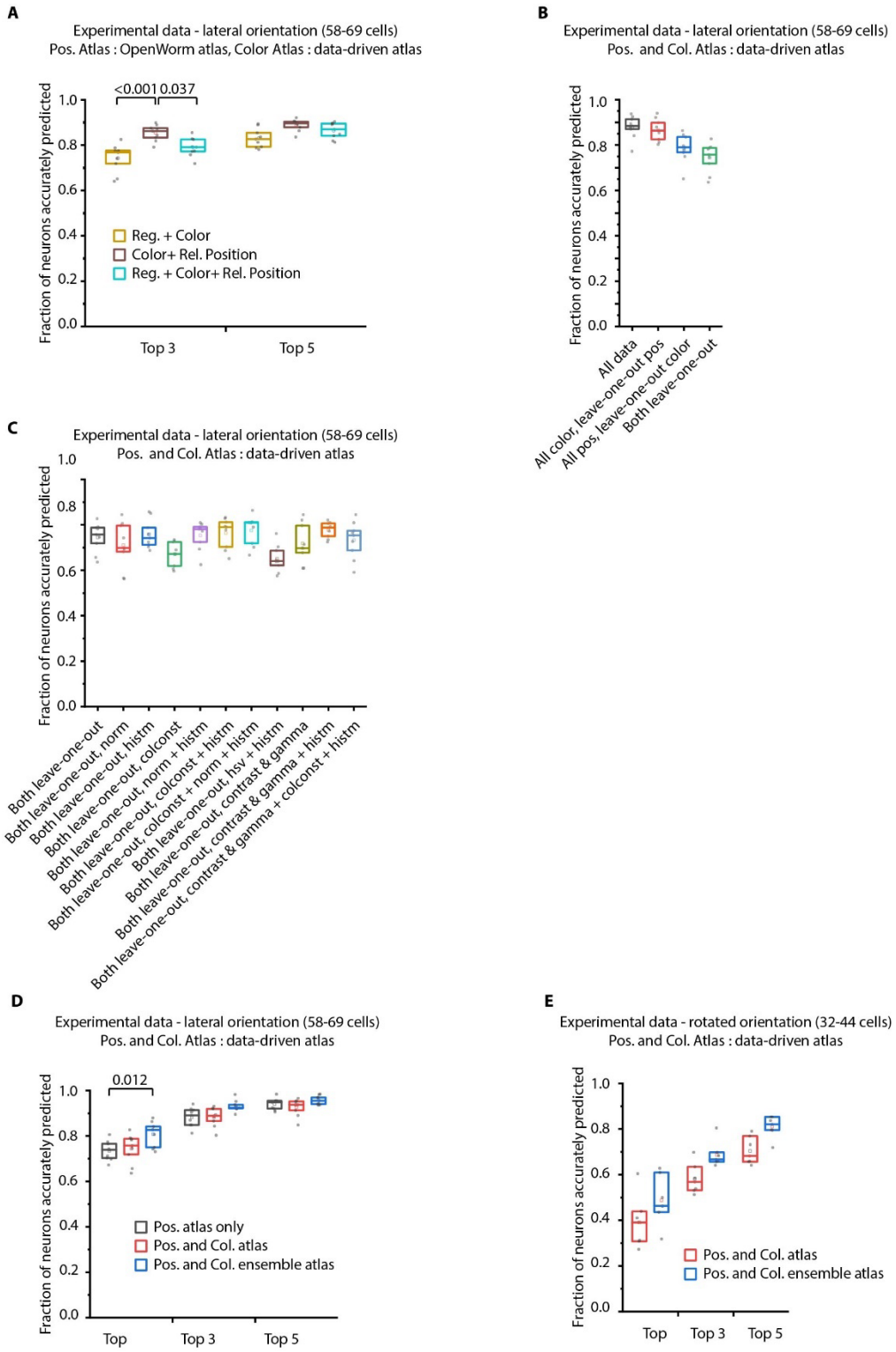


Figure 6 - figure supplement 1. ***Additional results on prediction performance of CRF_ID method on NeuroPAL data: comparison against registration method and utility of ensemble of color atlases.***

A) Comparing accuracy of top 3 and top 5 identities predicted by different methods show CRF_ID framework with pairwise positional relationship features performs better than registration method (top 3 identities case). All methods

used the same leave-one-out color atlas (see Appendix – extended methods S2.3). Experimental data comes from OH15495 strain. n = 9 animals. All comparisons performed with Bonferroni paired comparison test. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

B) Prediction accuracy of CRF_ID framework on experimental datasets across different kinds of data-driven atlases. 'All' atlas includes positional relationships and color information from all datasets including test dataset. For 'All color, leave-one-out pos.' atlas, test dataset is held out from positional relationships atlas only. For 'All pos., leave-one-out-color' atlas, test dataset is held out from color atlas only. For 'Both leave-one-out' atlas, test dataset is held out from both positional relationship and color atlases. Experimental data comes from OH15495 strain. n = 9 animals. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

C) Effect of different color distribution alignment methods on prediction accuracy. 'Both leave-one-out' case is the baseline case that uses leave-one-out atlases for both positional relationships and color, and color atlases are built by simple aggregation of RGB values across datasets. 'norm' indicates normalization of color channels, 'histm' indicates histogram matching of training datasets (images used to build atlas) to test dataset. 'colconst' indicates color invariant transformation applied to images. 'norm + histm' indicates normalization of color channels and then histogram matching of training images to test image. 'colconst + histm' indicates color invariant transformation and then histogram matching of training images to test image. 'colconst + norm + histm' indicates color normalization, subsequent color invariant transformation and finally histogram matching of training images to test image. 'hsv + histm' indicates using hsv color space instead of RGB color space and histogram matching. 'contrast and gamma' indicates contrast and gamma adjustment of image channels. 'contrast and gamma + histm' indicates contrast and gamma adjustment of image channels and subsequent histogram matching. 'contrast and gamma + colconst + histm' indicates contrast and gamma adjust of image channels, subsequent color invariant transformation and finally histogram matching. See Appendix – extended methods S2.4 for more details on methods. Experimental data comes from OH15495 strain. n = 9 animals. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

D) Comparison of prediction accuracy of CRF_ID framework across different kinds of data-driven atlases used for prediction. Test dataset was held out from data-driven atlases. 'Pos. atlas only' uses only positional relationship features atlas for prediction (these results are same as Figure 2A). 'Pos. and Col. atlas' uses positional relationship features and baseline color atlas (Appendix – extended methods S2.4) built by simple aggregation of RGB values of cells in training data used to build atlas. 'Pos. and Col. ensemble atlas' uses ensemble of 2 color atlases for prediction along with positional relationship features atlas. In this case, color distributions in training images were aligned to test data using color invariants and histogram matching prior to building atlas (Appendix – extended methods S2.4). (n = 9 animals, Bonferroni paired comparison test). Experimental data comes from OH15495 and OH15000 strains. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

E) Similar to panel B for animals non-rigidly rotated about AP axis (n = 7 animals). Test dataset was held out from data-driven atlases. In this case, Positional relationship and Color data-driven atlases were built using data from animals imaged in lateral orientation as well as rotated animals. Comparisons performed with Bonferroni paired comparison test. Experimental data comes from OH15495 and OH15000 strains. Top, middle and bottom lines in box plot indicate 75th percentile, median and 25th percentile of data respectively.

Next to assess true potential of CRF_ID framework, instead of using OpenWorm atlas, we used data-driven positional relationship atlases, so that the predictions are now performed with data-driven atlases for both positional relationships and color. To test the generalizability of the method on unseen datasets, we compared the accuracy of CRF_ID framework across several kinds of data-driven atlases (Figure 6 – figure supplement 1B). These included the following: (1) positional relationship and color atlases, which include information from all

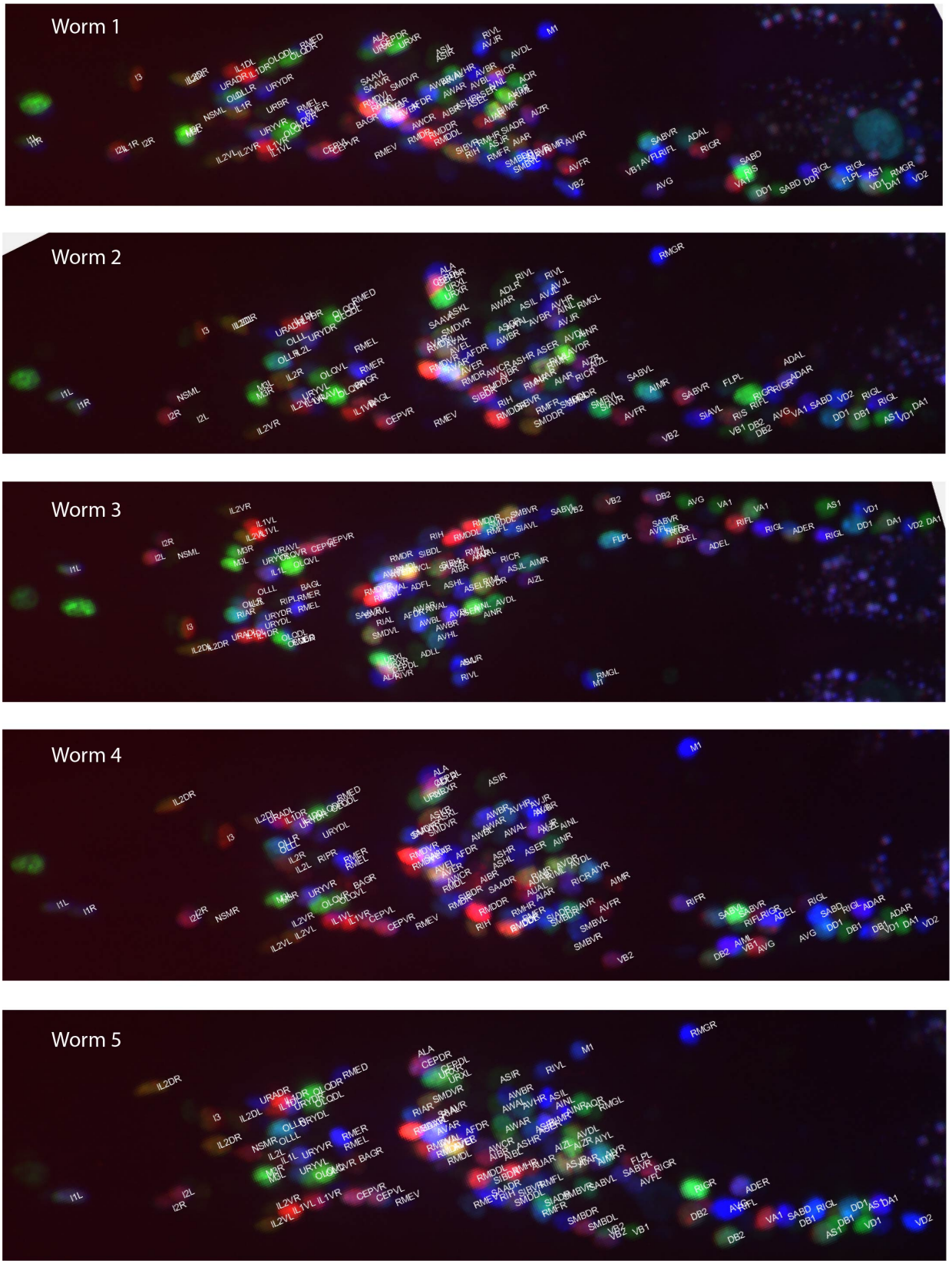
946 datasets including test dataset, (2) color information from all datasets and leave-one-out
947 atlases, which were built for positional relationships with test dataset held out, (3) positional
948 relationship information, which comes from all datasets and leave-one-out color atlases, and
949 (4) leave-one-out atlases for both positional relationships and color. The analysis revealed that
950 accuracy falls more sharply with using color leave-one-out atlases compared to the leave-one-
951 out positional relationship atlases. This implies that in the datasets used, positional relationship
952 features are more consistent compared to color features. Thus, leave-one-out positional
953 relationships atlases can represent positional relationships among cells in test datasets.
954 Further, to assess the contribution of color information to prediction accuracy, we compared
955 the accuracy of the case using both positional relationship and color leave-one-out atlases
956 (Figure 6 – figure supplement 1B last column) to the case where predictions were performed
957 using only leave-one-out positional relationship atlases shown earlier in Figure 2A. We found
958 that color contributed little to improving accuracy. This is because in the datasets used, the
959 color variability in raw RGB values across animals is far greater than the position variability
960 across animals; hence, the distribution of color features in the training data does not match the
961 distribution of features in the test data. This could be due to inherent variations in fluorophore
962 expressions across animals, or variations in imaging settings (i.e. exposure time of each channel,
963 laser power etc.) across sessions. Thus, a naive approach of building color atlas by directly
964 aggregating RGB values from training images contributed little to improving accuracy. The
965 problem of mismatched feature distributions in test data compared to training data is
966 commonly solved by domain adaptation methods in machine-learning community. We adopted
967 a simple domain adaptation strategy for dealing with color discrepancies and developed a two-

968 step method (Appendix – extended methods S2.4). First, we aligned the distributions of RGB
969 values in training datasets to the test dataset by several methods such as simple normalization
970 of color channels, histogram matching color channels in training images to test data set,
971 contrast and gamma adjustment of image channels, and transforming the color space of all
972 images with color invariants[49] (Figure 6 – figure supplement 1C). Note that this alignment does
973 not rely on cell identity information at all. These color alignment methods by themselves or in
974 combination with other methods improved accuracy for some datasets but not all datasets.
975 Second, we used an ensemble of leave-one-out color atlases for prediction, i.e. predictions
976 were performed using multiple leave-one-out color atlases each built with a different color
977 alignment technique. The ensemble, in comparison to single atlases, provides a combination of
978 color features from aligned color distributions, thus improving accuracy. The two-step method
979 improves accuracy by 6% over the naïve approach (Figure 6 – figure supplement 1D). Overall, a
980 significant improvement in the model accuracy was achieved by using data-driven atlas to
981 account for biological variability in both the positional relationships and color (Figure 6C; Figure
982 6 – figure supplement 2). Using the data-driven atlas, accuracy improved to 81% (top labels);
983 more than 93% of the neurons have their true identities in the top three labels chosen by the
984 model. We expect that more datasets for the atlas will continue to improve the accuracy.

985 Lastly, we show that our framework is equipped to work with realistic complex scenarios of
986 animals imaged in different orientations, often not rigid rotations (Figure 6D). Identifying cells
987 in these cases is challenging: manual annotation using the 2D-atlas[11] is not possible since it
988 lacks left-right information; further, due to low-z sampling of image stacks, segmented positions
989 of cells along z-axis are noisier. These challenges can be addressed by using the data-driven

990 atlas. We manually annotated data collected for animals imaged with varying degrees of non-
991 rigid rotations and built data-driven atlases for positional relationships and color. Further, we
992 combined rotated animals' atlas with the previous atlas built from animals imaged in lateral
993 orientation to build a super atlas. With the test data held out in atlases, the prediction accuracy
994 of top labels was 48.8%; 68.7% for top three labels (Figure 6E)) which are reasonable for
995 practical purposes. In this case too, aligning the color distributions in atlas to the test data set
996 and using ensemble of color atlases with different alignment techniques helped in significantly
997 improving accuracy over the naïve approach to build color atlases (Figure 6 – figure supplement
998 1E).

Automatic annotation results
Worm lying on LR axis



Automatic annotation results
Worm lying on LR axis

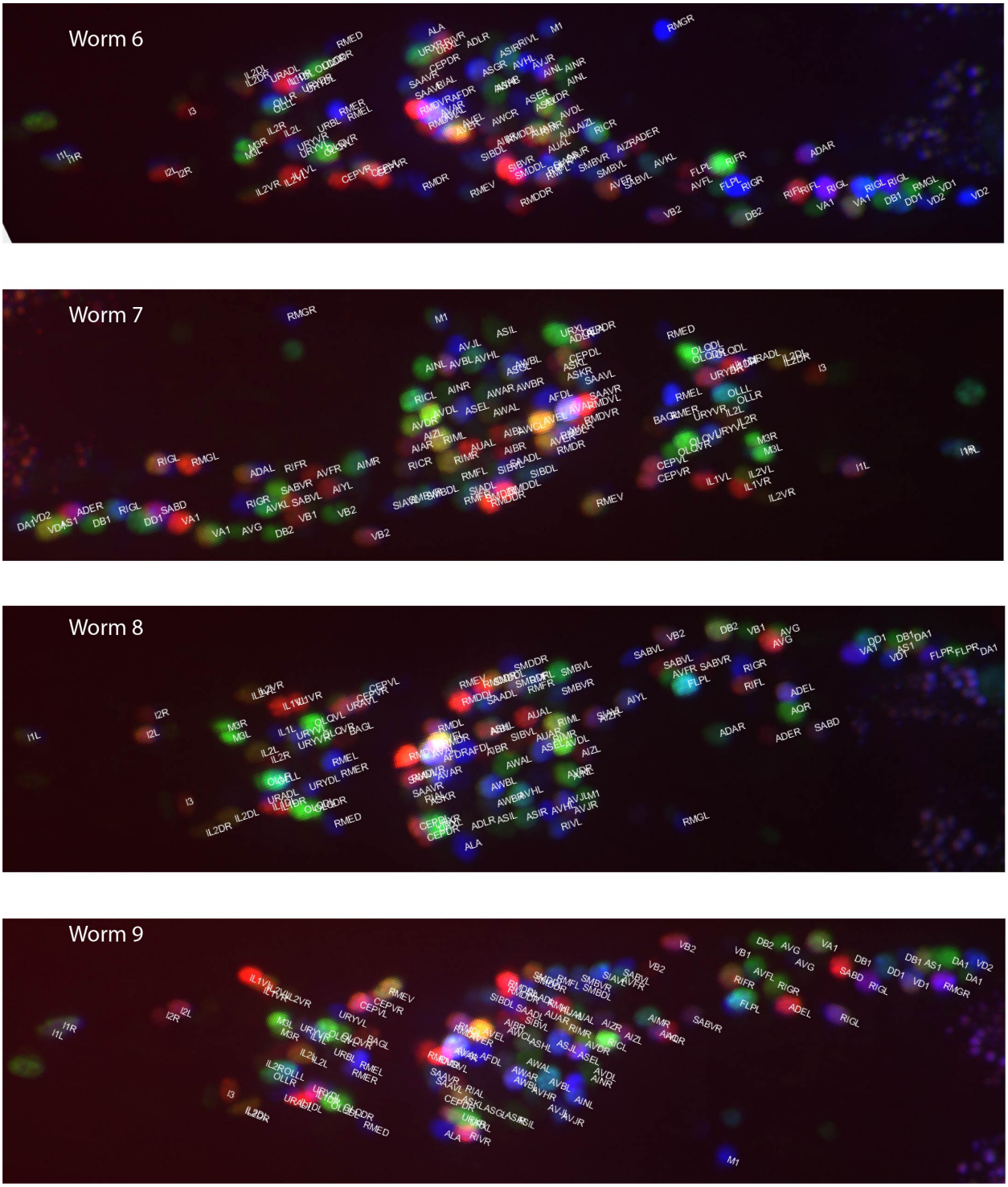
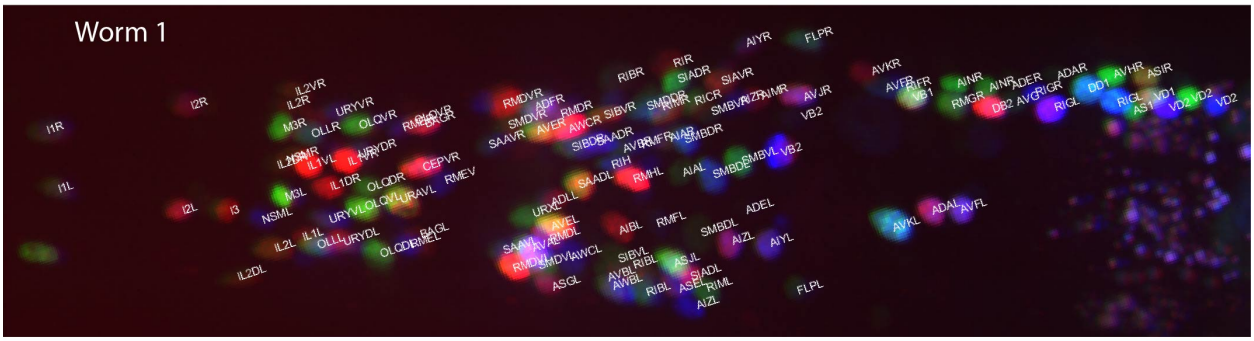
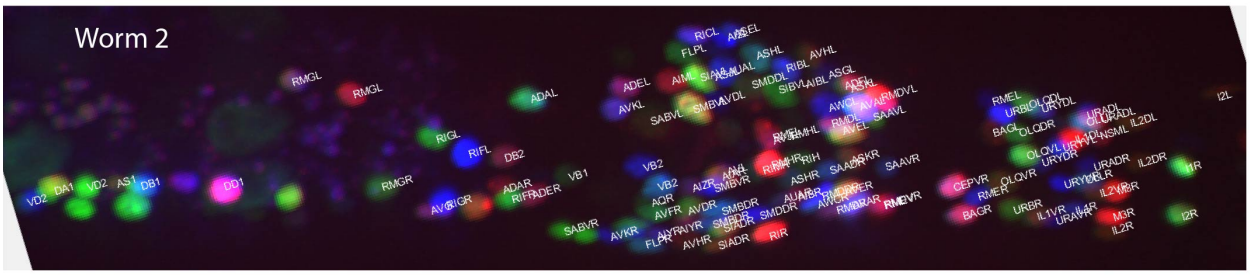


Figure 6 - figure supplement 2. ***Example annotations predicted by the CRF_ID framework for animals imaged lying on the LR axis.*** *Data comes from OH15495 strain.*

Automatic annotation results for rotated worms
Worm lying on DV axis





Automatic annotation results for rotated worms
Worm lying on DV axis

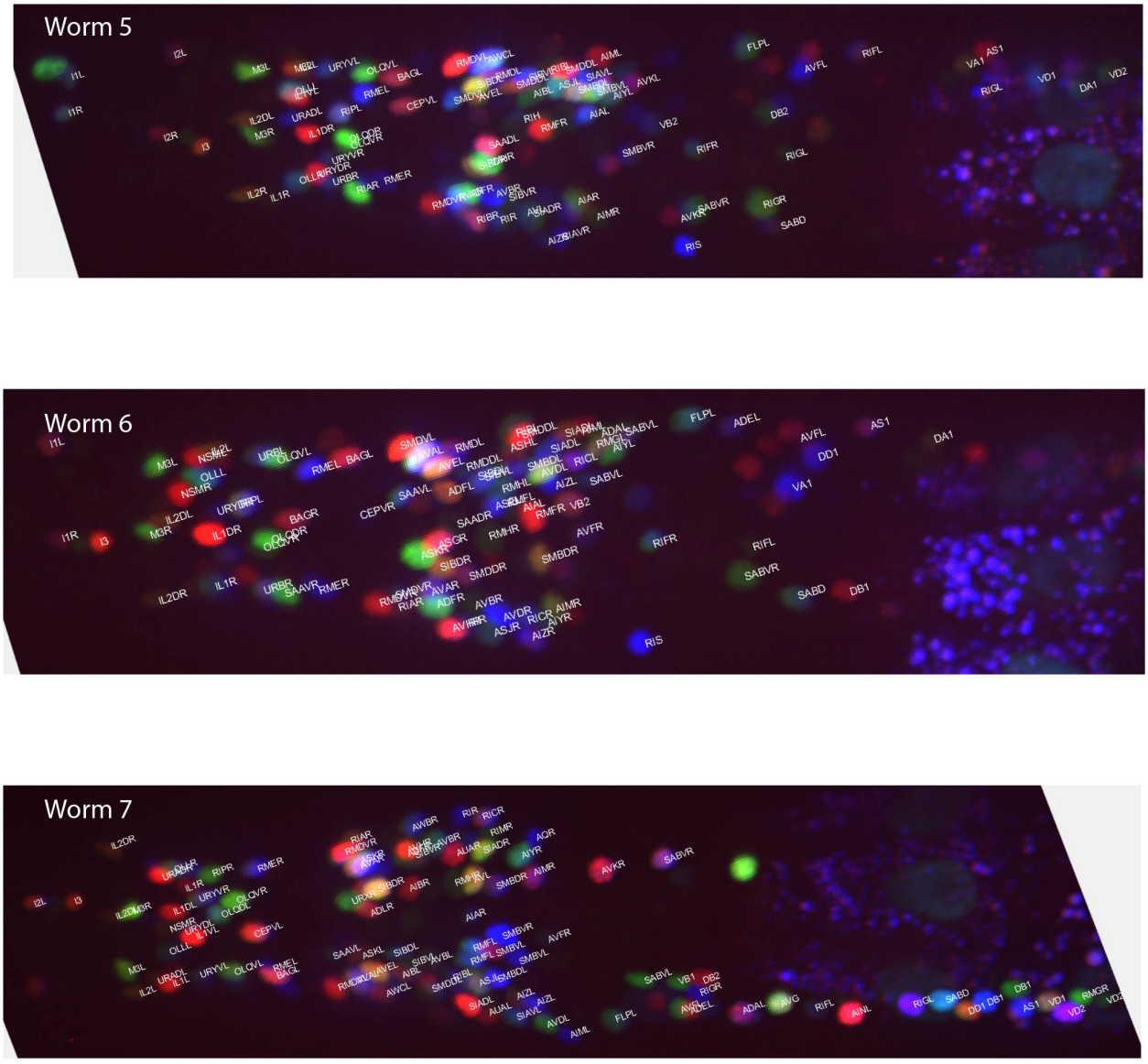


Figure 6 - figure supplement 3. ***Example annotations predicted by the CRF_ID framework for animals twisted about the anterior-posterior axis (note the anterior and lateral ganglions show clear left-right separation whereas retrovesicular ganglion instead of being in the middle is more towards one of the left or right sides).*** *Data comes from OH15495 and OH15500 strains.*

1014 # Discussion

1015 Annotating anatomical features and cellular identities in biological images are important
1016 tasks for many applications. Here, we demonstrated our CRF_ID framework is suitable for
1017 fluorescently labeled cells in 3D images for many applications. Using both ground-truth
1018 experimental data of whole-brain image stacks and synthetic data generated from an atlas, we
1019 showed that our framework is more accurate compared to existing approaches. We
1020 demonstrated using real examples how the pipeline can be used for analysis of gene expression
1021 pattern for instance, and for neuron identification from dense multi-cell or whole-brain imaging
1022 experiments. Further, our CRF_ID framework significantly speeds up the cell identification
1023 compared to manual labeling while reducing bias.

1024 With the pipeline, we address several challenges. There is ample evidence that anatomy
1025 varies from individual to individual, and from condition to condition. This variability, or position
1026 noise, is a major source of roadblock in effectively applying previous methods to annotate the
1027 whole-brain recording data. Because our framework leverages intrinsic similarity[19], it performs
1028 better than registration methods in handling position noise (Figure 2E; Figure 2 – figure
1029 supplement 5C). Further, CRF_ID formulation is more accurate in handling count noise i.e.
1030 cases of missing or undetectable cells in images (Figure 2D; Figure 4C; Figure 2 – figure
1031 supplement 5B; Figure 4 – figure supplement 1A), because the missing neurons do not upset
1032 the relationships among the detectable neurons in the CRF_ID formulation while missing
1033 neurons introduce large uncertainty in registration methods. Lastly, CRF_ID method predicts
1034 identities with sufficient accuracy for different postural orientations of the worms often seen in

1035 our microfluidic experiments. We expect that this superiority is maintained for any data that
1036 have relational information preserved, this is the case virtually in all biological samples where
1037 tissues are connected by matrix materials, such as in other whole-brain recordings or for
1038 registration of fixed tissues.

1039 Building and using data-driven atlases in the pipeline is simple and yet highly effective.
1040 We expect that data from more animals, different orientations, age, and imaging techniques
1041 will further improve the generalizability. Since building such data-driven atlas for our
1042 framework requires only cheap mathematical operations (Appendix – extended methods S1.7),
1043 incorporating more data is quite simple and easily scalable. In contrast, previous registration-
1044 based methods may require simultaneous or batch-wise registration of multiple images to one
1045 reference image; this would require solving multiple constrained regression problems on
1046 increasingly large data sets, thus rendering them computationally unscalable. Further, without
1047 systematic methodology of which image should be chosen as reference image, atlas gets biased
1048 towards the chosen reference image or by the order in which blocks of images are registered to
1049 the reference image. Tackling this challenge is an active field of research[50,51]. In comparison, in
1050 CRF method, atlas building is unbiased towards any image set because there is no concept of
1051 reference image. Additionally, atlas can be built from all images simultaneously because of the
1052 cheapness of mathematical operations.

1053 Another major advantage of data-driven atlases in our framework is that the atlases can
1054 be built incrementally using partially or fully annotated datasets, e.g. using lines that label
1055 partial and distinct subsets of cells. In comparison, registration-based methods cannot build

atlas from lines that label distinct subset of cells. This is because registration-based methods build probabilistic spatial atlases by first establishing correspondence among cells in images and subsequently registering images. However, this is not possible if the cells in different images do not have any overlapping cells or have very few overlapping cells. In comparison, atlas built in CRF_ID framework store probabilistic positional relationship features among cells observed within each image. Hence, correspondence between images is not required. Thus, in principle, CRF_ID framework can combine manually annotated data across different lines, generated by different researchers (and across labs) in the form of data-driven atlases. Automated annotation using such atlases removes individual biases in annotating cells. Further, it greatly supports researchers with no prior experiences with cell identification. We expect that using our framework, large-scale atlases can be built in the future with community contributed data.

Finally, a distinction of CRF_ID framework is its ability to build and annotate with complete atlases covering all cells, with the efficient utilization of data even from strains with non-overlapping cells. Annotating against a complete atlas is crucial because commonly in practice, no prior information is available on exactly which cells are missing from the images before annotation. Registration-based or unary potential-based methods are limited in building atlas by the availability of overlapping strains. Thus, in these methods, cells that are missing in the atlas can never be assigned to cells in images; hence these methods do not perform completely unbiased annotation. In comparison, CRF_ID framework uses a complete atlas to assign any possible label in the atlas to cells in the images, thus performing unbiased annotation, resulting in better handling of count noise in images.

1077 CRF framework offers the flexibility of combining arbitrary unary features with arbitrary
1078 pairwise features for cell annotation task. We demonstrate the utility of such flexibility by
1079 combining color information of cells in NeuroPAL strain with positional relationship features
1080 and show higher accuracy compared to other methods. Our experiments show that in order to
1081 be able to utilize color information of cells in NeuroPAL for automatic annotation of cell
1082 identities, color consistency across animals needs to be maintained, either experimentally or by
1083 post hoc corrections. Experimentally, consistent protocol/imaging settings across sessions
1084 should be maintained as much as possible. Even with consistent protocol, color variation may
1085 exist due to inherent differences across animals in relative expressions of fluorophores that
1086 define the chromatic code of cells. This can be tackled by 1) collecting large volume of data to
1087 capture each cells' full RGB variations, and 2) using computational domain adaptation
1088 techniques. More advancement in image color transfer and domain adaptation techniques will
1089 further improve accuracy in future.

1090 While we only considered pairwise features in the current formulation, feature
1091 functions with arbitrary dependency can be included in the model that may further improve
1092 prediction accuracy[27,52]. Advances in structured energy minimization field[27,53,54] will enable
1093 tackling the increased complexity of combinatorial optimization in these cases. Our workflow
1094 borrows techniques developed in metric object/shape matching literature for annotation in
1095 biological images. Log-linear parameterization in our framework makes the model a member of
1096 the exponential families[55]; thus, the objective function in our framework has interesting
1097 connections with max-entropy models and with the entropy-regularized optimal transport
1098 objective functions[30,56]. Therefore, improvements in computational speed can be achieved by

1099 borrowing fast optimization techniques for quadratic assignment problems developed in
1100 optimal transport literature. Advances in these fields will continue to improve the method
1101 development in image analysis.

1102 We anticipate that by using our generalizable formulation, similar pipelines can be set
1103 up to annotate more image sets in other organisms and build atlases. Data in many anatomical
1104 annotation problems (e.g. brain atlas construction, registering images from different
1105 modalities, comparing animals or related species to one another for developmental studies)
1106 share a similar property, in that the anatomical features of interest maintain a cohesion from
1107 sample to sample. This underlining cohesion lends itself to the CRF framework. As we have
1108 shown, the pipeline is extremely flexible in incorporating new information. Thus, framework
1109 should be easily modifiable catering to the data demands in other organisms including features
1110 besides landmarks and spectral information such as cellular morphology and expected cellular
1111 activities (e.g. calcium transients). Because the only inputs to our framework are segmented
1112 anatomical regions in images and positional relationships among them, information already
1113 available in data across organisms[57–60], the framework proposed here should be generally
1114 useful for many problems in model organisms such as *Drosophila*[57,61], zebrafish[60], mammalian
1115 brains[58,59]. Besides fluorescence, the pipeline should also be able to work with data from other
1116 modalities including EM, live imaging, and fluorescence imaging from cleared tissues.

1117 # Materials and Methods

1118 ## Reagents

1119 For all experiments, animals were cultured using standard techniques[62]. A detailed list of strains

1120 used is provided below

Name	Genotype	Experiments	Reference
GT290	aEx18[unc-47p::NLS::CyOFP1::egl-13NLS]	Strain with 9 neuronal landmarks in head	This work
GT298	aEx22[unc-47p::NLS::CyOFP1::egl-13NLS + gcy-32p::NLS::CyOFP1::egl-13NLS]	Strain with 12 neuronal landmarks in head	This work
AML32	wtfIs5 [rab-3p::NLS::GCaMP6s + rab-3p::NLS::tagRFP]	Strain used to make whole-brain imaging strain with CyOFP labelled landmarks GT296	[7]
AML70	wtfIs5 [rab-3p::NLS::GCaMP6s + rab-3p::NLS::tagRFP]; lite-1(ce314) X	Strain used to make whole-brain imaging strain with CyOFP labelled landmarks GT293	[18]
KG1180	lite-1(ce314) X	Strain used to make whole-brain imaging strain with CyOFP labelled landmarks GT296	CGC
GT296	wtfIs5 [rab-3p::NLS::GCaMP6s + rab-3p::NLS::tagRFP]; aEx18[unc-47p::NLS::CyOFP1::egl-13NLS]; lite-1(ce314) X	Strain used for whole-brain functional imaging experiments (Figures 4, 5) and quantifying cell position variability.	This work
GT293	wtfIs5 [rab-3p::NLS::GCaMP6s + rab-3p::NLS::tagRFP]; lite-1(ce314) X; aEx22[unc-	Strain used for quantifying cell position variability.	This work

	47p::NLS::CyOFP1::egl-13NLS + gcy-32p::NLS::CyOFP1::egl-13NLS]		
AML 5	otIs355 [rab-3p(prom1)::2xNLS::TagRFP] IV. kyIs51 [odr-2p::GFP + lin-15(+)]	Strain used for mock gene-expression pattern analysis and mock multi-cell calcium imaging experiments	[7]
OH15495	otIs696 [UPN::NLS::TagRFP-T + acr-5::NLS::mTagBFP2::H2B + flp-1::NLS::mTagBFP2::H2B + flp-6::NLS::mTagBFP2::H2B + flp-18::NLS::mTagBFP2::H2B + flp-19::NLS::mTagBFP2::H2B + flp-26::NLS::mTagBFP2::H2B + gcy-18::NLS::mTagBFP2::H2B + ggr-3::NLS::mTagBFP2::H2B + lim-4::NLS::mTagBFP2::H2B + pdfr-1::NLS::mTagBFP2::H2B + srab-20::NLS::mTagBFP2::H2B + unc-25::NLS::mTagBFP2::H2B + cho-1::NLS::CyOFP1::H2B + flp-13::NLS::CyOFP1::H2B + flp-20::NLS::CyOFP1::H2B + gcy-36::NLS::CyOFP1::H2B + gpa-1::NLS::CyOFP1::H2B + nlp-12::NLS::CyOFP1::H2B + nmr-1::NLS::CyOFP1::H2B + ocr-1::NLS::CyOFP1::H2B + osm-9::NLS::CyOFP1::H2B + srh-79::NLS::CyOFP1::H2B + sri-1::NLS::CyOFP1::H2B + srsx-3::NLS::CyOFP1::H2B + unc-8::NLS::CyOFP1::H2B + acr-2::NLS::mNeptune2.5 + ceh-2::NLS::mNeptune2.5 + dat-1::NLS::mNeptune2.5 + dhc-3::NLS::mNeptune2.5 + eat-4::NLS::mNeptune2.5 + flp-3::NLS::mNeptune2.5 + gcy-35::NLS::mNeptune2.5 + glr-1::NLS::mNeptune2.5 + gcy-21::NLS::CyOFP1::H2B::T2A::NLS::mTagBFP2::H2B + klp-6::NLS::mNeptune2.5::T2A::NLS::CyOFP1::H2B + lim-6::NLS::mNeptune2.5::T2A::NLS::CyOFP1::H2B + mbr-	NeuroPAL strain demonstrating the ease of incorporating color information, and thus demonstrating generalizability	[10]

	1::NLS::mNeptune2.5::T2A::NLS::mTagBFP2::H2B + mec-3::NLS::CyOFP1::H2B::T2A::NLS::mTagBFP2::H2B + odr-1::NLS::mNeptune2.5::T2A::NLS::mTagBFP2::H2B + srab-20::NLS::mNeptune2.5::T2A::NLS::mTagBFP2::H2B]		
OH15500	otIs672 [rab-3::NLS::GCaMP6s + arrd-4:NLS:::GCaMP6s]. otIs669 [UPN::NLS::TagRFP-T + acr-5::NLS::mTagBFP2::H2B + flp-1::NLS::mTagBFP2::H2B + flp-6::NLS::mTagBFP2::H2B + flp-18::NLS::mTagBFP2::H2B + flp-19::NLS::mTagBFP2::H2B + flp-26::NLS::mTagBFP2::H2B + gcy-18::NLS::mTagBFP2::H2B + ggr-3::NLS::mTagBFP2::H2B + lim-4::NLS::mTagBFP2::H2B + pdfr-1::NLS::mTagBFP2::H2B + srab-20::NLS::mTagBFP2::H2B + unc-25::NLS::mTagBFP2::H2B + cho-1::NLS::CyOFP1::H2B + flp-13::NLS::CyOFP1::H2B + flp-20::NLS::CyOFP1::H2B + gcy-36::NLS::CyOFP1::H2B + gpa-1::NLS::CyOFP1::H2B + nlp-12::NLS::CyOFP1::H2B + nmr-1::NLS::CyOFP1::H2B + ocr-1::NLS::CyOFP1::H2B + osm-9::NLS::CyOFP1::H2B + srh-79::NLS::CyOFP1::H2B + sri-1::NLS::CyOFP1::H2B + srsx-3::NLS::CyOFP1::H2B + unc-8::NLS::CyOFP1::H2B + acr-2::NLS::mNeptune2.5 + ceh-2::NLS::mNeptune2.5 + dat-1::NLS::mNeptune2.5 + dhc-3::NLS::mNeptune2.5 + eat-4::NLS::mNeptune2.5 + flp-3::NLS::mNeptune2.5 + gcy-35::NLS::mNeptune2.5 + glr-1::NLS::mNeptune2.5 + gcy-21::NLS::CyOFP1::H2B::T2A::NLS::mTagBFP2::H2B + klp-6::NLS::mNeptune2.5::T2A::NLS::CyOFP1::H2B + lim-6::NLS::mNeptune2.5::T2A::NLS::CyOFP1::H2B + mbr-1::NLS::mNeptune2.5::T2A::NLS::mTagBFP2::H2B +	NeuroPAL strain demonstrating the ease of incorporating color information, and thus demonstrating generalizability	[10]

	mec-3::NLS::CyOFP1::H2B::T2A::NLS::mTagBFP2::H2B + odr-1::NLS::mNeptune2.5::T2A::NLS::mTagBFP2::H2B + srab-20::NLS::mNeptune2.5::T2A::NLS::mTagBFP2::H2B] V		

1121

1122 ## Imaging

1123 All imaging was performed using either a Perkin Elmer spinning disk confocal microscope (1.3
1124 NA, 40x, oil objective) or Brucker Opterra II Swept field confocal microscope (0.75 NA, 40x, Plan
1125 Fluor air objective), with an EMCCD camera.

1126 To acquire data used for framework validation and comparison against other methods (Figure
1127 2), gene expression pattern analysis (Figure 3), multi-cell calcium imaging (Figure 4), imaging
1128 landmark strain (Figure 4) and NeuroPAL imaging (Figure 6), animals were synchronized to L4
1129 stage and were imaged in an array microfluidic device[63]. A single 3D stack was acquired with
1130 either 0.5 µm or 1 µm spacing between z-planes and 10 ms exposure time (except for NeuroPAL
1131 strain where exposure times of different channels were chosen based on the guidelines
1132 provided in NeuroPAL manuals[10]).

1133 Whole-brain functional recording data while providing chemical stimulus were acquired using a
1134 microfluidic device designed for applying chemical stimulation[44] to the nose-tip of the animal.
1135 Here image stacks were acquired with 1 µm spacing between z-planes and 10 ms exposure for
1136 each z-plane. This enabled recording videos at 1.1 volumes/s while imaging two channels
1137 simultaneously (GCaMP and RFP). Animals were synchronized to Day-1 adult stage.

1138 Generating synthetic data for framework tuning and comparison against other methods

1139 Synthetic data was generated using the freely available 3D atlas at OpenWorm[12]. Atlas available
1140 at Worm Atlas[11] was not used as it provides only a 2D view. To mimic the conditions
1141 encountered in experimental data, two noise perturbations were applied to the 3D atlas (Figure
1142 2 – figure supplement 2). First, due to inherent biological variability, positions of cells observed
1143 in images do not exactly match the positions in atlas. Thus, position noise was applied to each
1144 cell in the atlas. This noise was sampled from a normal distribution with zero mean and fixed
1145 variance $\Sigma = diag([\sigma_x, \sigma_y, \sigma_z])$. Here σ_x, σ_y and σ_z denote variances along x, y and z image
1146 dimensions and $diag(\mathrm{x})$ denotes diagonalizing vector x. Hence, the position of i^{th}cell $p_i \in \mathbb{R}^3$
1147 in synthetic data was defined as $p_i = p_{i,atlas} + \epsilon, \epsilon \sim \mathcal{N}(0, \Sigma)$. Here $p_{i,atlas}$ is the position of the
1148 i^{th} cell in the atlas. To determine the variance Σ, we quantified the variance of cell positions
1149 observed in experimental data (Figure 2 – figure supplement 2A, C, E) using the strains GT293,
1150 GT295 with neuronal landmarks. We calculated the 25th percentile and 75th percentile of the
1151 variance across all cells across all animals (n = 31) to define the lower bound and upper bound
1152 position noise levels observed in experimental data. However, this variability cannot be directly
1153 applied to the atlas due to different spatial scales in images and atlas. Thus, we first normalized
1154 the observed variance of cell positions in images with inter-cell distances in images and then
1155 scaled it according to the inter-cell distances in atlas (Figure 2 – figure supplement 2B,D,F,G,H)
1156 to define lower bound and upper bound noise to be applied to the atlas. More position noise
1157 levels such as those in Figure 2E and Figure 2 – figure supplement 5 were set by varying Σ
1158 between zero and upper-bound noise level.

1159 Second, although there are 195-200 neurons in head ganglion in *C. elegans,* only 100-130 cells
1160 were detected in most image stacks. Remaining cells are not detected either due to low-
1161 expression levels of fluorophores, variability in expression levels of genetic marker used to label
1162 cells (mosaicism, incomplete coverage etc.) or inability of segmentation methods to resolve
1163 densely packed cells. This increases the complexity of determining the labels of cells. To
1164 illustrate, matching 195 cells in an image to 195 cells in the atlas is easier as only one or very
1165 few possible configurations of label assignments exist that maximally preserves the positional
1166 relationships among cells. In contrast, in the case of matching 100 cells in an image to 195 cells
1167 in atlas, many possible labeling arrangements may exist that will equally preserve the positional
1168 relationships among cells. Thus, to simulate count noise in synthetic data, randomly selected
1169 cells in atlas were marked as missing and synthetic data was generated from the atlas with
1170 remaining cells. Hence, identities were predicted for remaining cells only in synthetic data using
1171 the full atlas. Number of cells set as missing was set by the count noise level parameter, defined
1172 as the fraction of total cells in the atlas that were set as missing. Since no prior information was
1173 available on which regions of the head ganglion had more cells missing, we selected the missing
1174 cells uniformly across brain regions.

1175 Finally, bounds on prediction accuracy (shown as gray regions in Figure 2, Figure 2 – figure
1176 supplement 1) were obtained as the average prediction accuracy across runs obtained on
1177 synthetic data by applying lower bound and upper bound position noise.

1178 ## Generating ground-truth data for framework tuning and comparison against other
1179 methods

1180 NeuroPAL reagents OH15495 and OH15500 were used to generate ground-truth data. 3D image
1181 stacks were acquired following the guidelines provided in NeuroPAL manual[10]. Identities were
1182 annotated in image stacks using the example annotations provided in NeuroPAL manual.
1183 Individual channel image stacks were read in MATLAB, gamma and contrast were adjusted for
1184 each channel individually so that the color of cells in the RGB image formed by combining the
1185 individual channels matched as much as possible (perceptually) with the colors of cells in
1186 NeuroPAL manuals. To annotate identities in the 3D stack, Vaa3D software was used[32].

1187 ## Model comparison against previous methods

1188 Detailed description of the methodology used for each method that our CRF_ID framework was
1189 compared against is provided in Appendix – extended methods S2. Note, for fair comparisons,
1190 standard 3D OpenWorm atlas was used by all methods as the reference: either for defining
1191 positions of cells (used by registration methods) or for defining positional relationships among
1192 cells (used by the CRF_ID framework).

1193 ## Simulations for choosing landmark locations

1194 Landmarks (cell with known identities) improve prediction accuracy by constraining the
1195 optimization problem as it forces the CRF_ID framework to choose optimal labels for all cells
1196 such that they preserve their positional relationships with the cells with fixed identities.
1197 However, choosing an optimal set of landmarks is difficult. This is because the combinatorial
1198 space of choosing landmarks is huge (~10^{14} for 10 landmark cells out of 195 in head ganglion).

1199 Simulating each such combination and predicting identities is not computationally tractable.
1200 Thus, we asked which regions of the brain landmark cells should lie in. We divided the head
1201 ganglion region into three groups: anterior group consisting of anterior ganglion, middle group
1202 consisting of lateral, dorsal and ventral ganglion, and posterior group consisting of
1203 retrovesicular ganglion. Two hundred runs were performed for each group with 15 randomly
1204 selected landmarks in each run. We constrained the landmarks cells to lie in a specific group
1205 and assessed how well the landmarks in that group perform in predicting the identities of cells
1206 in other regions. Overall, landmarks in anterior and posterior groups performed badly in
1207 predicting identities of cells in posterior and anterior groups respectively. Landmarks in the
1208 middle group and landmarks spatially distributed throughout the head performed equally
1209 (Figure 4 – figure supplement 2). We chose landmarks spatially distributed throughout the head
1210 due to practical advantages: spatially distributed landmarks can be easily identified manually in
1211 image stacks thus can be used as inputs to the CRF_ID framework. In contrast cells in middle
1212 group are densely packed and may not be identified easily. We tested this using several
1213 reporter strains with GFP labeled cells. Further, landmarks should be reliably expressed across
1214 animals, should have known and verified expression patterns and should label neither too few
1215 cells (not useful) nor too many cells (difficult identification). Thus, we chose *unc-47* and *gcy-32*
1216 reporters for labeling landmarks.

1217 Construction of landmark strains

1218 We constructed two different transgenic strains in which nine (GT290) and twelve (GT298)
1219 neurons, respectively, were labelled with the fluorescent protein CyOFP1[64]. Due its long Stokes
1220 shift, CyOFP1 can be excited by the same laser line as GCaMP, while emitting red-shifted

1221 photons. This conveniently allows us to perform three-color imaging on our two-channel
1222 confocal microscope. We designed an optimized version of the CyOFP1 gene using the *C.*
1223 *elegans* Codon Adapter[65], which was then synthesized (Integrated DNA Technologies) and sub-
1224 cloned into a kanamycin-resistant vector to yield the pDSP11 plasmid. Our CyOFP1 construct
1225 contains two different nuclear localization sequences (NLS), SV40 NLS at the N-terminus and
1226 EGL-13 NLS at the C-terminus, a strategy which has been shown to more effective at trafficking
1227 recombinant proteins to the nucleus of worm cells[66]. The nuclear localization of the CyOFP1
1228 protein allows us to unambiguously identify labelled cells in the densely packed head ganglion
1229 of *C. elegans*.

1230 We tested two different labelling strategies in our study. The first used the promoter of the
1231 *unc-47* gene to drive expression CyOFP1 in all 26 GABAergic neurons of the worm[67]. As our
1232 study focused on the head ganglion, only the RMEL, RMER, RMEV, RMED, AVL, RIS, DD1, VD1,
1233 and VD2 neurons are labelled by this promoter in this region (Strain GT296, Figure 4D top
1234 panel). Our second strategy used the *unc-47* CyOFP1 construct in combination with a second
1235 driven by the promoter of the *gcy-32* gene, which is expressed in the AQR, PQR, and URX
1236 neurons[68], to label twelve cells in the head ganglion (Strain GT293, Figure 4D bottom panel).
1237 The DNA sequence of each promoter was amplified from N2 (wild type) genomic DNA and
1238 incorporated into a NotI-digested linear pDSP11 vector backbone using the NEBuilder HiFi DNA
1239 Assembly master mix (New England Biolabs) to yield the vectors pSC1 and pSC2. The following
1240 primers were used to amplify the promoters: *unc-47* Forward 5'-
1241 cagttacgctggagtctgaggctcgtcctgaatgatatgcCTGCCAATTTGTCCTGTGAATCGT-3' and Reverse 5'-
1242 gcctctcccttggaaaccatCTGTAATGAAATAAATGTGACGCTGTCGT, *gcy-32* Forward 5'-

1243 cagttacgctggagtctgaggctcgtcctgaatgatatgcTTGTATAGTGGGAAATACTGAAATATGAAACAAAAAATA

1244 TAGCTG-3' and Reverse 5'- gcctctcccttggaaaccatTCTATAATACAATCGTGATCTTCGCTTCGG-3'.

1245 To make landmark strains pSC1 and pSC2 were injected into N2 strain to make GT290 and

1246 GT298. GT290 and GT298 strains can be crossed with any strain where cells need to be

1247 identified. Landmarks in these strains help in defining a coordinate system in head and also

1248 improve the accuracy of automatic annotation framework by constraining optimization

1249 problem. To make strain GT293 for whole-brain imaging experiments, AML70 was crossed with

1250 GT298; *lite-1(ce314)* was confirmed by sequencing. To make strain GT296 for whole-brain

1251 imaging experiments, AML32 was crossed with GT290 and subsequently crossed with KG1180,

1252 *lite-1(ce314)* was confirmed by sequencing.

1253 Whole-brain data analysis

1254 All videos were processed using custom software in MATLAB for automatic segmentation and

1255 tracking of nuclei in whole-brain image stacks. Tracks for nuclei with minor tracking errors were

1256 corrected in post-processing steps. Tracks with large tracking errors were dropped from the

1257 data.

1258 <u>Segmentation</u> - Neurons were automatically segmented in image stacks using a Gaussian

1259 Mixture model based segmentation technique. Briefly, intensity local maxima are detected in

1260 images to initialize the mixture components and subsequently a 3D gaussian mixture model is

1261 fitted to the intensity profiles in image stacks using Expectation-Maximization (EM) algorithm.

1262 The number of components in the model and the ellipsoidal shape of each component

1263 determines the number of nuclei segmented and their shapes.

1264 Tracking – Custom software was used for tracking cells. Briefly, segmented nuclei at each
1265 timepoint in image stacks are registered to a common reference frame and temporally nearby
1266 frames to produce globally and locally consistent matching. Based on these matchings,
1267 consistency constraints such as transitivity of matching were imposed in the post-processing
1268 step to further improve tracking accuracy. A custom MATLAB GUI was used to quickly manually
1269 inspect the accuracy of tracking. Tracks of cells with minor tracking errors were resolved using
1270 semi-automated method.

1271 Cell identification – Identities were predicted using the CRF_ID framework with positional
1272 features (Appendix – extended methods S1) using the data-driven atlas. Landmarks cells with
1273 known identities were identified in the CyOFP channel and were provided as input to the
1274 framework to achieve higher accuracy.

1275 Identification of stimulus tuned neurons – To identify stimulus tuned neurons, the power
1276 spectrum of activities of all cells within the stimulus application window (100 s – 180 s) was
1277 calculated using "fft" function in MATLAB. Cells that showed significant power (> 0.08) at 0.1 Hz
1278 (due to 5s on 5s off stimulus, Figure 5) were selected. This criterion identified all cells except
1279 two with low response amplitude to the stimulus however the response could be manually
1280 seen in the video. Thus, these cells were manually selected.

1281 PCA and Sparse PCA – Principal Component analysis (PCA) of neuron activity time-series data
1282 was performed using in-built functions in MATLAB. Sparse Principal component analysis (SPCA)
1283 was performed using freely available MATLAB toolbox [69].

1284 Neuron activities correlation to animal motion – To ascertain that the motion of the worm in
1285 device has signatures of wave-propagation in freely moving animals, we looked for phase shift
1286 in the velocity of the different regions of the animal in the device (similar to phase shift in
1287 curvature of body parts of animals seen in freely moving animals[70]). To calculate the velocity,
1288 displacement of randomly selected cells along the anterior-posterior axis of the animal was
1289 calculated (Figure 5 – video supplement 2) based on the tracking of cells. Cell displacements
1290 were smoothed using Savitzky-Golay filter. Subsequently, velocity of each cell was calculated by
1291 differentiating the displacement of each cell.

1292 Mutual information (MI) of the obtained velocity signal was calculated with 1) stimulus tuned
1293 neurons, 2) neurons with significant weights in sparse principal components 1-3, and 3)
1294 remaining cells. MI analysis requires estimating the joint probability density of velocity and
1295 neuron activity. We used the kernel density estimation method to do so that uses Gaussian
1296 kernel with bandwidth parameters (that specify the variance of gaussian kernel) set to [0.05,
1297 0.05]. Cells grouped in SPC2 always had the largest mutual information with velocity regardless
1298 of the choice of the bandwidth parameter.

1299 Runtime comparison - To compare optimization runtimes of CRF and registration based method
1300 CPD[34], synthetic data was generated using OpenWorm atlas as described previously with
1301 randomly selected 10, 20, 50, and 130 cells. Annotation was performed using only positional
1302 relationship features. Full head ganglion OpenWorm atlas (206 cells) was used for annotation.
1303 Simulations were run on standard desktop computer (Intel® Xeon® CPU E5-1620 v4 @3.5GHz,
1304 32GB RAM).

Statistical Analysis

Standard statistical tests were performed using Paired Comparisons App in OriginPro 2020. Details regarding the tests (sample size, significance, method) are reported in figure legends. Following asterisk symbols are used to denote significance level throughout the manuscript - * ($p < 0.05$), ** ($p < 0.01$), *** ($p < 0.001$). Significance level not indicated in figures implies not significantly different (n.s).

Appendix – extended methods

S1: Detailed description of CRF_ID model for cell annotation

S1.1 Advantages of CRF_ID framework

There are several advantages of CRF based formulation for automatic cell annotation as discussed below.

1. The major advantage of CRF framework is that arbitrary feature relationships f_k among non-independent observations $\mathbf{x}$ can be specified[26]. This enables optimizing user-defined arbitrary features. Depending on the scope of potential functions Φ_c, these features may be cell specific i.e. unary features ($\Phi_c = \Phi_i$), pairwise features ($\Phi_c = \Phi_{ij}$) or higher order features. E.g. position of cells in ganglion, position related to a landmark cell with known identity, color etc.
2. Since we are modeling a probability distribution over labels assigned to neurons, a set of candidate list of names can be generated for each neuron. A straightforward way to do this is by using the marginal probabilities of labels assigned to each cell. Once the normalization constant Z has been calculated, marginal probability of cell i taking a label y_i can be obtained as $P\left({}^{y_i}/_{\mathbf{x}}\right) = \sum_{j \in V, j \neq i} P(\mathrm{y}/\mathbf{x})$. Other computationally efficient methods to estimate the marginal probabilities using eigenvector of potential functions[71,72] have been proposed as well. We propose an alternative method guided to generate such list (see Appendix – extended methods S1.5) taking into account the undetected neurons in image stack.
3. Log-linear parameterization of feature functions as in (Eq. 2), CRF models belong to exponential family models[55] or max-entropy models. Thus, the joint probability distribution over labels assigned to neurons that we infer is maximally unbiased (maximum entropy) subject to some empirical constraints (sufficient statistics to define the probability distribution). In our case, these empirical constraints are geometrical relationships among cell positions. Interestingly, the maximum entropy nature of the objective function in our model also makes it very similar to the entropy regularized optimal transport problems[30,56].
4. CRF framework is a trainable algorithm[73,74]. Thus, if annotated image stacks are available, the weights λ_k of the feature functions can be optimized directly such that the cell labels in annotated image stacks match the predicted labels. Further, we show that annotated experimental data can be utilized by building a data driven atlas and feature functions can updated based on data-driven atlas (S1.7). For pairwise feature functions in our model, building such data-driven atlas requires cheap mathematical operations (averaging).

S1.2 Features in CRF_ID based annotation framework

During manual annotation of neuron identities in images, researchers use intuitive features such as positions of neurons in image and atlas, positional relationships among neurons, proximity of neurons to one another, neuron with known identities such as neurons expressing

specific markers etc. In this section we describe how such intuitive features are encoded in our model.

S1.2.1 Unary potentials – Positions along AP axis

In empirical data we observed that anterior-posterior (AP) positional relationships among the cells are most stereotypical (Figure 2 – figure supplement 6C, D) and consistent with the 3D atlas. Further since, x-y sampling of image stacks is much higher than z sampling, AP axis' sampling is always higher than LR and DV axes. Thus, positions along AP axes detected by segmentation method are less noisy. Thus, we included a feature based on positions of cell along AP axis as unary feature.

$$f_a(m; i, \boldsymbol{x}) = \exp\left(-\frac{||x_{i,AP} - y_{m,AP}||_2}{\sigma_u^2}\right) \tag{5}$$

Here $x_{i,AP}$ is the position of cell i along AP axes in data and $y_{m,AP}$ is the position of cell with label m in atlas. σ_u is the bandwidth parameter. To account for scale differences between image and atlas, positions along AP axis are normalized first in image and atlas. Low values of σ_u greatly penalizes the deviation of cell position along AP axis in image from atlas position thus favoring that labelling should preserve positions along AP axis. Large values of σ_u decreases the effect of deviation of cell position along AP axis. Thus, this feature restricts the assignment of a particular label to certain cells along AP axes making the labelling AP consistent. Low values of σ_u also increases the influence (weight) of this unary potential compared to the pairwise potentials in labelling. We set this parameter as 1 to set equal influence of unary and pairwise potentials.

S1.2.2 Pairwise potentials – Binary positional relationships

These features encode that the labels assigned to neuron in an image should preserve the relative positional relationships among them in the atlas. E.g. if neuron i is to the left of neuron j in the image stack and these neurons are assigned labels m and n respectively, then the neuron with label m in the atlas should be to the left of neuron with label n. Similar constraints can also be applied on anterior-posterior relationship and dorsal-ventral relationship of neurons in the image stack. Let $[x_{i,AP}, x_{i,LR}, x_{i,DV}]$ denote the coordinate of neuron i in the image along anterior-posterior (AP), left-right (LR) and dorsal-ventral (DV) axes respectively. Similarly, $[y_{m,AP}, y_{m,LR}, y_{m,DV}]$ be the coordinates of neuron with label m in the atlas. The feature functions f_{AP} is defined as

$$f_{AP}(m, n; i, j, \mathbf{x}) = \lambda_{AP} \begin{cases} 1, & (x_{i,AP} - x_{j,AP})(y_{m,AP} - y_{n,AP}) > 0 \\ 0, & (x_{i,AP} - x_{j,AP})(y_{m,AP} - y_{n,AP}) < 0 \end{cases} \tag{6}$$

Thus, this feature implies that if atlas labels m and n assigned to neurons i and j in image are consistent with the AP positional relationship of cells i and j, then the feature value is 1 else 0. Note that the feature values for cells i and j is same (1 or 0) irrespective of the labels m and n assigned to cells. This is true only if annotation is performed using a static atlas or an atlas built from only one data source. We expand more on this in section S1.7 (Building data-driven

atlases) and explain how label dependent feature functions are formed with the availability of empirical hand-annotated datasets.

Similarly, features are defined for left-right and dorsal-ventral relationships, f_{LR} and f_{DV} respectively.

$$f_{LR}(m,n;i,j,\mathbf{x}) = \lambda_{LR}\begin{cases}1, & (x_{i,LR}-x_{j,LR})(y_{m,LR}-y_{n,LR})>0\\ 0, & (x_{i,LR}-x_{j,LR})(y_{m,LR}-y_{n,LR})<0\end{cases} \tag{7}$$

$$f_{DV}(m,n;i,j,\mathbf{x}) = \lambda_{DV}\begin{cases}1, & (x_{i,DV}-x_{j,DV})(y_{m,DV}-y_{n,DV})>0\\ 0, & (x_{i,DV}-x_{j,DV})(y_{m,DV}-y_{n,DV})<0\end{cases} \tag{8}$$

Here λ_{AP}, λ_{LR}, and λ_{DV} are hyperparameters in the model that weighs positional relationship features against other pairwise features in the model. We set these parameters as 1 to give equal weightage to all features.

S1.2.3 Pairwise potentials – Proximity relationships

While manually annotating images by comparing positions of neurons to atlas, researchers often use proximity relationship among neurons i.e. if neuron i is anatomically far from neuron j then the identities to be assigned to these neurons from atlas should not belong to neighboring or nearby neurons. To encode such intuition in the model, we include proximity feature similar to the Gromov-Wasserstein discrepancy used in shape matching[30,31]

$$f_{proximity}(m,n;i,j,\mathbf{x}) = -\lambda_{proximity}||d(x_i,x_j)-d(y_m,y_n)||_2 \tag{9}$$

Here $d(x_i,x_j)$ is any distance measure between neurons i and j in the image stack and $d(y_m,y_n)$ is the same distance measure between neurons with labels m and n in the atlas. We use geodesic distances between cells. To calculate geodesic distances, graphs were constructed by connecting each neuron to its nearest 6 neighbors. $\lambda_{proximity}$ is hyperparameter that weight proximity relationship feature function against other features in the model. We set $\lambda_{proximity}$ as 1.

We compared geodesic distances rather than Euclidean $\mathcal{L}_2$ distances because geodesic distances are invariant to spatial scale in the image and atlas. This is critical since the scale of spatial distribution of neurons in the atlas is very different (much lower) than those in the images. Also, the spatial scale of distribution of cells may vary across images, depending on the size of worm in images, however geodesic distances among neurons should be preserved.

S1.2.4 Pairwise potentials – Angular positional relationships

Relative positional relationship features described in S1.2.2 encode information about positional relationships along axes independently i.e. each feature contains information about positional relationship along one axis only. E.g. f_{AP} encodes whether neuron i is anterior to or posterior to neuron j and how the labels should be assigned to these cells. A feature that simultaneously accounts for positional relationships along all axes may additionally help in determining identities of neurons. Such a feature could be formed by multiplying AP, LR and DV positional relationship features. However, a multiplied feature will still contain binary

information only about whether neuron i is anterior, dorsal and to the right of neuron j or not. It would not tell anything about fine scale directional relationships. Thus, we formulated an angular relationship feature. Let $p_i{}'$ and $p_j{}'$ be the 3D vectors associated with coordinates of neurons i and j in the image stack. Also let $p_m{}''$ and $p_n{}''$ be the 3D vectors associated with coordinates of neurons with labels m and n in the atlas. Then the feature is defined as

$$f_{angle}(m,n;i,j,\mathbf{x}) = \lambda_{angle}\left(1+\left(\frac{p_i'-p_j{}'}{||p_i'-p_j'||_2}\cdot\frac{p_m''-p_n{}''}{||p_m''-p_n''||_2}\right)\right)\Big/_2 \tag{10}$$

Thus, if the vector $\overrightarrow{p'q'}$ aligns perfectly with the vector $\overrightarrow{p''q''}$, $f_{angle}=1$ and $f_{angle}=0$ if the vectors point in completely opposite directions. This feature encodes directional agreement of the labels m and n assigned to neurons i and j. Here λ_{angle} is hyperparameter that weight angular relationship feature function against other features in the model. We set λ_{angle} as 1.

S1.3 Defining AP, LR and DV axes

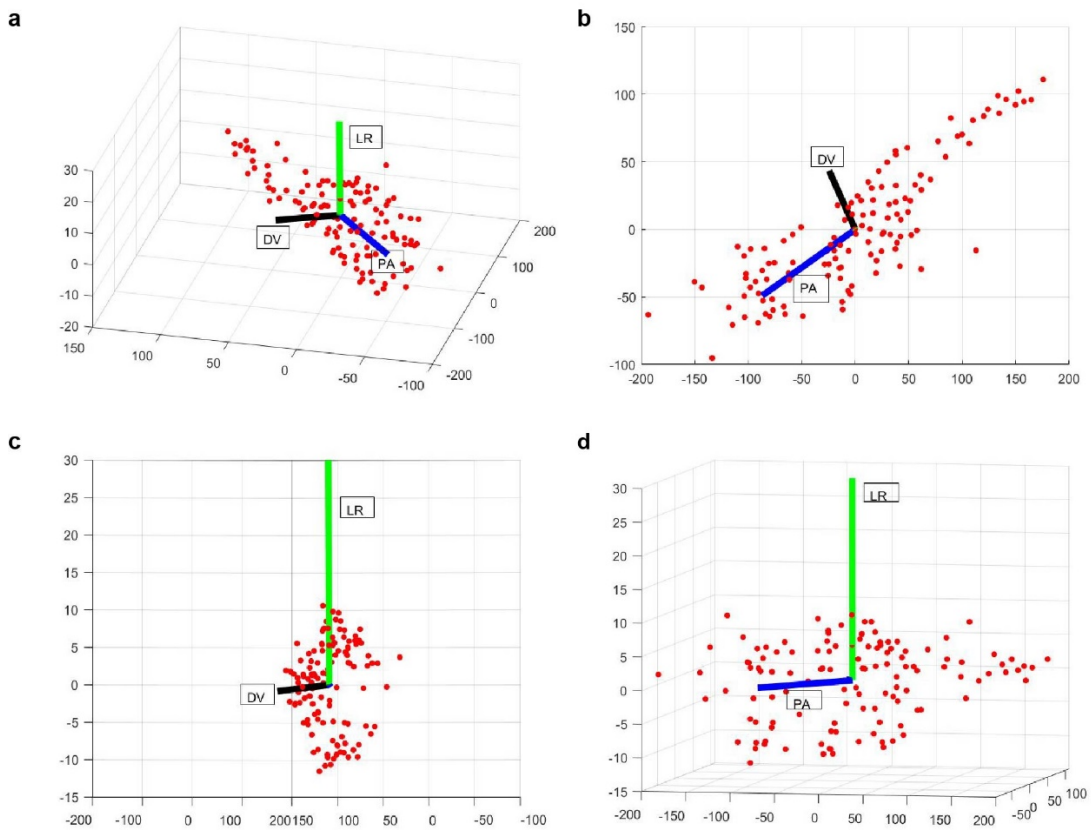
To compare positional relationships among neurons in image and atlas, it is necessary to define anterior-posterior (AP), left-right (LR) and dorsal-ventral (DV) axes in image as well in atlas. 3D coordinates of neurons along these axes are then used to define features described above. We use two methods to define these axes. In method 1, we use Principal Component Analysis to obtain these axes. Let $\mathbf{p}=[p_1,\dots p_N]\in\mathbb{R}^{3\times N}$ be the centered coordinates (zero mean) of N neurons detected in the image stack or atlas. Then the principal components correspond to the eigenvectors of the matrix $\mathbf{pp}^T$. Since the spatial spread of neurons in image as well as in atlas is maximum along AP axes, the first principal component (eigenvector corresponding to maximum eigenvalue) always corresponds to AP axis. Second and third PCs can be assigned to LR and DV axes depending on the orientation of worm in image as described below. Due to rotation of worm about AP axis, the second and third eigenvectors may not always correspond to LR and DV axes. Thus, we designed two methods for these different scenarios –

1. **Worm lying on LR axis** – In this case LR axis is assigned to the third eigenvector. This is because z-sampling of image stacks is much smaller compared to x-y sampling. Thus, the spread of neurons along LR axis is smallest. Figure below shows axes obtained using PCA.
2. **Worm rotated about AP axis** – In this case we developed an alternative method. First, we define LR axis. To do so we use left-right pair of neurons that are easily detectable in image stack. We used RMEL-RMER for landmark strain (Figure 4D) and RMDVL-RMDVR for NeuroPAL strain (Figure 6E, F). Using these neuron-pair coordinates, an LR vector, $\overrightarrow{lr}$ was defined as $p_r-p_l\big/||p_r-p_l||_2$. Next AP vector, $\overrightarrow{ap}$ was determined by solving constrained optimization problem

$$\begin{aligned}&\overrightarrow{ap}=\max(\overrightarrow{ap}.\vec{v}_1)\\ &\text{s.t.}\ \ \overrightarrow{ap}.\overrightarrow{lr}=0\ and\ \left|\left|\overrightarrow{ap}\right|\right|_2=1\end{aligned} \tag{11}$$

i.e. a unit vector which is orthogonal to LR vector and in the direction of first principal component $\overrightarrow{v_1}$. Next, $\overrightarrow{dv}$ vector is obtained by defining a vector orthogonal to both $\overrightarrow{ap}$ and $\overrightarrow{lr}$ vectors.

Finally, we check the consistency of $\overrightarrow{ap}$, $\overrightarrow{lr}$, $\overrightarrow{dv}$ vectors i.e. these vectors should point to the anterior, right and ventral of the worm and should satisfy cross product rule to constitute a valid coordinate system. This is necessary because PCA axes are determined up to a multiplication factor of -1 i.e. coordinate system specified by the principal components (PC1, PC2, PC3) is same as the coordinate system specified by (-PC1, -PC2, -PC3). Thus, a user input is taken in the framework while defining axes. Users can easily click on any neuron in the anterior portion and the posterior portion of the worm image, when asked to do so, to specify PC1 direction.



SN1 Figure 1. Examples of PA (blue), LR (green) and DV (black) axes generated automatically in a whole-brain image stack. Here red dots correspond to the segmented nuclei in image stack. Shown are 3D view (a), XY (b), YZ (c) and XZ (d) views of the image stack

S1.4 Inferring neuron identities

To infer most probable identity of neurons, energy function in (Eq. 4) is to be maximized. Exact inference techniques for maximizing energy functions over arbitrary graph structures, such as the fully connected graph structure of our model, are not available[27]. Thus, we use a commonly used approximate inference method that has been used successfully in past for several applications, Loopy Belief Propagation (LBP)[28,75] to infer optimal labelling i.e. the maximum of joint-probability distribution $p({}^{\mathbf{y}}/_{\mathbf{x}})$ as well as the marginal probabilities of labels assigned to

each node. We implement our model in MATLAB using an open source package for undirected graphical models[76].

S1.5 Resolving duplicate assignments with label consistency score

In general, we define pairwise feature functions in the model such that it penalizes duplicate assignments of any label m to cells i and j in image as follows

$$f_{angle}(m, m; i, j, \mathbf{x}) = -\infty \tag{12}$$

Even with this measure, we still see some duplicate assignments when the LBP optimization converges. To resolve such assignments, we mark all cells that were not assigned duplicate labels as assigned. Next, we calculate a label consistency score for each cell that was assigned a duplicate label (examples shown in Figure 1 – figure supplement 1C). This score measures how consistent the current label assigned to the cell is in terms of preserving its positional relationships with other cells that were not assigned any duplicate labels (i.e. cells marked as assigned). Among all the cells that were assigned same duplicate label, the cell with the highest consistency score is assigned the duplicate label and marked as assigned. Remaining cells are marked as unassigned. After resolving all such duplicate assignments for all labels, optimization is run again only for the unassigned cells while keeping the identities of other cells fixed. We calculate the label consistency score for each cell as follows

$$bin.pos.rel.(i) = \sum_{j \in V'} f_{AP}(m, n; i, j, \mathbf{x}) + f_{LR}(m, n; i, j, \mathbf{x}) + f_{DV}(m, n; i, j, \mathbf{x}) \tag{13}$$

$$ang.rel.(i) = \sum_{j \in V'} f_{angle}(m, n; i, j, \mathbf{x}) \tag{14}$$

$$proximity\ rel.(i) = \sum_{j \in V'} f_{proximity}(m, n; i, j, \mathbf{x}) \tag{15}$$

$$consistency\ score = bin.pos.rel. + ang.rel. + proximity\ rel. \tag{16}$$

Here i is a cell that was assigned a duplicate label and V' denotes the set of all other cells that were not assigned duplicate labels. Intuitively, correctly predicted cells should have higher consistency score. This is because correctly predicted cells will preserve their relationship with other correctly predicted cells and won't preserve their relationship with incorrectly predicted cells. In contrast, incorrectly predicted cells will neither preserve their relationship with correctly predicted cells nor with incorrectly predicted cells thus having a lower consistency score. This was observed in simulations as well (Figure 1 – figure supplement 1D). Thus, label consistency score also serves as a good criterion for sorting candidate list of labels predicted by the framework.

Label consistency score is combination of features in the model that define the probability distribution $p(\mathrm{y}/\mathbf{x})$ in our model. Thus, another way to look at the label consistency score is that it tries to maximize the pseudo-loglikelihood[77,78] of labels to be assigned to all cells with duplicate labels conditioned on the labels of other cells that were not assigned duplicate labels i.e.

$\sum_{i \in dup} \log\big(p(y_i; y_{V'}, \mathbf{x})\big)$. Here dup denotes the set of all cells that were assigned duplicate labels.

S1.6 Simulating missing cells and generating candidate name list for each cell

The 3D atlas of *C. elegans* neurons we used is freely available[12]. There are 195 cells in head ganglion in this atlas, i.e. the label list from which identities are to be assigned to cells in data has 195 elements. However, empirically we detect only ~120-130 neurons in whole-brain image stacks. Remaining neurons are undetected due to either no/low expression levels of fluorophore in these cells or false-negatives in automated detection algorithm. Approximately similar number of neurons were detected by other labs as well[5]. Further, which cells are undetected is not known *a priori*. Thus, to take into account missing cells while annotating identities with the model, we define a hidden variable $\mathbf{h} \in \{0,1\}^N$ that specifies the cells missing in images i.e. $\mathbf{h_k} = 1$ if cell with label k in atlas is missing in the image. Here N is the number of labels/cells in atlas. With the hidden variable, we model the joint probability distribution $\mathrm{P}\left({}^{\mathrm{y},\mathbf{h}}/_{\mathbf{x}}\right)$ as

$$P\left({}^{\mathrm{y},\mathbf{h}}/_{\mathbf{x}}\right) = \frac{1}{Z}\prod_{i \in V} \Phi_i(m; i, \mathbf{x}) \prod_{e_{ij} \in \mathcal{E}} \Phi_{ij}(m, n; i, j, \mathbf{x}) \prod_{i \in V} \Phi_i(m, \mathbf{h})\, \Phi(\mathbf{h}, \mathbf{x}) \tag{17}$$

Here $Z = \sum_{\mathrm{y} \in \mathcal{Y}, \mathbf{h}} \prod_{i \in V} \Phi_i(y_i; i, \mathbf{x}) \prod_{e_{ij} \in \mathcal{E}} \Phi_{ij}\big(y_i, y_j; i, j, \mathbf{x}\big) \prod_{i \in V} \Phi_i(y_i, \mathbf{h}) \Phi(\mathbf{h}, \mathbf{x})$. Unary potential functions Φ_i and pairwise potential functions Φ_{ij} are same as described in S1 and S1.2. Potential function $\Phi_i(m, \mathbf{h})$ captures the dependencies between label m assigned to cell i when the hidden variable is $\mathbf{h}$. We set the potential function as

$$\Phi_i(m, \mathbf{h}) = \begin{cases} 0, & if\ \mathbf{h}_m = 1 \\ 1, & if\ \mathbf{h}_m = 0 \end{cases} \tag{18}$$

Thus, the potential specifies that if m^{th} label in atlas is missing in the image then that label cannot be assigned to any cell in the image. Further, if m^{th} label is not missing in the image then each cell has equal potential to be assigned that label which will be updated based on positional relationship potentials. Further, the potential function $\Phi_i(\mathbf{h}, \mathbf{x})$ captures the dependence between the observed image $\mathbf{x}$ and the hidden variable $\mathbf{h}$. Defining this potential function is not trivial for whole-brain images since predicting which cells are missing just based on observed 3D whole-brain image stack is difficult. However, in specific cases confidence values of each label missing in the image may be available, denoted by $\beta_k, 0 \leq \beta_k \leq 1$. Such confidence values may be specified based on prior knowledge e.g. based on expression pattern of fluorophore labeling cells in strain, cells that have low detection probability may have higher parameter β_k. Using these confidence values, we define $\Phi(\mathbf{h}, \mathbf{x})$ as

$$\Phi(\mathbf{h}, \mathbf{x}) = \prod_{k=1}^{N} \beta_k^{h_k} (1 - \beta_k)^{(1-h_k)} \tag{19}$$

Our goal is to calculate $P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right)$ which can be obtained by marginalizing (Eq. 17) over the hidden states $\mathbf{h}$. However, since the number of elements in the space of $\mathbf{h}$ is huge, marginalizing as well

as calculating the normalization constant Z is not tractable. But, for specific cases such as we describe below, the calculation can be simplified.

In the absence of any prior information about which labels are missing in images for whole-brain imaging case, we assigned equal confidence value for each label missing in the image i.e. $\beta_k = \beta$. Thus, for a fixed number P of missing cells in image, $\Phi(\mathbf{h}, \mathbf{x}) = \beta^P (1-\beta)^{N-P}$. A consequence of defining potential function this way is that for a given number of missing cells in image, each combination of missing labels in atlas is equally probable as long as labels of missing cells are not assigned to any cell in image

$$P\left({}^{\mathbf{h}}/_{\mathrm{y}} = y, \mathbf{x}\right) = \frac{P\left({}^{y,\mathbf{h}}/_{\mathbf{x}}\right)}{\sum_{\mathbf{h}} P\left({}^{y,\mathbf{h}}/_{\mathbf{x}}\right)} \tag{20}$$

$$= \frac{\prod_{i\in V} \Phi_i(m; i, \mathbf{x}) \prod_{e_{ij}\in\mathcal{E}} \Phi_{ij}(m, n; i, j, \mathbf{x}) \prod_{i\in V} \Phi_i(m, \mathbf{h})\, \Phi(\mathbf{h}, \mathbf{x})}{\sum_{\mathbf{h}} \prod_{i\in V} \Phi_i(m; i, \mathbf{x}) \prod_{e_{ij}\in\mathcal{E}} \Phi_{ij}(m, n; i, j, \mathbf{x}) \prod_{i\in V} \Phi_i(m, \mathbf{h})\, \Phi(\mathbf{h}, \mathbf{x})} \tag{21}$$

$$= \frac{\prod_{i\in V} \Phi_i(m, \mathbf{h})\, \Phi(\mathbf{h}, \mathbf{x})}{\sum_{\mathbf{h}} \prod_{i\in V} \Phi_i(m, \mathbf{h})\, \Phi(\mathbf{h}, \mathbf{x})} = \frac{1}{|\mathbf{h}'|} \tag{22}$$

Here $\mathbf{h}' \subset \{0,1\}^{\boldsymbol{N}}$ such that $\mathbf{h}_k = 0\ \forall\ k^{th}\ label\ assigned\ to\ cells\ in\ y\ and\ \sum_k \mathbf{h}_k = P$. Further, $|\mathbf{h}'|$ is the number elements in $\mathbf{h}'$. Thus, we can randomly and uniformly sample $\mathbf{h}$ and keep it fixed while predicting $P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right)$. Our goal is to calculate $P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right)$. We do so as

$$\arg\max_{\mathrm{y}\in\mathcal{Y}} P\left({}^{\mathrm{y}}/_{\mathbf{x}}\right) = \sum_{\mathbf{h}} \arg\max_{y\in\mathcal{Y}} P\left({}^{\mathrm{y},\mathbf{h}}/_{\mathbf{x}}\right) \tag{23}$$

Therefore, we randomly select P cells in atlas that are considered to be missing in the image data. These P cells are selected uniformly across different regions of the head ganglion following the discussion above. Labels of these P cells are removed from the list of possible labels that can assigned to neurons and identities of cells are predicted. This process is repeated ~1000 times to sample multiple possible combinations of $\mathbf{h}$. Finally, a candidate list of names is generated using (Eq. 23) i.e. compiling a list of optimum labels predicted for each cell in each run (by maximizing $P\left({}^{\mathrm{y},\mathbf{h}}/_{\mathbf{x}}\right)$ in each run) and choosing the top frequent labels for each cell across all runs.

S1.7 Building data-driven consensus atlas

Here we describe the procedure and the intuition behind building data driven consensus atlas in our framework. We also describe the intuition behind why it is computationally more efficient than building data driven atlases for registration methods.

First, we describe how features in the model are defined using a static atlas such as OpenWorm atlas and then extend the formulation to building and using data-driven atlas. Positional relationships among cells based on the OpenWorm atlas are stored as matrices of size

$N \times N$ where N is the number of cells in atlas. Each cell in matrix records the positional relationship between a pair of cells observed in the atlas. For example, for AP positional relationship matrix, if a column cell such as RMEL is anterior to a row cell such as AIZR in atlas, then the corresponding cell in matrix will denote 1 and otherwise 0. Here 1 implies that according to the prior information available from OpenWorm, cell RMEL is observed to be anterior to cell AIZR with 100% probability. Let $F_{AP} \in \mathbb{R}^{N \times N}$ be the AP positional relationship matrix and m, n be labels in atlas. Then,

$$F_{AP}(m,n) = \begin{cases} 1, & n > m \; along \; AP \\ 0, & n < m \; along \; AP \end{cases} \tag{24}$$

Note that $F_{AP}(n,m) = 1 - F_{AP}(m,n)$. Using the matrix F_{AP}, the AP positional relationship feature function $f_{AP}(m,n;i,j,\mathbf{x})$ for cells i and j (as described in S1.2.2) can be defined as

$$f_{AP}(m,n;i,j,\mathbf{x}) = \begin{cases} F_{AP}(m,n), & j > i \; along \; AP \\ F_{AP}(n,m), & j < i \; along \; AP \end{cases} \tag{25}$$

Here $f_{AP}(m,n;i,j,\mathbf{x})$ denotes the AP positional relationship feature for assigning labels m and n in atlas to cells i and j in image. Note that Eq. 25 is consistent with Eq. 6 as shown below by expanding $F_{AP}(m,n)$ and $F_{AP}(n,m)$ terms using Eq. 24 in Eq. 25.

$$f_{AP}(m,n;i,j,\mathbf{x}) = \begin{cases} 1, & n > m \;\cap\; j > i \; along \; AP \\ 0, & n < m \;\cap\; j > i \; along \; AP \\ 1, & m > n \;\cap\; j < i \; along \; AP \\ 0, & m < n \;\cap\; j < i \; along \; AP \end{cases} \tag{26}$$

Thus, the AP positional relationship matrix F_{AP} stores the prior knowledge available on anterior-posterior relationships among cells. Since, F_{AP} is built using only a single data source i.e. OpenWorm atlas, all elements in the matrix are either 1 or 0. This implies that for all pairs of cells m and n in atlas, the F_{AP} matrix says that cell m is anterior to cell n with either 100% probability or 0 probability. Also, note that since F_{AP} consists of only 1's and 0's, the AP positional relationship feature, $f_{AP}(m,n;i,j,\mathbf{x})$**,** also consists of only 1's and 0s, thus it is independent of labels m and n assigned to cells i and j.

In contrast to using a single data source, if additional information is available in the form of annotated experimental datasets, the prior knowledge on anterior-posterior relationships can be updated. E.g. it may be possible that cell m is observed to be anterior to cell n with 80% probability in annotated datasets (e.g. in 8 out of 10 experimental datasets) and posterior to cell n with 20% probability. This empirically observed AP positional relationships among cells can be updated in matrix F_{AP} as

$$F_{AP,data-driven}(m,n) = \begin{cases} w_{mn}, & n > m \; along \; AP \\ 1 - w_{mn}, & n < m \; along \; AP \end{cases} \tag{27}$$

Here $w_{mn}, 0 \leq w_{mn} \leq 1$ is the fraction of annotated datasets in which labels m is annotated to the anterior of label n. Thus, instead of specifying a hard constraint based on a single data source static atlas (as in Eq. 24), w_{mn} specifies a soft constraint based on positional relationships observed in experimental annotated datasets. Formally, $w_{mn}$is defined as

$$w_{mn} = \frac{1}{D}\sum_{d=1}^{D} I(m, n; \mathbf{x}, d) \tag{28}$$

$$I(m, n; \mathbf{x}, d) = \begin{cases} 1, & x_n^d > x_m^d \\ 0, & x_n^d < x_m^d \end{cases} \tag{29}$$

Here, $d \in \{d_i\}_1^D$ denotes an annotated data set, D is the total number of annotated datasets used to build atlas, $x_m^d \in \mathbf{x}$ and $x_n^d \in \mathbf{x}$ are the coordinates along AP axis of cells annotated labels m and n in dataset d. Eq. 26 can be seen as a generalization of Eq. 24. Further, the AP positional relationship feature function $f_{AP}(m, n; i, j, \mathbf{x})$ can be updated based on data-driven atlas using Eq. 25. The above discussion can be similarly extended to LR, DV relationships, angular relationship and proximity relationship.

Similarly, for angular relationship feature, instead of using a fixed vector $\overrightarrow{p''q''}$ (refer to Eq. 10) between cells m and n that is provided in static atlas, we use an average vector obtained from annotated experimental datasets.

$$\overrightarrow{p''q''}_{data-driven} = \frac{1}{D}\sum_{d=1}^{D} \frac{p''_{m,d} - p_{n,d}{}''}{||p''_{m,d} - p''_{j,d}||_2} \tag{30}$$

Here $p''_{m,d}$ and $p''_{n,d}$ are position coordinates of cells with labels m and n in data d. Similarly, for proximity relationship feature (refer to Eq. 9), instead of using a fixed distance $d(y_m, y_n)$ between cells with labels m and n in atlas, we use an average distance obtained from annotated experimental datasets.

$$d(y_m, y_n)_{data-driven} = \frac{1}{D}\sum_{d=1}^{D} d(y_{m,d}, y_{n,d}) \tag{31}$$

Here $d(y_{m,d}, y_{n,d})$ is the distance between cells with labels m and n in data d.

The key difference between registration-based methods and our framework in building data-driven atlases lies in the underlying methodology used by these methods to annotate cells. Registration methods annotate cells by maximizing extrinsic similarity between images and atlas. This requires the pre-alignment of spaces in which the image and the atlas exist. Thus building-data driven atlas for registration-based methods also requires pre-alignment of all annotated images to the same space. This is typically done by simultaneously registering or block-wise registering all annotated images which requires solving several constrained regression problems.

In contrast, our framework annotates cells by maximizing intrinsic similarity between images and atlas which is independent to the pre-alignment of images to same space. Thus, in our framework positional relationship features can be calculated for each annotated dataset in its own space and subsequently the features are aggregated together by simple averaging operations, which is computationally efficient, to build data-driven atlases.

S1.8 Discussion on previous methods using Registration to annotate cells

Qu et al.[15] and Long et al.[1] annotate cells in *C. elegans* images by spatially deforming an atlas with known cell identities and registering it to the image data. Similar method was proposed by Aerni et al.[16]. Here, several other features were used along with location feature such as cell shape, cell size, fluorophore expression level etc. In Long et al.[14], *C. elegans* images were annotated by comparing each cell's location in image to cell locations in multiple template images to generate initial matches. Generated matches were then pruned by checking relative position consistency such as anterior-posterior relationship among cells. Thus, although pairwise positions were used, they were not systematically optimized to predict cell labels and were only used for enforcing consistency in post-registration matching. In Scholz et al.[18], cell identities were determined by registering image data to the 2D atlas[11]. Registering 3D data to 2D atlas makes it difficult to disambiguate the identities along LR axis since the LR neurons are not exactly symmetrical along that axis. Similarly, registration-based cell annotation was proposed in Toyoshima et al.[17]; however, in this case the authors generated their own atlases by registering several partial atlases with subset of cells labeled in each atlas. Registration-based cell identification was proposed in Yemini et al.[10] as well. Here additional color information was integrated as a feature for registration along with spatial location of cells.

S1.9 Registration methods do not consider intrinsic similarity features such as relative positional relationships

One of the major reasons of higher cell annotation accuracy achieved by our framework (Figure 2C,D,E, Figure 3B, Figure 4B, Figure 6B, Figure 2 – figure supplement 5) is that our model systematically includes and optimizes pairwise positional relationships between cells to determine cell labels thus. In comparison, registration-based methods that predict cell identities by registering image stack to the atlas maximize the extrinsic similarity between images and atlas[19]. Due to inherent biological variability, spatial distribution of cells in worms differ significantly (Figure 2 – figure supplement 6A, B) from the positions of cells in atlas while the relative positional relationships are still preserved (Figure 2 – figure supplement 6C, D). Below we provide a mathematical argument for why registration-based methods do not include pairwise positional relationships. Further we show that how registration information can be included in our model.

Following the description in S2.1, the objective function to be optimized in registration methods[79–83] is similar to

$$E \propto \sum_{ik} -c_{ik} ||x_i - \boldsymbol{\mathcal{T}}(y_k)||^2_{\Sigma_k^{-1}} + \mathcal{R}(\boldsymbol{\mathcal{T}}) \tag{32}$$

Here $x_i \in \mathbb{R}^3$ is the coordinate of ith cell in the image, $\boldsymbol{\mathcal{T}}(y_k) \in \mathbb{R}^3$ is the registered coordinate of kth cell with label k in the atlas, $c_{ik}$is the posterior probability that i^{th} cell in image matches to the kth in cell in atlas, and $\mathcal{R}(\boldsymbol{\mathcal{T}})$ is the regularization term on the transformation applied to cells to ensure smoothness of transformations for non-rigid registration. If only the first term in the energy function (Eq. 4) is kept in our model, then energy function can be written as

$$\mathrm{E}(\mathrm{y}, \mathbf{x}) = \left(\sum_{i=1}^{V} \sum_{a} \lambda_a f_a(k; i, \boldsymbol{x}) \right) \tag{33}$$

This model considers only unary (cell specific) features for predicting cell names. Further, if we consider only one unary feature function in the model based on spatial locations of cells in atlas defined as below

$$f_a(k; i, \boldsymbol{x}) = ||x_i - \boldsymbol{\mathcal{T}}(y_k)||^2_{\Sigma_k^{-1}} \tag{34}$$

and substituted back in (Eq. 32), then the energy function is equivalent to the objective function in registration algorithms.

$$\mathrm{E}(\mathrm{y}, \mathbf{x}) = \left(\sum_{i=1}^{V} \sum_{a} \lambda_a ||x_i - \boldsymbol{\mathcal{T}}(y_k)||^2_{\Sigma_k^{-1}} \right) \tag{35}$$

Thus, objective function in registration algorithms can be specified using only unary features in CRF_ID framework. This shows that registration methods do not account for relative positional relationship features. Further, it highlights that CRF_ID framework can easily integrate registration information, as unary feature term, with relative positional relationship features for predicting cell identities.

S2: Description of different methods that were compared

Below we provide a brief description of the different methods that we compared. Each model uses a different and specific combination of information for predicting cell labels thus helps in dissecting what information is most useful in predicting cell biological names. Some of these methods consist of methods proposed previously for cell annotation in *C. elegans.* Note, for fair comparison, static 3D OpenWorm[12] atlas or 2D atlas available on wormatlas.org was used as reference for all methods: absolute positions of cells were used from atlases for registration methods, positional relationships were derived from the same atlases were used for CRF_ID framework.

S2.1 Registration

In this case, detected neurons in image stacks or neurons in synthetic data were registered to the atlas using a widely used non-rigid registration algorithm[79]. For synthetic data, the process was repeated ~200 times with new synthetic data generated each time. Here, we provide a brief description of the registration algorithm method. Two point-sets are registered by iteratively applying smooth deformation to one of the point-sets. Let two point-sets be $\mathbf{X} =$

$[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m] \in \mathbb{R}^{3\times m}$ and $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n] \in \mathbb{R}^{3\times n}$ consisting of m and n points respectively. Here $\mathbf{x}_i$ and $\mathbf{y}_i$ are coordinates of point-sets. In our case, 3D coordinates of nuclei detected in the image form point set $\mathbf{X}$ and cell positions in the atlas form the point-set $\mathbf{Y}$. In CPD, each point in point-set $\mathbf{X}$ is considered to be a random sample drawn from a mixture of gaussian distributions. Further the point set $\mathbf{Y}$ specifies the centroids of gaussian components of this distribution. Thus, the probability of observing a point $\mathbf{x}_i$ is given by

$$P(\mathbf{x}_i) = {}^{\omega}/_{m} + {}^{(1-\omega)}/_{n} \sum_{k=1}^{n} \mathcal{N}(\mathbf{x}_i; \mathbf{y}_k, \mathbf{\Sigma}_k) \tag{36}$$

Here ω is outlier ratio. A smooth transformation $\mathcal{T}(\mathbf{y}_i, \mathbf{W}): \mathbb{R}^3 \rightarrow \mathbb{R}^3$ is applied to points in point-set $\mathbf{Y}$ given by $\mathcal{T}(\mathbf{y}_i, \mathbf{W}) = \mathbf{y}_i + \sum_{k=1}^{n} \mathbf{G}\,(i,k)\mathbf{W}_k$ so as to maximize the likelihood of point-set $\mathbf{X}$. Here $\mathbf{G} \in \mathbb{R}^{n\times n}$ is gaussian kernel matrix defined as $\mathbf{G}(i,k) = exp\left(-{}^{(\mathbf{y}_i - \mathbf{y}_k)^2}/_{\beta^2}\right)$ and $\mathbf{W}_k$ is the kth column of parameters matrix $\mathbf{W} \in \mathbb{R}^{3\times n}$. Thus, the transformation $\mathcal{T}(\mathbf{y}_i, \mathbf{W})$ lies in a Reproducible Kernel Hilbert Space (RKHS) with gaussian kernel. The aim of parameterizing transformation in this way is to ensure the smoothness of transformation.

Parameters of transformation matrix $\mathbf{W}$ are estimated by maximizing the joint likelihood of data $\mathbf{X}$ and n latent variables corresponding to mixture components. This is done using Expectation-Maximization algorithm. E-step is equivalent to calculating the posterior probability that point $\mathbf{x}_i$ is generated from component k, keeping the current parameters $\mathbf{W}, \mathbf{\Sigma}_k$ fixed.

$$P\left({}^{k}/_{\mathbf{x}_i}\right) = \alpha_{ik} = \frac{\frac{1-\omega}{n} exp\left(-1/2||\mathbf{x}_i - \mathcal{T}(\mathbf{y}_k, \mathbf{W})||^2_{\mathbf{\Sigma}_k^{-1}}\right)}{\frac{\omega}{m} + \frac{(1-\omega)}{n}\sum_{k=1}^{n} exp\left(-\frac{1}{2}||\mathbf{x}_i - \mathcal{T}(\mathbf{y}_k, \mathbf{W})||^2_{\mathbf{\Sigma}_k^{-1}}\right)} \tag{37}$$

Here, $||\mathbf{x}_i - \mathbf{y}_k||^2_{\mathbf{\Sigma}_k^{-1}} = (\mathbf{x}_i - \mathbf{y}_k)^{\mathrm{T}}\mathbf{\Sigma}_k^{-1}(\mathbf{x}_i - \mathbf{y}_k)$. In M-step, component parameters are determined by maximizing the expected value of complete data log-likelihood $\mathcal{L}$ (objective function). Additional regularization term is added to the objective function to minimize norm of $\mathcal{T}$ in RKHS for controlling the complexity (smoothness) of transformation.

$$\mathcal{L} = -\sum_{i,k}^{m,n} \alpha_{ik} \parallel \mathbf{x}_i - \mathcal{T}(\mathbf{y}_k, \mathbf{W}) \parallel^2_{\mathbf{\Sigma}_k^{-1}} - \frac{\lambda}{2} tr(\mathbf{W}\mathbf{G}\mathbf{W}^{\mathrm{T}}) \tag{38}$$

Prior to registration, cells in image stack and in atlas were aligned by defining the position coordinates of cells in AP, LR and DV coordinate system (as described in S1.3) to improve the registration accuracy. Cell identities were predicted based on the correspondences generated by the registration algorithm. Note that this case does not consider co-dependent features among cells (as discussed in S1.10) as registration algorithms utilize only absolution positions of cells.

Additionally, we modified the registration method described above to account for missing cells in images, in a manner similar to the CRF_ID framework and for fair comparison with the CRF_ID annotation method. Identities were predicted iteratively with uniformly and randomly selected cells across head-ganglion considered missing thus the labels of those cells were removed from the atlas list. Cells in images were registered to the remaining cells in atlas. This process was repeated ~1000 times to sample multiple combinations of missing cells and identities were predicted by pooling results across each iteration. We did see an improvement in prediction accuracy with this modification compared to the base that does not account for missing cells. Hence all comparisons were performed with this modification and all results shown are for the modified method.

For registration based matching we used a popular registration method [34]. The parameter settings of the method are below.

Parameter	Value	Description
opt.method	'nonrigid'	Non-rigid or rigid registration
opt.beta	1	The width of gaussian kernel (smoothness)
opt.lambda	3	Regularization weight
opt.viz	0	Don't show any iteration
opt.outliers	0.3	Noise weight
opt.fgt	0	Do not use FGT (fast gaussian transform)
opt.normalize	1	Normalize to unit variance and zero mean before registering
opt.corresp	1	Compute correspondence vector at the end of registration
opt.max_it	100	Max number of iterations
opt.tol	1e-10	tolerance

S2.2 Relative Position (Rel. Pos)

In this case, full CRF based annotation framework as described in Appendix – extended methods S1 was used. We considered only co-dependent features (i.e. pairwise relative positional features as described in S1.2) between all cells. Optimal labels of cells were predicted using the optimization procedure described in S1.4. Note that the information used by model in this case is different than the information used in Registration method as no information about absolute positions of cells is used. Thus, comparing prediction accuracy across these cases helps in verifying that co-dependent features are more useful in predicting neuron identities.

S2.3 Registration + Rel Position

In this case we used both cell specific features (i.e. unary features) and co-dependent features (i.e. pairwise features) to predict cell labels. Pairwise feature terms were the same as described in S1.2. Unary feature term was modified in this case as

$$f_{reg}(k; i, \boldsymbol{x}) = \exp\left(-\frac{||\mathbf{x}_i - \boldsymbol{\mathcal{T}}(\mathbf{y}_k)||_2}{\sigma_{reg}^2}\right) \tag{39}$$

Here $x_i \in \mathbb{R}^3$ and $y_k \in \mathbb{R}^3$ are the coordinates of cell i in image stack and cell k in the atlas respectively (in AP-LR-DV coordinate system). Thus, cell i in image has a higher potential to take label l_k of cell k in atlas if the distance between registered cell i in image and cell k in atlas is small. Optimal transformation $\mathcal{T}(\mathbf{y}_k)$ was inferred using the non-rigid registration method as described in S2.1. Here parameter σ_{reg} controls the weight between registration term and CRF based matching term. E.g. if σ_{reg} is small then the registered cell i in image strongly prefers matching to the nearest cell in atlas. Thus, the relative position features have little influence on altering the cell label based on relative position consistency criteria. In contrast if σ_{reg} is big then the registered cell i has uniform preference of matching to any cell atlas. This allows more flexibility to CRF_ID method to pick optimal labels enforcing the consistency criteria. We set $\sigma_{reg} = 1$ to give equal weightage to the registration term and other features. Optimal labels of cells were predicted using the optimization procedure described in S1.4.

S2.4 Color

In this case, only color information was used to identify the names of cells without using position information of cells at all. This helps in gauging the prediction accuracy that can be attained by using color information. Further, comparing this case with the cases that combine color information with position information (described below) helps in gauging the contribution of cell position information on predicting cell names. We describe below, both the baseline (naïve) method for building color atlas and also building ensemble of color atlases with color distributions in training images aligned to test image.

Baseline color atlas

In this case, color atlas was built by aggregating raw RGB values of cells across images used to build atlas leaving the test image out. Feature function in this case was defined as the kernelized Mahalanobis distance between the colors of cells in image stack and colors of cells in atlas. Note that this is a unary feature (as it is cell specific)

$$f_{col}(k; i, \boldsymbol{x}) = \exp(-(\mathbf{C}_i - \mathbf{C}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{C}_i - \mathbf{C}_k)) \tag{40}$$

Here $\mathbf{C}_i \in \mathbb{R}^3$ is the mean RGB value of the ith cell in test image, $\mathbf{C}_k \in \mathbb{R}^3$ is the mean RGB value of cell with label k in atlas and $\boldsymbol{\Sigma}_k$ is the covariance matrix of the RGB values of cell with label k in atlas. Let $\boldsymbol{r}_k = [r_1, \dots, r_N] \in \mathbb{R}^{3\times N}$ be the observed RGB values of cells with label k in N training images used to build the color atlas, then $\mathbf{C}_k = 1/N(\boldsymbol{r}_k \mathbf{1})$ and $\boldsymbol{\Sigma}_k = (\boldsymbol{r}_k - \mathbf{C}_k \mathbf{1}^T)(\boldsymbol{r}_k - \mathbf{C}_k \mathbf{1}^T)^T$ where $\mathbf{1} \in \mathbb{R}^{N\times 1}$ is a vector of ones. Thus, this feature specifies that cell i in image has a high potential of taking label k in atlas if the Mahalonobis distance between the colors of cell i and cell k is small.

Ensemble atlas with aligned color distributions

We found that when baseline color data-driven atlas was combined with positional relationship features data-driven atlas to annotate cells, the performance increased marginally over the case when only positional relationship features were used for annotation (Figure 6 – figure supplement 1B). This implies that color information did not contribute to annotation task. We

reasoned that this is because distributions of color values vary a lot across images thus color distributions in training data did not reflect the color distribution in test data. This may be due to inherent difference in fluorophore expression across animals, differences in imaging settings across sessions (exposure time, laser power) etc. The problem of different feature distributions in test data compared to training data is solved by domain transfer techniques in machine learning. Here we develop a simple two step domain transfer method.

First, we align color distributions in training images used to build the atlas to the color distribution in test image using several methods –

1. Normalization of each color channel in all images so that pixel values lie between 0 and 1. Let $I_R(x)$ denote raw intensity value of red color channel (mNeptune channel) at location x in image. Then the normalized values are calculated as

$$I_R^{norm}(x) = {(I_R(x) - \min(I_R(x))}\Big/{(\max(I_R(x)) - \min(I_R(x))} \tag{41}$$

   Similar normalization is performed for CyOFP and BFP channels.
2. Histogram matching of each color channel in training image to the corresponding channel in test image. Histogram matching transforms the color distribution in an image so that the color histogram matches the color histogram of reference image. This was implemented using 'imhistmatchn' function in MATLAB.
3. Color invariant transformation of training set images and test image, and subsequent histogram matching of training color invariant images to the test color invariant image. Color invariant transformation transforms the color space of image to remove dependency on lighting geometry and illumination[49]. The transformation is performed as follows. Let $C_i = [r_x, g_x, b_x] \in \mathbb{R}^{p\times 3}$ be the matrix that stores RGB values of all voxels in i^{th} image in the training set. Here p is the number of voxels in the image. Then the color invariant image is obtained by sequentially and repeatedly normalizing the rows and columns of C_i for a fixed number of iterations.

Note that aligning color distribution of training image to test image does not require cell identity information at all and is performed using all RGB pixel values in the image. After aligning the color distributions, the color atlas is built similar to the baseline color atlas i.e. by aggregating the RGB values of cells across images. However, in comparison to the baseline case, now RGB values come from the aligned distributions using one of the methods mentioned above.

Second, an ensemble of data-driven color atlases is used for predicting cell identities i.e. two data-driven color atlases are used with different color alignment techniques used in each atlas. Feature function in this case was defined as

$$f_{col}(k; i, \boldsymbol{x}) = \sum_l \lambda_l \exp\left(-(\mathbf{C}_i - \mathbf{C}_{k,l})^T \boldsymbol{\Sigma}_{k,l}^{-1} (\mathbf{C}_i - \mathbf{C}_{k,l})\right) \tag{42}$$

Here $\mathbf{C}_i \in \mathbb{R}^3$ is the mean RGB value of the ith cell in test image, $\mathbf{C}_{k,l} \in \mathbb{R}^3$ is the mean RGB value of cell with label k in l^{th} atlas, $\mathbf{\Sigma}_{k,l}$ is the covariance matrix of the RGB values of cell with label k in l^{th} atlas, and λ_l is the mixing weight of each atlas, $\sum_l \lambda_l = 1$. Note that each atlas is built with using RGB values from all training images.

In practice we used two atlases in the ensemble atlas. The first atlas used method 2 i.e. histogram matching of raw RGB distribution in training images to the test image. The second atlas used method 3 i.e. color invariant transformation was applied to all images (including test image) and subsequently color histogram of training images was matched to the test image. Mixing weights of 0.2 and 0.8 were selected by cross-validation.

Note that in both cases of color atlas, we do not consider any co-dependent features among cell, thus predicting cell names in this case is equivalent to minimizing the following energy function

$$y = \arg\min_{y \in \mathcal{Y}} \sum_{ik} -f_{col}(k; i, \boldsymbol{x}) \tag{43}$$

This is equivalent to maximum weight bipartite graph match problem and thus we used Hungarian algorithm to find the optimal solution[84].

S2.5 Color + Registration

In this case, we used both color information and position information of cells to predict cell identities. The features used in this case were $f_{col}(k; i, \boldsymbol{x})$ and $f_{reg}(k; i, \boldsymbol{x})$ as described in S2.1 and S2.4. Note that in this case too, we do not use co-dependent cell features. Cell labels were predicted by minimizing the following energy function

$$y = \arg\min_{y \in \mathcal{Y}} \sum_{ik} -f_{col}(k; i, \boldsymbol{x}) - f_{reg}(k; i, \boldsymbol{x}) \tag{44}$$

Here again we used Hungarian algorithm[84] to find the optimal solution. By comparing prediction accuracy in this case to the Color only case (S2.4), we can gauge the contribution of position information of cells on prediction accuracy. Here again, we accounted for missing cells in images by predicting the identities iteratively (similar to S2.1). In this case too, we observed improvement in prediction accuracy by accounting for missing cells hence all the comparisons are shown for modified case.

Below we briefly show how the objective function in (Eq. 44) naturally arises in registration algorithms that combine multiple features such as color in spatial registration method[85]. First, as defined in registration methods[79,86] that use only spatial location feature $\mathbf{x}_i$ of each cell, the probability of observing each cell is (as described in S2.1)

$$P(\mathbf{x}_i) = \sum_{k=1}^{n} P\left({}^{\mathbf{x}_i}/_{z_k}\right) P(z_k) = \sum_{k=1}^{n} \pi_k \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \mathbf{\Sigma}_k) \tag{45}$$

Here z_k denotes a spatial mixture component with mean $\boldsymbol{\mu}_k$ and covariance $\boldsymbol{\Sigma}_k^S$. If color is also available then to maximize complete data log-likelihood using EM method, we need to define the joint probability of observing both spatial location $\mathbf{x}_i$ and color feature y_i of each cell, $p(\mathbf{x}_i, y_i)$. Next we will discuss two possible cases as discussed in[85] that differ in the way the joint-probability $p(\mathbf{x}_i, y_i)$ is defined.

First case, if the color information is considered to be independent of spatial information then the joint probability factorizes as

$$\begin{aligned} P(\mathbf{x}_i, y_i, z_i = k, c_i = l) &= P\left({}^{x_i}/_{z_i = k}\right) P\left({}^{y_i}/_{c_i = l}\right) P(z_i = k) P(c_i = l) \\ &= \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k^S) F(y_i; \theta_l) \pi_k \pi_l \end{aligned} \tag{46}$$

Here z_i and c_i denote the spatial and color mixture components (latent variables) from which $\mathbf{x}_i$ and y_i are drawn. Also, π_l, θ_l and $F(c_i, \theta_l)$ denote the mixture weights, parameters and density of observing y_i from a mixture density. The independence assumption is akin to assuming that distribution of colors observed in images in different cells is not dependent on spatial location of cells. However, this assumption is not valid since in NeuroPAL each cell is color coded thus observed color depends on spatial location of cells.

In second case, dependence of color on spatial location of cell is accounted for. Further, it is modeled, that for each spatial component (cell), color is drawn from a location specific mixture density. Thus, the joint probability factorizes as

$$P(\mathbf{x}_i, y_i, z_i = k, c_i = l) = P\left({}^{x_i}/_{z_i = k}\right) P\left({}^{y_i}/_{c_i = l, z_i = k}\right) P\left({}^{c_i}/_{z_i = k}\right) P(z_i = k) \tag{47}$$

Next, we need to define $P\left({}^{y_i}/_{c_i, z_i}\right)$ to define complete data likelihood. For NeuroPAL, this is easy since each cell is assigned a unique color [10] i.e. for each spatial component (cell) the color density mixture has only 1 component. Thus, $P\left({}^{y_i}/_{c_i, z_i}\right) = P\left({}^{y_i}/_{z_i}\right)$ and $P\left({}^{c_i}/_{z_i}\right) = 1$.

With updated definitions, the complete likelihood of data is defined as

$$\begin{aligned} P(X, Y, Z, C) &= \prod_i P(\mathbf{x}_i, y_i, z_i, c_i) = \prod_i P\left({}^{x_i}/_{z_i}\right) P\left({}^{y_i}/_{c_i}\right) P(z_i) \\ &= \prod_i \prod_k \pi_k{}^{z_{ik}} \mathcal{N}(\mathbf{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k^S)^{z_{ik}} F(c_i; \theta_k)^{z_{ik}} \end{aligned} \tag{48}$$

Now, if $F(c_i, k)$ is defined as $\exp\left(-(y_i - \mathbf{C}_k)^T \boldsymbol{\Sigma}_k^{c^{-1}} (y_i - \mathbf{C}_k)\right)$ (as described in S2.4) and expected complete data log-likelihood is maximized using the EM method then the M-step is equivalent to maximizing

$$\mathcal{L} = -\sum_{i,k} \alpha_{ik} \left(\| \mathbf{x}_i - \mathcal{T}(\boldsymbol{\mu}_k, \mathbf{W}) \|^2_{\boldsymbol{\Sigma}_k^{S^{-1}}} + (y_i - \mathbf{C}_k)^T \boldsymbol{\Sigma}_k^{c^{-1}} (y_i - \mathbf{C}_k) \right) - \frac{\lambda}{2} tr(\mathbf{W}\mathbf{G}\mathbf{W}^{\mathrm{T}}) \tag{49}$$

Thus, it can be seen that the first term in Eq. 49 is equivalent to Eq. 44.

S2.6 Color + Rel. Position

In this case we combined color information of cells with relative position information of cells. Note that in contrast to S2.5, in this case we use co-dependent features (that is pairwise relative position features) in combination with color information whereas in S2.5, the feature was dependent on cell specific information (absolute position of cells). The unary feature in this case is same as the Color only method (S2.4).

$$f_{col}(k; i, \boldsymbol{x}) = \exp\left(-(\mathbf{C}_i - \mathbf{C}_j)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{C}_i - \mathbf{C}_k)\right) \tag{50}$$

Pairwise features in this case were the same as described in S1.2. Optimization of the objective function in this case was performed using Loopy Belief Propagation algorithm as described in S1.4. Comparing prediction accuracy in this case to the Color + Registration (S2.5) method helps in verifying that the higher accuracy achieved by this method in predicting cell identities is due to co-dependent cell features thus highlighting the importance of such features.

S2.7 Registration + Color + Rel. Position

In this case we combined all the cell independent and co-dependent features in one model. Thus, the unary features in this case were $f_{col}(k; i, \boldsymbol{x})$ and $f_{reg}(k; i, \boldsymbol{x})$ as described in S2.4 and S2.1, and co-dependent features were the same as the described in S1.2. Here again, objective function was optimized using Loopy Belief Propagation algorithm as described in S1.4. We simulated this condition to see if prediction accuracy can be boosted by combining the co-dependent position features with registration algorithm. However, in most cases we saw a decrease in prediction accuracy. This is because the competition between registration term and relative position features term in objective function decreases accuracy.

Code and Data availability

Code and data used in this study can be accessed at https://github.com/shiveshc/CRF_Cell_ID.git. This repository contains the following 1) All code and individual components necessary for using CRF_ID framework to annotate cells in new data, visualize results, and build new atlases based on annotated data 2) Code to reproduce results for comparison shown against other methods in this study, and 3) all raw datasets used in this study as well as human annotations created for those datasets except whole-brain imaging datasets.

Acknowledgments

The authors thank the Hobert Lab for providing NeuroPAL strains. The authors acknowledge the funding support of the U.S. NIH (R21DC015652, R01NS096581, NIH R01GM108962, and R01GM088333) and the U.S. NSF (1764406 and 1707401) to HL. Some nematode strains used in this work were provided by the Caenorhabditis Genetics Center (CGC), which is funded by the NIH (P40 OD010440), National Center for Research Resources and the International *C. elegans* Knockout Consortium. SC and HL designed the algorithm, experiments, and methods. SC collected ground-truth validation data. SC, SL collected whole-brain imaging data. SC, SL, YL, and DSP developed strain with neuronal landmarks; SC, SL, YL, and HL analyzed the data. SC and HL wrote the paper. The authors declare no competing interest. All raw datasets (except whole-brain imaging datasets) used in this study as well as ground-truth human annotations created for those datasets are available at https://github.com/shiveshc/CRF_Cell_ID/tree/master/Datasets.

References

1. 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. *Nat. Methods* **6,** 667–72 (2009).

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

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

4. Schrödel, T., Prevedel, R., Aumayr, K., Zimmer, M. & Vaziri, A. Brain-wide 3D imaging of neuronal activity in Caenorhabditis elegans with sculpted light. *Nat. Methods* **10,** 1013–1020 (2013).

5. Kato, S. *et al.* Global Brain Dynamics Embed the Motor Command Sequence of Caenorhabditis elegans. *Cell* **163,** 656–669 (2015).

6. Venkatachalam, V. *et al. Pan-neuronal imaging in roaming Caenorhabditis elegans. Proceedings of the National Academy of Sciences of the United States of America* **113,** (2016).

7. Nguyen, J. P. *et al.* Whole-brain calcium imaging with cellular resolution in freely behaving Caenorhabditis elegans. *Proc. Natl. Acad. Sci. U. S. A.* 33 (2015). doi:10.1073/pnas.1507110112

8. Williams, A. H. *et al.* Unsupervised Discovery of Demixed, Low-Dimensional Neural Dynamics across Multiple Timescales through Tensor Component Analysis. *Neuron* **98,** 1099–1115.e8 (2018).

9. Kobak, D. *et al.* Demixed principal component analysis of neural population data. *Elife* **5,** (2016).

10. Yemini, E. *et al.* NeuroPAL: A Neuronal Polychromatic Atlas of Landmarks for Whole-Brain Imaging in C. elegans. *bioRxiv* 676312 (2019). doi:10.1101/676312

11. Altun, Z. F. & Hall, D. H. Worm Atlas. *Wormatlas* (2009). doi:10.3908/wormatlas.1.14. Edited for the web by Laura A. Herndon. Last revision: April 30, 2012

12. Szigeti, B. *et al.* OpenWorm: an open-science approach to modeling Caenorhabditis elegans. *Front. Comput. Neurosci.* **8,** (2014).

13. Toyoshima, Y. *et al.* Accurate Automatic Detection of Densely Distributed Cell Nuclei in 3D Space. *PLoS Comput. Biol.* **12,** (2016).

14. Long, F., Peng, H., Liu, X., Kim, S. & Myers, G. Automatic Recognition of Cells (ARC) for 3D images of C. elegans. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **4955 LNBI,** 128–139 (2008).

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

16. Aerni, S. J. *et al.* Automated cellular annotation for high-resolution images of adult Caenorhabditis elegans. in *Bioinformatics* **29,** (2013).

17. Toyoshima, Y. *et al.* An annotation dataset facilitates automatic annotation of whole-brain activity imaging of C. elegans. *bioRxiv* 698241 (2019). doi:10.1101/698241

18. Scholz, M. *et al.* Predicting natural behavior from whole-brain neural dynamics. *bioRxiv* 445643

(2018). doi:10.1101/445643

19. Bronstein, A. M., Bronstein, M. M. & Kimmel, R. Rock, paper, and scissors: Extrinsic vs. intrinsic similarity of non-rigid shapes. in *Proceedings of the IEEE International Conference on Computer Vision* (2007). doi:10.1109/ICCV.2007.4409076

20. Bronstein, A. M., Bronstein, M. M. & Kimmel, R. Topology-invariant similarity of nonrigid shapes. *Int. J. Comput. Vis.* **81,** 281–301 (2009).

21. Bakir, G. H. *et al. Predicting Structured Data (Neural Information Processing)*. (The MIT Press, 2007).

22. Nowozin, S. Structured Learning and Prediction in Computer Vision. *Found. Trends® Comput. Graph. Vis.* **6,** 185–365 (2010).

23. Caelli, T. & Caetano, T. Graphical models for graph matching: Approximate models and optimal algorithms. *Pattern Recognit. Lett.* **26,** 339–346 (2005).

24. Kappes, J. H. *et al.* A Comparative Study of Modern Inference Techniques for Structured Discrete Energy Minimization Problems. *Int. J. Comput. Vis.* **115,** 155–184 (2015).

25. Lafferty, J., McCallum, A. & Pereira, F. C. N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML '01 Proc. Eighteenth Int. Conf. Mach. Learn.* **8,** 282–289 (2001).

26. Sutton, C. & McCallum, A. An Introduction to Conditional Random Fields. *Mach. Learn.* **4,** 267–373 (2010).

27. Kohli, P., Ladický, L. & Torr, P. H. S. Robust higher order potentials for enforcing label consistency. *Int. J. Comput. Vis.* **82,** 302–324 (2009).

28. Murphy, K. P., Weiss, Y. & Jordan, M. I. Loopy belief propagation for approximate inference: An empirical study. *Proc. Uncertain. AI* **9,** 467–475 (1999).

29. Peyre, G., Cuturi, M. & Solomon, J. Gromov-wasserstein averaging of kernel and distance matrices. in *33rd International Conference on Machine Learning, ICML 2016* **6,** 3927–3935 (2016).

30. Solomon, J., Peyré, G., Kim, V. G. & Sra, S. Entropic metric alignment for correspondence problems. in *ACM Transactions on Graphics* **35,** (2016).

31. Mémoli, F. Gromov-Wasserstein Distances and the Metric Approach to Object Matching. *Found. Comput. Math.* **11,** 417–487 (2011).

32. Peng, H., Ruan, Z., Long, F., Simpson, J. H. & Myers, E. W. VAA3D enables real-time 3D visualization and quantitative analysis of large-scale biological image data sets. *Nat. Biotechnol.* **28,** 348-U75 (2010).

33. Quattoni, A., Wang, S., Morency, L.-P., Collins, M. & Darrell, T. Hidden conditional random fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **29,** 1848–1853 (2007).

34. Myronenko, A. & Song, X. Point set registration: Coherent point drifts. *IEEE Trans. Pattern Anal. Mach. Intell.* **32,** 2262–2275 (2010).

35. Gendrel, M., Atlas, E. G. & Hobert, O. A cellular and regulatory map of the GABAergic nervous system of *C. elegans*. *Elife* **5,** 1–38 (2016).

36. Pereira, L. *et al.* A cellular and regulatory map of the cholinergic nervous system of *C. elegans*. *Elife* **4,** (2015).

37. Vidal, B. *et al.* An atlas of Caenorhabditis elegans chemoreceptor expression. *PLOS Biol.* **16,** e2004218 (2018).

38. Bentley, B. *et al.* The Multilayer Connectome of Caenorhabditis elegans. *PLOS Comput. Biol.* **12,** e1005283 (2016).

39. Piatkevich, K. D. *et al.* Population imaging of neural activity in awake behaving mice. *Nature* **574,** 413–417 (2019).

40. Piatkevich, K. D. *et al.* A robotic multidimensional directed evolution approach applied to fluorescent voltage reporters article. *Nat. Chem. Biol.* **14,** 352–360 (2018).

41. Chalasani, S. H. *et al.* Dissecting a circuit for olfactory behaviour in Caenorhabditis elegans. *Nature* **450,** 63–70 (2007).

42. L'Etoile, N. D. & Bargmann, C. I. Olfaction and odor discrimination are mediated by the C. elegans guanylyl cyclase ODR-1. *Neuron* **25,** 575–586 (2000).

43. Bargmann, C. I., Hartwieg, E. & Horvitz, H. R. Odorant-selective genes and neurons mediate olfaction in C. elegans. *Cell* **74,** 515–527 (1993).

44. Cho, Y. *et al.* Multimodal Stimulation in a Microfluidic Device Facilitates Studies of Interneurons in Sensory Integration in C. elegans. *Small* **n/a,** 1905852 (2020).

45. Liu, H. *et al.* Reciprocal modulation of 5-HT and octopamine regulates pumping via feedforward and feedback circuits in C. elegans . *Proc. Natl. Acad. Sci.* **116,** 7107–7112 (2019).

46. Wakabayashi, T. *et al.* In vivo calcium imaging of OFF-responding ASK chemosensory neurons in C. elegans. *Biochim. Biophys. Acta - Gen. Subj.* **1790,** 765–769 (2009).

47. Zaslaver, A. *et al.* Hierarchical sparse coding in the sensory system of Caenorhabditis elegans. *Proc. Natl. Acad. Sci. U. S. A.* **112,** 1185–9 (2015).

48. Zou, H., Hastie, T. & Tibshirani, R. Sparse principal component analysis. *J. Comput. Graph. Stat.* **15,** 265–286 (2006).

49. Finlayson, G. D., Schiele, B. & Crowley, J. L. Comprehensive colour image normalization. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **1406,** 475–490 (1998).

50. Wang, F., Vemuri, B. C., Rangarajan, A. & Eisenschenk, S. J. Simultaneous nonrigid registration of multiple point sets and atlas construction. *IEEE Trans. Pattern Anal. Mach. Intell.* **30,** 2011–2022 (2008).

51. Evangelidis, G. D., Kounades-Bastian, D., Horaud, R. & Psarakis, E. Z. A generative model for the joint registration of multiple point sets. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **8695 LNCS,** 109–122 (2014).

52. Najafi, M., Taghavi Namin, S., Salzmann, M. & Petersson, L. Non-associative higher-order markov networks for point cloud classification. in *Lecture Notes in Computer Science (including subseries*

Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **8693 LNCS,** 500–515 (2014).

53. Komodakis, N. & Paragios, N. Beyond pairwise energies: Efficient optimization for higher-order mrfs. in *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009* **2009 IEEE,** 2985–2992 (2009).

54. Krähenbühl, P. & Koltun, V. in *Advances in Neural Information Processing Systems 24* (eds. Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F. & Weinberger, K. Q.) 109–117 (Curran Associates, Inc., 2011).

55. Wainwright, M. J. & and Jordan, M. I. Graphical Models, Exponential Families, and Variational Inference. *Found. Trends{®} Mach. Learn.* **1,** 1–305 (2007).

56. Nitzan, M., Karaiskos, N., Friedman, N. & Rajewsky, N. Gene expression cartography. *Nature* (2019). doi:10.1038/s41586-019-1773-3

57. Robie, A. A. *et al.* Mapping the Neural Substrates of Behavior. *Cell* **170,** 393–406.e28 (2017).

58. Kim, Y. *et al.* Mapping social behavior-induced brain activation at cellular resolution in the mouse. *Cell Rep.* **10,** 292–305 (2015).

59. Chen, Y. *et al.* An active texture-based digital atlas enables automated mapping of structures and markers across brains. *Nat. Methods* **16,** 341–350 (2019).

60. Ronneberger, O. *et al.* ViBE-Z: A framework for 3D virtual colocalization analysis in zebrafish larval brains. *Nat. Methods* **9,** 735–742 (2012).

61. Vaadia, R. D. *et al.* Characterization of Proprioceptive System Dynamics in Behaving Drosophila Larvae Using High-Speed Volumetric Microscopy. *Curr. Biol.* **29,** 935–944.e4 (2019).

62. Stiernagle, T. Maintenance of C. elegans. *WormBook : the online review of C. elegans biology* 1–11 (2006). doi:10.1895/wormbook.1.101.1

63. Lee, H. *et al.* A multi-channel device for high-density target-selective stimulation and long-term monitoring of cells and subcellular features in C. elegans. *Lab Chip* **14,** 4513–4522 (2014).

64. Chu, J. *et al.* A bright cyan-excitable orange fluorescent protein facilitates dual-emission microscopy and enhances bioluminescence imaging in vivo. *Nat. Biotechnol.* **34,** 760–767 (2016).

65. Redemann, S. *et al.* Codon adaptation-based control of protein expression in C. elegans. *Nat. Methods* **8,** 250–252 (2011).

66. Lyssenko, N. N., Hanna-Rose, W. & Schlegel, R. A. Cognate putative nuclear localization signal effects strong nuclear localization of a GFP reporter and facilitates gene expression studies in Caenorhabditis elegans. *Biotechniques* **43,** 596–600 (2007).

67. McIntire, S. L., Reimer, R. J., Schuske, K., Edwards, R. H. & Jorgensen, E. M. Identification and characterization of the vesicular GABA transporter. *Nature* **389,** 870–876 (1997).

68. Yu, S., Avery, L., Baude, E. & Garbers, D. L. Guanylyl cyclase expression in specific sensory neurons: A new family of chemosensory receptors. *Proc. Natl. Acad. Sci. U. S. A.* **94,** 3384–3387 (1997).

69. Sjöstrand, K., Clemmensen, L. H., Einarsson, G., Larsen, R. & Ersbøll, B. SpaSM: A MATLAB toolbox

for sparse statistical modeling. *J. Stat. Softw.* **84,** (2018).

70. Stephens, G. J., Johnson-Kerner, B., Bialek, W. & Ryu, W. S. Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput. Biol.* **4,** (2008).

71. Leordeanu, M. & Hebert, M. A spectral technique for correspondence problems using pairwise constraints. in *Proceedings of the IEEE International Conference on Computer Vision* **II,** 1482–1489 (2005).

72. Leordeanu, M. & Hebert, M. Unsupervised learning for graph matching. *2009 IEEE Conf. Comput. Vis. Pattern Recognit.* 864–871 (2009). doi:10.1109/CVPR.2009.5206533

73. Taskar, B., Chatalbashev, V. & Koller, D. Learning associative Markov networks. in *Twenty-first international conference on Machine learning - ICML '04* 102 (2004). doi:10.1145/1015330.1015444

74. Taskar, B., Guestrin, C. & Koller, D. Max-margin Markov networks. *Adv. Neural Inf. Process. Syst. 16 - NIPS'03* 25–32 (2003). doi:10.1.1.129.8439

75. Ikeda, S., Tanaka, T. & Amari, S. Stochastic Reasoning, Free Energy, and Information Geometry. *Neural Comput.* **16,** 1779–1810 (2004).

76. Schmidt, M. UGM: A Matlab toolbox for probabilistic undirected graphical models. http://www.cs.ubc.ca/~schmidtm/Software/UGM.html, 2007. (2007).

77. Sutton, C. & McCallum, A. Piecewise pseudolikelihood for efficient training of conditional random fields. in *ACM International Conference Proceeding Series* **227,** 863–870 (2007).

78. Hyvärinen, A. Consistency of pseudolikelihood estimation of fully visible Boltzmann machines. *Neural Comput.* **18,** 2283–2292 (2006).

79. Myronenko, A. & Song, X. Point-Set Registration: Coherent Point Drift. *Adv. Neural Inf. Process. Syst. 19* 1009–1016 (2009). doi:10.1109/TPAMI.2010.46

80. Panaganti, V. & Aravind, R. Robust nonrigid point set registration using graph-laplacian regularization. in *Proceedings - 2015 IEEE Winter Conference on Applications of Computer Vision, WACV 2015* 1137–1144 (2015). doi:10.1109/WACV.2015.156

81. Ge, S., Fan, G. & Ding, M. Non-rigid point set registration with global-local topology preservation. in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* 245–251 (2014). doi:10.1109/CVPRW.2014.45

82. Ma, J. *et al.* Robust L2E estimation of transformation for non-rigid registration. *IEEE Trans. Signal Process.* **63,** 1115–1129 (2015).

83. Chui, H. & Rangarajan, A. A new point matching algorithm for non-rigid registration. *Comput. Vis. Image Underst.* **89,** 114–141 (2003).

84. Kuhn, H. W. The Hungarian Algorithm for the Assignment Problem. *Nav. Res. Logist. Q.* **2,** 83–97 (1955).

85. Danelljan, M., Meneghetti, G., Khan, F. S. & Felsberg, M. A Probabilistic Framework for Color-Based Point Set Registration. in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 1818–1826 (2016). doi:10.1109/CVPR.2016.201

86. Jian, B. & Vemuri, B. C. A Robust Algorithm for Point Set Registration Using Mixture of Gaussians. *Proc. IEEE Int. Conf. Comput. Vis.* **2,** 1246–1251 (2005).