

---

# PROBABILISTIC SEGMENTATION AND LABELING OF *C. elegans* NEURONS

---

A PREPRINT

immediate

May 29, 2020

## ABSTRACT

Automatic identification and segmentation of the neurons of *C. elegans* enables evaluating nervous system mutations, positional variability, and allows us to conduct high-throughput population studies employing many animals. A recently introduced strain of *C. elegans*, termed "NeuroPAL", has enabled the efficient annotation of neurons and the construction of a statistical atlas of their positions. Previous atlas-based segmentation approaches have modelled images of cells as a mixture model. The expectation maximization (EM) algorithm and its variants are used to find the (local) maximum likelihood parameters for this class of models. However, we can only incorporate our prior information about the proportion of observed data points associated with each cluster in a soft manner in these existing methods. We present a variation of EM algorithm called SinkhornEM (sEM) that uses Sinkhorn iterations to enforce hard constraints on the marginals of the joint distribution of observed variables and latent assignments in order to inject our prior information about the cluster-data assignment proportions. We apply our method to the problem of segmenting and labeling neurons in fluorescent microscopy images of *C. elegans* specimens. We show empirically that sEM outperforms existing constrained mixture models both in terms of achieving higher-likelihood and higher-labeling accuracy.

## 1 Introduction

Whole-brain functional imaging of *Caenorhabditis elegans* has been recently introduced to enable the measurement of neural activity at unprecedented temporal and spatial resolution [Kato et al., 2015]. Obtaining a complete measurement of neuron positioning and activity enables the study of a wide range of hypotheses including the identification of brain-wide dynamic networks involved in action sequences and decisions, decoding of nervous system responses to repulsive and attractive stimuli from distinct modalities and monitoring of neuronal identity, to reveal neuronal-fate defects in mutation scenarios [Nguyen et al., 2016]. However, a significant analysis bottleneck is the segmentation and identification of all imaged neurons which, so far, has been done either manually or reduced to a smaller subset of imaged neurons.

Several attempts in developing automated cell labeling and segmentation have been reported. Similar to our work most of these studies use an atlas to of cells. Some groups have developed techniques only for the labeling problem [Hirose et al., 2018, Tokunaga et al., 2014, Aerni et al., 2013, Toyoshima et al., 2019]. One study used ellipsoid fitting to find regions associated with each cell but, due to a lack of unique markers, they were unable to resolve cell identities [Toyoshima et al., 2016]. A recent study investigated both segmentation and labeling [Chaudhary et al., 2020] but used two separate models to solve these two problems. Another work formulated the segmentation problem in terms of pixel classification and updated cell centers using non-parametric equations [Qu et al., 2011].

A novel transgenic strain of *C. elegans* called NeuroPAL (a Neuronal Polychromatic Atlas of Landmarks) has introduced differential fluorescent coloring of neurons to help resolve unique neural identities [Yemini et al., 2019]. Here, we present a statistical pipeline for joint segmentation and labeling of neural identities in the NeuroPAL images. We formulate the segmentation problem as a posterior inference over the latent variables of a mixture model and show that the neural labeling arises naturally from our formulation. We further present a novel technique to constrain the poste-

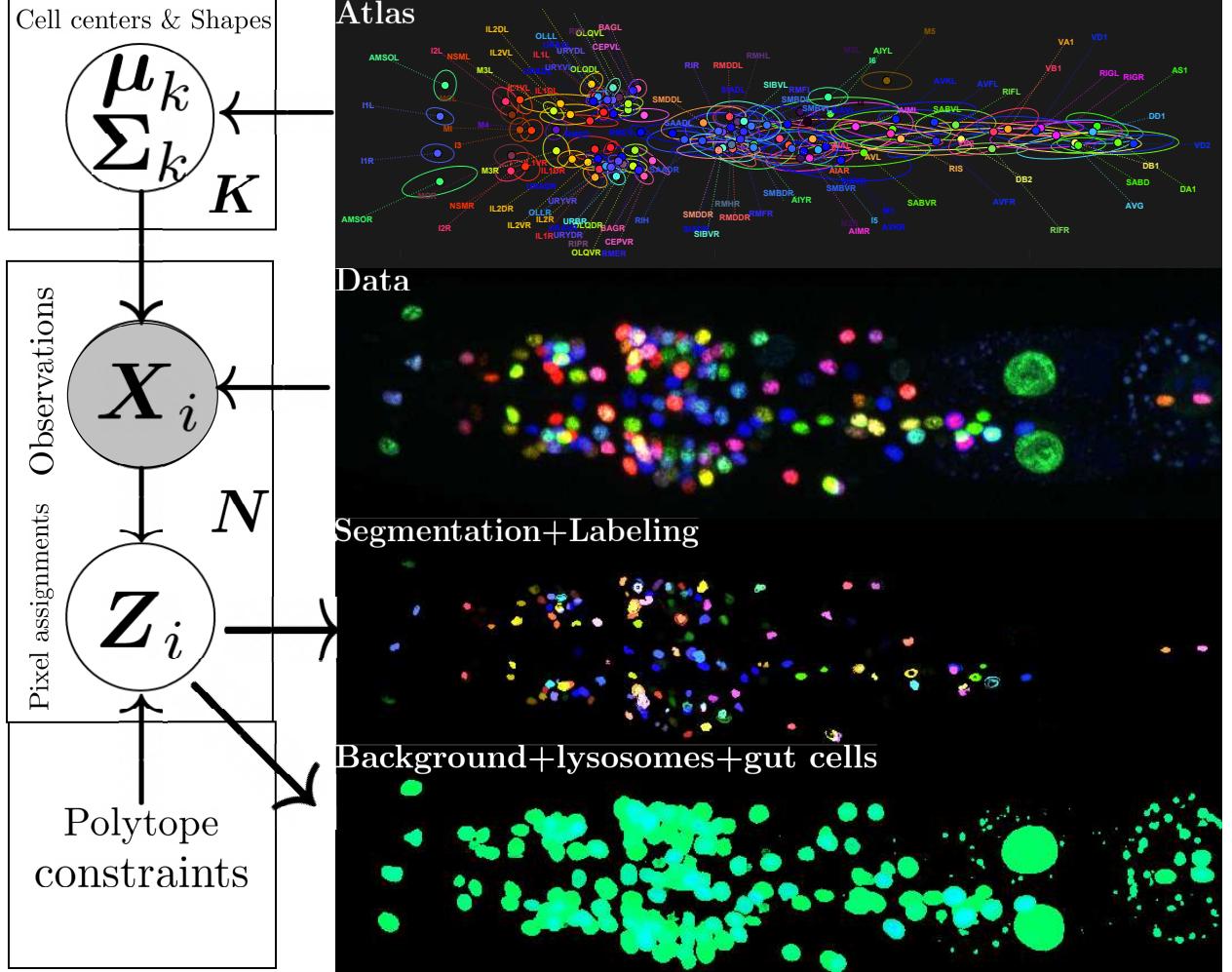


Figure 1: Segmentation and labelling of fluorescently-colored neurons, in image volumes of NeuroPAL worms, using the proposed method. Left: The graphical model of the probabilistic inference procedure employed to identify and segment neurons. Right: The model uses atlas parameters as a prior to assign latent neuron identities to each observed pixel subject to polytope constraints that account for cell mass.

rior distribution in the expectation-maximization (EM) algorithm using the Birkhoff polytope [Linderman et al., 2017] and utilizing the Sinkhorn-Knopp algorithm [Sinkhorn, 1967] for inference. Our experimental results illustrate that our method outperforms vanilla EM and achieves superior qualitative and quantitative performance in terms of segmentation quality and model fitting.

## 2 Method

The segmentation of neurons and assignment of their identities within the images of model organism *C. elegans* has applications ranging from cell-fate tracking throughout development (including mutant phenotypes) to extracting calcium signals from behaving worms. Here we describe how the proposed method can be used to efficiently identify regions corresponding to each cell in the volumetric images of the *C. elegans*. The nervous system of *C. elegans* incorporates a considerable amount of stereotypy and, therefore, cell positioning is similar across worms, with some bounds on variability. Taking advantage of this stereotypy, the authors of [Yemini et al., 2019] built NeuroPAL, a novel biological technique that colors the neurons of *C. elegans* using combinations of fluorescent proteins in order to identify every neuron in the worm via its color and ganglion position. This technique provides a deterministic color scheme that permits us to resolve neuronal identities with unprecedented accuracy. Using expert annotations of the neural centers and identities in NeuroPAL image volumes, they quantify the variability of cell locations and colors and

construct a statistical atlas that encodes the canonical neuron positions and colors in the form of a multivariate normal distribution.

We formulate the problem of identifying cell locations, colors, and shapes as a Gaussian Mixture Model with neurons as its components. Instead of applying a two-step procedure to first detect the neurons and then use their shape and color information to identify them, we bypass the detection step by directly treating the pixels of the image as observations of our mixture model. We use the pre-trained statistical atlas as a prior distribution over the cell centers and colors. We assume that the atlas is roughly aligned to the image and, in the last section, we demonstrate that our method is capable of tuning the alignment even if the atlas is slightly misaligned with the image. In all the experiments we use 4-11 neuron annotations to find a rough initial alignment.

### 2.0.1 EM algorithm:

A typical mixture model can be written as follows:

$$P(\mathbf{X}, \boldsymbol{\theta}) = \sum_{k=1}^K \pi_k P_k(\mathbf{X}|\boldsymbol{\theta}) P(\boldsymbol{\theta})$$

Where  $\pi_k$ s are the component proportions,  $\mathbf{X}$  is the observed data, and  $\boldsymbol{\theta}$  is the set of model parameters (means  $\boldsymbol{\mu}_{1:K}$  and covariances  $\boldsymbol{\Sigma}_{1:K}$  in the Gaussian case). The  $P(\boldsymbol{\theta})$  represents our prior information about the model parameters. To find the (local) maximum likelihood estimate of the model parameters a common strategy is to introduce a latent assignment variable  $Z$  that assigns each observation to one of the mixture components. We then maximize the expected complete log likelihood where the expectation is taken under the posterior distribution of the assignment variable. This function lower bounds the log likelihood (posterior) and maximizing it improves the likelihood.

$$P(\mathbf{X}|Z = k) = F_k(\mathbf{X}) \quad (1)$$

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}_n) = \mathbb{E}_{P_{\boldsymbol{\theta}_n}(Z|\mathbf{X})}[\log P_{\boldsymbol{\theta}}(\mathbf{X}, \boldsymbol{\theta}|Z)] \quad (2)$$

$$\begin{cases} \text{E-step:} & \text{evaluate } Q(\boldsymbol{\theta}|\boldsymbol{\theta}_n) \\ \text{M-step:} & \text{solve } \boldsymbol{\theta}_{n+1} = \arg \max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}_n) \end{cases} \quad (3)$$

The E-step consists of computing a matrix  $\gamma$  known as the responsibility matrix with  $\gamma_{kj} = P(Z = k|\mathbf{X}_j, \boldsymbol{\theta})$ . By definition the row sum of this matrix is 1; however, in the vanilla EM algorithm, the only way to control the column sum of this matrix is through the constraints on component proportions or distribution-specific component parameters (such as covariances for the Gaussian distribution). Both types of constraints effectively act as regularization on  $\gamma$  matrix.

### 2.0.2 Constraining the assignment matrix $\gamma$ :

In applications such as image segmentation, often there exists some information about the rough count of pixels for each component. To incorporate this information into the EM algorithm we explicitly put constraints on the column sum of  $\gamma$  matrix while keeping the row sum normalized to one. In each iteration of EM we aim to find a matrix  $\hat{\gamma}$  that is close to  $\gamma$  while satisfying  $\sum_j \gamma_{kj} = 1$  and  $\sum_k \gamma_{kj} = K\alpha_j$  with predetermined  $\alpha_j$ s that sum to 1. The  $\alpha_j$  encodes our prior knowledge about the expected number of observations for component  $j$ . Introduced by [Cuturi, 2013], the Sinkhorn algorithm efficiently approximates the solution to this constrained problem by iteratively normalizing the row and column of  $\gamma$  matrix to the desired values (Fig. 1). Theoretical results guarantee the convergence of this algorithm to the optimal solution in the constrained set [Idel, 2016]. We observed that in practice, these constraints on the row and column sums not only are effective in avoiding local minima of the log likelihood, but also they improve the labeling accuracy (Fig. 3).

In the context of cell labeling the posterior distribution over the latent assignment variable  $Z$  can be seen as a soft probabilistic segmentation of the image. Since  $Z_i$  is a categorical distribution that probabilistically assigns each pixel observation  $\mathbf{X}_i$  to a neuron component in the mixture model, then the regions with high posterior probability under the model provide a segmentation of the image into  $K$  neuronal components and  $B$  background components.

In the vanilla EM algorithm this posterior distribution is usually spread over multiple components and the original formulation of EM does not allow hard constraints on the posterior. However, in the *C. elegans* NeuroPAL cell segmentation and many other similar problems, the number of pixels per component is approximately known a priori. To incorporate this information in each iteration of EM we enforce row-sum and column-sum constraints on the posterior matrix  $P(Z_i|\mathbf{X}_j)$ . These constraints make the probabilistic segmentation maps sharper and more localized which results in better parameter estimates in the next EM iterations. We term our proposed algorithm as Sinkhorn Expectation

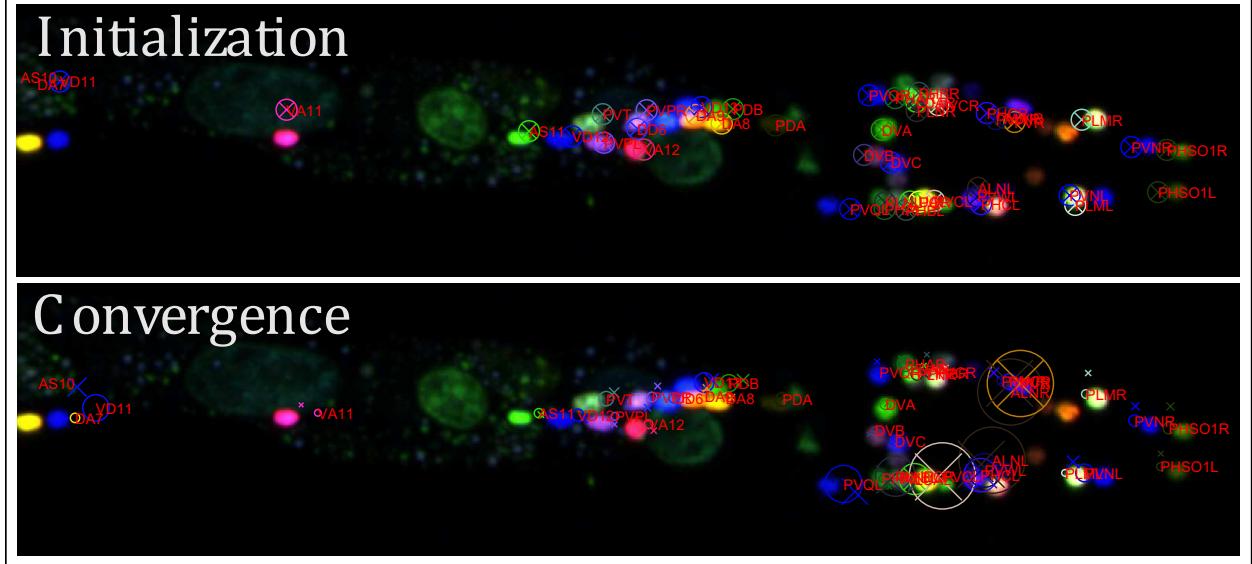


Figure 2: The update of the prior locations based on multiple covariance regression. Colored crosses are the prior locations, colored circles are the GMM locations.

Maximization (sEM) and show that it outperforms the vanilla EM both qualitatively and quantitatively in the results section.

Each observed image is in the form of a 4-dimensional tensor where the first 3 dimensions are spatial coordinates  $x$ ,  $y$ ,  $z$  and the 4th dimension represents the color channels. We reformat the data into  $N$  tuples  $\mathbf{X}_i = (\mathbf{l}_i, \mathbf{c}_i) \in \mathbb{R}^{3+C}$  where  $\mathbf{l}_i = (x_i, y_i, z_i) \in \mathbb{R}^3$  corresponds to the location of pixel  $i$ .  $C$  is the total number of color channels and  $\mathbf{c}_i = (c_i^1, \dots, c_i^C)$  is a vector indicating the color intensity of pixel  $i$ . To untangle the neurons from the background noise and non-neuronal components, such as lysosomes and gut cells, we further add a few components to the mixture model that are uniformly distributed in space and normally distributed in color. More formally, the functional form of the distributions that we use for the cell identification is as following:

$$P(\mathbf{X}, \boldsymbol{\theta}) = F(\boldsymbol{\theta}) \sum_{k=1}^{K+B} \pi_k F_k(\mathbf{X}) \quad (4)$$

$$F_k(\mathbf{X} = [\mathbf{l}, \mathbf{c}]) = \begin{cases} \mathcal{N}(\mathbf{l} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) & k \leq K \\ \mathcal{U}(\mathbf{l} | \mathbf{l}_{\min}, \mathbf{l}_{\max}) \mathcal{N}(\mathbf{c} | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) & k > K \end{cases} \quad (5)$$

Here  $\boldsymbol{\theta} = \{\boldsymbol{\mu}_{1:K}, \boldsymbol{\Sigma}_{1:K}\}$  is the set of model parameters and we choose  $F(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\mu}_{1:K} | \boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a)$  where  $\boldsymbol{\mu}_a$  and  $\boldsymbol{\Sigma}_a$  are the hyperparameters provided by the pre-trained statistical atlas. Components 1 to  $K$  represent the spatial distribution of the pixels corresponding to cells and components  $K + 1$  to  $K + B$  represent background and artifact components in different colors. In practice, we find that 3 background components can capture the noise and artifact pixels accurately. Once we specify the functional form of the distributions, we can analytically derive the quantities  $P(Z|\mathbf{X}, \boldsymbol{\theta})$  and  $\log P(\mathbf{X}, Z, \boldsymbol{\theta})$  using the following equations and write down  $Q(\boldsymbol{\theta}|\boldsymbol{\theta}_n)$  by taking the expectation of the complete log likelihood with respect to the posterior distribution of the assignment variable:

$$\gamma_k \triangleq P_{\boldsymbol{\theta}_n}(Z = k | \mathbf{X}, \boldsymbol{\theta}) = \frac{\pi_k F_k(\mathbf{X}) \mathcal{N}(\boldsymbol{\mu}_{1:K} | \boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a)}{[\sum_{k'} \pi_{k'} F_{k'}(\mathbf{X})] \mathcal{N}(\boldsymbol{\mu}_{1:K} | \boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a)} \quad (6)$$

$$\log P_{\boldsymbol{\theta}}(\mathbf{X}_{1:N}, Z = k, \boldsymbol{\theta}) = \sum_{i=1}^N \mathbb{I}[Z = k] (\log \pi_k + \log F_k(\mathbf{X}_i)) + \log \mathcal{N}(\boldsymbol{\mu}_{1:K} | \boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a) \quad (7)$$

**Algorithm 1** Sinkhorn Expectation Maximization

---

**Input:**  $\mathbf{X}_{1:N} = [\mathbf{l}, \mathbf{c}]_{1:N} \in \mathbb{R}^{N \times D}$ , statistical atlas mean  $\boldsymbol{\mu}_{a,1:K} \in \mathbb{R}^{K \times D}$ , statistical atlas covariances  $\boldsymbol{\Sigma}_{a,1:K} \in \mathbb{R}^{D \times D \times K}$ , component proportions  $\boldsymbol{\alpha} \in \mathbb{R}^{K+B}$ , entropic regularization:  $\lambda > 0$ , convergence tolerance  $\epsilon > 0$ .

**Initialization:**  $\boldsymbol{\mu} = \boldsymbol{\mu}_a$ ,  $\boldsymbol{\Sigma}_{1:K} = \mathbb{I}_D$ ,  $\boldsymbol{\Sigma}_{K+1:K+B} = \mathbb{I}_C$

**while** not converged **do**

- $n \leftarrow n + 1$
- ▷ Updating  $\boldsymbol{\gamma}$ :
- for**  $k = 1, \dots, K$  **do**

  - Set LP =  $-\frac{1}{2}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_{a,k})^T \boldsymbol{\Sigma}_{a,k}^{-1} (\boldsymbol{\mu}_k - \boldsymbol{\mu}_{a,k}) - \frac{D}{2} \log 2\pi - \frac{1}{2} \log \det \boldsymbol{\Sigma}_{a,k}$
  - Set LL =  $-\frac{1}{2} \boldsymbol{\Sigma}_k^{-1} (\mathbf{X} - \boldsymbol{\mu}_k) (\mathbf{X} - \boldsymbol{\mu}_k)^T - \frac{D}{2} \log 2\pi - \frac{1}{2} \log \det \boldsymbol{\Sigma}_k$
  - $\gamma_k \leftarrow \log \pi_k + \text{LL} + \text{LP}$

- end for**
- for**  $k = K + 1, \dots, B$  **do**

  - Set LL =  $-\log(l_{\max} - l_{\min}) - \frac{1}{2} (\boldsymbol{\Sigma}_k^c)^{-1} (\mathbf{c} - \boldsymbol{\mu}_k^c) (\mathbf{c} - \boldsymbol{\mu}_k^c)^T - \frac{C}{2} \log 2\pi - \frac{1}{2} \log \det \boldsymbol{\Sigma}_k^c$
  - $\gamma_k \leftarrow \log \pi_k + \text{LL}$

- end for**
- $\boldsymbol{\gamma} \leftarrow \exp(\boldsymbol{\gamma} - \log \sum \exp \gamma)$
- ▷ Computing  $\hat{\boldsymbol{\gamma}}$ :
- Set  $\hat{\boldsymbol{\gamma}} = \text{sinkhorn}(\boldsymbol{\gamma}, \boldsymbol{\alpha}, \lambda)$  [Algorithm 3.2 in [\[Idle, 2016\]](#)]
- ▷ Updating  $\boldsymbol{\mu}$ :
- for**  $k = 1, \dots, K$  **do**

  - $\boldsymbol{\mu}_k \leftarrow [\sum_i \hat{\boldsymbol{\gamma}}_{k,i} \mathbf{1}_D - \boldsymbol{\Sigma}_{a,k} \boldsymbol{\Sigma}_k^{-1}]^{-1} [\sum_i \hat{\boldsymbol{\gamma}}_{k,i} \mathbf{X}_i - \boldsymbol{\Sigma}_k \boldsymbol{\Sigma}_{a,k}^{-1} \boldsymbol{\mu}_{a,k}]$

- end for**
- ▷ Updating  $\boldsymbol{\Sigma}$ :
- for**  $k = 1, \dots, K$  **do**

  - $\boldsymbol{\Sigma}_k \leftarrow \frac{\sum_i \hat{\boldsymbol{\gamma}}_{k,i} (\mathbf{X}_i - \boldsymbol{\mu}_k)^T (\mathbf{X}_i - \boldsymbol{\mu}_k)}{\sum_i \hat{\boldsymbol{\gamma}}_{k,i}}$

- end for**
- ▷ Compute log evidence and check convergence:
- Sample  $Z_i^{\text{sample}}$  from  $\gamma_{:,i}$
- Log evidence:  $\text{LE} = Z^{\text{sample}} \odot \hat{\boldsymbol{\gamma}}$
- Check convergence:  $\|\text{LE}^{(n)} - \text{LE}^{(n-1)}\|_2 \stackrel{?}{\leq} \epsilon$
- end while**
- return** Model parameters:  $\boldsymbol{\theta} = \{\boldsymbol{\mu}_{1:K}, \boldsymbol{\Sigma}_{1:K}\}$ , Posterior distribution  $P(Z|\mathbf{X}, \boldsymbol{\theta})$ .

---

$$\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}_n) = & \sum_{k,i} \gamma_k(\mathbf{X}_i) \left[ \log \pi_k - \frac{1}{2} \log \det \boldsymbol{\Sigma}_k - \frac{D}{2} \log 2\pi \right. \\
& - \frac{1}{2} (\mathbf{X}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{X}_i - \boldsymbol{\mu}_k) - \frac{1}{2} \log \det \boldsymbol{\Sigma}_a \\
& \left. - \frac{DK}{2} \log 2\pi - \frac{1}{2} (\boldsymbol{\mu}_{1:K} - \boldsymbol{\mu}_a)^T \boldsymbol{\Sigma}_a^{-1} (\boldsymbol{\mu}_{1:K} - \boldsymbol{\mu}_a) \right]
\end{aligned} \tag{8}$$

The E-step involves computing the  $\boldsymbol{\gamma}$  matrix and running entropic regularized optimal transport to obtain the properly normalized  $\hat{\boldsymbol{\gamma}}$  but, for the M-step we need to take the gradient of the  $Q$  with respect to model parameters  $\boldsymbol{\mu}_{1:K}$  and

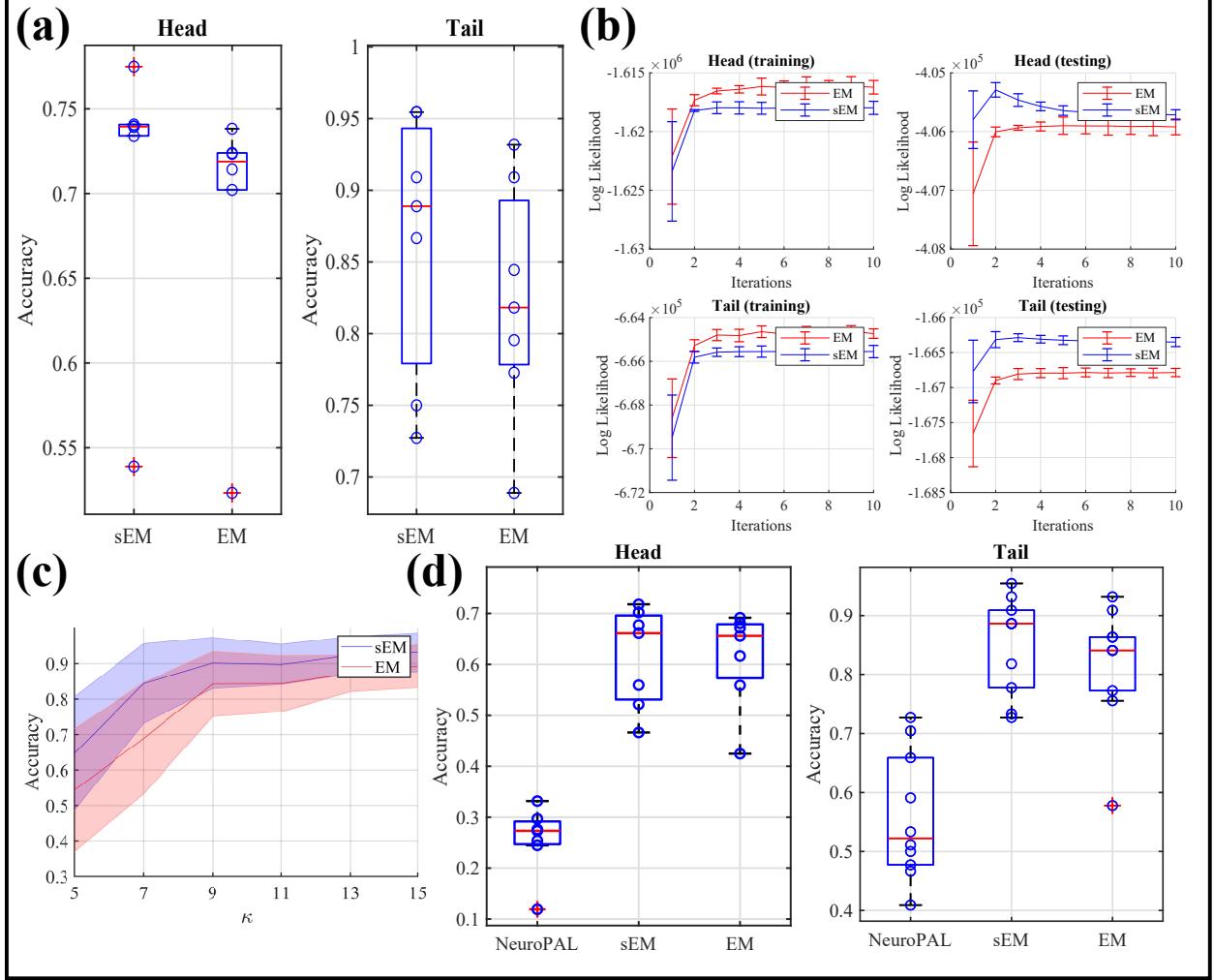


Figure 3: (a) The accuracy of the proposed method (blue) vs. vanilla EM method for correct labelling of neurons. (b) The log-evidence of the model fit for the proposed method (blue) and vanilla EM (red). (c) The accuracy of sEM and EM as a function of the number of randomly chosen landmark cells. (d) The out of sample accuracy comparison between NeuroPAL [Yemini et al., 2019], sEM and EM.

$\Sigma_{1:K}$  as following:

$$\begin{aligned} \frac{\partial Q(\boldsymbol{\theta}|\boldsymbol{\theta}_n)}{\partial \boldsymbol{\mu}_k} = 0 &\Rightarrow \boldsymbol{\mu}_k^{n+1} = \left[ \sum_i \hat{\gamma}_{k,i} \mathbf{1}_D + \boldsymbol{\Sigma}_{a,k} \boldsymbol{\Sigma}_k^{-1} \right]^{-1} \\ &\quad \left[ \sum_i \hat{\gamma}_{k,i} \mathbf{X}_i + \boldsymbol{\Sigma}_k \boldsymbol{\Sigma}_{a,k}^{-1} \boldsymbol{\mu}_{a,k} \right] \\ \frac{\partial Q(\boldsymbol{\theta}|\boldsymbol{\theta}_n)}{\partial \boldsymbol{\Sigma}_k} = 0 &\Rightarrow \boldsymbol{\Sigma}_k^{n+1} = \frac{\sum_i \hat{\gamma}_{k,i} (\mathbf{X}_i - \boldsymbol{\mu}_k)^T (\mathbf{X}_i - \boldsymbol{\mu}_k)}{\sum_i \hat{\gamma}_{k,i}} \end{aligned}$$

This way of formulating the problem enables us to implement a stochastic version of sEM in which each EM iteration uses a randomly sampled subset of data to update the sufficient statistics. In our experiments we observed that the stochastic optimization is much faster but it takes more iterations to converge.

### 2.0.3 Updating the prior:

The alignment between the statistical atlas and the image plays a crucial role in the proposed method. In our experiments, we use 4-11 landmark neurons (annotated by an expert) to find an initial, rough alignment. To correct for the

mismatch between this initial alignment and the optimal alignment, after each iteration of the sEM algorithm, we solve the following multiple covariance regression problem to update the alignment.

$$\hat{\beta}^{n+1} = \underset{\beta}{\operatorname{argmin}} \sum_{k=1}^K \|\mu_{a,k}^n \beta - \mu_k^n\|_{\Sigma_{a,k}}, \quad \mu_{a,k}^{n+1} = \mu_{a,k}^n \hat{\beta}^{n+1}$$

### 3 Results

We ran the proposed algorithm on a dataset consisting of 6 worm heads and 7 worm tails. We quantified the convergence of both EM and sEM by monitoring the changes in the log evidence across iterations (Fig. 3). For each worm image, we first spatially smoothed it using a small 3D Gaussian filter (width  $.5 \mu\text{m}$  in each dimension). We then removed the low intensity background pixels using a small threshold to ensure that the pixels corresponding to neurons are not removed from the image (70-th percentile). We observed that both EM and sEM converge after a maximum of 10 iterations. Each method outputs the cell centers, colors, and shapes in terms of covariances as well as a  $\gamma$  matrix that includes neuron-specific probabilistic segmentation maps. We use the cell centers to quantify our proposed algorithm and compare it to the EM and we use the probabilistic segmentation maps for qualitative comparisons. We compute the accuracy of each method by counting the number of mixture components that are within a radius of 3 microns from their true location (annotated by an expert), dividing this by the total number of neurons. As shown in Fig 3 sEM outperforms EM significantly. In practice we observed that updating the prior always increases the accuracy. To show this, we looked at the prior cell locations in iterations 1 and 10 and observed that the atlas better aligns with the image after updating the prior (Fig. 2). In all the cases sEM has higher labeling accuracy compared to EM. Also, sEM always achieves a better testing likelihood while the training likelihood is greater for EM. This means that sEM is injecting prior information into the model that is compatible with data and hence it avoids overfitting (Fig. 3).

To monitor the quality of the segmentation, as illustrated in Fig. 1, we observed that the sEM provides sharper and more localized segments for cells with boundaries that are more visible and more similar to the original dataset. We also demonstrate another application of our method in Fig. 5, creating a database of cell-nucleus shapes in different pictures to enable further variability analysis and nuclear cell-shape atlas construction.

#### 3.1 Robustness to initialization and landmark cells

To evaluate the robustness of sEM and vEM to the selection of landmark cells we did the following experiment: For  $\kappa \in \{5, 7, 9, 11, 13\}$  we randomly selected a subsample of cells with size  $\kappa$  and ran both sEM and vEM using those randomly selected cells as landmark cells for computing the initial alignment. As expected, the average accuracy increases with more landmark cells but the best results are given if the landmark cells are carefully chosen, meaning that they are spread in the image and cover the borders Fig. 3.

We further evaluated the robustness of our algorithms to initialization. Instead of initializing the centers and colors using the atlas we initialized them using a random subset of the points selected uniformly from the observed pixels. Because of the strong prior information of the cell centers and colors both sEM and vEM algorithms nicely reorder the components to the right order and after 3 iterations the accuracy of the methods reach %80 (std = 0.07) and %79 (std = 0.1) with larger std for vEM. An observation that we made is that when initializing randomly larger values of lambda provide better accuracy and faster convergence (lambda = 10 vs. lambda = 1).

#### 3.2 Comparison

The original NeuroPAL paper [Yemini et al., 2019] proposes a two-step detect and identify procedure for cell labeling. First some candidate location for cells are detected using a greedy algorithm. Once the initial point cloud is detected the user refines the outcome by manually removing mis-detects. The refined point cloud is then passed to a point set registration algorithm that aligns to the atlas by an exhaustive search in the rotation space. Finally, Hungarian (or Sinkhorn) algorithm is used to label the cells.

To show the utility of using a combined approach as opposed to a two-step procedure we compared our method to NeuroPAL. We used the same subset of landmark neurons to align the atlas to the point cloud in both methods. To avoid double-dipping, we trained the atlas based on all the other images and used that to evaluate the accuracy on the image at hand. The average accuracy of the segmentation in the tails is  $0.55 \pm 0.1$  for NeuroPAL,  $0.82 \pm 0.08$  for sEM, and  $0.77 \pm 0.09$  for vEM (Fig. 3).

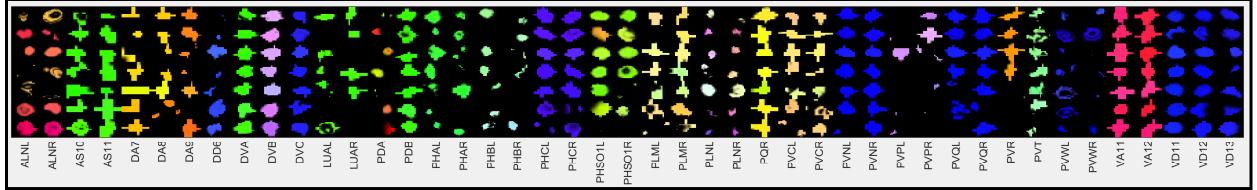


Figure 4: The extracted shapes of *C. elegans* tail neuronal nuclei. Head nuclei are in supplementary.

## 4 Conclusion

In this paper we studied the problem of probabilistic cell segmentation and labeling in the nematode *C. elegans*. We formulated the segmentation problem as a mixture model and derived the equations for updating the model parameters. To deal with the background and artifact pixels in the image, we added a few components to the mixture model and distributed them uniformly in space. We then showed how the segmentation arises naturally from our formulation by simply taking the posterior matrix of the assignment variable  $P(Z_i|X_j)$ . To incorporate our information about the pixel count per component, we enforced hard constraints on the  $\gamma$  matrix which can be done efficiently using the Sinkhorn algorithm. The probabilistic nature of the algorithm enables the encoding of uncertainty in the segmentation, allowing a robust means of analyzing microscopy images.

## References

- [Aerni et al., 2013] Aerni, S. J., Liu, X., Do, C. B., Gross, S. S., Nguyen, A., Guo, S. D., Long, F., Peng, H., Kim, S. S., and Batzoglou, S. (2013). Automated cellular annotation for high-resolution images of adult caenorhabditis elegans. *Bioinformatics (Oxford, England)*, 29(13):i18–i26. 23812982[pmid]. 1
- [Chaudhary et al., 2020] Chaudhary, S., Lee, S. A., Li, Y., Patel, D. S., and Lu, H. (2020). Automated annotation of cell identities in dense cellular images. *bioRxiv*. 1
- [Cuturi, 2013] Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in neural information processing systems*, pages 2292–2300. 3
- [Hirose et al., 2018] Hirose, O., Kawaguchi, S., Tokunaga, T., Toyoshima, Y., Teramoto, T., Kuge, S., Ishihara, T., Iino, Y., and Yoshida, R. (2018). Spf-celltracker: Tracking multiple cells with strongly-correlated moves using a spatial particle filter. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(6):1822–1831. 1
- [Idel, 2016] Idel, M. (2016). A review of matrix scaling and sinkhorn’s normal form for matrices and positive maps. 3, 5
- [Kato et al., 2015] Kato, S., Kaplan, H. S., Schrödel, T., Skora, S., Lindsay, T. H., Yemini, E., Lockery, S., and Zimmer, M. (2015). Global brain dynamics embed the motor command sequence of caenorhabditis elegans. *Cell*, 163(3):656–669. 1
- [Linderman et al., 2017] Linderman, S. W., Mena, G. E., Cooper, H., Paninski, L., and Cunningham, J. P. (2017). Reparameterizing the birkhoff polytope for variational permutation inference. *arXiv preprint arXiv:1710.09508*. 2
- [Nguyen et al., 2016] Nguyen, J. P., Shipley, F. B., Linder, A. N., Plummer, G. S., Liu, M., Setru, S. U., Shaevitz, J. W., and Leifer, A. M. (2016). Whole-brain calcium imaging with cellular resolution in freely behaving caenorhabditis elegans. *Proceedings of the National Academy of Sciences*, 113(8):E1074–E1081. 1
- [Qu et al., 2011] Qu, L., Long, F., Liu, X., Kim, S., Myers, E., and Peng, H. (2011). Simultaneous recognition and segmentation of cells: application in c.elegans. *Bioinformatics (Oxford, England)*, 27(20):2895–2902. 21849395[pmid]. 1
- [Sinkhorn, 1967] Sinkhorn, R. (1967). Diagonal equivalence to matrices with prescribed row and column sums. *The American Mathematical Monthly*, 74(4):402–405. 2
- [Tokunaga et al., 2014] Tokunaga, T., Hirose, O., Kawaguchi, S., Toyoshima, Y., Teramoto, T., Ikebata, H., Kuge, S., Ishihara, T., Iino, Y., and Yoshida, R. (2014). Automated detection and tracking of many cells by using 4d live-cell imaging data. *Bioinformatics (Oxford, England)*, 30(12):i43–i51. 24932004[pmid]. 1
- [Toyoshima et al., 2016] Toyoshima, Y., Tokunaga, T., Hirose, O., Kanamori, M., Teramoto, T., Jang, M. S., Kuge, S., Ishihara, T., Yoshida, R., and Iino, Y. (2016). Accurate automatic detection of densely distributed cell nuclei in 3d space. *PLOS Computational Biology*, 12(6):1–20. 1

[Toyoshima et al., 2019] Toyoshima, Y., Wu, S., Kanamori, M., Sato, H., Jang, M. S., Oe, S., Murakami, Y., Teramoto, T., Park, C., Iwasaki, Y., Ishihara, T., Yoshida, R., and Iino, Y. (2019). An annotation dataset facilitates automatic annotation of whole-brain activity imaging of *c. elegans*. *bioRxiv*. [1](#)

[Yemini et al., 2019] Yemini, E., Lin, A., Nejatbakhsh, A., Varol, E., Sun, R., Mena, G. E., Samuel, A. D., Paninski, L., Venkatachalam, V., and Hobert, O. (2019). Neuropal: A neuronal polychromatic atlas of landmarks for whole-brain imaging in *c. elegans*. *bioRxiv*. [1](#), [2](#), [6](#), [7](#), [10](#)

## 5 Supplemental Figures

### 5.1 Results on individual worm datasets

Below is the accuracy of sEM and vEM on individual files:

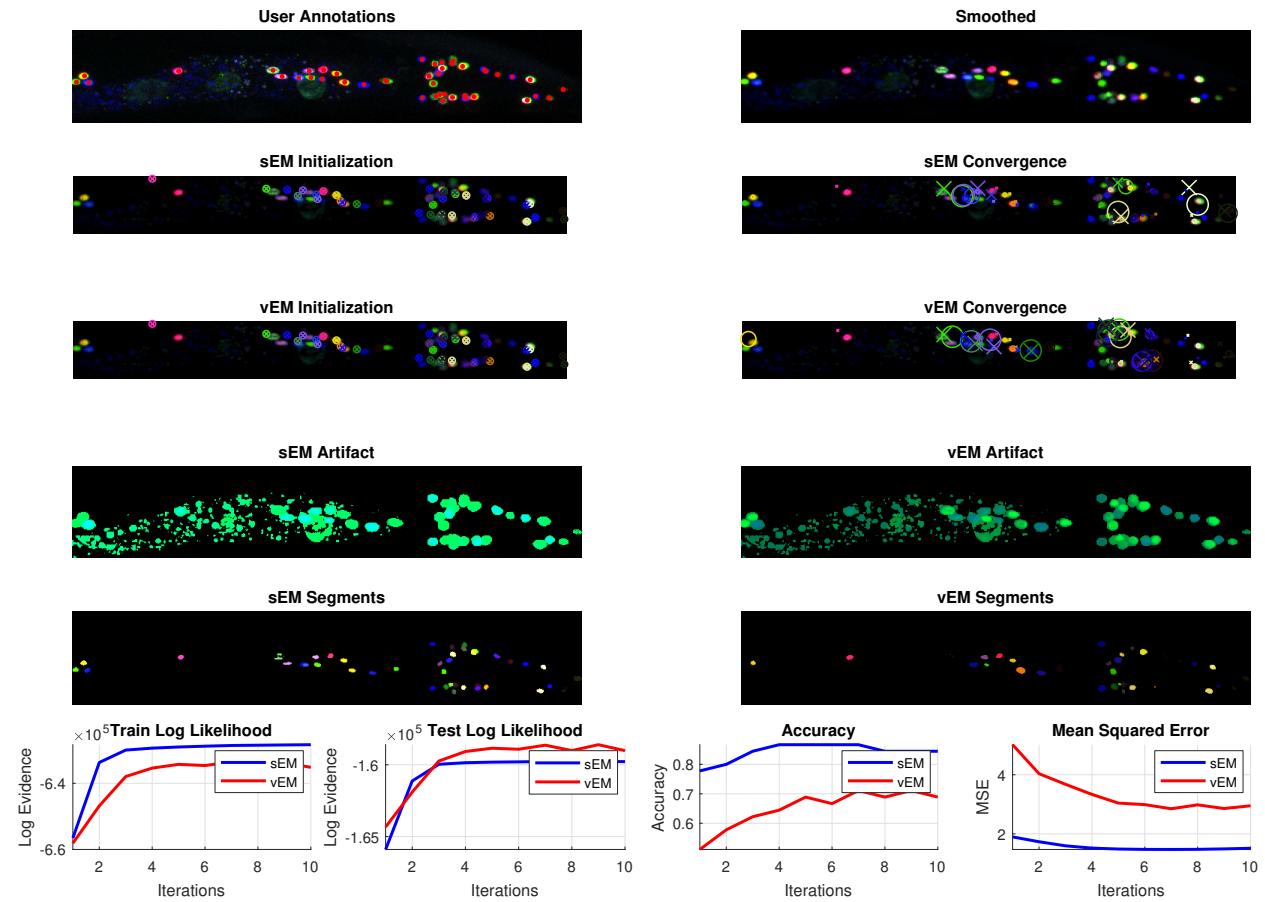
Table 1: Comparison of the accuracy of sEM and vEM algorithms for individual tail data files.

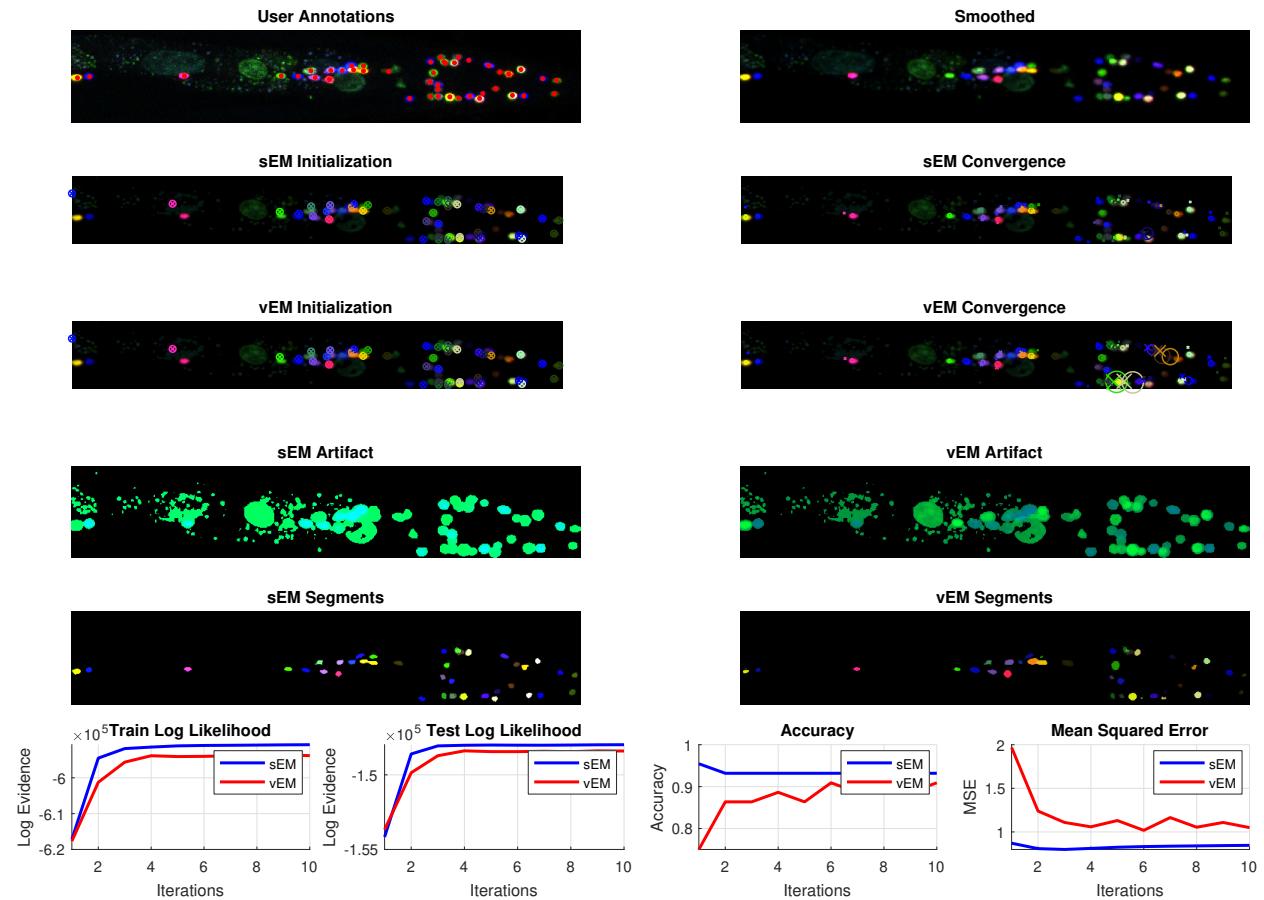
<b>sEM</b>	0.90	0.88	0.95	0.95	0.72	0.86	0.75
<b>vEM</b>	0.90	0.68	0.93	0.81	0.77	0.84	0.79

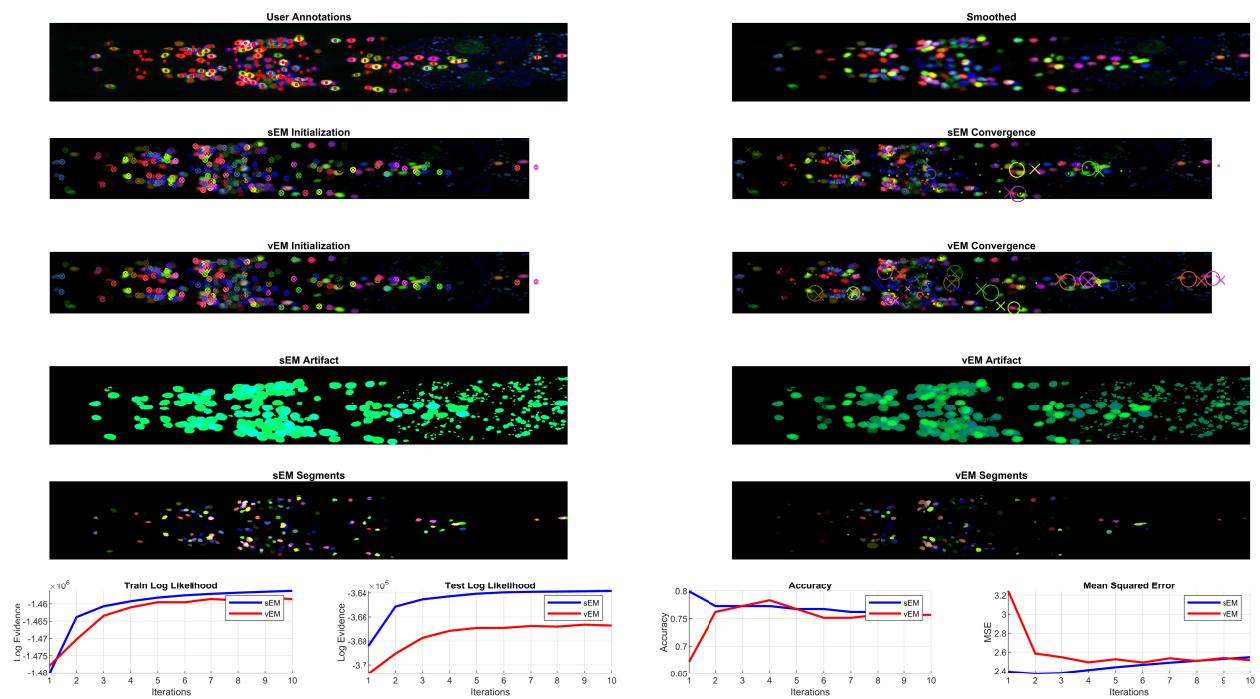
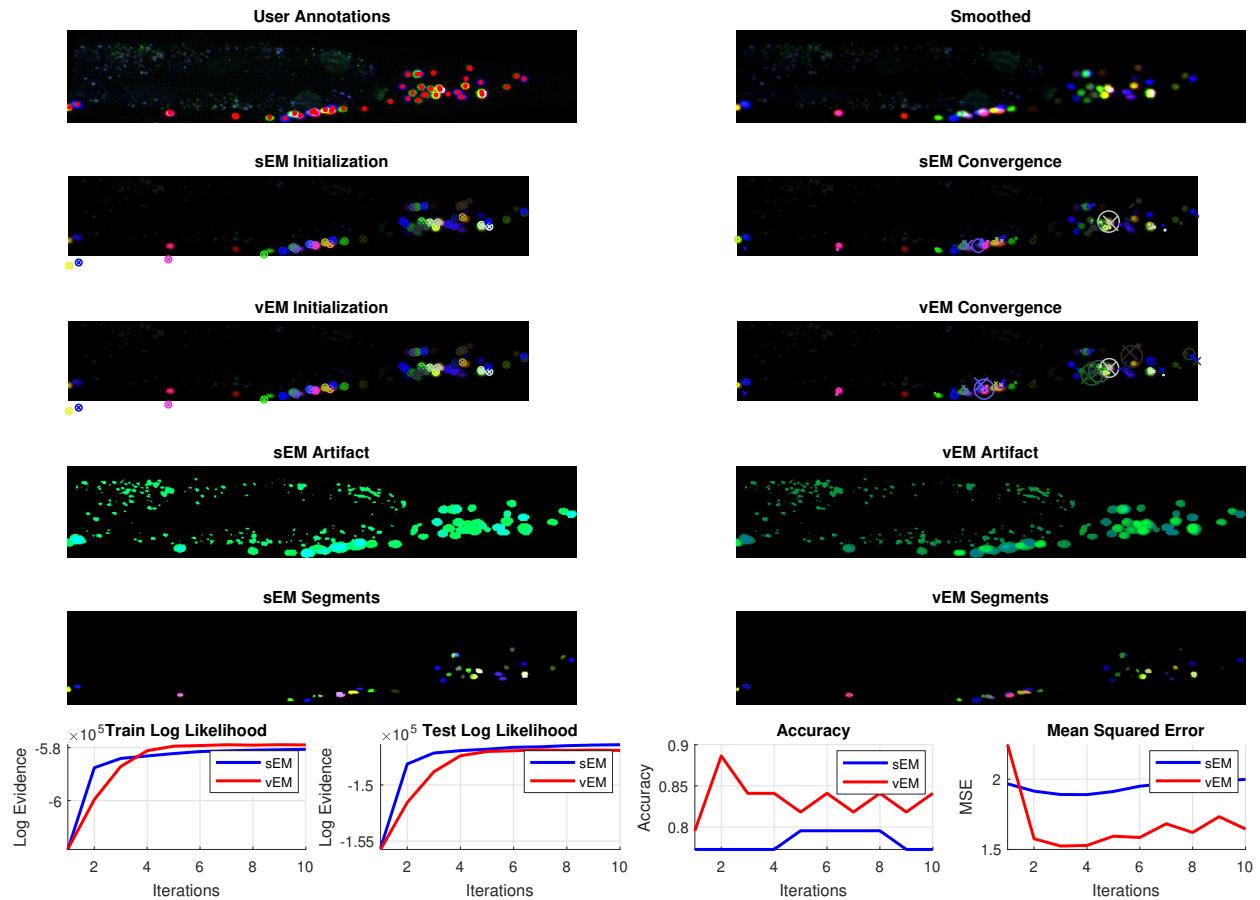
Table 2: Comparison of the accuracy of sEM and vEM algorithms for individual head data files.

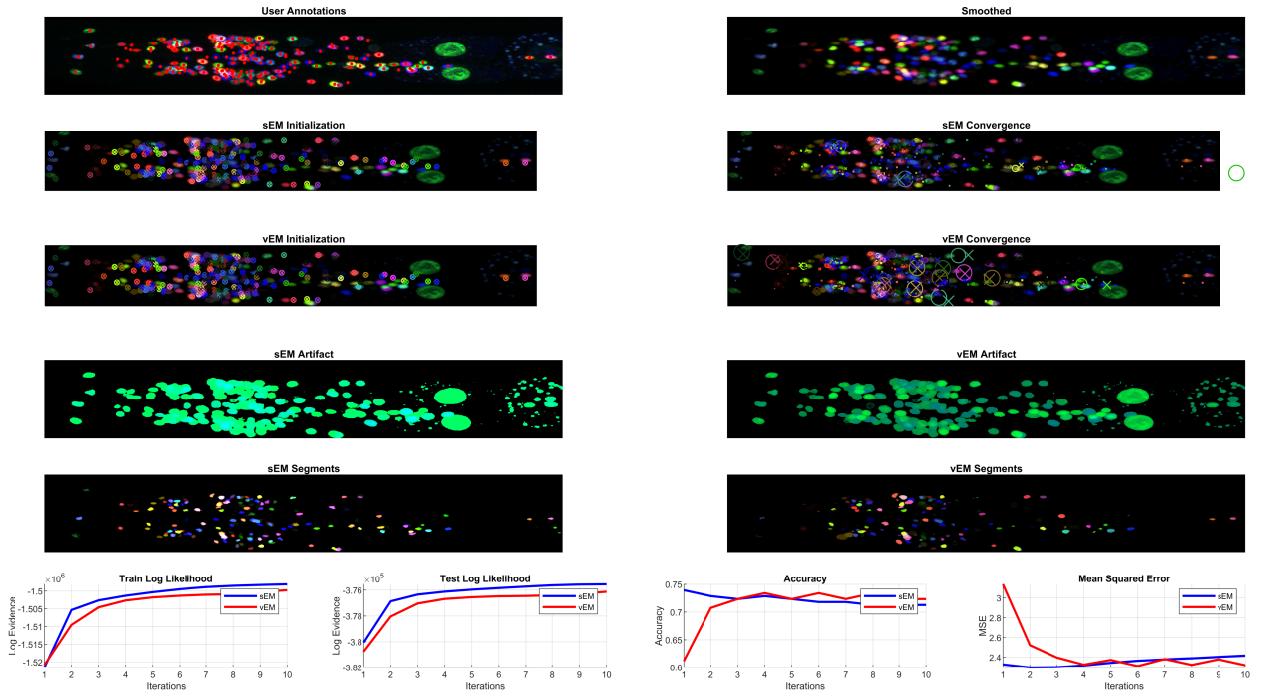
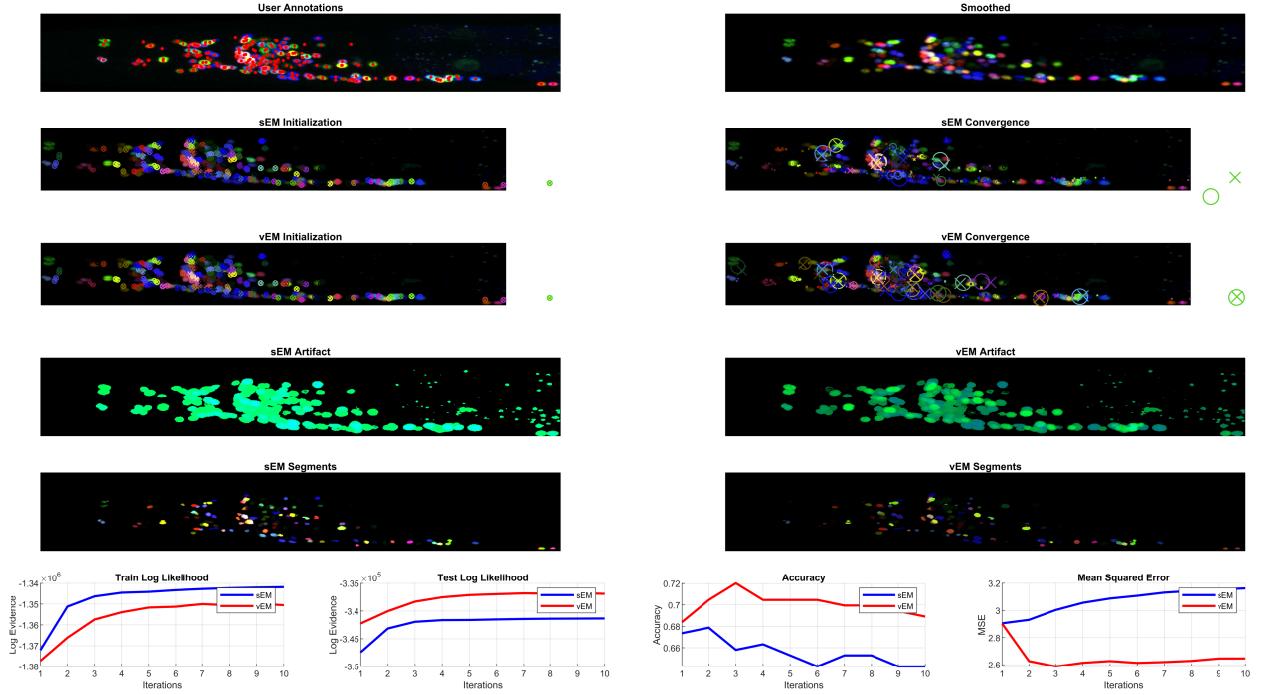
<b>sEM</b>	0.77	0.74	0.73	0.73	0.53	0.73
<b>vEM</b>	0.73	0.71	0.72	0.70	0.52	0.70

For each of the datasets we have expert annotations of the cell centers shown as red dots in the subfigure (1,1). Starting from the original RGB volumetric image we first smooth the image in each channel with a small Gaussian filter (.5  $\mu\text{m}$ ) to reduce the effect of noise. The Gaussian filtered image is shown in the subfigure (1,2). We then use a few landmark cells (6 cells for the tail and 11 cells for the head) to solve a multiple covariance registration and find an initial rough alignment between a pre-trained statistical atlas [Yemini et al., 2019] and the image. The result of this alignment is shown in subfigures (2,1) and (3,1) for sEM, vanilla EM (vEM) where each circle (and cross) represents the location and color of an atlas cell aligned to the current image. The size of the radius of each circle is proportional to the largest eigenvalue of the covariance matrix associated with that neuron ( $\sigma_k$ ). The bigger the size the more uncertain the methods is about the location or color of the cell. Since in all experiments we initialize  $\mu_k$  as  $\mu_k^a$  and  $\sigma_k$  as  $\sigma I_6$  the circles all have equal sizes in the initialization subfigures (2,1) and (3,1). After running the algorithm for 10 iterations the training log-likelihood converges and the inferred centers and colors are fairly stable. This is shown in the subfigures (2,2), (3,2) for sEM and vEM where the crosses correspond to prior locations (*i.e.* aligned atlas locations after a few update iterations) and the circles correspond to  $\mu_k$ . The size of each circle uses the same convention as subfigure (2,1). Once the algorithm is converged we can reformat the  $N$  by  $K$   $\gamma$  matrix into a  $(W, H, D, K)$  tensor where the volumetric image at channel  $k$  shows a probabilistic assignment of the pixels to the component  $k$ . If we color these probabilistic maps according to the inferred cell colors we get a segmentation image where each color represents high probability regions in the image for a cell in that color. The segmentation result is shown in subfigures (4,1) through (4,6) where we max projected the resulting 3D image onto the  $Z$  dimension. The more crisp this image is the more certain the method is about the pixels associated with each component. An observation that we made was that sEM provides more crisp segmentation with clearly separated borders and edges than vEM thanks to polytope constraints. The subfigures (4,1) and (4,2) represents the probability density of the noise components max projected on the  $Z$  dimension. The noise and artifact (lysosomes and gut cell) pixels are captured by the noise components and because we are max projecting on the  $Z$  dimension the noise pixels around the 3D cells cover the entire area of the cells. The four subfigures in the last row show the convergence of the log-likelihood for both sEM and vEM methods as well as the evaluation of both algorithms through iterations.









## 5.2 Neuron shape atlas

As described in the paper our algorithm returns the  $\gamma$  matrix which can be considered as a probabilistic assignment of the pixels to neurons. For each neuron we consider all the pixels with non-zero assignment probability for that neuron and we color those pixels by the color inferred by sEM for that neuron. Repeating this procedure for all the files and all the neurons we create a shape atlas for every neuron in the head and tail of *C. elegans* shown in figure 5

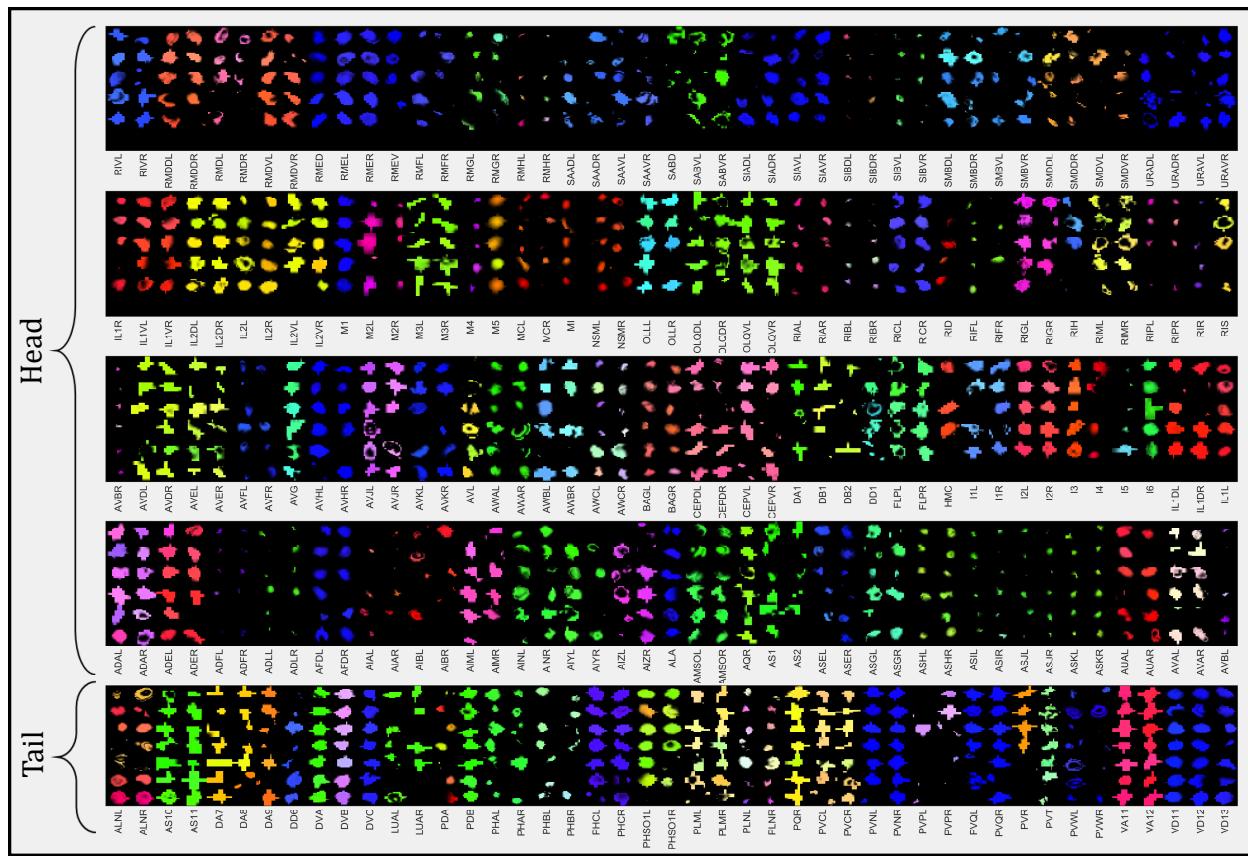


Figure 5: The extracted shapes of *C. elegans* head and tail neuronal nuclei.