frontiers in Artificial Intelligence

ORIGINAL RESEARCH
published: 29 June 2021
doi: 10.3389/frai.2021.668395

Topological Data Analysis of *C. elegans* Locomotion and Behavior

Ashleigh Thomas[1], Kathleen Bates[1], Alex Elchesen[2†], Iryna Hartsock[2†], Hang Lu[1] and Peter Bubenik[2]*

[1]School of Chemical and Biomolecular Engineering, Georgia Institute of Technology, Atlanta, GA, United States, [2]Department of Mathematics, University of Florida, Gainesville, FL, United States

We apply topological data analysis to the behavior of *C. elegans*, a widely studied model organism in biology. In particular, we use topology to produce a quantitative summary of complex behavior which may be applied to high-throughput data. Our methods allow us to distinguish and classify videos from various environmental conditions and we analyze the trade-off between accuracy and interpretability. Furthermore, we present a novel technique for visualizing the outputs of our analysis in terms of the input. Specifically, we use representative cycles of persistent homology to produce synthetic videos of stereotypical behaviors.

Keywords: persistent homology, topological data analysis, delay embedding, sliding window embedding, *C. elegans*, behavior phenotyping

OPEN ACCESS

Edited by:
Kathryn Hess, École Polytechnique Fédérale de Lausanne, Switzerland

Reviewed by:
Jose Andres Perea, Michigan State University, United States
Samir Chowdhury, Stanford University, United States

****Correspondence:***
Ashleigh Thomas
althomas41@gmail.com

[†]These authors have contributed equally to this work

Specialty section:
This article was submitted to Machine Learning and Artificial Intelligence, a section of the journal Frontiers in Artificial Intelligence

Received: *16 February 2021*
Accepted: *12 May 2021*
Published: *29 June 2021*

Citation:
Thomas A, Bates K, Elchesen A, Hartsock I, Lu H and Bubenik P (2021) Topological Data Analysis of C. elegans Locomotion and Behavior. Front. Artif. Intell. 4:668395. doi: 10.3389/frai.2021.668395

1 INTRODUCTION

Model organisms are indispensable in understanding basic principles of biology. Studies of model organisms have played a major role in discoveries of disease mechanisms, disease treatment, and neuroscience principles. The behavior of these model organisms can illuminate responses and phenotypes important for understanding the effects of experimental conditions on subjects. Behavior can be affected by neuron activity, external stimuli, and past experiences (learning), so being able to adequately measure and compare behaviors is a useful evaluation tool for a wide range of experiments.

We propose persistent homology as a new tool for assessing behavior of *Caenorhabditis elegans*, worms that are a widely used model organism. Persistence has been successfully used to study high-dimensional time series, especially those that exhibit some quasi-periodic behavior like the undulation of *C. elegans* (Tralie, 2016; Tralie and Perea, 2018). But to the authors' knowledge, persistent homology has not been previously used to analyze *C. elegans* behavior, though it and similar techniques have been used to study *C. elegans* neural data (Petri et al., 2013; Backholm et al., 2015; Sizemore et al., 2019; Helm et al., 2020; Lütgehetmann et al., 2020).

In this paper we use persistent homology to study the locomotion of *C. elegans* in two settings. In our initial study (**Section 3.1**), we follow one worm as it moves on the surface of an agar plate. Under these conditions there are no barriers to movement and the locomotion is both smooth and complex. We show that persistent homology is able to detect and differentiate between various characteristic behaviors such as forward crawling, backward crawling, and transitioning between the two. We also show a unique contribution of persistence: the synthesis of skeleton data of *C. elegans* performing stereotyped, periodic behaviors. This translates into videos of, for example, forward crawling that are smooth when looped (see **Supplementary Material** for an example). Furthermore, this mapping from persistence features to behavior gives a concrete and biologically relevant interpretation of

results: features of interest—such as a feature that is detected in one sample and not another—can be expressed as videos of synthetic behavior.

We also analyze data from an experiment of the effect of environment on *C. elegans* (**Section 3.2**). In this setting a more controlled environment is required, so the organisms are submerged in a solution and confined to wells in microfluidic devices. Our main results study *C. elegans*' locomotion in solutions that have various levels of viscosity. We show that we are able to use persistent homology—and average persistence landscapes in particular—to summarize *C. elegans* locomotion in a way that allows the classification of the viscosity of an animal's environment with a high level of accuracy, and in fact a much higher level of accuracy than simpler methods based on speed and variety of postures. Our results indicate that persistent homology is a promising tool for quantifying the impact of changes to genotype and environment on *C. elegans* locomotion.

1.1 Related Work

Caenorhabditis elegans is a free-living soil nematode that has been a workhorse genetic model system. The nematode's transparent tissue, simple anatomy, and fast reproduction contribute to both ease in culture and a literal window into the internal workings of a living organism. Its completely sequenced genome contains many genes that are homologous to human genes, and importantly the ability to manipulate genes with relative ease makes it an extremely attractive model system. For neuroscience in particular, *C. elegans* presents a unique opportunity with its simple nervous system (just 302 neurons) that is complex enough to exhibit many sensory modalities, including mechanosensation, chemosensation, and response to heat, osmolarity, and smell.

Behavior characterization in *C. elegans* was historically qualitative, mainly relying on experimentalists specifying end-point assessment (e.g. whether the worm chemotaxes to a particular source of odor within a certain amount of time), or experimentalists using heuristics to assess behavior (e.g. naming worms genes "*unc*" for uncoordinated). In the last decade, machine vision tools first replaced human identifications of worms from images and videos, which allows much larger dynamic datasets to be annotated and analyzed. In recent years, further development in quantitative behavior characterization tools such as tracking (Stirman et al., 2011; Swierczek et al., 2011; Husson et al., 2012; Yemini et al., 2013; Porto et al., 2019), eigenworms (Stephens et al., 2008), behavior "dictionaries" (Brown et al., 2013), and t-SNE (Berman et al., 2016; Liu et al., 2018) have moved the field away from merely describing the outcome to understanding the types of behavior the brain of this simple system can generate. While many of these techniques do well in quantitatively describing behavior and distinguishing differences in behavior, behavioral dynamics are rich and opportunities abound in exploring behavioral dynamics using other mathematical tools.

Persistent homology has been used to analyze time series data in many different settings. Some earlier work was theoretical and studied the interaction between persistence and sliding window embeddings—which we used in this research—as well as proposed possible applications (Firas and Elizabeth, 2015; Perea and Harer, 2015; Perea, 2016). Research into gene expression has used persistent homology to detect patterns or classify whether a signal is periodic (Dequéant et al., 2008; Perea et al., 2015). Frequently, persistence has been used to study neural data (Petri et al., 2013; Backholm et al., 2015; Stolz et al., 2017; Sizemore et al., 2019; Helm et al., 2020; Lütgehetmann et al., 2020), and in many cases neural data from *C. elegans*, but the analysis tends to rely on clique complexes as the topological space of interest instead of sliding window embeddings.

2 MATERIALS AND METHODS

In this section we describe the collection and preprocessing of experimental data (**Section 2.1**), mathematical background (**Sections 2.2** and **2.3**), and pipeline for using topological data analysis on *C. elegans* behavior data (**Section 2.4**).

2.1 Description of Data

C. elegans (N2 strain) were cultured at 20°C under standard conditions on agar plates seeded with OP50 *E. Coli*. Animals were age-synchronized via hatch-off and cultured on plate until they reached day 1 of adulthood. For behavior experiments on agar, animals were prepared, imaged, and tracked as previously described (Porto et al., 2019). For behavior experiments in methylcellulose media, synchronized populations were then washed off of culture plates with M9 buffer. Unless otherwise noted, video data was collected on a dissecting microscope (Leica MZ16) using a CMOS camera (Thorlabs DCC3240M), with a frame rate of 30 frames per second and a magnification of ×1.2.

Behavior data was collected with animals confined with microfluidic devices. In these devices the cavities in which worms are loaded have only slightly greater depth than the width of an adult worm, which restricts worms to the focal plane of the microscope and to almost entirely 2-dimensional behavior. Microfluidic devices were fabricated as described previously (Chung et al., 2011). Methylcellulose solutions were prepared at concentrations of 0.5%, 1%, 2%, and 3% weight in volume of M9 buffer. To ensure that single animals could be isolated in single chambers of the unbonded microchamber microfluidic device, we first picked animals onto a room-temperature, unseeded plate. To ensure that animals were fully immersed in methylcellulose mixture, we used a glass pipet to aspirate a small amount of methylcellulose solution, and then aspirated animals from the unseeded plate one at a time into the methylcellulose solution. Then, single animals surrounded by methylcellulose mixture were pipetted into individual chambers of an unbonded PDMS chamber device. The device could then be flipped over onto a sterile 10 cm Petri dish and gently pressed down until the individual chamber walls came into contact with the Petri dish, preventing animals from leaving their chambers. Animals were then imaged in devices for about 5 min at 30 frames per second, resulting in time series data with 10,665 points.

To extract midline data from videos, we first found masks for each frame to isolate the worm from the background using a combination of Otsu thresholding (Otsu, 1979), image smoothing

using a Gaussian kernel, and size filtration. Otsu thresholding is a thresholding algorithm based on the gray-level histogram of an image. The threshold is identified by the grayscale pixel value that minimizes the intra-class variance of background and foreground pixels. We then broadly followed the method used in Stephens et al. (Stephens et al., 2008) to represent the worm's posture in "worm-centric" coordinates. Briefly, we found the midline of the worm in each frame by thinning the mask to a single line and interpolating between pixels of this line such that the midline was represented by 101 evenly spaced points. We calculated the tangent angle between each pair of adjacent points along the midline so that the animal's posture could be represented as a vector of angles, and then transformed those vectors with PCA so users could balance accuracy requirements and resource limitations via truncation of the data. We replaced frames in which animals were self-occluded with the data from the most recent non-self-occluded frame. We used untruncated PCA data for most computations because it has the same persistence output as the raw angle data. We used truncated PCA data (the first five principal components) for the computations in **Section 3.1**.

The videos for this study were selected from a much larger set of data based on how well they could be segmented and skeletonized. Some videos have subsequences that are difficult to automatically skeletonize because the animals self-occlude, i.e. bend in such a way as to cross over themselves. Thus, this dataset is likely biased toward less complex behaviors like thrashing and there are some cases where there are multiple videos of the same animal. The resulting data set has 40 samples of 10,665 points each with 10 samples for each viscosity condition.

2.2 Sliding Window Embeddings

Sliding window embeddings turn time series data into point cloud data in a way that does not forget the temporal information of the time series. There are some additional benefits to sliding window embeddings, including that they "separate" points that intersect each other in a time series, such as in Examples 2.6 and 2.7.

Definition 2.1: A time series is a sequence of vectors $(x_t)_{t\in T} = (x_t, x_{t+1}, x_{t+2}, \ldots)$ where each x_t is in the same finite-dimensional vector space V and T is a totally ordered set.

Remark 2.2: The totally ordered set T, which indexes the time series, can be $\mathbb{Z}$, $\mathbb{N}$, or a finite set like $[N] = \{1, 2, \ldots, N\}$. For many applications including the ones in this paper, the indexing set is finite and will be omitted in notation for brevity, as in (x_t). Given any time series we can construct a new time series called a sliding window embedding, which is also known as a time delay embedding with a lag or delay time of 1.

Definition 2.3: Given a time series $\tau = (x_t)_t$ with vectors $x_t \in V$, a sliding window embedding of window length l *of* τ is a new time series, $\tau^l = (\widetilde{x_t})_t$, with

$$\widetilde{x_t} = [x_t \quad x_{t+1} \quad \ldots \quad x_{t+l-1}] \in V^l,$$

where $[\cdot]$ is concatenation of vectors.

That is, the t^{th} vector in the new time series is the concatenation of l consecutive vectors in the original time series and has dimension equal to $l \cdot \dim(V)$.

Remark 2.4: If the original time series has N points, then the sliding window embedding of window length l has $N - l + 1$ points, as one can see in Example 2.5.

Example 2.5: Consider the time series $\tau = ([1, 2], [3, 4], [5, 6], [7, 8], [9, 10])$ in $\mathbb{R}^2$. The sliding window embedding of τ of window length $l = 3$ is

$$\tau^3 = ([1, 2, 3, 4, 5, 6],\ [3, 4, 5, 6, 7, 8], [5, 6, 7, 8, 9, 10]) \subseteq \mathbb{R}^6,$$

which has $5 - 3 + 1 = 3$ points.

We applied persistent homology (**Section 2.3**) to sliding window embeddings of *C. elegans* video data in order to quantify behavior. Degree one persistent homology detected cycles in these sliding window embeddings which we show correspond to particular behaviors.

The cycles that persistent homology detects may consist of collections of points that trace out a closed curve. A cycle is "large" or highly persistent if it encloses an area that could fit a large ball; a cycle that is tall and skinny has small persistence.

Below we see two examples where a time series exhibits a single periodic behavior but persistent homology will detect either two or zero non-trivial cycles. In contrast, the persistent homology of a sliding window embedding detects exactly one non-trivial cycle in both examples.

Example 2.6: **Figure 1A** displays one period of a periodic time series in $\mathbb{R}^2$ with the property that if successive points are connected by line segments then the path of the time series self-intersects. To discover this figure-eight-shaped loop, one might try to use persistent homology (**Section 2.3**). However, persistent homology would detect two distinct loops, each comprising half of the period. See **Figure 2D** for an illustration of these loops.

Figures 1B,C show two-dimensional PCA projections of sliding window embeddings of the figure-eight for $l = 10$ and $l = 20$, respectively, using the first and third principal components. In these point clouds the time series draw out simple closed curves, and in fact in each of these cases persistence detects a single loop.

Notice that as the window length increases, the "size" of the loop increases. This increase in the loop's persistence makes it easier for persistent homology to robustly detect it.

Example 2.7: **Figure 3A** shows a 1-dimensional time series that is a discretization of a sine wave. This periodic behavior creates no loops — in fact, because the points take values in $\mathbb{R}$, the time series cannot produce degree 1 homology. However, a sliding window embedding, in this case of window length 4, creates a loop that is detected by persistent homology. That loop in $\mathbb{R}^4$ is projected down to two dimensions in **Figure 3B**.

2.3 Persistent Homology

In this section we provide an overview of persistent homology and how it may be used to produce quantitative summaries of the shape of a collection of points such as the sliding window embedding discussed above.

Definition 2.9: A simplicial complex on a set of vertices V is a collection K of non-empty subsets of V such that if $\tau \in K$ and $\tau' \subset \tau$, then $\tau' \in K$. An element $\tau \in K$ is called a simplex. An n-simplex or simplex of dimension n is a simplex $\tau \in K$ with size

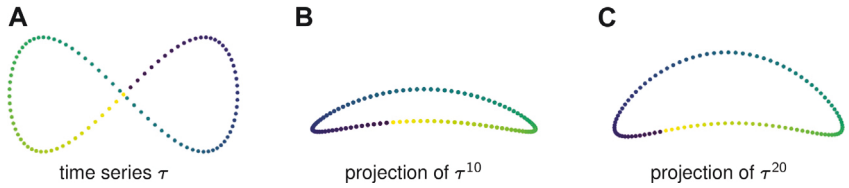

FIGURE 1 | (A) A time series in $\mathbb{R}^2$ determines a self-intersecting curve. **(B)** The sliding window embedding of window length 10 separates the previously intersecting segments of the curve. **(C)** A sliding window embedding with a higher window length separates the intersecting segments even further. With too small of a window length *l*, the resulting loop will be relatively flat and long and will therefore have small persistence and be difficult to differentiate from noise.

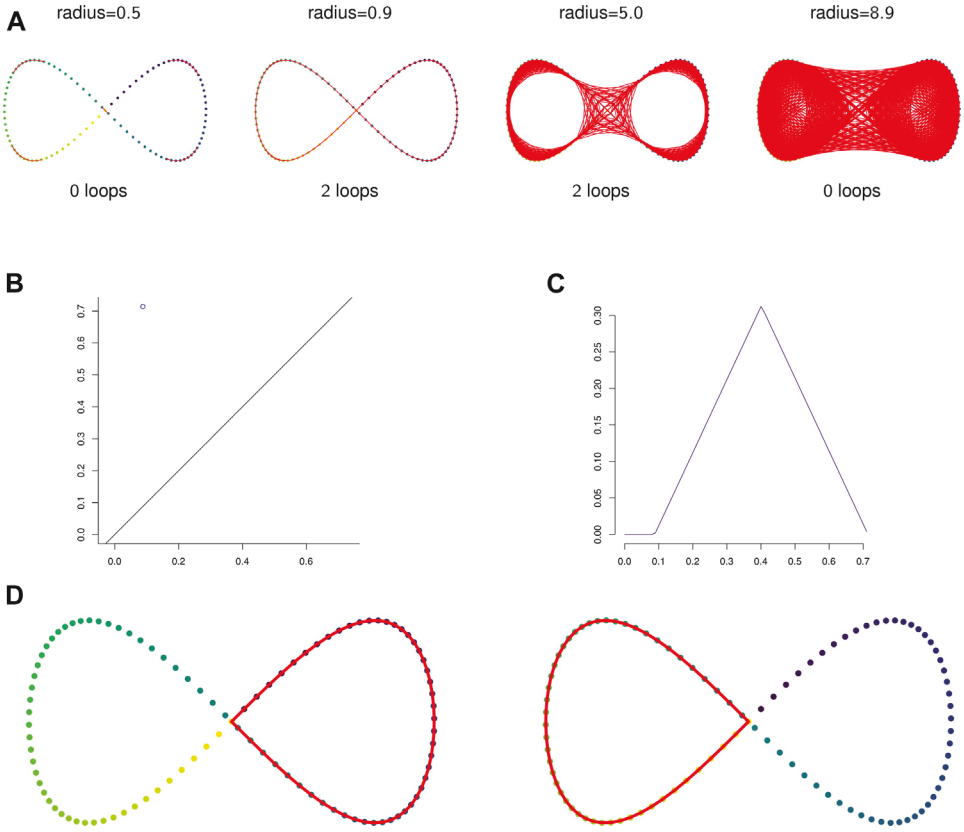

FIGURE 2 | (A) The 1-skeleton of the Vietoris-Rips filtered simplicial complex of a figure-eight-shaped point cloud at four scales. **(B)** The degree 1 persistence diagram of the figure eight in 2-dimensions. Note that the point has multiplicity 2. **(C)** The corresponding degree 1 persistence landscape. The first and second landscapes are nonzero and identical and all other landscapes are trivial. **(D)** The two loops that generate the homology of the Vietoris-Rips complex on the figure eight.

$|\tau| = n+1$. The 1-skeleton of a simplicial complex K is the set of simplices with dimension at most one. A filtered simplicial complex or filtration is a collection $\{K_r\}_{r\in S}$ of simplicial complexes K_r where $S \subseteq \mathbb{R}$ such that $K_r \subseteq K_s$ for all $r, s \in S$ with $r \leq s$.

Definition 2.10: Let $X \subset \mathbb{R}^d$ be a finite set and let $r \geq 0$. The Vietoris-Rips complex of X at scale r, denoted $\mathcal{R}_r(X)$, is the simplicial complex with vertex set X and whose simplices are given as follows. A subset $\{x_0, \ldots, x_n\} \subset X$ is an n-simplex in $\mathcal{R}_r(X)$ if and only if $|x_i - x_j| \leq r$ for all $i, j \in \{0, \ldots, n\}$.

Definition 2.11: The Vietoris-Rips filtration of a finite set $X \subset \mathbb{R}^d$ is the collection $\mathcal{R}(X) := \{\mathcal{R}_r(X)\}_{r\geq 0}$.Note that while the Vietoris-Rips complex of X is parameterized by the non-negative reals, the finiteness of X guarantees that $\mathcal{R}(X)$ consists of only finitely many distinct simplicial complexes.

Example 2.12: **Figure 2A** shows the 1-skeleton of the Vietoris-Rips complex of a pointcloud in $\mathbb{R}^2$ at four scales. Notice that each simplicial complex includes into the next.

The persistent homology of a Vietoris-Rips filtration can be represented by a multiset in $\mathbb{R}^2$ called a persistence diagram in

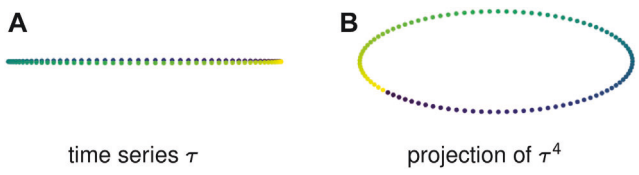

FIGURE 3 | Sliding window embeddings encode periodic behavior in the form of loops. **(A)** A periodic time series that has trivial first homology. **(B)** A sliding window embedding with $l = 4$ of the original time series that has nontrivial first homology.

which each point gives the scale of the appearance and disappearance of a topological feature (such as a loop) in the filtration.

Example 2.13: **Figures 2B,C** show the persistent homology in degree 1 of Example 2.12. Notice that both **Figure 2B**—the persistence diagram—and **Figure 2C**—the persistence landscape (see Definition 2.14)—show two cycles, but because they are born and die at exactly the same radius parameters they are plotted in the same place. The two cycles are shown in **Figure 2D**.

It is difficult to apply standard tools of statistics and machine learning directly to persistence diagrams, which, for example, need not have unique averages (Mileyko et al., 2011). A solution is to map persistence diagrams into a vector space or Hilbert space. One such mapping is the persistence landscape. See (Bubenik, 2015) for the following definitions and results.

Definition 2.14: For $a < b$ let $f_{a,b} : \mathbb{R} \to \mathbb{R}$ be the piecewise-linear function given by

$$f_{a,b}(t) = \begin{cases} t - a, \text{ if } a \le t \le \dfrac{a+b}{2} \\ b - t, \text{ if } \dfrac{a+b}{2} \le t \le b \\ 0, \text{ otherwise.} \end{cases}$$

Given a persistence diagram $\mathrm{Dgm}_p(\mathcal{K})$, the corresponding kth persistence landscape is the function $\lambda_k : \mathbb{R} \to \mathbb{R}$ given by defining $\lambda_k(t)$ to be the kth largest value of $f_{a,b}(t)$ over all points $(a, b) \in \mathrm{Dgm}_p(\mathcal{K})$. The persistence landscape is the sequence $(\lambda_k)_k$. The parameter k is called the depth of the persistence landscape. For a point cloud X, we will denote by PL (X) the persistence landscape obtained by applying degree 1 persistent homology to the Vietoris-Rips filtration of X.

Persistence landscapes have unique averages, satisfy the law of large numbers and central limit theorems, and can be discretized for computations. Because the sequence of functions that make up a landscape are nested, they can all be graphed on the same plot as in the right column of **Figure 4**.

While the persistence landscape is defined to be an object in a space of continuous functions, it can be discretized and turned into a finite-dimensional vector. Through discretization, each depth of the landscape transforms from a continuous function on $\mathbb{R}$ to a vector where the i^{th} entry in the vector corresponds to the function value at the i^{th} discrete parameter value. The vectors for each depth of the landscape are concatenated together to produce a single high-dimensional vector. These discrete landscapes can be computed directly, which we did for the computations outlined in **Section 2.4**.

This vector space (in fact, Hilbert space) setting lets us use linear algebra-based statistical and machine learning techniques such as principal component analysis (PCA). The principal components from PCA on discretized landscapes can be converted into a format much like a persistence landscape—a sequence of continuous functions on $\mathbb{R}$—but the principal components are not themselves persistence landscapes because the functions fail to be nonnegative.

2.4 Pipeline

In this section we give details for our analysis of *C. elegans* data. The input consist of piecewise linear midlines of *C. elegans* from video recordings as described in **Section 2.1**. These midlines were parameterized by the 100 angles between adjacent segments and then were transformed using PCA, so each sample input to our system was a time series τ of 100-dimensional vectors measured in radians. See Figure 1 in (Stephens et al., 2008) and the accompanied description for more details on this parameterization of the *C. elegans* midlines or see our short summary of the procedure in **Section 2.1**.

The time domain of this time series was divided into overlapping patches of a given size called the patch length, resulting in a collection $\{\tau_i\}_i$ of smaller time series. For our experiments, a patch length of 300 was chosen, with adjacent patches overlapping by half of the patch length. The sliding window embeddings of the τ_i were then computed with window length parameter $l = 20$, resulting in a new collection $\{(\tau_i)^l\}_i$ of time series of length $300 - l + 1 = 281$. This analysis is not particularly sensitive to the choices of the hyperparameters patch length and window length; only extreme changes in either parameter lead to significant changes in results. The hyperparameter choices were motivated by the timescales at which *C. elegans* complete meaningful behaviors: for patch length, 150 frames of 30 fps video is 5 s of behavior; for window lengths, 20 frames is 0.67 s and corresponds to roughly one period of forward crawling in adult *C. elegans* submerged in the 0.5% methylcellulose environment. Strategies for choosing appropriate window lengths are described in (Perea and Harer, 2015). The method for cross validation of the choice of window length is described at the end of this section.

Persistence diagrams were computed for each of the patches $(\tau_i)^l$. This step accounts for the vast majority of the computational resources of the pipeline, and the computational costs are made worse by the concatenation of vectors in a sliding window embedding. This is where we greatly benefit from the preprocessing that turns video data, which is extremely high-dimensional (see (Tralie and Perea, 2018) Section 3.1), into a 100-dimensional time series. On a 2017 15-inch MacBook Pro with a 2.8 GHz Intel Core i7 processor and 16 GB of RAM, this step took 22588.632 s, or about 6 h and 15 min.

For each $(\tau_i)^l$, a (discretized) persistence landscape PL $((\tau_i)^l)$ was computed from the persistence diagram of the Vietoris-Rips complex $\mathcal{R}((\tau_i)^l)$. The grid of parameter values on which the

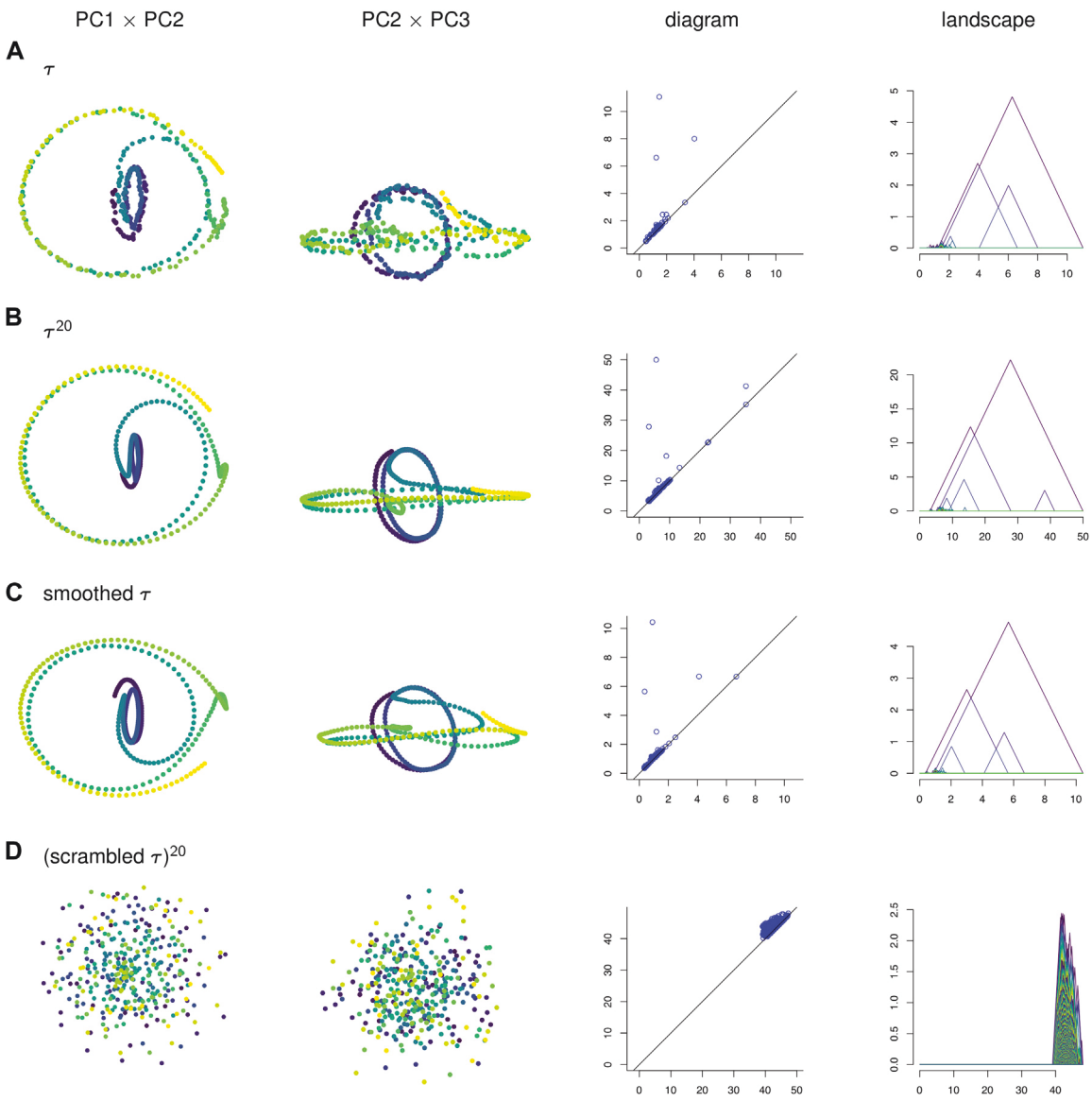

FIGURE 4 | (A) A time series τ, **(B)** its sliding window embedding of window length $l = 20$, **(C)** its smoothing by moving average filter of window length 20, and **(D)** the null model, all projected onto 2-dimensional axes of principal components. The corresponding persistence diagrams and persistence landscapes are to the right.

persistence landscape was evaluated to produce its discretization was chosen to include all of the bars of the barcode and to be sufficiently fine to produce nice visualizations. Since the persistence landscapes are piecewise linear with slope bounded by ± 1, the step size of this discretization bounds the error and there is eventually little to be gained from a finer discretization. The step size of the discretization we used was 0.1. The maximum depth was chosen to include all nonzero depths of the persistence landscapes.

The collection $\{\mathrm{PL}((\tau_i)^l)\}_i$ of persistence landscapes was then assembled into a single summary for a given video by averaging the persistence landscapes across patches to result in a single average persistence landscape associated to each video. These steps are summarized in **Figure 5**. For each environment viscosity, the average persistence landscapes for each video were averaged to give an average persistence landscape of the class.

The persistence landscapes for each class were used for analysis of the sliding window embedding as follows. Distances between each class' average persistence landscapes were computed using the usual Euclidean distance. The pairwise distances were visualized using multidimensional scaling to give a 2-dimensional visualization of the similarities between the classes; see **Figure 6A**.

Principal component analysis (PCA) was applied to the set of average persistence landscapes for each video and the first two principal components were plotted as sequences of functions. We

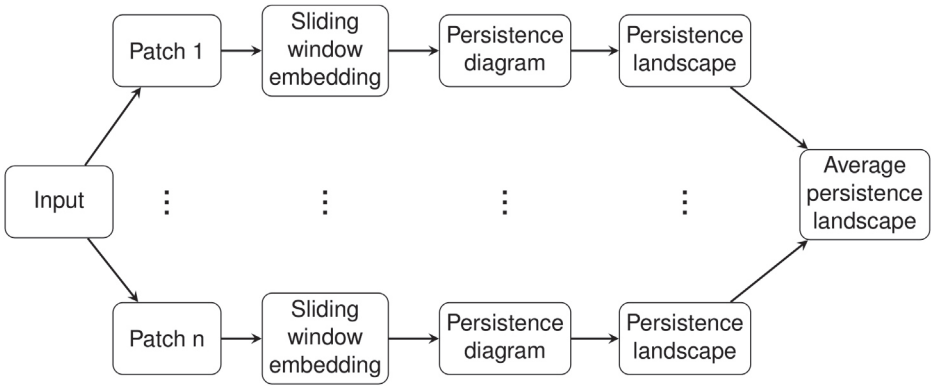

FIGURE 5 | Pipeline for a single worm.

plotted the PCA projection of these video average persistence landscapes together with the average of each class; see **Figure 7**. These plots visualize some of the similarity between classes and variation within classes.

Next, we further studied the variation within classes in two ways. First, the standard deviations of each coordinate were computed for the average persistence landscapes of the videos in each class. These were visualized as sequences of functions in **Figure 8B** to show the variation in different parts of the average persistence landscapes. Second, we applied PCA to the average persistence landscapes of the videos in each class and plotted the cumulative variances explained by the first n principal components for $n = 1, \ldots, 10$ (10 is the number of samples in each class). The first three principal components were also computed. See **Figure 9**.

We conducted a permutation test on the pairwise Euclidean distances between the average persistence landscapes of each videos. We used 10,000 permutations for each permutation test. The approximate p-value equals the percentage of cases in which the distance is at least as large as the observed distance. Results can be found in **Table 2A**.

We applied multiclass support vector machines (SVM) to classify samples according to viscosity of their environments. We use the ksvm function from the kernlab package in R on the average persistence landscapes of the videos. Accuracy was estimated using 10-fold cross validation with cost set to 10. Cross validation was repeated 20 times and the results were averaged. A confusion matrix for one instance of SVM with 10-fold cross validation was also computed and is shown in **Table 2B**.

We used support vector regression (SVR) to approximate viscosity of the worm environments given the worm's behavior data. The goal was to assess how predictive our techniques are. Accuracy was estimated by averaging 10 repetitions of 10-fold cross validation. Results are plotted in **Figure 9**.

As a final step we applied cross validation to the hyperparameter window length. We cross-validated the choice of window length by running the above pipeline for window lengths of 1, 10, 20, and 30 on a subset of the data. To reduce total computation time, we restricted to the first minute of each video, which corresponds to 1800 frames. We then compared the permutation test and multiclass SVM results to see which hyperparameter choice gave the best results. The results from cross validation of window length can be found in **Section 3.2.1**.

2.5 Validation

To help validate our pipeline, we compared our results to those obtained by applying the same computational procedure to a null model given by randomly permuting the frames in each video.

We also studied the effects of using a preprocessing step different from sliding window embeddings: moving average filters. A moving average filter of window length l of time series data creates a new time series. This time series has the same length as the corresponding sliding window embedding and each point in the time series is constructed using the same window of the original time series. The moving average filter, however, takes the average of the vectors in its window, instead of concatenating them.

Furthermore, we compared our results to the ones obtained from two simpler techniques. For the first technique we attempted to characterize *C. elegans* behavior using the speed of the worm. To do this, we computed 2-norms of the differences between two consecutive frames of angle data and then averaged all of those discrete derivatives, which resulted in a single value per sample. For the second technique, which could be described as measuring the variance of the worm's pose over a video, we computed standard deviations for the angle data coordinate-wize, which resulted in a vector of length 100 per sample. In each case we performed a permutation test and multiclass SVM on the resulting feature vectors. The results of these experiments appear in **Section 3.2.2**.

3 RESULTS

We present the results of a case study of a single sample of behavior data and an experiment on the effects of viscosity of the surrounding environment on *C. elegans* locomotion and

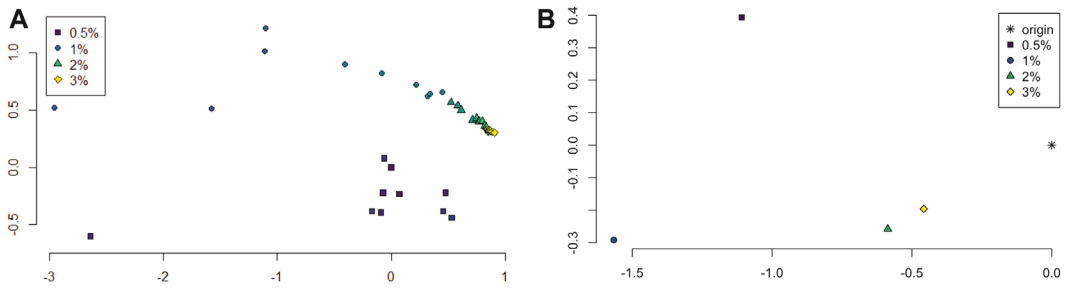

FIGURE 6 | (A) Multidimensional scaling of average persistence landscape of each sample. **(B)** Multidimensional scaling of average persistence landscapes of the classes and the origin.

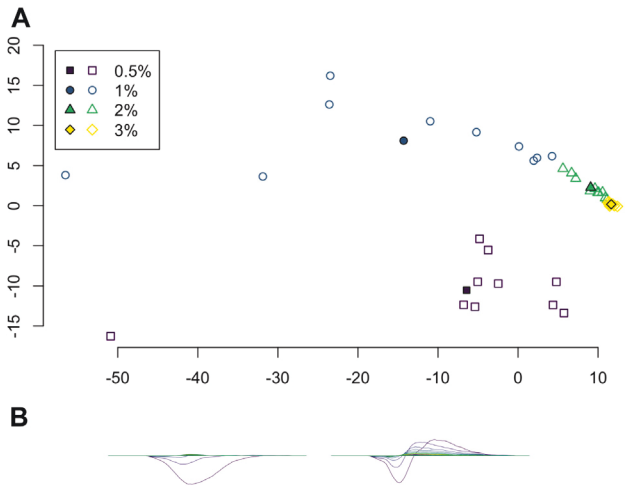

FIGURE 7 | (A) Projection of average persistence landscapes onto two principal components, with average persistence landscapes of videos given by outlined symbols and average persistence landscapes of classes given by solid symbols. **(B)** The first two principal components.

behavior. The case study assesses a significantly smaller data set and directly links topological features to specific behaviors. The experimental results in **Section 3.2** are more difficult to directly interpret in terms of specific behaviors but nonetheless we show the effectiveness of persistent homology in distinguishing variations in behaviors. We leave more explicit interpretation of average persistence landscapes in terms of specific behaviors for future work.

3.1 An Illustrative Case Study

The following results were obtained by carefully analyzing a sample of *C. elegans* behavior data from a video of a worm crawling on agar. The sample consists of 400 frames of a 30 frames per second video, so roughly 13 s of behavior. Having a solid surface to provide friction forces slower but more complicated behavior than we would see in an aqueous environment, and we take advantage of the resulting clarity of the data.

In the data we analyzed the subject exhibits the following behaviors in chronological order:

1. crawl forward,
2. crawl backward,
3. pause, and
4. crawl backward again.

Below we analyze the time series τ from this data, the corresponding sliding window embedding τ^{20}, the corresponding moving average filter, and the sliding window embedding of the corresponding null model. These comparisons illustrate various strengths of the sliding window embedding: it smooths the noise from the original time series; it retains more geometric data than the moving average filter; and it captures temporal data from the original time series that is destroyed in the null model. Then, using representative cycles we construct synthetic *C. elegans* midline data that produce a forward crawl and explain how this process gives concrete interpretations of persistence features in terms of synthetic behavior data. See **Section 3.1.1**.

To visualize the four point clouds on which we will compute persistence, we apply PCA and project onto the first few principal components. Some of these projections are shown in the two left-most columns of **Figure 4**. In contrast to the three other time series, the null model time series in **Figure 4D** has no discernible geometric structure. It appears that the corruption of temporal information has destroyed the interpretability of the visualizations of sliding window embeddings. Meanwhile, the original time series τ in **Figure 4A** has a similar shape to its sliding window embedding τ^{20} in **Figure 4B**, with the caveat that the sliding window embedding has the effect of smoothing the data and making it more robust to noise. The moving average filter in **Figure 4C** also has this smoothed property. Though the three time series in **Figures 4A–C** have similar shapes, persistence diagrams, and persistence landscapes, they vary in one important feature: the pause.

In **Figure 10A** the points in the sliding window embedding that correspond to frames where the worm is performing a specific behaviors are highlighted. The points corresponding to the pause behavior deviate from the path of points corresponding to crawling backwards. This deviation is small compared to the noisiness of the original time series, so the pause deviation does

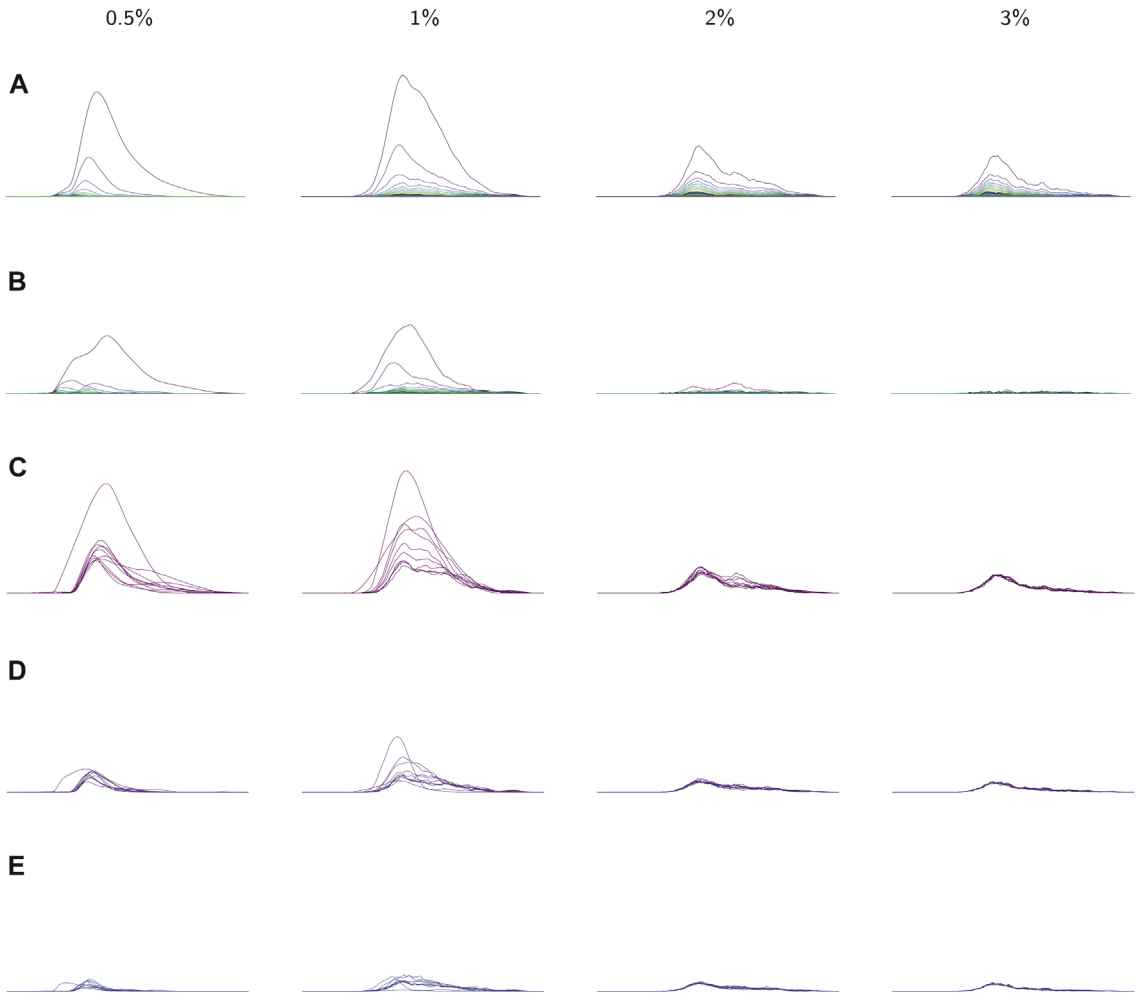

FIGURE 8 | (A) The average landscapes for each environmental viscosity, distinguished by percentage of methylcellulose present. The averages here are taken over the average persistence landscape of videos (i.e. individuals) in a given viscosity class. **(B)** Standard deviations of each coordinate in the average persistence landscapes for each class. **(C)** The first landscapes of each sample landscape, organized by class. These concurrently plotted first landscapes show the variation in the samples for each class. **(D,E)** The second and third landscapes, respectively, for each sample according to its class. All plots share the same *x*-axis; the groups of plots in **(A)**, **(B)**, and **(C–E)** each have their own *y*-axis scale.

not create a large topological feature in the graph of the original time series. This is reflected in the persistence diagrams and landscapes of **Figure 4** as well; the original time series diagram and landscape **Figure 4A** show only three significant topological features, while the diagram and landscape of the sliding window embedding **Figure 4B** and moving average filter **Figure 4C** show four.

The sliding window embedding and moving average filter both smooth the input data and detect the pause behavior, but they are qualitatively different. One piece of geometric information that the sliding window embedding retains that the moving average filter does not is the direction of the time series. Consider plotting a time series and the corresponding reverse-chronological-order time series in the same ambient space. The points of the original time series would align exactly with its reverse, so the two corresponding point clouds are the same. This is also true of a moving average filter on that time series. The sliding window embedding, however, can have distinct point clouds.

Retaining the direction of time in a time series is particularly important for data that has certain types of symmetry. A natural occurrence of such data is the sine wave data in Example 2.7. Because the data in this time series follows a path and then backtracks along that path, it never produces a loop with any significant persistence. There is no loop in the moving average filter of the data, either.

3.1.1 Interpretability: Mapping Persistence Features to (Synthetic) Behaviors

We computed representative cycles for persistent homology classes for each of the longest-persisting topological features in the sliding window embedding using Dionysus (Morozov, 2017). These are shown in **Figure 10B**. The homology classes that correspond to each of these representative

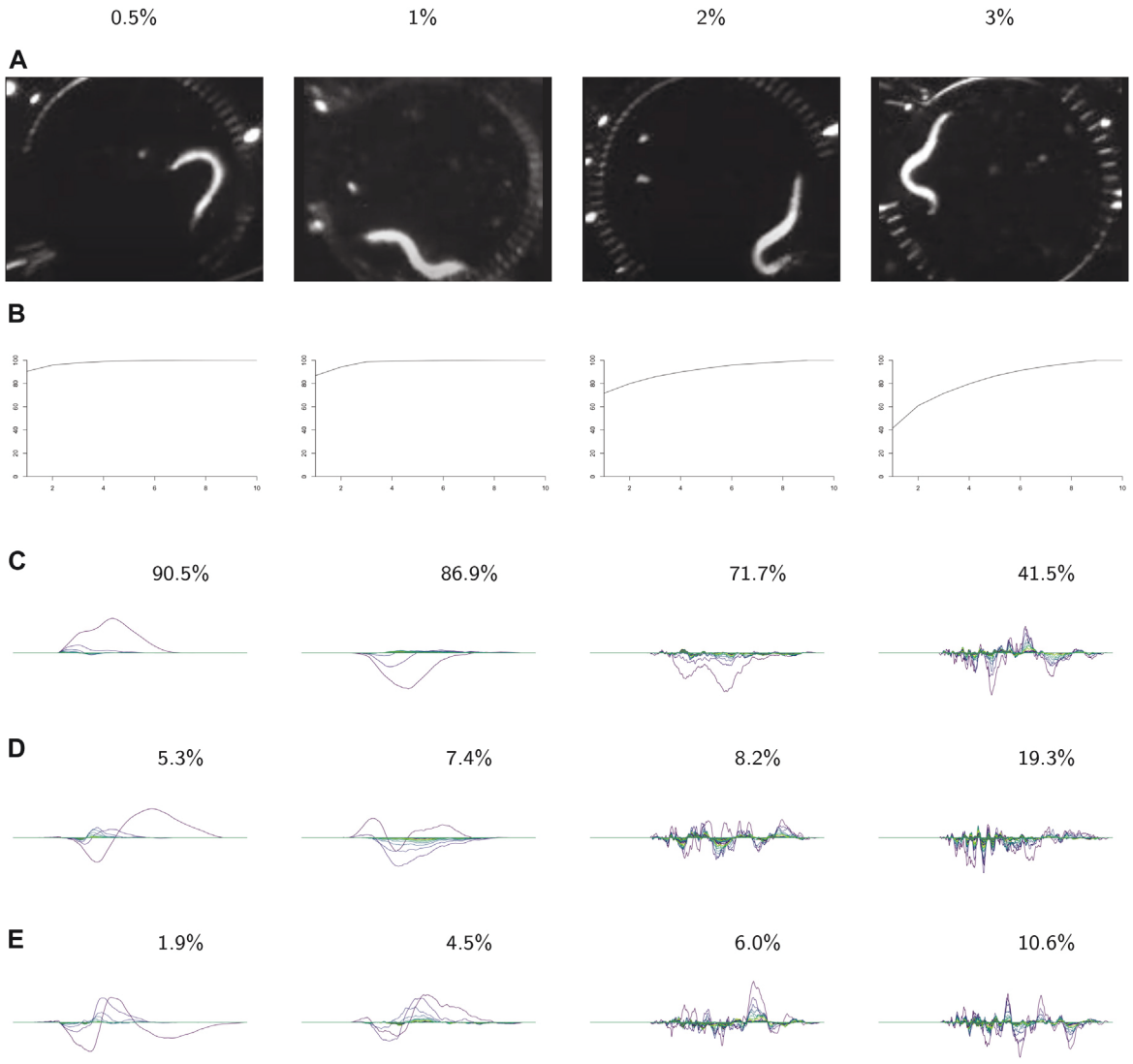

FIGURE 9 | PCA on the videos in each class. For each class: **(A)** video frames showing representative postures. **(B)** The cumulative variance of the first *n* principal components. **(C–E)** The first three principal components, labeled with the percent of the variance described by that component. These results show increasing complexity of shape and behavior as the environment becomes more viscous.

cycles are highlighted in **Figure 10C**. We remark that instead of the representative cycles produced by Dionysus one may want to use (approximate) shortest cycle representatives (Jeff Erickson, 2012; Dey et al., 2018; Obayashi, 2018; Day et al., 2019).

One of the benefits of using persistence for behavior analysis is that these representative cycles give a direct translation from persistence back into *C. elegans* behavior. Each point in the cycle corresponds to l poses and these points have a defined sequence in the cycle. The cycle lacks a direction (which way is forward in time vs which way is backward) but in many cases a direction can be inferred by subsets of the sequence that correspond to contiguous sequences in the original time series. Given all this data and a way to combine l poses into one "average" pose, we can construct synthetic, periodic behavior data from representative cycles. An example of frames from such a video is shown in **Figure 10D** and the corresponding video is available in the **Supplementary Materials**.

TABLE 1 | Normalized pairwise distances between average persistence landscapes of each class, where distance is Euclidean distance between vectors in $\mathbb{R}^{255969}$ and the normalization is such that the average distance to the origin is 1.

	0.5%	1%	2%	3%
origin	1.1873106	1.5934976	0.6777196	0.5414722
0.5%		0.8330355	0.8380992	0.8809327
1%			0.9972502	1.1255496
2%				0.1758501

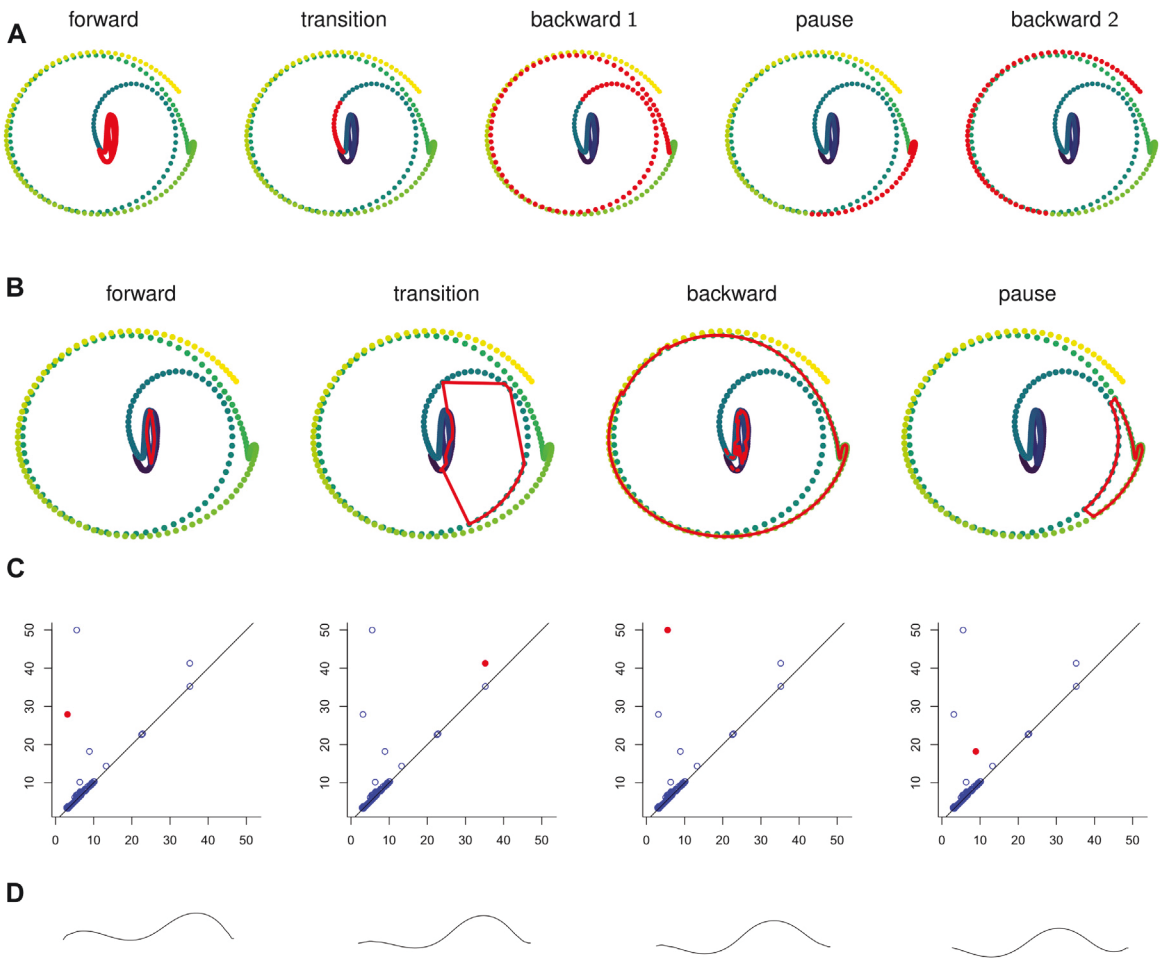

FIGURE 10 | (A) Points in the sliding window embedding τ^{20} that correspond to each of the labeled behaviors are highlighted. **(B)** The representative cycles with longest persistence from automated persistence software correspond to specific behaviors. **(C)** The persistence diagram with the homology class corresponding to the above cycle representative highlighted. **(D)** Still frames from a looping video of forward crawling data. The full video is in **Supplementary Materials**. This synthetic data was constructed from the forward representative cycle in **(B)**.

3.2 Experimental Results

We present our analysis of an experiment where *C. elegans* are submerged in solutions with varying viscosities. The viscosity of the solution is correlated with how much methylcellulose is added and experimental conditions are labeled with their methylcellulose content, usually in order from low to high methylcellulose and viscosity.

Average persistence landscapes for each class are shown in **Figure 8A**. The lower viscosity conditions allow for larger depth one landscapes (λ_1) but have relatively few non-zero higher-depth landscapes, which means there are a smaller number of larger loops detected in the sliding window embeddings. This indicates that in lower-viscosity environments, *C. elegans* exhibit behaviors of higher amplitude but either

TABLE 2 | Classification statistics. (A) Permutation test results. (B) Confusion matrix for one instance of SVM with 10-fold cross validation.

A

	1%	2%	3%
0.5%	0.0077	0.0001	0.0000
1%		0.0000	0.0000
2%			0.0000

B

	0.5%	1%	2%	3%
0.5%	10	0	0	0
1%	0	9	0	0
2%	0	1	9	0
3%	0	0	1	10

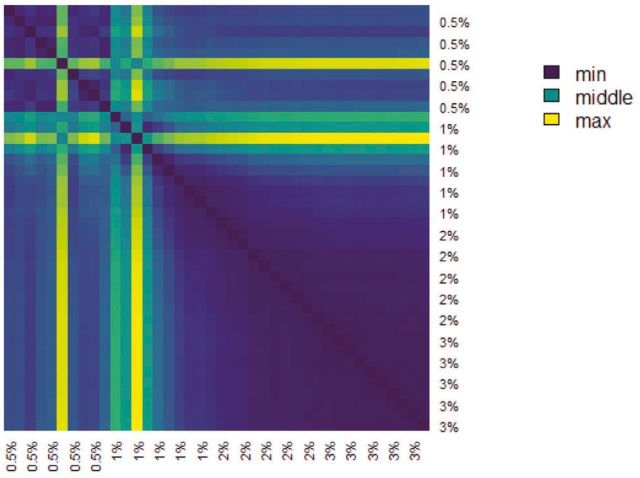

FIGURE 11 | Heatmap of the distance matrix of average persistence landscapes of samples.

demonstrate fewer distinct behaviors or have much less variation between repetitions of behaviors. Conversely, the high-viscosity classes show many more cycles in the sliding window embeddings, with each cycle being small compared to the cycles found in the low-viscosity environments. These observations suggest that at high-viscosity, behaviors do not involve large changes in posture and are more varied. From observing the raw video data, it is apparent that in higher-viscosity environments *C. elegans* can make smaller, tighter body bends, which is consistent with these results.

We also observed that as viscosity increases, the support of the persistence landscapes stretches further to the right and cycles are born at higher radius values. The worms seemed to exhibit less varied behaviors in lower-viscosity environments, so perhaps in such environments they continued "retracing their steps" through the sliding window embedding space which resulted in more densely sampled curves and thus homology classes formed at lower radii.

The pairwise distances between landscapes for each sample are visualized in **Figure 11**. The normalized pairwise distances between the average persistence landscapes for each class are shown in **Table 1**. We include the origin—the zero persistence landscape, i.e. the 0 vector—in these distance computations to complete the normalization. Normalization is such that the average distances between each class and the origin is 1. Multidimensional scaling on these distances visualizes the similarities between samples and classes, respectively, and are shown in **Figure 6**. From the raw distances and the multidimensional scaling of the distances, we can see that the high-viscosity classes (2% and 3% methylcellulose) are closest together and that this pair, the 0.5% class, and the 1% class are roughly equidistant from one another.

Principal component analysis on the average persistence landscapes for each sample gives the graphs in **Figure 7**. In **Figure 7A**, the projections of the average persistence landscapes of the samples onto the first two principal components are given by hollow symbols and projections of the average persistence landscapes of the classes are given by solid symbols. Here we see results similar to those from the multidimensional scaling in **Figure 6**: the low-viscosity class landscapes are far from each other and the high-viscosity classes, while the high-viscosity class landscapes are quite close. We can also see some of the variance within classes. The low-viscosity classes have much more variability than the high-viscosity classes, with the highest-viscosity class, 3% methylcellulose, having very little variation in these first two principal components.

These conclusions about variation in each of the classes are supported by the standard deviations of each coordinate in the average (discrete) persistence landscapes of each class. In **Figure 8B** the standard deviations of each coordinate are graphed as sequences of functions so that the standard deviations can be easily matched up with their corresponding locations on the average persistence landscapes. We conclude that there is little variation in the 3% class, slightly more in the 2% class, and much more in the 0.5% and 1% classes. The 1% class showed more variation in higher-depth landscapes than the 0.5% class, suggesting that *C. elegans* can produce slightly more complex behaviors in a slightly higher viscosity environments. The variances of the 0.5% and 1% classes also exhibit a distinct pattern; the 0.5% samples varied more toward the lower radius parameters (the left side of the graph), whereas the 1% samples varied more toward higher (more to the right) radius parameters.

In the following analysis we study the complexity of behavior expressed in each class. **Figure 9** shows results from using PCA on the videos in each class. The viscosity of the environment is negatively correlated with the percent of variance explained by the first principal component, which suggests that behaviors in low-viscosity environments are simpler than those in high-viscosity environments. Viscosity appears to be correlated with the number of nonzero landscapes, which also suggests that high-viscosity environments allow for more varied behaviors.

We conducted permutation tests between pairs of classes to determine how well average persistence landscapes can distinguish between samples from different classes. The *p*-values for these computations are shown in **Table 2A**. The permutation test gives strong evidence of statistical significance, i.e. that the topological summaries of samples from each class are significantly different.

We then used multiclass support vector machines (SVM) to build a classifier for the samples. The estimated accuracy of the classifier, computed by averaging accuracies across 20 instantiations of the multiclass SVM classifier, was 95.125%. This indicates that persistent homology is able to produce meaningful, distinguishing features from the *C. elegans* videos. A sample confusion matrix for one instance of SVM is shown in **Table 2B**. Finally, we used support vector regression to estimate the methylcellulose content in the environment for each sample. The results are plotted in **Figure 12**. There are two outliers on this graph which are estimated as having negative methylcellulose content. The two animals in these samples moved much more quickly than their peers, so we believe that the SVR is picking up on the strong negative correlation between viscosity of the

environment and speed, and based on these animals' fast speed, assigning a methylcellulose content that is so low that it is negative.

3.2.1 Cross Validation of Window Length

As was shown in Example 2.6, changes in the window length of the sliding window embedding can affect the conclusions drawn from our computational pipeline. In fact, sliding window embeddings have been critiqued for their need of this seemingly arbitrary parameter that can cause significant artifacts when dealing with volatile data (Lindquist et al., 2014). Our data is not particularly volatile since it is constrained by physical limitations of *C. elegans*, but we have still justified our choice of window length by conducting cross validation.

We assembled the results from running our computational pipeline with window lengths of 1, 10, 20, and 30. We found that different window lengths could be more useful for different tasks.

The permutation test and multiclass SVM results show that using a window length of 1 (i.e., not using a sliding window embedding) is the most predictive and that the smaller the window length, the better the accuracy. Window length 1 had the largest multiclass SVM accuracy at 95.500% compared to accuracies 91.750%, 83.375%, and 79.125% for window lengths 10, 20, and 30, respectively. The permutation test results were less definitive, with window length 1 showing slightly more separation between the 0.5% and 1% methylcellulose classes but all window lengths giving strong results. The runtimes of each computation differed significantly, with window length one data running for just over 5 min while window length 30 data took 90 min. This indicates that for predictive purposes and doing statistics on large data sets, using persistence on as close to the raw data as possible is best, but sliding window embeddings can still be useful if they are desirable for other reasons.

These statistical results contrast with the ease of visualization and interpretability of the analyses using the different window lengths. PCA projections of the cross validation data show much more differentiation between classes when the window length is 10, 20, or 30 compared to when it is 1. This is presumably because the first two principal components do not explain as much of the variation in the raw data, so separation between classes takes more principal components to describe. Meanwhile, sliding window embeddings consolidate variation in the samples into fewer principal components, so there are fewer significant principal components to visualize and interpret in terms of the original application. Essentially, sliding window embeddings give a slightly simplified but still predictive representation of the raw data.

We can also see from **Figure 4** that PCA projections of behavior data are easier to interpret when we use a sliding window embedding than when we look at just the raw time series. Recall that one of the behavioral features from the data—the pause—was not detectable over noise when looking at the PCA projection of the original time series but could be identified in both the PCA projection and the persistence landscape of the sliding window embedding. Because we were able to identify the topological feature we were also able to compute a corresponding representative cycle, which in turn allowed an estimate of the locomotion corresponding to the pause behavior to be constructed. Identification and construction of the corresponding locomotion of the pause behavior would have been more difficult using the original time series.

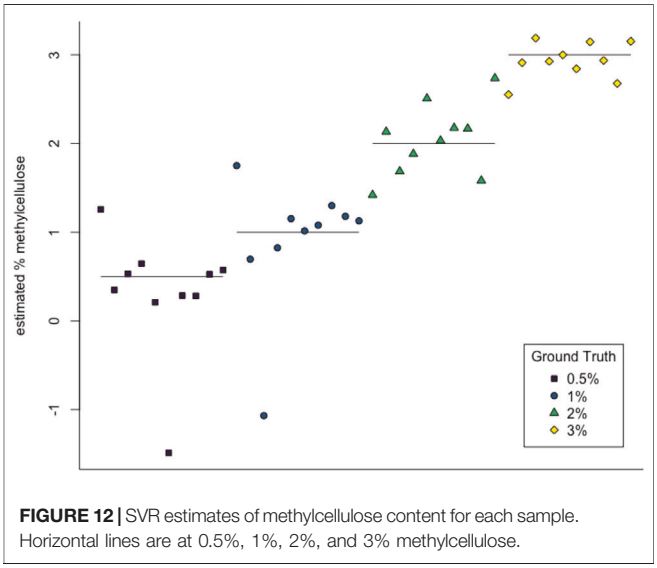

FIGURE 12 | SVR estimates of methylcellulose content for each sample. Horizontal lines are at 0.5%, 1%, 2%, and 3% methylcellulose.

3.2.2 Validation Results

We conducted permutation tests and applied multiclass SVM for data from two simple behavior quantification techniques which are described in **Section 2.5**. Results of the permutation test are listed in **Table 3**. For the method based on averages of 2-norms of consecutive frames of vector angles the cross validation error was 12.5% and training error was 7.5%. For the method based on standard deviations of vector angles the cross validation error was 70% and training error was 35%. Though more computationally taxing, persistence and sliding window embeddings produced much more accurate results than either of these simpler techniques.

Computations for the null model involve permuting the original time series of each sample and running our sliding window embedding and persistence techniques on that permuted data. Analysis of the null model showed that temporal information is necessary for our techniques to give good accuracy differentiating between samples taken in differing viscosity environments.

For the null model, the average error across 20 iterations of 10-fold cross validation on SVMs was 51.625%. The permutation tests showed that we could distinguish the 0.5% methylcellulose class without temporal information, but could not do as well differentiating between the three higher viscosity classes. It seems that the distribution of poses in the 0.5% methylcellulose class was different enough from the other classes to produce noticeably larger topological features that resulted in larger landscapes. This is probably because the lowest viscosity class had more extreme poses and thus the diameter of the space of poses for that class was significantly larger.

TABLE 3 | Permutation test results on simpler techniques which use rough measures of each worm's average movement speed and variation in posture. (A) Speed: averages of 2-norms of the difference between consecutive frames of vector angles. (B) Posture change: standard deviations of vectors angles.

A

	1%	2%	3%
0.5%	0.3497	0.0000	0.0000
1%		0.0492	0.0480
2%			0.1026

B

	1%	2%	3%
0.5%	0.0002	0.0000	0.0000
1%		0.9961	0.0625
2%			0.0411

4 DISCUSSION

We have demonstrated that persistent homology is a viable technique for studying *C. elegans* behavior and provides useful interpretations and visualizations. Our method consists of constructing sliding window embeddings of time series of piecewise linear *C. elegans* skeletons and using degree one persistent homology to create topological summaries for each patch of each video. These topological summaries, called persistence landscapes, are averaged over patches to produce a single average persistence for each video. These average persistence landscapes are our topological summary statistics and they are the statistics to which we apply further statistical analysis and machine learning techniques, such as principal component analysis, multidimensional scaling, permutation tests, and multiclass support vector machines. As far as we are aware, this is the first application of persistent homology to *C. elegans* behavior data.

Our analysis showed that persistence is able to detect variability in *C. elegans* behavior data, but also that it can provide interpretable conclusions and useful visualizations. The potential of persistence for interpretability and visualization results is demonstrated in the case study of **Section 3.1**, where topological features were connected directly to behavioral features and persistence was used to create synthetic behavior data corresponding to stereotyped behaviors such as forward crawling. Our analysis of experimental data shows that persistent homology can detect the variation of behavior induced by changes in the viscosity of the environment. It also suggests that persistence can measure complexity of behavior and that sliding window embeddings with low window lengths can be more predictive while sliding window embeddings with higher window lengths can be more useful for producing clear and interpretable visualizations, including video of synthetic data.

Persistent homology produces powerful summaries of the "shape" of data. However, using persistent homology in a way that is interpretable by experimentalists is a challenge and a topic of current research. We take a step in this direction by using representative cycles of the most persistent features of a sliding window embedding to produce synthetic videos of characteristic cyclic behaviors. At this time, there does not exist a straightforward way to similarly interpret our composite summaries, the average persistence landscapes. However, there is work in progress toward this goal (Bubenik and Wagner, 2018), and our pipeline would be able to incorporate such advances.

Our analysis has implications for future experimental design. We observed that low-viscosity environments allow for the detection of variation between samples, while high-viscosity environments may allow animals to perform more complex and varying behaviors. Tuning the viscosity of the environment for an experiment or performing experiments in multiple fluid environments with varying viscosities could allow for more easily assessing results regarding variations within populations or variations in behavior.

An extension to this experiment that could provide more validation for our techniques would be to include samples from two new environmental conditions: buffer, which would correspond to 0% methylcellulose and a lower viscosity than appears in our current data; and agar, which provides a solid surface for the worms to crawl on and surrounding air as opposed to an aqueous environment to be submerged and swim in. We would expect the new buffer class to allow for only fast, simple behaviors in line with the experiments already done, and the agar environment to allow more complex behaviors in the subjects.

The method that we have developed for applying topological data analysis to *C. elegans* locomotion data will facilitate the future study of biological phonemena such as aging. In particular, our rich quantitative summary of locomotion suggests that we may be able to measure not just lifespan, but "healthspan," the length of time an individual is healthy and physically capable. Many therapies and medicines for humans and other organisms have a goal of expanding healthspan, and therefore require a detailed measure of the ability to locomote, such as those provided by our methods.

DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: https://www.youtube.com/playlist?list=PL5pzQyEKVlEjcmBWn9IVFivLJ4nqKKWC8.

AUTHOR CONTRIBUTIONS

KB collected and preprocessed the data under the guidance of HL. AT, AE, and IH analyzed the data under the guidance of PB. AT, HL, and PB regularly discussed the direction of the project from both biological and mathematical points of view.

FUNDING

This research was partially supported by the Southeast Center for Mathematics and Biology, an NSF-Simons Research Center for Mathematics of Complex Biological Systems, under National Science Foundation Grant No. DMS-1764406 and Simons Foundation Grant No. 594594. This material is based upon work supported by, or in part by, the Army Research Laboratory and the Army Research Office under contract/grant number W911NF-18-1-0307. Research reported in this publication was supported in part by the National Institutes of Health under award numbers R01NS096581, R01NS115484, and R01AG056436. The collection of the data used in this research was partially supported by the National Institutes of Health's Ruth L. Kirschstein NRSA award 1F31GM123662.

ACKNOWLEDGMENTS

The authors would like to thank the referees whose many comments considerably improved our manuscript. AT would also like to thank Kim Le for helpful conversations.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/frai.2021.668395/full#supplementary-material

Supplementary Video 1 | Synthetic *C. elegans* behavior data of forward crawling constructed using data from **Section 3.1**. The data is constructed using a representative cycle from a persistent homology class in a sliding window embedding of real behavior data.

REFERENCES

Backholm, M., Kasper, A. K. S., Schulman, R. D., Ryu, W. S., and Dalnoki-Veress, K. (2015). "The Effects of Viscosity on the Undulatory Swimming Dynamics of *C. elegans*," in *Physics of Fluids*, 091901. 10897666. doi:10.1063/1.4931795

Berman, G. J., Bialek, W., and Shaevitz, J. W. (2016). "Predictability and Hierarchy in Drosophila Behavior," in *Proceedings of the National Academy of Sciences of the United States of America*, 11943–11948. 10916490. doi:10.1073/pnas.1607601113 url: https://pubmed.ncbi.nlm.nih.gov/27702892/.

Brown, A. E. X., Brown, E. I., Yemini, L. J., Jucikas, T., and Schafer, W. R. (2013). "A Dictionary of Behavioral Motifs Reveals Clusters of Genes Affecting *Caenorhabditis elegans* Locomotion," in *Proceedings of the Na- Tional Academy of Sciences of the United States of America*, 791–796. 00278424. doi:10.1073/pnas.1211447110 url: https://pubmed.ncbi.nlm.nih.gov/23267063/.

Bubenik, P. (2015). "Statistical Topological Data Analysis Using Persistence Landscapes," in *Journal of Machine Learning Research*, 77–102. 15337928. url: http://jmlr.org/papers/v16/bubenik15a.html.

Bubenik, P., and Wagner, A. (2018). "A Topological Heatmap," in preparation.

Chung, K., Zhan, M., Srinivasan, J., Sternberg, P. W., Gong, E., Schroeder, F. C., et al. (2011). "Microfluidic Chamber Arrays for Whole-Organism Behavior-Based Chemical Screening," in *Lab on a Chip*, 3689–3697. 14730189. doi:10.1039/c1lc20400aurl: https://pubmed.ncbi.nlm.nih.gov/21935539/.

Dequéant, M.-L., Ahnert, S., Edelsbrunner, H., Fink, T. M. A., Glynn, E. F., Hattem, G., et al. (2008). "Comparison of Pattern Detection Methods in Microarray Time Series of the Segmentation Clock," in *PLoS ONE* Editor R Khanin, e2856. 1932-6203. doi:10.1371/journal.pone.0002856

Dey, T. K., Hou, T., and Mandal, S. (2019). "Persistent 1-Cycles: Definition, Computation, and its Application," in *Computational Topology in Image Con-Text*. Editors R. Marfil, M. Calderón, F. del Río, P. Real, and A. Bandera (Cham: Springer International Publishing), 123–136. 978-3-030-10828-1.

Dey, T. K., Li, T., and Wang, Y. (2018). "Efficient Algorithms for Computing a Minimal Homology Basis," in *Lecture Notes in Computer Science (In- Cluding Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Springer-Verlag), 376–398. 9783319774039. doi:10.1007/978-3-319-77404-6_28

Firas, A. K., and Elizabeth, M. (2015). "Chatter Detection in Turning Using Persistent Homology," in *Mechanical Systems and Signal Processing*, 527–541. 10961216. doi:10.1016/j.ymssp.2015.09.046

Helm, A., Blevins, A. S., and Bassett, D. S. (2020). *The Growing Topology of the* C. elegans *Connectome*. arXiv: 2101.00065. url: http://arxiv.org/abs/2101.00065.

Husson, S. J., Wagner, S. C., Schmitt, C., and Gottschalk, A. (2012). *Keeping Track of Worm Trackers*. doi:10.1895/wormbook.1.156.1url: https://pubmed.ncbi.nlm.nih.gov/23436808/.

Jeff Erickson, J. (2012). "Combinatorial Optimization of Cycles and Bases," in *Advances in Applied and Computational Topology* (Providence, RI: Proc. Sympos. Appl. Math. Amer. Math. Soc.), 195–228. doi:10.1090/psapm/070/591

Lindquist, M. A., Xu, Y., Nebel, M. B., and Caffo, B. S. (20142014). "Evaluating Dynamic Bivariate Correlations in Resting-State fMRI: A Comparison Study and a New Approach," in *NeuroImage*, 531–546. 10959572. doi:10.1016/j.neuroimage.2014.06.052

Liu, M., Sharma, A. K., Shaevitz, J. W., and Leifer, A. M. (2018). "Temporal Processing and Context Dependency in caenorhabditis Elegans Response to Mechanosensation," in *eLife*. 2050084X. doi:10.7554/eLife.36419url: https://pubmed.ncbi.nlm.nih.gov/29943731/.

Lütgehetmann, D., Govc, D., Smith, J. P., and Levi, R. (2020). "Computing Persistent Homology of Directed Flag Complexes," in *Algorithms*, 19. 1999-4893. doi:10.3390/a13010019 url: https://www. mdpi.com/1999-4893/13/1/19.

Mileyko, Y., Mukherjee, S., and Harer, J. (2011). "Probability Measures on the Space of Persistence Diagrams," in *Inverse Problems*, 124007. 02665611. doi:10.1088/0266-5611/27/12/124007url: http://stacks.iop.org/0266-5611/27/i=12/a=124007?key=crossref. 95e8c511fc5be094303f6bde8cb2bafd.

Morozov, D. (2017). *Dionysus*. url: https ://www.mrzv.org/software/dionysus/.

Obayashi, I. (2018). "Volume-Optimal Cycle: Tightest Representative Cycle of a Generator in Persistent Homology," in *SIAM Journal on Applied Algebra and Geometry 2*, 508–534. 24706566. doi:10.1137/17M1159439url: http://www.siam.org/journals/siaga/2-4/M115943.html.

Otsu, N. (1979). "A Threshold Selection Method from gray-level Histograms," in *IEEE Transactions on Systems, Man, and Cybernetics*, 62–66. doi:10.1109/tsmc.1979.4310076

Perea, J. A., Deckard, A., Haase, S. B., and Harer, J. (2015). "SW1PerS: Sliding Windows and 1-persistence Scoring; Discovering Periodicity in Gene Expression Time Series Data," in *BMC Bioinformatics*, 257. 1471-2105. doi:10.1186/s12859-015-0645-6 url: http://bmcbioinformatics. biomedcentral.com/articles/10.1186/s12859-015-0645-6%20http://www.ncbi.nlm.nih.gov/pubmed/26277424%20 http://www.pubmedcentral. nih.gov/articlerender.fcgi?artid=PMC4537550.

Perea, J. A., and Harer, J. (2015). "Sliding Windows and Persistence: An Application of Topological Methods to Signal Analysis," in *Foundations of Computational Mathematics*, 799–838. doi:10.1007/s10208-014-9206-z url: https://link.springer.com/article/10. 1007/s10208 - 014 - 9206 - z % 20http ://link . springer . com/10 . 1007/s10208-014-9206-z.

Perea, J. A. (2016). "Persistent Homology of Toroidal Sliding Window Embeddings," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings* (Institute of Electrical and Electronics Engineers Inc.), 6435–6439. 9781479999880. doi:10.1109/ICASSP.2016.7472916

Petri, G., Scolamiero, M., Donato, I., and Vaccarino, F. (2013). "Topological Strata of Weighted Complex Networks," in *PLoS ONE*, e66506. 19326203. doi:10.1371/journal.pone.0066506 www.plosone.org.

Porto, D. A., Giblin, J., Zhao, Y., and Lu, H. (2019). "Reverse-Correlation Analysis of the Mechanosensation Circuit and Behavior in *C. elegans* Reveals Temporal and Spatial Encoding," in *Scientific Reports*, 1. 20452322. doi:10.1038/s41598-019-41349-0 url: https://pubmed.ncbi. nlm.nih.gov/30914655/.

Sizemore, A. E., Phillips-Cremins, J. E., Ghrist, R., and Bassett, D. S. (2019). "The Importance of the Whole: Topological Data Analysis for the Network Neuroscientist," in *Network Neuroscience*, 656–673. 24721751. doi:10.1162/netn_a_00073url: https://pubmed.ncbi.nlm.nih.gov/31410372/.

Stephens, G. J., Johnson-Kerner, B., Bialek, W., and Ryu, W. S. (2008). "Dimensionality and Dynamics in the Behavior of *C. elegans*," in *PLoS Computational Biology*. Editor O. Sporns, e1000028. 1553734X. doi:10.1371/journal.pcbi.1000028url: https://pubmed.ncbi.nlm.nih.gov/18389066/.

Stirman, J. N., Crane, M. M., Husson, S. J., Wabnig, S., Schultheis, C., Gottschalk, A., et al. (2011). "Real-time Multimodal Optical Control of Neurons and Muscles in Freely Behaving *Caenorhabditis elegans*," in *Nature Methods*, 153–158. 15487091. doi:10.1038/nmeth.1555url: https://pubmed.ncbi.nlm.nih.gov/21240278/.

Stolz, B. J., Harrington, H. A., and Porter, M. A. (2017). "Persistent Homology of Time-dependent Functional Networks Constructed from Coupled Time Series," in *Chaos*, 047410. 10541500. doi:10.1063/1.4978997

Swierczek, N. A., Giles, A. C., Rankin, C. H., and Kerr, R. A. (2011). "High-throughput Behavioral Analysis in *C. elegans*," in *Nature Methods*, 592–598. 15487091. doi:10.1038/nmeth.1625url: https://pubmed.ncbi.nlm.nih.gov/21642964/.

Tralie, C. J. (2016). "High Dimensional Geometry of Sliding Window Embeddings of Periodic Videos," in *Leibniz International Proceedings in Informat- Ics, LIPIcs*, 71.1–71.5. 18688969. doi:10.4230/LIPIcs.SoCG.2016.71

Tralie, C. J., and Perea, J. A. (2018). "(Quasi) Periodicity Quantification in Video Data, Using Topology," in *SIAM Journal on Imaging Sciences*, 1049–1077. 19364954. doi:10.1137/17M1150736url: https://github.com/ctralie/SlidingWindowVideoTDA.

Yemini, E., Jucikas, T., Grundy, L. J., Brown, A. E. X., and Schafer, W. R. (2013). "A Database of *Caenorhabditis elegans* Behavioral Phenotypes," in *Nature Methods*, 877–879. 15487091. doi:10.1038/nmeth.2560 url: pmc/articles/PMC3962822/%20/pmc/articles/PMC3962822/?report=abstract%20https://www.ncbi.nlm.nih. gov/pmc/articles/PMC3962822/.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.