NeuroSC: Exploring Neurodevelopment via Spatiotemporal Collation of Anatomical Networks

Noelle L. Koonce[1‡], Sarah E. Emerson[1‡], Dhananjay Bhaskar[2], Manik Kuchroo[2], Mark W. Moyle[1,11], Pura Arroyo-Morales[1], Nabor Vázquez Martínez[1], Jamie I. Emerson[3], Smita Krishnaswamy[2,4,5,6], William Mohler[7*], Daniel Colón-Ramos[1,8,9,10*]

***For correspondence:**
daniel.colon-ramos@yale.edu (DCR);
wmohler@neuron.uchc.edu (WM)

[‡]These authors contributed equally to this work

[1]Department of Neuroscience and Department of Cell Biology, Wu Tsai Institute, Yale University, New Haven, CT, USA; [2]Department of Genetics, Yale School of Medicine, New Haven, CT, USA; [3]Bilte Co. Ventura, CA, USA; [4]Department of Computer Science, Yale University, New Haven, CT, USA; [5]Program for Computational Biology and Bioinformatics, Yale University, New Haven, CT, USA; [6]Program for Applied Mathematics, Yale University, New Haven, CT, USA; [7]Department of Genetics and Genome Sciences and Center for Cell Analysis and Modeling, University of Connecticut Health Center, Farmington, CT, USA; [8]MBL Fellow, Marine Biological Laboratory, Woods Hole, MA, USA; [9]Wu Tsai Institute, Yale University, New Haven, CT, USA; [10]Instituto de Neurobiología, Recinto de Ciencias Médicas, Universidad de Puerto Rico, San Juan, Puerto Rico; [11]Department of Biology, Brigham Young University-Idaho, Rexburg, ID, USA

Abstract Volume electron microscopy (vEM) datasets such as those generated for connectome studies allow nanoscale quantifications and comparisons of the cell biological features underpinning circuit architectures. Quantifying cell biological relationships in the connectome yields rich, multidimensional datasets that benefit from data science approaches, including dimensionality reduction and integrated graphical representations of neuronal relationships. We developed NeuroSC *(also known as NeuroSCAN)* an open source online platform that bridges sophisticated graph analytics from data science approaches with the underlying cell biological features in the connectome. We analyze a series of published *C. elegans* brain neuropils and demonstrate how these integrated representations of neuronal relationships facilitate comparisons across connectomes, catalyzing new insights into the structure-function relationships of the circuits and their changes during development. NeuroSC is designed for intuitive examination and comparisons across connectomes, enabling synthesis of knowledge from high-level abstractions of neuronal relationships derived from data science techniques to the detailed identification of the cell biological features underpinning these abstractions.

Introduction

Neural circuit structure supports function. The underlying image data that yields anatomical connectomes (or wiring diagrams) are typically obtained using volume electron microscopy (vEM) tech-

niques (**Collinson et al., 2023**). Since the first complete connectome was published for *C. elegans* (**White et al., 1986**), these last decades have seen an increase in the generation of vEM datasets, as reviewed in (**Kaiser, 2023**) and others. The expansion in available anatomical connectomes has resulted from recent advancements in: 1) data generation (via automation of EM data acquisition (**Xu et al., 2017**; **Eberle and Zeidler, 2018**; **Zheng et al., 2018**; **Phelps et al., 2021**); and 2) alignment, segmentation and reconstruction (including recent implementation of AI-driven methods) as reviewed in (**Galili et al., 2022**; **Choi et al., 2024**) and others. As these developing methodologies continue to improve, they will continue to facilitate the generation of additional connectomes of whole brains and organisms.

The increasing availability of vEM datasets, including the first series of developmental connectomes published for *C. elegans* (**Witvliet et al., 2021**) has highlighted the need for new tools to enable intuitive examination and comparisons across connectomes to promote novel discoveries (**Kasthuri et al., 2015**; **Lichtman et al., 2014**; **Barabási et al., 2023**; **Xu et al., 2021**). It has also underscored the fact that vEM datasets contain a wealth of untapped information that has yet to be fully examined, represented and integrated for more comprehensive analyses (**Perez et al., 2014**; **Brittin et al., 2021**). For example, vEM datasets enable nanoscale explorations of the underlying cell biological features that govern the properties of neural circuit architectures (**Rivlin et al., 2024**; **Brittin et al., 2021**; **Moyle et al., 2021**; **Witvliet et al., 2021**; **Heinrich et al., 2021**; **Cuentas-Condori et al., 2019**). Yet most of these cell biological features (cell morphologies, contact profiles, organelle positions and shapes, etc) are not currently represented in most anatomical connectomes. Quantification of cell biological data result in high-dimensional datasets that require new approaches for their analyses and representations. The advances in vEM data generation and the resulting need for new methodologies in data science and integrated representations of neuronal relationships (e.g. from neuronal positions to neuropil structures) is akin to how advances in genetic sequencing required new methodologies in bioinformatics and new, integrated representations of genomic data (e.g. from gene sequence to gene structure) (**Swanson and Lichtman, 2016**). Addressing this gap holds the promise of integrating new knowledge from the fields of cell biology, neurodevelopment, physiology and systems neuroscience towards explaining how nervous system structure underpins its function.

Most representations of anatomical connectomes have focused on defining neuronal relationships at the level of the chemical synapse (NemaNode; WormWiring; EleganSign; FlyWire) (**Witvliet et al., 2021**; **Cook et al., 2019**; **Fenyves et al., 2020**; **Dorkenwald et al., 2023**). While the existence of chemical synapses between neuron pairs is an important feature of neuronal communication, these representations do not capture other neuroanatomical features that also underlie neuron structure and function, including contact sites from adjacent (or nearby) neurons. Recent work in *C. elegans* examined neuronal relationships by quantifying neuron-neuron contact sites to build contact profiles, or contactomes (**Brittin et al., 2021**). Examination of the contactome with data science approaches uncovered structural principles that were not evident from interrogating the synaptic connectome alone (**Moyle et al., 2021**; **Brittin et al., 2021**). These included the existence of higher-order structural motifs and the stratification of neurons (**Moyle et al., 2021**), whose hierarchical assembly during development is guided by centrally located pioneer neurons (**Rapti et al., 2017**). Moreover, integrating neuronal adjacencies (contactome) with synaptic profiles (connectome) allowed for a deeper understanding of the functional segregation of neurons within the stratified neuropil structures (**Brittin et al., 2021**; **Moyle et al., 2021**). Key to achieving this were data science approaches such as Diffusion Condensation (DC) and C-PHATE (**Brugnone et al., 2019**; **Moon et al., 2019**), which resulted in reduced dimensionality of the neuronal relationships, revealing architectural motifs across various scales of granularity, from individual neurons within circuits, to individual circuits within the neuropil. These techniques produced graphs that enabled exploration of these computationally identified groups (**Moyle et al., 2021**). DC/C-PHATE graphs are powerful tools, but they have yet to be integrated to connectomics datasets as to enable explorations of the underlying cell biological features. This limits their effectiveness for hypothesis generation and

comparative analyses across connectomes.

To address this, we generated NeuroSC, a tool for exploring neuroarchitectures across vEM datasets via novel representations of the connectome, contactome, and anatomical networks. NeuroSC is an online, open-source platform that facilitates comparisons of neuronal features and relationships across vEM data to catalyze new insights of the relationships that underpin architectural and functional motifs of the nerve ring neuropil. NeuroSC builds on recent publications in whole-brain EM datasets, integrating the latest set of developmental connectomes (***Witvliet et al., 2021***) and employing data science tools (***Brugnone et al., 2019***; ***Moon et al., 2019***) to examine neuronal relationships based on contact profiles. NeuroSC was purposefully developed with a different and complementary goal to existing tools that offer web-based visualization and access to large-scale EM datasets, such as Neuroglancer and Webknossos ***Maitin-Shepard et al.*** (***2021***); ***Boergens et al.*** (***2017***). The explicit goal of NeuroSC is to provide a platform optimized for examining *neuronal relationships* across connectomic datasets. To achieve this NeuroSC builds on the segmentations emerging from programs like NeuroGlancer and Webknossos, but with tools tailored to explore relationships such as contact profiles in the context of neuronal morphologies and synaptic positions, and across datasets that represent different animals or different developmental stages. Designed as an open-source and modular platform, NeuroSC is intended to integrate with these existing tools and datasets, supporting a synergistic approach to navigating, analyzing, and deriving meaning from complex connectomic resources.

We demonstrate how these integrated representations of neuronal relationships facilitate comparisons across these connectomes, catalyzing new insights on their structure-function and changes during development. NeuroSC achieves this by addressing three challenges in current neuronal representations: 1) accessibility of specific neuronal cell biological features (i. e. synapses and contacts), 2) integration of features for examining neuronal relationships across anatomical scales, and 3) spatiotemporal comparisons of these features across developmental datasets. These challenges were addressed by 1) creating representations of contact sites and establishing the ability to visualize subsets of synaptic sites; 2) enabling synchronous visualization of neuron morphologies, contacts and synapses and integrating these cell biological features with algorithmically-generated graphical representations of neuronal relationships; and 3) enabling simultaneous exploration of these relational representations across developmental connectomes. NeuroSC was designed as a suite of tools that facilitates future incorporation of additional datasets and representations with the goal of enabling integrated data exploration beyond the available *C. elegans* connectomes. The NeuroSC-based approaches used here for *C. elegans* could be applicable to other systems as new EM-based datasets and reconstructions become available.

Results (Comparing contactome-based relationships using C-PHATE.)

The adult hermaphrodite *C. elegans* nerve ring is a neuropil of 181 neurons of known identities, morphologies, contact profiles, and synaptic partners (***White et al., 1986***). Even for this relatively small neuropil, representations of a single feature type, such as neuronal contact profiles, constitute over 100,000 data points of multidimensional information: cell identity, region of contact, presence of synapses, etc. Analysis of this multidimensional information requires approaches that can both capture higher-order patterns of organization while enabling researchers to access the underlying cell biological features resulting in these relationships. We implemented Diffusion Condensation (DC), a clustering algorithm that iteratively groups neurons based on the quantitative similarities of their 'contact' or 'adjacency' profiles (***Brugnone et al., 2019***; ***Moyle et al., 2021***). Briefly, DC makes use of pair-wise quantifications of adjacent neuron contacts to, in a graph, move neurons with similar adjacency profiles closer together by applying a diffusion filter in a multidimensional manifold. At each iteration, the diffusion filter smooths the data across the manifold, such that local variability (or noise) in the adjacency profiles is reduced, highlighting broader, higher order pattern similarities across neurons. As iterations proceed, individual neurons (and eventually groups of neurons) are clustered together based on how close their contact profiles are to one another in

the manifold (***Brugnone et al., 2019***). In this way, DC uncovers hierarchical neuronal relationships in the contactome (***Moyle et al., 2021***).

To ensure accurate comparisons of DC across available EM datasets (***Witvliet et al., 2021***; ***White et al., 1986***), we empirically determined minimum-distance adjacency thresholds (measured in pixels; Supplementary Table 1) to construct adjacency profiles. Each neuron's adjacency profile is a quantitative measure that captures the extent of contact with neighboring neurons within its spatial vicinity. By individually setting distance thresholds for each dataset, we ensured that the degree of adjacency—defined by both the presence and extent of contact—could be accurately compared across datasets generated at different times and using diverse methodologies (see also Methods and Materials), schematized in Figure 1A-C). We applied an established adjacency algorithm to quantify the extent of contact between neuron pairs by measuring the number of shared pixels within the defined distance threshold for adjacency (***Brittin et al., 2021***). Pixel counts were summed for each neuron across EM slices within the defined nerve ring region (see Methods), resulting in an adjacency matrix representing pairwise shared pixel counts for each of the seven selected *C. elegans* contactome datasets (L1, 0 hours post hatch (hph); L1, 5hph; L2, 16hph; L3, 27hph; L4, 36hph; Adult 48hph (Figure 1C; See also Methods and Materials). These adjacency matrices were fed into DC to reveal iterative clusters of neurons with similar adjacency profiles. To visualize and compare the results from DC, we used a graphical representation of the algorithm output called C-PHATE (***Moon et al., 2019***; ***Moyle et al., 2021***), a 3-D visualization tool that builds a hierarchical, visual representation of the DC agglomeration procedure (Figure 1D-E). In C-PHATE visualizations, the DC output is mapped in 3-D space with spheres. Initially, all individual neurons in the neuropil dataset are at the periphery of the C-PHATE graph (left hand side in schematic in Figure 1D, edges of graph in Figure 1E). Neurons are iteratively condensed together based on the similarity of their adjacency profiles (schematized in Figure 1D). In the last iteration of DC, there is a single point at the center of the C-PHATE graph which represents the entire neuropil (Figure 1E, red dot). C-PHATE representations enable visualization and comparisons of contactomes across datasets, and explorations of neuronal relationship trajectories, from individual neuron interactions to circuit-circuit bundling (Figure 1F and Figure 2).

By Larval stage 1 (L1) 90% of neurons in the neuropil (161 neurons out of the 181 neurons) have grown into the nerve ring and adopted characteristic morphologies and positions. Although the organism grows approximately 5 fold from L1 to the adult, contacts in the nerve ring are also largely established by L1 and preserved during postembryonic growth (***Witvliet et al., 2021***). In agreement with this, when we used DC and C-PHATE to examine contactomes from these datasets we consistently identified four main super clusters– Stratum 1, Stratum 2, Stratum 3, and Stratum 4, using the clusters found at the highest modularity score (the iteration at which the algorithm has the highest clustering confidence) (***Newman, 2006***) (Figure 2B-F). DC outputs for each strata across animals can also be inspected using Sankey diagrams (Supplemental Tables 3-6). These diagrams detail the neuron members at each iteration of DC, allowing the user to derive quantitative comparisons of clustering events. The alignment of the neuronal morphologies of strata members reveals a persistent layered organization to the nerve ring neuropil (Figure 2 G-K), and the functional identities of the neurons in each stratum suggests that there is spatial segregation of sensory information and motor outputs ((***Moyle et al., 2021***), Supplementary Tables 3, 4, 5, 6). Our findings are consistent with previous studies on the Larval Stage 4 (L4) and adult contactomes (***Moyle et al., 2021***), and support that neurons establish core relationships during embryogenesis and maintain them during postembryonic growth, consistent with previous studies (***Witvliet et al., 2021***). Our findings also demonstrate the utility of DC and C-PHATE analyses in extracting, visualizing and comparing the structure of the neuropil architecture across contactomes.

Because DC and C-PHATE allow for the examination of relationships at varying levels of granularity, these diagrams also facilitate the interrogation of the architectural motifs that underlie distinct neural strata. A more detailed examination of clusters reveals that while the overall strata are preserved, the underlying neuronal configurations undergo changes during postembryonic

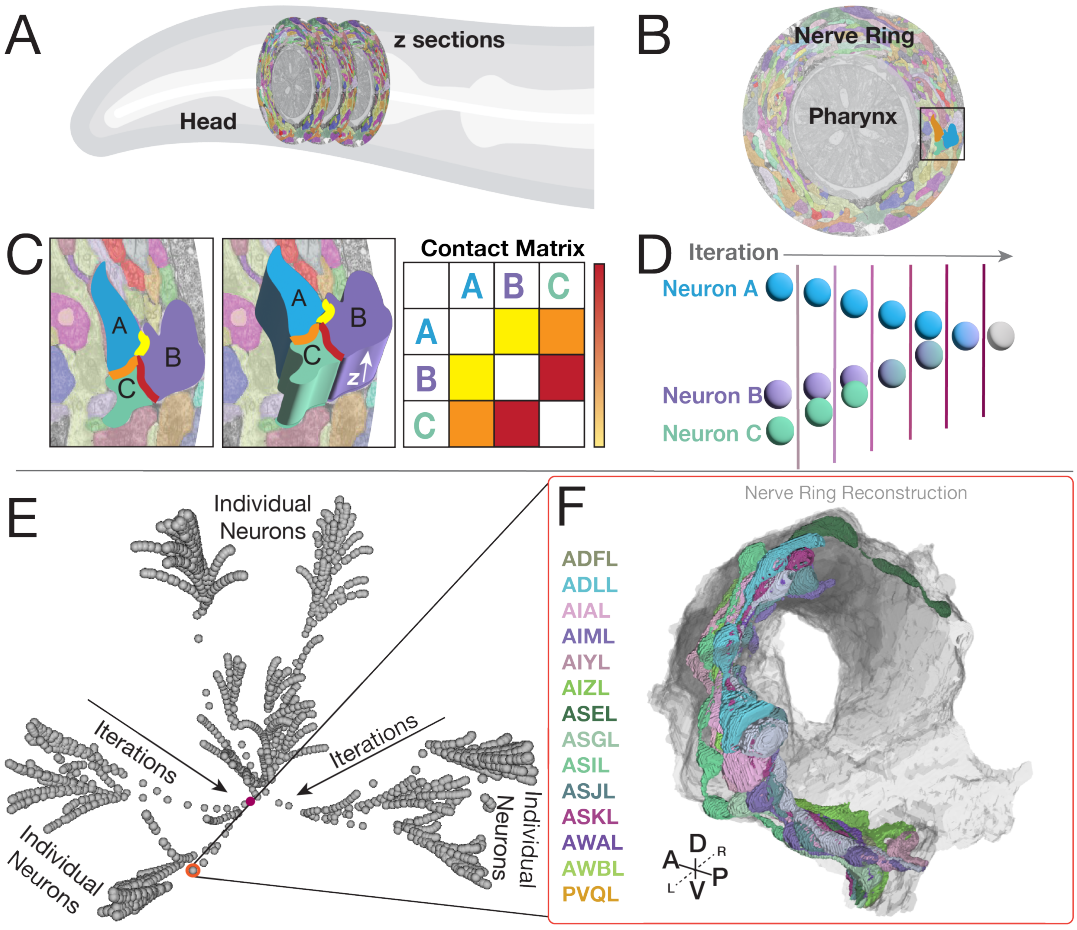


Figure 1. DC/C-PHATE representations of contactome-based relationships. DC/C PHATE graphs enable representations of neuronal contact relationships. To build DC/C-PHATE graphs we **(A)** analyzed serial section EM datasets of the *C. elegans* nerve ring neuropil (located in the head of the animal). **(B)** Single cross section of the nerve ring (surrounding the pharynx), with segmented neurites pseudo-colored. Dark box corresponds to the zoomed-in image in (C). The cross section is from the JSH dataset digitally segmented (***Brittin et al., 2021***). **(C)** Zoom-in cross section with three arbitrary neurons (called A, B, C) highlighted by overlaying opaque cartoon (2-D, left image) and 3-D shapes (middle image) to represent the segmentation process in the z-axis (arrow) and the neuronal contact sites (highlighted Yellow, orange and Red). Contacts are quantified for all neuron pairs across the contactome (See Methods), to generate a Contact Matrix (represented here as a table, schematized for the three arbitrary neurons selected and in which specific contact quantities are represented by a color scale and not numerical values). Yellow represents little contact, and red represents a large degree of contact. Here, as an example you can see that neuron B and C have the largest degree of contact. In an actual contact matrix, this would be a large number of shared pixels. **(D)** Schematic of how the Diffusion Condensation algorithm (visualized with C-PHATE) works. DC/C-PHATE makes use of the contact matrix to group neurons based on similar adjacency profiles (Brugnone et al. 2019; 2019; Moyle et al. 2021), schematized here for the three neurons in (C). **(E)** Screenshot of the 3-D C-PHATE graph from a Larval stage 1 (L1; 0 hours post hatching;) contactome, with individual neurons represented as spheres at the periphery. Neurons were iteratively clustered towards the center, with the final iteration containing the nerve ring represented as a sphere in the center of the graph (Highlighted in maroon). **(F)** Integration in NeuroSC of the DC/C-PHATE and EM-derived 3-D neuron morphology representations allow users to point to each sphere in the graph and determine cellular or cluster identities for each iteration. Shown here and circled in Red, an arbitrarily selected cluster (in E), with the identities of the neurons belonging to that cluster (four letter codes in the column to the left of F) and the corresponding neuronal morphologies (right) of this group of neurons in the EM-reconstructed nerve ring (with individual neurons pseudo-colored according to their names to the left). Compass: Anterior (A), Posterior (P), Dorsal (D), Ventral (V), Left (L), Right (R).

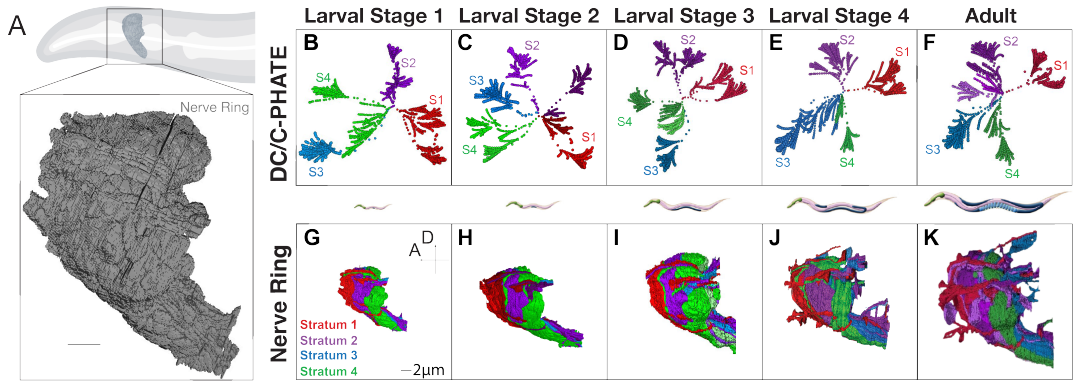


Figure 2. Implementation of DC/C-PHATE to developmental contactomes reveal a conserved layered organization maintained during post-embryonic growth. (A) Cartoon of the *C. elegans* head and nerve ring (outlined with black box). Below, nerve ring reconstruction from EM data of an L1 animal (5 hours post hatching), with all neurons in gray. Scale bar 2 μm. **(B-F)** DC/C-PHATE plots generated for available contactomes across *C. elegans* larval development, colored by stratum identity as described (***Moyle et al., 2021***). Individual neurons are located at the edges of the graph and condense centrally. The four super-clusters identified and all iterations before are colored accordingly. The identity of the individual neurons belonging to each stratum, and at each larval stage, were largely preserved, and are provided in Supplementary Tables 3-6. Some datasets contain 5 or 6 super-clusters (colored hues of the stratum that they most closely identify with). These clusters are classified as groups of neurons that are differentially categorized across the developmental connectomes. Note in B the blue cluster extends far to the left due to rotation of the 3D image. **(G-K)** Volumetric reconstruction of the *C. elegans* neuropil (from EM serial sections for the indicated larval stages (columns) with the neurons colored based on their strata identity. Scale bar 2 μm; Anterior (A) left, Dorsal (D) up.

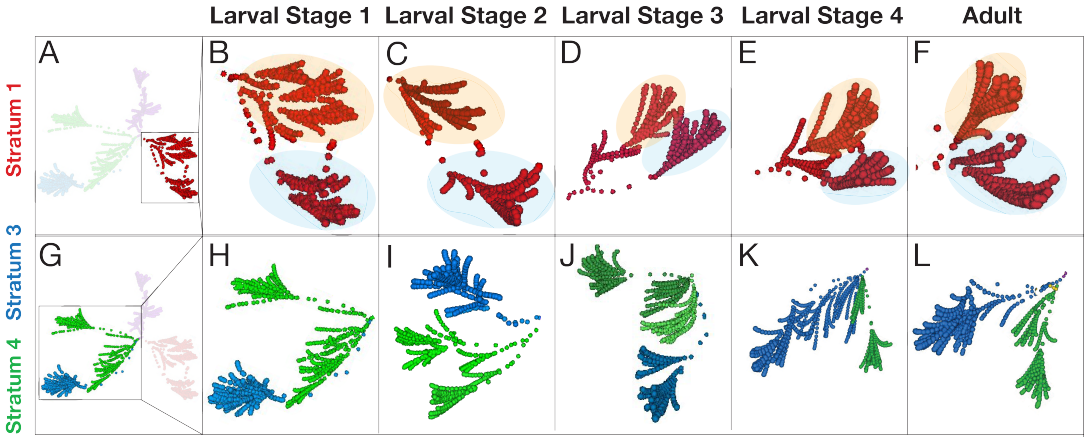


Figure 3. Examination of the architectural motifs underlying the distinct strata across development. Visualization of **(A-F)** Stratum 1 (Red) and **(G-L)** Strata 3 and 4 (Blue and Green) reveal motifs that are preserved (Stratum 1) and change (Strata 3 and 4) across developmental contactomes (L1 to Adult, left to right, as indicated by labels on top). **(B-F)** Cropped view of Stratum 1 at each developmental stage showing a similar shape of two 'horn-like' clusters in the C-PHATE graphs (as seen by orange and blue shaded areas). These two clusters have similar neuronal memberships, which are largely invariant across developmental contactomes (Supplementary Table 3). **(H-L)** Cropped view of Strata 3 and 4 at each developmental stage highlighting differences in the organization and number of neurons contained in each of the Blue and Green strata, which is particularly distinct when comparing (H) L1 and (K) L4 (Supplementary Tables 5, 6).

6).

growth (Figure 2 B-F, Figure 3, see: Supplementary Tables 3, 4, 5, 6). Three general features were extracted from these analyses: 1) individual neurons renegotiate their positions in the context of the identified C-PHATE clusters in different developmental contactomes, suggesting developmental changes; 2) the degree of these changes varied across the distinct strata; and 3) the degree of these changes mapped onto strata containing neurons with functions known to require either more or less neuronal plasticity, such as integrative behaviors versus more fixed reflexive behaviors, respectively. For example, Stratum 1, which contains most neurons contributing to shallow reflex circuits, controlling aversive head movements in response to noxious stimuli, displayed the fewest changes among the developmental connectomes (Figure 3 B-F; Supplementary Table 3). On the other hand, *C. elegans* exhibit tractable behaviors which can adapt due to changing environmental conditions ***Flavell et al.*** (***2020***). Strata 3 and 4 contain most neurons belonging to circuits associated with such learned behaviors, including chemo, mechano and thermo sensation. This is reflected by strata 3 and 4 being the regions of the most change in neuronal relationships across postembryonic development (Figure 3G-L; Supplementary Tables 5, 6).

To examine the changes in DC/C-PHATE during postembryonic development, we made the C-PHATE plots fully interactive. This enables users to hover over and identify members of each intermediate cluster, to highlight specific cell trajectories via pseudo-coloring, and compare specific neuronal relationship dynamics across development within a multi-view window of distinct C-PHATE plots (Figure 1 E-F, Supplementary Figure 6, Supplementary Video 1). Because C-PHATE graphs ultimately represent cells of known identities, we reasoned that interactive mapping of the C-PHATE cluster objects to their component cellular identities and anatomies could yield greater insights on neurodevelopmental changes by linking the algorithmic abstractions of the relationships with the cell biological features and their changes across development (Figure 4).

To evaluate our hypothesis and assess the utility of C-PHATE for discovery, we examined specific regions where the distribution or 'shape' of super clusters changed across developmental contactomes. This approach accounts for differences in contact profiles, which directly impacts the overall clustering structure. Based on these criteria, we focused on a region displaying changes in Strata 3 and 4 and using the interactive C-PHATE graphs (Figure 4 A-E), we determined the identities of neurons that changed clustering patterns across the developmental contactomes. Specifically, we focused on two interneurons, named AIML and PVQL, which we observed undergo a change in their cluster assignment from Stratum 4 (at L1) to Stratum 3 (at Larval stage 4, L4; Figure 4A and D). We pseudo-colored the trajectories of the AIML and PVQL neurons in C-PHATE to explore the changes in their C-PHATE trajectories throughout the developmental stages (Figure 4F-I, Supplementary Figure 1, Supplementary Table 7). Comparison of the identities of the neurons that co-cluster with AIML and PVQL suggests that the contact relationships varied across developmental stages (Figure 4F and G, Supplementary Figure 1), co- cluster members can also be evaluated via Sankey diagrams, showing a switch in membership of AIM and PVQ from clusters of stratum 4 at L1- L2, pre-AVF ingrowth, to include a transitionary stratum at L3, finally to stratum 3 at L4-adult after AVF ingrowth, (see Supplementary Table 7).

Visualizing contact profiles in individual cells.

DC/C-PHATE changes should result from changes in contact profiles. To link the observed changes in the C-PHATE graphs with the cell-biological changes in contact profiles, we generated a tool that would simultaneously enable: 1) 3D visualization of the cell-cell contact sites onto individual neuronal morphologies; 2) examination and comparisons of these contact profiles throughout development for the available contactomes; and 3) integration with DC/C-PHATE to link C-PHATE cluster objects to the 3-D morphologies of the algorithmically clustered cells. With these capabilities integrated, we could simultaneously view the contactome from two complementary perspectives – at an abstract systems level via DC/C-PHATE and at a cell biological level via 3D contact modeling – to perceive the architectural themes that underlie similar network patterns.

To create this tool, we generated 3D models of the area of physical contact between adjacent

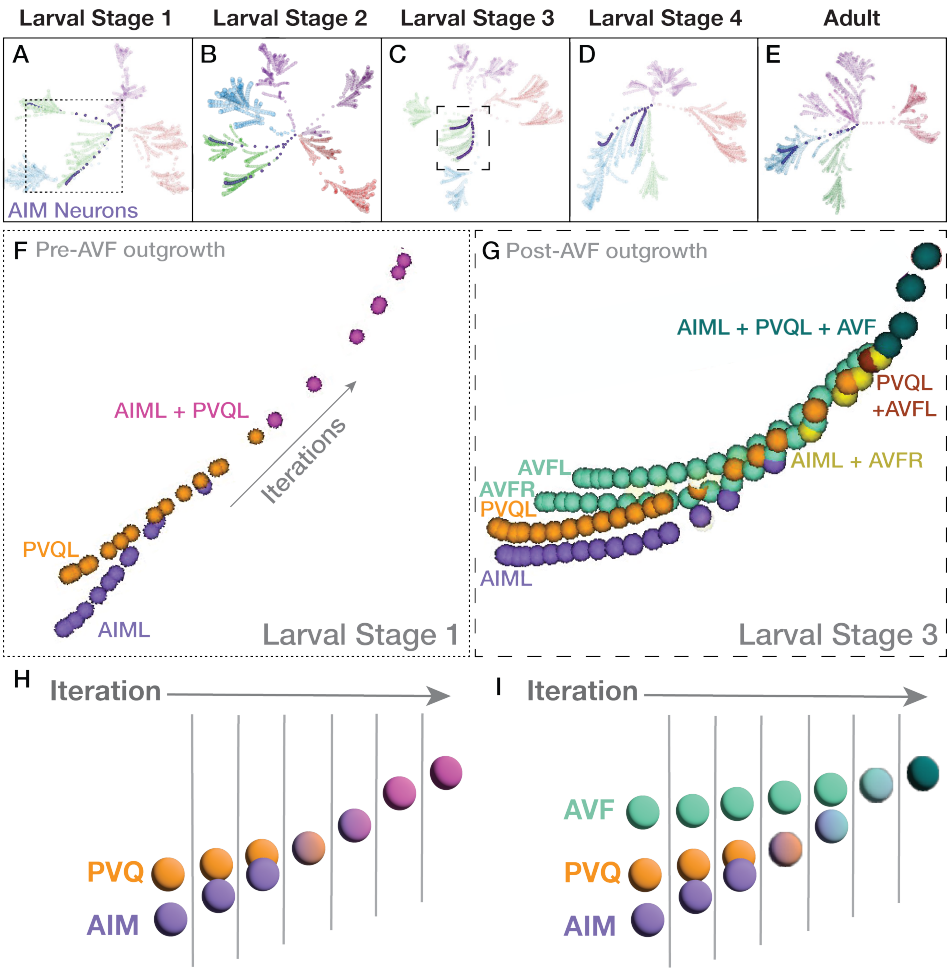


Figure 4. Case study: AIML and PVQL neurons change clustering patterns across the developmental contactomes. (A-E) C-PHATE plots across development, with the trajectories of AIM neurons (in purple) and the rest of the spheres colored by stratum identity (see Figure 2). **(F-G)** Zoom in of the AIML and PVQL trajectories corresponding to larval Stage 1 (pre-AVF ingrowth) (A, dotted box) and in (G), Larval Stage 3 with AVFL/R present (C, dashed box). Note how the relationship between AIM and PVQ neurons in the C-PHATE graph varies for each of the examined contactomes across development. (Supplementary Figure 1, Supplementary Table 7). **(H,I)** simplified schematics of F and G based on neuron class.

neuron pairs (Supplementary Tables 1, 2; Methods and Materials; Figure 5) (Supplementary Figure 2). Visualizing contacts from all adjacent neurons builds a multi-colored skeleton of the neuron morphology mapped onto the boundaries of this neuron (Figure 5A and C). Because the identities of the neurons are known and linked to the 3D contact models, we built text pop-ups that define the contact partners for each site (Figure 5C). Furthermore, since neuron names are consistent across the EM datasets, we can link and compare contact sites throughout development (Figure 5 D). Additionally, we can analyze the representations of contact sites in the context of DC/C-PHATE clustering profiles (Figure 4F-I), 3D models of neuronal morphologies (Figure 1F), and 3D models of synaptic sites for any neuron(s) across development (Figure 7).

We used the integrated tools of DC/C-PHATE and 3D representations of the contact profiles to examine the potential cell biological changes leading to the DC/C-PHATE clustering changes observed for the AIML neuron during development. With these tools, we observed changes in the identities of the contacts made in the dorsal region of the AIML neurite (Figure 5D; Supplementary Figure 3). Specifically, in the L2 stage (as compared to L1), we observed a decrease in the contacts from PVQL and an increase in contacts from the AVF neurons. This change persists to the adult

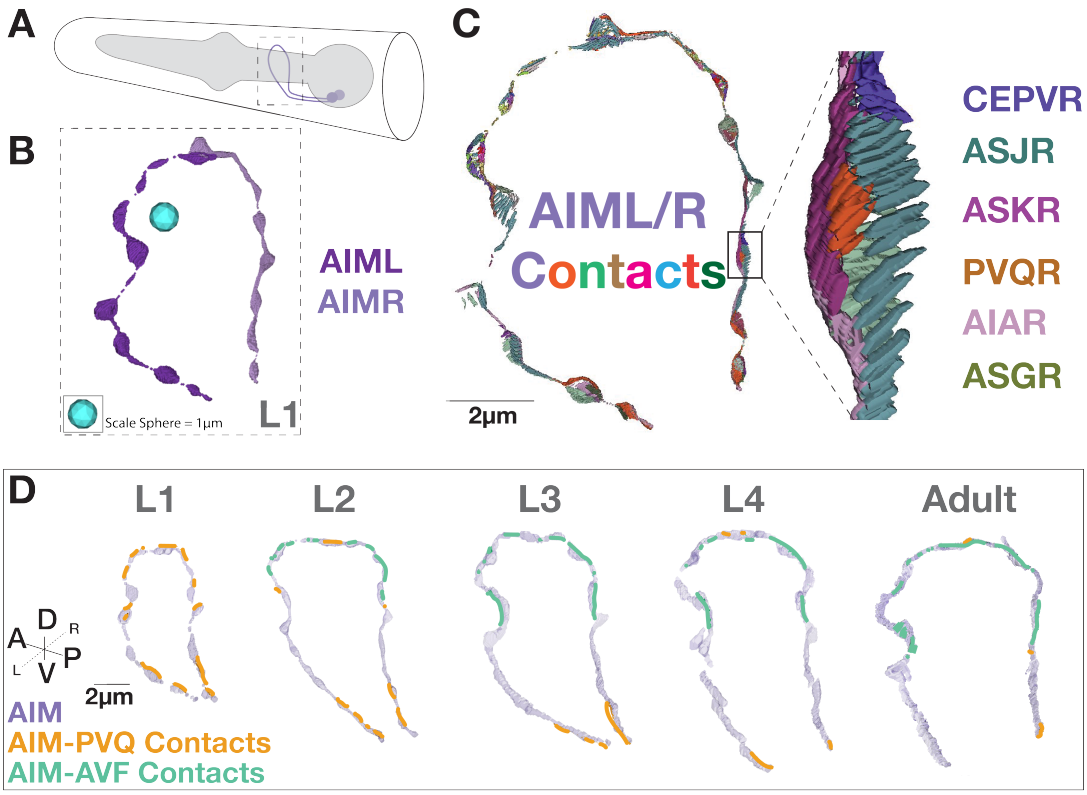


Figure 5. Case Study: Visualization of contact profiles in individual neurons. (A) Cartoon schematic of the head of the animal with the AIM neurons (purple) and pharynx (gray), and (dotted box) a 3-D reconstruction of the AIM neuron morphology from the L1 (0 hours post-hatching) dataset. **(B)** AIML and AIMR neurites rendered in 3D from L1. Note that we did not implement any surface smoothing methods to objects, so there might be gaps in the renderings. This was done intentionally, with the goal of producing the most accurate representation of the available data segmentation and avoid any rendering interpretations. **(C)** 3-D representation of all contacts onto the AIM neuron morphology in an L1 animal, colored based on contacting partner identity, as labeled (right) in the detailed inset (black box) region. **(D)** AIM-PVQ contacts (in orange) and AIM-AVF contacts (in green), projected onto the AIM neurons (light purple) across developmental stages and augmented for clarity in the figure (see non-augmented contacts in (Supplementary Figure 5). Scale bar 2 μm.

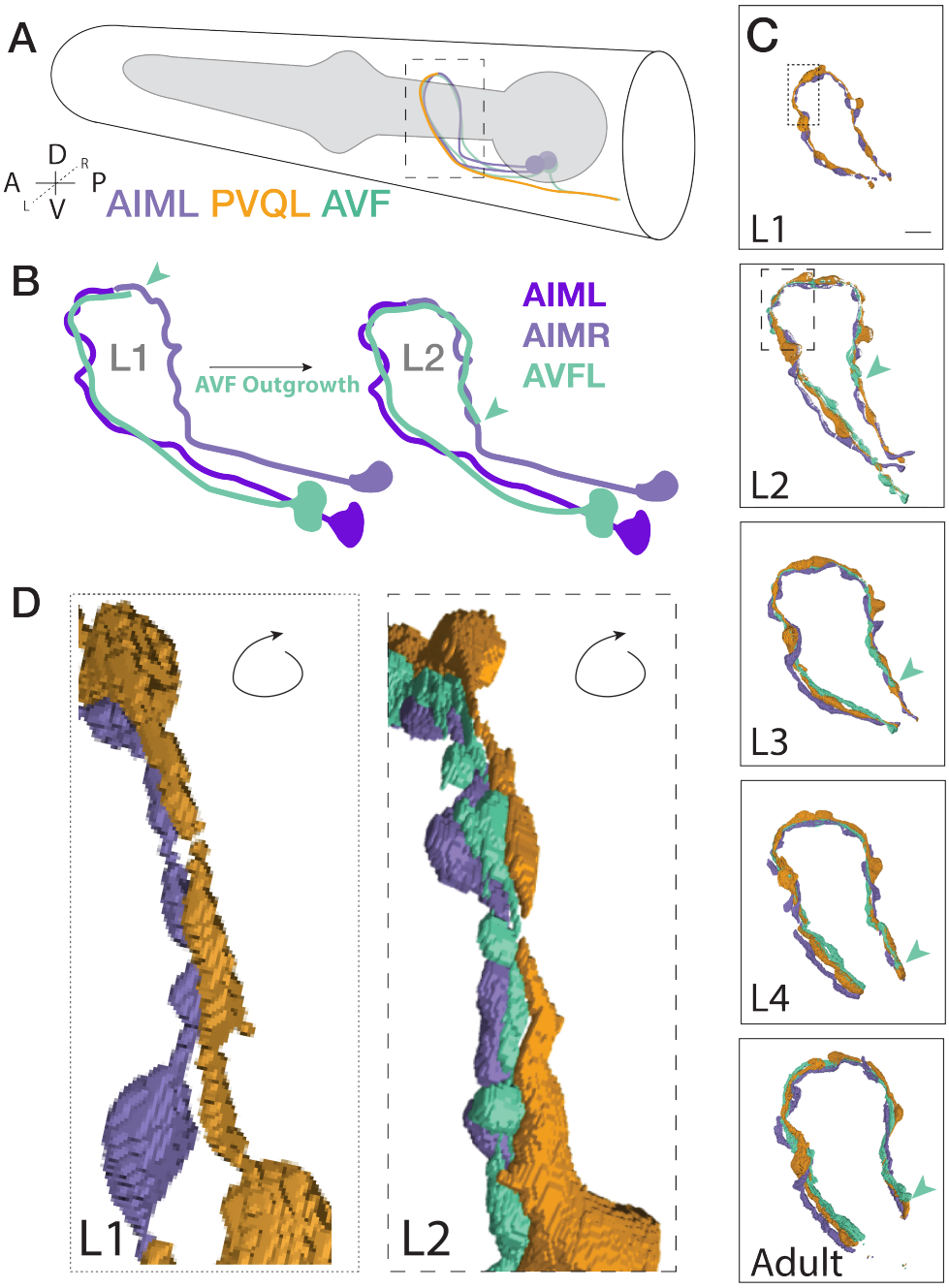


Figure 6. Case study: Segmented morphologies of AIM, PVQ and AVF across larval development. (A) Cartoon schematic of the *C. elegans* head, pharynx (gray) and examined neurons with dashed black box representing the nerve ring region. **(B)** Schematic representation of the outgrowth path of the AVF neurons as observed by EM (***Witvliet et al., 2021***). Note that in development, AVFL and AVFR both grow along AIML and extend in parallel around the nerve ring. The distal end of the AVF neurite is highlighted with a green arrowhead in the schematic. **(C)** Neuronal morphologies of AIM (purple), PVQ (orange), AVF (green) across postembryonic development, as indicated, with green arrowhead pointing to AVF outgrowth tip. Scale bar = 2 μm. Regions for insets (L1, dotted box; L2, dashed box) correspond to (D). **(D)** Morphologies of these neurons (rotated to the posterior view) display the AVF neurons' positions between the AIM and PVQ neurons at the L1 and L2 stage. Indicated outgrowth between neurons continues to the Adult stage (Supplementary Video 2). Note how AVF outgrowth alters contact between PVQ and AIM (Figure 5D).

stage (Figure 5D; Supplementary Figure 3).

To then determine the possible source of these developmental changes in contacts, we visualized 3D models of the segmented morphologies for these neurons across L1 to adulthood (Figure 6). We find that AIM and PVQ neurons maintain similar morphologies throughout development (Figure 6C), while AVF neurons undergo substantial neurite outgrowth onto new regions of contact between AIM and PVQ (Figure 6 B-D). These observations are consistent with lineaging studies which demonstrated that AVF neurons are generated from neuronal precursors (P0 and P1) at the end of the L1 stage (***Sulston et al.*** (***1983***); ***Sun and Hobert*** (***2023***); ***Poole et al.*** (***2024***); ***Hall and Altun*** (***2008***); ***Sulston and Horvitz*** (***1977***). By comparing the EM datasets across development, we observe that the AVF neurons grow into the nerve ring during the L2 stage, and continue to grow until the Adult stage (Figure 6 B-D). We also consistently observe throughout the individual datasets, that the AVF neurons grow in between the AIM and PVQ neurons (Figure 6D), altering their contact profiles, which likely contributes to the observed changes in the C-PHATE graphs. While the DC/C-PHATE representations systematically cluster neurons based on relative similarities across contact profiles, and not solely by scoring changes in specific contacts within any given pair, our findings demonstrate the use of DC/C-PHATE as a discovery tool to identify cell-biological contact changes during development. (Figure 4F and G; Figure 5 D; Supplementary Video 2). Consistent with this, we also observe that both AVFL and AVFR grow into the nerve ring alongside AIML, later continuing to grow around to reach AIMR, and that these relationships were reflected in the C-PHATE graphs in terms of the clustering profiles throughout development (Figure 4G; Supplementary Figure 1).

We then examined if the developmental changes in contact profiles result in changes in circuitry. We examined this by layering on synaptic information. Despite dwindling AIM-PVQ contacts, AIM and PVQ neurons maintained their synaptic relationship throughout development, with synaptic sites observed primarily at the base of AIM neurons, a region of persistent contact with PVQ (Figure 7A-B). We observed that increases in contacts between AIM and AVF neurons resulted in additional en passant synapses at the new points of contact, beginning at the L2 stage and continuing to adulthood (Figure 7A-B). We also observed that AVF forms synapses with the adjacent PVQ neurons (Figure 7; Supplementary Figure 4).

In summary, by integrating, representing and comparing datasets using the new C-PHATE tools and contact profiles in NeuroSC, we identified developmental changes in the relationships of AIM, AVF and PVQ. This case-study highlights the utility of combining cell biological representations (such as morphologies, contacts and synapses) with coarse-grained systems-level representations (like DC/C-PHATE) of vEM datasets to uncover developmental changes which could be further explored experimentally. Therefore, NeuroSC serves as a powerful platform for generating hypotheses for empirical testing, which can lead to insights into the dynamics of circuit development.

NeuroSC: Facilitating multi-layered interrogation of neuronal relationships in the *C. elegans* nerve ring throughout larval development

NeuroSC is built as a web-based client-server system designed to enable the sharing of anatomical connectomics data with an emphasis on facilitating the analyses of neuropil relationships across hierarchies and scales. To achieve this, we integrated tools of neuroanatomical investigation from the available *C. elegans* nerve ring connectomes and contactomes with a collection of 3-D modeled elements (morphologies, contacts and synapses and C-PHATE) representing different aspects of neuronal architecture and relationships (Figure 8). NeuroSC differs from other available web-based tools in this area with the integration of C-PHATE graphs that enable exploration of hierarchical organizations of stratified fascicles, the availability of new tools to examine the contactome, and the integration of these data with existing connectome and morphological datasets across developmental stages.

NeuroSC has eight key user-driven features: (1) C-PHATE, with the ability to highlight clusters containing neurons of interest (Supplementary Figure 6, Supplementary Video 1), (2) interactive reconstructions of neuronal morphologies (Supplementary Figure 10, Supplementary Video 3), with

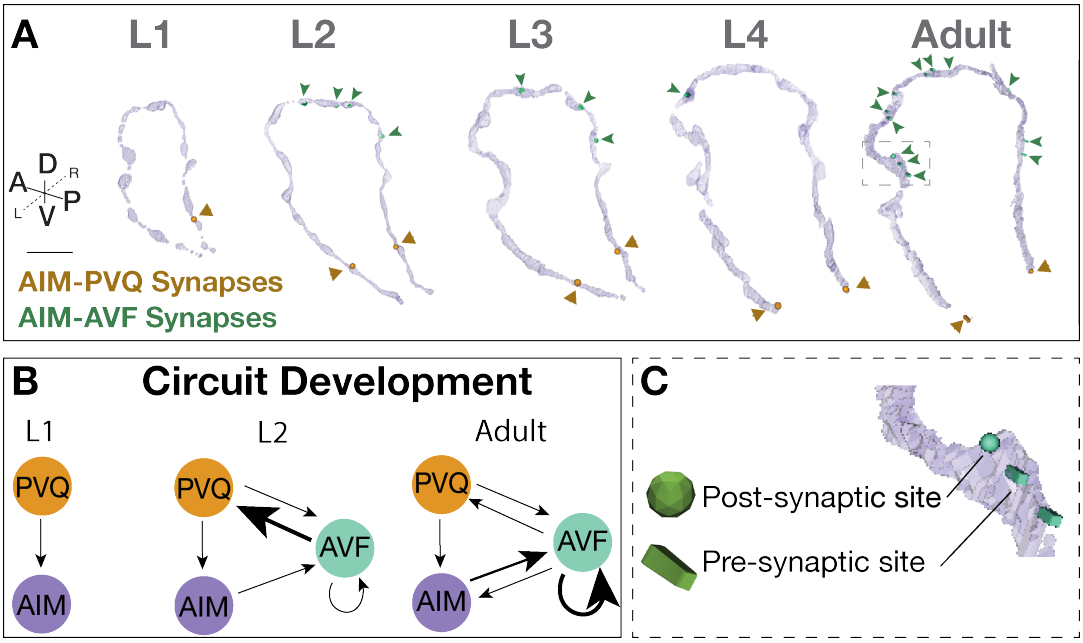


Figure 7. Case study: AIM-PVQ and AIM-AVF synaptic positions across development. (A) AIM-PVQ synaptic sites (dark orange arrowheads) and AIM-AVF synaptic sites (dark green arrowheads) in the segmented AIM neurons and reconstructed across postembryonic development from original connectomics data. Scale bar = 2 μm. **(B)** Schematic of the AIM, PVQ and AVF circuitry across development based on synaptic connectivity and focusing on the stage before AVF outgrowth (L1), during AVF outgrowth (L2) and Adult; arrow direction indicates pre to post synaptic connection, and arrow thickness indicates relative number of synaptic sites (finest, <5 synapses; medium, 5-10 synapses; thickest, 11-30 synapses). **(C)** Zoom in of synaptic sites (green) in the Adult connectome and embedded into the AIM neuron morphology (light purple). In NeuroSC, presynaptic sites are displayed as blocks and postsynaptic sites as spheres, and a scaling factor is applied to the 3-D models (Materials and Methods).

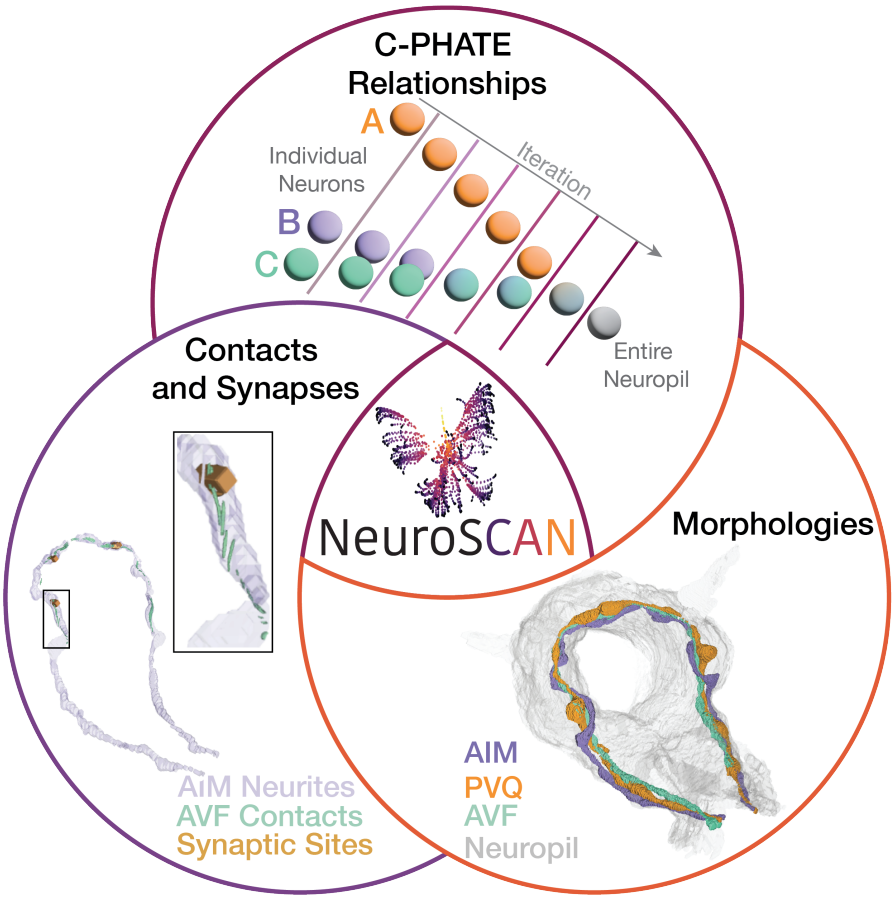


Figure 8. NeuroSC is a tool that enables integrated comparisons of neuronal relationships across development. With NeuroSC, users have integrated access to: C-PHATE plots, 3-D morphological renderings, neuronal contact sites and synaptic representations. Through stage-specific C-PHATE renderings, users can explore neuronal relationships from high dimensional contactome data. **(Top)** On C-PHATE plots, schematized here, each sphere represents an individual neuron, or a group of neurons clustered together during algorithm iterations. **(Right)** 3D renderings of select neurons can be visualized in the context of the entire nerve ring or other circuits (gray). **(Left)** AIM contact sites at L1 and the same region showing synapses. Inset shows zoomed in of contacts and synapses - presynaptic sites (blocks) and postsynaptic sites (spheres). Data depicted here are from the L1 stage (0 hours post hatching).

click-based display of cell statistics, including total volume and surface area within the defined neuropil region (see Methods), (3) reconstructions of neuronal morphologies of C-PHATE cluster members with a right-click on C-PHATE clusters (Supplementary Video 1), (4) 3-D renderings of neuronal contacts to visualize the spatial distribution of contact profiles, with click-based displays of quantitative statistics, including total contact area, rank among all contacts of the primary neuron, and surface area percentages relative to both the neuron and the nerve ring (Supplementary Figure 5, Supplementary Video 4), (5) 3-D representations of synaptic sites with the option to visualize subsets of those sites, and a click-based display showing the number of synapses of the selected identity within the primary neuron, including any polyadic synapse combinations involving the primary neurons (Supplementary Figure 7, Supplementary Video 4) (6) the ability to perform side-by-side comparisons across development by cloning a rendered scene across time into a new window at the click of a button using the in viewer developmental stage slider (Supplementary Figure 11, Supplementary Video 3), (7) the option to pseudo color each object to highlight points of interest (Supplementary Figure 11, Supplementary Video 3) and (8) each item is an individual object with the ability to be further customized by the user (Supplementary Figures 11, 12).

NeuroSC: practical considerations

We offer seven practical considerations for users. First, NeuroSC is available on mobile platforms as a quick and convenient way to look up neuron morphologies and relationships. Second, since contact sites offer the ability to explore the surrounding neurons and the position(s) of contact between adjacent neurons, NeuroSC is designed to enable studies of adjacent neurons (e.g. phenotypes that result in site-specific ectopic synapses; neuron morphology changes that may affect specific surrounding neurons; developmental events requiring communication between neurons, etc.). Third, C-PHATE can be used to identify neurons with similar contact profiles. Because contact profiles are associated with circuit identities (***Moyle et al., 2021***), exploration of neuronal relationships via C-PHATE can be used to identify new relationships between specific neurons and circuits. Fourth, visualization of subsets of synaptic and contact sites allows direct comparisons to light microscopy approaches such as cell-specific labeling of synapses or GFP-Reconstitution across synaptic partners (Feinberg et al. 2008). Fifth, because the color and transparency of each 3-D model can be customized, users can further integrate NeuroSC outputs of additional atlases (for gene expression, neurotransmitter and receptor expression, functional connectivity, etc. (***Packer et al., 2019***; ***Taylor et al., 2021***; ***Wang et al., 2023***; ***Fenyves et al., 2020***; ***Randi et al., 2023***; ***Maitin-Shepard et al., 2021***) and directly use the NeuroSC outputs to create figures and comparisons (as done for this paper). Sixth, although synaptic sites with BWM (body wall muscles) are included in NeuroSC, the current data model limits the ability to search for these non-neuronal cells. Users can search for neurons with synapses to BWM to find this datatype. Seventh, to enable direct comparisons between our data representations and the primary EM data, the original annotations have been preserved and can be accessed by users via the sister app, CytoSHOW (CytoSHOW.org). As the data continues to be curated, the modular design of NeuroSC and its companionship with CytoSHOW enables integration of future annotations.

Discussion

NeuroSC is an integrative tool for analyzing detailed, web-based representations of neuronal connectomes and contactomes throughout post-embryonic development in *C. elegans*. Connectomes and contactomes are derived from volume electron microscopy (vEM) micrographs of neuropil regions (***Witvliet et al., 2021***; ***White et al., 1986***). These EM micrographs are information-rich and have the potential to reveal architectural motifs across scales, from the nanoarchitecture of the neuron to the neuroanatomy of each circuit in the brain. Cell biological features, such as contact profiles and synaptic positions, can be rigorously quantified and systematically represented as graphs capturing multidimensional relationships. These representations require methodologies from data science that enable dimensionality reduction and comparisons of the architecture

across scales. Yet to derive new intuitions about the spatiotemporal events leading to the architecture that shapes its function, it is necessary to integrate and compare these various representations, bridging knowledge from the cell biological events to the systems-level network relationships. NeuroSC is designed to achieve this integration, enabling synthesis of knowledge ranging from the abstractions of neuronal relationships in C-PHATE to the cell biological features underpinning these abstractions. We provide a case study to illustrate how integration of analyses performed in NeuroSC can result in new insights. First, we demonstrated the discovery process with C-PHATE representations to identify neurons that undergo changes in their contactome during development. Second, we developed 3-D representations of contact sites to analyze the local neuronal regions that were identified via DC/C-PHATE analysis. Third, we visualized and compared these representations across development to identify cell biological changes in neuronal morphologies and synaptic positions across neuron classes. Our case study demonstrates the utility NeuroSC to facilitate exploration of neuronal relationships, leading to new insights on structural features of the connectome and hypotheses for empirical testing.

Comparisons of NeuroSC to other connectomics atlases

NeuroSC is one of several efforts centered around interpreting the *C. elegans* EM datasets. Other open-source tools for data exploration in *C. elegans* include efforts to capture neuron morphologies and synaptic information (including integration of new connectomes across larval development), to map neurotransmitter and receptor expression, and to record whole brain functional connectivity across genotypes (***Witvliet et al., 2021***; ***Altun, Z.F. et al., 2002***; ***Cook et al., 2019***; ***Fenyves et al., 2020***; ***Randi et al., 2023***; ***Maitin-Shepard et al., 2021***; ***Boergens et al., 2017***). NeuroSC was designed to interface and enhance these existing resources. NeuroSC focuses on comparative, biologically grounded analysis of segmented neuronal features across datasets. This includes new tools for rendering contact profiles, synchronizing 3D views, and navigating C-PHATE diagrams, all of which support multi-scale comparisons of neuronal organization. NeuroSC was designed as a series of modular, open-source tools that can be integrated onto other programs or existing resources, enhancing the existing landscape of connectomic tools. NeuroSC was inspired by tools like NemaNode and WormWiring (***Witvliet et al., 2021***; ***Cook et al., 2019***), which enable 3-D visualizations of neuronal morphologies and synaptic sites with synaptic subsets restricted to pre or postsynaptic sites. In NeuroSC we sought to generate and integrate information beyond the synaptic connectome to include local neuronal regions (contactome) and neuronal morphologies across available developmental vEM datasets. Contactomes represent features that have been largely overlooked in connectomic datasets, and which capture circuit structures not evident by inspecting solely synaptic relationships (***Brittin et al., 2018***). NeuroSC extends existing representations to also offer user-driven experience with choice over the visualization of specific synaptic sites, the option to search for synaptic partners, and the ability to customize the color of each synaptic representation (Figure 7). NeuroSC representations complement resource databases like WormAtlas, which hosts digitized electron micrographs and schematics of neuron morphologies with aggregated information on each neuron (***Altun, Z.F. et al., 2002***). Neuroglancer (***Maitin-Shepard et al., 2021***) developed by Google, is a powerful interactive tool for visualizing large-scale 3D electron microscopy (EM) and other neuro imaging datasets beyond the *C. elegans* community. Similar in functionality to Webknossos ***Boergens et al.*** (***2017***), it provides detailed views of high-resolution EM data and enables users to explore individual datasets effectively, with a strong emphasis on synaptic connectivity. NeuroSC builds upon these capabilities by offering a complementary approach focused on comparative connectomics and broader neuronal relationships. NeuroSC enables the visualization of EM datasets alongside neuron reconstructions while allowing users to compare neuron relationships across multiple carefully curated datasets. These datasets have been standardized across time points and methodologies, making it possible to track developmental changes in neural circuits with confidence. In addition to synaptic connections, NeuroSC highlights the contactome with rendered contact patches, providing a more complete picture of

spatial relationships. Users can dynamically adjust colors of rendered objects, generate scale bars for precise comparisons, and clearly visualize synapses with large geometric representations of pre- and post-synaptic structures. Importantly, neuron names and object relationships appear directly when hovering over a rendering, eliminating the need for external lookup tables. NeuroSC also supports downloading models and creating publication-ready figures, making it a valuable tool for both research and presentation. While NeuroSC does not prioritize the display of raw EM data, this information is accessible through http://NeuroSC.cytoshow.org/ (link also available on NeuroSC.net in the "About" section) for those who want direct EM inspection. By emphasizing comparability and standardized dataset integration, NeuroSC enables researchers to uncover insights into neural development and connectivity that might not be as easily accessible with single-dataset visualization tools. Together, existing tools and NeuroSC provide complementary ways to explore and analyze complex neural datasets, each offering unique strengths to the neuroscience community.

NeuroSC design and future directions

NeuroSC code and development was intentional in its design as an open-source resource that is modular and allows integration of additional features and data structures (***Cantarelli et al., 2018***). It is a hypothesis-generating tool that can be equally used by educators seeking to teach neuroanatomical principles, and researchers seeking to identify changes across connectome datasets. NeuroSC could be integrated into emerging datasets, including developmental time-courses of cell-specific transcriptomic data that would enable further insights on the molecular events underpinning neuronal development and function– from synaptogenic processes to the logic of neurotransmitter use (***Packer et al., 2019***; ***Taylor et al., 2021***; ***Fenyves et al., 2020***) and how it sustains functional connectivity (***Randi et al., 2023***). Future iterations of NeuroSC could also include positions and relationships of neurons to non-neuronal cell types, as well as the relative networks of segmented and quantified organelles within cells. NeuroSC could be used to compare new datasets from genetic variants, from animals trained under specific conditions or from additional developmental datasets across embryogenesis. As such, the pipeline and design of NeuroSC can serve as a sandbox to examine the value of the integration of datasets in exploring representations of neuronal relationships across connectomes. Currently, the addition of new datasets must be performed in-house. In the future, standardized practices for EM generation and annotation data could facilitate the implementation of a standard importer.

NeuroSC forms part of a longer tradition that has leveraged the pioneering datasets generated for *C. elegans* connectomes towards exploring structure-function relationships in the nervous system. While the smaller scale of the *C. elegans* neuropil allowed us to rigorously vet the utility of these approaches, we suggest that these same methods would be beneficial in comparative studies in neuropils of other species, including those with less stereotypically formed connectomes. We recognize that more complex organisms possess orders of magnitude more neurons, with significantly larger neuron populations per cell class. However, our previous work has demonstrated that DC/CPHATE clustering of *C. elegans* neurons consistently pulls out clusters of shared neuron classes and shared functional roles ***Moyle et al.*** (***2021***). Building on this foundation, we envision applying similar clustering approaches to larger connectomes, aiming to identify classes and functionally related neuronal groups in more complex nervous systems. We suggest that contact profiles, along with neuron morphologies and synaptic partners, can act as 'fingerprints' for individual neurons and neuron classes. These 'fingerprints' can be aligned across animals of the same species to create identities for neurons. Frameworks for systematic connectomics analysis in tractable model systems such as *C. elegans* are critical in laying a foundation for future analyses in other organisms with up to a billion-fold increase in neurons (***Toga et al., 2012***). We think of these collective efforts akin to the foundational work from *C. elegans* in pioneering genomic analysis and annotations ahead of the Human Genome Project (***Stein et al., 2001***; ***Collins and Fink, 1995***). We believe that further integration of datasets in platforms like NeuroSC would be key in determining the rep-

resentations and features necessary for the interpretation and analyses of other connectomes.

Methods and Materials

Lead Contact

Further information and requests can be directed to Daniel.colon-ramos@yale.edu.

Data Code and Availability

Figures in this article have been generated with NeuroSC (Figures 5D, Figures 6-7, Figure S2G-I, Figure S3, Figure S4, Figure S5 A-B, Figure 8, Figures S6-S12, Videos S1-S4) and CytoSHOW (Figures 1-4, Figure 5A and C, Figure S1, Figure S 5C). Data can be visualized via the viewer at NeuroSC.net or by downloading glTF files from NeuroSC and using a glTF viewer to visualize them. Additionally, the data generated for NeuroSC is available in .OBJ file format (and can be visualized from a local hard drive with CytoSHOW (http://NeuroSC.cytoshow.org/). All excel files for Diffusion Condensation iterations and adjacency quantifications can be found in Tables S3- S13. Tutorials for NeuroSC are available on NeuroSC.net upon opening the website, within the main menu of the website (Figure S8), and in the supplementary materials (Figure S5-S12; Videos S1 and S3-S4). These tutorials generally cover the process of engaging in analysis at and across specific developmental stages by filtering the data items and adding items to viewers (Figure S10). General understanding for how to use C-PHATE to analyze neuronal relationships can be found in Figure 1, Figure 4, Figure S6, Video S1, and in our previous publication (***Moyle et al., 2021***). For additional information on filters and in-viewer changes to the data (colors, developmental stages, downloading data) see Figure S5, Figure S7, Figure S11, Figure S12, and Videos S3-S4. All code for website development is available at Github (https://github.com/colonramoslab/NeuroSCAN) and for information on website architecture and data model see Figures S13-S14.

Experimental Model and Subject Details

Volume electron microscopy (vEM) data and segmentation of neurons and synapses were analyzed from (***Witvliet et al., 2021***; ***White et al., 1986***; ***Brittin et al., 2018***; ***Cook et al., 2019***). We analyzed available EM datasets that were transversely sectioned and segmented (***Witvliet et al., 2021***; ***Brittin et al., 2021***; ***White et al., 1986***). We deleted the CAN neurons in the L1-L3 datasets to keep these datasets consistent with the legacy datasets L4 and Adult (N2U), which do not contain CAN neurons (as in (***Moyle et al., 2021***)).

Method Details

All 3-D object isosurfaces (Morphologies (Neurons), Contacts, Synapses, C-PHATE plots) were generated from segmented EM datasets using a modified version of the ImageJ 3D viewer plug-in (Schmid et al. 2010) implemented in CytoSHOW (cytoshow.org). This tool employs the marching cubes algorithm for polygon-generation. All 3-D objects are first exported as wavefront (.OBJ) files then converted to GL Transmission Format (.glTF) file format which does not distort the resolution but compacts the file information to enable faster loading times in the web-based 3-D viewer. We intentionally avoided surface smoothing to renderings to preserve the details of the raw EM data.

Pixel Threshold Distance for Adjacency Profiles and Contacts

We identified two challenges in compiling Electron Microscopy (EM) datasets for comparisons: 1) how to uniformly capture neuronal relationships based on areas of physical adjacency (contact) across datasets that have differences in volume depth and in x-y-z resolutions, and 2) how to standardize across datasets in which membrane boundaries had been called using a variety of methods, including contrast methods and segmentation methods (hand-drawn vs predicted via centroid node expansion by a shallow convolutional neural network) (***Witvliet et al., 2021***; ***Brittin et al., 2018***; ***White et al., 1986***). To address this, we first standardized the region of the neuropil across all developmental stages as in (***Moyle et al., 2021***). Briefly, all cell bodies were deleted, and

we used the entry of the nerve ring neurons into the ventral cord as the posterior boundary landmark for the entire volume, focusing on the AIY Zone 2 (***Colón-Ramos et al., 2007***); slice range Table S1). Previously reported adjacency profiles used 10 pixels (or 45 nm) as the pixel threshold distance for the L4 (JSH) and Adult (N2U) datasets (***Moyle et al., 2021***). To account for differences in resolution (x-y axis) and in calling membrane boundaries between the L4 and Adult datasets and L1-L3 datasets, we designed a protocol to define the pixel threshold for each dataset. In short, for two cells that are in direct contact (Figure S2 D) in the manually segmented datasets (L4 and Adult), we calculated the length of overlap needed to reach from the segmented edge of one cell, across the membrane, and into the adjacent cell, when the segmented area of one cell is expanded by 45 nm (10 pixels). This results in an average overlap of 30 nm for directly contacting cells in the L4 dataset. Then, in each computationally segmented dataset (L1-L3), we empirically tested the distance (e.g. 55 nm, 60 nm, 62 nm) required to achieve a similar overlap of 30 nm in direct contact cells. That empirical number (in nm) was used for adjacency calculations and rendering of contacts. The numbers were converted from nanometers into pixels to create a pixel threshold distance for each dataset, and these are shown in Table S1. Once these corrections had been applied, we calculated the cell-to-cell adjacency scores for all cell pairs in each dataset by using the measure_adjacency algorithm from https://github.com/cabrittin/volumetric_analysis; (***Brittin et al., 2018***) (Tables S8-S13). Adjacency matrices were used for Diffusion condensation (***Brugnone et al., 2019***). The contact surface areas, shown in the contact stats boxes, each represent the sum of the lengths of each all contact outlines for a given cell pair throughout the EM volume, multiplied by the reported z axis spacing between the slices, giving units of nm^2 .

Diffusion Condensation

Diffusion condensation (DC) is a dynamic, time-inhomogeneous process designed to create a sequence of multiscale data representations by condensing information over time (***Brugnone et al., 2019***). The primary objective of this technique is to capture and encode meaningful abstractions from high-dimensional data, facilitating tasks such as manifold learning, denoising, clustering, and visualization. The underlying principle of diffusion condensation is to iteratively apply diffusion operators that adapt to the evolving data representation, effectively summarizing the data at multiple scales. The diffusion condensation process begins with the initialization of an initial data representation, typically the raw high-dimensional data or a preprocessed version. This initial representation is used to construct a diffusion operator, a matrix derived from a similarity matrix that reflects the local geometry of the data. The similarity metric, such as Euclidean distance or cosine similarity, plays a crucial role in defining these local relationships. For contactome datasets, distances between neurons are determined by the pixel overlap between their segmented shapes in the EM dataset. We use these distances to build a graph with weighted edges, in which the weight of the edge represents the pixel overlap (the adjacency in the actual EM segmentation). Affinities between neurons, which are a proxy for their distance in the graph, are then computed as now revised in Box 1, Algorithm 1. This process is done iteratively as neurons cluster. Once the initial diffusion operator is established, the algorithm proceeds to the diffusion step. In this step, the diffusion operator is applied to the data, smoothing it by spreading information along the edges of the similarity graph. This operation captures the intrinsic geometry of the data while reducing noise. The specific form of the diffusion operator, such as the heat kernel or graph Laplacian, significantly impacts how information is propagated during this step. Following the diffusion step, the condensation step updates the data representation by aggregating diffused data points if the distance between them falls below a 'merge threshold'. This step creates a more compact and abstract representation of the data. These diffusion and condensation steps are iteratively repeated. At each iteration, the diffusion operator is recomputed based on the updated diffuse data representation, ensuring that the process adapts to the evolving structure of the data. The iterations continue until a stopping criterion is met, such as convergence of the data representation to a single point. The output of the diffusion condensation process is a sequence of multiscale data

representations. Each representation in this sequence captures the data at a different level of abstraction, with earlier representations preserving more detailed information and later representations providing more condensed summaries. This sequence of representations can be utilized for various tasks, including manifold learning, denoising, clustering, and visualization. By iteratively smoothing and condensing the data, diffusion condensation reveals the underlying structure of high-dimensional datasets. The threshold (epsilon) used to merge data points in each iteration is set as a small fraction of the spatial extent of the data: for each coordinate dimension (x, y, z), we compute the range (maximum minus minimum), take the maximum of these three values, and divide it by 10,000. This process is performed iteratively for each round of clustering until all data points cluster into a single point. During diffusion Condensation (DC) we track the modularity of the resulting clusters at each iteration and select the iteration with the highest modularity to define the clusters that represent the strata ***Moyle et al.*** (***2021***); ***Brugnone et al.*** (***2019***). Mathematically, modularity is calculated by comparing the actual number of edges within clusters to the expected number of such edges in a randomized network with the same degree distribution (Newman,2006). A higher modularity value implies that nodes within the same cluster are more densely connected to each other than to nodes in other clusters. A detailed algorithm description is provided in Box 1 and Algorithm 1.

Diffusion Condensation

Initialization:

Let $\mathbf{X} = \{x_1, x_2, \ldots, x_n\}$ be the set of n data points in a high-dimensional space. Construct the affinity matrix $\mathbf{A}$, where A_{ij} measures the similarity between x_i and x_j. Typically,

$$A_{ij} = \exp\left(-\frac{d(x_i, x_j)^2}{2\sigma^2}\right)$$

for a chosen scale parameter σ and distance metric d. In the case of the contactome dataset, the data points represent segmented neurons, and affinities are computed using the number of shared pixels shared by pairs of neurons at their boundary.

Diffusion Operator:

Define the degree matrix $\mathbf{D}$ as a diagonal matrix where $D_{ii} = \sum_j A_{ij}$. Construct the diffusion operator

$$\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}$$

which normalizes the affinity matrix.

Diffusion Step:

Apply the diffusion operator to the data:

$$\mathbf{Y} = \mathbf{P}\mathbf{X}$$

This step smooths the data, capturing the intrinsic geometry.

Condensation Step:

After each diffusion step, merge data points that are within a small distance, ϵ, from each other to form a condensed representation. Specifically, data points y_i and y_j are merged if

$$\|y_i - y_j\| < \epsilon.$$

The threshold ϵ is computed as a small fraction of the coordinate-wise (element-wise) maximum pairwise distance among all pairs of points after the first diffusion step:

$$\epsilon = \frac{1}{10,000} \cdot \max_{i,j} \|y_i - y_j\|_\infty$$

where $\|\cdot\|_\infty$ denotes the element-wise (coordinate-wise) maximum norm.

Let $\pi : \{1, 2, \ldots, n\} \to \{1, 2, \ldots, k\}$ be a mapping that assigns each data point index i to a cluster index $\pi(i)$ after condensation. The condensed cluster centers are then given by

$$\mathbf{C} = \{c_1, c_2, \ldots, c_k\}, \quad \text{where} \quad c_j = \frac{1}{|\pi^{-1}(j)|} \sum_{i \in \pi^{-1}(j)} y_i.$$

Modularity:

To evaluate the community structure after condensation, compute the modularity score M

$$M = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{d_i d_j}{2m}\right) \delta(\pi(i), \pi(j)),$$

where:

- $d_i = \sum_j A_{ij}$ is the degree of node i,
- $m = \frac{1}{2} \sum_{i,j} A_{ij}$ is the total edge weight in the graph, and
- $\delta(\pi(i), \pi(j)) = 1$ if $\pi(i) = \pi(j)$ (i.e., x_i and x_j belong to the same cluster), and 0 otherwise.

Iteration:

Repeat the diffusion and condensation steps, adjusting the parameter σ adaptively and keeping track of the modularity score, until all points are merged. Output iteration with highest modularity score.

Box 1: Mathematical description of Diffusion Condensation

Algorithm 1 Diffusion Condensation

Input: Data matrix $\mathbf{X} = \{x_1, x_2, \ldots, x_n\} \in \mathbb{R}^{n \times d}$, number of iterations T, scale parameter σ, condensation threshold ϵ
Output: Condensed data matrix $\mathbf{X}_{\text{condensed}}$
Initialize: Construct affinity matrix $\mathbf{A}$, degree matrix $\mathbf{D}$, and diffusion operator $\mathbf{P}$
$\mathbf{A}_{ij} \leftarrow \exp\left(-\frac{d(x_i,x_j)^2}{2\sigma^2}\right)$ for all i, j
$\mathbf{D} \leftarrow \text{diag}\left(\sum_j \mathbf{A}_{ij}\right)$
$\mathbf{P} \leftarrow \mathbf{D}^{-1}\mathbf{A}$
for iteration $= 1$ to T **do**
  Diffusion Step:
    $\mathbf{Y} \leftarrow \mathbf{PX}$
  Condensation Step:
    Merge data points x_i and x_j if $d(x_i, x_j) < \epsilon$ to form condensed cluster centers $\mathbf{C} = \{c_1, c_2, \ldots, c_k\}$
    $\mathbf{X} \leftarrow \mathbf{C}$
  Update:
    $\mathbf{A}_{ij} \leftarrow \exp\left(-\frac{d(x_i,x_j)^2}{2\sigma^2}\right)$ for all i, j
    $\mathbf{D} \leftarrow \text{diag}\left(\sum_j \mathbf{A}_{ij}\right)$
    $\mathbf{P} \leftarrow \mathbf{D}^{-1}\mathbf{A}$
end for
Return: $\mathbf{X}_{\text{condensed}} \leftarrow \mathbf{X}$

C-PHATE

C-PHATE is an extension of the PHATE technique (***Moon et al., 2019***) which is specifically aimed at handling and visualizing high-dimensional biological data. C-PHATE is specifically designed to handle compositional data, which are datasets where the components represent parts of a whole and are inherently constrained. It learns the intrinsic manifold of the data, effectively capturing non-linear relationships and structures that are not apparent with traditional methods like PCA or t-SNE. The C-PHATE algorithm starts by loading affinity matrices associated with specific clustering obtained from diffusion condensation. These matrices are normalized to generate kernel matrices that emphasize the strength of connections within each cluster. The algorithm then builds a connectivity matrix by integrating these kernel matrices based on cluster assignments over multiple time points. This is achieved by first initializing the matrix with kernel matrices along its diagonal and then filling in off-diagonal blocks with transition probabilities that reflect how clusters transition from one time point to the next. Next, we apply the PHATE dimensionality reduction technique to the connectivity matrix to generate 3D embeddings of the data. These embeddings are derived from multiple iterations of diffusion condensation, capturing the geometry of the data at various levels of granularity. The resulting coordinates are saved for subsequent analysis. The final step involves visualizing the PHATE results in a 3D graphics tool, CytoSHOW (Java-based; CytoSHOW.org; https://github.com/mohler/CytoSHOW; (***Moyle et al., 2021***)). The results are plotted in a 3D environment, with functionality enabling rollover labels to display information about clustered cells. This requires cross-referencing output tables from the original data collection. CytoSHOW is an interactive tool that allows for assigning colors and annotations to individual neurons and clusters of interest. A detailed algorithm description is provided in Box 2 and Algorithm 2. The python code for C-PHATE allows for user specification of four numerical parameters within the command line, and we used the same set of values for all C-PHATE plots shown in this report (100, 30, 50, 1). The first two integers define the weighting of connectivity between the current condensation step t

and previous steps t-1 (weighting = 100) or t-2 (30), respectively, during construction of the connectivity matrix. Values 100 and 30 consistently resulted in a series of plotted clustering trajectories that form a dome-like convergence of paths, enhancing our visual perception of relative relationships and showcasing the super clusters that constitute anatomical strata in the nerve ring neuropil (Video S1). The reproducibility of the dome shape depends on assigning two specific PHATE parameters (https://phate.readthedocs.io/en/stable/api.html) to non-default values when calling PHATE, the "t" value is set to 50; the "randomstate" value is set to 1.

C-PHATE

Given n data points, $\mathbf{X} = \{x_1, x_2, \ldots, x_n\}$, and the diffusion condensation output, consisting of $\mathbf{C}_t = \{c_1, c_2, \ldots, c_k\}$ denoting the merged data points and $\mathbf{A}_t$ denoting the affinity matrix at iteration t.

Kernel Matrix: For each iteration, t, compute the degree matrix $\mathbf{D}$, where $D_{ii} = \sum_j A_{ij}$. Then, normalize the affinity matrix to construct the kernel matrix $\mathbf{K}_t$:

$$\mathbf{K}_t = \mathbf{D}^{-1/2}\mathbf{A}_t\mathbf{D}^{-1/2}$$

Initial Connectivity Matrix: Initialize the connectivity matrix $\mathbf{C}_{\text{PHATE}}$ with zeros. Next, populate it with the kernel matrices, $\mathbf{K}_t$, along its diagonal, reflecting self-connections within each cluster at each time point.

Update Transition Probabilities: For each pair of adjacent time points t and $t+1$, compute a transition probability matrix to determine how points transition between clusters C_t and C_{t+1}. Each entry p_{ij} in this matrix represents the probability of moving from cluster i at time t to cluster j at time $t+1$. p_{ij} is calculated by counting the number of points moving from cluster i to cluster j and normalizing by the total number of points in cluster i at time t. This can be expressed as:

$$p_{ij} = \frac{\text{Number of points moving from } i \text{ to } j}{\text{Total number of points in cluster } i \text{ at time } t}$$

Use these transition probabilities to populate the off-diagonal blocks of $\mathbf{C}_{\text{PHATE}}$

Dimensionality Reduction: Apply the PHATE algorithm to the final connectivity matrix $\mathbf{C}_{\text{PHATE}}$ to obtain the low-dimensional embedding $\mathbf{Y}$:

$$\mathbf{Y} = \text{PHATE}(\mathbf{C}_{\text{PHATE}})$$

Visualization: Visualize low-dimensional embedding $\mathbf{Y}$ in CytoSHOW.

Box 2: Mathematical description of C-PHATE

Algorithm 2 C-PHATE

Input: Output of the Diffusion Condensation algorithm, number of iterations T
Output: Low-dimensional embedding $\mathbf{Y}$
Initialize: Load affinity matrices and cluster assignments from diffusion condensation output
for $t = 1$ to T **do**
  Load affinity matrix $\mathbf{A}_t$ from file
  Compute degree matrix $\mathbf{D}_t$ where $D_{t_{ii}} = \sum_j A_{t_{ij}}$
  Normalize to get kernel matrix $\mathbf{K}_t = \mathbf{D}_t^{-1/2}\mathbf{A}_t\mathbf{D}_t^{-1/2}$
end for
$\mathbf{C}_{\text{PHATE}} \leftarrow$ zero matrix with kernel matrices $\mathbf{K}_t$ along the diagonal
for $t = 1$ to $T - 1$ **do**
  Compute transition probabilities matrix $\mathbf{P}_{t,t+1}$ for clusters from time t to $t + 1$
    $\mathbf{P}_{t,t+1}[i, j] \leftarrow \frac{\text{Number of points moving from cluster } i \text{ to cluster } j}{\text{Total number of points in cluster } i \text{ at time } t}$
  Update off-diagonal blocks of $\mathbf{C}_{\text{PHATE}}$ based on $\mathbf{P}_{t,t+1}$
end for
Compute the PHATE embedding $\mathbf{Y}$ from $\mathbf{C}_{\text{PHATE}}$
Return: $\mathbf{Y}$

Electron Microscopy based 3-D Models

To make 3-D models of neuron morphologies from vEM datasets, we created Image-J format regions of interest (ROIs) using published segmentation data (***Witvliet et al., 2021***; ***White et al., 1986***; ***Brittin et al., 2021***). For a given cell, the stack of all sectioned ROIs was then used to draw binary image masks as input to a customized version of the marching cubes algorithm (***Schmid et al., 2010***) to build and save a 3-D isosurface. All steps of this pipeline were executed within the ImageJ-based Java program, CytoSHOW (***AU Duncan et al., 2019***). Slightly modified versions of this workflow were also followed for: 1) generating cell-to-cell contact ROIs and 2) for generating 3-D representations of synaptic objects. To align the 3-D models from the variously oriented vEM datasets, all surfaces from a given specimen were rotated and resized to fit a consensus orientation and scale. This was achieved by applying a rotation matrix multiplication and scaling factor to all vertex coordinates in isosurfaces comprising each modeled dataset (Table S2). Each 3-D object (morphology, contact or synapse) was then exported as a Wavefront file (.OBJ) and then web-optimized by conversion to a Draco-compressed .GLTF file. Each neuron was assigned a type-specific color that is consistent across all datasets to enable facile visual comparison. All the original EM annotations that were used to create the representative 3D models in NeuroSC have been preserved, and can be accessed via the sister app, CytoSHOW (https://github.com/mohler/CytoSHOW; (***AU Duncan et al., 2019***)).

Morphologies

Neuron morphologies were linked across datasets for users to visualize changes over time. To enhance 3-D graphics performance without sacrificing gross morphologies we employed a defined amount of data reduction when building each cell-morphology object. NeuroSC can therefore display multiple (or even all) neurons of a specimen within a single viewer. The number of vertices for a given object was decreased by reducing 10-fold the pixel resolution of the stacked 2-D masks input into the marching cubes algorithm of CytoSHOW.

Nerve Ring

To make a simplified mesh of the overall nerve ring shape, individual neuron ROIs were fused together into a single nerve-ring-scale-stack of image masks. This was used for input to the marching

cubes algorithm. The union of all overlapping enlarged neurite ROIs in a vEM section was data reduced (20-fold reduced pixel resolution). This rendered a performance-friendly outer shell of the nerve ring.

Contacts

To build 3-D representations of neuron-neuron contacts, we captured the degree of overlap when an adjacent cell outline was expanded by the specimen-specific, empirically-defined pixel threshold distance listed in Table S1 (see Figure S2). This was done for each cell outline. This expansion step employs a custom-written method in CytoSHOW that increases the scale of the adjacent outlined region by the pixel threshold distance (Table S1; Figure S2 B and E), while maintaining its congruent shape. The entire collection of captured 2-D contact overlaps (Figure S2 C and F) for each adjacent neuron pair was then reconstructed as a single 3-D object (Figure S2 H). Contact patches shown in NeuroSC are largely reciprocal (e.g. if there is a AIML contact from PVQL then there will be a PVQL contact from AIML), but rarely, 2-D overlap regions may be too small to be reliably converted to 3-D isosurfaces by the marching cubes algorithm, resulting in absence of an expected reciprocal contact model within the collection. Contacts, like cell morphology models, are named to be automatically linked across time-point datasets and to facilitate user-driven visualization of changes over time.

Synapses

Synaptic positions were derived from the original datasets and segmentations, which annotate synaptic sites in the EM cross-sections (***White et al., 1986***; ***Cook et al., 2019***; ***Witvliet et al., 2021***). To represent these coordinates in the 3-D segmented neurons, we used Blocks (presynaptic sites), Spheres (postsynaptic sites) and Stars (electrical synapses). The synaptic 3-D objects were placed at the annotated coordinates (***White et al., 1986***; ***Cook et al., 2019***; ***Witvliet et al., 2021***). Additionally, the objects were scaled with the scaling factor (Table S2). Synaptic objects were named by using standard nomenclature across all datasets, as explained in Supplementary Figure 7.

We note that the L4 and Adult datasets and the L1-L3 datasets were prepared and annotated by different groups (***White et al., 1986***; ***Cook et al., 2019***; ***Witvliet et al., 2021***). Integration of these datasets reveals nanoscale disagreements in the alignment of the boundaries and synapses. Our representations reflect the original annotations by the authors. Because of these disagreements in annotations, the synapses are not linked across datasets. However, all the original EM annotations that were used to create the representative 3D models in NeuroSC, including the synaptic annotations, have been preserved, and can be accessed by the users via the sister app, CytoSHOW (CytoSHOW.org).

Website Architecture

The NeuroSC website architecture and data structure were designed to integrate these key user-driven features via a modular platform and linked datasets. The architecture uses Geppetto, an open-source platform designed for neuroscience applications, modularity, and large datasets (***Cantarelli et al., 2018***). Briefly, the architecture is effectively separated into two applications, a front end React/JavaScript bundle that is delivered to the client, rendering the neuron data and assets, and a Golang application that exposes a JSON API, serving the neuron data and assets based on user interactions (Supplementary Figure 13). The backend uses a Postgres Database to store underlying data (Supplementary Figure 14), a Persistent Storage Volume that houses and serves static assets, and a variable number of Virtual Machines to run the front end and backend application code, scaling as needed to accommodate traffic. The User Interface is a React application that allows users to filter, sort, and search through the Neurons so that they can be added to an interactive canvas (Supplementary Figure 13). When users add Neurons to a viewer, a .gltf file is loaded in for a given model (Synapses, Neurons, Contacts) at the selected developmental stage (Supplementary Figure 13), which can then be manipulated in the 3D environment or layered with other meshes as

needed. NeuroSC can be used on common web-browsers (e.g. Google Chrome, Safari) and mobile devices.

The underlying data model makes use of tables representing Synapses, Neurons, Contacts and Developmental Stages. Relationships between these models are represented by foreign keys (Supplementary Figure 14). Source data is defined in a file-tree structure containing various assets (such as .gltf files representing various entities), as well as CSV's which store relationships across entities. The directory structure outlines a vertical hierarchy, starting at the developmental stages, then branching downwards onto neuron and synapse data. A Python script is invoked to traverse the directory tree and parse the files, writing to the database accordingly. This configuration enables: 1) verification of the ingested data and 2) quick search times through the datasets to identify related items. Code is version-controlled in GitHub (https://github.com/colonramoslab/NeuroSCAN) and deployed through a CI/CD pipeline when updates are committed to the main branch (Supplementary Figure 13).

Acknowledgments

We are grateful for current and former members of the Colón-Ramos lab for their guidance and suggestions, in particular, Agustín Almoril-Porras and Malcom Díaz García for assisting with data formatting, Patricia Chanabá-López and Andrea Cuentas-Condori for feedback on the NeuroSC website, Mayra Blakey for administrative roles in managing contracts for funding distribution, and Ben Clark and Milind Singh for feedback on the paper. We also thank Stephen Larson, Dario Del Piano and Zoran Sinnema (MetaCell) for initial website software development, method reporting and hosting services. We thank Brandi Mattson for editing early paper drafts. We acknowledge Ryan Christensen and Hari Shroff (Janelia Research Campus) and Patrick La Riviere (University of Chicago) for helpful discussions and guidance for the NeuroSC website. We thank the Research Center for Minority Institutions program, the Marine Biological Laboratories (MBL), and the Instituto de Neurobiología de la Universidad de Puerto Rico for providing meeting and brainstorming platforms. D.A.C-R. acknowledges the Whitman Fellows program at MBL for providing funding and space for discussions valuable to this work. Research in D.A.C-R. and W.A.M. labs was supported by NIH grant R24-OD016474. This work was also funded by the NIH/NINDS grant R35 NS132156-01, DP1 NS111778 and R01 NS076558–2.

Authorship Contributions

N.L.K. Conceptualization; Data curation; Investigation; Methodology; Project Administration; Validation; Visualization; Writing- Original Draft S.E.E. Conceptualization; Data curation; Formal Analysis; Investigation; Project Administration; Software; Visualization; Writing- Original Draft: reviewer edits; Final Draft D.B. Formal Analysis; Software; Writing - Original Draft M.W.M. Conceptualization; Formal Analysis; Investigation; Project Administration; Software; Writing- Review, Editing P.A.-M. Data curation; Writing- Review, Editing N.V.M. Data curation; Investigation; Writing - Review, Editing S.K. Resources; Supervision J.I.E - Software development W.A.M. Conceptualization; Data curation; Formal Analysis; Funding Acquisition; Methodology; Resources; Software; Validation; Writing- Review, editing; Corresponding Author D.A.C.-R. Conceptualization; Funding Acquisition; Resources; Supervision; Visualization; Writing - Review, editing; Corresponding Author

Competing Interests

Authors do not declare any competing interests.

Declaration of generative AI and AI-assisted technologies in the writing process.

During the preparation of this work the author(s) used ChatGPT in order to improve readability. After using this tool, the author(s) reviewed and edited the content as needed and take full responsibility for the content of the published article.

References

Altun, Z F , Herndon, L A , Wolkow, C A , Crocker, C , Lints, R , Hall, D H (ed s), WormAtlas; 2002. http://www.wormatlas.org.

AU Duncan LH, AU Moyle MW, AU Shao L, AU Sengupta T, AU Ikegami R, AU Kumar A, AU Guo M, AU Christensen R, AU Santella A, AU Bao Z, AU Shroff H, AU Mohler W, AU Colón-Ramos DA. Isotropic Light-Sheet Microscopy and Automated Cell Lineage Analyses to Catalogue Caenorhabditis elegans Embryogenesis with Subcellular Resolution. JoVE. 2019 Jun; (148):e59533. https://www.jove.com/t/59533, doi: 10.3791/59533, publisher: MyJoVE Corp.

Barabási DL, Bianconi G, Bullmore E, Burgess M, Chung S, Eliassi-Rad T, George D, Kovács IA, Makse H, Nichols TE, Papadimitriou C, Sporns O, Stachenfeld K, Toroczkai Z, Towlson EK, Zador AM, Zeng H, Barabási AL, Bernard A, Buzsáki G. Neuroscience Needs Network Science. The Journal of Neuroscience: The Official Journal of the Society for Neuroscience. 2023 Aug; 43(34):5989–5995. doi: 10.1523/JNEUROSCI.1014-23.2023.

Boergens KM, Berning M, Bocklisch T, Bräunlein D, Drawitsch F, Frohnhofen J, Herold T, Otto P, Rzepka N, Werkmeister T, Werner D, Wiese G, Wissler H, Helmstaedter M. webKnossos: efficient online 3D data annotation for connectomics. Nature Methods. 2017; 14(7):691–694. https://www.nature.com/articles/nmeth.4331, doi: 10.1038/nmeth.4331, publisher: Nature Publishing Group.

Brittin CA, Cook SJ, Hall DH, Emmons SW, Cohen N. A multi-scale brain map derived from whole-brain volumetric reconstructions. Nature. 2021 Mar; 591(7848):105–110. doi: 10.1038/s41586-021-03284-x.

Brittin CA, Cook SJ, Hall DH, Emmons SW, Cohen N. Volumetric reconstruction of main Caenorhabditis elegans neuropil at two different time points. bioRxiv. 2018; p. 485771. doi: 10.1101/485771.

Brugnone N, Gonopolskiy A, Moyle MW, Kuchroo M, van Dijk D, Moon KR, Colon-Ramos D, Wolf G, Hirn MJ, Krishnaswamy S. Coarse Graining of Data via Inhomogeneous Diffusion Condensation. Proc IEEE Int Conf Big Data. 2019 Dec; 2019:2624–2633. doi: 10.1109/BigData47090.2019.9006013.

Cantarelli M, Marin B, Quintana A, Earnshaw M, Court R, Gleeson P, Dura-Bernal S, Silver RA, Idili G. Geppetto: a reusable modular open platform for exploring neuroscience data and models. Philosophical transactions of the Royal Society of London Series B, Biological sciences. 2018 Sep; 373(1758). doi: 10.1098/rstb.2017.0380.

Choi YK, Feng L, Jeong WK, Kim J. Connecto-informatics at the mesoscale: current advances in image processing and analysis for mapping the brain connectivity. Brain Informatics. 2024 Jun; 11(1):15. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11150223/, doi: 10.1186/s40708-024-00228-9.

Collins FS, Fink L. The Human Genome Project. Alcohol health and research world. 1995; 19(3):190–195.

Collinson LM, Bosch C, Bullen A, Burden JJ, Carzaniga R, Cheng C, Darrow MC, Fletcher G, Johnson E, Narayan K, Peddie CJ, Winn M, Wood C, Patwardhan A, Kleywegt GJ, Verkade P. Volume EM: a quiet revolution takes shape. Nature Methods. 2023 Jun; 20(6):777–782. https://www.nature.com/articles/s41592-023-01861-8, doi: 10.1038/s41592-023-01861-8, publisher: Nature Publishing Group.

Colón-Ramos DA, Margeta MA, Shen K. Glia promote local synaptogenesis through UNC-6 (netrin) signaling in C. elegans. Science (New York, NY). 2007 Oct; 318(5847):103–106. doi: 10.1126/science.1143762.

Cook SJ, Jarrell TA, Brittin CA, Wang Y, Bloniarz AE, Yakovlev MA, Nguyen KCQ, Tang LTH, Bayer EA, Duerr JS, Bülow HE, Hobert O, Hall DH, Emmons SW. Whole-animal connectomes of both Caenorhabditis elegans sexes. Nature. 2019 Jul; 571(7763):63–71. https://doi.org/10.1038/s41586-019-1352-7, doi: 10.1038/s41586-019-1352-7.

Cuentas-Condori A, Mulcahy B, He S, Palumbos S, Zhen M, Miller I David M. *C. elegans* neurons have functional dendritic spines. eLife. 2019 Oct; 8:e47918. https://doi.org/10.7554/eLife.47918, doi: 10.7554/eLife.47918, publisher: eLife Sciences Publications, Ltd.

Dorkenwald S, Matsliah A, Sterling AR, Schlegel P, Yu Sc, McKellar CE, Lin A, Costa M, Eichler K, Yin Y, Silversmith W, Schneider-Mizell C, Jordan CS, Brittain D, Halageri A, Kuehner K, Ogedengbe O, Morey R, Gager J, Kruk K, et al. Neuronal wiring diagram of an adult brain. bioRxiv. 2023 Jan; p. 2023.06.27.546656. http://biorxiv.org/content/early/2023/07/11/2023.06.27.546656.abstract, doi: 10.1101/2023.06.27.546656.

Eberle AL, Zeidler D. Multi-Beam Scanning Electron Microscopy for High-Throughput Imaging in Connectomics Research. Frontiers in Neuroanatomy. 2018 Dec; 12. https://www.frontiersin.org/journals/neuroanatomy/articles/10.3389/fnana.2018.00112/full, doi: 10.3389/fnana.2018.00112, publisher: Frontiers.

Fenyves BG, Szilágyi GS, Vassy Z, Sőti C, Csermely P. Synaptic polarity and sign-balance prediction using gene expression data in the Caenorhabditis elegans chemical synapse neuronal connectome network. PLOS Computational Biology. 2020 Dec; 16(12):e1007974. https://doi.org/10.1371/journal.pcbi.1007974, doi: 10.1371/journal.pcbi.1007974.

Flavell SW, Raizen DM, You YJ. Behavioral States. Genetics. 2020; 216(2):315–332. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7536859/, doi: 10.1534/genetics.120.303539.

Galili DS, Jefferis GS, Costa M. Connectomics and the neural basis of behaviour. Current opinion in insect science. 2022 Dec; 54:100968. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7614087/, doi: 10.1016/j.cois.2022.100968.

Hall DH, Altun ZF, WormAtlas: The anatomy of *Caenorhabditis elegans*; 2008. Accessed May 21, 2025. https://www.wormatlas.org.

Heinrich L, Bennett D, Ackerman D, Park W, Bogovic J, Eckstein N, Petruncio A, Clements J, Pang S, Xu CS, Funke J, Korff W, Hess HF, Lippincott-Schwartz J, Saalfeld S, Weigel AV, COSEM Project Team. Whole-cell organelle segmentation in volume electron microscopy. Nature. 2021 Nov; 599(7883):141–146. doi: 10.1038/s41586-021-03977-3.

Kaiser M. Connectomes: from a sparsity of networks to large-scale databases. Frontiers in Neuroinformatics. 2023 Jun; 17. https://www.frontiersin.org/journals/neuroinformatics/articles/10.3389/fninf.2023.1170337/full, doi: 10.3389/fninf.2023.1170337, publisher: Frontiers.

Kasthuri N, Hayworth KJ, Berger DR, Schalek RL, Conchello JA, Knowles-Barley S, Lee D, Vázquez-Reina A, Kaynig V, Jones TR, Roberts M, Morgan JL, Tapia JC, Seung HS, Roncal WG, Vogelstein JT, Burns R, Sussman DL, Priebe CE, Pfister H, et al. Saturated Reconstruction of a Volume of Neocortex. Cell. 2015 Jul; 162(3):648–661. https://www.sciencedirect.com/science/article/pii/S0092867415008247, doi: 10.1016/j.cell.2015.06.054.

Lichtman JW, Pfister H, Shavit N. The big data challenges of connectomics. Nature Neuroscience. 2014 Nov; 17(11):1448–1454. https://www.nature.com/articles/nn.3837, doi: 10.1038/nn.3837, publisher: Nature Publishing Group.

Maitin-Shepard J, Baden A, Silversmith W, Perlman E, Collman F, Blakely T, Funke J, Jordan C, Falk B, Kemnitz N, tingzhao, Roat C, Castro M, Jagannathan S, moenigin, Clements J, Hoag A, Katz B, Parsons D, Wu J, et al., google/neuroglancer:. Zenodo; 2021. https://zenodo.org/records/5573294, doi: 10.5281/zenodo.5573294.

Moon KR, van Dijk D, Wang Z, Gigante S, Burkhardt DB, Chen WS, Yim K, Elzen Avd, Hirn MJ, Coifman RR, Ivanova NB, Wolf G, Krishnaswamy S. Visualizing structure and transitions in high-dimensional biological data. Nature Biotechnology. 2019 Dec; 37(12):1482–1492. https://doi.org/10.1038/s41587-019-0336-3, doi: 10.1038/s41587-019-0336-3.

Moyle MW, Barnes KM, Kuchroo M, Gonopolskiy A, Duncan LH, Sengupta T, Shao L, Guo M, Santella A, Christensen R, Kumar A, Wu Y, Moon KR, Wolf G, Krishnaswamy S, Bao Z, Shroff H, Mohler WA, Colón-Ramos DA. Structural and developmental principles of neuropil assembly in C. elegans. Nature. 2021 Mar; 591(7848):99–104. doi: 10.1038/s41586-020-03169-5.

Newman MEJ. Modularity and community structure in networks. Proceedings of the National Academy of Sciences of the United States of America. 2006; 103(23):8577–8582. doi: 10.1073/pnas.0601602103.

Packer JS, Zhu Q, Huynh C, Sivaramakrishnan P, Preston E, Dueck H, Stefanik D, Tan K, Trapnell C, Kim J, Waterston RH, Murray JI. A lineage-resolved molecular atlas of C. elegans embryogenesis at single-cell resolution. Science (New York, NY). 2019 Sep; 365(6459). doi: 10.1126/science.aax1971, place: United States.

Perez AJ, Seyedhosseini M, Deerinck TJ, Bushong EA, Panda S, Tasdizen T, Ellisman MH. A workflow for the automatic segmentation of organelles in electron microscopy image stacks. Frontiers in Neuroanatomy. 2014 Nov; 8. https://www.frontiersin.org/journals/neuroanatomy/articles/10.3389/fnana.2014.00126/full, doi: 10.3389/fnana.2014.00126, publisher: Frontiers.

Phelps JS, Hildebrand DGC, Graham BJ, Kuan AT, Thomas LA, Nguyen TM, Buhmann J, Azevedo AW, Sustar A, Agrawal S, Liu M, Shanny BL, Funke J, Tuthill JC, Lee WCA. Reconstruction of motor control circuits in adult Drosophila using automated transmission electron microscopy. Cell. 2021 Feb; 184(3):759–774.e18. https://www.cell.com/cell/abstract/S0092-8674(20)31683-4, doi: 10.1016/j.cell.2020.12.013, publisher: Elsevier.

Poole RJ, Flames N, Cochella L. Neurogenesis in *Caenorhabditis elegans*. Genetics. 2024; 228(2):iyae116. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC11457946/, doi: 10.1093/genetics/iyae116.

Randi F, Sharma AK, Dvali S, Leifer AM. Neural signal propagation atlas of Caenorhabditis elegans. Nature. 2023 Nov; 623(7986):406–414. https://doi.org/10.1038/s41586-023-06683-4, doi: 10.1038/s41586-023-06683-4.

Rapti G, Li C, Shan A, Lu Y, Shaham S. Glia initiate brain assembly through noncanonical Chimaerin-Furin axon guidance in C. elegans. Nature Neuroscience. 2017 Oct; 20(10):1350–1360. doi: 10.1038/nn.4630.

Rivlin PK, Januszewski M, Longden KD, Neace E, Scheffer LK, Ordish C, Clements J, Phillips E, Smith N, Takemura S, Umayam L, Walsh C, Yakal EA, Plaza SM, Berg S, Connectomic Analysis of Mitochondria in the Central Brain of Drosophila. bioRxiv; 2024. https://www.biorxiv.org/content/10.1101/2024.04.21.590464v1, doi: 10.1101/2024.04.21.590464, pages: 2024.04.21.590464 Section: New Results.

Schmid B, Schindelin J, Cardona A, Longair M, Heisenberg M. A high-level 3D visualization API for Java and ImageJ. BMC Bioinformatics. 2010 May; 11(1):274. https://doi.org/10.1186/1471-2105-11-274, doi: 10.1186/1471-2105-11-274.

Stein L, Sternberg P, Durbin R, Thierry-Mieg J, Spieth J. WormBase: network access to the genome and biology of Caenorhabditis elegans. Nucleic acids research. 2001 Jan; 29(1):82–86. doi: 10.1093/nar/29.1.82.

Sulston JE, Horvitz HR. Post-embryonic cell lineages of the nematode *Caenorhabditis elegans*. Developmental Biology. 1977; 56(1):110–156. doi: 10.1016/0012-1606(77)90158-0.

Sulston JE, Schierenberg E, White JG, Thomson JN. The embryonic cell lineage of the nematode Caenorhabditis elegans. Dev Biol. 1983 Nov; 100(1):64–119. doi: 10.1016/0012-1606(83)90201-4.

Sun H, Hobert O. Temporal transitions in the postembryonic nervous system of the nematode Caenorhabditis elegans: Recent insights and open questions. Special Issue: Temporal patterning in the CNS. 2023 Jun; 142:67–80. https://www.sciencedirect.com/science/article/pii/S1084952122001872, doi: 10.1016/j.semcdb.2022.05.029.

Swanson LW, Lichtman JW. From cajal to connectome and beyond. Annual Review of Neuroscience. 2016; 39(Volume 39, 2016):197–216. https://www.annualreviews.org/content/journals/10.1146/annurev-neuro-071714-033954, doi: https://doi.org/10.1146/annurev-neuro-071714-033954.

Taylor SR, Santpere G, Weinreb A, Barrett A, Reilly MB, Xu C, Varol E, Oikonomou P, Glenwinkel L, McWhirter R, Poff A, Basavaraju M, Rafi I, Yemini E, Cook SJ, Abrams A, Vidal B, Cros C, Tavazoie S, Sestan N, et al. Molecular topography of an entire nervous system. Cell. 2021 Aug; 184(16):4329–4347.e23. doi: 10.1016/j.cell.2021.06.023.

Toga AW, Clark KA, Thompson PM, Shattuck DW, Van Horn JD. Mapping the Human Connectome. Neurosurgery. 2012 Jul; 71(1):1–5. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3555558/, doi: 10.1227/NEU.0b013e318258e9ff.

Wang C, Vidal B, Sural S, Loer C, Aguilar GR, Merritt DM, Toker IA, Vogt MC, Cros C, Hobert O. A neurotransmitter atlas ofC. elegansmales and hermaphrodites. CSH. 2023 Dec; http://dx.doi.org/10.1101/2023.12.24.573258, doi: 10.1101/2023.12.24.573258, publisher: Cold Spring Harbor Laboratory.

White JG, Southgate E, Thomson JN, Brenner S. The structure of the nervous system of the nematode *Caenorhabditis* elegans. Philosophical Transactions of the Royal Society of London B, Biological Sciences. 1986; 314(1165):1–340. doi: doi:10.1098/rstb.1986.0056.

Witvliet D, Mulcahy B, Mitchell JK, Meirovitch Y, Berger DR, Wu Y, Liu Y, Koh WX, Parvathala R, Holmyard D, Schalek RL, Shavit N, Chisholm AD, Lichtman JW, Samuel ADT, Zhen M. Connectomes across development reveal principles of brain maturation. Nature. 2021 Aug; 596(7871):257–261. doi: 10.1038/s41586-021-03778-8.

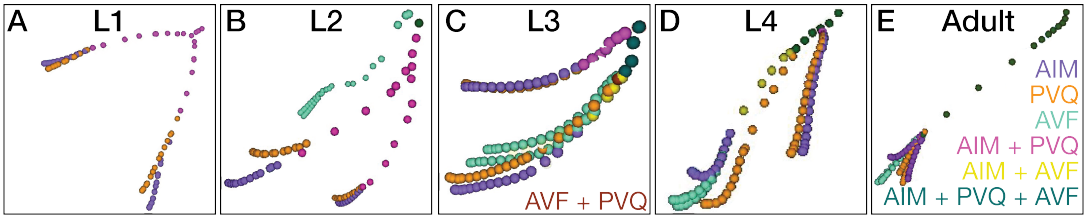
Xu CS, Hayworth KJ, Lu Z, Grob P, Hassan AM, García-Cerdán JG, Niyogi KK, Nogales E, Weinberg RJ, Hess HF. Enhanced FIB-SEM systems for large-volume 3D imaging. eLife. 2017 May; 6:e25916. https://doi.org/10.7554/eLife.25916, doi: 10.7554/eLife.25916.

Xu CS, Pang S, Shtengel G, Müller A, Ritter AT, Hoffman HK, Takemura SY, Lu Z, Pasolli HA, Iyer N, Chung J, Bennett D, Weigel AV, Freeman M, van Engelenburg SB, Walther TC, Farese RV, Lippincott-Schwartz J, Mellman I, Solimena M, et al. An open-access volume electron microscopy atlas of whole cells and tissues. Nature. 2021 Nov; 599(7883):147–151. doi: 10.1038/s41586-021-03992-4.

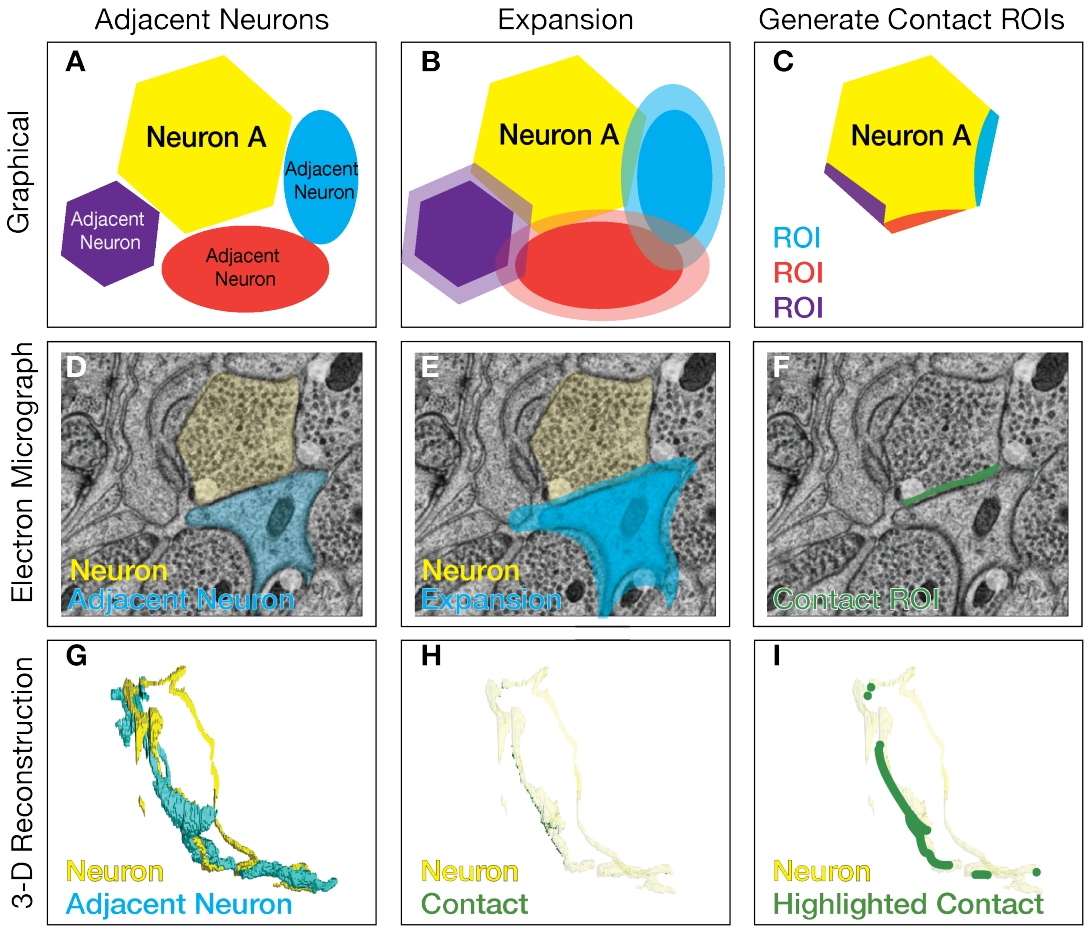
Zheng Z, Lauritzen JS, Perlman E, Robinson CG, Nichols M, Milkie D, Torrens O, Price J, Fisher CB, Sharifi N, Calle-Schuler SA, Kmecova L, Ali IJ, Karsh B, Trautman ET, Bogovic JA, Hanslovsky P, Jefferis GSXE, Kazhdan M, Khairy K, et al. A Complete Electron Microscopy Volume of the Brain of Adult Drosophila melanogaster. Cell. 2018 Jul; 174(3):730–743.e22. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6063995/, doi: 10.1016/j.cell.2018.06.019.

Supplementary material

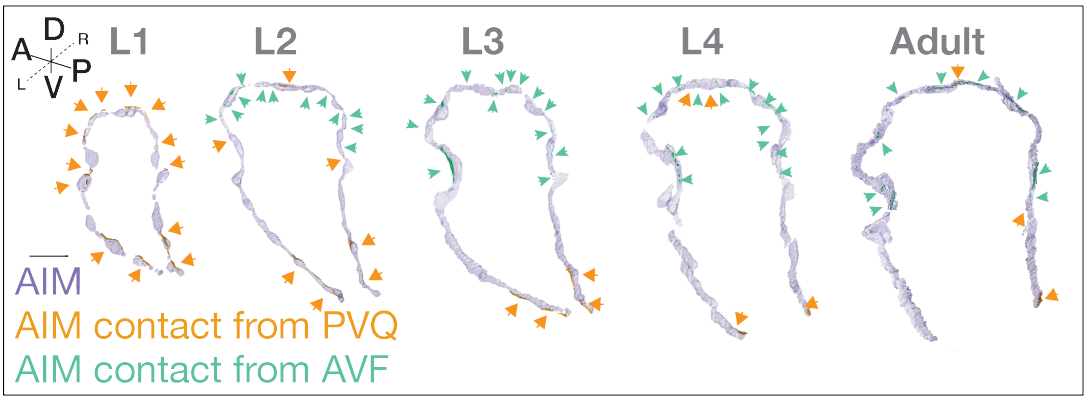
Supplementary Figures



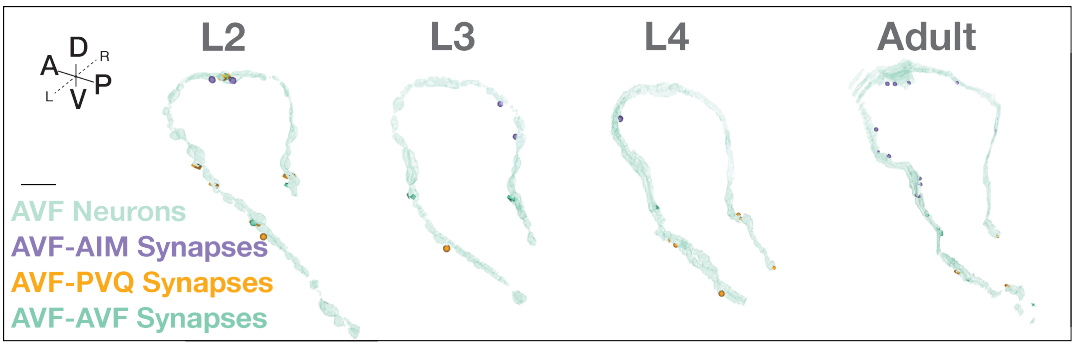
Supplementary Figure S1. DC/C-PHATE clustering of AIM, PVQ, and AVF across postembryonic development. (A-E) A cropped view of the DC/C-PHATE plot colored to identify individual neurons and clustering events in (A) Larval stage 1 (5 hours post hatching); (B) Larval stage 2 (23 hours post hatching); (C) Larval Stage 3 (27 hours post hatching); (D) Larval stage 4 (36 hours post hatching); and (E) Adult (48 hours post hatching). See also Video S1 and Table S7.



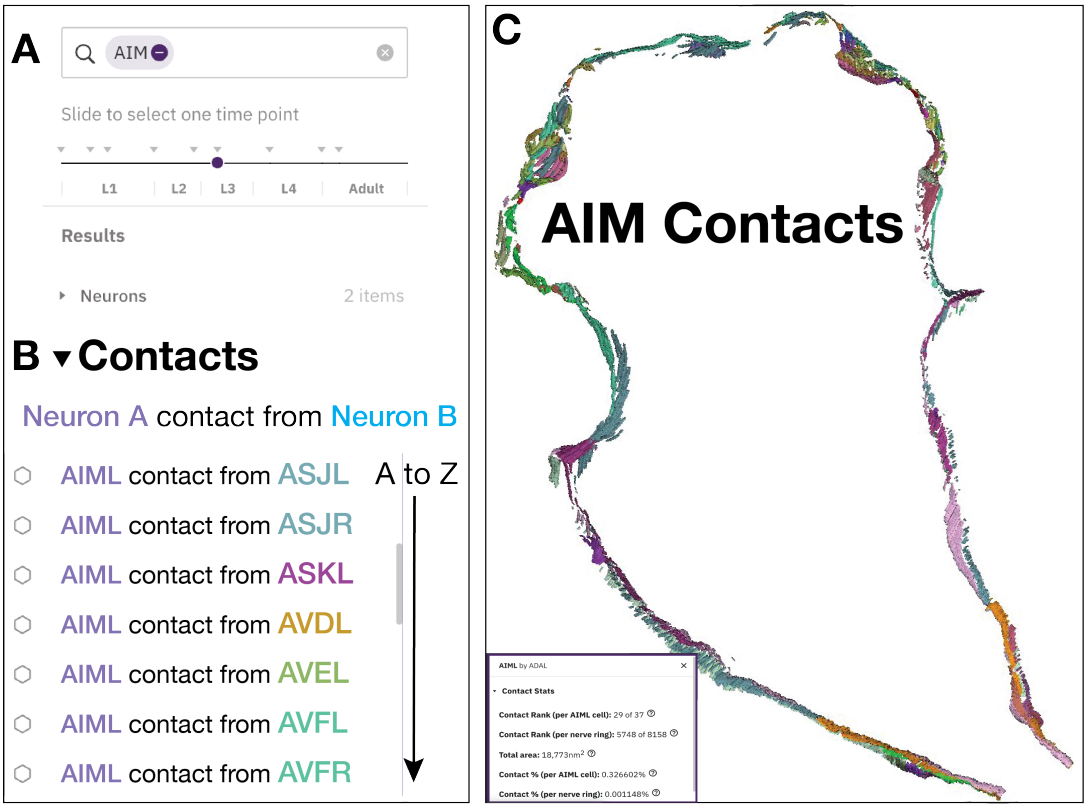
Supplementary Figure S2. Projecting contact profiles onto the segmented neuronal shapes. (A-C) Graphical representations of the strategy utilized for creating the contact profiles for each of the adjacent neurons (purple, red, cyan) onto a cross section of the neuron of interest (Neuron A, yellow). (D-F) Electron micrograph from the L4 dataset with two adjacent neurons colored yellow and cyan. To build 3-D reconstructions of contact sites from adjacent neurons, we analyzed segmented neurons from the electron microscopy datasets in each slice (A, D). Each adjacent neuron is expanded in all directions to the pixel threshold distance (specified for each dataset; Table S1; Methods; CytoSHOW.org) (B, E). A new ROI (region of interest; purple, red, cyan in C; green in F) is created from the overlapping areas between the neuron of interest (yellow) and the adjacent neurons (C,F). (G-I) 3-D reconstruction of neuron (yellow) (G) with adjacent neuron (cyan), (H) with contact sites captured (green) across all slices, and (I) with contact areas from the adjacent neuron augmented (green) as seen in Figure 5 D.



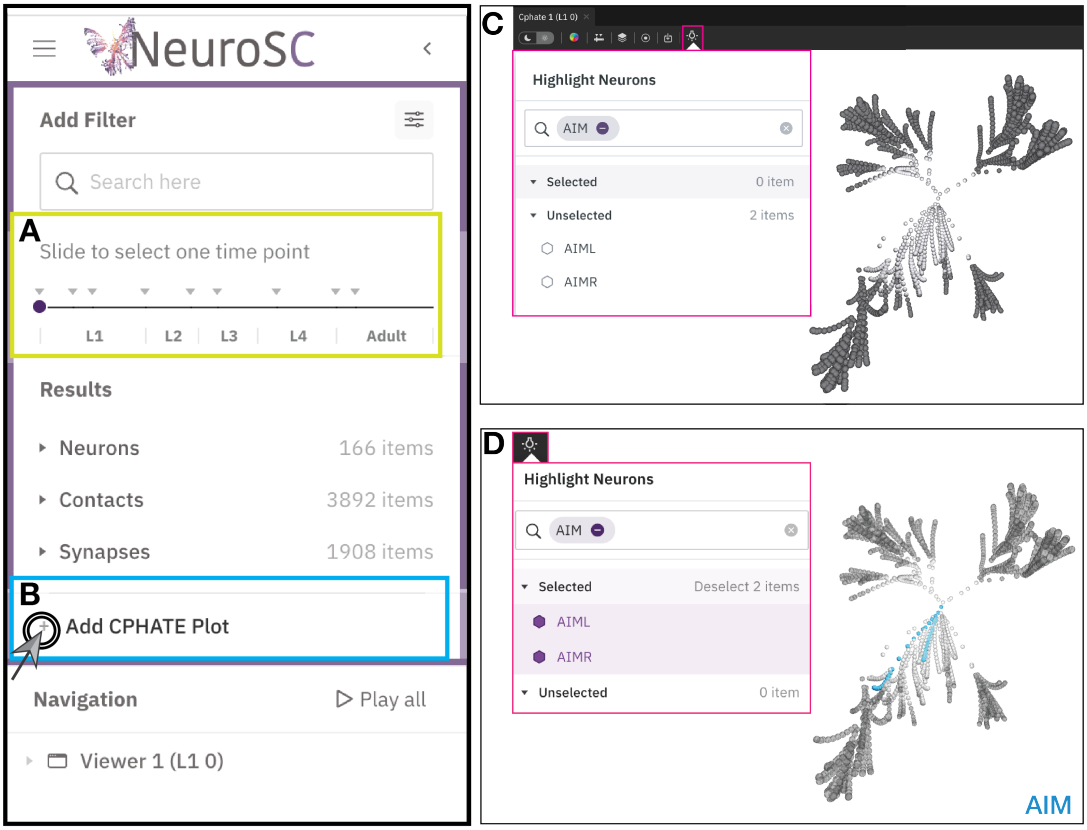
Supplementary Figure S3. AIM contact sites. Contact sites from PVQ (Orange and highlighted with orange arrowheads) and from AVF (Green and highlighted with green arrowheads) across developmental stages (as indicated) and projected onto the segmented AIM neurons (transparent purple). This figure is the unmodified NeuroSC outputs of contact profiles that corresponds to Figure 5D. In Figure 5D these contact profiles were augmented. Scale bar = 2 um. See also Figure 5 and Video S4.



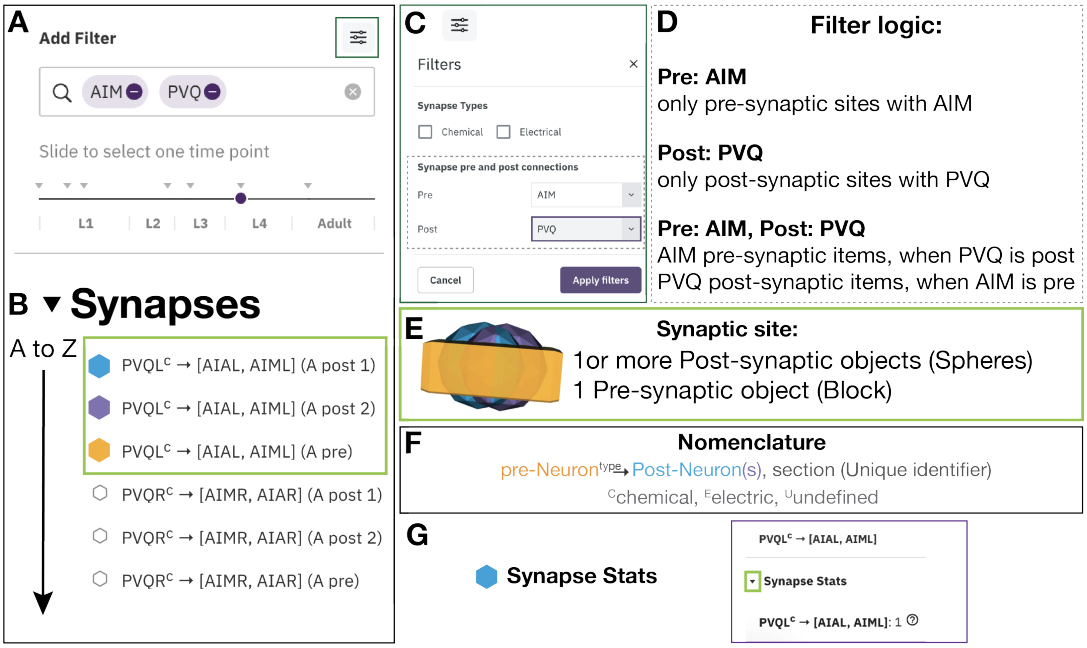
Supplementary Figure S4. AVF synaptic sites. Synaptic sites displayed onto transparent (green) AVF neurons across developmental stages. Presynaptic sites (spheres) and postsynaptic sites (Blocks) are visualized between the AVF neurons and the AIM (Purple) neurons, PVQ (Orange) neurons and other AVF (either AVFL or AVFR; opaque green) neuron; Scale bar = 2 um.



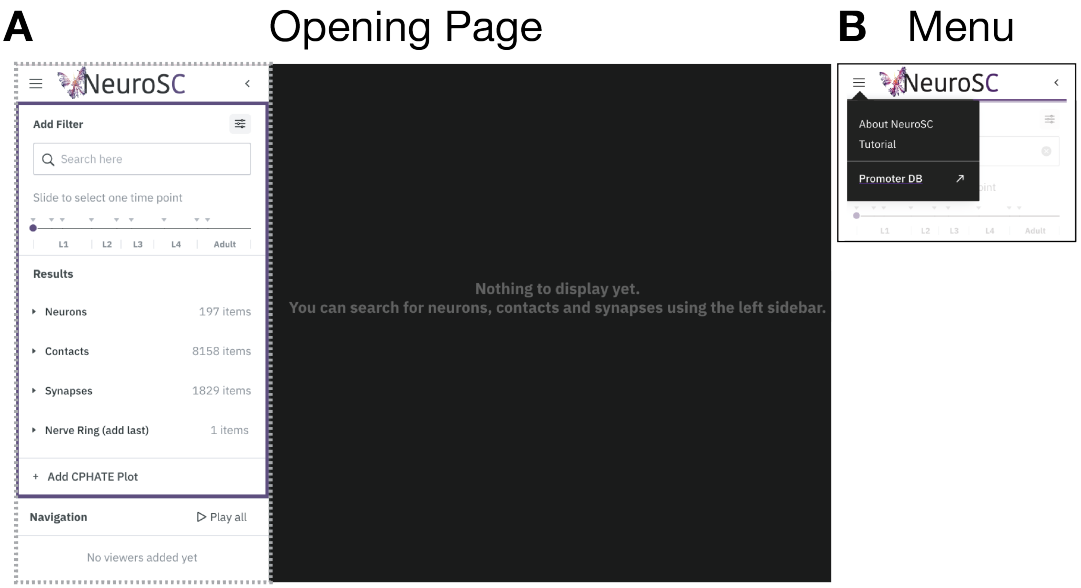
Supplementary Figure S5. Visualization of contact sites in NeuroSC. (A) Search for a specific neuron (here, AIM) to filter (B) the list of contacts corresponding to the developmental slider. Neuron A (AIML, here) is the neuron onto which the contacts will be mapped. The Contacts drop down menu sorts neurons alphabetically (here, colored according to the contact patch color in C). (C) 3-D reconstruction of all AIM contacts at L3 stage. See also Video S3-S4. In the Figure 5D, contacts are augmented. Inset: Click on a contact rendering to show 'contact stats'. This pop up displays quantifications of the selected contact relationship. Rank compares the summed surface area of contacts ("patches") between these two neurons relative to all other contact relationships for the primary neuron. A rank of 1 means this neuron pair shares the largest contact area. Total surface area is the total surface area in nm2 that covers the primary neuron in the contact relationship. Here the primary neuron is AIM. Contact area percentages are also shown for the cell and the whole nerve ring.



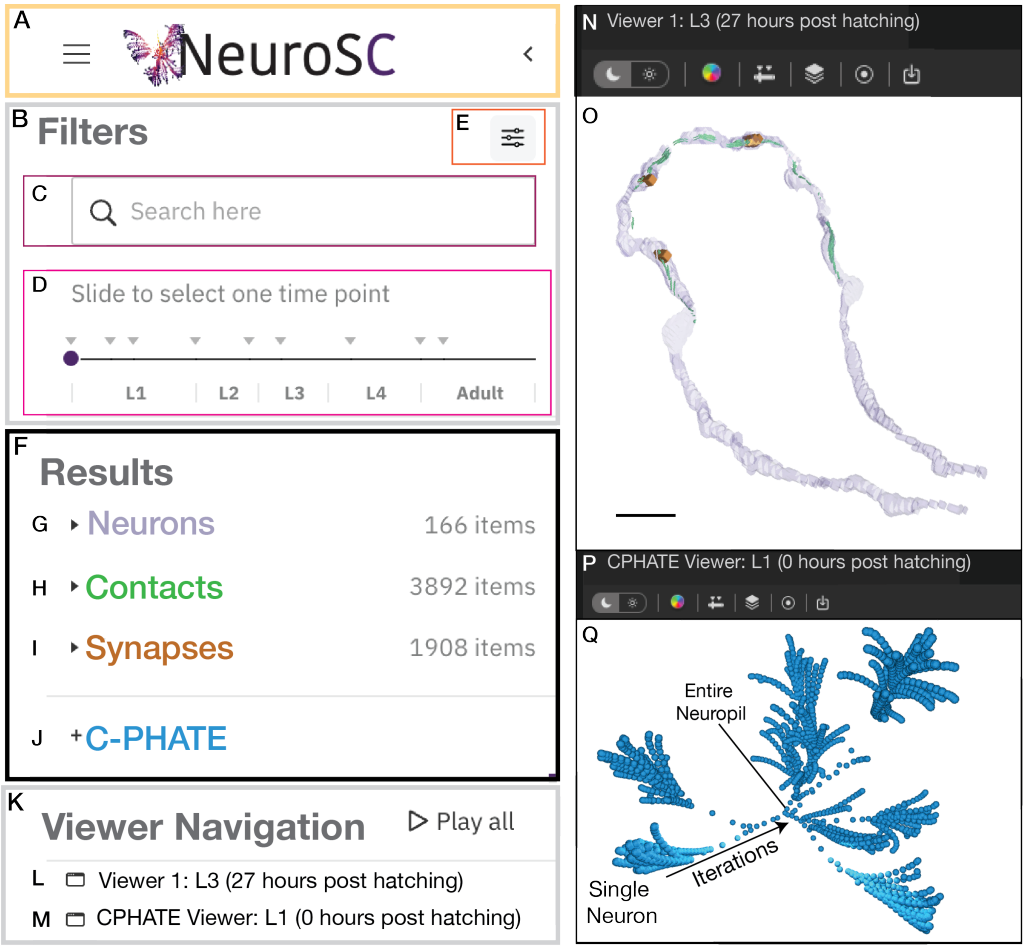
Supplementary Figure S6. C-PHATE tutorial in NeuroSC. (A) Add the C-PHATE plot corresponding to the position of the purple circle on the developmental slider (yellow box) by clicking (B) the + sign. (C) Screenshot of C-PHATE plot at L4 (36 hours post hatching), spheres represent individual neurons at the outer edge of the plot and DC iterations increase towards the center where spheres represent clusters of neurons and eventually the entire nerve ring. (D) Screenshot of C-PHATE plot at L4 (36 hours post hatching) with the spheres/clusters containing the AIM neurons highlighted (Blue) by selecting the AIM neurons within the light bulb menu (red box). See also Video S1. NeuroSC features in this figure are not shown to scale.



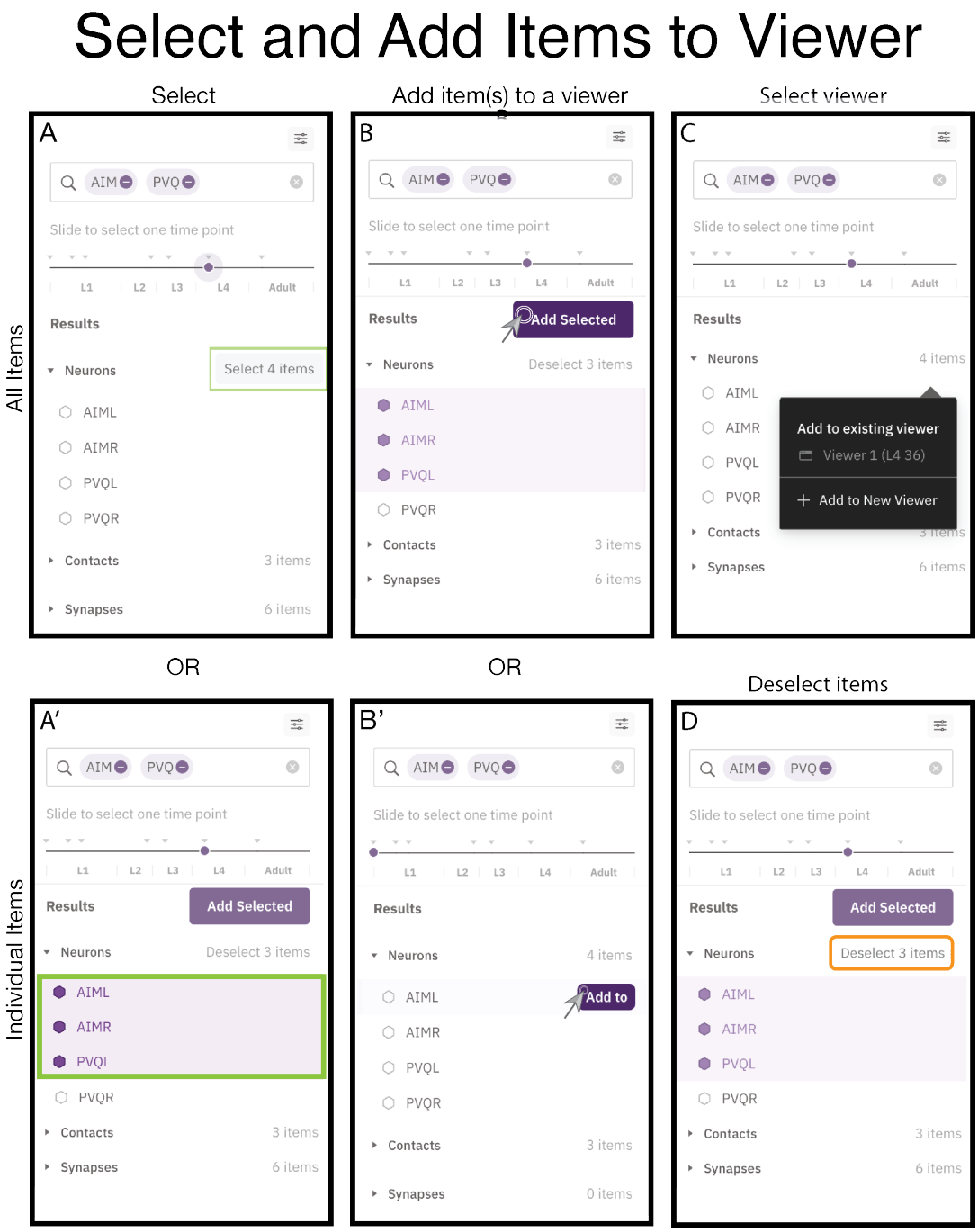
Supplementary Figure S7. Visualization of synaptic sites with NeuroSC. (A) Search for synaptic sites for specific neuron(s) (e.g., AIM, PVQ) and choose a developmental time point with the slider. (B) Synapses dropdown menu contains a list of objects representing pre- and postsynaptic sites corresponding to all neuron names in the search bar and sorted alphabetically. Searched neurons can be used with the synaptic filter (C) to select for synapse type (electrical or chemical; Note: only use this feature for L4_36 hours post hatching and Adult_48 hours post hatching) and to filter objects by synaptic specialization (pre or post; gray dotted box), (D) which will follow the filter logic (example shown for AIM and PVQ). (E) To enable visualization of subsets of synapses and differentiate between pre- and postsynaptic sites, each synapse contains object(s) representing the postsynaptic site(s) as spheres (Blue and Purple) and the presynaptic site as a block (Orange). These are ordered "by synapse", with all postsynaptic objects, then the presynaptic object. This specific example corresponds to a 3-D representation of the PVQL (Orange, Pre) AIAL (Blue, Post), AIML (Purple, Post) synapse. (F) All synaptic sites contain the name of the presynaptic neuron (Orange), neuron type (chemical, electrical, or undefined), list of postsynaptic neuron(s) (Blue), and Unique identifier (Black; Section, letter) for cases with multiple synapses between the same neurons. The 'section' is unique to each synapse between specified neurons and at that specific developmental stage. It is listed in order of its antero-posterior position in the neuron. Synapse names are not linked through developmental datasets. If the synapse is polyadic, there will be multiple postsynaptic neuron names and objects associated with a single presynaptic site. See also Video S4. (G) Right click on a rendered synapse to open synapse stats. Expand (click arrow in green box) to show the numbers of synapses, broken down by polyadic variations if applicable, per the primary neuron (in this case the primary neuron is PVQL).



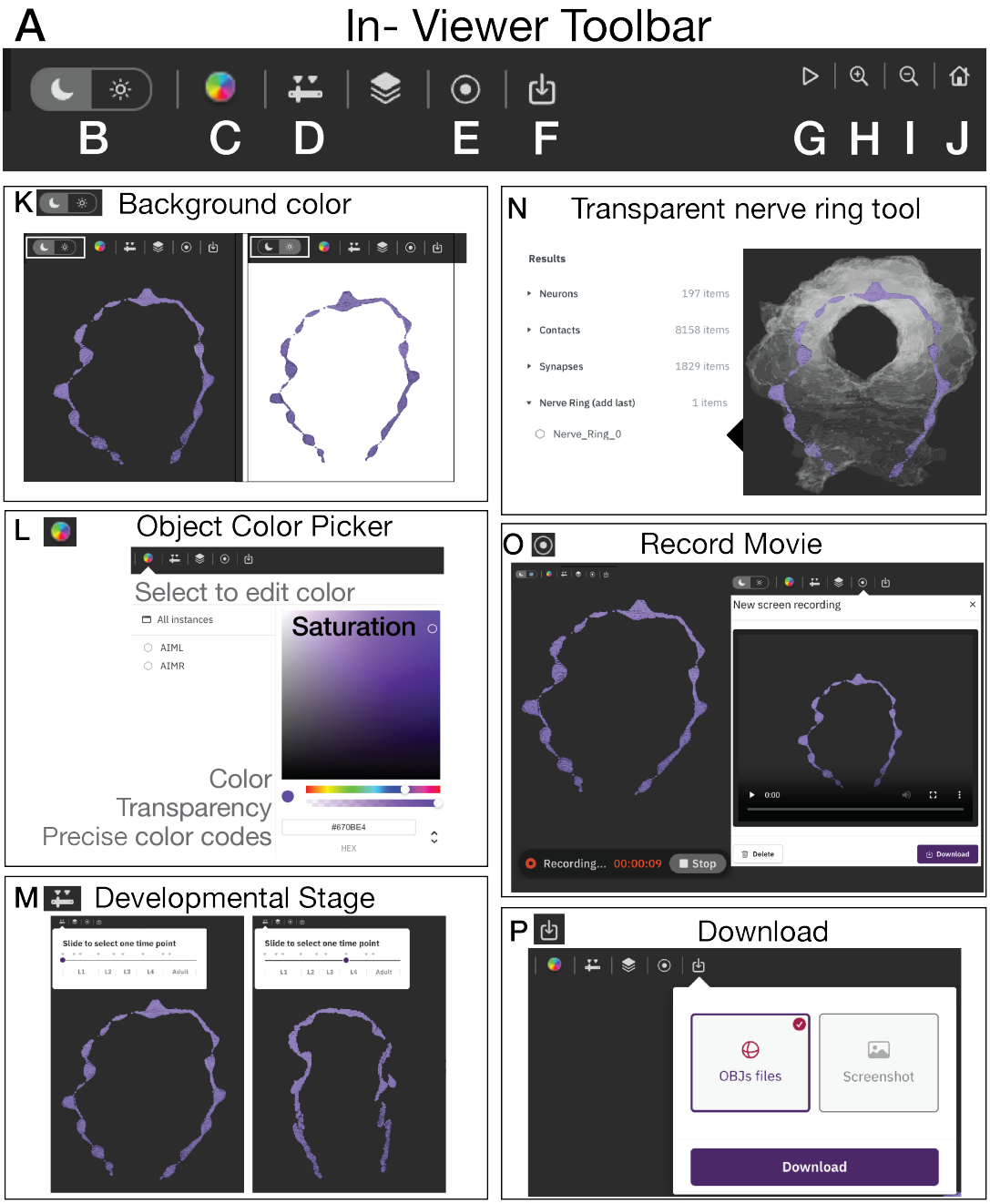
Supplementary Figure S8. Opening page view and menu.(A) View of opening page. (B) Menu for access to the 'About' window for referencing source information, the Tutorial, and the developmental Promoter database. See also Video S3.



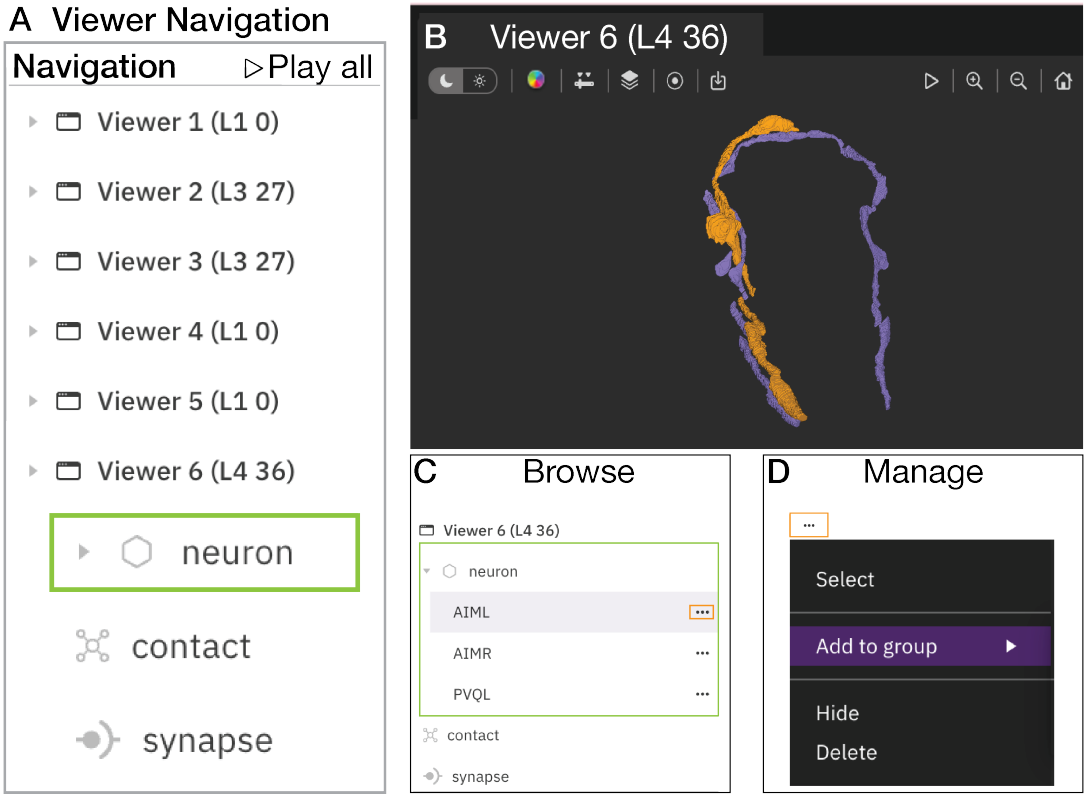
Supplementary Figure S9. The NeuroSC interface enables interrogation of neuronal relationships across development. (A) The left facing arrow to minimize the left panel and optimize space for the viewer windows. The interface contains four main parts: (B-E) Filters, (F-J) Results, (K-M) Viewer Navigation, and (N-Q) viewer windows. Filter Results by (C) searching for neuron names, (D) selecting a dataset with the developmental slider (in hours post- hatching), (E) and filtering synapses based on the pre- or post-synaptic partner on the neurons that are on the search bar. (F) Results drop down menus (filtered by B) for (G) Neuronal morphologies (shown in the viewer as purple in (O)), (H) Contacts (shown in green (O)); (I) Synapses (shown in Orange in (O)); and (J) C-PHATE (shown in (Q)), which gets filtered by the developmental slider in (D). (K) Viewer Navigation to rotate the 3-D projections in all viewers simultaneously (Play All) and which contains a drop-down menu for each viewer (L,M). The viewers are named as Viewer 1 (L, N) or CPHATE viewer (M, P) and followed by information of the developmental stage and the hours post hatching for the objects in the viewer. (O) Reconstruction of the AIM neurons with AVF contacts and synapses at L3 (27 hours post hatching; scale bar = 2 um. (Q) C-PHATE plot at L1 (0 hours post hatching). See also Video S3.



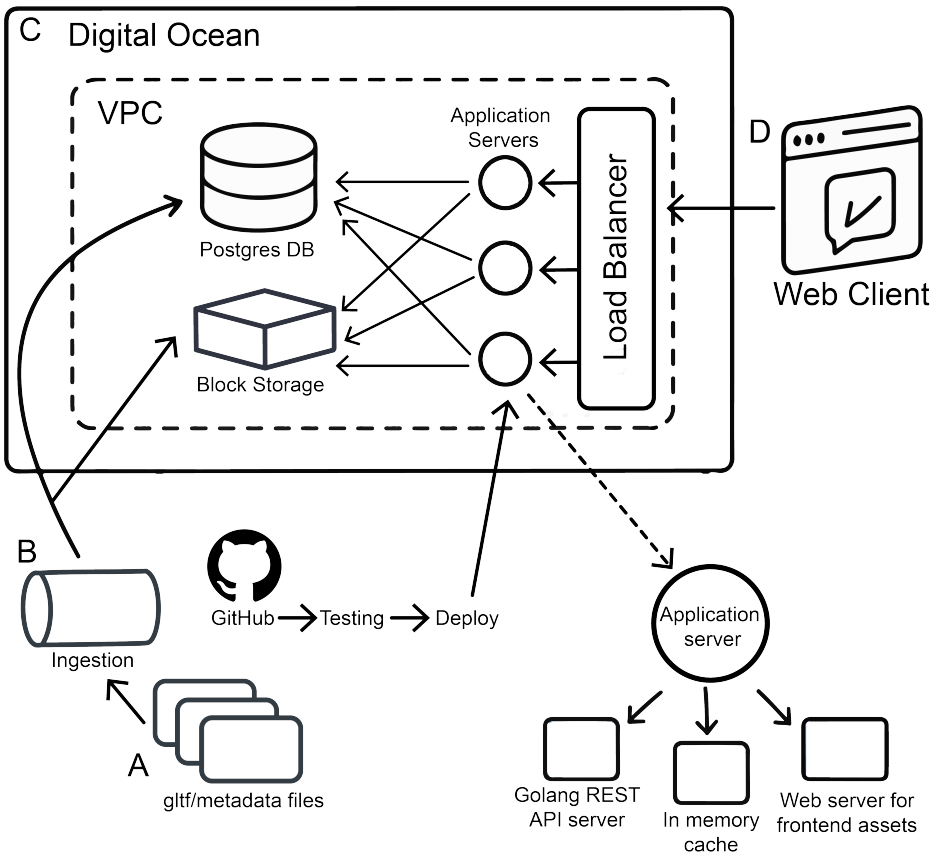
Supplementary Figure S10. Select and Add objects to viewers. (A) Click "select (number) items" to select all items in the drop down list (green box), or (A') click the hexagon next to each item (green box). (B) Click "Add Selected" (purple box) to add all selected items or (B') click "Add to" (purple box) to add each item individually. (C) To add the selected item(s) to an existing viewer of the same developmental stage or to a new viewer, choose a viewer as indicated. (D) Click "Deselect (number) items" (orange box) to deselect items. See also Video S3 and S4.



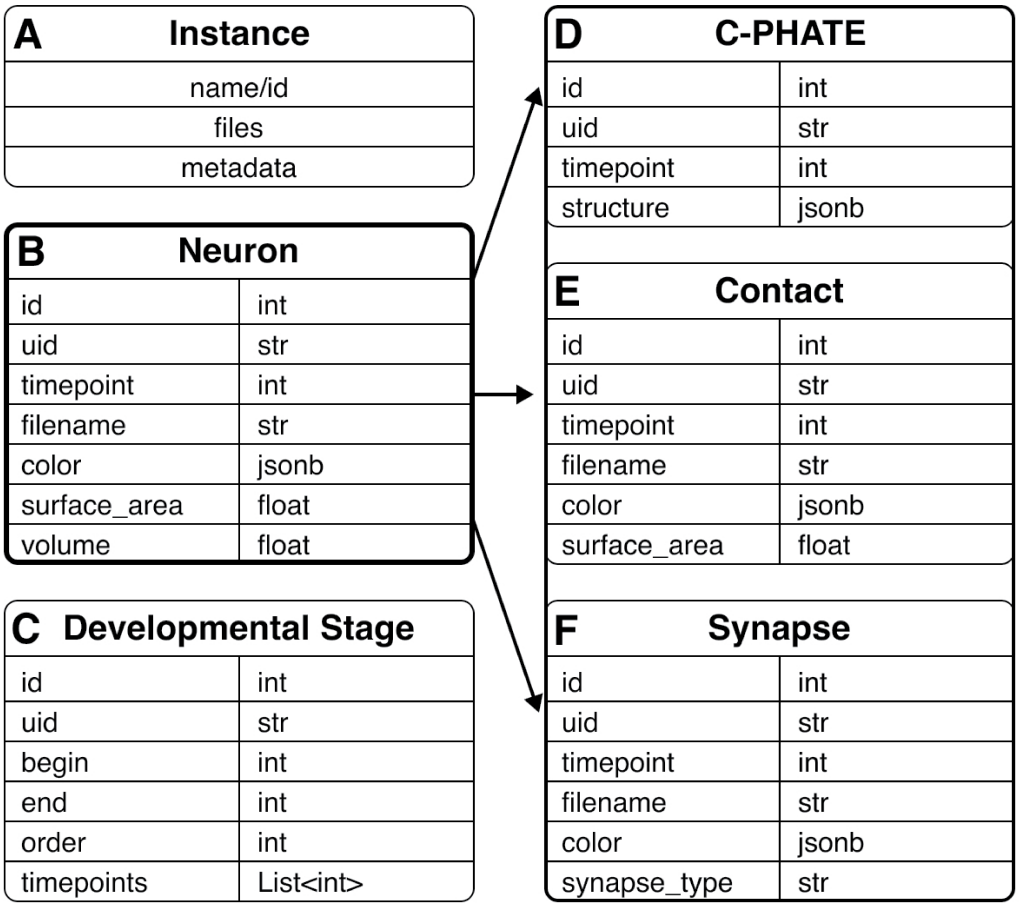
Supplementary Figure S11. In-viewer toolbar features (A) In-viewer toolbar for Neurons, Contacts and Synapses and C-PHATE (shown here, only Neurons). (B, K) Change the background color of viewer from dark (white box, moon) to white (white box, sun). (C, L) Change the color of any objects by selecting a desired color, transparency or color code and selecting the object (or instance) name (here, AIML and AIMR). (D, M) Copy and paste your neuron and contact profile scenes through time into a comparative side by side window changing the developmental stage for items in the viewer by using the in-viewer developmental slider. Note a new window will be generated, which can be dragged next to the original window.(N) Add 3-D representations of the Nerve Ring for that developmental stage, note: must be added to the scene last. (E, O) Record and download movies for the viewer. (F,P) Download .gltf files and viewer screenshot (png). (G) Rotate objects around the y-axis. (H) Zoom in and (I) zoom out, and (J) reset objects to original positions in the viewer. See also Video S3.



Supplementary Figure S12. Viewer navigation menu. ((A) Navigation bar contains a drop-down menu for each viewer (shown here, six viewers at varied developmental stages) and a "Play all" button for simultaneously rotating all objects in each viewer around the y-axis (Video S3). Each viewer drop down menu contains a drop down menu for Neurons (green box), Contacts and Synapses. (B) Viewer 6 with reconstructions of three neurons (AIML and AIMR, purple; PVQL, orange) at Larval Stage 4 (L4), 36 hours post hatching. (C) Browse and Select objects in the viewer by navigating the nested drop down menus. (D) Manage objects in viewers with options to select, group, hide, and delete objects in each viewer. Objects can be deleted with "select" and keyboard "delete". See also Video S3.



Supplementary Figure S13. NeuroSC architecture. (A) Source data are defined in a file tree structure that contains various assets such as .gltf files representing various entities, as well as CSVs storing entity metadata. The directory structure outlines a vertical hierarchy starting at the developmental stages, then branching downwards through neuron, C-PHATE, contact and synapse data.(B) A Golang script can be invoked to traverse the directory tree and ingest the files. This parses the hierarchical file path and file contents, verifying the data and associating it with underlying entities, writing it to the database.(C) The backend consists of a Postgres Database to store underlying data, a Block Storage Volume that houses static assets (.gltf files, javascript bundle, css styles, etc), and a Golang application. The Golang application surfaces a JSON REST API, leveraging caching strategies to reduce DB load. It also serves the static assets to the client, from the block storage. A variable number of Virtual Machines run the backend application code, scaling as needed to accommodate traffic.(D) The client-side is a React application that uses Geppetto for entity rendering (Cantarelli et al. 2018). User interactions on the frontend result in queries to the backend to filter, sort, and search via the REST API, resulting in the entities and metadata to be rendered to an interactive canvas.



Supplementary Figure S14. NeuroSC data model. (A) Reference scheme for B-F; Instance refers to the category (e.g., B, Neuron; C, Developmental Stage), which contains a name or identifier (id) for each object, lists of files associated with the instance (C, Developmental Stage does not have files), and metadata to further describe each instance, which is usually a string (str) or an integer (int). (B) The neuron name is the foundation for the Contacts, Synapses, and C-PHATE, which enables integration across each of these representations and across developmental stages (time points) with metadata from WormAtlas (wormatlas.org/MoW_built0.92/MoW.html). (C) The Developmental Stages are named by the larval stages (L1, L2, L3, L4, Adult), and the metadata captures the list of time points within those developmental stages (i.e., L1, 0 hours post hatching, and L1, 5 hours post hatching). (D) C-PHATE objects are named with a list of Neurons. (E) Contacts link to the Neuron names (Neuron A and Neuron B nomenclature in Figure S5), and metadata annotates the surface area in nm^2 of contact quantified in the source Electron Microscopy micrographs. (F) Synapses link to the Neuron names (Pre, type (chemical, elecrical or undefined), Post, and section described in Figure S7).

Supplementary Videos

Video S1. Visualization of hierarchical relationships using C-PHATE plots in NeuroSC. The process for rendering a C-PHATE plot at the L4 stage (36 hours post hatching) with the real-time loading speed. In the viewer, 3-D visualization of a C-PHATE plot (shades of cyan), which is rotated to show the dome-shape of the plots and to orient the plot to correspond to Figures 2 and 4. The highlight functionality is used to show the spheres containing AIM (teal), then PVQ (teal). The spheres of the first iterations, containing AIM and PVQ, are identified, selected and colored magenta. The AVF neurons are highlighted in teal, and the first AIM and AVF containing clusters are identified, selected and colored yellow. The first clusters containing AIML, AVF and PVQL are identified and colored green. Neurons in the left yellow and magenta clusters are reconstructed with a right click on the sphere and "Add to new viewer" selection.

Video S2. Analysis of AIM, PVQ and AVF neuronal morphologies in developmental datasets. 3-D visualizations of AIM (Purple), PVQ (Orange) and AVF (Green) at (Left viewer) L1 (5 hours post hatching) and (Right viewer) L3 (27 hours post hatching) in NeuroSC. Note that at L1, AVF has not grown into the nerve ring, therefore, only AIM and PVQ are present, but by L3, the AVF neurons have grown between the AIM and PVQ neurons.

Video S3. Navigating NeuroSC features that enable integration of Neurons, Contacts and Synapses across developmental datasets. Upon first opening NeuroSC, a tutorial will launch (Figure S8). In the NeuroSC menu one can read about NeuroSC, access the tutorial, and navigate to the embryonic promoter database (Figure S8). The video shows the user searching neurons (AIM and PVQ) and adding neurons to the viewers (Figure S10). Side-by-side viewers with AIML, AIMR, and PVQL enable comparisons across developmental stages (L1, 0 hours post hatching and L4, 36 hours post hatching). Also shown in the video are the use of the in-viewer toolbar (Figure S11) and navigation menu (Figure S12) for object exploration.

Video S4. Exploring Contacts and Synapses using NeuroSCAN. Video of user navigating the tools of NeuroSC to examine synapses and contact profiles to yield results as in (Figures S7 and S9). AIM neurons (Transparent Purple), AIM (Purple)-PVQ synaptic sites (Orange), and AIM-PVQ contact sites (Orange) at L1 (5 hours post hatching) are added into Viewer 1. AIM neurons (Transparent Purple), AIM(Purple)-PVQ synaptic sites (Orange), and AIM-PVQ contact sites (Orange), AVF (Green)-AIM synaptic sites, and AVF-AIM contact sites (Green) at L3 (27 hours post hatching) are added into Viewer 2. Contact sites and synaptic sites are compared across developmental stages by hiding AIM neurons. All contact sites for AIM are added for L1 (5 hours post hatching) into Viewer 3.

Supplementary Tables

Table S1. Nerve ring regions, resolutions, and pixel threshold distances used to calculate adjacency matrices and to create contact sites for each dataset.

Table S2. Scaling factors and rotation corrections for 3-D representations of Neurons, Contacts and Synapses for each dataset.

Table S3. Stratum 1 (Red) Sankey diagrams of clustered neurons for each Diffusion Condensation iteration in each dataset.

Table S4. Stratum 2 (Purple) Sankey diagrams of clustered neurons for each Diffusion Condensation iteration in each dataset.

Table S5. Stratum 3 (Blue) Sankey diagrams of clustered neurons for each Diffusion Condensation iteration in each dataset.

Table S6. Stratum 4 (Green) Sankey diagrams of clustered neurons for each Diffusion Condensation iteration in each dataset.

Table S7. Sankey diagrams of AIM, PVQ and AVF containing clusters for each Diffusion Condensation iteration in each dataset.

Tables S8. L1 (0 hours post hatching) adjacency counts and searchable counter for summed adjacencies. Type the name of a "Neuron of Interest" (NOI) in the indicated cell to filter for the summed adjacency counts for each contact partner. For each partner, there are two columns:

Total number of contacts (number of EM sections NOI and partner are in contact) and Total Weights (summed number of pixels NOI and partner contacts).

Tables S9. L1 (5 hours post hatching) adjacency counts and searchable counter for summed adjacencies. Type the name of a "Neuron of Interest" (NOI) in the indicated cell to filter for the summed adjacency counts for each contact partner. For each partner, there are two columns: Total number of contacts (number of EM sections NOI and partner are in contact) and Total Weights (summed number of pixels NOI and partner contacts).

Tables S10. L2 (23 hours post hatching) adjacency counts and searchable counter for summed adjacencies. Type the name of a "Neuron of Interest" (NOI) in the indicated cell to filter for the summed adjacency counts for each contact partner. For each partner, there are two columns: Total number of contacts (number of EM sections NOI and partner are in contact) and Total Weights (summed number of pixels NOI and partner contacts).

Tables S11. L3 (27 hours post hatching) adjacency counts and searchable counter for summed adjacencies. Type the name of a "Neuron of Interest" (NOI) in the indicated cell to filter for the summed adjacency counts for each contact partner. For each partner, there are two columns: Total number of contacts (number of EM sections NOI and partner are in contact) and Total Weights (summed number of pixels NOI and partner contacts).

Tables S12. L4 (36 hours post hatching) adjacency counts and searchable counter for summed adjacencies. Type the name of a "Neuron of Interest" (NOI) in the indicated cell to filter for the summed adjacency counts for each contact partner. For each partner, there are two columns: Total number of contacts (number of EM sections NOI and partner are in contact) and Total Weights (summed number of pixels NOI and partner contacts).

Tables S13. Adult (48 hours post hatching) adjacency counts and searchable counter for summed adjacencies. Type the name of a "Neuron of Interest" (NOI) in the indicated cell to filter for the summed adjacency counts for each contact partner. For each partner, there are two columns: Total number of contacts (number of EM sections NOI and partner are in contact) and Total Weights (summed number of pixels NOI and partner contacts).