Extraction of bouton-like structures from neuropil calcium imaging data

Kazushi Fukumasu[a], Akinao Nose[a,b] and Hiroshi Kohsaka[b,c*]

[a] Department of Physics, Graduate School of Science, the University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, 133-0033, Japan

[b] Department of Complexity Science and Engineering, Graduate School of Frontier Science, the University of Tokyo, 5-1-5 Kashiwanoha, Kashiwa, Chiba, 277-8561, Japan

[c] Division of General Education, Graduate School of Informatics and Engineering, the University of Electro-Communications, 1-5-1, Chofugaoka, Chofu, Tokyo, 182-8585, Japan

* For correspondence: (kohsaka@edu.k.u-tokyo.ac.jp, +81-42-443-5528)

Abstract

The neuropil, the plexus of axons and dendrites, plays a critical role in operating the circuit processing of the nervous system. Revealing the spatiotemporal activity pattern within the neuropil would clarify how the information flows throughout the nervous system. However, calcium imaging to examine the circuit dynamics has mainly focused on the soma population due to their discrete distribution. The development of a methodology to analyze the calcium imaging data of a densely packed neuropil would provide us with new insights into the circuit dynamics. Here, we propose a new method to decompose calcium imaging data of the neuropil into populations of bouton-like synaptic structures with a standard desktop computer. To extract bouton-like structures from calcium imaging data, we introduced a new type of modularity, a widely used quality measure in graph theory, and optimized the clustering configuration by a simulated annealing algorithm, which is established in statistical physics. To assess this method's performance, we conducted calcium imaging of the neuropil of *Drosophila* larvae. Based on the obtained data, we established artificial neuropil imaging datasets. We applied the decomposition procedure to the artificial and experimental calcium imaging data and extracted individual bouton-like structures successfully. Based on the extracted spatiotemporal data, we analyzed the network structure of the central nervous system of fly larvae and found it was scale-free. These results demonstrate that neuropil calcium imaging and its decomposition could provide new insight into our understanding of neural processing.

1. Introduction

Calcium imaging of the cell bodies of neurons has provided rich information on the functional dynamics in neural circuits. Functional imaging of neuron populations has revealed the existence of maps in several regions of the central nervous system (CNS), including the visual cortex, auditory cortex, hippocampus, amygdala, and spinal cord (Gauthier & Tank, 2018; Gründemann et al., 2019; Hackett et al., 2011; Tinnermann et al., 2021; Zhuang et al., 2017). The configuration of cell bodies is one of the critical factors for neural circuit computation.

While the calcium imaging of somas has been widely used, the cell body is not the sole component of a neuron. For synapse transduction, axons and dendrites are major structures forming the brain's neuropil. Considering the physiological roles in the neuropil, pan-neuronal calcium imaging of the neuropil would bring us several new types of information on the circuit dynamics (London & Häusser, 2005; Poirazi & Papoutsi, 2020). First, the calcium signal in the neuropil directly reflects the synaptic transmission, which is critical to understanding spatiotemporal dynamics in the nervous system. Second, the activation of one neuron could generate heterogeneous activity among its multiple presynaptic sites (Bilz et al., 2020). This heterogeneity can be detected by neuropil imaging but not by somatic imaging. Third, since the neural wiring diagram is embedded in the neuropil, the activity dynamics in the neuropil can represent the flow of information in the neural circuit (Kohsaka et al., 2019; Kreiner et al., 1987; Lee et al., 2017; Moyle et al., 2021).

Whereas neuropil imaging would be informative in studying circuit dynamism, the data analysis of calcium imaging in the neuropil has been challenging. Because synapses and neurites are densely packed in the nervous system, it is hard to identify synapses and extract their activity from pan-neuronal imaging data. A variety of algorithms to decompose calcium imaging data into cellular activities have been developed based on several mathematical approaches, including principal component analysis, independent component analysis, similarity analysis, singular value decomposition, non-negative matrix factorization, and graph theory (Maruyama et al., 2014; Mölter et al., 2018; Mukamel et al., 2009; Pnevmatikakis et al., 2016; Shibue & Komaki, 2020). However, automatic approaches to decompose the neuropil remain to be developed. Especially most of the decomposition algorithms require that activities of neuronal populations are sufficiently sporadic and

independent so that signal traces of proximate cell bodies can be clearly differentiated. However, synapse population activity patterns in the neuropil are not always sporadic but spatiotemporally coordinated in some cases, such as global bursts and wave-like propagations (Churchland et al., 2012; Lemon et al., 2015). Therefore, the existing algorithms might not be suitable for processing calcium imaging data with spatially coupled activities.

In this study, we propose a new methodology for analyzing neuropil calcium imaging data and an unsupervised algorithm to extract pre-synaptic bouton-like structures and their activity traces. We apply an iteration method to adaptively infer the baseline of the signals in spatial dimensions, which enables the elimination of coarse signals from out of the focal plane and the extraction of well-focused signals. To label each bouton-like component in the pre-processed data, we introduce a new type of modularity measure (Humphries, 2011; Newman & Girvan, 2004) and optimize the clustering configuration by a simulated annealing algorithm (Kirkpatrick et al., 1983; Metropolis et al., 1953), which is established in statistical physics. We apply this procedure to artificial and biological calcium imaging data and successfully extract population activity in the neuropil for motor control showing wave-like activity patterns.

2. Methods

In this study, we developed a novel methodology for analyzing neuropil calcium imaging data requiring low computational resources. We realized this by taking advantage of the observation that each bouton had a compact, round shape. By omitting unnecessary computation between distant pixels, we built an effective decomposing algorithm with high sensitivity.

We used python3.7 combined with C++ and python libraries, NumPy, SciPy, and OpenCV on a standard desktop computer (maximum speed: 3.60 GHz, 4-core CPU, 8 GB RAM) for the calculation.

2.1. Patch-wise modularity-based pixel clustering

To extract bouton-like structures, we defined a patch-wise similarity matrix based on inter-pixel signal similarity, obtained a patch-wise modularity matrix, and solved the optimization problems using the Gibbs sampling.

2.1.1. Similarity matrices

First, we applied pixel clustering based on comparing the temporal profile of pixels' signal intensity. This procedure was devised with the idea that the profile of the calcium signal from the points within the same bouton should be similar. Because boutons were densely packed in the neuropil, it was difficult for the computer to recognize them solely by their morphological features. Accordingly, we utilized their temporal information for clustering. However, it was still difficult to compare the pixels in pairs because the number of the pixel pairs was too large (about 10^{12} pairs) for standard desktop computers. Instead of analyzing every pair, we compared pixel pairs whose distance was within a patch radius r (section 2.1.2). The patch radius was set to 3-6 pixels. By so doing, the number of the comparisons could be small enough to handle with standard desktop computers, and bouton-sized regions (the diameter of about five pixels) were extracted (section 2.1.3).

To define the similarity for clustering, we referred to experimental observation. The calcium signals of pixels within each single bouton changed synchronously and exhibited no apparent time lag among pixels (Fig. C.1.A-C). Therefore, as a temporal similarity between two pixels μ and ν ($\tilde{A}_{\mu\nu}$), we adopted the inner product between the normalized data vectors,

$$\tilde{A}_{\mu\nu} = \frac{\vec{f}_\mu \cdot \vec{f}_\nu}{\|\vec{f}_\mu\| \, \|\vec{f}_\nu\|} \tag{2.1}$$

where $\vec{f}_\mu = \left(f_{\mu 0}, f_{\mu 1}, \dots, f_{\mu t}, \dots, f_{\mu (T-1)}\right)$ (μ: pixel, t: frame) was a vector of signal intensity, and $\|\cdot\|$ indicated the Euclidean norm. In the following analysis, we regarded the normalized inner products as weights in a graph. Although most (>99.99%) of the values were non-negative in our calcium imaging datasets, a few values were negative. Accordingly, if the value was negative, we replaced it with zero. In the end, we defined the similarity matrix $A_{\mu\nu}$ as follows:

$$A_{\mu\nu} = \begin{cases} \tilde{A}_{\mu\nu}, & \tilde{A}_{\mu\nu} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.2}$$

2.1.2. Calculation of patch-wise modularity matrices

By using the similarity matrix, we calculated the modularity matrix $B_{\mu\nu}$ that described a set of clusters of components based on their similarity. Conventionally, the

modularity matrix was defined as the following (Humphries, 2011; Newman, 2006):

$$B_{\mu\nu} \coloneqq A_{\mu\nu} - P_{\mu\nu} \tag{2.3}$$

where $P_{\mu\nu}$ was the expected value of similarity for weight-randomized networks keeping the same degree distribution with $A_{\mu\nu}$ ($\sum_\nu A_{\mu\nu} = \sum_\nu P_{\mu\nu}$). In the randomization step, the similarity value of an edge was supposed to be determined by the node degree at the two ends of the edge. Accordingly, $P_{\mu\nu}$ was described as the product of the selection probabilities of two nodes and the total edge weight. The selection probability of node μ was d_μ/m and the total edge weight was m, where $d_\mu \coloneqq \sum_\nu A_{\mu\nu}, m \coloneqq \sum_{\mu,\nu} A_{\mu\nu} = \sum_\mu d_\mu$. Therefore, $P_{\mu\nu} = d_\mu/m \cdot d_\nu/m \cdot m = d_\mu d_\nu/m$ was derived. From the definition of the modularity matrix (Eq. (2.3)), $B_{\mu\nu}$ indicated the extent of similarity between μ and ν compared with the expectation. With a community configuration $S = \{s_\mu | \mu\text{: pixel}, s_\mu\text{: community id of pixel } \mu\}$, the quality function termed modularity was defined as follows:

$$Q(S) \coloneqq \sum_{\mu,\nu} B_{\mu\nu} \cdot \delta(s_\mu, s_\nu) \tag{2.4}$$

where δ was the Kronecker delta function ($\delta(x, y) = 1$ if $x = y$, and 0 otherwise). Because $B_{\mu\nu}$ was added to the modularity Q if μ and ν were in the same community (therefore, the modularity was upregulated or downregulated according to whether $B_{\mu\nu}$ was positive or negative, respectively), $B_{\mu\nu}$ could be assumed as the interaction intensity between components (attraction if positive and repulsion if negative). Therefore, the optimized community configuration that maximized the modularity divided components into communities most clearly, so that the components were more similar within them and less similar between them.

However, the conventional modularity matrix could not be used for our pixel clustering. First, it was practically impossible to calculate all components of $A_{\mu\nu}$ because the number of pixel combinations was too large to handle. The method needed to hold connectivity strength between every combination of nodes, which in our case involved data for more than two million pixels, a task that was impossible to carry out on a standard computer. Second, modularity-based community detection had problems with the "resolution limit" (Fortunato & Barthélemy, 2007). Since the expected number of links in a random network decreased in a large network, the elements in the modularity matrices tended to be large, and clusters would be merged in a large network. This effect could decrease the spatial

resolution of the clustering and make bouton-like structures merge.

To engineer a modularity-based pixel clustering method, we defined "patch-wise modularity" by modifying the conventional definition (for the detailed definition, see Appendix A). The concept of patch-wise modularity was based on the idea that only spatially close pixels should be compared to recognize bouton-sized regions. At first, we introduced a weight matrix to restrict the similarity calculation to neighborhoods as pixel-centered circle patches:

$$W_{\mu\nu} = \begin{cases} 1, & \mathrm{d}(\mu,\nu) < r \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

where $\mathrm{d}(\mu,\nu)$ was the Euclidean distance between pixels μ and ν, while r was the radius of patches that determined the scale of output pixel clusters. The patch radius r was chosen from 3.0-5.0 pixels, a typical diameter of a single bouton in our calcium imaging data. Instead of $A_{\mu\nu}$, we used $\hat{A}_{\mu\nu} \coloneqq W_{\mu\nu} \cdot A_{\mu\nu}$ as the similarity matrix. Since the number of the non-zero components in $\hat{A}_{\mu\nu}$ was reduced drastically compared with $A_{\mu\nu}$, the calculation and memory costs became to be manageable size (about 1/3,000 times smaller than the original one). Then, we calculated $\hat{P}_{\mu\nu}$, the expected similarity of networks with the same degree distribution ($\sum_\nu \hat{A}_{\mu\nu} = \sum_\nu \hat{P}_{\mu\nu}$). Unlike the conventional $P_{\mu\nu}$, we additionally imposed a localization condition, $\hat{P}_{\mu\nu} = 0$ when $d(\mu,\nu) \geq r$, which was necessary to ensure the modularity within each local region. However, the conventional definition $\hat{P}_{\mu\nu} \coloneqq \hat{d}_\mu \hat{d}_\nu / \hat{m}$, where $\hat{d}_\mu \coloneqq \sum_\nu \hat{A}_{\mu\nu}$ and $\hat{m} \coloneqq \sum_{\mu,\nu} \hat{A}_{\mu\nu} = \sum_\mu \hat{d}_\mu$, could not generally fulfil the localization condition. Therefore, we expanded the conventional definition to meet the localization condition and obtained $\hat{P}_{\mu\nu} \coloneqq W_{\mu\nu} k_\mu k_\nu$, where k_μ was a vector chosen to satisfy $\sum_\nu \hat{A}_{\mu\nu} = \sum_\nu \hat{P}_{\mu\nu}$. For given $W_{\mu\nu}$ and $\hat{A}_{\mu\nu}$, there always existed one and only one k_μ (Theorem 1, Appendix B). Because k_μ could not be described explicitly, we approximated $\hat{P}_{\mu\nu}$ by an iteratively updated matrix according to the following recurrence formula:

$$\hat{P}_{\mu\nu}^{(0)} = W_{\mu\nu}, \qquad \hat{P}_{\mu\nu}^{(n+1)} = \hat{P}_{\mu\nu}^{(n)} \cdot \sqrt{\frac{\hat{d}_\mu}{\hat{d}_\mu^{(n)}} \cdot \frac{\hat{d}_\nu}{\hat{d}_\nu^{(n)}}} \tag{2.6}$$

where $\hat{d}_\mu^{(n)} \coloneqq \sum_\nu \hat{P}_{\mu\nu}^{(n)}$. It was proved that the limit $\lim_{n\to\infty} \hat{P}_{\mu\nu}^{(n)}$ gave $\hat{P}_{\mu\nu} = W_{\mu\nu} k_\mu k_\nu$, where $k_\mu = \lim_{n\to\infty} \prod_{i=1}^{n} \sqrt{\hat{d}_\mu / \hat{d}_\mu^{(i)}}$ satisfying $\sum_\nu \hat{A}_{\mu\nu} = \sum_\nu \hat{P}_{\mu\nu}$ (Theorem 2, Appendix B). Therefore, we put $\hat{P}_{\mu\nu} \cong \hat{P}_{\mu\nu}^{(100)}$ which had sufficiently converged to a specific matrix. Then, the patch-wise

modularity matrix was calculated as follows:

$$\hat{B}_{\mu\nu} \coloneqq \hat{A}_{\mu\nu} - \hat{P}_{\mu\nu} \tag{2.7}$$

For the following optimization step, $\hat{B}_{\mu\nu}$ was divided by the standard deviation of $\hat{B}_{\mu\nu}$ for pairs μ and ν where $W_{\mu\nu} = 1$ to normalize the convergence speed in the optimization process.

2.1.3. Optimization of the patch-wise modularity

Patch-wise modularity $\hat{Q}(S)$ was defined from $\hat{B}_{\mu\nu}$ as an extension of Eq. (2.4), and $\hat{Q}(S)$ was maximized by optimizing the configuration of pixel clusters $S = \{s_\mu\}$:

$$\hat{Q}(S) \coloneqq \sum_{\mu,\nu} \hat{B}_{\mu\nu} \cdot \delta(s_\mu, s_\nu), \qquad S^* \coloneqq \underset{S}{\operatorname{argmax}}\, \hat{Q}(S) \tag{2.8}$$

where $s_\mu \in \mathbb{N}$ ($\mathbb{N}$ denoted natural numbers). The optimized configuration S^* divided pixels into clusters based on the proximity and activity similarity.

To maximize the patch-wise modularity, we adopted simulated annealing (SA) method (Kirkpatrick et al., 1983) (Fig. 1.A), which was computationally efficient, especially when the interaction between elements (pixels in our case) was localized. In SA, the system was developed randomly according to the thermal state distribution with its energy (in our case, the energy $E = -\hat{Q}$) at a specific temperature. As the temperature gradually approached absolute zero, the system finally settled into the minimum energy state (then, $\hat{Q}$ is maximized). The thermal state distribution was produced by iterative system update by the Gibbs sampling. The calculation cost of the Gibbs sampling was drastically reduced if the interaction among elements was localized.

For each iteration during the optimization processing with SA, the cluster identities of pixels s_μ were updated one by one. At first, each pixel was randomly assigned with one of the labels whose number was the number of pixels within a patch of the radius r. This label number was sufficient to assign different clusters with distinct labels locally. Note that since the label number (about 100) was smaller than the total cluster number (about 5,000), each cluster could have multiple connected regions. Each single connected region would be isolated at the later step (see section 2.1.4.) Even when the total label number was set to larger numbers (200 and 400), the results were consistent with the case of 100 (Fig. C.1.D-F). The new s_μ of each pixel was sampled according to the conditional probability

distribution given by the Gibbs sampling, while identities of the other pixels were fixed:

$$\mathrm{P}\left(s_\mu = x \mid s_\nu\ \forall \nu \neq \mu\,;\beta\right) = \begin{cases} \dfrac{e^{\beta \hat{Q}(S)|_{s_\mu = x}}}{\sum_{k \in S_\mu \text{ or } k = s_\mu^*} e^{\beta \hat{Q}(S)|_{s_\mu = k}}} & x \in S_\mu \text{ or } x = s_\mu^* \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

where β was the inverse temperature, $S_\mu \coloneqq \{s_\nu \mid d(\mu,\nu) < r,\ \mu \neq \nu\}$ was cluster identities of pixels within the patch around the pixel μ except for μ itself, and $s_\mu^* \coloneqq \min\{n \mid n \in \mathbb{N}, n \notin S_\mu\}$. At each step, a candidate cluster label s_μ for pixel μ was selected from labels around pixel μ (S_μ) or a label not around it (s_μ^*). If the relabeling increased the pair-wise modularity $\hat{Q}$, the candidate label was accepted, and the clustering configuration was updated with a high probability defined by Eq. (2.9). Because of the localized connectivity, most terms in the score $\hat{Q}$ were independent of new s_μ in Eq. (2.9). Accordingly, the formula of the probability was reduced to the following:

$$\mathrm{P}\left(s_\mu = x \mid s_\nu\ \forall \nu \neq \mu\,;\beta\right) = \begin{cases} \dfrac{e^{\beta \Delta\hat{Q}_\mu(x)}}{\sum_{k \in S_\mu \text{ or } k = s_\mu^*} e^{\beta \Delta\hat{Q}_\mu(k)}} & x \in S_\mu \text{ or } x = s_\mu^* \\ 0 & \text{otherwise} \end{cases} \tag{2.10}$$

where $\Delta\hat{Q}_\mu(x) = \sum_{\substack{\nu \neq \mu \\ \mathrm{d}(\nu,\mu) < r}} \hat{B}_{\mu\nu}\delta(x, s_\nu)$. The number of terms in $\Delta\hat{Q}_\mu$ was highly smaller than that in $\hat{Q}$, so the calculation cost for Eq. (2.10) was reduced from that of Eq. (2.9) by the ratio of $|S_\mu|/N_p^2$ ($|S_\mu|$: the number of elements in S_μ, N_p: the number of pixels). Within one iteration of SA, the identity of every pixel was updated according to the conditional probability distribution (Algorithm 1). The order of pixel update was randomly shuffled at every iteration ($\mathrm{shuffle}(\{1,2,\ldots,N_p\})$). Along with the iterations, the inverse temperature β_i was raised. Because $B_{\mu\nu}$ was normalized by the standard deviation of its elements, $\hat{Q}$ was almost converged when β reached 20 regardless of the data property, including pixel numbers. By comparing arithmetic or geometric progression of β, we found $\hat{Q}$ was maximized more successfully in the geometric case (Fig. C.1.G-I). Therefore, we raised β according to $\beta_i = 0.04 \times 1.004^{i-1}$ ($i = 1,2,\ldots$), where i is the iteration number. The configuration updates were terminated when β_i reached $\beta_{\max} = 20$ (i = 1,557).

Initialize $S = \{s_\mu\}$.

$i = 1$.

while $\beta_i < \beta_{\max}$ **do**

  for μ **in** $\mathrm{shuffle}(\{1,2,\ldots,N_p\})$ **do**

    Sample s_μ^{new} by $\mathrm{P}(s_\mu = x \mid s_\nu\ \forall \nu \neq \mu_1\,;\beta_i)$.

    $s_\mu \leftarrow s_\mu^{\mathrm{new}}$.

  end for

  $i \leftarrow i + 1$.

end while

Algorithm 1. Simulated annealing method.

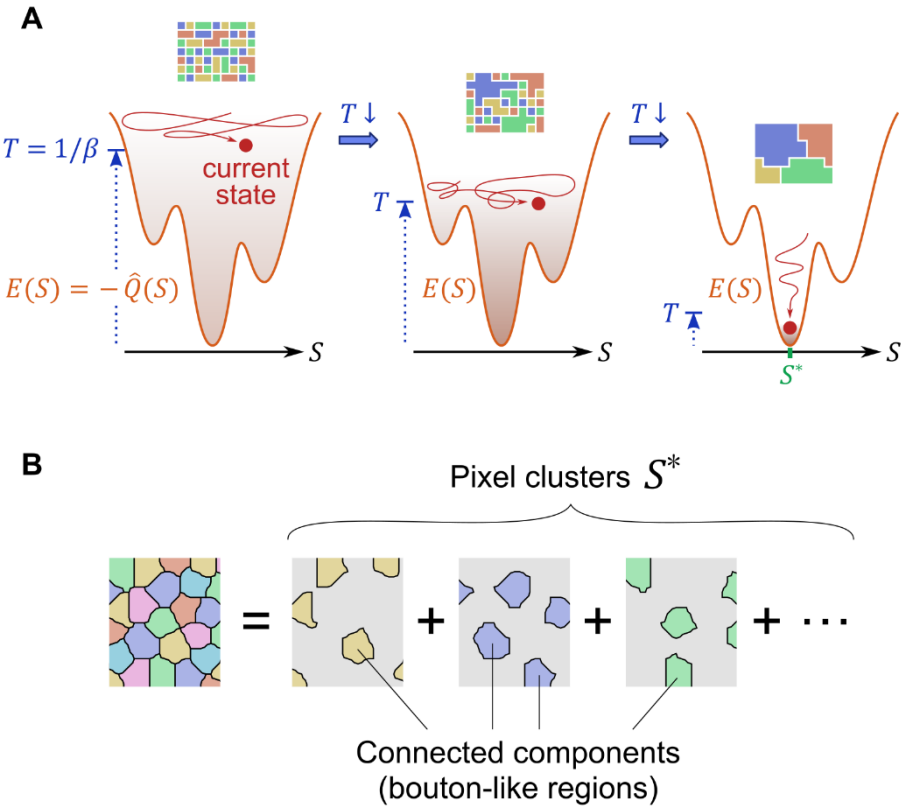

Fig. 1. Optimization and extraction of bouton-like regions. (A) Schematic diagram explaining how simulated annealing works. The bottom three energy landscapes show that the energy is minimized, and the modularity $\hat{Q}$ is maximized by searching the configuration space S as the temperature falls. The top three panels represent that the pixels in the image are

clustered through the optimization process, as shown by enlarging domains of different colors. (B) The source image is decomposed into a set of pixel clusters, S^*. Each pixel cluster consists of a group of connected components, the diameter of which is around the threshold (patch radius r). Bouton-like regions are extracted as connected components in each of the pixel clusters in S^*.

2.1.4. Extraction of bouton-like regions

The optimized configuration S^* was used to define bouton-like regions in the neuropil imaging data. Each pixel cluster in S^* consisted of multiple connected regions (Fig. 1.B). However, since presynaptic boutons are a local and segregated structure, the multiple connected regions in a certain cluster could be regarded as distinct presynaptic regions. Therefore, S^* was further decomposed into spatially connected components $\{R_n \mid n = 1,2,\dots,N_c\}$ (N_c: the number of connected components), and the signals $f_{\mu t}$ in each R_n were averaged ($F_{nt} = \underset{\mu \in R_n}{\text{average}}(f_{\mu t})$).

2.1.5. Selection of active bouton-like regions

Some of the bouton-like regions extracted from the calcium imaging data of the fly larval CNS were located outside of the ventral nerve cord and had almost no calcium signal. To filter out inactive regions from the data, we conducted threshold processing in terms of the S/N ratio of F_{nt}. To evaluate the S/N ratio, the following two quantities were calculated:

$$w_n^2 = \underset{t}{\text{average}}\left[\left(\frac{F_{n(t+1)} - F_{nt}}{2}\right)^2\right], \qquad b_n^2 = \underset{t}{\text{variance}}\left(\frac{F_{n(t+1)} + F_{nt}}{2}\right) \tag{2.11}$$

Here, w_n was the average of the deviation of adjacent data points, and b_n was the deviation of the average of adjacent data points. While w_n measured the fluctuation between the consecutive frames, the fluctuation was averaged out in b_n. Therefore, we regarded w_n and b_n as measures of noise and signal, and assumed the fraction b_n/w_n as an S/N ratio indicator. The larger b_n/w_n was, the better the S/N ratio was (white noise shows $b_n/w_n = 1$). Bouton-like structures were determined by choosing regions with b_n/w_n larger than 5.0 (Fig. C.2.A-C). The pixels in each extracted bouton-like region showed relatively similar calcium signal profiles (Fig. C.2.D). Furthermore, the size of the bouton-like structures was consistent with that of presynaptic boutons measured by immunostaining against

presynaptic marker (Fig. C.2.E). To sum, the extracted regions would correspond to presynaptic structures in the neuropil.

2.2. Data acquisition and pre-processing

2.2.1. Calcium imaging of the neuropil

Calcium imaging data were obtained from the CNS of third instar larvae in *Drosophila melanogaster* (Fig. 2.A). The genotype was *UAS-CD4::GCaMP6f; nSyb-Gal4*, which expresses a membrane-bound form calcium sensor (CD4::GCaMP6f) in all neurons (*nSyb-Gal4*: Bloomington #58763, *UAS-CD4::GCaMP6f*: (Kohsaka et al., 2019)).

Calcium imaging of the CNS of fly larvae was carried out as described previously (Kohsaka et al., 2014, 2019) (Fig. 2.B). In brief, third-instar larvae of the *UAS-CD4::GCaMP6f; nSyb-Gal4* fly were dissected with microscissors, and the CNS was isolated. The isolated CNS was mounted on an adhesive slide glass (MAS coated slide glass S9215, Matsunami Glass, Japan) and soaked in an insect saline, TES buffer (TES 5 mM, NaCl 135 mM, KCl 5 mM, CaCl2 2 mM, MgCl2 4 mM, sucrose 36 mM; pH = 7.15). The fluorescence signal from the CNS was recorded by an EMCCD camera (iXon+ DU-897E-CS0-#BV, Andor, UK) (63x, 0.27 μm x 0.27 μm /px, 330 ms/frame, EM gain: 144, data depth: 14 bit) with attached spinning disk confocal unit (CSU21, YOKOGAWA, Japan). The CNS preparation was illuminated by a blue laser (power: 300 μW ~ 500 μW; the wavelength: 488 nm; CSU-LS2WF, Solution Systems, Japan) through a water immersion objective lens (ACROPLAN 63X, Zeiss, Germany). To align the larval ventral nerve cord (VNC) on the focal plane, the slide glass was put on an adjustable slope by moving a fulcrum. The recording duration was 233 seconds (700 frames). To suppress the noises, we applied smoothing processing with the method described in (Eilers, 2003), using a second-order penalty and $\lambda = 1.5$. We clipped a region of 100x100 pixels from one sample for the application of our patch-wise modularity-based clustering and non-negative matrix factorization (section 2.4) (Fig. 7.C).

In the following sections, imaging data of each focal plane is described by l_{txy} $\left(l_{txy} = 0,1,\ldots,2^{14}-1\right)$, where t is the frame number and x, y are pixel coordinates ($t = 0,1,\ldots,T-1,\ \ x,y = 0,1,\ldots,D-1,\ \ T = 700\ \text{frames},\ \ D = 512\ \text{px}$).

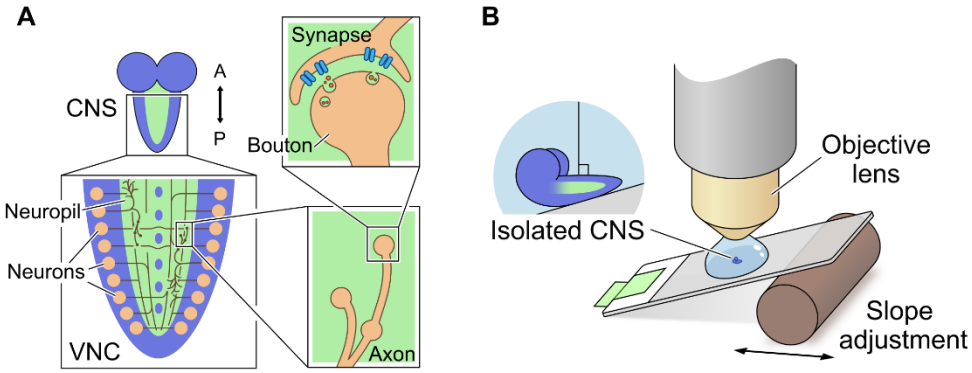

Fig. 2. Calcium imaging of the VNC neuropil in an isolated CNS. (A) Schematics of the CNS, the VNC, the neuropil, and boutons in the *Drosophila* larvae. (B) Schematic diagram illustrating the setup for calcium imaging of the isolated CNS.

2.2.2. Baseline correction and elimination of deformation and drift

The signal intensity at each pixel could be changed by two main factors: change in fluorescence intensity according to sensing calcium ions by the probe, and experimental artifacts, including bleach of the fluorophore of the sensors and drift of the sample. We needed to extract the former from the calcium imaging data, and the latter should be removed. Here we assumed that the artifacts changed slowly, and we could treat them as a change in the baseline of the signal. Therefore, the signal intensity at each pixel of the fluorescence movie was assumed to be the product of the baseline and the ratiometric change in probe signal intensity as follows:

$$l_{txy} = b_{txy} \times s_{txy} \tag{2.12}$$

where b_{txy} was the baseline, and s_{txy} was signal-to-baseline ratio.

To suppress the effect of temporal change of the baseline, we extracted the component of b_{txy} by applying baseline detection processing as described in (F. Zhang et al., 2020) with $\lambda = 2 \times 10^5$.

The extracted b_{txy} was a fluorescence distribution without calcium transient (Fig. C.3.A). Therefore, b_{txy} was first used to correct the sample deformation and drift. Each frame of b_{txy} was transformed to match the specific frame (b_{t_0xy}, $t_0 = T/2$; T = the total frame number) with a rigid transformation (homography), using cv2 function

"findTransformECC" (Fig. C.3.B). Then, the baseline detection was applied again to the spatially transformed imaging data ($x, y \to x', y'$) and new baselines $b_{tx'y'}$ were calculated (Fig. C.3.C). Finally, the temporal profiles of calcium signals without baseline change were extracted (Fig. C.3.D). The pure neuronal signal $s_{tx'y'}$ was calculated by rearranging the Eq. (2.12):

$$s_{tx'y'} = \frac{l_{tx'y'}}{b_{tx'y'}} \tag{2.13}$$

The primes of the spatial variables are omitted in the following sections: s_{txy}.

2.2.3. Extraction of signals from focal planes

The data s_{txy} obtained above contained blurred signals from activities out of focus because the focal planes were placed in a three-dimensional object filled with calcium probes (Fig. 3.A). These blurred and coarse signals made it difficult to extract individual bouton activities. We formulated s_{txy} as a summation of fine (well-focused) and coarse (out-of-focus) components:

$$s_{txy} = f_{txy} + c_{txy} \tag{2.14}$$

where f and c were fine and coarse components, respectively.

The data s_{txy} was decomposed into f_{txy} and c_{txy} based on the difference in the spatial scale of intensity change. In each data frame, the coarse component looked like a gently changing baseline, and narrow peaks of the well-focused component were mounted on that baseline (Fig. 3.B). Therefore, like the decomposition of l_{txy} into s_{txy} and b_{txy} in section 2.2.2, the spatial structure of f_{txy} and c_{txy} could be segregated by utilizing the scale difference of intensity change (Fig. C.4.A). In this study, we applied two-dimensional baseline detection to s_{txy}, using Gaussian blur. Gaussian blur is defined as follows:

$$G(x, y; \sigma) \coloneqq e^{-(x^2+y^2)/2\sigma^2}$$

$$\mathcal{G}(s_{txy}; \sigma) \coloneqq \sum_{x',y'=0}^{D-1} s_{tx'y'} \cdot G(x - x', y - y'; \sigma) \Big/ \sum_{x',y'=0}^{D-1} G(x - x', y - y'; \sigma) \tag{2.15}$$

At first, Gaussian blur was applied to the extracted signals s_{txy} ($\sigma_1 = 1$) to reduce detailed fluctuation (Fig. 3.C.i):

$$s'_{txy} \coloneqq \mathcal{G}(s_{txy}; \sigma_1)$$

Then, the coarse component that corresponded to the baseline was calculated by

the recurrence formula of $c_{txy}^{(n)}$:

$$c_{txy}^{(0)} = \mathcal{G}(s'_{txy}; \sigma_2), \qquad c_{txy}^{(n+1)} = c_{txy}^{(n)} - \mathcal{G}\left(\max\left\{0,\ c_{txy}^{(n)} - s'_{txy}\right\};\ \sigma_2\right) \tag{2.16}$$

In this formula, we took advantage of the fact that when we set the standard deviation of the Gaussian filter between the scales of the coarse component and fine component, the coarse component was robust to the Gaussian filter while the fine component was sensitive. First, $c_{txy}^{(0)}$ was obtained by applying the Gaussian filter to the original image s'_{txy} (Fig. 3.C.ii). Since the intensity of the signals was larger than that of the background, $c_{txy}^{(0)}$ became smaller than s'_{txy} in the pixels that had significant signals from the well-focused plane (object region, hereafter. See horizontal bars in Fig. 3.C), but larger than s'_{txy} in the other region (non-object region, hereafter). Therefore, the sign of $c_{txy}^{(0)} - s'_{txy}$ could be used to segregate the object region from the non-object region. By applying the Gaussian filter to the region where $c_{txy}^{(0)} - s'_{txy}$ was positive, we could obtain the coarse component of the difference between $c_{txy}^{(0)}$ and s'_{txy} in the non-object region. By subtracting the coarse component of the difference from $c_{txy}^{(0)}$ itself, a new coarse component $c_{txy}^{(1)}$ that had less difference from the original image s'_{txy} in the non-object region but didn't have a fine component was obtained (Fig. 3.C.iii). By repeating these steps, $c_{txy}^{(n)}$ would converge to the curve that almost fit the original image s'_{txy} in the non-object region but didn't have a fine component. The well-converged curve $c_{txy}^{(5000)}$ was treated as the baseline, and $f_{txy} = s_{txy} - c_{txy}^{(5000)}$ was used as signals from the focal plane.

The significance of this decomposition was observed in the comparison between activity waves spontaneously propagating in forward or backward directions on the VNC (Fig. C.4.B). It has been shown that several interneurons are activated in either forward or backward waves (Carreira-Rosario et al., 2018; Fushiki et al., 2016; Hiramoto et al., 2021; Kohsaka et al., 2019). In our data of pan-neuronal calcium recordings in the neuropil, such direction-specific activity could be observed only in the fine component (Fig. C.4.B). Therefore, the decomposition step was critical to extracting the proper activity of the neurons.

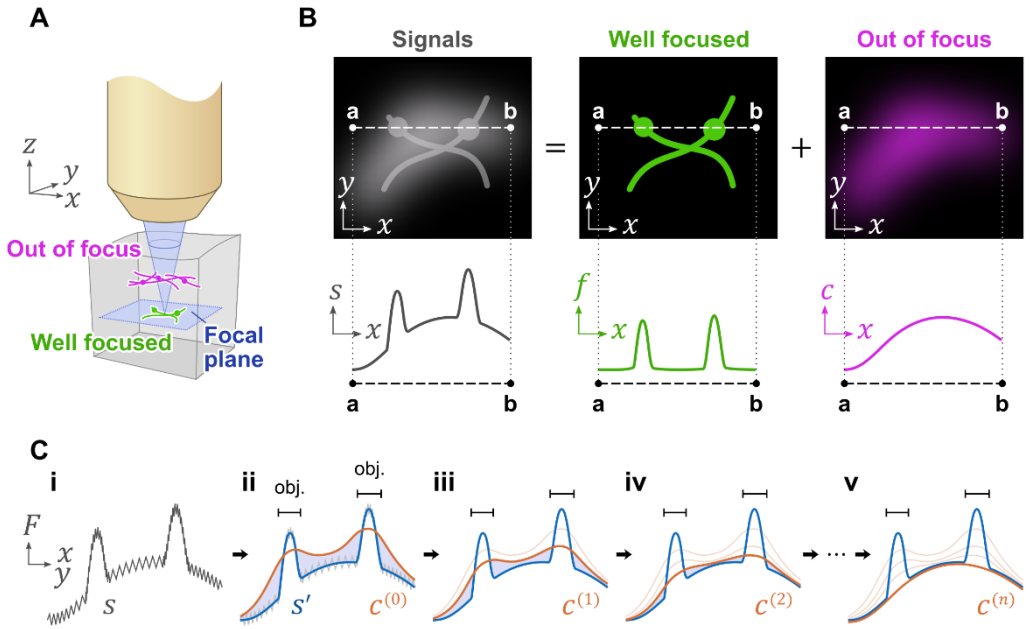

Fig. 3. Baseline detection based on image contrast. (A) Schematics show that fluorescence signals contain fluorescence from both the focal plane and planes out of focus. (B) (Top): The detected signals (left) are the summation of signals from the focal plane (middle) and from out of the focal plane (right). (Bottom) Plot of signal intensities between points a and b in the corresponding top images. Note that the contrast of the signals in the well-focused image is high while those out of focus are low. (C) Principle of the two-dimensional baseline detection. The baseline is extracted by repeated calculation of a recurrence formula. The gray line in panel i is the original signal. Blue lines in panels ii to v indicate the smoothed signals, and orange lines show the candidates of its baselines. The signals from the objects on the focal plane (obj.) are extracted as the difference between the orange and blue lines by iteration. See section 2.2.3 for details.

2.3. Artificial data of calcium signals

Artificial spatiotemporal data simulating calcium imaging were generated to test the performance of clustering methods. The dataset was made in three steps: building a 3-dimensional neuropil model with boutons, creating a calcium signal for each of the boutons in the 3-dimensional neuropil, and generating spatiotemporal signals in a 2-dimensional section from the 3-dimensional while adding realistic noises and perturbations.

2.3.1. Building a 3-dimensional neuropil model with boutons

A group of boutons were built by splitting a 3-dimensional array of voxels (50 x 50 x 50 voxels) using the k-means algorithm with a fixed number of boutons. The x-y plane where $z = 25$ was treated as the focal plane in our artificial data (see section 2.3.3.) The total bouton number on the focal plane was set by tuning the total bouton numbers in the whole volume. The total bouton number in the volume was set to around 225, 450, 900 or 1,800 to let the bouton number on the focal plane be 25, 50, 100, or 200, respectively. First, each voxel was randomly assigned to one of the boutons. Then, the bouton assignment for all the voxels was updated five times by the k-means algorithm based on the Euclidean distance between voxels. In the end, each voxel was assigned to one bouton, and the whole 3-dimensional array was split into the pre-fixed number of boutons

2.3.2. Creating calcium signals in the 3-dimensional neuropil model

Next, we generated a temporal signal to be assigned to each of the boutons. The length of the signals was 100, 200, 400, 800, and 1600 frames, and each of the temporal signals included multiple peaks. Each peak was modelled by a Gaussian function with time as a variable. By changing the temporal differences between the peaks of different bouton signals, two activity patterns were created: burst-like and sporadic activity. The former simulated a cooperative firing pattern seen in wave propagation, whereas the latter exhibited a less cooperative activity pattern. To simulate a peak, the time, width, and height of the peak were required. The time of the peaks in the signals was sampled from a Gaussian distribution, and the two different patterns (burst-like and sporadic activity) were made by setting a distinct deviation for the timing of signal peaks among the boutons. For the signal data of 100 frames, we set the mean of the peak times to be at the frames of 20, 55, and 80, and the standard

deviation of them to be 2 frames for the burst-like activity and 10 for the sporadic activity. For the longer signal data of 200, 400, 800, and 1600 frames, the same samplings were repeated the required number of times (for example, 16 times for the data of 1600 frames) and concatenated in series. The width and height of the peaks were sampled from the Gaussian distribution (mean = 4 frames and standard deviation = 0.5 frames for the burst-like activity; mean = 2 frames and standard deviation = 0.5 frames for the sporadic activity) and the gamma distribution (shape index = 3 and scale index = 1/3), respectively. We set the baseline of the signal to be the same as the mean value of the gamma distribution 1.

2.3.3. Simulation of realistic 2-dimensional imaging data

To simulate realistic calcium imaging data, we added several perturbations to the spatiotemporal signals generated above, mimicking the variation in the probe expression level, photobleaching, mixing of fluorescence signals from out of the focal plane, specimen drift and deformation, and random noises.

At first, to simulate the variation of the expression level of calcium probes in boutons, we sampled a relative probe level b_n for the n-th bouton from the Gaussian distribution (mean = 2, standard deviation = 0.3). By multiplying the calcium signals generated above by the relative probe level b_n, the signal considering the expression variation a'_{tn} could be described as follows:

$$a'_{tn} = a_{tn} \cdot b_n$$

where a_{tn} was the artificial signal of the n-th bouton at frame t ($t = 0, 1, \ldots, T-1$; T = the total frame number).

Next, the photobleaching during the imaging was modelled by multiplying all the signals by the exponential decay factor:

$$a''_{tn} = a'_{tn} \cdot \exp(-t/\tau)$$

where $\tau = 1{,}000$ frames was the decay time.

To mimic the mixing of fluorescence signals among different focal planes, blurred images out of the focal plane were superimposed on the image of the focal plane ($z_{\mathrm{focus}} = 25$). Gaussian blur was applied to each image of the focal planes ($z = 1, \ldots, 50$) in the 3-dimensional array of voxels (50 x 50 x 50 voxels). The sigma for the Gaussian blur, which determined the size of the blurring filter, was obtained by $\sigma_z = (|z - z_{\mathrm{focus}}| + 1) \cdot \tan\alpha$, where

z was the position of the plane along the z-axis ($z = 0, \ldots, 49$), $z_{\text{focus}} = 25$, $\sin\alpha = \text{NA}/n_{\text{water}}$, $\text{NA} = 0.9$, and $n_{\text{water}} = 1.333$. Then, the blurred images, including that of the focal plane, were averaged by the weight of $1/\sigma_z^2$:

$$a'''_{txy} = \underset{z}{\text{average}}\left(\mathcal{G}\left(a''_{txyz}; \sigma_z\right); \frac{1}{\sigma_z^2}\right)$$

where $a''_{txyz} = a''_{tn}$ when $(x, y, z) \in n\text{-th bouton}$, and $\underset{i}{\text{average}}(x_i; w_i) \coloneqq \sum_i w_i x_i / \sum_i w_i$.

The specimen drift and deformation were simulated by applying homography transformation to the image of the focal plane. The transformation could be uniquely determined by setting the destination of four corners of the image ($(x, y) = (0, 0)$, $(L, 0)$, $(0, L)$, (L, L)), where $L = 50$. The transformation mapped the corners onto $\left(d(t), d(t)\right)$, $\left(d(t), L - d(t)\right)$, $\left(L + d(t), -d(t)\right)$, $\left(L + d(t), L + d(t)\right)$, where $d(t) = 3(2t/T - 1)$ at frame t. The other points were mapped by interpolation. The signal intensity at each pixel in the transformed data was calculated by interpolation. This definition gave a drift in the x-direction, contraction of the side $x = 0$, and extension of the side $x = L$.

Finally, we added random noise sampled from a Gaussian distribution to each pixel at each frame of the data independently. The mean of the Gaussian distribution was zero, and the standard deviation σ_{noise} was chosen from 0.1-1.5 to change noise intensity. The noise level was comparable to that in biological calcium imaging data (Fig. C.5). The spatiotemporal data generated by these steps were used as the artificial imaging data. The x-y plane image where $z = 25$ in the data generated in section 2.3.1 was regarded as the correct configuration of boutons. The signal data assigned to each bouton built in section 2.3.2 was treated as the correct calcium signal. The performance of the clustering algorithms was assessed by comparing the outcome of the algorithms and the correct configuration or calcium signals.

2.4. Data analysis using CaImAn

Our algorithm was compared with CaImAn, an open-source tool for calcium imaging data analysis (version 1.9.9) (Giovannucci et al., 2019). Following the documentation of CaImAn, the analysis pipeline was built by modifying parameters in a released demo pipeline for one-photon imaging data decomposition ("demo_pipeline_cnmfE.ipynb", See Table 1 for the parameter modification.) CaImAn was originally developed to extract cell bodies from

calcium imaging data. To apply it to our bouton imaging data, the bouton size was used as the parameter of neuron size in CaImAn.

In the CaImAn pipeline, piece-wise rigid motion correction and spatiotemporal deconvolution based on a constrained non-negative matrix factorization (CNMF) (Pnevmatikakis et al., 2016) algorithm were performed. CNMF detected each bouton (spatial component) and extracted its activity (temporal component). The spatial components from CNMF described a bouton as a 32-bit float image. To compare the output from CNMF with the one from our algorithm, the 32-bit images were binarized by Otsu's thresholding method (Otsu, 1979). The CaImAn pipeline had an option to use a deconvolution algorithm to estimate spike trains under the assumption that spike trains were sparse. However, since the activity patterns in the fly larval motor circuits were not sparse but burst-like, the deconvolution algorithm was not applied.

Parameter	**default**	**modified**
fr	10	3
gSig_filt	(3, 3)	(5, 5)
p	1	0
gSig	(3, 3)	(s, s), where s was chosen from 1.8, 2.5, 3.5, and 4.9, corresponding to the bouton radius in pixels.
rf	40	20
stride	20	10
tsub	2	1
nb	0	-2

Table 1. Parameter modification in the CaImAn demo pipeline.

2.5. Evaluation of clustering performance

The performance of clustering methods was assessed by evaluating spatial and temporal accuracy in signal detection. The spatial accuracy was measured by using the Dice coefficient. Let R_i^* be the set of pixels included in the i-th true bouton, and R_j be the set of pixels included in the j-th detected bouton-like region. The Dice coefficient between R_i^* and R_j is defined as follows:

$$\mathrm{DC}\left(R_i^*, R_j\right) \coloneqq 2\frac{\left|R_i^* \cap R_j\right|}{\left|R_i^*\right| + \left|R_j\right|}$$

Where $|\cdot|$ indicates the number of pixels included in the set. $\mathrm{DC}\left(R_i^*, R_j\right)$ indicates similarity of R_i^* and R_j, taking a value between 0 and 1. To measure the spatial accuracy of the bouton detection, we defined spatial similarity (A_s) by a weighted average of maximum Dice coefficient with the weight of the pixel number in boutons:

$$(\text{Spatial accuracy}) = A_s(\boldsymbol{R}^*, \boldsymbol{R}) \tag{2.17}$$

$$A_s(\boldsymbol{R}^*, \boldsymbol{R}) \coloneqq \begin{bmatrix} \underset{i}{\text{average}}\left(\underset{j}{\max}\left(\mathrm{DC}\left(R_i^*, R_j\right)\right); \left|R_i^*\right|\right) \\ + \underset{i}{\text{average}}\left(\underset{j}{\max}\left(\mathrm{DC}\left(R_i, R_j^*\right)\right); \left|R_i\right|\right) \end{bmatrix} \Big/ 2.$$

Note that $A_s\left(\boldsymbol{R}^{(1)}, \boldsymbol{R}^{(2)}\right) = A_s\left(\boldsymbol{R}^{(2)}, \boldsymbol{R}^{(1)}\right)$ and $0 \le A_s \le 1$. $A_s\left(\boldsymbol{R}^{(1)}, \boldsymbol{R}^{(2)}\right) = 1$ if and only if $\boldsymbol{R}^{(1)}$ and $\boldsymbol{R}^{(2)}$ were identical. The weight of the average ($\left|R_i^*\right|$ or $\left|R_i\right|$) were imposed because the Dice coefficient was not stable when either region was small.

To measure the temporal accuracy of activity extraction, we defined temporal similarity (A_t) between the true signal of the i-th true bouton F_{it}^* and the extracted signal of the j-th detected bouton F_{jt}, which had the maximum Dice coefficient with the i-th true bouton. The Pearson correlation coefficient between F_{it}^* and F_{jt} in terms of t was calculated. The average of the correlation coefficient over all the true boutons was used as the temporal accuracy:

$$(\text{Temporal accuracy}) = A_t\left((\boldsymbol{R}^*, \boldsymbol{F}^*), (\boldsymbol{R}, \boldsymbol{F})\right)$$

$$A_t\left((\boldsymbol{R}^*, \boldsymbol{F}^*), (\boldsymbol{R}, \boldsymbol{F})\right) \coloneqq \begin{bmatrix} \underset{i}{\text{average}}\left(\underset{t}{\text{correlation}}\left(F_{it}^*, F_{j(R_i^*, \boldsymbol{R})t}\right); \left|R_i^*\right|\right) \\ + \underset{i}{\text{average}}\left(\underset{t}{\text{correlation}}\left(F_{it}, F_{j(R_i, \boldsymbol{R}^*)t}^*\right); \left|R_i\right|\right) \end{bmatrix} \Big/ 2$$

where $\underset{t}{\text{correlation}}(x_t, y_t)$ is the Pearson correlation coefficient between x_t and y_t in terms of t, F_{it}^* was the true signal of the i-th true bouton, F_{jt} was extracted signal of the j-th detected bouton and $j\left(R_i^{(1)}, \boldsymbol{R}^{(2)}\right) \coloneqq \underset{j'}{\operatorname{argmax}}\left(\mathrm{DC}\left(R_i^{(1)}, R_{j'}^{(2)}\right)\right)$. Note that $A_t\left(\left(\boldsymbol{R}^{(1)}, \boldsymbol{F}^{(1)}\right), \left(\boldsymbol{R}^{(2)}, \boldsymbol{F}^{(2)}\right)\right) = A_t\left(\left(\boldsymbol{R}^{(2)}, \boldsymbol{F}^{(2)}\right), \left(\boldsymbol{R}^{(1)}, \boldsymbol{F}^{(1)}\right)\right)$ and $-1 \le A_t \le 1$. $A_t\left(\left(\boldsymbol{R}^{(1)}, \boldsymbol{F}^{(1)}\right), \left(\boldsymbol{R}^{(2)}, \boldsymbol{F}^{(2)}\right)\right) = 1$ if and only if $\left(\boldsymbol{R}^{(1)}, \boldsymbol{F}^{(1)}\right)$ and $\left(\boldsymbol{R}^{(2)}, \boldsymbol{F}^{(2)}\right)$ were identical except for scaling and offset of $\boldsymbol{F}$.

Our algorithm used random processes to optimize the configuration of pixel clusters.

Therefore, the results could differ between data processing trials with the same data. To test the consistency of the signal extractions among trials, we calculated a consistency measure defined as follows:

$$(\text{Consistency}) = \underset{\substack{n,m \\ n<m}}{\text{average}}\left(A_s\left(\boldsymbol{R}^{(n)}, \boldsymbol{R}^{(m)}\right)\right) \tag{2.18}$$

where $\boldsymbol{R}^{(n)} = \left\{R_i^{(n)}\right\}$ was the set of detected bouton-like regions from the n-th trial. The consistency from ten trials for each artificial data of distinct parameters was calculated (Fig. 5).

2.6. Evaluation of calculation time

To measure calculation duration, we applied the bouton detection to artificial data with frame numbers 100, 200, 300 and 400, while the pixel number was fixed to 2,500 (50 x 50). (Fig. 9.A-D). In addition, the calculations were applied to artificial data with pixel numbers 1,024 (32 x 32), 2,500 (50 x 50), 3,969 (63 x 63), 5,476 (74 x 74) and 7,056 (84 x 84), while the numbers of the true boutons were proportional to the pixel numbers (41, 100, 159, 219 and 282), and the frame number was fixed to 100 (Fig. 9.E-H). The artificial data was generated as the burst-like signals with σ_{noise} = 0.1.

3. Results

3.1. Performance assessment of the Patch-wise modularity-based clustering with artificial data

We established a novel method, Patch-wise modularity-based clustering ("PQ-based clustering" hereafter), to cluster bouton-like structures from calcium imaging data (See section 2.1 for details.). In short, to cluster pixels in a calcium imaging movie into a set of boutons, a similarity between a pair of pixels was defined based on their activity patterns. Under the assumption that each bouton was spatially compact, the similarity was calculated between pixels within a pre-determined distance to obtain a patch-wise similarity matrix. Then, a set of boutons in the image were extracted by optimizing the clustering of pixels to let the pixels within each cluster have high similarity. To assess the performance of the PQ-based clustering, spatial accuracy (the bouton configuration) and temporal accuracy (the

calcium signal) were analyzed (See section 2.5.)

To this aim, we generated artificial calcium imaging data. The artificial data were designed referring to the actual calcium imaging data from the neuropil of the larval motor circuits. The neuropil consisted of densely packed boutons and exhibited propagation activity, a progression of local burst-like activity along the body axis. To model these features, an array of voxels (50 x 50 x 50) was divided into hundreds of boutons (Fig. 4.A), and cooperative burst-like activity was assigned to the boutons (Fig. 4.B) (See sections 2.3.1 and 2.3.2 for details.) In addition, several perturbations were added to the artificial data to mimic real calcium imaging data, considering the variation in the probe expression level, photobleaching, mixing of fluorescence signals from out of the focal planes, specimen drift and deformation, and random noises (Fig. 4.A' and B') (See section 2.3.3 for details.)

3.1.1. Performance of the PQ-based clustering method

We applied PQ-based clustering to the artificial data to test the signal extraction performance. From the data of the burst-like activity with a moderate noise (σ_{noise} = 0.1), the PQ-based clustering method extracted most of the boutons (Fig. 4.C). The signal from the extracted boutons was consistent with its correct calcium signal (Fig. 4.D). It should be noted that some small boutons that were less than 10% of the average size could not be detected (arrowheads in Fig. 4.A). And when two adjacent boutons had similar temporal activities, they were extracted as a single bouton (compare stars in Fig. 4.A and C). Although some pixels were mislabeled (< 30%), the PQ-based clustering extracted the majority of the boutons from the artificial dataset. Even when the noise level in the artificial data was increased up to σ_{noise} = 0.5, the majority of the boutons and consistent calcium signals were still extracted (Fig. 4.E and F). If the noise level increased to $\sigma_{\mathrm{noise}} = 1$, which corresponded to the condition when the noise intensity was comparable to that of calcium signals, PQ-based clustering could detect only a half of the boutons (Fig. 4.G) but still with high temporal accuracy (Fig. 4.H). Accordingly, PQ-based clustering could extract the calcium signals from boutons in the artificial dataset mimicking real calcium imaging data. In particular, the spatial accuracy was high when the noise level was lower than the signal intensity.

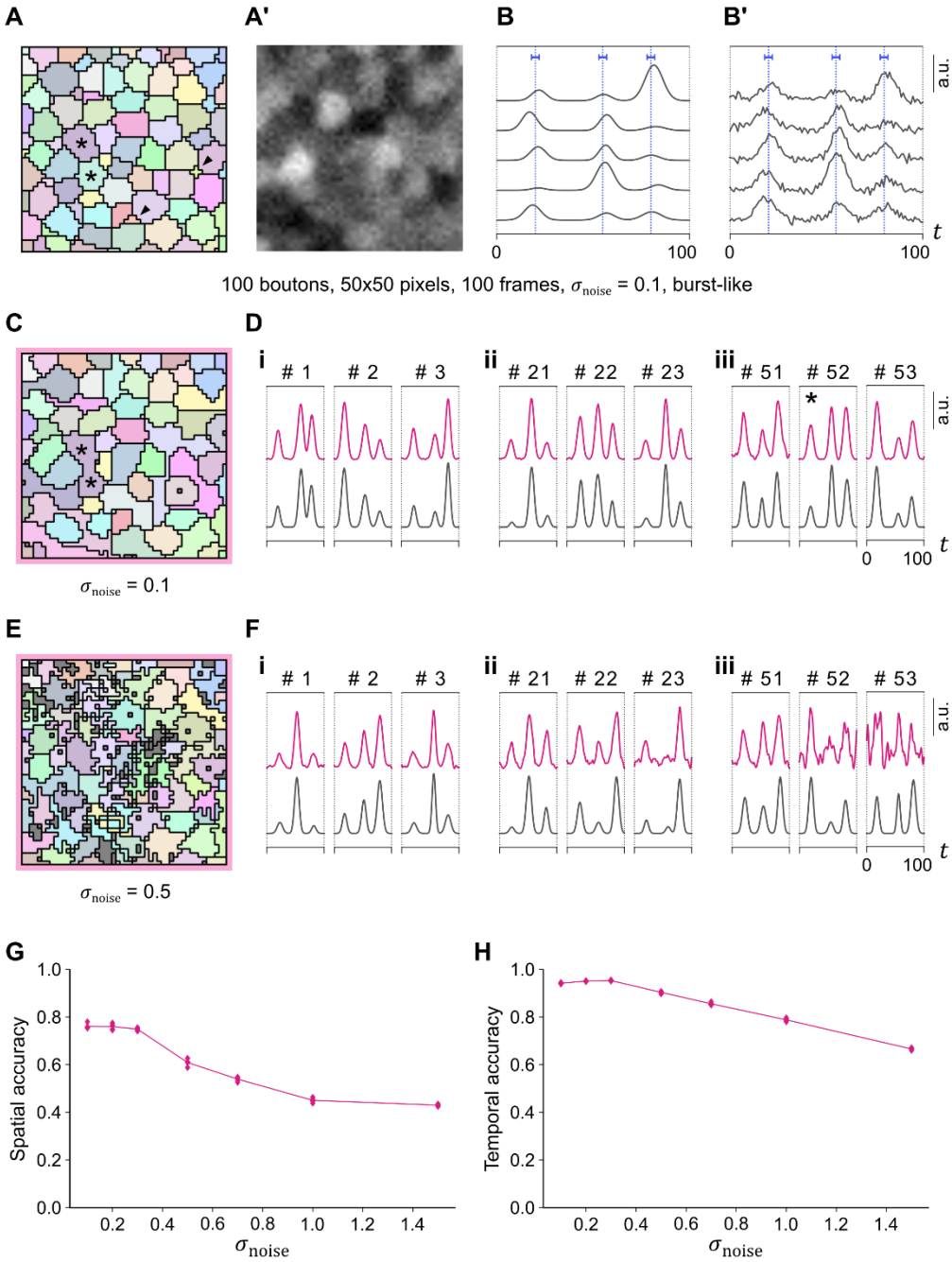

Fig. 4. PQ-based clustering successfully detected boutons and calcium signals from burst-like artificial data. (A) Correct configuration of boutons in the artificial data. One hundred boutons were included in the 50 x 50 pixel array. Different colors indicate each bouton. Arrowheads indicate small boutons that the clustering could not recognize. (A') Signal intensity in one frame in the burst-like artificial data. (B) Correct signals of five example pixels.

(B') Several perturbations were added to the correct signals to achieve realistic artificial data. The traces show the signal in five pixels selected in (B) with the perturbations under the condition of σ_{noise} = 0.1. Vertical blue lines in (B) and (B') indicate the center of each burst-like event, and horizontal blue bars in (B) and (B') indicate the standard deviation of peak timings around each center. (C-F) Bouton extraction from the burst-like artificial data with σ_{noise} = 0.1 (C and D) and σ_{noise} = 0.5 (E and F). (C and E) Detected bouton configuration by PQ-based clustering. The color of each cluster corresponds to that of its closest correct bouton in (A). Stars in (A) and (C) indicate boutons that the clustering could not discriminate. (D and F) Extracted signals (magenta traces) by PQ-based clustering from the data shown in (C) and (E), respectively, and the correct signals (gray traces). Pairs of correct and detected boutons with large Dice coefficient (i: top 1-3, ii: top 21-23, iii: top 51-53) are shown. The pair of the 52nd Dice coefficient (star) is the merged boutons indicated by stars in (C). (G and H) Spatial (G) and temporal (H) accuracy of PQ-based clustering of the burst-like artificial data of each noise intensity.

3.1.2. The adjustability of extracted bouton size in the PQ-based clustering method

In the PQ-based clustering, the patch radius r, which determined the range of distance between pixels to calculate the similarity, was a key hyper-parameter and could influence the clustering performance. This effect was tested by analyzing the results of the PQ-based clustering with a distinct patch radius (r = 3, 4, and 5 pixels). When the data whose mean bouton radius was 1.8 pixels, the PQ-based clustering with a patch radius of 3 gave the highest spatial accuracy. On the other hand, for the data with an average bouton radius of 4.9, the PQ-based analysis with r = 5 provided the best spatial accuracy (Fig. 5.A). Temporal accuracy was not significantly affected by the difference in the patch radius (Fig. 5.B). Because patch radius r determined the range of pixel pairs to be calculated, the calculation time of the clustering was increased by the upregulation of r. However, the increased calculation time was maintained at a manageable level (Fig. 5.C). Therefore, the patch radius could adjust the spatial scale in the PQ-based clustering. In the following analyses, the artificial data with an average bouton size of 2.5 pixels and the patch radius r = 4 were used unless otherwise noted

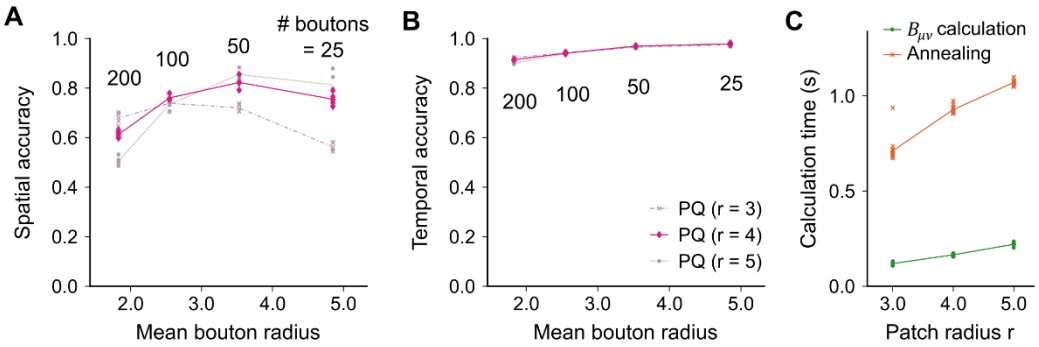

Fig. 5. Detection accuracy could be optimized by tuning the patch radius r. (A and B) Spatial (A) and temporal (B) accuracy of PQ-based clustering with a different patch radius, r = 3 (gray crosses), 4 (magenta diamonds), and 5 (gray circles). (C) Calculation time for PQ-based clustering with a different patch radius r. Green dots and orange crosses indicate the calculation time for $B_{\mu\nu}$ and simulated annealing, respectively.

3.1.3. Consistency of the PQ-based clustering method

The optimization step in the PQ-based clustering contained random processes (see section 2.1.3.) Therefore, distinct trials could give different results even if the data was identical. To check the consistency of the outcome of the PQ-based clustering, we compared the results among multiple trials with the same artificial imaging data (For the measure of consistency, see section 2.5.) The consistency was about 0.8 or more for every case and kept at about 0.9 if the noise level was low (Fig. 6.A and B). Accordingly, even though random processes were included, the PQ-based clustering method could provide consistent results over multiple trials.

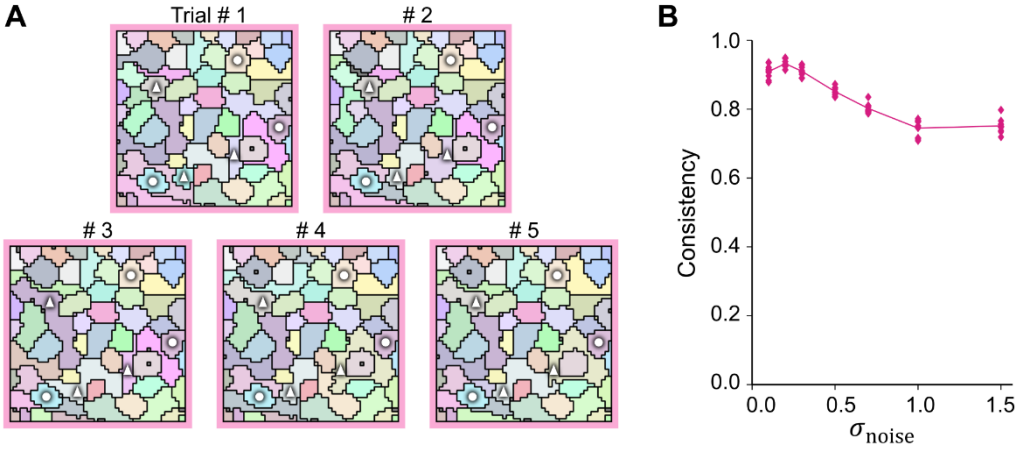

Fig. 6. PQ-based clustering provided consistent results over multiple trials. (A) Detected bouton configuration of five trials from the burst-like artificial data with σ_{noise} = 0.1. The color of each cluster corresponds to that of its closest correct bouton shown in Fig. 4.A. White circles and triangles mark the same clusters among the five trials. The circles indicate regions with relatively high spatial consistency, while the triangles indicate regions with relatively low spatial consistency. (B) Spatial consistency was quantified as the average spatial similarity among five trials (see section 2.5 for details.) Each data point indicates the spatial similarity of a pair of trials.

3.2. Performance comparison with the CaImAn algorithm

We next compared the PQ-based algorithm with CaImAn (Giovannucci et al., 2019), one of the standard tools to extract cell bodies from calcium imaging data (see section 2.4 for details.)

3.2.1. Detection accuracy in the analysis of the burst-like activity data

The CaImAn algorithm detected cell bodies under the assumption that neurons exhibited sparse firing activity but not cooperative burst-like activity. We tested the CaImAn algorithm's performance to analyse our artificial data with burst-like activity. From the data with moderate noises (σ_{noise} = 0.1), CaImAn could detect most correct boutons (Fig. 7.A). However, the spatial accuracy was lower than that of the PQ-based clustering (Fig. 7.E). This was because CaImAn extracted some clusters that didn't correspond to any boutons whose

number was about the total number of the correct boutons. One reason for this mislabeling would be that the nature of cooperative activity made CaImAn difficult to find the boundary of some boutons. In the temporal aspect, some peaks were not detected in CaImAn (Fig. 7.B), and the temporal accuracy of CaImAn was lower than that of the PQ-based clustering (Fig. 7.F). This might be caused by CaImAn's ability to decompose a mixture of activity signals from spatially overlapped components: the correct boutons were not overlapped in our artificial data, but CaImAn possibly assumed that some activity peaks came from other overlapped boutons and eliminated them from the activity extraction. Therefore, some peaks would be removed. When the noise level was increased up to σ_{noise} = 0.5 and more, the overall bouton configuration and spatial accuracy were maintained (Fig. 7.C and E). However, the noises strongly affected the extracted signals, and temporal accuracy decreased (Fig. 7.D and F). Overall, PQ-based clustering could detect bouton configurations and calcium signals more accurately for the artificial burst-like activity data than the CaImAn algorithm.

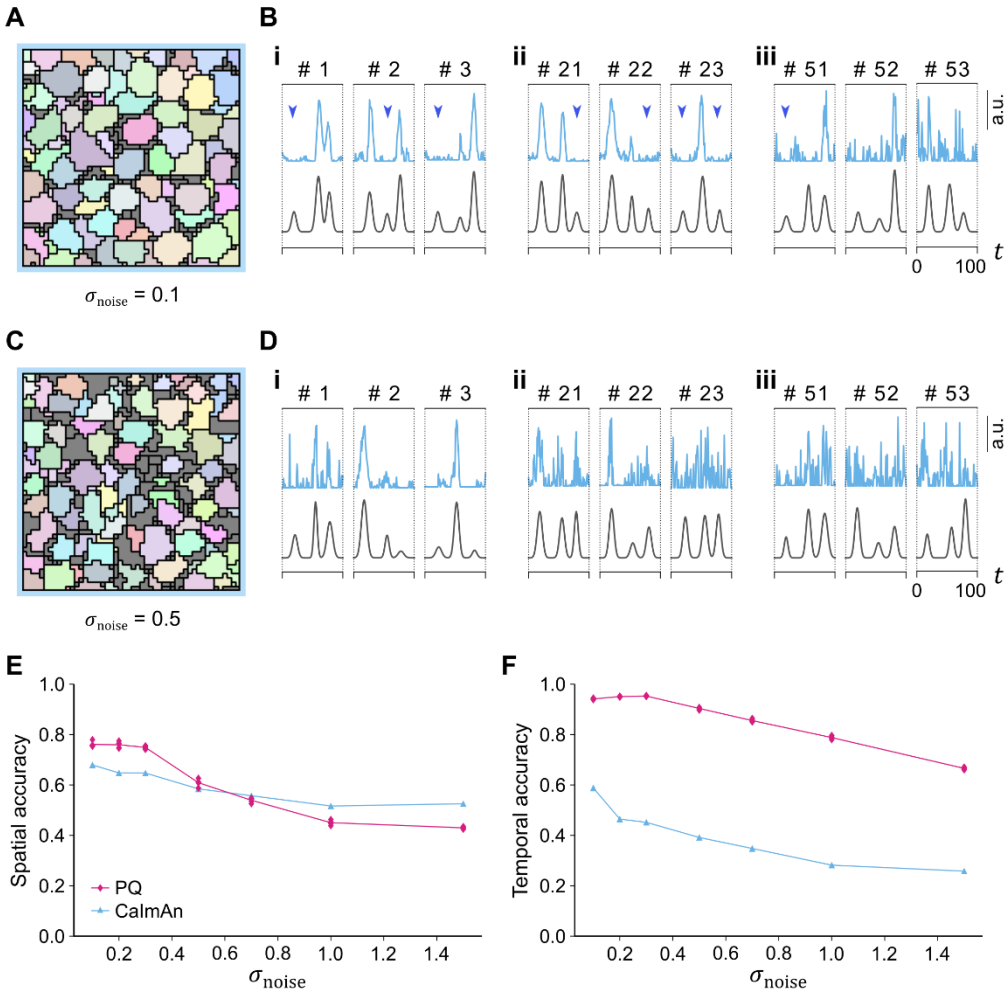

Fig. 7. Comparison of PQ-based clustering and CaImAn in the analysis of the burst-like artificial data. (A-D) Bouton extraction from the burst-like artificial data with σ_{noise} = 0.1 (A and B) and σ_{noise} = 0.5 (C and D). (A and C) Detected bouton configuration by CaImAn. The color of each cluster corresponds to that of its closest correct bouton shown in Fig. 4.A. (B and D) Extracted signals (cyan traces) by CaImAn from the data shown in (A) and (C), respectively, and the correct signals (gray traces). Pairs of true and detected boutons with large Dice coefficient (i: top 1-3, ii: top 21-23, iii: top 51-53) are shown. Blue arrowheads indicate the timings of peaks missed in the results by CaImAn. (E and F) Spatial (E) and temporal (F) accuracy in the analysis of the burst-like artificial data of each noise intensity by PQ-based clustering (magenta diamonds) and CaImAn (cyan triangles). The plot of PQ-based clustering in (E) and (F) are the same as in Fig. 4.G and H, respectively.

3.2.2. Detection accuracy in the analysis of sporadic activity data

The observation above implied the low detection accuracy attributed to the nature of burst-like activity patterns. To pursue this possibility, we generated artificial data with sporadic activities and applied the two clustering algorithms to them (see section 2.3 for generating the sporadic activity data.) CaImAn extracted peaks in correct signals more robustly, drastically improving its temporal accuracy (compare Fig. 8.F and H with Fig. 7.B and F.) These results suggested that the PQ-based clustering should be useful to extract bouton configurations and calcium signals, especially from calcium imaging data with burst-like activities.

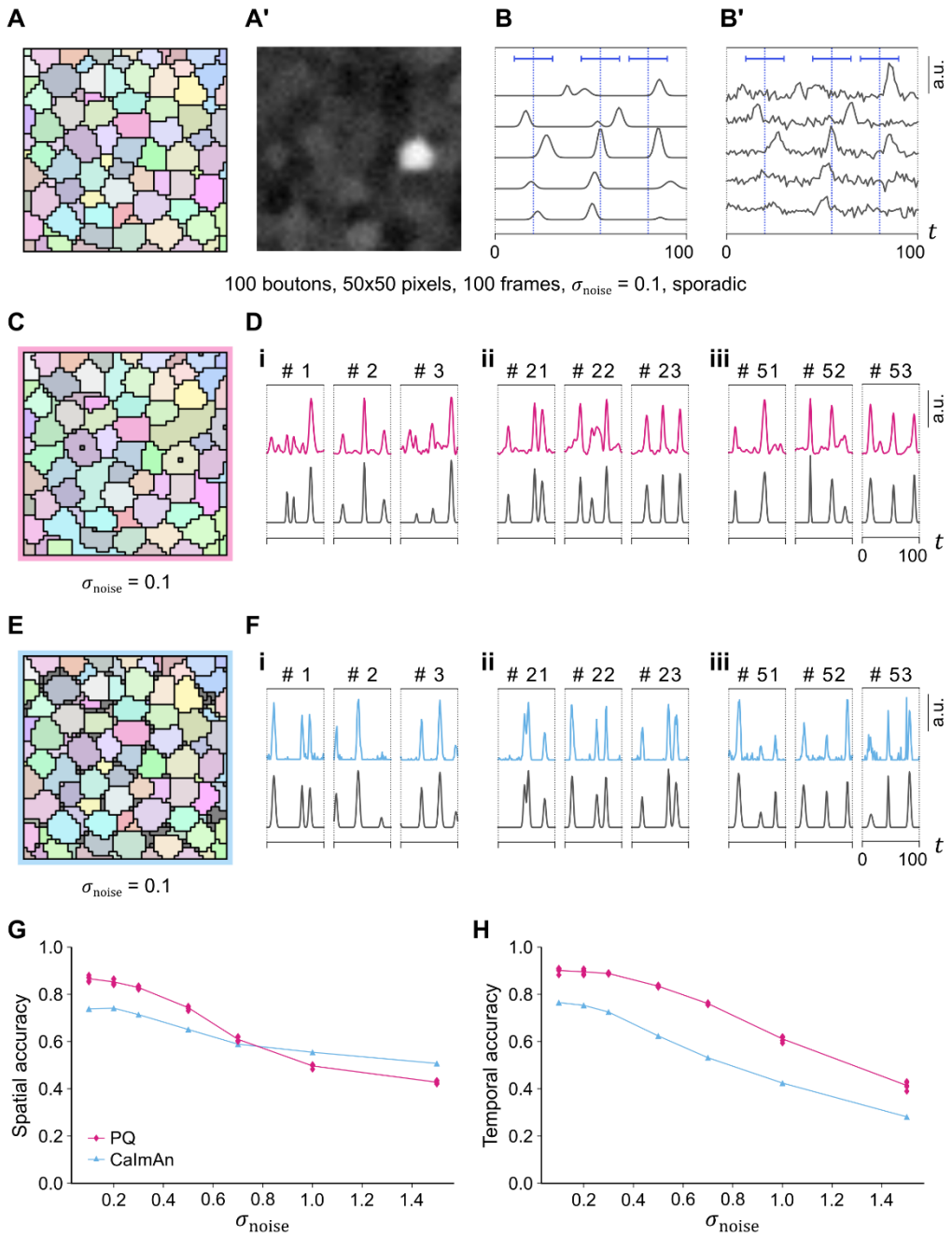

Fig. 8. Both PQ-based clustering and CaImAn successfully detected boutons and calcium signals from sporadic artificial data. (A) Correct configuration of boutons in the artificial data. This data is the same as in Fig. 4.A. (A') Signal intensity in one frame in the sporadic artificial data. (B) Correct signals of five example pixels. (B') Several perturbations were added to the correct signals to achieve realistic artificial data. The traces show the signal in the five pixels selected in (B) with the perturbations under the condition of σ_{noise} = 0.1. Vertical blue lines

indicate the center of each sporadic event, and horizontal blue bars indicate the standard deviation of peak timings around each center. (C-F) Bouton extraction from the sporadic artificial data with σ_{noise} = 0.1 by PQ-based clustering (C and D) and CaImAn (E and F). (C and E) Detected bouton configuration of PQ-based clustering and CaImAn, respectively. The color of each cluster corresponds to that of its closest correct bouton in (A). (D and F) Extracted signals by PQ-based clustering (magenta traces in D) and CaImAn (cyan traces in F) from the data shown in (C) and (E), respectively, and the correct signals (gray traces). Pairs of correct and detected boutons with large Dice coefficient (i: top 1-3, ii: top 21-23, iii: top 51-53) are shown. (G and H) Spatial (G) and temporal (H) accuracy in the analysis of the sporadic artificial data of each noise intensity by PQ-based clustering (magenta diamonds) and CaImAn (cyan triangles).

3.2.3. Calculation time

The calculation time was one of the critical factors in the data analysis. The time required to conduct clustering on a standard desktop computer with the PQ-based clustering and CaImAn was analyzed (see section 2.6 for details.) In processing the 100-frame data, the calculation time for the PQ-based clustering was shorter than that for the CaImAn algorithm (Fig. 9.A). To test this observation in different data sizes, we generated artificial data with a different frame or an image size. First, the frame number was changed from 100 to 1600. While the calculation time for both algorithms increased as the number of frames increased, the time for PQ-based clustering was shorter in the range tested while keeping better detection accuracy of the PQ-based clustering (Fig. 9.A, C, and D). Next, the image size was changed from 1000 to 8000 pixels. Calculation time for both CaImAn and PQ-based clustering increased linearly as the number of pixels went up, and the calculation duration for PQ-based clustering was about a half or one-third of that for CaImAn while keeping better detection accuracy of the PQ-based clustering (Fig. 9.E, G, and H). Consequently, the PQ-based clustering required less calculation time and provided more accurate clustering in the range of data size we tested.

The time for PQ-based clustering consisted of three parts: preprocessing, calculating the patch-wise modularity matrix, and simulated annealing. Of these three,

preprocessing occupied the longest time in the processing time. On the other hand, calculation of the patch-wise modularity matrix and simulated annealing accounted for less time (Fig. 9.B and F). Accordingly, efficient computing in the patch-wise calculation that drastically reduced the number of the calculation of the similarity between pixels would underlie the fast bouton detection by the PQ-based clustering.

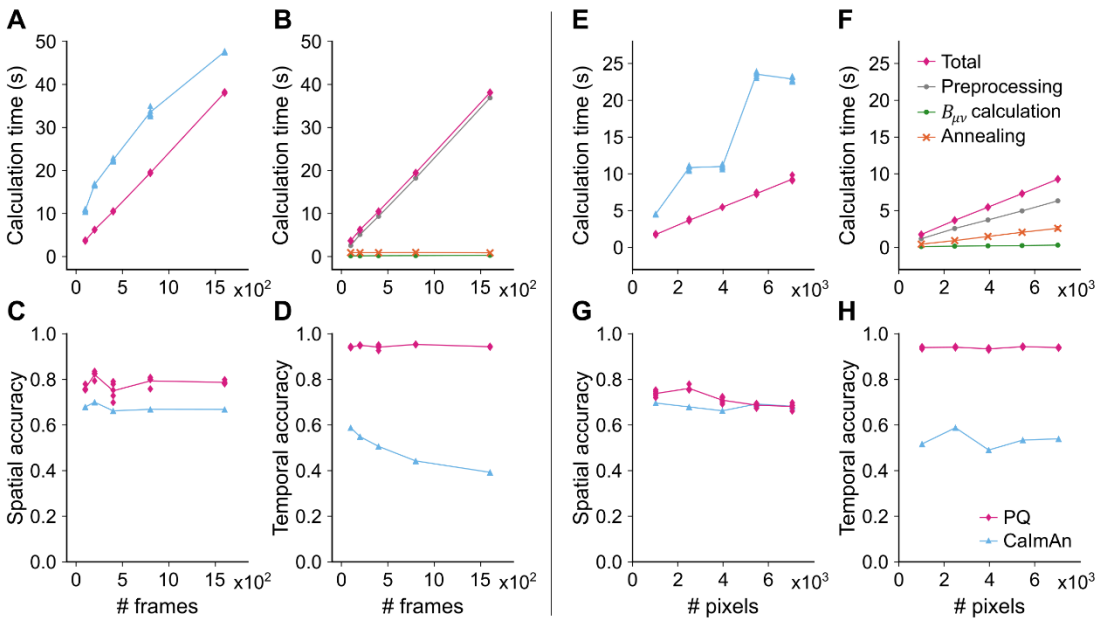

Fig. 9. Comparison of the calculation time between the PQ-based clustering and CalmAn. (A-D) Performance of the PQ-based clustering and CalmAn in the analysis of burst-like artificial data of various total frame numbers. (A) Calculation time. (B) The breakdown of (A). (C) Spatial accuracy. (D) Temporal accuracy. (E-H) Performance of the PQ-based clustering and CalmAn in the analysis of burst-like artificial data of various total pixel numbers. (E) Calculation time. (F) The breakdown of (E). (G) Spatial accuracy. (H) Temporal accuracy. In (A), (C), (D), (E), (G), and (H), magenta diamonds indicate the results of PQ-based clustering, whereas cyan triangles indicate those of CalmAn. In (B) and (F), magenta diamonds, gray circles, green circles, and orange crosses represent the total calculation time for PQ-based clustering, the time for preprocessing, the time to calculate the modularity matrix $B_{\mu\nu}$, and the time for simulated annealing.

3.3. Performance assessment with experimental data

To assess the performance of the clustering method with experimental data, we performed pan-neuronal calcium imaging of the neuropil of the fly VNC and applied preprocessing, such as baseline correction, the elimination of deformation and specimen drift, and the extraction of signals from the focal plane, to the data (see Methods section 2.2.) We obtained spatiotemporal data (512 x 512 pixels, 700 frames) from three isolated CNS (Fig. 10.A and A'). In each imaging data, the propagations of neural activity waves were observed multiple times.

First, we compared the performance between PQ-based clustering and CaImAn by image data of small size. A region of 100x100 pixels from one sample was clipped (Fig. 10.A, the square of dashed lines). Both the PQ-based clustering and CaImAn could detect bouton-like regions from the calcium imaging data. The size of bouton-like structures extracted by the PQ-based clustering was consistent with that by CaImAn (the PQ-based clustering: 0.6 ± 0.4 μm (n = 401), CaImAn: 0.7 ± 0.2 μm (n = 1,959)). However, since CaImAn allows the overlapping of boutons, the total numbers of the extracted boutons were larger in the CaImAn output than in the PQ-based clustering result (Fig. 10.B and C). Furthermore, while the total areas of the extracted boutons in the PQ-based clustering results were the same as the total area of the original image, those in the CaImAn results were 4 times larger than the original image (400 ± 10%), which implied that the results by CaImAn should include a large number of clusters that didn't correspond to correct boutons. To investigate the signals from the bouton-like structures extracted by the two clustering methods, we matched the clusters from the two algorithms based on the overlapped areas measured by the Dice coefficient and compared the signals of matched clusters (Fig. 10.D). Some matched boutons exhibited consistent activity between the two (Fig. 10.D.i). However, as observed in the analysis of the artificial data (Fig. 4.D and 7.B), some peaks were detected by the PQ-based clustering but not by CaImAn (Fig. 10.D.ii and iii). Therefore, the peak heights of the activity of the bouton-like structures were higher in the result of the PQ-based clustering than that of CaImAn (Fig. 10.E). These observations suggest that the PQ-based clustering can detect peaks of bouton activity more reliably than CaImAn.

Next, we applied PQ-based clustering to the full-size imaging data and extracted bouton-like regions (4,950 ± 640 (mean ± std., n = 3)) and their temporal profiles (Fig. 10.F

and F'). The calculation for the clustering of the full-size data using PQ-based clustering took 40.6 ± 0.6 (mean ± std., n = 3) minutes (preprocessing: 93%, calculation of $B_{\mu\nu}$: 2%, simulated annealing: 4%). Accordingly, PQ-based clustering allowed us to extract bouton-like structures from experimental calcium imaging datasets.

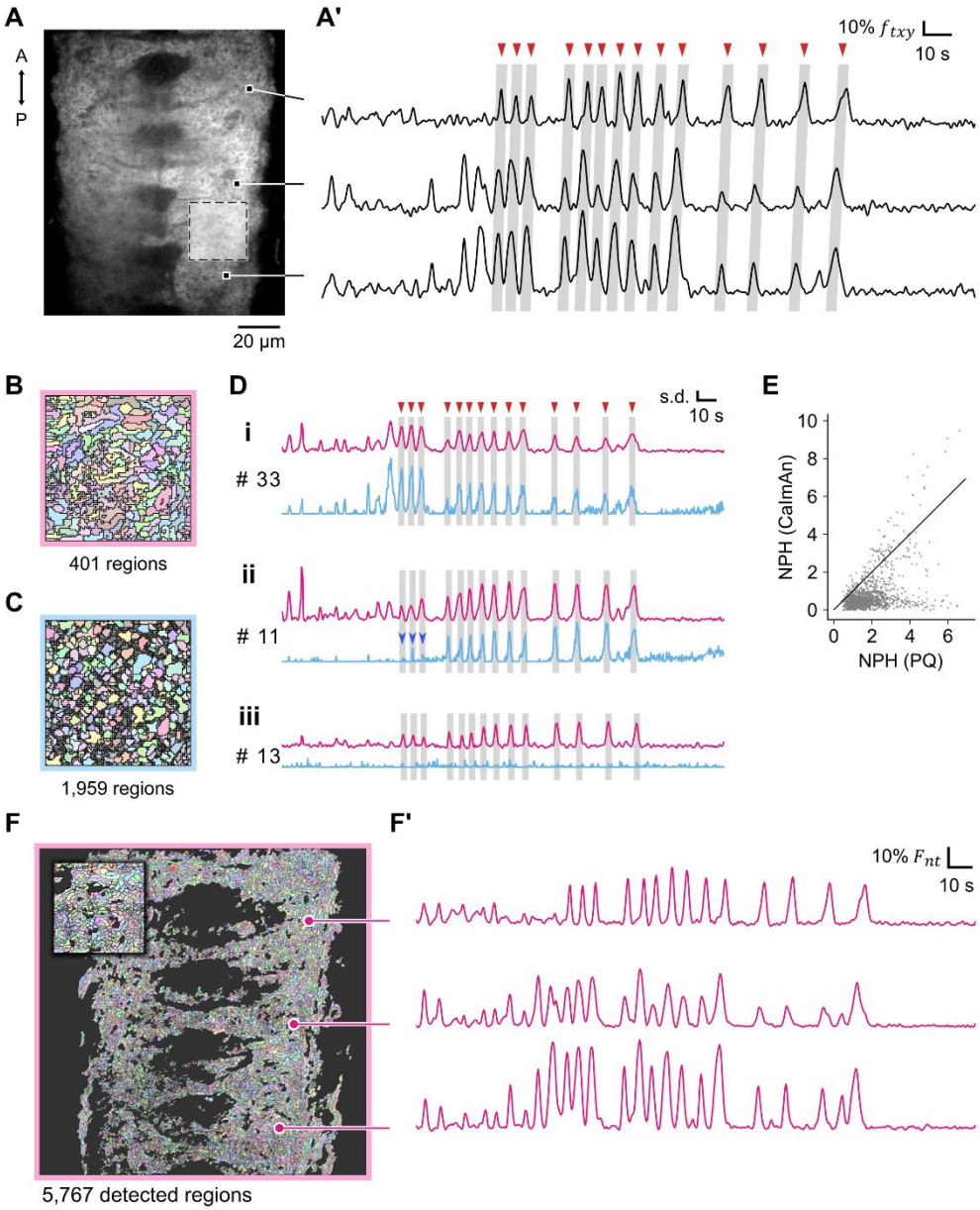

Fig. 10. PQ-based clustering was applicable to real calcium imaging data. (A) Snapshot image of calcium imaging of the isolated CNS of a *Drosophila* larva. Dashed square indicates the position where the image was clipped for (B) and (C). (A') Preprocessed temporal traces of example pixels in (A). Red arrowheads and shaded areas indicate activity propagations from the posterior to anterior segments corresponding to fictive crawling behavior. (B and C) Clustering results of the clipped data by the PQ-based clustering (B) and CaImAn (C). Each detected cluster in (B) is indicated by a different color. For each cluster in (B), a cluster in (C) that overlapped with the cluster in (B) the most was filled with the same color. Clusters in (C) that were not painted based on the overlap with clusters in (B) were filled with gray. (D) Extracted temporal signals from the clipped data using PQ-based clustering (magenta) or CaImAn (cyan). Three pairs of detected clusters from the two methods with a large Dice coefficient (i: top 33, ii: top 11, iii: top 13) are shown. Blue arrowheads indicate the timings of peaks missed in the results by CaImAn. Red arrowheads and shaded areas indicate the same activity propagation events in (A'). The signal intensity obtained by each method was normalized by the standard deviation (s.d.) of peak heights of all the detected clusters in the indicated activity propagation events (PQ-based clustering or CaImAn). (E) Comparison of the peak heights of spatially matched 110 clusters from PQ-based clustering and CaImAn. Each data point represents normalized peak heights (NPHs) of a matched cluster. The vertical axis shows NPH by the PQ-based clustering, while the horizontal axis shows that by CaImAn. The black line indicates NPH (PQ) = NPH (CaImAn), where the normalized peak height by the PQ-based clustering is the same as that by CaImAn. (F) Detected bouton-like regions in the whole imaging data of the same sample by PQ-based clustering. Each of the bouton-like regions is shown by a different color. The upper-left window shows an expanded image of the region indicated by the white dotted box in (A). (F') Temporal traces of the example bouton-like regions in (F) taken from the same positions in (A). Note that this clustering was done without manual tracing and showed consistent results with (A').

3.4. Network analysis among the extracted bouton-like regions

Finally, as an application of the clustering data, we attempted to extract a network structure of the fly CNS based on the clustering results. We used the correlation of calcium signals between two bouton-like structures to define the interaction between them. We

calculated the correlation coefficient between every bouton pair's temporal profiles, binarized them with a threshold of 0.9, and obtained an unweighted graph of bouton-like structures and their connections (Fig. 11.A).

Network analysis of the graphs suggested they were scale-free (Barabási & Albert, 1999). We found that the distribution of degrees, which were the numbers of edges from bouton-like structures, followed the power law (Fig. 11.B). The scaling exponent α was 1.14 ± 0.01 (mean ± std., n=3), which implied that there existed hub nodes in the networks (Cohen et al., 2000; Pastor-Satorras & Vespignani, 2001; Wang et al., 2011). This was consistent with previous studies that had shown functional connectivity in the human brain forms scale-free graphs (Eguíluz et al., 2005; Stam & Reijneveld, 2007; van den Heuvel et al., 2008).

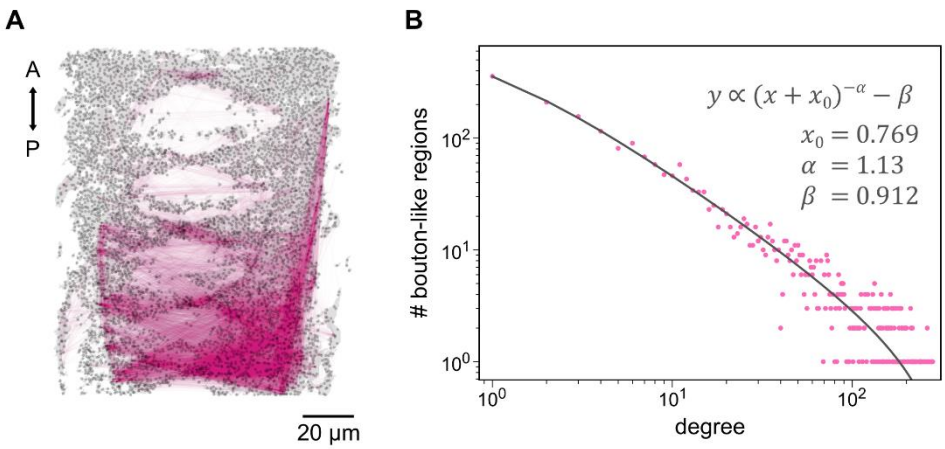

Fig. 11. Correlation networks among activities of bouton-like regions formed scale-free networks. (A) Extracted network structure of the same sample of the CNS of a *Drosophila* larva in Fig. 10. Each black dot indicates the centroid of each detected region. Each magenta line connects two regions with a correlation coefficient larger than 0.9. (B) Histogram of degrees in the network of neural activity in the CNS of *Drosophila* larvae. Magenta dots show the network's degree distribution in (A) in a log-log plot. The horizontal axis indicates the degree of each bouton-like structure, while the vertical axis indicates the number of bouton-like regions. The black line fits the histogram whose model is shown in the panel.

4. Discussion

Here, we introduced a new methodology to detect bouton-like regions from artificial and experimental calcium imaging data. To detect bouton-like regions from images where boutons were densely concentrated, we made progress in three aspects of algorithms and theories: first, we established pre-processing methods to extract signals from focal planes in experimental imaging data by using a two-dimensional baseline detection algorithm; second, we defined patch-wise modularity, which is an expanded version of Newman and Girvan's modularity (Newman & Girvan, 2004), and mathematically proved its validity and availability; third, we engineered an algorithm to maximize the patch-wise modularity based on simulated annealing method (Kirkpatrick et al., 1983).

The first progress was to extract signals from well-focused planes in the imaging data. Since the neurites in the neuropil are densely packed in a three-dimensional space, imaging data for population activities in the neuropil typically include fluorescence signals from out of the focal plane. In this study, we extracted signals from the well-focused plane by a two-dimensional baseline detection algorithm. Several algorithms have been proposed to determine baselines of one-dimensional spectral data, such as Raman and infrared spectra, based on a wavelet transform (Bertinetto & Vuorinen, 2014; Qian et al., 2017; Shao & Griffiths, 2007), derivative methods (Leger & Ryder, 2006; Schulze et al., 2005), polynomial fitting (Gan et al., 2006; Lan et al., 2007; Lieber & Mahadevan-Jansen, 2003; Zhao et al., 2007), and penalized least squares (Baek et al., 2015; He et al., 2014; F. Zhang et al., 2020; Z. M. Zhang et al., 2010). However, these methods were not readily expanded to the two dimensions because their implementation in higher dimensions could be complicated. In our approach, we introduced iterative Gaussian blur to realize baseline determination in two-dimensional images and successfully detected baseline from experimental calcium imaging data.

The second achievement was the introduction of patch-wise modularity. Although modularity is a widely used quality measure to detect community structures in a network (Bruno et al., 2015; Chen et al., 2009; Fine et al., 2006; Humphries, 2011; Sporns & Betzel, 2016; Torre et al., 2019), it has been criticized because the size of its detected communities is affected by the size of the whole network (Fortunato & Barthélemy, 2007; Lancichinetti & Fortunato, 2011; Rosvall & Bergstrom, 2013). This "resolution limit problem" arises because

the definitions of the modularity involve the term the total strength of all edges. In our calcium imaging data analysis, the resolution limit was about at least nine pixels, which hindered the detection of bouton-like structures whose size was about five pixels. Moreover, our targeting networks of similarity among pixels were large and dense (basically, all node pairs were connected), so it was almost impossible to calculate the total strength. Several researchers have reported that the resolution limit problem can be eliminated by multi-resolution modularity (Arenas et al., 2008; Reichardt & Bornholdt, 2006) or an edge weighting scheme (Cao et al., 2016; Haq et al., 2019; Lu et al., 2018). However, these methods could not be applied to our analysis because they still needed to calculate the total strength. Or other methods tackled the resolution limit problem by using quality measures without such global summation of edge strength (Li et al., 2008; Muff et al., 2005). However, these measures were also inapplicable to our case because they were efficiently calculated only when edges were sparse enough. Our patch-wise modularity was defined without global quantities. Moreover, the pixel pairs contributing to the modularity estimation were restricted within each pixel-centered circle patch. Therefore, the calculation cost was within the acceptable level even with our large networks. In addition, since arbitral diameters of the patches were allowed in this measure, the virtual network size around each pixel could be easily modified. The flexibility in size enabled us to adjust the preferred size of pixel clusters obtained by optimizing the patch-wise modularity. Thanks to this measure, we succeeded in extracting bouton-sized pixel clusters independent of the resolution limit. The adjustability of the typical size of clusters (Fig. 5) might support the accessibility of structures in different scales from that of boutons in the neuropil, including cell bodies.

There is one limitation in using our method when decomposing a network. In our application of the detection of bouton-like structures, we took advantage of the fact that there is a stereotyped scale in boutons in the CNS. Using this bouton scale, we set the weight matrix as a binary function described in Eq. (2.5). In general applications, network decomposition methods are used to simultaneously detect communities of various sizes (Lancichinetti & Fortunato, 2011; Li et al., 2008; Rosvall & Bergstrom, 2013). The patch-wise modularity used in this paper cannot be applied to detect clusters with different sizes in the current form. However, the patch-wise modularity can be extended to meet this demand by setting other forms of the weight factor instead of Eq. (2.5). This factor is adjustable according

to the purpose of each network decomposition, and we gave a mathematical proof of the validity of using any form of the weight matrix in the patch-wise modularity framework (Appendix B). Therefore, some modified version of the patch-wise modularity might detect various sizes of communities simultaneously in real-world networks by setting biases properly.

Finally, we developed a maximization algorithm for the patch-wise modularity based on the simulated annealing method. While there were several existing algorithms to maximize modularity (Clauset et al., 2004; Duch & Arenas, 2005; Guimerà et al., 2004; Newman, 2004, 2006; Reichardt & Bornholdt, 2006; Torre et al., 2019; White & Smyth, 2005), we made much account of the speed in the calculation, since the similarity networks among the imaging pixels were large and complex: about 2×10^5 nodes and 7×10^6 edges even after restricted within the patches, and 10^4 clusters to be detected. Most maximization methods require calculation costs proportional to at least quadratics of these numbers, making it challenging to manipulate our data. The simulated annealing method can search parameter space at a fast speed (Metropolis et al., 1953). Furthermore, this method was suitable for our application: In the simulated annealing methods, the assignment of clusters was iteratively changed during the maximization process. Since boutons had continuous shapes, each pixel should belong to boutons in its neighbors. Moreover, we permitted duplicated identities for different and spatially segregated clusters since we could easily split them afterwards while the number of clusters to calculate could be reduced. These all properties contributed to the drastic reduction in the range of the search space and calculation costs. Thanks to these optimizations, the algorithm achieved the calculation costs proportional to simply the pixel number, with the fixed radius of the patches and number of iterations.

This study showed that our patch-wise modularity-based pixel clustering algorithm provided better results in the detection of bouton-like structure from neuropil data with burst-like activity than CaImAn (Giovannucci et al., 2019), which was based on constrained non-negative matrix factorization (CNMF) (Pnevmatikakis et al., 2016). Two properties of the neuropil data could be considered as its causes: 1. Temporal overlap of peaks between different boutons, and 2. Dense configuration of boutons.

In the burst-like activity data, signals of boutons that were close to each other exhibited cooperative activation (Fig. 4.B and 10.A'). This property was challenging for the NMF-based algorithm because NMF attempted to extract spatial components showing

946 mutually different signals, which didn't allow nearby boutons to exhibit similar activity. This
947 issue could be found in the case of other methods to decompose calcium imaging data. For
948 example, several decomposition methods based on principal component analysis,
949 independent component analysis, similarity analysis, and singular value decomposition
950 (Mölter et al., 2018; Mukamel et al., 2009; Shibue & Komaki, 2020) assume that the activities
951 of different neurons are highly independent. In addition, CNMF deconvolves calcium signals
952 into spike trains by promoting the sparseness of spikes. However, the neural spikes in our
953 calcium imaging data were not sparse but showed coordinated burst activity related to motor
954 control.

955 The dense configuration of boutons in the neuropil imaging data could also affect
956 the clustering performance of CaImAn. CNMF is supposed to be used to extract cellular
957 components that are dispersed in the neural tissue. In addition, various imaging data
958 decomposition algorithms are based on the assumption that there is a background space
959 between cell bodies. For example, an NMF-based algorithm by (Maruyama et al., 2014)
960 includes background components, and the activity-based level set segmentation (ABLE)
961 (Reynolds et al., 2017) detects regions on the premise that the target regions are surrounded
962 by background regions. In contrast, boutons were densely packed in the neuropil images,
963 and there was no space between boutons, which would make CNMF and other
964 decomposition algorithms hard to segregate individual boutons. Furthermore, while clusters
965 extracted by the PQ-based method were not spatially overlapped, CaImAn and other NMF-
966 based methods allowed clusters to be overlapped. This property could underlie the
967 observation that excess boutons were detected by CaImAn (Fig. 10.C).

968 It is important to note a limitation of our clustering method. Pixel clusters obtained
969 by our methods can include not only a whole single bouton but also a fragment of a single
970 bouton or a group of boutons. Because our method fixes the typical size of a cluster, a bouton
971 larger than that scale might be split into multiple parts, or smaller boutons might be grouped
972 into a single bouton. Actually, in the simulation with the artificial data, our method could not
973 detect small boutons compared with patch radius r = 4.0 pixels and merged them with
974 adjacent boutons (compare Fig. 4.A and C.) Our algorithm might group spatially overlapping
975 boutons if the focal plane has a non-negligible thickness because distinct boutons with
976 different depths can merge onto the same pixel. While some previous algorithms, including

NMF, can decompose a pixel into multiple overlapping components (Maruyama et al., 2014; Mukamel et al., 2009; Pnevmatikakis et al., 2016; Shibue & Komaki, 2020), our pixel clustering method distributes exactly one cluster identity to each pixel. Our simulation showed the case where spatially overlapping boutons were grouped into a single cluster (Fig, 8.D): a part of detected boutons in high spatial accuracy exhibited additional activity peaks that were not included in the correct signals but were originated from spatially overlapped neighboring boutons.

Nevertheless, our method provides vast accessibility to the spatiotemporal population activity in the neuropil. Based on the extracted activity data, we could analyze network properties in the motor circuits (Fig. 11). Therefore, the spatiotemporal population activity of boutons could be a novel approach to examining neural network dynamics at the cellular level.

Acknowledgements

We thank Bloomington Drosophila Stock Center for providing fly lines. We thank Dr. Ken Nakae for the critical comments on this paper. We also thank Drs. Dohjin Miyamoto, Suguru Takagi, Atsuki Hiramoto, Yingtao Liu, and Xiangsunze Zeng for informative discussions. This work was supported by MEXT/JSPS KAKENHI grants [17K19439, 19H04742, and 20H05048 to A.N. and 17K07042 and 20K06908 to H.K.].

Appendix A: Patch-wise modularity

In one of the standard procedures for the analysis of modularity, for a given network whose connectivity strength between nodes μ and ν is given by $A_{\mu\nu}$ ($A_{\mu\nu} = A_{\nu\mu} \geq 0$), the modularity matrix is defined as follows:

$$B_{\mu\nu} \coloneqq A_{\mu\nu} - P_{\mu\nu}$$

where $P_{\mu\nu} \coloneqq d_\mu d_\nu / m$, $d_\mu \coloneqq \sum_\nu A_{\mu\nu}$, $m \coloneqq \sum_\mu d_\mu = \sum_{\mu,\nu} A_{\mu\nu}$. When we assume $p_\mu = d_\mu / m$ is the probability distribution of connections linked to μ, it is clear that $P_{\mu\nu} = m \cdot p_\mu p_\nu$ is the expected connectivity strength of arbitral networks with the same degree distribution d_μ (note that $\sum_\nu P_{\mu\nu} = \sum_\nu A_{\mu\nu} = d_\mu$.)

In this study, we impose a constraint on the link between two nodes based on their spatial proximity: two nodes separated by more than five pixels are not linked. This condition cannot be met by the conventional assumption $P_{\mu\nu} \coloneqq d_\mu d_\nu / m$ since it doesn't use the information about their position. Accordingly, instead of defining $P_{\mu\nu}$ as being proportional to the product of d_μ / m and d_ν / m, we assume that $P_{\mu\nu}$ should have non-zero components only if $\mathrm{d}(\mu, \nu) < r$ (See Eq. (2.5) in the Method section) and follow the same distribution of degrees as that of $\hat{A}_{\mu\nu} \coloneqq W_{\mu\nu} \cdot A_{\mu\nu}$. By this assumption, we can write the null network as follows:

$$\hat{P}_{\mu\nu} \coloneqq W_{\mu\nu} \cdot k_\mu k_\nu \tag{A.1}$$

Here, $W_{\mu\nu}$ is defined in Eq. (2.5) and k_μ is a positive vector that is chosen to satisfy $\sum_\nu \hat{P}_{\mu\nu} = \sum_\nu W_{\mu\nu} \cdot k_\mu k_\nu = \hat{d}_\mu$. For each given $W_{\mu\nu}$ (not only the one defined in Eq. (2.5)) and $\hat{d}_\mu$, there always exists one and only one k_μ (Theorem 1, Appendix B). $\hat{P}_{\mu\nu}$ can be computationally given by using Theorem 2 in Appendix B. We define a patch-wise modularity matrix $\hat{B}_{\mu\nu}$ by this null network.

$$\hat{B}_{\mu\nu} \coloneqq \hat{A}_{\mu\nu} - \hat{P}_{\mu\nu} \tag{A.2}$$

Because $\hat{B}_{\mu\nu}$ reduces to the original modularity matrix in case of $W_{\mu\nu} = 1$ for every μ, ν (then $k_\mu = d_\mu / \sqrt{m}$), the definition gives an expansion of the conventional modularity. For a given cluster configuration $S = \{s_\mu\}$, the modularity is defined as follows:

$$\hat{Q}(S) \coloneqq \sum_{\mu,\nu} \hat{B}_{\mu\nu} \cdot \delta(s_\mu, s_\nu)$$

Appendix B: Theorems and proofs

Definition 1:

Let μ and ν be integers in $\{1, 2, \dots, N\}$ where $N \in \mathbb{N}$.

$W_{\mu\nu}$ is a $N \times N$ non-negative symmetric matrix with positive diagonal components:

$$W_{\mu\nu} = W_{\nu\mu} \geq 0, \qquad W_{\mu\mu} > 0 \qquad \forall \mu, \nu$$

k_μ and d_μ are N-dimensional positive vectors:

$$k_\mu, d_\mu > 0 \qquad \forall \mu$$

Definition 2:

For an N-dimensional vector $\vec{x}$, $\vec{g}: \mathbf{R}^N \to \mathbf{R}^N$ $\left(\vec{g}(\vec{x}) = \left(g_1(\vec{x}),\ g_2(\vec{x}), \dots,\ g_N(\vec{x})\right)\right)$ is defined as follows:

$$g_\mu(\vec{x}) \coloneqq \sum_\nu W_{\mu\nu} x_\mu x_\nu \tag{B.1}$$

where $\vec{x} = (x_1, x_2, \dots, x_N)$.

Theorem 1:

For given $W_{\mu\nu}$ and d_μ, there always exists one and only one k_μ satisfying the following equations:

$$\sum_\nu W_{\mu\nu} k_\mu k_\nu = d_\mu \qquad \forall \mu \tag{B.2}$$

Proof:

Lemma 1: $x_\mu \geq 0 \quad \forall \mu \ \Rightarrow\ g_\mu(\vec{x}) \geq 0 \quad \forall \mu$

Proof:

By the assumption and Definition 1, all of the terms in $g_\mu(\vec{x})$ are positive. So, this statement is true. ■

Lemma 2: For a given $\mu_0 \in \{1, 2, \dots, N\}$, $x_{\mu_0} = 0 \ \Leftrightarrow\ g_{\mu_0}(\vec{x}) = 0$

Proof:

Sufficiency: $x_{\mu_0} = 0 \Rightarrow\ g_{\mu_0}(\vec{x}) = x_{\mu_0} \sum_\nu W_{\mu_0\nu} x_\nu = 0$

Necessity: $g_{\mu_0}(\vec{x}) = \sum_\nu W_{\mu_0\nu} x_{\mu_0} x_\nu \geq W_{\mu_0\mu_0} x_{\mu_0}^2 \quad \left(\because W_{\mu_0\nu} x_{\mu_0} x_\nu \geq 0 \ \forall \nu\right) \geq 0$

$$\therefore g_{\mu_0}(\vec{x}) = 0 \ \Rightarrow\ W_{\mu_0\mu_0} x_{\mu_0}^2 = 0 \Leftrightarrow x_{\mu_0} = 0 \quad \left(\because W_{\mu_0\mu_0} > 0\right) \ ■$$

From Lemma 1 and 2, $x_\mu > 0 \quad \forall \mu \ \Rightarrow\ g_\mu(\vec{x}) > 0 \quad \forall \mu$. Therefore, when $D^N = \left\{\vec{x} \,\middle|\, x_\mu > 0 \ \forall \mu\right\}$, the mapping $\vec{g}: \mathbf{R}^N \to \mathbf{R}^N$ includes a partial mapping $\vec{g}: D^N \nrightarrow D^N$ ($\because D^N \supseteq \vec{g}(D^N)$). Because Theorem 1 is equivalent to the existence of the inverse mapping $\vec{g}^{-1}: D^N \nrightarrow D^N$, we attempt to prove $\vec{g}: D^N \nrightarrow D^N$ is bijective.

Lemma 3: The partial mapping $\vec{g}: D^N \nrightarrow D^N$ is injective.

Proof:

Lemma 3 is proved by contradiction.

Assumption 1: $\exists \vec{x}^{(1)}, \vec{x}^{(2)} \in D^N$ s.t. $\vec{x}^{(1)} \neq \vec{x}^{(2)}$, $\vec{g}(\vec{x}^{(1)}) = \vec{g}(\vec{x}^{(2)})$

By using a line $\vec{h}(t) = \vec{x}^{(1)} + (\vec{x}^{(2)} - \vec{x}^{(1)}) \cdot t$, which passes through the two points $\vec{x}^{(1)}$ and $\vec{x}^{(2)}$, a scalar function $\gamma_\mu(t) \coloneqq g_\mu\left(\vec{h}(t)\right)$ is defined. From Definition 2, $\gamma_\mu(t)$ is described as a polynomial of t, which is at most quadratic. Therefore, from $\gamma_\mu(0) = \gamma_\mu(1)$ $\left(\Leftrightarrow g_\mu(\vec{x}^{(1)}) = g_\mu(\vec{x}^{(2)})\right)$, $\gamma_\mu(t) = \gamma_\mu(1-t)$ $\forall t \in \mathbf{R}$ has to be satisfied for every μ. Accordingly, the following equation is required:

$$\begin{aligned} \vec{g}\left(\vec{h}(t)\right) &= \vec{g}\left(\vec{h}(1-t)\right) \\ &= \vec{g}\left(\vec{x}^{(1)} + (\vec{x}^{(2)} - \vec{x}^{(1)}) \cdot (1-t)\right) \\ &= \vec{g}\left(\vec{x}^{(1)} + \vec{x}^{(2)} - \vec{h}(t)\right) \quad \forall t \in \mathbf{R} \end{aligned} \tag{B.3}$$

Here, there is at least one μ_0 which satisfies $x_{\mu_0}^{(2)} - x_{\mu_0}^{(1)} \neq 0$ $(\because \vec{x}^{(1)} \neq \vec{x}^{(2)})$. When $t_0 = -x_{\mu_0}^{(1)} / \left(x_{\mu_0}^{(2)} - x_{\mu_0}^{(1)}\right)$, $h_{\mu_0}(t_0) = 0$ is valid. Then, the following equations are derived:

$$\begin{aligned} h_{\mu_0}(t_0) = 0 &\Leftrightarrow g_{\mu_0}\left(\vec{h}(t_0)\right) = 0 && (\because \text{Lemma 2}) \\ &\Leftrightarrow g_{\mu_0}\left(\vec{x}^{(1)} + \vec{x}^{(2)} - \vec{h}(t_0)\right) = 0 && (\because \text{Eq. (B.3)}) \\ &\Leftrightarrow x_{\mu_0}^{(1)} + x_{\mu_0}^{(2)} - h_{\mu_0}(t_0) = 0 && (\because \text{Lemma 2}) \\ &\Leftrightarrow x_{\mu_0}^{(1)} + x_{\mu_0}^{(2)} = 0 && (\because h_{\mu_0}(t_0) = 0) \end{aligned}$$

However, this is inconsistent with $x_{\mu_0}^{(1)}, x_{\mu_0}^{(2)} > 0$ $(\because \vec{x}^{(1)}, \vec{x}^{(2)} \in D^N)$. Therefore, Assumption 1 cannot be realized, and lemma 3 is true. ■

Lemma 4: The partial mapping $\vec{g}: D^N \nrightarrow \vec{g}(D^N)$ is a homeomorphism.

Proof:

Because the range is limited to the image of $\vec{g}$, the partial mapping $\vec{g}: D^N \nrightarrow \vec{g}(D^N)$ is surjective. Also, the mapping is injective from Lemma 3. Therefore, $\vec{g}: D^N \nrightarrow \vec{g}(D^N)$ is bijective. From Definition 2, both $\vec{g}: D^N \nrightarrow \vec{g}(D^N)$ and $\vec{g}^{-1}: \vec{g}(D^N) \nrightarrow D^N$ are continuous. Accordingly, the statement is true. ■

Lemma 5: The partial mapping $\vec{g}: D^N \nrightarrow D^N$ is surjective.

Proof:

Lemma 5 is proved based on mathematical induction.

Case 1: $N = 1$

For all $y > 0$, there exists $x > 0$, which satisfies $y = W_{11}x^2$ $\left(\because x = \sqrt{y/W_{11}}\right)$. $\vec{g}: D^1 \nrightarrow D^1$ is surjective.

Case 2: $N > 1$

Assumption 2: $\vec{g}: D^{N-1} \nrightarrow D^{N-1}$ is surjective.

The boundary of D^N is described as the following:

$$\partial D^N = \left\{\vec{x} \,\middle|\, x_\mu \geq 0 \ \ \forall\mu,\ \exists\mu_0 \in \{1, 2, \dots, N\} \ \text{s.t.} \ \ x_{\mu_0} = 0\right\}$$

From Assumption 2 and Lemma 2, $\vec{g}(\partial D^N) = \partial D^N$. From Lemma 4, $\vec{g}(\partial D^N) = \partial\vec{g}(D^N)$. Therefore, $\partial\vec{g}(D^N) = \partial D^N$. Then, from Lemma 4 and $\vec{g}(D^N) \subseteq D^N$, $\vec{g}(D^N) = D^N$ is derived. This indicates $\vec{g}: D^N \nrightarrow D^N$ is surjective. ■

From Lemma 3 and 5, the partial mapping $\vec{g}: D^N \nrightarrow D^N$ is shown to be bijective. This proves the validity of Theorem 1. ■

Definition 3:

A matrix sequence $P_{\mu\nu}^{(n)}$ $(n = 0, 1, 2, \dots)$ is defined by the following recurrence formula:

$$P_{\mu\nu}^{(0)} = W_{\mu\nu}, \qquad P_{\mu\nu}^{(n+1)} = P_{\mu\nu}^{(n)} \cdot \sqrt{\frac{d_\mu}{\sum_\sigma P_{\mu\sigma}^{(n)}} \cdot \frac{d_\nu}{\sum_\sigma P_{\sigma\nu}^{(n)}}}$$

For $P_{\mu\nu}^{(n)}$, $k_\mu^{(n)}$ is defined as follows:

$$k_\mu^{(0)} \coloneqq 1, \qquad k_\mu^{(n)} \coloneqq \prod_{i=0}^{n-1} \sqrt{\frac{d_\mu}{\sum_\sigma P_{\mu\sigma}^{(i)}}} \quad (n = 1, 2, 3, \dots)$$

Theorem 2:

For given $W_{\mu\nu}$ and d_μ, the positive vector k_μ which satisfies Eq. $(\mathrm{B}.2)$ is composed by the following:

$$\lim_{n\to\infty} k_\mu^{(n)} = k_\mu \quad \forall\mu$$

Proof:

Lemma 6: $P_{\mu\nu}^{(n)} = P_{\nu\mu}^{(n)} \geq 0,\ P_{\mu\mu}^{(n)} > 0 \quad \forall\mu, \nu$

Proof:

Lemma 6 is proved based on mathematical induction.

Case 1: $n = 0$

From $P_{\mu\nu}^{(0)} = W_{\mu\nu}$ and Definition 1, $P_{\mu\nu}^{(0)} = P_{\nu\mu}^{(0)} \geq 0$, $P_{\mu\mu}^{(0)} > 0 \quad \forall \mu, \nu$

Case 2: $n > 0$

Assumption 3: $P_{\mu\nu}^{(n-1)} = P_{\nu\mu}^{(n-1)} \geq 0$, $P_{\mu\mu}^{(n-1)} > 0 \quad \forall \mu, \nu$

Obviously, $\sum_\sigma P_{\mu\sigma}^{(n-1)} = \sum_\sigma P_{\sigma\mu}^{(n-1)} > 0$. By combining with $d_\mu > 0$,

$$\sqrt{\frac{d_\mu}{\sum_\sigma P_{\mu\sigma}^{(n-1)}} \cdot \frac{d_\nu}{\sum_\sigma P_{\sigma\nu}^{(n-1)}}} = \sqrt{\frac{d_\nu}{\sum_\sigma P_{\nu\sigma}^{(n-1)}} \cdot \frac{d_\mu}{\sum_\sigma P_{\sigma\mu}^{(n-1)}}} > 0$$

From $P_{\mu\nu}^{(n)} = P_{\mu\nu}^{(n-1)} \cdot \sqrt{d_\mu / \sum_\sigma P_{\mu\sigma}^{(n-1)} \cdot d_\nu / \sum_\sigma P_{\sigma\nu}^{(n-1)}}$ and Assumption 3,

$$P_{\mu\nu}^{(n)} = P_{\nu\mu}^{(n)} \geq 0, \quad P_{\mu\mu}^{(n)} > 0 \qquad \forall \mu, \nu$$

From Cases 1 and 2, Lemma 6 is valid for all $n \in \{0, 1, 2, \ldots\}$. ■

From Definition 3 and Lemma 6,

$$P_{\mu\nu}^{(n)} = W_{\mu\nu} k_\mu^{(n)} k_\nu^{(n)} \tag{B.4}$$

Lemma 7: $\lim_{n\to\infty} \sum_\nu P_{\mu\nu}^{(n)} = d_\mu$ for all μ is a sufficient condition to satisfy Theorem 2.

Proof:

$$\lim_{n\to\infty} \sum_\nu P_{\mu\nu}^{(n)} = d_\mu \quad \forall \mu$$

$$\Rightarrow \lim_{n\to\infty} \sum_\nu W_{\mu\nu} k_\mu^{(n)} k_\nu^{(n)} = d_\mu \quad \forall \mu \quad \left(\because \text{Eq.}\,(\text{B.}4)\right)$$

From Lemmas 4 and 5, the partial mapping $\vec{g}: D^N \nrightarrow D^N$ $(k_\mu^{(n)} \longmapsto \sum_\nu W_{\mu\nu} k_\mu^{(n)} k_\nu^{(n)})$ is a homeomorphism. Because k_μ is defined as the vector satisfying Eq. $(\text{B.}2)$, $\lim_{n\to\infty} k_\mu^{(n)} = k_\mu$ for all μ. ■

Because of Lemma 7, by using $\alpha_\mu^{(n)} \coloneqq (1/2) \ln\left(d_\mu / \sum_\nu P_{\mu\nu}^{(n)}\right)$, we attempt to show the following equation ($\ln$ is the natural logarithm):

$$\lim_{n\to\infty} \alpha_\mu^{(n)} = 0 \qquad \forall \mu \tag{B.5}$$

From Definition 3, $P_{\mu\nu}^{(n+1)} = P_{\mu\nu}^{(n)} e^{\alpha_\mu^{(n)}} e^{\alpha_\nu^{(n)}}$ (e is Napier's constant).

By substituting the right side for $P_{\mu\nu}^{(n+1)}$ in $\alpha_\mu^{(n+1)} = (1/2)\ln\left(d_\mu / \sum_\nu P_{\mu\nu}^{(n+1)}\right)$, the following equation is obtained:

$$\alpha_\mu^{(n+1)} = \frac{1}{2}\left(-\alpha_\mu^{(n)} + \ln\frac{d_\mu}{\sum_\nu P_{\mu\nu}^{(n)} e^{\alpha_\nu^{(n)}}}\right) \tag{B.6}$$

Here, because of $P_{\mu\nu}^{(n)} \geq 0$, the following inequality is satisfied $\left(\alpha_m^{(n)} \coloneqq \min_\mu \alpha_\mu^{(n)},\ \alpha_M^{(n)} \coloneqq \max_\mu \alpha_\mu^{(n)}\right)$:

$$e^{\alpha_m^{(n)}} \cdot \sum_\nu P_{\mu\nu}^{(n)} \ \leq\ \sum_\nu P_{\mu\nu}^{(n)} e^{\alpha_\nu^{(n)}} \ \leq\ e^{\alpha_M^{(n)}} \cdot \sum_\nu P_{\mu\nu}^{(n)} \tag{B.7}$$

Therefore, from Eq. (B.6), the following inequalities are derived:

$$\frac{\alpha_\mu^{(n)} - \alpha_M^{(n)}}{2} \leq \alpha_\mu^{(n+1)} \leq \frac{\alpha_\mu^{(n)} - \alpha_m^{(n)}}{2} \tag{B.8}$$

$$\Leftrightarrow\ \frac{\alpha_m^{(n)} - \alpha_M^{(n)}}{2} \leq \alpha_m^{(n+1)} \leq \alpha_M^{(n+1)} \leq \frac{\alpha_M^{(n)} - \alpha_m^{(n)}}{2} \tag{B.9}$$

$$\Rightarrow\ \alpha_M^{(n)} - \alpha_m^{(n)} \geq \alpha_M^{(n+1)} - \alpha_m^{(n+1)} \geq 0 \tag{B.10}$$

From Inequalities (B.10), there is a non-negative finite value ω which satisfies $\omega = \lim_{n\to\infty}\left(\alpha_M^{(n)} - \alpha_m^{(n)}\right)$.

Because $\alpha_M^{(n)} - \alpha_m^{(n)}$ monotonously decreases according to n, $\alpha_M^{(n+1)} - \alpha_m^{(n+1)} \geq \omega$.

Therefore, from Inequalities (B.9),

$$\frac{\alpha_m^{(n)} - \alpha_M^{(n)}}{2} \leq \alpha_m^{(n+1)} \leq \alpha_M^{(n+1)} - \omega \leq \frac{\alpha_M^{(n)} - \alpha_m^{(n)}}{2} - \omega$$

From $\lim_{n\to\infty}\left(\alpha_m^{(n)} - \alpha_M^{(n)}\right)/2 = \lim_{n\to\infty}\left[\left(\alpha_M^{(n)} - \alpha_m^{(n)}\right)/2 - \omega\right] = -\omega/2$,

$$\lim_{n\to\infty} \alpha_m^{(n)} = -\frac{\omega}{2} \tag{B.11}$$

Accordingly,

$$\lim_{n\to\infty} \alpha_M^{(n)} = \frac{\omega}{2} \tag{B.12}$$

Lemma 8: $P_{\mu\mu}^{(n)}$ does not converge to zero as $n \to \infty$ for all μ.

Proof:

Lemma 8 is proved by contradiction.

Assumption 4: $\exists \mu_1 \in \{1, 2, \dots, N\}$ s.t. $\lim_{n\to\infty} P_{\mu_1\mu_1}^{(n)} = 0$

From Assumption 4 and Eq. (B.4), $\lim_{n\to\infty} k_{\mu_1}^{(n)} = 0 \quad (\because W_{\mu_1\mu_1} > 0)$.

Here, from Eq. (B.12), $\alpha_{\mu_1}^{(n)}$ does not diverge to infinity as $n \to \infty$. Therefore, from the definition of $\alpha_{\mu}^{(n)}$, $\sum_\nu P_{\mu_1\nu}^{(n)}$ does not converge to zero. Accordingly, from $\sum_\nu P_{\mu_1\nu}^{(n)} = k_{\mu_1}^{(n)} \sum_\nu W_{\mu_1\nu} k_\nu^{(n)} \quad (\because \text{Eq.}(\text{B.}4))$,

$\sum_\nu W_{\mu_1\nu} k_\nu^{(n)} \to \infty \quad (n \to \infty) \left(\because k_{\mu_1}^{(n)} \to 0\right) \Rightarrow \exists \mu_2 \in \{1, 2, \dots, N\}$ s.t. $k_{\mu_2}^{(n)} \to \infty \ (n \to \infty)$.

Then, from $\sum_\nu P_{\mu_2\nu}^{(n)} = \sum_\nu W_{\mu_2\nu} k_{\mu_2}^{(n)} k_\nu^{(n)} \geq W_{\mu_2\mu_2} k_{\mu_2}^{(n)} k_{\mu_2}^{(n)} \to \infty \ (n \to \infty)$, $\alpha_{\mu_2}^{(n)} \to -\infty \ (n \to \infty)$. This is inconsistent with Eq. (B.11). Therefore, Assumption 4 cannot be realized.

■

When $m_n \coloneqq \operatorname*{argmin}_{\mu} \alpha_{\mu}^{(n)}$, the followings are derived:

$$\alpha_m^{(n)} = \alpha_{m_n}^{(n)}$$

$$= \frac{1}{2}\left(-\alpha_{m_n}^{(n-1)} + \ln \frac{d_{m_n}}{\sum_\nu P_{m_n\nu}^{(n-1)} e^{\alpha_\nu^{(n-1)}}}\right) \quad (\because \text{Eq. (B.6)})$$

$$\geq \frac{1}{2}\left(-\alpha_{m_n}^{(n-1)} + \ln \frac{d_{m_n}}{e^{\alpha_M^{(n-1)}} \sum_\nu P_{m_n\nu}^{(n-1)}}\right) \quad (\because \text{Inequalities (B.7)})$$

$$= \frac{\alpha_{m_n}^{(n-1)} - \alpha_M^{(n-1)}}{2} \geq \frac{\alpha_m^{(n-1)} - \alpha_M^{(n-1)}}{2}$$

$$\therefore \lim_{n\to\infty} \alpha_{m_n}^{(n-1)} = -\frac{\omega}{2} \quad (\because \text{Eqs. (B.11) and (B.12)}) \tag{B.13}$$

From $\lim_{n\to\infty} \alpha_m^{(n)} = \lim_{n\to\infty} \left(\alpha_m^{(n-1)} - \alpha_M^{(n-1)}\right)/2 = -\omega/2 \quad (\because \text{Eqs. (B.11) and (B.12)})$,

$$\lim_{n\to\infty}\left(-\alpha_{m_n}^{(n-1)} + \ln \frac{d_{m_n}}{\sum_\nu P_{m_n\nu}^{(n-1)} e^{\alpha_\nu^{(n-1)}}}\right) = \lim_{n\to\infty}\left(-\alpha_{m_n}^{(n-1)} + \ln \frac{d_{m_n}}{e^{\alpha_M^{(n-1)}} \sum_\nu P_{m_n\nu}^{(n-1)}}\right)$$

$$\Leftrightarrow \lim_{n\to\infty}\left(\ln \frac{d_{m_n}}{\sum_\nu P_{m_n\nu}^{(n-1)} e^{\alpha_\nu^{(n-1)}}}\right) = \lim_{n\to\infty}\left(\ln \frac{d_{m_n}}{e^{\alpha_M^{(n-1)}} \sum_\nu P_{m_n\nu}^{(n-1)}}\right) \quad (\because \text{Eq. (B.13)})$$

$$\Rightarrow \lim_{n\to\infty}\left(\ln\frac{d_{m_n}}{\sum_\nu P_{m_n\nu}^{(n-1)}e^{\alpha_\nu^{(n-1)}}}-\ln\frac{d_{m_n}}{e^{\alpha_M^{(n-1)}}\sum_\nu P_{m_n\nu}^{(n-1)}}\right)=0$$

$$\Leftrightarrow \lim_{n\to\infty}\left(\ln\left(e^{\alpha_M^{(n-1)}}\right)-\ln\frac{\sum_\nu P_{m_n\nu}^{(n-1)}e^{\alpha_\nu^{(n-1)}}}{\sum_\nu P_{m_n\nu}^{(n-1)}}\right)=0 \tag{B.14}$$

Here, $\sum_\nu P_{m_n\nu}^{(n-1)}e^{\alpha_\nu^{(n-1)}}/\sum_\nu P_{m_n\nu}^{(n-1)}$ is a weighted average of $\left\{e^{\alpha_\nu^{(n-1)}}\middle|\nu\right\}$. Because $e^{\alpha_M^{(n-1)}}$ is the maximum value in $\left\{e^{\alpha_\nu^{(n-1)}}\middle|\nu\right\}$ and $P_{m_nm_n}^{(n-1)}$ does not converge to zero as $n\to\infty$ ($\because$ Lemma 8), $e^{\alpha_{m_n}^{(n-1)}}$ is necessary to approach $e^{\alpha_M^{(n-1)}}$ infinitely with $n\to\infty$ to satisfy Eq. $(\mathrm{B.14})$. Therefore, the following equations are valid:

$$\lim_{n\to\infty}\left(e^{\alpha_M^{(n-1)}}-e^{\alpha_{m_n}^{(n-1)}}\right)=0$$

$$\Leftrightarrow \lim_{n\to\infty}\alpha_{m_n}^{(n-1)}=\lim_{n\to\infty}\alpha_M^{(n-1)}=\frac{\omega}{2}\quad\left(\because \mathrm{Eq.(B.12)}\right) \tag{B.15}$$

From Eqs. $(\mathrm{B.13})$ and $(\mathrm{B.15})$, $\omega=0$ $(\because -\omega=\omega)$. Therefore, from Eqs. $(\mathrm{B.11})$ and $(\mathrm{B.12})$, Eq. $(\mathrm{B.5})$ is valid. ■

Appendix C. Supplementary data

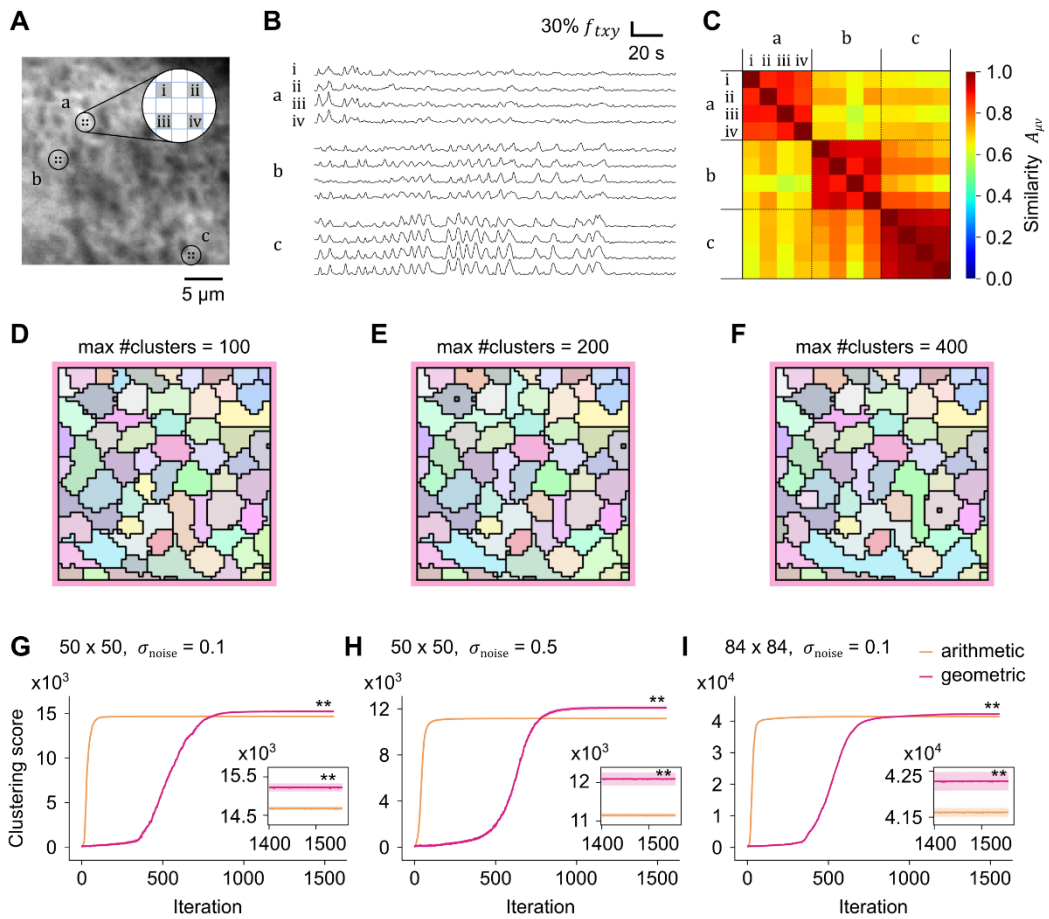

Fig. C.1. Validation of the cosine-based similarity and the geometric progression-based annealing schedule. (A) Basal fluorescence of the calcium imaging of the neuropil of a fly larva. Three boutons (a, b, and c) were marked. For each bouton, four pixels (i-iv) were sampled. (B) Preprocessed signals of the sampled pixels in (A). See section 2.2 for the preprocessing procedure. (C) Signal similarity between the pixels in (A), calculated as cosine between signal traces in (B), assumed as vectors. (D, E, and F) Clustering with a different maximum number of clusters (D: 100, E: 200, and F: 400). (G, H, and I) Clustering score development during simulated annealing. Magenta indicates geometric progression of inverse temperature beta, whereas orange indicates arithmetic progression. Beta reached 20 at the final iteration 1,557 in both progressions. Shaded regions show the standard deviation of five trials. The final clustering score of geometric progression was significantly

larger than that of arithmetic progression (**: $p < 10^{-2}$, Mann-Whitney U test). (G) Artificial data of 50 x 50 pixel array and sig_noise = 0.1. (H) Artificial data of 50 x 50 pixel array and σ_{noise} = 0.5. (I) Artificial data of 84 x 84 pixel array and σ_{noise} = 0.1.

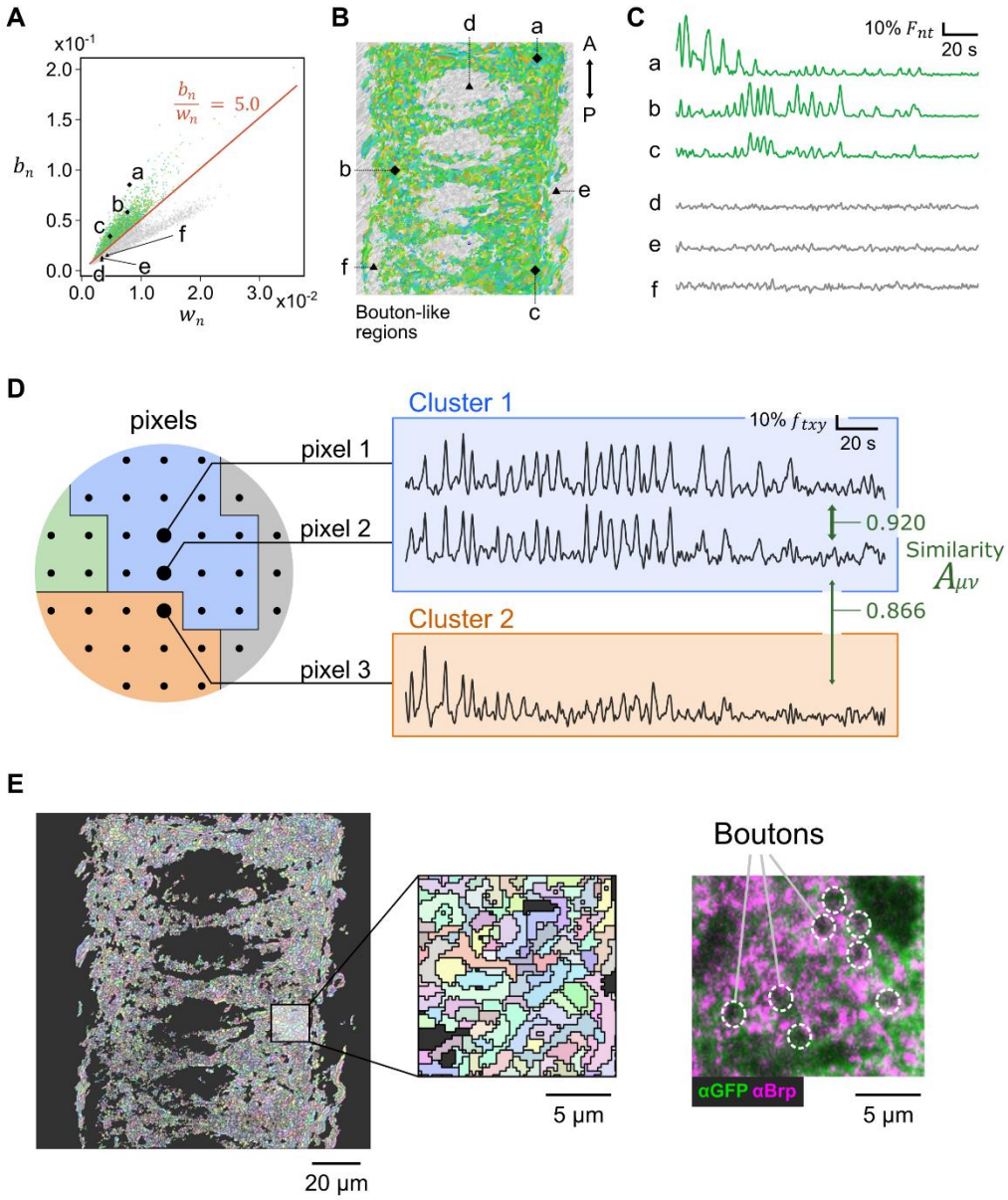

Fig. C.2. Pixels in imaging data were clustered into boutons. (A, B, and C) Detected bouton-like regions with sufficient (green) and insufficient (gray) S/N ratio from one sample. Regions a, b, and c showed sufficient S/N ratio, whereas d, e, and f didn't. (A) Scatter diagram of bouton-like regions showing the intensity of noise (horizontal) and signals (vertical). Orange

line is the threshold level of the S/N ratio. (B) Spatial configuration of the bouton-like regions. (C) Signal traces of the regions a-f. (D) Examples of activity traces of pixels. Left: Spatial configuration of pixels colored based on their cluster identity. Pixels 1 and 2 were included in cluster 1 (blue), whereas pixel 3 belonged to cluster 2 (orange). Right: Temporal traces of pixels 1, 2, and 3. The numbers colored in green are the values in the similarity matrix (see section 2.1.1.). (E) The sizes of extracted patches and actual boutons were comparable. Left: Bouton-like regions. Each region is shown by a different color. Middle: Expanded image of the white square in (E). Right: Immunostaining image of the neuropil with anti-GFP and anti-Brp (the marker of presynaptic sites), shown on the same scale as the middle panel. Dashed circles indicate boutons.

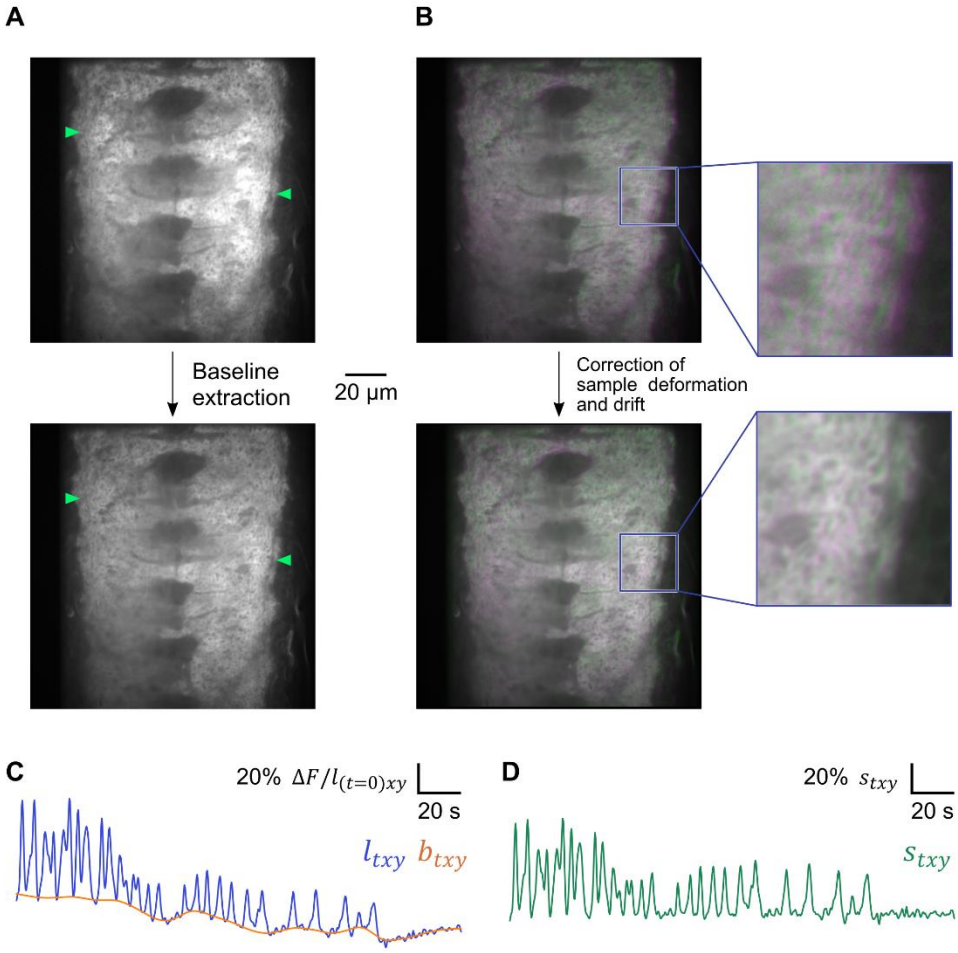

Fig. C.3. Baseline detection and correction. (A) An imaging frame showing neural activity (top) and the detected temporal baselines sectioned on the same frame (bottom). Top and bottom panels are shown in the same contrast. Green arrowheads indicate regions showing strong neural activity in the top panel. (B) Calcium imaging on two distinct frames (green: $t = 245$, magenta: $t = 699$) is merged. Top and bottom panels show images before and after correction of sample deformation and drift, respectively. Right panels are expansion images of left panels, showing misalignment was drastically reduced after the processing. (C) An activity trace of one pixel and detected baseline. The blue line is the original data, and the orange line is the detected baseline. (D) Baseline-corrected trace of the same pixel in (C).

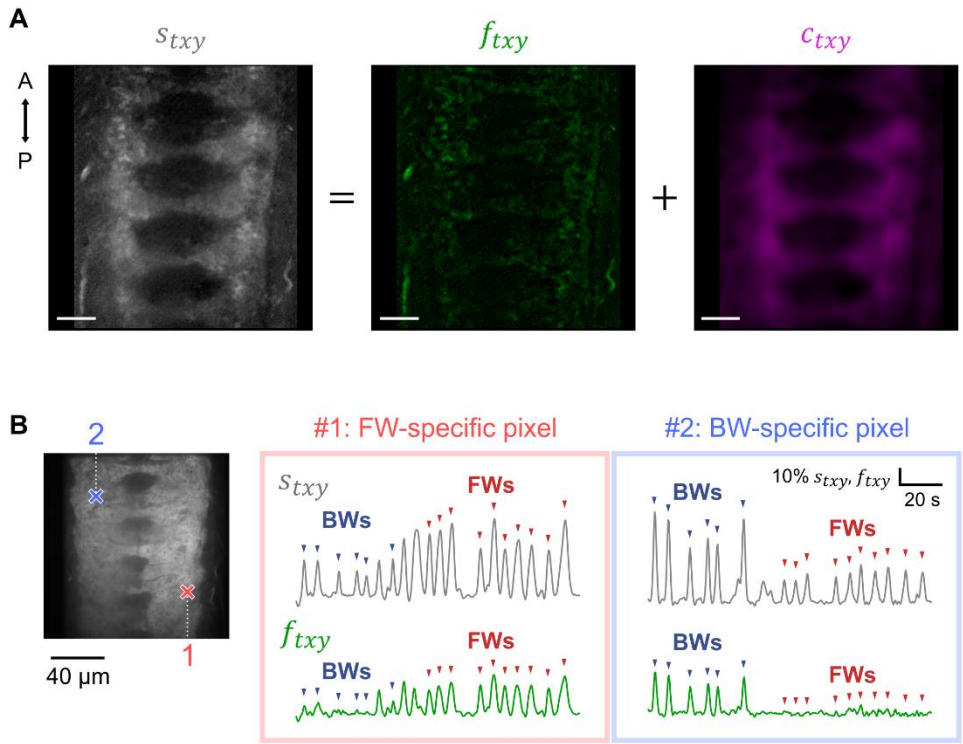

Fig. C.4. Calcium imaging data were successfully decomposed into well-focused and out-of-focus structures. (A) Fine-coarse decomposition on one example frame. Gray image shows original signals, while green and magenta images show decomposed fine and coarse components, respectively. Scale bar: 20 μm. (B) Forward- or backward-specific pixels can be recognized only in fine components. FW is a forward wave, and BW is a backward wave. Pixels #1 and #2 show FW-specific and BW-specific activity, respectively. Left: Spatial sites

of pixels #1 and #2. Right: Temporal traces of pixels #1 and #2. Gray lines are original signals, while green lines are signals of extracted fine components. Red and blue arrowheads indicate FW and BW activity patterns, respectively.

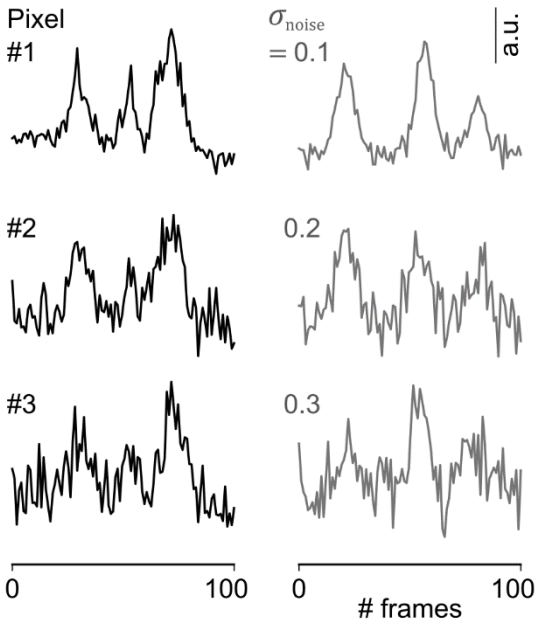

Fig. C.5. Comparison of biological calcium recording data and artificial data. Left: Temporal profiles of three example pixels in biological calcium imaging data. Right: Artificial burst-like signals of three distinct noise intensities (σ_{noise}). The noise intensities of the real calcium imaging data corresponded to σ_{noise} = 0.1-0.3 in the artificial data.

References

Arenas, A., Fernández, A., & Gómez, S. (2008). Analysis of the structure of complex networks at different resolution levels. *New Journal of Physics*, *10*. https://doi.org/10.1088/1367-2630/10/5/053039

Baek, S.-J., Park, A., Ahn, Y.-J., & Choo, J. (2015). Baseline correction using asymmetrically reweighted penalized least squares smoothing. *The Analyst*, *140*(1), 250–257. https://doi.org/10.1039/c4an01061b

Barabási, A. L., & Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, *286*, 509–512.

Bertinetto, C. G., & Vuorinen, T. (2014). Automatic baseline recognition for the correction of large sets of spectra using continuous wavelet transform and iterative fitting. *Applied Spectroscopy*, *68*(2), 155–164. https://doi.org/10.1366/13-07018

Bilz, F., Geurten, B. R. H., Hancock, C. E., Widmann, A., & Fiala, A. (2020). Visualization of a Distributed Synaptic Memory Code in the Drosophila Brain. *Neuron*, 963–976. https://doi.org/10.1016/j.neuron.2020.03.010

Bruno, A. M., Frost, W. N., & Humphries, M. D. (2015). Modular deconstruction reveals the dynamical and physical building blocks of a locomotion motor program. *Neuron*, *86*(1), 304–318. https://doi.org/10.1016/j.neuron.2015.03.005

Cao, J., Bu, Z., Gao, G., & Tao, H. (2016). Weighted modularity optimization for crisp and fuzzy community detection in large-scale networks. *Physica A: Statistical Mechanics and Its Applications*, *462*, 386–395. https://doi.org/10.1016/j.physa.2016.06.113

Carreira-Rosario, A., Zarin, A. A., Clark, M. Q., Manning, L., Fetter, R., Cardona, A., & Doe, C. Q. (2018). A command-like descending neuron that coordinately activates backward and inhibits forward locomotion. *BioRxiv*, 1–28. https://doi.org/10.1101/339556

Chen, J., Zaïane, O. R., & Goebel, R. (2009). Detecting communities in social networks using max-min modularity. *Society for Industrial and Applied Mathematics - 9th SIAM International Conference on Data Mining 2009, Proceedings in Applied Mathematics*, *2*, 973–984. https://doi.org/10.1137/1.9781611972795.84

Churchland, M. M., Cunningham, J. P., Kaufman, M. T., Foster, J. D., Nuyujukian, P., Ryu, S. I., Shenoy, K. V., & Shenoy, K. V. (2012). Neural population dynamics during reaching. *Nature*, *487*(7405), 51–56. https://doi.org/10.1038/nature11129

Clauset, A., Newman, M. E. J., & Moore, C. (2004). Finding community structure in very large networks. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, *70*(6), 6. https://doi.org/10.1103/PhysRevE.70.066111

Cohen, R., Erez, K., ben Avraham, D., & Havlin, S. (2000). Resilience of the Internet to Random Breakdowns. *Physical Review Letters*, *85*(21), 4626–4628. https://doi.org/10.1016/0273-

1177(91)90013-A

Duch, J., & Arenas, A. (2005). Community detection in complex networks using extremal optimization. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *72*(2), 1–4. https://doi.org/10.1103/PhysRevE.72.027104

Eguíluz, V. M., Chialvo, D. R., Cecchi, G. A., Baliki, M., & Apkarian, A. V. (2005). Scale-free brain functional networks. *Physical Review Letters*, *94*(1), 1–4. https://doi.org/10.1103/PhysRevLett.94.018102

Eilers, P. H. C. (2003). A perfect smoother. *Analytical Chemistry*, *75*(14), 3631–3636. https://doi.org/10.1021/ac034173t

Fine, P., Di Paolo, E., & Philippides, A. (2006). Spatially constrained networks and the evolution of modular control systems. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *4095 LNAI*, 546–557. https://doi.org/10.1007/11840541_45

Fortunato, S., & Barthélemy, M. (2007). Resolution limit in community detection. *Proceedings of the National Academy of Sciences of the United States of America*, *104*(1), 36–41. https://doi.org/10.1073/pnas.0605965104

Fushiki, A., Zwart, M. F., Kohsaka, H., Fetter, R. D., Cardona, A., & Nose, A. (2016). A circuit mechanism for the propagation of waves of muscle contraction in Drosophila. *ELife*, *5*(FEBRUARY2016), 1–23. https://doi.org/10.7554/eLife.13253

Gan, F., Ruan, G., & Mo, J. (2006). Baseline correction by improved iterative polynomial fitting with automatic threshold. *Chemometrics and Intelligent Laboratory Systems*, *82*(1-2 SPEC. ISS), 59–65. https://doi.org/10.1016/j.chemolab.2005.08.009

Gauthier, J. L., & Tank, D. W. (2018). A Dedicated Population for Reward Coding in the Hippocampus. *Neuron*, *99*(1), 179-193.e7. https://doi.org/10.1016/j.neuron.2018.06.008

Giovannucci, A., Friedrich, J., Gunn, P., Kalfon, J., Brown, B. L., Koay, S. A., Taxidis, J., Najafi, F., Gauthier, J. L., Zhou, P., Khakh, B. S., Tank, D. W., Chklovskii, D. B., & Pnevmatikakis, E. A. (2019). Caiman an open source tool for scalable calcium imaging data analysis. *ELife*, *8*, 1–45. https://doi.org/10.7554/eLife.38173

Gründemann, J., Bitterman, Y., Lu, T., Krabbe, S., Grewe, B. F., Schnitzer, M. J., & Lüthi, A. (2019). Amygdala ensembles encode behavioral states. *Science*, *364*(6437). https://doi.org/10.1126/science.aav8736

Guimerà, R., Sales-Pardo, M., & Amaral, L. A. N. (2004). Modularity from fluctuations in random graphs and complex networks. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, *70*(2), 4. https://doi.org/10.1103/PhysRevE.70.025101

Hackett, T. A., Barkat, T. R., O'Brien, B. M. J., Hensch, T. K., & Polley, D. B. (2011). Linking

topography to tonotopy in the mouse auditory thalamocortical circuit. *Journal of Neuroscience*, *31*(8), 2983–2995. https://doi.org/10.1523/JNEUROSCI.5333-10.2011

Haq, N. F., Moradi, M., & Wang, Z. J. (2019). Community structure detection from networks with weighted modularity. *Pattern Recognition Letters*, *122*, 14–22. https://doi.org/10.1016/j.patrec.2019.02.005

He, S., Zhang, W., Liu, L., Huang, Y., He, J., Xie, W., Wu, P., & Du, C. (2014). Baseline correction for Raman spectra using an improved asymmetric least squares method. *Analytical Methods*, *6*(12), 4402–4407. https://doi.org/10.1039/c4ay00068d

Hiramoto, A., Jonaitis, J., Niki, S., & Kohsaka, H. (2021). Regulation of coordinated muscular relaxation by a pattern- generating intersegmental circuit. *Nature Communications*, 1–14. https://doi.org/10.1038/s41467-021-23273-y

Humphries, M. D. (2011). Spike-train communities: Finding groups of similar spike trains. *Journal of Neuroscience*, *31*(6), 2321–2336. https://doi.org/10.1523/JNEUROSCI.2853-10.2011

Kirkpatrick, S., Gelatt, Jr., C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, *220*(4598).

Kohsaka, H., Takasu, E., Morimoto, T., & Nose, A. (2014). A group of segmental premotor interneurons regulates the speed of axial locomotion in drosophila larvae. *Current Biology*, *24*(22), 2632–2642. https://doi.org/10.1016/j.cub.2014.09.026

Kohsaka, H., Zwart, M. F., Fushiki, A., Fetter, R. D., Truman, J. W., Cardona, A., & Nose, A. (2019). Regulation of forward and backward locomotion through intersegmental feedback circuits in Drosophila larvae. *Nature Communications*, *10*(1), 1–11. https://doi.org/10.1038/s41467-019-10695-y

Kreiner, T., Kirk, M. D., & Scheller, R. H. (1987). Cellular and synaptic morphology of a feeding motor circuit in Aplysia californica. *Journal of Comparative Neurology*, *264*(3), 311–325. https://doi.org/10.1002/cne.902640304

Lan, T., Fang, Y., Xiong, W., & Kong, C. (2007). Automatic baseline correction of infrared spectra. *Chinese Optics Letters*, *5*(10), 613–616.

Lancichinetti, A., & Fortunato, S. (2011). Limits of modularity maximization in community detection. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *84*(6), 1–8. https://doi.org/10.1103/PhysRevE.84.066122

Lee, S., Meyer, J. F., Park, J., & Smirnakis, S. M. (2017). Visually driven neuropil activity and information encoding in mouse primary visual cortex. *Frontiers in Neural Circuits*, *11*(July), 1–18. https://doi.org/10.3389/fncir.2017.00050

Leger, M. N., & Ryder, A. G. (2006). Comparison of derivative preprocessing and automated polynomial baseline correction method for classification and quantification of narcotics in

solid mixtures. *Applied Spectroscopy*, *60*(2), 182–193. https://doi.org/10.1366/000370206776023304

Lemon, W. C., Pulver, S. R., Höckendorf, B., McDole, K., Branson, K., Freeman, J., & Keller, P. J. (2015). Whole-central nervous system functional imaging in larval Drosophila. *Nature Communications*, *6*(May). https://doi.org/10.1038/ncomms8924

Li, Z., Zhang, S., Wang, R. S., Zhang, X. S., & Chen, L. (2008). Quantitative function for community detection. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *77*(3), 1–9. https://doi.org/10.1103/PhysRevE.77.036109

Lieber, C. A., & Mahadevan-Jansen, A. (2003). Automated Method for Subtraction of Fluorescence from Biological Raman Spectra. *Applied Spectroscopy*, *57*(11), 1363–1367. https://doi.org/10.1366/000370203322554518

London, M., & Häusser, M. (2005). Dendritic computation. *Annual Review of Neuroscience*, *28*, 503–532. https://doi.org/10.1146/annurev.neuro.28.061604.135703

Lu, X., Kuzmin, K., Chen, M., & Szymanski, B. K. (2018). Adaptive modularity maximization via edge weighting scheme. *Information Sciences*, *424*, 55–68. https://doi.org/10.1016/j.ins.2017.09.063

Maruyama, R., Maeda, K., Moroda, H., Kato, I., Inoue, M., Miyakawa, H., & Aonishi, T. (2014). Detecting cells using non-negative matrix factorization on calcium imaging data. *Neural Networks*, *55*, 11–19. https://doi.org/10.1016/j.neunet.2014.03.007

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, *21*(6), 1087–1092. https://doi.org/10.1063/1.1699114

Mölter, J., Avitan, L., & Goodhill, G. J. (2018). Detecting neural assemblies in calcium imaging data. *BMC Biology*, *16*(1), 1–20. https://doi.org/10.1186/s12915-018-0606-4

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

Muff, S., Rao, F., & Caflisch, A. (2005). Local modularity measure for network clusterizations. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *72*(5), 1–4. https://doi.org/10.1103/PhysRevE.72.056107

Mukamel, E. A., Nimmerjahn, A., & Schnitzer, M. J. (2009). Automated Analysis of Cellular Signals from Large-Scale Calcium Imaging Data. *Neuron*, *63*(6), 747–760. https://doi.org/10.1016/j.neuron.2009.08.009

Newman, M. E. J. (2004). Fast algorithm for detecting community structure in networks.

Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics, *69*(6), 5. https://doi.org/10.1103/PhysRevE.69.066133

Newman, M. E. J. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences of the United States of America*, *103*(23), 8577–8582. https://doi.org/10.1073/pnas.0601602103

Newman, M. E. J., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *69*(2 2), 1–15. https://doi.org/10.1103/PhysRevE.69.026113

Otsu, N. (1979). A Threshold selection method from gray-Level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*(1), 62–66.

Pastor-Satorras, R., & Vespignani, A. (2001). Epidemic spreading in scale-free networks. *Physical Review Letters*, *86*(14), 3200–3203. https://doi.org/10.1103/PhysRevLett.86.3200

Pnevmatikakis, E. A., Soudry, D., Gao, Y., Machado, T. A., Merel, J., Pfau, D., Reardon, T., Mu, Y., Lacefield, C., Yang, W., Ahrens, M., Bruno, R., Jessell, T. M., Peterka, D. S., Yuste, R., & Paninski, L. (2016). Simultaneous Denoising, Deconvolution, and Demixing of Calcium Imaging Data. *Neuron*, *89*(2), 285. https://doi.org/10.1016/j.neuron.2015.11.037

Poirazi, P., & Papoutsi, A. (2020). Illuminating dendritic function with computational models. *Nature Reviews Neuroscience*, *21*(6), 303–321. https://doi.org/10.1038/s41583-020-0301-7

Qian, F., Wu, Y., & Hao, P. (2017). A fully automated algorithm of baseline correction based on wavelet feature points and segment interpolation. *Optics and Laser Technology*, *96*, 202–207. https://doi.org/10.1016/j.optlastec.2017.05.021

Reichardt, J., & Bornholdt, S. (2006). Statistical mechanics of community detection. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, *74*(1), 1–14. https://doi.org/10.1103/PhysRevE.74.016110

Reynolds, S., Abrahamsson, T., Schuck, R., Jesper Sjöström, P., Schultz, S. R., & Dragotti, P. L. (2017). ABLE: An activity-based level set segmentation algorithm for two-photon calcium imaging data. *ENeuro*, *4*(5), 1–13. https://doi.org/10.1523/ENEURO.0012-17.2017

Rosvall, M., & Bergstrom, C. T. (2013). An information-theoretic framework for resolving community structure in complex networks. *International Journal of Web Based Communities*, *9*(3), 130. http://dl.acm.org/citation.cfm?id=2499306

Schulze, G., Jirasek, A., Yu, M. M. L., Lim, A., Turner, R. F. B., & Blades, M. W. (2005). Investigation of selected baseline removal techniques as candidates for automated implementation. *Applied Spectroscopy*, *59*(5), 545–574. https://doi.org/10.1366/0003702053945985

Shao, L., & Griffiths, P. R. (2007). Automatic baseline correction by wavelet transform for

quantitative open-path fourier transform infrared spectroscopy. *Environmental Science and Technology*, *41*(20), 7054–7059. https://doi.org/10.1021/es062188d

Shibue, R., & Komaki, F. (2020). Deconvolution of calcium imaging data using marked point processes. *PLoS Computational Biology*, *16*(3), 1–25. https://doi.org/10.1371/journal.pcbi.1007650

Sporns, O., & Betzel, R. F. (2016). Modular brain networks. *Annual Review of Psychology*, *67*, 613–640. https://doi.org/10.1146/annurev-psych-122414-033634

Stam, C. J., & Reijneveld, J. C. (2007). Graph theoretical analysis of complex networks in the brain. *Nonlinear Biomedical Physics*, *1*, 1–19. https://doi.org/10.1186/1753-4631-1-3

Tinnermann, A., Büchel, C., & Cohen-Adad, J. (2021). Cortico-spinal imaging to study pain. *NeuroImage*, *224*(October 2020), 117439. https://doi.org/10.1016/j.neuroimage.2020.117439

Torre, G., Fernández-Lugo, S., Guarino, R., & Fernández-Palacios, J. M. (2019). Network analysis by simulated annealing of taxa and islands of Macaronesia (North Atlantic Ocean). *Ecography*, *42*(4), 768–779. https://doi.org/10.1111/ecog.03909

van den Heuvel, M. P., Stam, C. J., Boersma, M., & Hulshoff Pol, H. E. (2008). Small-world and scale-free organization of voxel-based resting-state functional connectivity in the human brain. *NeuroImage*, *43*(3), 528–539. https://doi.org/10.1016/j.neuroimage.2008.08.010

Wang, Y., Hu, Y., Di, Z., & Fan, Y. (2011). The effect of hub nodes on the community structure in scale-free networks. *Physica A: Statistical Mechanics and Its Applications*, *390*(21–22), 4027–4033. https://doi.org/10.1016/j.physa.2011.06.031

White, S., & Smyth, P. (2005). A spectral clustering approach to finding communities in graphs. *Proceedings of the 2005 SIAM International Conference on Data Mining, SDM 2005*, 274–285. https://doi.org/10.1137/1.9781611972757.25

Zhang, F., Tang, X., Tong, A., Wang, B., Wang, J., Lv, Y., Tang, C., & Wang, J. (2020). Baseline correction for infrared spectra using adaptive smoothness parameter penalized least squares method. *Spectroscopy Letters*, *53*(3), 222–233. https://doi.org/10.1080/00387010.2020.1730908

Zhang, Z. M., Chen, S., & Liang, Y. Z. (2010). Baseline correction using adaptive iteratively reweighted penalized least squares. *Analyst*, *135*(5), 1138–1146. https://doi.org/10.1039/b922045c

Zhao, J., Lui, H., Mclean, D. I., & Zeng, H. (2007). Automated autofluorescence background subtraction algorithm for biomedical raman spectroscopy. *Applied Spectroscopy*, *61*(11), 1225–1232. https://doi.org/10.1366/000370207782597003

Zhuang, J., Ng, L., Williams, D., Valley, M., Li, Y., Garrett, M., & Waters, J. (2017). An extended retinotopic map of mouse cortex. *ELife*, *6*, 1–29. https://doi.org/10.7554/eLife.18372