Multi-area functional modules mediate feedforward and recurrent processing in visual cortical hierarchy

Xiaoxuan Jia[1]*, Joshua H. Siegle[1], Séverine Durand[1], Greggory Heller[1], Tamina Ramirez[1], Shawn R. Olsen[1]*

[1]Allen Institute, Seattle, WA 98109, USA.

*Correspondence to: jxiaoxuan@gmail.com, shawno@alleninstitute.org

Abstract

The visual cortex is organized hierarchically, but extensive recurrent pathways make it challenging to decipher the flow of information with single neuron resolution. Here, we characterize spiking interactions in populations of neurons from six interconnected areas along the visual hierarchy in awake mice. We generated multi-area, directed graphs of neuronal communication and uncovered two spatially-distributed functional modules. One module is positioned to transmit feedforward sensory signals along the hierarchy, while the other receives convergent input and engages in recurrent processing. The modules differ in layer and area distributions, convergence and divergence, and population-level temporal dynamics. These results reveal a neuronal-resolution cortical network topology in which distinct processing modules are interlaced across multiple areas of the cortical hierarchy.

Main text

Sensory information processing in the neocortex involves signal representation, transformation, and transmission between processing levels or modules (*1*). Whereas stimulus representation (such as single neuron tuning preference and population coding) has been extensively studied (*2–6*), the principles by which spiking signals are transmitted between neuronal populations in the cortical network are unclear (*7*). The mammalian visual system, from mice to primates, is organized as a hierarchy of areas (*8*, *9*). Cortical areas at different levels of the hierarchy are thought to represent distinct stages of processing linked by feedforward (FF) and feedback (FB) projections. However, signal transmission is unlikely to follow a simple path up and down the hierarchy due to substantial recurrent connections between areas at different hierarchical levels (*9*, *10*). For example, the primary visual cortex (V1) has projections to all higher visual areas (*9*, *11*), and single cells can project in parallel to multiple areas via branching axons (*12*). Given the complexity of inter-area connections, the logic of signal propagation at the cellular-scale is hard to elucidate. Tracking signal propagation in such a distributed and recurrent network requires measurement of large numbers of interacting neurons both within and between cortical areas (*13*), combined with computational tools to reveal the logic of these interactions.

We previously built a recording platform with multiple Neuropixels probes to make simultaneous spike measurements from large populations of neurons in the mouse brain and have used this to generate a massive open database of spiking activity across six areas of the visual cortical hierarchy in the mouse brain (*14*) (Fig. 1A). Here we combine functional connectivity analysis, clustering algorithms, and network analysis (*15*) to discover functional modules of neurons that differentially process signals in this visual cortical network.

In the mouse visual system, each cortical area has its own map of visual space (*16*), and resides at a different hierarchical level as determined both anatomically (*9*, *17*) and physiologically (*14*): area V1 is at the bottom of the hierarchy, followed by areas RL/LM, AL, PM, and AM at the top (Fig. 1A). Each of the experimental recording sessions in this study yielded 632 ± 18 simultaneously recorded neurons (a.k.a. sorted units (*14*)) distributed across cortical layers and areas ($n = 19$ mice, mean ± SEM) (Fig. 1B). To facilitate functional connectivity analysis, neurons were filtered by firing rate and receptive field location ($n = 3487$; 29% of total units recorded across all mice; see Methods and Fig. S1).

Full-field drifting grating stimuli were used to provide a strong sensory input to the system. Consistent with the known visual hierarchy in the mouse (*9*, *14*), the mean response latency in each area followed a sequential progression (Fig. 1C), yet all areas were co-active for substantial portions of the sensory response, thereby providing opportunities for recurrent interactions.

To characterize functional interactions relevant to signal transmission during sensory drive, in each mouse we quantified spiking correlations between all

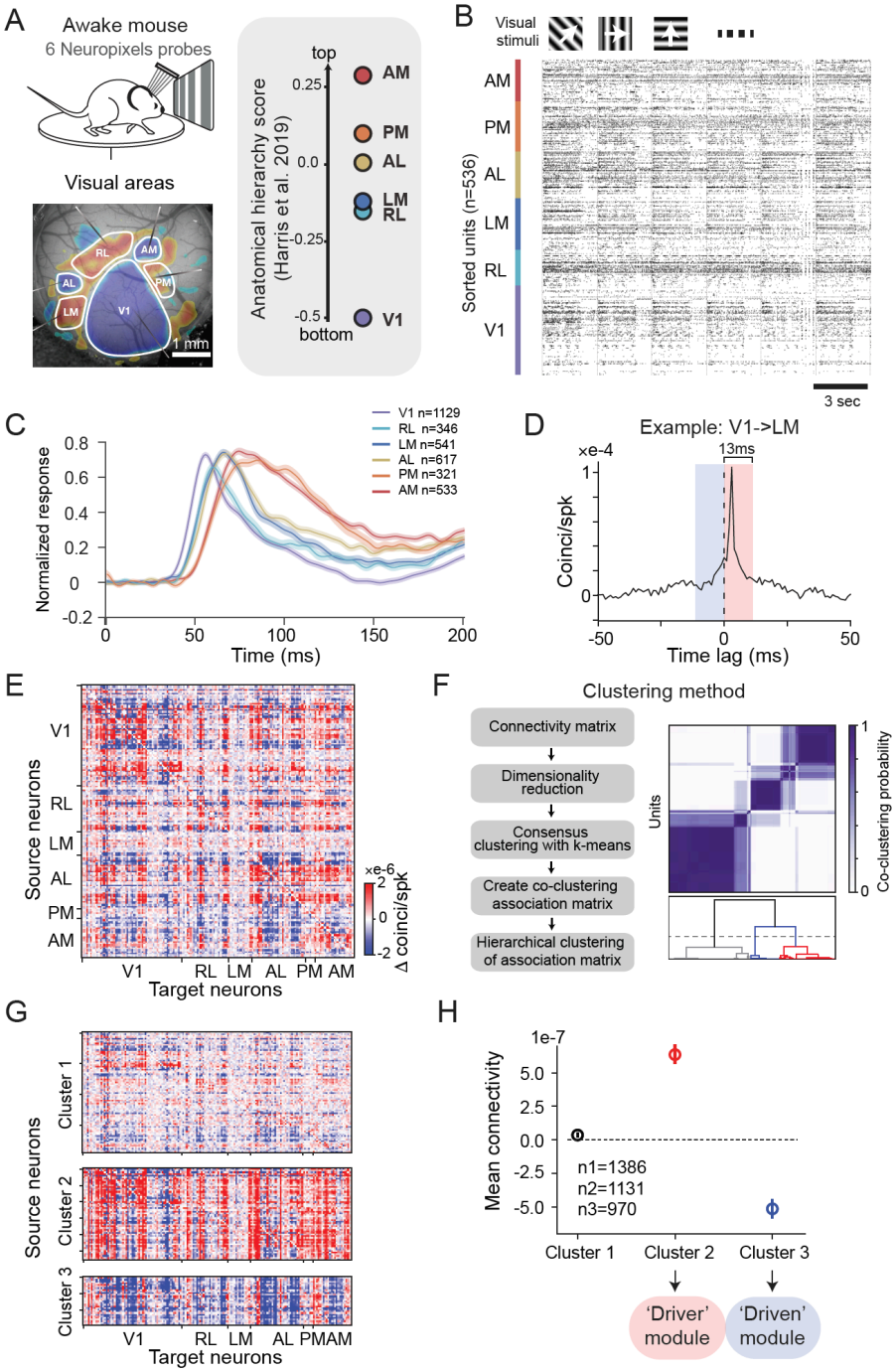


Figure 1. Identification of functional modules in multi-area spike recordings. A) *Top*: schematic of experimental setup. *Bottom*: Neuropixels recordings from 6 visual cortical areas of mouse left hemisphere. Retinotopic sign map is overlaid on vasculature image to guide area targeting. *Right*: replotting of anatomical hierarchy score from *(9)* Fig. 6e. B) Raster plot of 536 simultaneously recorded neurons during drifting gratings stimulation (6 sequential trials are shown, each with 2 s grating presentation followed by 1 s gray period). C) Normalized PSTHs in response to grating stimuli from 6 visual cortical areas for selected units. D) Example jitter-corrected CCG between a V1 and an LM neuron (peak offset = 3 ms). The connection weight is the difference of the CCG in a 13 ms window before and after 0 time lag. E) Directed cellular-scale connectivity matrix from one example mouse. Neurons are sorted by area and depth. F) Clustering procedure and result from example mouse. G) Connectivity profiles of three functional clusters from the example mouse. H) Functional connection strength averaged over all neurons in each cluster across all mice (n = 19 mice; Error bars show 95% confidence interval). Mean connection strength of cluster 1 is not significantly different from 0 (1-sample t-test, p = 0.02). Cluster 2 is significantly positive (p = 1.3e-62). Cluster 3 is significantly negative (p = 1.5e-43).

pairs of neurons using jitter-corrected cross-correlogram (CCG) analysis, which captures relative spike timing between two neurons within the jitter window (25 ms) but removes stimulus-locked signals and correlations longer than the jitter window (*18*, *19*). For each neuronal pair we quantified a connection weight by computing the difference of the CCG in a 13 ms window before and after 0 time lag, which reflects functional interactions within that delay window (Fig. 1D).The sign of this weight describes the signal flow direction (temporally leading or following) between pairs of neurons. Computing this for all pairs produced a connectivity matrix describing the directed, brief timescale functional interactions of all simultaneously recorded neurons in each mouse (Fig. 1E). These functional connectivity matrices displayed non-random structure (Fig. S2), and inspection suggested there existed separable modular components consisting of groups of neurons sharing similar patterns of functional connectivity.

To systematically identify sets of source neurons with shared functional connectivity patterns, we clustered the connectivity matrix by treating connections from each source neuron to all target neurons as features (Fig. 1E,F; Fig. S3; see Methods for details). In each mouse, this procedure yielded three robust clusters of source neurons (Fig. S4 and Fig. S5): cluster 1 mostly had weak connections; neurons in cluster 2 were dominated by strong positive connection weights indicating they tended to drive/lead the network activity, whereas neurons in cluster 3 were dominated by strong negative connection weights indicating they are mainly driven-by/follow activity in the network (Fig. 1G). Given the bias in the proportion of positive and negative connection weights from each cluster (and its functional implication of directionality), we refer to the set of cluster 2 source neurons as the 'driver' module and the set of cluster 3 source neurons as the 'driven' module (Fig. 1H). Supporting the robustness of these clusters, we observed similar network modules using spectral clustering and bi-clustering algorithms (*20*) (Fig. S4).

Graph theory suggests network topology constrains patterns of signal flow (*21*). Network convergence (*input* to one neuron from many others) and divergence (*output* from one neuron to many others) are two network motifs that differentially support the distribution versus integration of signals (*22*). In our recordings, the mean connection weight of the neurons in the two modules indicates they likely have distinct convergence and divergence properties. To quantify this directly, we computed convergence and divergence of individual neurons within each module. The divergence of each source neuron was determined by the fraction of the network to which it provided input (weight > 10^{-6}; threshold value was defined by half of the standard deviation of the weight distribution across all mice; see Methods); likewise, the convergence was computed as the fraction of neurons in the network providing inputs to each neuron (weight < -10^{-6}). The driver module had higher divergence than the driven module (Fig. 2A; 2-way ANOVA across areas, between modules $F = 1058.6$, $p = 3e-188$; among areas $F = 33.2$, $p = 1e-32$; interaction $F = 3.0$, $p = 0.013$). In contrast, the driven module had higher convergence (Fig. 2B; between modules $F = 239.9$, $p = 5.6e-51$; among areas $F = 20.6$, $p = 3.9e-20$; interaction $F = 3.1$, $p = 0.007$). Consequently, neurons in the driver module are better positioned to distribute information, whereas neurons in the driven module are better positioned for signal integration. This result reinforces the notion that the driver and driven modules perform distinct network operations.

In which cortical areas and layers do neurons in these separate modules reside? Neurons in both modules were spatially distributed across all levels of the cortical hierarchy rather than being localized to specific regions (Fig. 2C). Nonetheless, the relative proportion of neurons in the two modules showed area biases. Overall, the fraction of driver neurons gradually decreased along the hierarchy (Fig. 2C, right; Spearman's correlation with each area's hierarchical position (*9*): $r = -0.89$, $p = 0.019$), whereas driven neurons increased (Spearman's correlation = 0.89, $p = 0.019$ for driven module).

Both driver and driven neurons were present across cortical layers, but showed laminar biases. Driver neurons were enriched in the middle and superficial layers, whereas driven neurons were more common in deeper cortical layers (Fig 2D; layers were defined by current source density analysis, see Fig S6). Previous anatomical tracing studies have shown that neurons mediating feedforward projections tend to originate in superficial layers of the source area (*8*, *23*), and the fraction of feedforward projecting neurons in superficial layers decreases along the hierarchy (*24*, *25*). Neurons in the driver module followed this same pattern suggesting this module is more likely to be involved in feedforward processing (*25, 26*).

To explore signal transmission *between* modules, we examined the subnetworks defined by the connections from driver to driven neurons (Fig. 3A top) and from driven to driver neurons (Fig. 3A bottom). These subnetworks showed clear separation across the cortical depth as is evident by plotting the overlay of the two subnetworks for each source area (Fig. S7),

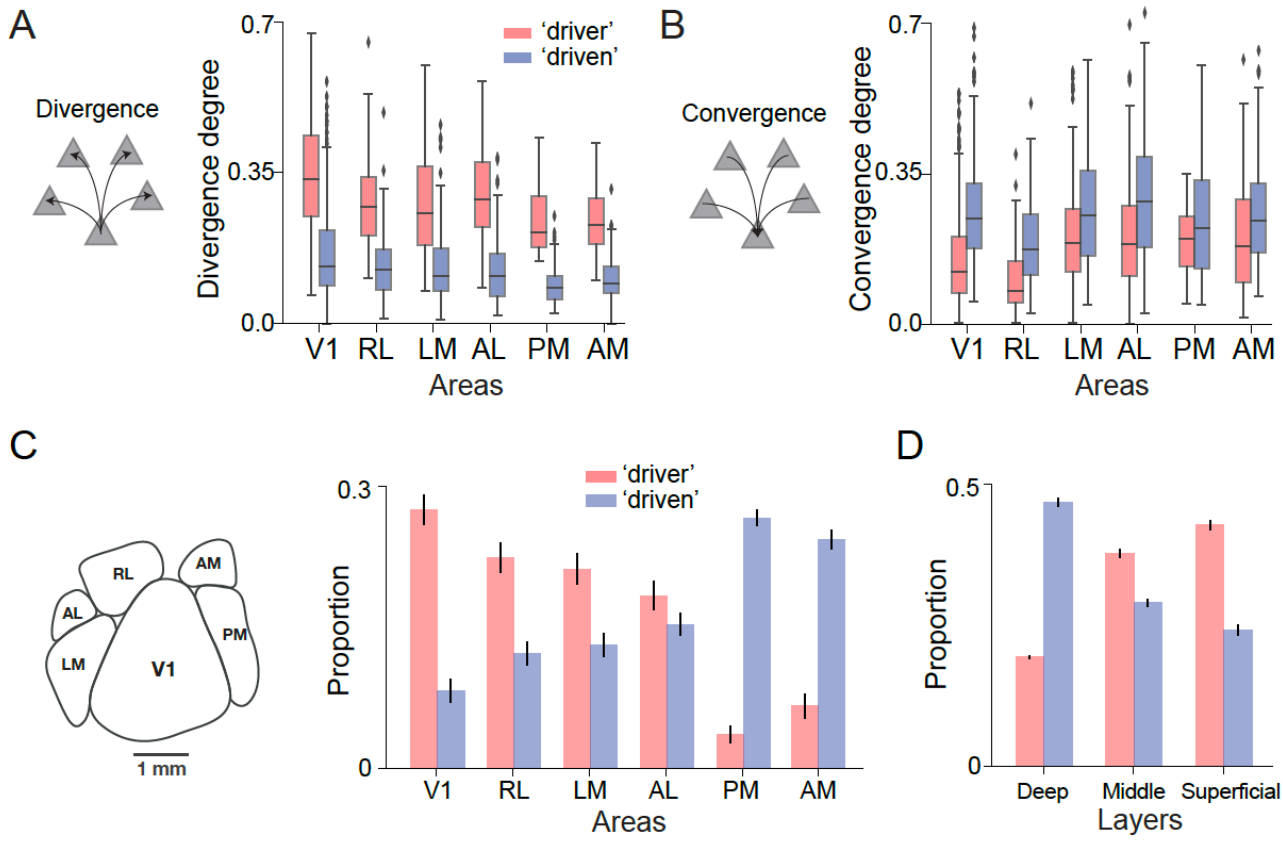


Figure 2. Network convergence/divergence and spatial distribution of neuronal modules. A) Boxplot of divergence degree distribution across areas for driver and driven modules. Divergence degree of driver module decreased along the visual hierarchy (Spearman's r = -0.83, p = 0.04). B) Convergence degree distribution for driver and driven modules. Convergence degree of driven module was not correlated with hierarchy level (Spearman's r = -0.03, p = 0.96). C) Left: Layout of visual cortical areas. Right: Proportion of neurons in driver and driven modules across the visual cortical hierarchy. D) Cortical layer bias of the driver and driven neural modules for all mice. Error bars in C) and D) represent bootstrapped standard error.

consistent with the laminar dependency in Fig 2D. Moreover, the subnetwork connections between modules were largely unidirectional; that is, the driver neurons from each source area (including top areas of the visual hierarchy, e.g. PM and AM) made output connections to driven neurons in the network, and the driven neurons in each source area received input connections from driver neurons (Fig. 3B). The asymmetry of these connections suggests a directional signal flow from the driver to driven module. To quantify this, we defined a metric called the in-out index, which describes the relative fraction of input versus output connections from a source area for a given subnetwork. An in-out index of 1 indicates all connections are inputs to the source area, and an in-out index of -1 indicates all connections are outputs (see Methods). The in-out index was close to -1 for all areas of the driver-to-driven subnetwork (Fig. 3E, top), indicating virtually all connections were from driver neurons in the source area to driven neurons in other areas. In contrast, the in-out index was close to 1 in each area for the driven-to-driver subnetwork (Fig. 3E, bottom).

The directional communication from driver to driven modules suggests these two modules might be sequentially activated during visual stimulation. To test this, we quantified the stimulus response latency for each module and found that spiking activity in the driver module preceded the driven module (time to peak = 60.5 ± 0.3 ms versus 80.0 ± 0.6 ms; Rank sum test statistics = -29, p = 7.1e-186) (Fig. 3F). We note that our jitter-corrected CCG method removes stimulus-locked correlations from the functional correlations, and therefore, the directionality of the functional connectivity between the modules is not a mere reflection of the average response latency. Moreover, we performed several simulations demonstrating that a temporal offset between modules in the stimulus-triggered average response does not by necessity entail the existence of brief timescale correlations between neurons in those modules, and vice versa (Fig. S8).

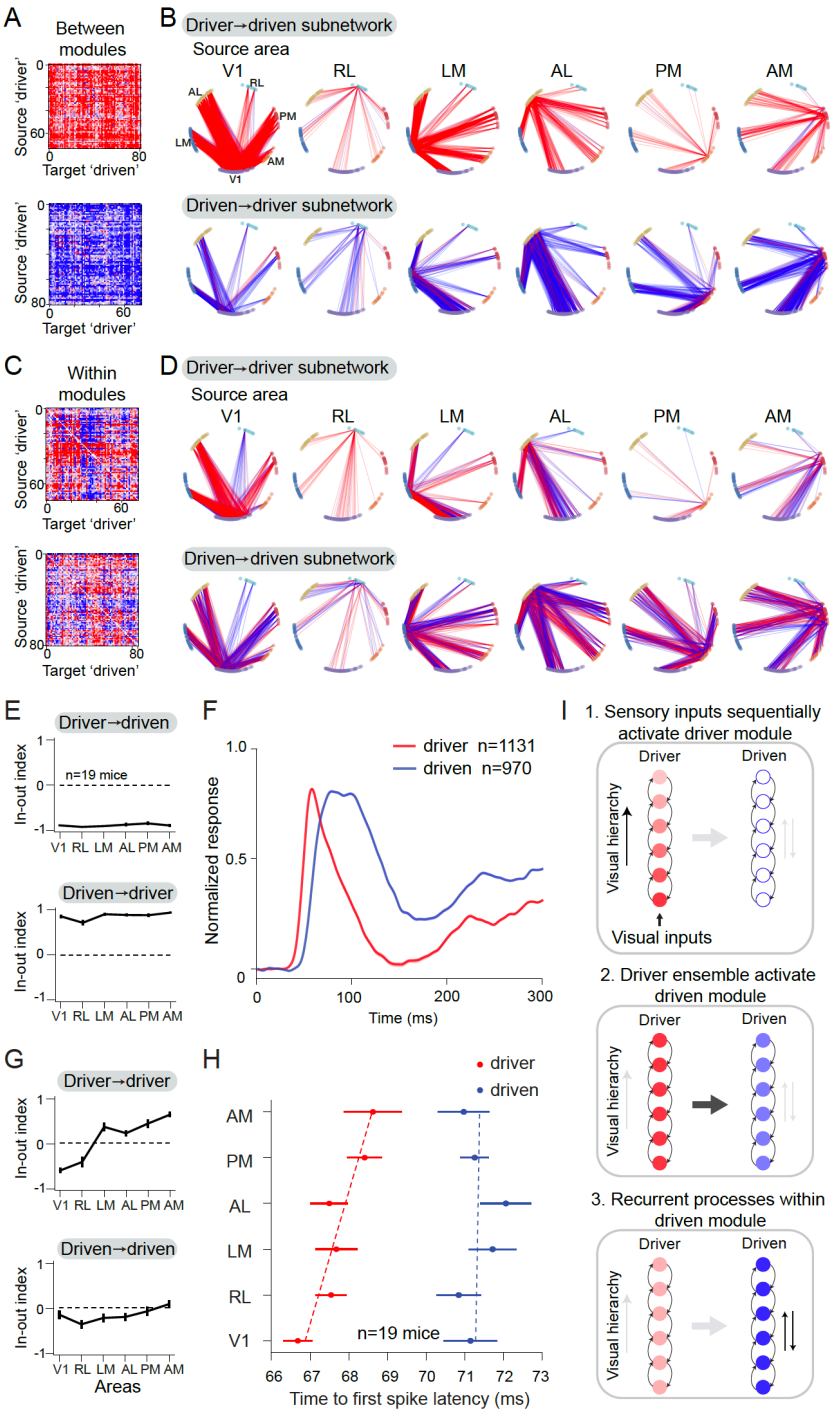


Figure 3. Signal flow in distinct subnetworks mediating feedforward and recurrent processes. A) Directed connectivity matrix for between-module subnetworks (*top*: driver (n=74) → driven (n=81); *bottom*: driven → driver) of one example mouse. B) Graph representation of strong connections (weight value > 10^{-6}) between source area (labeled at the top) and all other areas for between module subnetwork in A). Each node in the graph represents a recording site in cortex. Identity of visual areas is indicated by node color and arranged according to their anatomical location. Nodes in each area are arranged clockwise from superficial to deep layers. Line color indicates the directionality of connection (red: outputs; blue: inputs). C) Directed connectivity matrix for within-module subnetworks (*top*: driver → driver; *bottom*: driven → driven). D) Graph representation of within module subnetworks. E) In-out index averaged across mice (n=19 mice) for between-module subnetworks. F) Population average of normalized PSTH for two modules. G) In-out index for within-module subnetworks. H) Response latency (time-to-first spike) for different modules across areas (n = 19 mice). I) Diagram of the three temporal stages of information flow revealed by data. Intensity of color indicates activation level (darker is stronger). Error bar in this figure represents s.e.m.

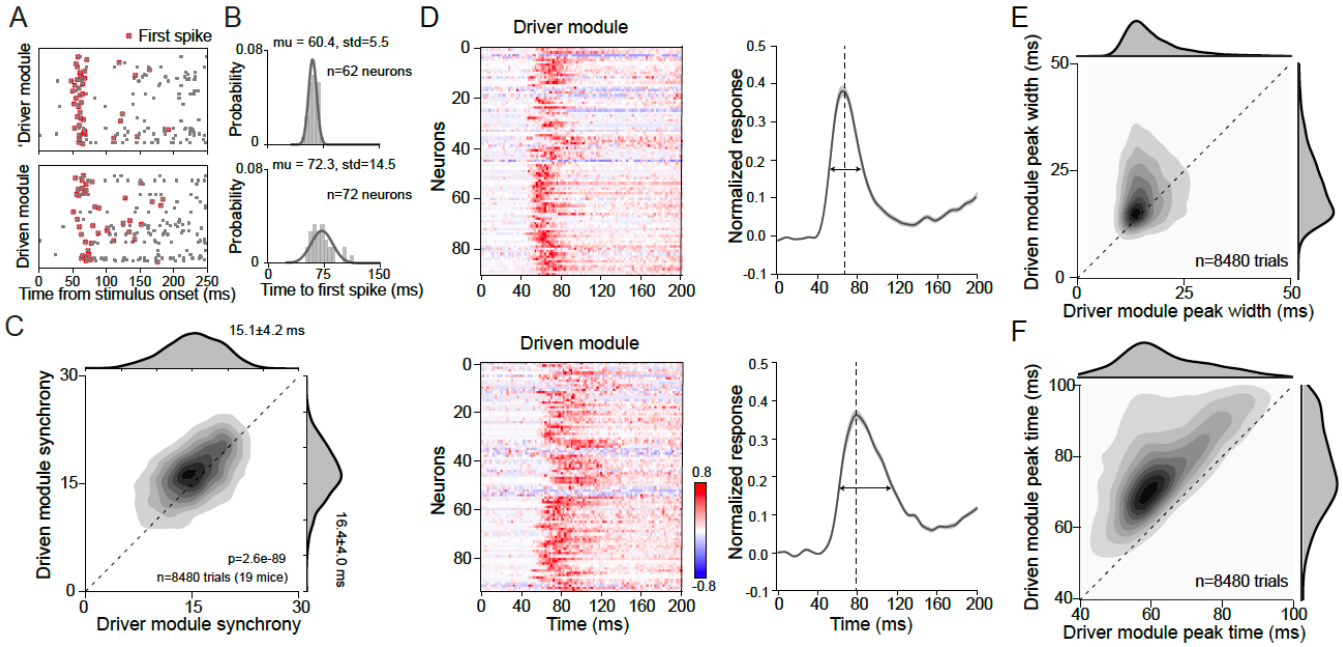


Figure 4. Population temporal structure in different neuronal modules. A) Raster plots for the driver module and driven module of a single trial from an example mouse. Each row is the spike times of one neuron. Red cross highlights the first stimulus-evoked spike. B) First spike latency distribution of each module from the trial shown in A). The spread quantifies module onset-latency synchronization in response to stimulus onset. C) Joint plot of module onset-latency spread comparing driver and driven module across trials (n = 8480 trials from 19 mice). D) Left: normalized PSTHs of individual neurons for driver (n = 91 neurons) and driven (n = 94 neurons) modules from an example mouse. Right: normalized population response. Driver module showed a first response peak at 66 ms with spread of 15.24 ms. Driven module showed a first peak at 78 ms with spread of 24.01ms (n = 75 trials). E) Joint scatter plot of population response first peak spread across trials. F) Joint scatter plot of population response first peak time across trials.

Whereas connectivity between modules was largely unidirectional, the functional connections *within* each module contained both input and output connections (Fig. 3C,D). To explore the within-module input-output structure, we computed the in-out index for each source area of the within-module subnetworks (Fig. 3G). Interestingly, within the driver module the in-out index systematically increased across the hierarchy: V1 had a negative in-out index and mostly made output connections to driver neurons in other areas; in contrast, driver neurons in AM received more inputs compared to outputs indicated by a positive in-out index (Fig. 3G, top; Spearman's correlation with hierarchy is -0.94, p = 0.0048). Connections within the driven module were more balanced: the in-out index was close to 0 for each area and did not significantly correlate with the hierarchy (Spearman's correlation, p = 0.16; Fig. 3G bottom). These within-module patterns of connectivity suggest that neurons in the driver module primarily relay feedforward signals across areas within the module itself, whereas the connections within driven module are positioned to mediate recurrent inter-areal interactions. Consistent with this, in the driver module, visually evoked response latencies systematically increased with location in the anatomical hierarchy (Fig. 3H ; correlation with mouse anatomical hierarchy score from *(9)*: Spearman's r = 0.83, p = 0.04). In contrast, visual latencies in the driven module were delayed relative to the driver module, and did not show an organized progression across the hierarchy (Spearman's r = 0.14, p = 0.79). Together, these results suggest a working model in which these separate neuronal modules serve as subsequent stages of signal propagation during visual processing: one module transmits feedforward signals about external stimuli along the hierarchy; the other integrates and processes recurrent signals given inputs from the driver module (Fig. 3I).

Modeling studies of feedforward networks have investigated how spiking signals are propagated across modules of sequential processing (*7*, *27*, *28*). Depending on network connections and synaptic weights, successive stages can propagate synchronous activity (e.g. a synfire chain) or asynchronous fluctuations in firing rate (rate-coded signals). *In vitro* experiments with cultured networks reported decreased within-module synchrony of spiking activity as signals were relayed across sequential

processing modules (*29*). However, evidence from networks in vivo in an awake animal is rare (*30*). The distributed functional modules we describe here could represent sequential stages of signal processing beyond specific anatomically defined areas. In this context, we compared onset-latency synchrony of the first stimulus-evoked spike on a trial-by-trial basis for the driver and driven modules (Fig. 4A-C). Neurons in the driver module were more tightly synchronized compared to those in the driven module (Fig. 4C driver 15.1 $\pm$ 4.2 ms and driven 16.4 $\pm$ 4.0 ms, n = 8480 trials, Student's t-test statistics T = -20.2, p = 2.6e-89). In addition, the spread of the first peak (pulse packet (*7*)) of the population response within driver module for each trial (Fig. 4D-F) was also more compact (Fig. 4E Student's t-test statistics T = -31.4, p = 3.3e-211, n = 8480 trials across 19 mice), with significantly earlier peak response (Fig. 4F T = -17.4, p = 1.4e-67). These results show that the transmission between the driver and driven module is associated with increased temporal spread of within-module spiking and decreased population onset synchronization. This is consistent with the concept of increasing recurrent interactions deeper into the processing chain (*31*, *32*).

Our results provide a multi-area, cellular scale description of functional connectivity in the mouse visual cortex and suggest a novel perspective of module-based signal transmission within this hierarchical recurrent neural network. Community detection methods have been used in human brain imaging studies to identify functional modules at the millimeter scale (*15*, *33*) with distributed organization (*34*). At the cellular-scale, the network structure we describe has a similar organizing principle: functional modules are spatially distributed across multiple cortical regions. Contrary to a simple feedforward model in which each cortical hierarchical level sequentially transmits signals up the chain of areas, we provide evidence for distributed functional modules that each included neurons from multiple hierarchical levels. These multi-area modules could correspond to core processing stages in the cortical network, and separately mediate feedforward and recurrent processes. Conceptually, our observations are in agreement with the counterstream theory (*25*, *35*), which suggests compartmentalization of FF and FB pathways, possibly by layer in each area. Whether similar functional modules exist in other cortical regions and species including primates requires further investigation. Unlike anatomical connectivity, functional connectivity is a dynamic entity that can be restructured by external stimuli, behavioral state, and top-down input according to computational demands (*15*). Future work can use module-specific perturbations to relate functional module dynamics to specific behavioral and cognitive operations.

Acknowledgments

We thank the Allen Institute founder, Paul G. Allen, for his vision, encouragement and support. We thank the Transgenic Colony Management for mouse breeding and Laboratory Animal Services for mouse import and wellness care. We thank the Neurosurgery and Behavior Team for surgical procedures and habituation. We thank Shiella Caldejon for running intrinsic signal imaging experiments, and Rusty Nicovich and Kiet Ngo for collecting optical projection tomography data. We thank the following for helpful discussions: Yazan Billeh, Uygar Sumbul, and Daniel Denman. We thank the following for helpful feedback on manuscript: Daniel Denman, Hannah Choi, Marina Garrett, Gabe Ocker, Adam Kohn and Christof Koch.

Author contributions

Conceptualization: X.J., J.H.S. and S.R.O.
Investigation, validation and methodology: J.H.S., X.J., S.D., G.H and T.R.
Formal analyses: X.J.
Visualization: X.J., S.R.O.
Original draft written by X.J. and S.R.O. with input and editing from J.H.S.
All co-authors reviewed the manuscript.

Competing interests: The authors declare no competing interests.

References

1. D. H. Perkel, T. Bullock, Neural Coding: A Report based on an NRP Work Session. **6** (1968).

2. A. J. Parker, W. T. Newsome, SENSE AND THE SINGLE NEURON: Probing the Physiology of Perception. *Annu. Rev. Neurosci.* **21**, 227–277 (1998).

3. D. H. Hubel, T. N. Wiesel, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J Physiol.* **160**, 106–154 (1962).

4. V. B. Mountcastle, G. F. Poggio, G. Werner, The relation of thalamic cell response to peripheral stimuli varied over an intensive continuum. *J. Neurophysiol.* **26**, 807–834 (1963).

5. J. P. Cunningham, B. M. Yu, Dimensionality reduction for large-scale neural recordings. *Nat. Neurosci.* **17**, 1500–1509 (2014).

6. B. B. Averbeck, P. E. Latham, A. Pouget, Neural correlations, population coding and computation. *Nat. Rev. Neurosci.* **7** (2006), pp. 358–366.

7. A. Kumar, S. Rotter, A. Aertsen, Spiking activity propagation in neuronal networks: Reconciling different perspectives on neural coding. *Nat. Rev. Neurosci.* **11**, 615–627 (2010).

8. D. J. Felleman, D. C. Van Essen, Distributed hierarchical processing in the primate cerebral cortex. *Cereb. cortex.* **1**, 1–47 (1991).

9. J. A. Harris, S. Mihalas, K. E. Hirokawa, J. D. Whitesell, J. Knox, A. Bernard, P. Bohn, S. Caldejon, L. Casal, A. Cho, D. Feng, N. Gaudreault, C. Gerfen, N. Graddis, P. A. Groblewski, A. Henry, A. Ho, R. Howard, L. Kuan, J. Lecoq, J. Luviano, S. McConoghy, M. Mortrud, M. Naeemi, L. Ng, S. W. Oh, B. Ouellette, S. Sorensen, W. Wakeman, Q. Wang, A. Williford, J. Phillips, A. Jones, C. Koch, H. Zeng, Hierarchical organization of cortical and thalamic connectivity. *Nature* (2019), doi:10.1101/292961.

10. R. Gămănuţ, H. Kennedy, Z. Toroczkai, M. Ercsey-Ravasz, D. C. Van Essen, K. Knoblauch, A. Burkhalter, The Mouse Cortical Connectome, Characterized by an Ultra-Dense Cortical Graph, Maintains Specificity by Distinct Connectivity Profiles. *Neuron.* **97**, 698-715.e10 (2018).

11. Q. Wang, A. Burkhalter, Area Map of Mouse Visual Cortex. *J. Comp. Neurol.* **502**, 339–357 (2007).

12. Y. Han, J. M. Kebschull, R. A. A. Campbell, D. Cowan, F. Imhof, A. M. Zador, T. D. Mrsic-Flogel, The logic of single-cell projections from visual cortex. *Nature.* **556**, 51–56 (2018).

13. G. Buzsáki, Large-scale recording of neuronal ensembles. *Nat. Neurosci.* **7**, 446–451 (2004).

14. J. H. Siegle, X. Jia, S. Durand, S. Gale, C. Bennett, G. Heller, T. K. Ramirez, H. Choi, J. A. Luviano, R. Ahmed, A. Arkhipov, A. Bernard, Y. N. Billeh, M. A. Buice, N. Cain, S. Caldejon, L. Casal, A. Cho, M. Chvilicek, T. C. Cox, K. Dai, D. J. Denman, S. E. J de Vries, R. Dietzman, L. Esposito, C. Farrell, D. Feng, J. Galbraith, M. Garrett, E. C. Gelfand, N. Hancock, J. A. Harris, R. Howard, B. Hu, R. Iyer, E. Jessett, K. Johnson, I. Kato, J. Kiggins, S. Lambert, J. Lecoq, P. Ledochowitsch, J. Hoon Lee, A. Leon, E. Liang, F. Long, K. Mace, J. Melchior, D. Millman, C. Nayan, L. Ng, K. Ngo, T. Nguyen, P. R. Nicovich, K. North, G. Koch Ocker, D. Ollerenshaw, M. Oliver, M. Pachitariu, M. Reding, D. Reid, M. Robertson, K. Ronellenfitch, S. Seid, C. Slaughterbeck, M. Stoecklin, D. Sullivan, B. Sutton, J. Swapp, K. Turner, W. Wakeman, J. D. Whitesell, D. Williams, R. Young, H. Zeng, S. Naylor, J. W. Phillips, R. Clay Reid, S. R. Olsen, C. Koch, A survey of spiking activity reveals a functional hierarchy of mouse corticothalamic visual areas. *bioRxiv*, 1–53 (2019).

15. O. Sporns, R. F. Betzel, Modular Brain Networks. *Annu. Rev. Psychol.* **67**, 613–640 (2016).

16. M. E. Garrett, I. Nauhaus, J. H. Marshel, E. M. Callaway, Topography and Areal Organization of Mouse Visual Cortex. *J. Neurosci.* **34**, 12587–12600 (2014).

17. R. D. D'Souza, Q. Wang, W. Ji, A. M. Meier, H. Kennedy, S. Cell, U. Lyon, U. Claude, B. Lyon, Canonical and noncanonical features of the mouse visual cortical hierarchy. *bioRxiv* (2020).

18. M. A. Smith, A. Kohn, Spatial and Temporal Scales of Neuronal Correlation in Primary Visual Cortex. *J. Neurosci.*, 12591–12603 (2008).

19. X. Jia, S. Tanabe, A. Kohn, γ and the coordination of spiking activity in early visual cortex. *Neuron.* **77**, 762–74 (2013).

20. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, others, Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).

21. A. Fornito, A. Zalesky, E. T. Bullmore, *Fundamentals of Brain Network Analysis* (Elsevier Inc., 2016).

22. G. Tononi, G. M. Edelman, O. Sporns, Complexity and coherency: Integrating information in the brain. *Trends Cogn. Sci.* **2** (1998), pp. 474–484.

23. N. T. Markov, M. M. Ercsey-Ravasz, A. R. Ribeiro Gomes, C. Lamy, L. Magrou, J. Vezoli, P. Misery, A. Falchier, R. Quilodran, M. A. Gariel, J. Sallet, R. Gamanut, C. Huissoud, S. Clavagnier, P. Giroud, D. Sappey-Marinier, P. Barone, C. Dehay, Z. Toroczkai, K. Knoblauch, D. C. Van Essen, H. Kennedy, A weighted and directed interareal connectivity matrix for macaque cerebral cortex. *Cereb. Cortex.* **24**, 17–36 (2014).

24. P. Barone, A. Batardiere, K. Knoblauch, H. Kennedy, Laminar distribution of neurons in extrastriate areas projecting to visual areas V1 and V4 correlates with the hierarchical rank and indicates the operation of a distance rule. *J. Neurosci.* **20**, 3263–81 (2000).

25. N. T. Markov, J. Vezoli, P. Chameau, A. Falchier, R. Quilodran, C. Huissoud, C. Lamy, P. Misery, P. Giroud, S. Ullman, P. Barone, C. Dehay, K. Knoblauch, H. Kennedy, Anatomy of hierarchy: Feedforward and feedback pathways in macaque visual cortex. *J. Comp. Neurol.* **522**, 225–259 (2014).

26. J. H. Marshel, Y. S. Kim, T. A. Machado, S. Quirin, B. Benson, J. Kadmon, C. Raja, A. Chibukhchyan, C. Ramakrishnan, M. Inoue, J. C. Shane, D. J. McKnight, S. Yoshizawa, H. E. Kato, S. Ganguli, K. Deisseroth, Cortical layer-specific critical dynamics triggering perception. *Science (80-.).* **365** (2019), doi:10.1126/science.aaw5202.

27. A. D. Reyes, Synchrony-dependent propagation of firing rate in iteratively constructed networks in vitro. *Nat. Neurosci.* **6**, 593–599 (2003).

28. T. P. Vogels, L. F. Abbott, Signal propagation and logic gating in networks of integrate-and-fire neurons. *J. Neurosci.* **25**, 10786–10795 (2005).

29. J. Barral, X. J. Wang, A. D. Reyes, Propagation of temporal and rate signals in cultured multilayer networks. *Nat. Commun.* **10** (2019), doi:10.1038/s41467-019-11851-0.

30. A. Zandvakili, A. Kohn, Coordinated Neuronal Activity Enhances Corticocortical Communication. *Neuron.* **87**, 827–839 (2015).

31. T. Lu, L. Liang, X. Wang, Temporal and rate representations of time-varying signals in the auditory cortex of awake primates. *Nat. Neurosci.* **4**, 1131–1138 (2001).

32. R. L. T. Goris, J. A. Movshon, E. P. Simoncelli, Partitioning neuronal variability. *Nat. Neurosci.* **17**, 858–865 (2014).

33. R. F. Betzel, J. D. Medaglia, D. S. Bassett, Diversity of meso-

scale architecture in human and non-human connectomes. *Nat. Commun.* **9** (2018), doi:10.1038/s41467-017-02681-z.

34. J. D. Power, A. L. Cohen, S. M. Nelson, G. S. Wig, K. A. Barnes, J. A. Church, A. C. Vogel, T. O. Laumann, F. M. Miezin, B. L. Schlaggar, S. E. Petersen, Functional Network Organization of the Human Brain. *Neuron*. **72**, 665–678 (2011).

35. Shimon Ullman, Sequence Seeking and Counter Streams: A Computational Model for Bidirectional Information Flow in the Visual Cortex | Cerebral Cortex | Oxford Academic. *Cereb. Cortex*. **5**, 1–11 (1995).

36. J. J. Jun, N. A. Steinmetz, J. H. Siegle, D. J. Denman, M. Bauza, B. Barbarits, A. K. Lee, C. A. Anastassiou, A. Andrei, Ç. Aydin, M. Barbic, T. J. Blanche, V. Bonin, J. Couto, B. Dutta, S. L. Gratiy, D. A. Gutnisky, M. Häusser, B. Karsh, P. Ledochowitsch, C. M. Lopez, C. Mitelut, S. Musa, M. Okun, M. Pachitariu, J. Putzeys, P. D. Rich, C. Rossant, W. L. Sun, K. Svoboda, M. Carandini, K. D. Harris, C. Koch, J. O'Keefe, T. D. Harris, Fully integrated silicon probes for high-density recording of neural activity. *Nature*. **551**, 232–236 (2017).

37. J. H. Siegle, A. C. López, Y. A. Patel, K. Abramov, S. Ohayon, J. Voigts, Open Ephys: an open-source, plugin-based platform for multichannel electrophysiology. *J. Neural Eng.* **14**, 045003 (2017).

38. J. W. Peirce, PsychoPy-Psychophysics software in Python. *J. Neurosci. Methods* (2007), doi:10.1016/j.jneumeth.2006.11.017.

39. C. Stringer, M. Pachitariu, N. Steinmetz, C. B. Reddy, M. Carandini, K. D. Harris, Spontaneous behaviors drive multidimensional, brainwide activity. *Science (80-.).* **364** (2019), doi:10.1126/science.aav7893.

40. M. Pachitariu, N. A. Steinmetz, S. N. Kadir, M. Carandini, K. D. Harris, Fast and accurate spike sorting of high-channel count probes with KiloSort. *Adv. Neural Inf. Process. Syst.* **29**, 4448–4456 (2016).

41. X. Jia, S. Tanabe, A. Kohn, Gamma and the Coordination of Spiking Activity in Early Visual Cortex. *Neuron*. **77**, 762–774 (2013).

42. P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, A. Vijaykumar, A. Pietro Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C. N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D. A. Nicholson, D. R. Hagen, D. V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G. A. Price, G. L. Ingold, G. E. Allen, G. R. Lee, H. Audren, I. Probst, J. P. Dietrich, J. Silterra, J. T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J. L. Schönberger, J. V. de Miranda Cardoso, J. Reimer, J. Harrington, J. L. C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N. J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P. A. Brodtkorb, P. Lee, R. T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T. J. Pingel, T. P. Robitaille, T. Spura, T. R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y. O. Halchenko, Y. Vázquez-Baeza, SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods*. **17**, 261–272 (2020).

43. S. Seabold, J. Perktold, "Statsmodels: Econometric and Statistical Modeling with Python" (2010).

44. A. Hagberg, D. Schult, P. Swart, Exploring Network Structure, Dynamics, and Function using NetworkX. *Proc. 7th Python Sci. Conf.* (2008), pp. 11–15.

45. M. A. Smith, A. Kohn, Spatial and Temporal Scales of Neuronal Correlation in Primary Visual Cortex. *J. Neurosci.* **28**, 12591–12603 (2008).

46. M. T. Harrison, S. Geman, A rate and history-preserving resampling algorithm for neural spike trains. *Neural Comput.* **21** (2009), pp. 1244–1258.

47. D. A. Spielman, (Institute of Electrical and Electronics Engineers (IEEE), 2008), pp. 29–38.

48. R. Tibshirani, G. Walther, T. Hastie, Estimating the number of clusters in a data set via the gap statistic. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **63**, 411–423 (2001).

49. D. T. Pham, S. S. Dimov, C. D. Nguyen, Selection of K in K-means clustering. *Proc. Inst. Mech. Eng. Part C J. Mech. Eng. Sci.* (2005), doi:10.1243/095440605X8298.

50. S. Monti, P. Tamayo, J. Mesirov, T. Golub, P. Sebastiani, I. S. Kohane, M. F. Ramoni, "Consensus Clustering: A Resampling-Based Method for Class Discovery and Visualization of Gene Expression Microarray Data" (2003).

51. D. N. Hill, S. B. Mehta, D. Kleinfeld, Quality metrics to accompany spike sorting of extracellular signals. *J. Neurosci.* **31**, 8699–8705 (2011).

52. J. E. Chung, J. F. Magland, A. H. Barnett, V. M. Tolosa, A. C. Tooker, K. Y. Lee, K. G. Shah, S. H. Felix, L. M. Frank, L. F. Greengard, A Fully Automated Approach to Spike Sorting. *Neuron*. **95**, 1381-1394.e6 (2017).

53. C. R. Stoelzel, Y. Bereshpolova, H. A. Swadlow, Stability of thalamocortical synaptic transmission across awake brain states. *J. Neurosci.* **29**, 6851–9 (2009).

54. L. McInnes, J. Healy, J. Melville, UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction (2018).

Supplementary Materials

Materials and Methods

Mice

Mice were maintained in the Allen Institute animal facility and used in accordance with protocols approved by the Allen Institute's Institutional Animal Care and Use Committee. Four mouse genotypes were used: wild-type C57BL/6J (Jackson Laboratories) (*n* = 11) or Pvalb-IRES-Cre (*n* = 1), Vip-IRES-Cre (*n* = 2), and Sst-IRES-Cre (*n* = 5) mice bred in-house and crossed with an Ai32 channelrhodopsin reporter line. Following surgery, all mice were single-housed and maintained on a reverse 12-hour light cycle. All experiments were performed during the dark cycle.

Data collection

Experimental data collection followed the procedures described in Siegle, Jia et al., 2019 (*14*). A summary of these methods is provided below. 13 out of 19 of datasets in this study were previously released on the Allen Institute website via the AllenSDK (https://github.com/AllenInstitute/AllenSDK).

Surgical methods

All surgical methods used here are the same as (*14*). Briefly, to enable co-registration across the surgical, intrinsic signal imaging, and electrophysiology rigs, each animal was implanted with a titanium headframe that provides access to the brain via a cranial window and permits head fixation in a reproducible configuration. To implant the headframe, mice were initially anesthetized with 5% isoflurane (1-3 min) and placed in a stereotaxic frame (Model# 1900, Kopf). Isoflurane levels were maintained at 1.5-2.5% for surgery and body temperature was maintained at 37.5°C. Carprofen was administered for pain management (5-10 mg/kg, S.C.). Atropine was administered to suppress bronchial secretions and regulate heart rhythm (0.02-0.05 mg/kg, S.C.). The headframe was placed on the skull and fixed in place with White C&B Metabond (Parkell). Once the Metabond was dry, the mouse was placed in a custom clamp to position the skull at a rotated angle of 20º, to facilitate the creation of the craniotomy over visual cortex. A circular piece of skull 5 mm in diameter was removed, and a durotomy was performed. The brain was covered by a 5 mm diameter circular glass coverslip, with a 1 mm lip extending over the intact skull. The bottom of the coverslip was coated with a layer of silicone to reduce adhesion to the brain surface. At the end of the procedure, but prior to recovery from anesthesia, the mouse was transferred to a photo-documentation station to capture a spatially registered image of the cranial window.

On the day of recording (at least four weeks after the initial surgery), the cranial coverslip was removed and replaced with an insertion window containing holes aligned to six cortical visual areas. First, the mouse was anesthetized with isoflurane (3%–5% induction and 1.5% maintenance, 100% O_2) and eyes were protected with ocular lubricant (I Drop, VetPLUS). Body temperature was maintained at 37.5°C (TC-1000 temperature controller, CWE, Incorporated). The cranial window was gently removed to expose the brain. An insertion window with holes for probe penetration based on each mouse's individual visual area map was then placed in the headframe well and sealed with Metabond. An agarose mixture was injected underneath the window and allowed to solidify. The mixture consisted of 0.4 g high EEO Agarose (Sigma-Aldrich), 0.42 g Certified Low-Melt Agarose (Bio Rad), and 20.5 mL ACSF (135.0 mM NaCl, 5.4 mM KCl, 1.0 mM MgCl2, 1.8 mM CaCl2, 5.0 mM HEPES). This mixture was optimized to be firm enough to stabilize the brain with minimal probe drift, but pliable enough to allow the probes to pass through without bending. A layer of silicone oil (30,000 cSt, Aldrich) was added over the holes in the insertion window to prevent the agarose from drying. A 3D-printed plastic cap was screwed into the headframe well to keep out cage debris. At the end of this procedure, mice were returned to their home cages for 1-2 hours prior to the Neuropixels recording session.

Intrinsic Signal Imaging
Intrinsic signal imaging was performed approximately 15 days after the initial surgery and 25 days before the experiment. Intrinsic signal imaging was used to obtain retinotopic maps representing the spatial relationship of the visual field (or, in this case, coordinate position on the stimulus monitor) to locations within each cortical area (Fig. 1A). The maps made it possible to delineate functionally defined visual area boundaries in order to target Neuropixels probes to retinotopically defined locations in primary and higher order visual cortical areas (*16*).

Habituation
Mice underwent two weeks of habituation in sound-attenuated training boxes containing a headframe holder, running wheel, and stimulus monitor. Each mouse was trained by the same operator throughout the 2-week period. During the first week, the operator gently handled the mice, introduced them to the running wheel, and head-fixed them with progressively longer durations each day. During the second week, mice run freely on the wheel and were exposed to visual stimuli for 10 to 50 min per day. The following week, mice underwent habituation sessions of 75 minutes and 100 minutes on the recording rig, in which they viewed a truncated version of the same stimulus shown during the experiment.

Electrophysiology Experiments
All neural recordings were carried out with Neuropixels probes (*36*). Each probe contains 960 recording sites, a subset of these (374 for “Neuropixels 3a” or 383 for “Neuropixels 1.0”) can be configured for recording at any given time. The electrodes closest to the tip were always used, providing a maximum of 3.84 mm of tissue coverage. The sites are oriented in a checkerboard pattern on a 70 µm wide x 10 mm long shank. The signals from each recording site are split in hardware into a spike band (30 kHz sampling rate, 500 Hz highpass filter) and an LFP band (2.5 kHz sampling rate, 1000 Hz lowpass filter).

The experimental rig was designed to allow six Neuropixels probes to penetrate the brain approximately perpendicular to the surface of visual cortex (*14*). Each probe was mounted on a 3-axis micromanipulator (New Scale Technologies, Victor, NY), which were in turn mounted on a solid aluminum plate, known as the probe cartridge. The mouse was placed on the running wheel and fixed to the headframe clamp. The tip of each probe was aligned to target the desired retinotopic region in each area. Brightfield photo-documentation images were taken with the probes fully retracted, after the probes reached the brain surface, and again after the probes were fully inserted. An IR dichroic mirror was placed in front of the right eye to allow an eyetracking camera to operate without interference from the visual stimulus. A black curtain was then lowered over the front of the rig, placing the mice in complete darkness except for the visual stimulus monitor.

Neuropixels data was acquired at 30 kHz (spike band) and 2.5 kHz (LFP band) using the Open Ephys GUI (*37*). Gain settings of 500x and 250x were used for the spike band and LFP band, respectively. Each probe was either connected to a dedicated FPGA streaming data over Ethernet (Neuropixels 3a) or a PXIe card inside a National Instruments chassis (Neuropixels 1.0). Raw neural data was streamed to a compressed format for archiving which was extracted prior to analysis.

Cortical Area Targeting
To confirm the identity of the cortical visual areas, images of the probes taken during the experiment were compared to images of the brain surface vasculature taken during the ISI session (see above). Vasculature patterns were used to overlay the visual area map on an image of the brain surface with the probes inserted (Fig 1A). To maximize measurable functional connectivity across areas, we targeted the center of gaze in all areas (except for RL, which targeted the center of mass because of geometry) with overlapping receptive fields (RF) guided by a retinotopic map. Targeting was validated by mapping receptive fields of all sorted units with small Gabor patches presented at different locations on the screen

(see below). All analysis was restricted to neurons with well-defined receptive fields within the screen boundaries.

Visual Stimulus

Visual stimuli were generated using custom scripts based on PsychoPy (*38*) and were displayed using an ASUS PA248Q LCD monitor, with 1920 x 1200 pixels (21.93 in wide, 60 Hz refresh rate). Stimuli were presented monocularly, and the monitor was positioned 15 cm from the mouse's right eye and spanned 120° x 95° of visual space prior to stimulus warping. Each monitor was gamma corrected and had a mean luminance of 50 cd/m^2. To account for the close viewing angle of the mouse, a spherical warping was applied to all stimuli to ensure that the apparent size, speed, and spatial frequency were constant across the monitor as seen from the mouse's perspective.

Visual stimuli for receptive fields (RFs): Receptive field location was mapped with small Gabor patches. The receptive field mapping stimulus consisting of 2 Hz, 0.04 cycles per degree drifting gratings (3 directions: 0º, 45º, 90º) with a 20º circular mask. These Gabor patches randomly appeared at one of 81 locations on the screen (9 x 9 grid with 10º spacing) for 250 ms at a time, with no blank interval.

Visual stimuli for current source density (CSD): Current source density for layer estimation used the full-field flash stimuli (a series of dark or light full field image with luminance = 100 cd/m^2). lasting 250 ms each and separated by a 1.75 second inter-trial interval.

Visual stimuli for functional connectivity: Functional connectivity during the stimulus-driven condition was measured using drifting grating stimuli, which were presented at 4 directions (0º, 45º, 90º, 135º), with temporal frequency equal to 2 cycle/sec and contrast equal to 0.8. In each trial, the grating is presented for 2 sec followed by 1 sec gray screen. Each condition was presented for 75-100 trials.

Spike Sorting

Prior to spike sorting, the spike-band data passed through 4 steps: DC offset removal, median subtraction, filtering, and whitening. First, the median value of each channel was subtracted to center the signals around zero. Next, the median across channels was subtracted to remove common-mode noise. The median-subtracted data file is the input to the Kilosort2 Matlab package (https://github.com/mouseland/kilosort2), which applies a 150 Hz high-pass filter, followed by whitening in blocks of 32 channels. The filtered, whitened data is saved to a separate file for the spike sorting step.

Kilosort2 was used to identify spike times and assign spikes to individual units (*39*). Kilosort2 attempts to model the complete dataset as a sum of spike "templates." The shape and locations of each template is iteratively refined until the data can be accurately reconstructed from a set of *N* templates at *M* spike times, with each individual template scaled by an amplitude, *a*. A critical feature of Kilosort2 is that it allows templates to change their shape over time, to account for the motion of neurons relative to the probe over the course of the experiment. Stabilizing the brain using an agarose-filled plastic window has virtually eliminated probe motion associated with animal running, but slow drift of the probe over ~3-hour experiments is still observed. Kilosort2 is able to accurately track units as they move along the probe axis, eliminating the need for the manual merging step that was required with the original version of Kilosort (*40*). The spike-sorting step runs in approximately real time (~3 hours per session) using a dual-processor Intel 4-core, 2.6 GHz workstation with an NVIDIA GTX 1070 GPU. We used the default parameters in Kilosort2, with an initial threshold of 12, and a final-pass threshold of 8.

The Kilosort2 algorithm will occasionally fit a template to the residual left behind after another template has been subtracted from the original data, resulting in double-counted spikes. This can create the appearance of an artificially high number of ISI violations for one unit or artificially high zero-time-lag

synchrony between nearby units. To eliminate the possibility that this artificial synchrony will contaminate data analysis, the outputs of Kilosort2 are post-processed to remove spikes with peak times within 5 samples (0.16 ms) and peak waveforms within 5 channels (~50 microns).

Kilosort2 generates templates of a fixed length (2 ms) that matches the time course of an extracellularly detected spike waveform. However, there are no constraints on template shape, which means that the algorithm often fits templates to voltage fluctuations with characteristics that could not physically result from the current flow associated with an action potential. The units associated with these templates are considered "noise," and are automatically filtered out based on 3 criteria: spread (single channel, or >25 channels), shape (no peak and trough, based on wavelet decomposition), or multiple spatial peaks (waveforms are non-localized along the probe axis).

Following the spike sorting step, data for each session was uploaded to the Allen Institute Laboratory Information Management System (LIMS). Each dataset was run through the same series of processing steps using a set of project-specific workflows (AllenSDK v1.0.2) in order to generate NeurodataWithoutBorders (NWB) files used for further analysis.

Analysis Methods

Dataset

In total, units from 19 mice were included in our functional connectivity analysis. Spike sorting, quality control, and preprocessing steps followed the same procedures as (*14*). 13 out of 19 of these datasets were previously released on the Allen Institute website via the AllenSDK (https://github.com/AllenInstitute/AllenSDK). On average, 632 $\pm$ 18 sorted cortical units were simultaneously recorded in each mouse. We set a firing rate threshold to select units for functional connectivity analysis. Firing rate (FR) was defined as the average number of spikes in a window from 50 ms to 500 ms after the onset of the drifting gratings stimulus (Fig. S1). Only units with mean FR > 2 spikes/second were used for pairwise cross-correlogram (CCG) calculation, which resulted in an average of 356 $\pm$ 7 units in each mouse (n = 6773 units in total). Because functional connectivity varies with receptive field position (*41*), we further constrained the dataset to include units with receptive field centers at least 10 degree away from the edge of the monitor (see Visual receptive fields section below; Fig. S1). After filtering by receptive field location, we ended up with 184 $\pm$ 8 per mouse used for the final clustering procedure (n = 3487 units in total). After applying clustering on the functional connectivity matrix constrained by both FR and RF location in each mouse, the total numbers of units belonging to each cluster were: n_cluster1 = 1386, n_cluster2 = 1131, n_cluster3 = 970.

Quantification and statistical analysis

All analyses were performed in Python. The main analysis packages used in this paper are Scipy (*42*), scikit-learn (*20*), statsmodels (*43*), and network (*44*). Error bars, unless otherwise specified, were computed as standard error of the mean. When comparing the difference between two independent variables, if their distribution is Gaussian like (normality test), we used Student's *t*-test; if their distribution is non-Gaussian, we used a rank sum test. When testing whether a distribution is significantly different from 0, we used a one-sample *t*-test. When comparing variables between modules across cortical areas, we used two-way analysis of variance (ANOVA) to assess both the main effect between modules and whether there is any interaction across areas. When comparing similarity to the previously established anatomical visual hierarchy in mouse (*9*), we calculated the correlation between our measured variable (e.g. first spike latency) and the previously calculated hierarchy score (V1: -0.50, RL: -0.12, LM: -0.13, AL: 0.00, PM: 0.11, AM: 0.29), using Spearman's correlation to estimate the rank order significance. Statistical details and *p*-values can be found in the Results section or figure legends.

Visual receptive fields

Receptive fields were mapped with Gabor patches (20 degree each; 3 different orientations (0, 45, 90), temporal frequency = 2 cyc/s, spatial frequency = 0.04 cyc/deg) shown randomly at 81 different locations (9 x 9 grid, 10º separation between pixel centers) with gray background on a 120° x 95° monitor (1920 x 1200 pixels, 21.93 inches wide, 60 Hz refresh rate). The receptive field map (RF) for one unit is defined as the mean 2D histogram of spike counts at each of 81 locations (Fig. S1A), each pixel covers a 10° x 10° square. The receptive field was then thresholded at 20% of maximum response (Fig. S1B) to remove potential noisy pixels. Then, a 2D Gaussian

$$f(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-\left[\frac{(x-\mu_x)^2}{2\sigma_x^2}+\frac{(y-\mu_y)^2}{2\sigma_y^2}\right]}$$

was fit to the thresholded visual receptive map to estimate the center of the receptive field location (Fig. S1C).

Peristimulus time histogram (PSTH)

To visualize the temporal dynamics of a neuronal population (Fig. 1, Fig. 3, and Fig. 4), the activity of each neuron was binned at 1 ms, averaged across trials (*n* = 75), smoothed with a Gaussian filter with standard deviation of 3 ms, baseline subtracted (baseline period from 0 to 0.03s relative to stimulus onset), and normalized by dividing the maximum of the response between 0 to 1.5 s after stimulus onset. The normalized PSTHs of individual neuron were averaged within a neuronal population; the error bars indicate standard error of the mean across neurons.

Functional connectivity

We analyzed functional interactions between pairs of simultaneously recorded neurons by calculating the spike train cross-correlogram (CCG) (*30*, *41*, *45*). For a pair of neurons with spike train x_1 and x_2, the CCG is defined as:

$$\mathrm{CCG}(\tau) = \frac{\frac{1}{\mathrm{M}}\sum_{\mathrm{i}=1}^{\mathrm{M}}\sum_{\mathrm{t}=1}^{\mathrm{N}} \mathrm{x}_1^{\mathrm{i}}(\mathrm{t})\mathrm{x}_2^{\mathrm{i}}(\mathrm{t}+\tau)}{\Theta(\tau)\sqrt{\lambda_1\lambda_2}}$$

where *M* is the number of trials, *N* is the number of bins in the trial, $\mathrm{x}_1^{\mathrm{i}}$ and $\mathrm{x}_2^{\mathrm{i}}$ are the spike trains of the two units on trial i, τ is the time lag relative to reference spikes, and λ_1 and λ_2 are the mean firing rates of the two units. The CCG is essentially a sliding dot product between two spike trains. $\theta(\tau)$ is the triangular function which corrects for the overlapping time bins caused by the sliding window. To correct for firing rate dependency, we normalized the CCG by the geometric mean spike rate. An individually normalized CCG is computed separately for each drifting grating orientation and averaged across orientations to obtain the CCG for each pair of units.

The jitter-corrected CCG was created by subtracting the expected value of CCGs produced from a resampled version of the original dataset with spike times randomly perturbed (jittered) within the jitter window (*45*, *46*). The correction term ($CCG_{jittered}$) is the true expected value which reflects the average over all possible resamples of the original dataset. $CCG_{jittered}$ is normalized by the geometric mean rate before subtracting from $CCG_{original}$. The analytical formula used to create a probability distribution of resampled spikes was provided in Harrison and Geman, 2009. This method disrupted the temporal correlation within the jitter window, while maintaining the number of spikes in each jitter window and the shape of the PSTH averaged across trials.

$$\mathrm{CCG}_{\mathrm{jitter_corrected}} = \mathrm{CCG}_{\mathrm{original}} - \mathrm{CCG}_{\mathrm{jittered}}$$

For our measurement, a 25 ms jitter window was chosen based on previous studies (*30*, *41*). This jitter-correction method removes both the stimulus-locked component of the response, as well as slow fluctuations larger than the jitter window. The remaining fast timescale correlation is more likely to be related to signal propagation between two neurons. Therefore, the jitter-corrected CCG reflects temporal correlations between a pair of neurons within the jitter-window (25ms).

We then calculated the directed connection weight by subtracting the sum of (-13 to 0) ms of the CCG from the sum of (0 to 13) ms of the jitter-corrected CCG (Fig. 1D). The 13ms window was defined as half of the 25 ms jitter window we used, and also because real functional delay between neurons in mouse occur on the timescale of milliseconds to tens of milliseconds (*14*). The resulting value indicates the strength and the sign indicates the directionality of the functional connection between a pair of neurons. Computing this for all pairs of neurons produced a directional, cellular-resolution connectivity matrix for each mouse (Fig. 1E).

Clustering

Non-randomness

We first tested whether there is modular structure (non-randomness) in the measured connectivity matrix by computing the graph spectrum (based on spectral graph theory (*47*). The eigenvalues of a graph are defined as the eigenvalues of its adjacency matrix (*21*). The set of eigenvalues of a graph forms a graph spectrum. The randomness of the matrix is quantified by comparing the graph spectrum of the original connectivity matrix with its shuffled connectivity matrix, where the x and y axis are shuffled independently, and a randomly generated connectivity matrix with the same size. We found that the graph spectrum of the original matrix showed significantly higher explained variance by the top eigenvalues than the shuffled matrix and the random matrix, suggesting that the measured connectivity matrix has non-random structure (Fig. S2).

Defining the number of clusters

The number of clusters was determined using several complementary methods (Fig. S4A):

1. The Elbow method estimates the percentage of variance explained for a given number of k. The number of cluster $\hat{k}$ is estimated at the point when the curve turns into a plateau. The following measure represents the sum of within-cluster distances (pairwise distances) between all points in a given cluster C_k containing n_k points:

   $$D_k = \sum_{x_i \in C_k} \sum_{x_j \in C_k} \| x_i - x_j \|^2 = 2n_k \sum_{x_j \in C_k} \| x_i - \mu_k \|^2$$

   Adding the normalized within-cluster sum-of-squares gives a measure of the compactness of our clustering, or the pooled within-cluster sum of squares around the cluster means:

   $$W_k = \sum_{k=1}^{K} \frac{1}{2n_k} D_k$$

   W_k increases monotonically with number of clusters k. The number of clusters is chosen at the point where the marginal gain drops (or the point slope change most dramatically), the 'elbow'.

2. Gap statistics (*48*) seeks to standardize the comparison of $\log W_k$ with a null reference distribution of the data, i.e. a distribution with no obvious clustering. The estimate for the optimal number of clusters K is the value for which $\log W_k$ falls the farthest below this reference curve. This information is contained in the following formula for the gap statistic:

   $$Gap_n(k) = E_n^* \left\{ \log W_k \right\} - \log W_k \ ,$$

Where E_n^* denotes the expectation under a sample of size *n* from the reference distribution. The estimate $\hat{k}$ will be the value maximizing $Gap_n(k)$ after we take the sampling distribution into account.

3. Clustering density estimates the data distribution density for a given k by calculating a density function $f(k)$ (*49*). The value of $f(k)$ is the ratio of the real distortion to the estimated distortion. When the data are uniformly distributed, the value of $f(k)$ is 1. When there are areas of concentration in the data distribution, the value of $f(k)$ decreases. Therefore, the number of $\hat{k}$ clusters is determined by finding the minimum value of $f(k)$.

Combining the estimation of $\hat{k}$ using the above three methods, we determined the optimal number of clusters to be 3.

Method for clustering

In order to find neurons that have correlated connectivity patterns to the rest of the network, we clustered the directed connectivity matrix by treating the connectivity pattern from each source neuron to all target neurons as features (Fig. 1F and Fig. S3). To reduce noise, we projected the connectivity features into a lower dimensional space with principal component analysis (PCA), only keeping the top principal components that explained 80% of total variance. We then applied a consensus clustering method (*50*) with k-means to obtain robust clusters that are not biased by random initial conditions. First, we constructed a co-clustering association matrix by running *k*-means with different initial conditions 100 times (reached stable co-clustering). Each entry in the matrix represents the probability of two units belonging to the same cluster. Then, we clustered the association matrix with hierarchical clustering to determine the cluster labels. The number of clusters was determined using methods described in the previous section.

Comparing different clustering methods

Our consensus clustering was based on *k*-means clustering methods, which measures the compactness of points based on features in the reduced PCA space (see above). We compared this clustering method with two other clustering methods to detect modular structure in the adjacency matrix: the spectral clustering method (sklearn.cluster.SpectralClustering) and bi-clustering method (sklearn.cluster.SpectralBiclustering).

Spectral clustering determines the clusters based on the connectivity of data points: points that are connected or immediately next to each other are placed in the same cluster. In spectral clustering, the data points are treated as nodes of a graph, and the clustering is treated as a graph partitioning problem. The nodes are then mapped to a low-dimensional space that can be easily segregated to form clusters. The spectral clustering is carried out in 3 steps: 1. Compute a similarity graph (*k*-nearest neighbors). 2. Project the data onto a low-dimensional space (compute Graph Laplacian, and eigenvalues and eigenvector for L). 3. Create clusters (based on the eigenvector corresponding to the 2nd eigenvalue to assign values to each node, then split the nodes with *k*-means for the given number of clusters).

Biclustering (or block clustering) is a method to simultaneously cluster the rows and columns of a matrix. For a *m* (sample) by *n* (feature) matrix, the algorithm generates biclusters, which are a subset of rows that exhibit similar connectivity pattern across a subset of columns.

The results of consensus clustering, spectral clustering, and biclustering of the functional connectivity matrix are shown in (Fig. S4C). The three methods showed relatively consistent clustering results (Fig.

S4D) in detecting units that belong to the three clusters dominated by different weight pattern. Therefore, our clustering findings are general and do not depend on the specific clustering method we used.

Cluster quality

We used *two methods,* which were previously used to evaluate spike sorting cluster quality (*14*), to quantify neuronal population cluster quality given different number of clusters (Fig. S4B). The *d*-prime (*d'*) was calculated using Fisher's linear discriminant analysis to find the line of maximum separation in PC space (*51*). d′ indicates the unbiased separability of the cluster of interest from all other clusters. The higher the value, the more distinguishable are the clusters. Hit-rate was calculated with nearest-neighbors method (n_neighbors = 3), which is a non-parametric estimate of exemplar contamination in each cluster. For each unit belonging to the cluster of interest, the three nearest units in principal-component space are identified. The "hit rate" is defined as the fraction of these units that belong to the cluster of interest. This metric is based on the "isolation" metric from (*52*). The higher the value, the less contamination in each cluster.

Module distribution

The area distribution of neurons within each module was quantified by calculating the proportional number of units in one area relative to the total number of units in all areas for a given module. The proportion of one module across areas sums to 1. To minimize sampling bias across areas, we subsampled units in each area to match the number of units across areas. The final result was a bootstrapped mean (sampling with replacement to match the number of units in each area; n_boot = 100). Error bars represents the bootstrapped standard deviation across all units in all mice. Results are only shown for the 'driver' and 'driven' modules. No systematic area bias was observed for cluster 1 (the cluster with non-significant connection) units (result not shown).

The distribution of each neuronal module across layers was quantified by first dividing units into superficial, middle, and deep layers according to the location of layer 4 estimated from the CSD (Fig. S6). We then calculating the proportion of units across these three layers for a given neuronal module. To minimize sampling bias across layers, we subsampled units in each layer to match the number of units across layers. Means and error bars were calculated using the same bootstrapping method as for the area distributions.

Graph creation

To create graphs visualizations (Fig. 3B,D), we first condensed our single-unit connectivity matrix to a single-recording-site connectivity matrix by combining units with peak channels on the same electrode. Then, we treat each site as a node in the graph. For an intuitive representation, nodes belonging to the same cortical area are close by and arranged clockwise from superficial to deep layer. The location of each area is determined by the top-down view of the physical locations of visual areas on the left hemisphere (Fig. 1A). The edges of the graph represent connections between sites, with red lines indicating projections from the source unit (positive weight) and blue lines indicating projections back to the source unit (negative weight). The threshold for significant connections is defined as an absolute weight larger than 10^{-6} coincidences/spike, which is half of the standard deviation of the weight distribution across all mice.

Divergence and convergence degree

Divergence degree is similar in concept to the outdegree of a graph. It is defined as the proportion of significant positive connections (weight > 10^{-6}) from a source neuron to the rest of the network (N neurons). $C_{i,+}$ represents the number of positive connections from neuron i to the network.

$$divergence = \frac{C_{i,+}}{N-1}$$

Convergence degree is similar in concept to the indegree of a graph. It is defined as the proportion of significant negative connections (weight < -10^{-6}) to source neuron i from the rest of the network.

$$convergence = \frac{C_{i,-}}{N-1}$$

Temporal dynamics analysis

Response latency

Two different measurements were used to estimate response latency. The peak response latency was defined as the time when a neuron's response reached its first peak after stimulus onset. The time to first spike was estimated in each trial by looking for the time of the first spike 30 ms after stimulus onset. If no spike was detected within 250 ms after stimulus onset, that trial was not included. The overall latency for each unit was defined as the mean time to first spike across trials.

Population onset response synchrony

We used the spread of time-to-first spike for all neurons within a module for a single trial as an indicator of population response onset synchronization. The spread was calculated by fitting a Gaussian to each trial's time-to-first-spike distribution:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2},$$

where x is the spike time relative to stimulus onset, μ is an estimate of the average time-to-first-spike and σ is an estimate of the spread of the time-to-first spike distribution for one trial.

Population response spread of the first peak

To quantify spike the response spread of a neuronal ensemble, we estimated the width of the population PSTH for each trial. The PSTH was calculated with 2 ms bins, and convolved with a Gaussian kernel of width 5 ms. The properties of the first peak were estimated using scipy.signal.find_peaks. The peak width represents the half-width at half maximum of the peak, while peak height represents the maximum of the peak. The spike spread of a neuronal population is an important parameter for quantifying how signals are transmitted through a feedforward network.

In-out index

To quantify the proportion of projections out from a source area relative to inputs back into the source area, we defined the in-out index as

$$index = \frac{C_{in} - C_{out}}{C_{in} + C_{out}}$$

where C_{out} is the number of connections from source to other areas in the given network and C_{in} is the number of connections from other areas to source area. This index reflects the asymmetry of in-and-out degree of a source area. When the value is close to -1, the source area is dominated by outward projections (positive weights). When the value is close to 1, the source area is dominated by inward projections (negative weights). When the value is 0, the source area has balanced outward and inward connections.

Layer definition

We estimated the depth of the middle layer of cortex by first calculating the current source density (CSD) using simultaneously recorded local field potentials (LFP) (Fig. S6). The CSD was computed using the method in (*53*), using the LFP within 250 ms after stimulus onset. First, we calculated the average evoked (stimulus locked) local field potential at each recording site. Next, we duplicated the uppermost and lowermost field traces and smoothed these signals across sites

$$\bar{\varphi}(r) = \frac{1}{4}\left(\varphi(r+h) + 2\varphi(r) + \varphi(r-h)\right)$$

where φ is the field potentials, r is the coordinate perpendicular to the layers, h is the spatial sampling interval (40 μm in our case). Then, we calculated the second spatial derivative

$$D = \frac{1}{h^2}\left(\bar{\varphi}(r+h) - 2\bar{\varphi}(r) + \bar{\varphi}(r-h)\right) .$$

In the resulting CSD map, current sinks are indicated by downward deflections and sources by upward deflections. To facilitate visualization, we smoothed the CSD with 2D Gaussian kernels ($\sigma_x = 1$; $\sigma_x = 2$). To find the middle layer, we defined the first sink within 100 ms after stimulus onset as the input layer (center channel) by searching for the local maximum on the CSD map (first sink), followed by source.

We used the middle layer estimation for two metrics in our paper. For the calculation of layer distribution bias of 'driver' and 'driven' modules, we partitioned the cortical layers into three layers: middle layer (center channel $\pm$ 8 channels, which is $\pm$ $40\mu\mathrm{m}$), superficial layer (channels above middle layer), and deep layers (channels below middle layer and above white matter). For the layer dependence of PSTH response latency (Fig. S6), we set the middle layer to depth = 0 and defined depths around the middle layer with 8-channel spacing ($40\mu\mathrm{m}$ spacing). Units within a depth range were grouped together to calculate a PSTH and the latency for that depth was estimated based on this grouped PSTH.

Simulations to test mathematical relationship between PSTH shape and CCG sharp peak

Because we observed that the functional connectivity defined 'driver' module responded earlier than the 'driven' module (Fig. 3D), we wondered whether the brief timescale relationship of 'driver' leading 'driven' was a consequence of the general latency reflected in averaged PSTH. Even though our jitter-correction method should have removed stimulus-locked components and the observed directionality should only reflect brief-timescale signal transmission, we still wanted to rule out the possibility that the observed asymmetry in the CCG is merely a reflection of the trial-averaged PSTH latency.

We used a simple simulation to carry out positive and negative controls (Fig. S7). The negative control tested whether two neurons with correlated, but temporally offset, PSTH traces will necessarily show significant peaks in their jitter-corrected CCG (25 ms jitter window). The positive control tested whether two neurons with uncorrelated PSTH traces can produce a significant peak in their CCG if we artificially introduce millisecond-timescale correlations. The mathematical expression of the tests is formulated as follows:

Given two PSTH traces: $\lambda_1(t)$ and $\lambda_2(t)$, we simulated *Poisson* spike trains:

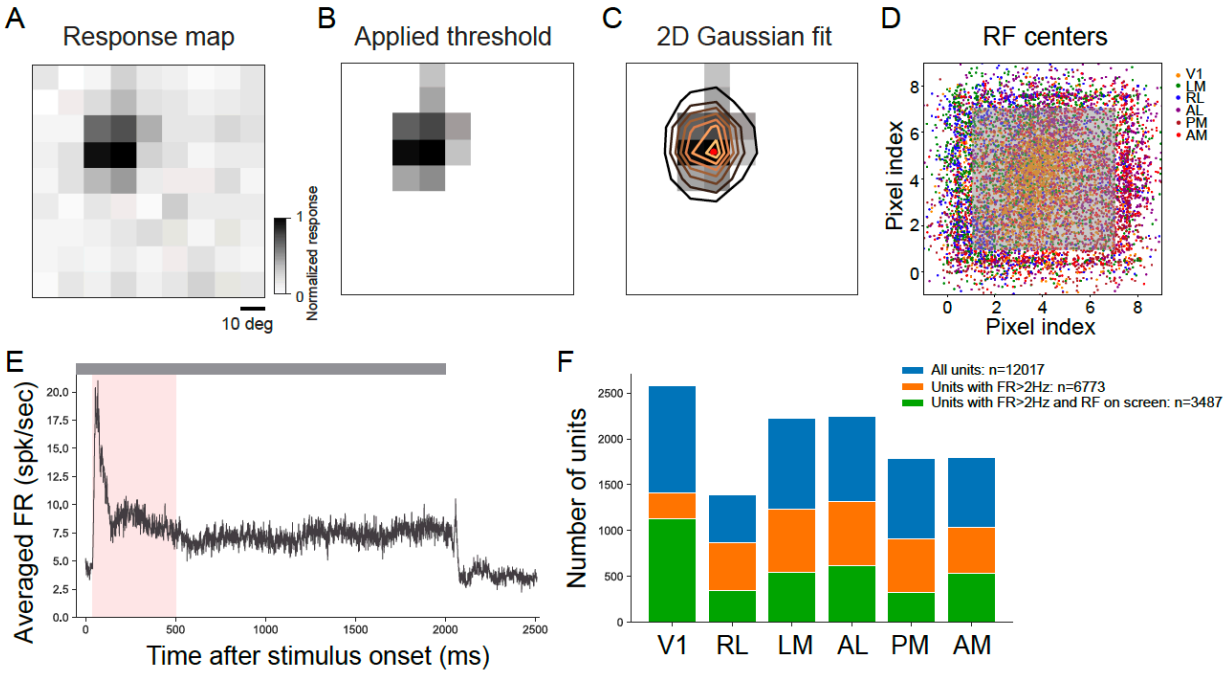
$$x_1(t) \sim Pois(\lambda_1(t))$$

$$x_2(t) \sim Pois(\lambda_2(t)),$$

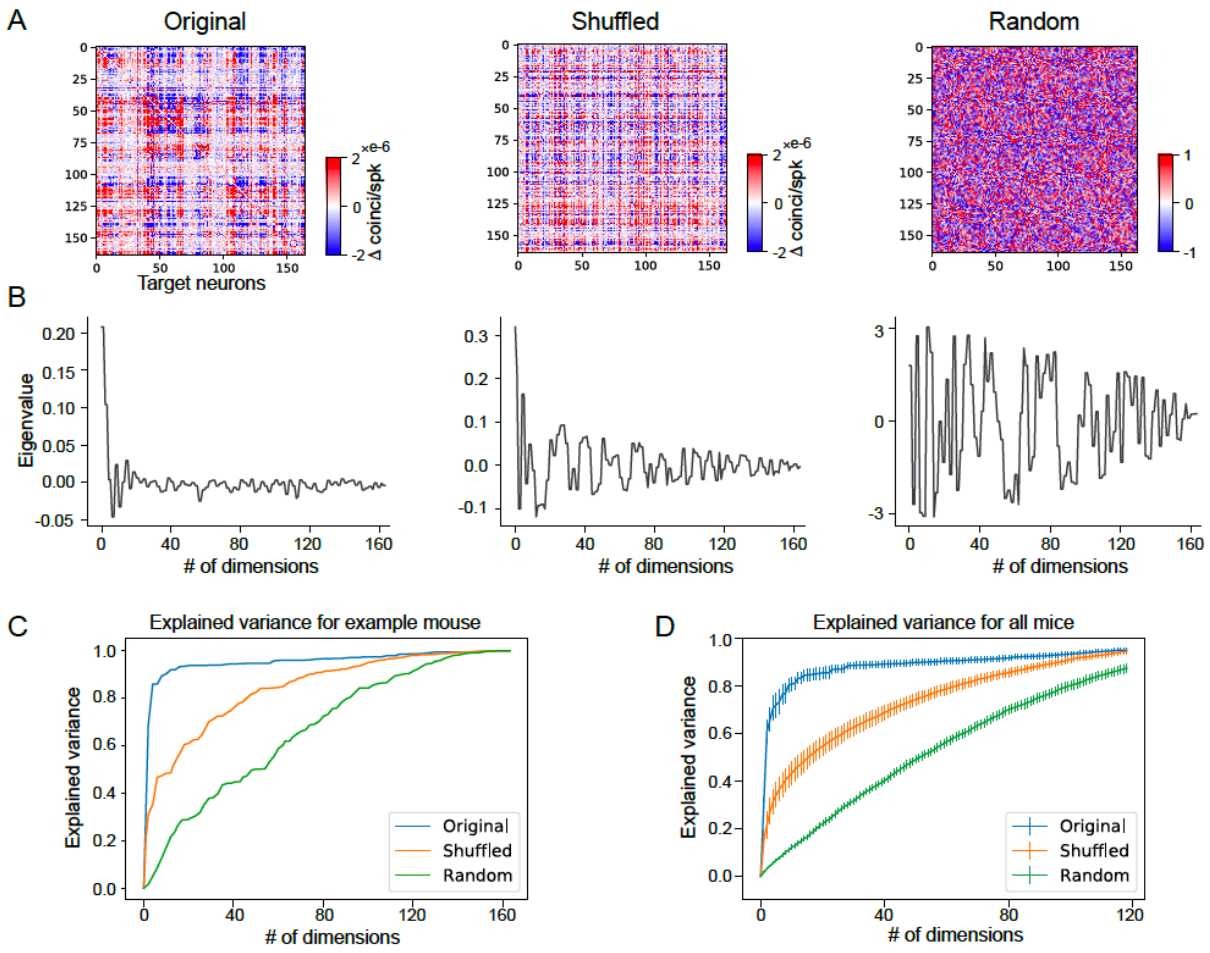
over time T for 100 repeats, where the PSTHs of the two simulated spike trains (X_1 and X_2) matched the shape of $\lambda_1(t)$ and $\lambda_2(t)$. Synchronized spikes were introduced to $x_1(t)$ and $x_2(t)$ only for the positive control. CCGs before and after jitter correction were calculated between X_1 and X_2. We found that brief timescale correlations between two neurons (identified by significant peaks in the CCG) do

not depend on the shape and relative timing of their PSTHs, but—as expected—reflect only their fine-timescale temporal relationship.

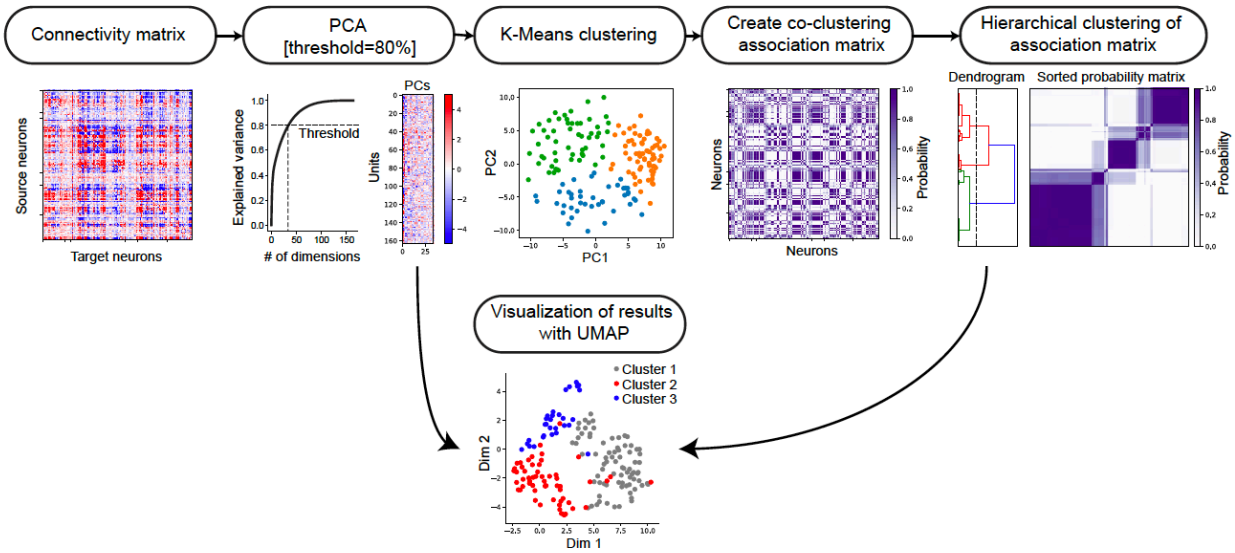
Data and code availability:
The majority of the data in this study (13 of 19 experiments) was publicly released as an open dataset on the Allen Institute website in October 2019, and is available via the AllenSDK (https://allensdk.readthedocs.io/en/latest/visual_coding_neuropixels.html). Additional data and software will be deposited to Github.



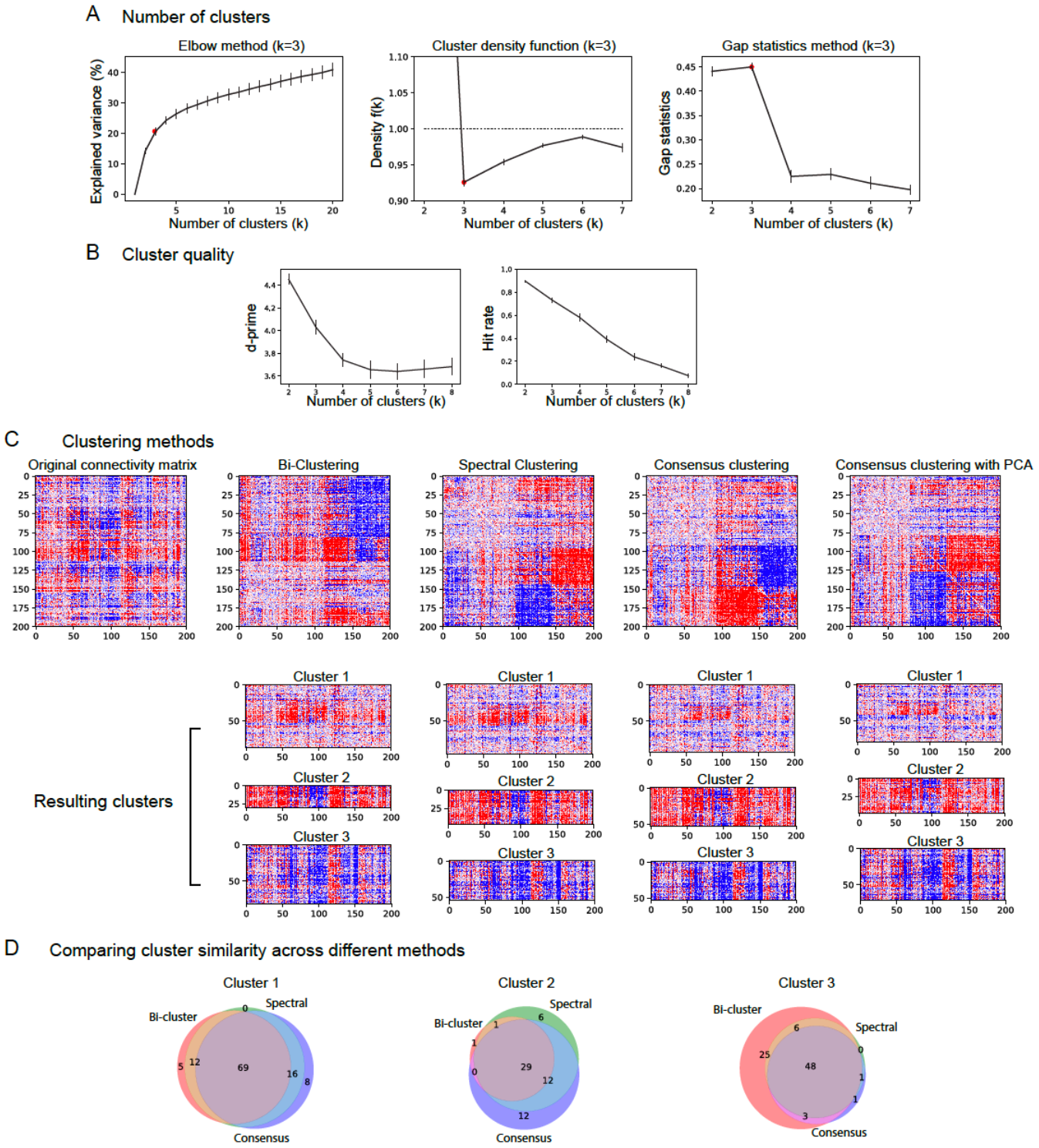
Supplemental Figure 1 Receptive field mapping and unit selection. A) Receptive fields were mapped with Gabor patches (20 degree each; 3 different directions (0º, 45º, 90º), temporal frequency = 2 cyc/sec, spatial frequency = 0.04 cyc/deg) shown randomly at 81 different locations (9 x 9 grid, 10º separation between location centers). The receptive field map for one unit is defined as the mean 2D histogram of spike counts at each of 81 locations. B) A threshold (20% of maximum) was applied to the response map to remove potential noisy pixels. C) A 2D Gaussian was fit to the filtered visual receptive map to estimate center of the receptive field location. D) Center of RFs from all units recorded in 19 mice with mean firing rate larger than 2 Hz. Different colors corresponds to different cortical areas. The gray box highlights the units with receptive field center on screen and at least 10 degrees away from the edge of the monitor, which was included for functional connectivity analysis. E) An example averaged PSTH in response to 2 second drifting grating stimuli (stimulus presentation duration is highlighted by gray bar). Averaged stimulus evoked firing rate (FR) was calculated between 50ms to 500ms after stimulus onset (high-lighted with pink) and averaged across orientations and trials for each units. F) Number of units in each visual cortical area that pass selection criteria. To have enough spikes for CCG calculation, we sub-selected units with FR > 2 spikes/sec. To maximize overlapped receptive fields and reduce edge effects, we further selected units with RF centers at least 10 degree away from the screen edge (see D). In total we recorded 12017 units in cortex across 19 mice. 56.4% remained after FR filter and 29.0% remained after filtering for both FR and receptive field center.



Supplemental Figure 2 Non-randomness of the functional connectivity matrix. A) Functional connectivity matrix from one example experiment (units are sorted by depth and areas). The shuffled matrix control was generated by randomly shuffling the x and y axes of the original matrix. The random matrix control was generated by simulating pure random connections between units. B) Eigenvalues of the three matrices in A). C) Explained variance based on eigenvalues in B). The original connectivity matrix showed the highest explained variance with relative few dimensions suggesting the presence of organized structure in the matrix compared to the shuffled and random controls. D) Quantification of explained variance for the measured and control connectivity matrices across mice (n=19 mice). Error bars indicate standard error of the mean.

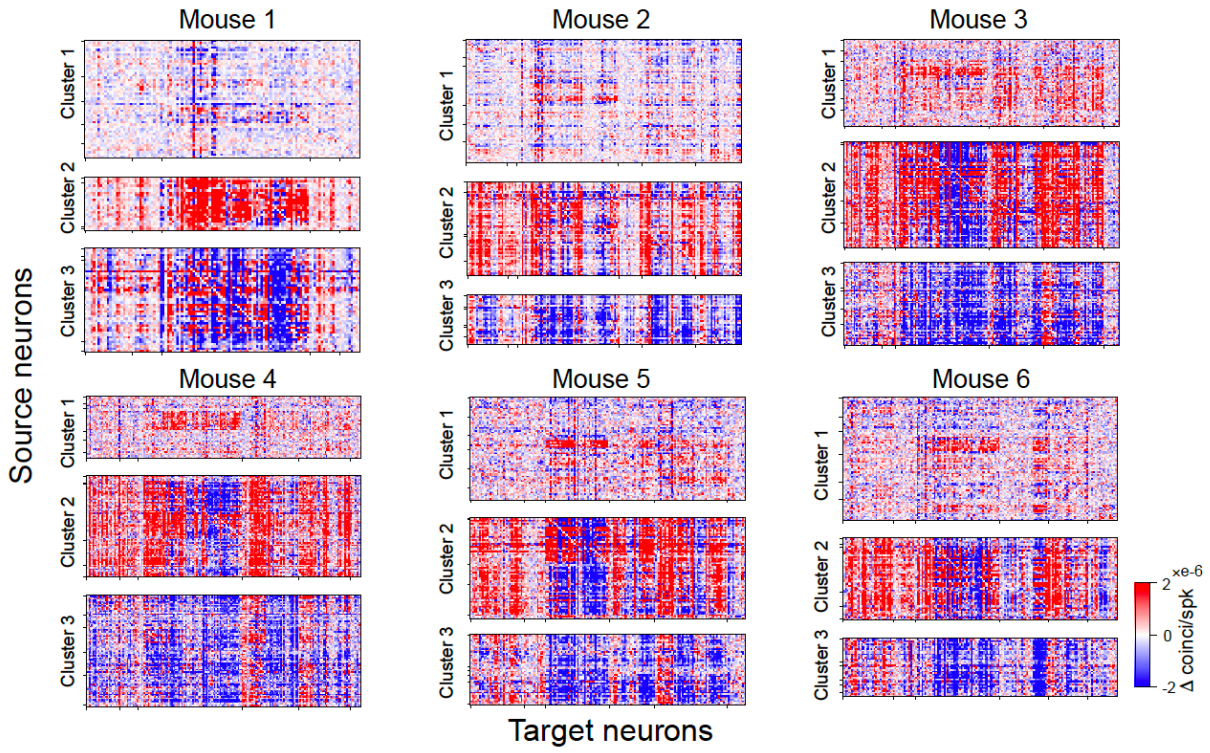


Supplemental Figure 3 Illustration of clustering method for identifying neuronal modules. Treating the connectivity pattern from each unit to the network as features, we first applied principle component analysis to reduce noise in the original connectivity matrix by keeping the top principle components that explain 80% of the total variance. Then we applied consensus clustering (*50*) to the PCs, by running K-means using the PCs 100 times until reaching a stable co-clustering association matrix, where each entry represents the probability of two units belonging to the same cluster. Finally, we applied hierarchical clustering to the association matrix and kept 3 clusters as the classification result, which was independently validated using the methods in Fig. S4A. As a result, the units are clustered based on their connectivity pattern into three modules. The bottom UMAP (*54*) is a visualization of 2D embedding with all the PCs and the layout agrees with the cluster labels from the consensus clustering result.

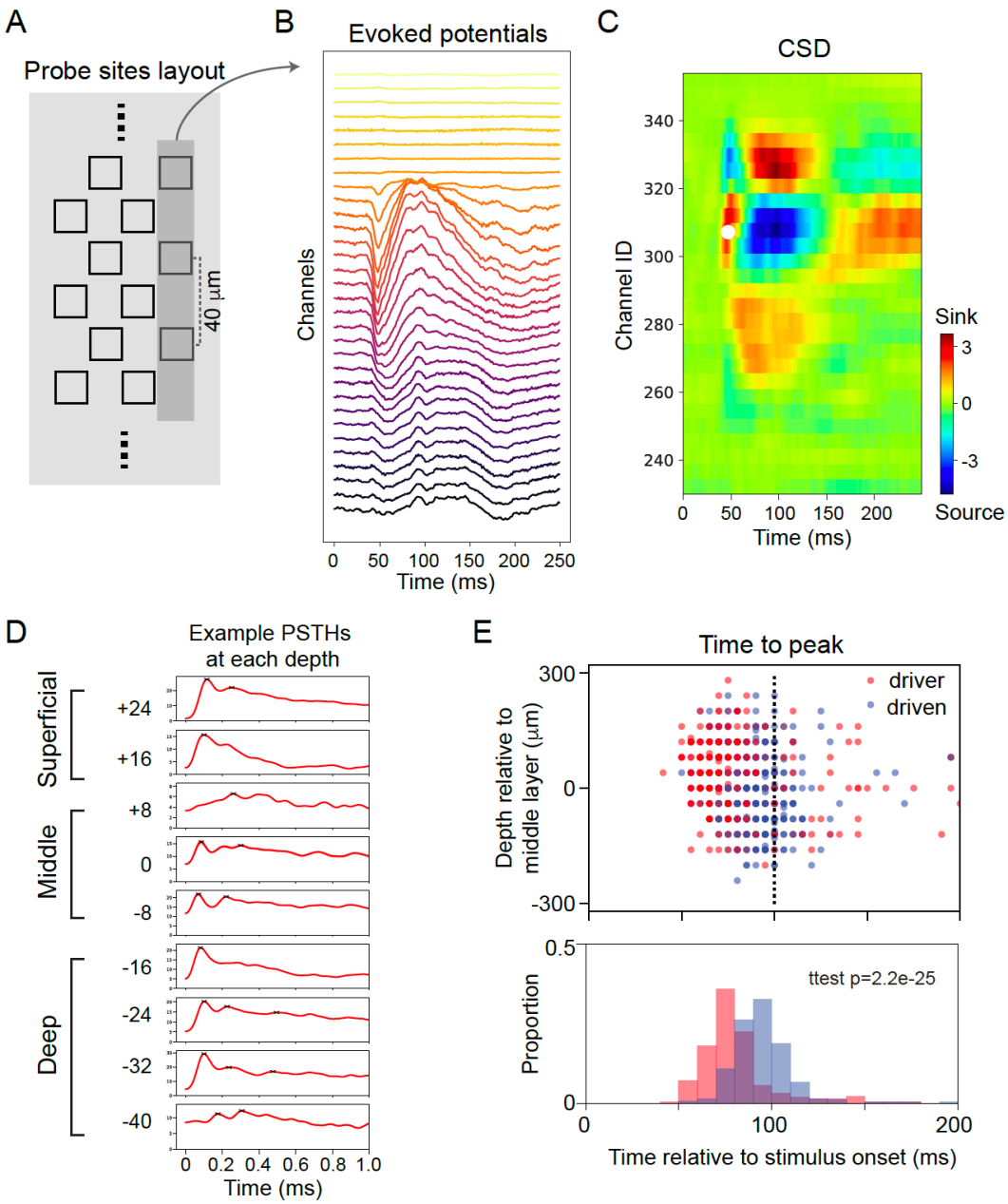


Supplemental Figure 4 Number of clusters, cluster quality, and comparison of clustering methods. A) Number of clusters estimated with three different methods (Elbow method; cluster density function; Gap statistics; see Methods for details). Errorbars indicate standard error across mice (n=19). B) Quantification of cluster quality as the number of clusters and estimation kernels. The d-prime (d') was calculated using Fisher's linear discriminant analysis to find the line of maximum separation in PC space. The higher the value, the more distinguishable the clusters are. Hit-rate was calculated nearest-neighbors method (n_neighbors=3), which is a non-parametric estimate of exemplar contamination in each cluster. For each unit belonging to the cluster of interest, the three nearest units in principal-component space are identified. The "hit rate" is defined as the fraction of these units that

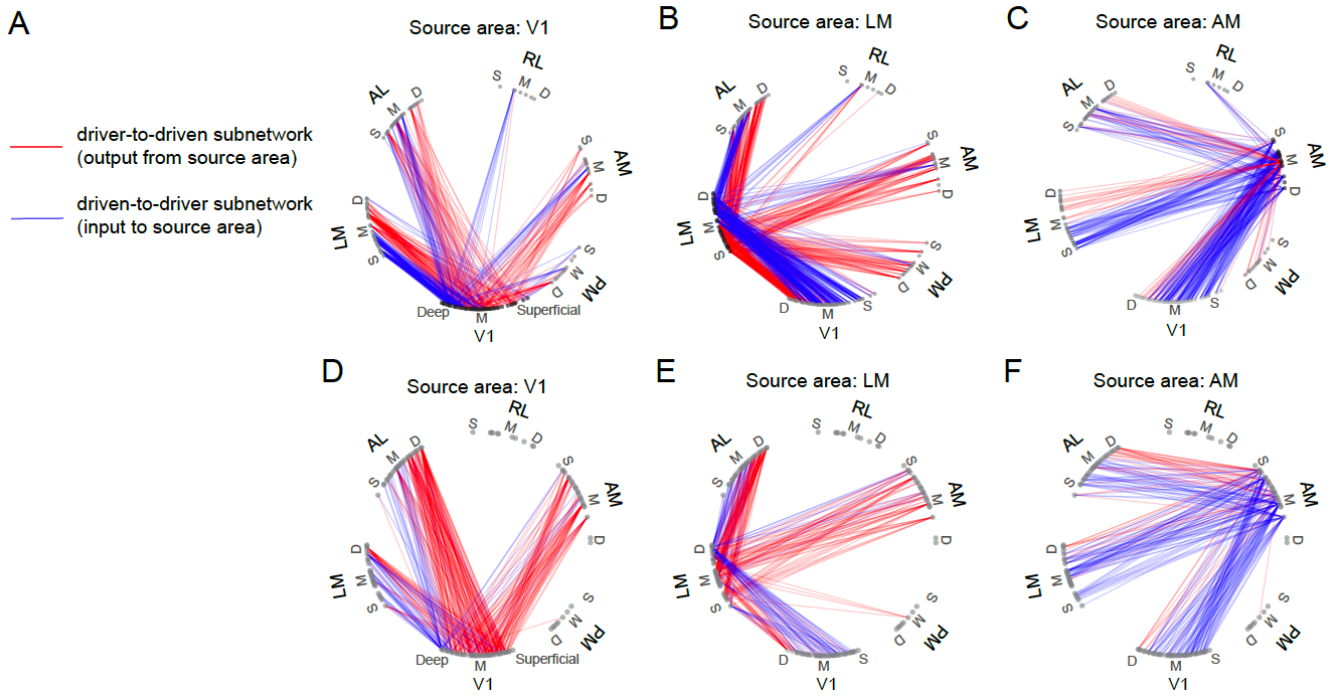
belong to the cluster of interest. This metric is based on the “isolation” metric from (*51*). The higher the value, the less contamination in each cluster. Error bars indicate standard error across mice (n=19). C) Original functional connectivity matrix and four different clustering methods: bi-clustering, spectral clustering, consensus clustering on original matrix, and consensus clustering after PCA (the method used in this paper). The lower plots show the resulting clusters from different methods. D) Venn diagram illustrating the overlapping degree of different clusters estimated with different methods from the example mouse in C. This result suggests that our clustering findings are fundamental and the clustered units are roughly similar across clustering methods.



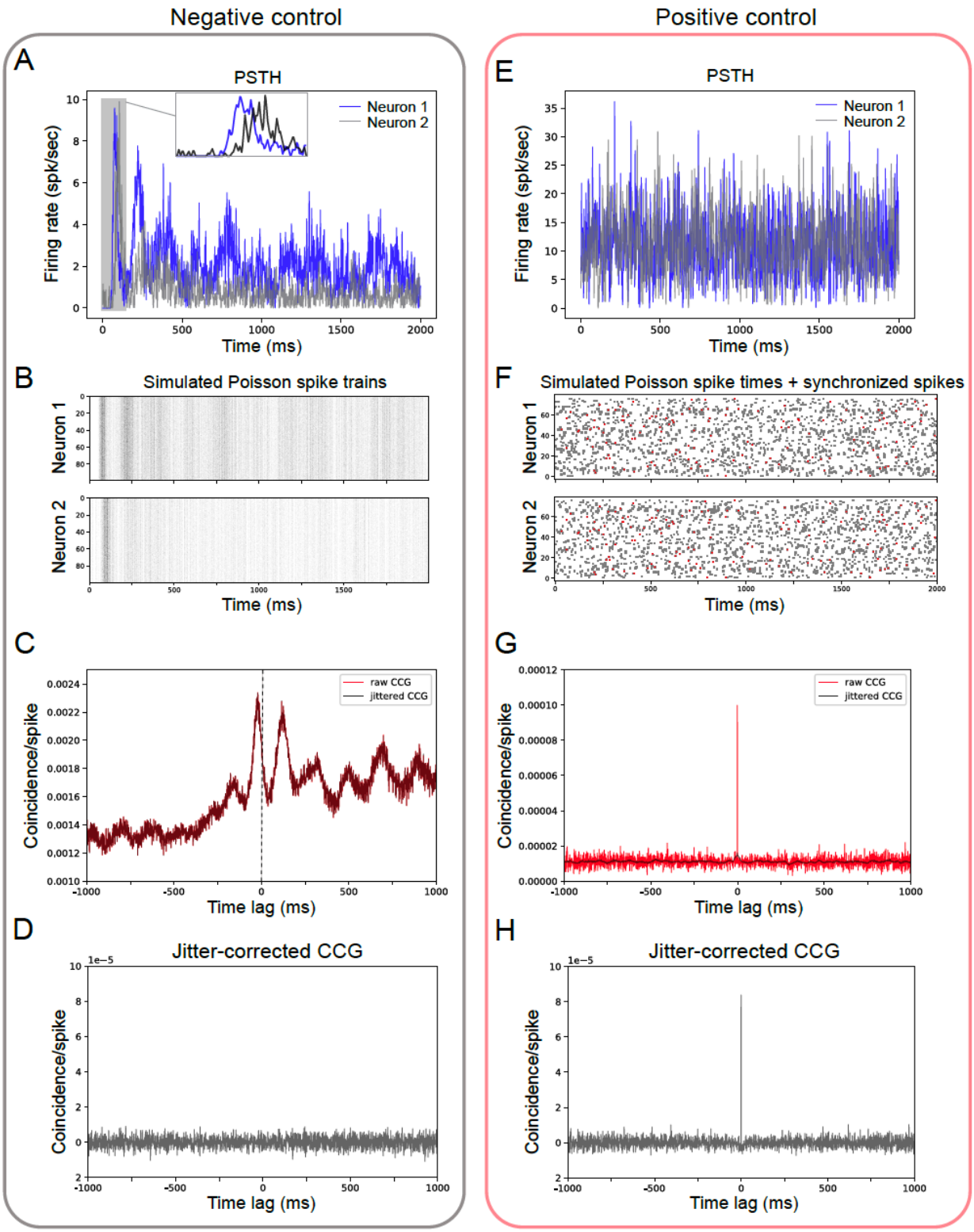
Supplemental Figure 5 Clustering results from different example mice. Cluster 1 corresponds to neurons dominated by weak connections. Cluster 2 neurons mostly make strong positive connections with the network, and therefore, we refer to this set of neurons as the 'driver' module. Cluster 3 neurons mostly make strong negative connections to the network, and therefore we refer to this set of neurons as the 'driven' module.



Supplemental Figure 6 Depth estimation and depth dependency of response latency for different modules. A) Probe sites layout. One vertical column of sites was used for current source density (CSD) analysis. Vertical spacing is 40 μm. B) Evoked local field potential from an example mouse in V1. C) CSD calculated based on evoked potential in B). Middle layer was determined by the first current sink, highlighted by white dot. D) Example PSTHs showing evoked spike rates at different depths. Channel at depth of first sink was set to 0 and middle layer was defined as ±8 channels around 0, which spans 160 μm. All channels above the middle layer were defined as "superficial," while all channels below the middle layers were defined as "deep." E) Response latency (peak of PSTH) at different depths plotted for 'driver' and 'driven' ensembles across 19 mice. 'Driven' units responded earlier than 'driver' units (T-test p = 2.2e-25) and were concentrated in the superficial layers.



Supplemental Figure 7. Overlay of driver-to-driven and driven-to-driver subnetworks for source areas V1, LM and AM. A) Directed graph representation for between-module subnetworks from source area V1 of one example mouse. Graph representation of strong connections (weight value > $2*10^{-6}$; larger weight value threshold for visualization purpose). Line color indicates the directionality of connection. *Red*: output from driver neurons in V1 → driven neurons in other areas; *blue*: input from driver neurons in other areas → driven neurons in V1. Each node in the graph represents a recording site in cortex. Identity of visual areas is indicated by node color and arranged according to their anatomical location. Nodes in each area are arranged clockwise from superficial to deep layers. There is a clear separation of input and output from source area V1 and the separation is cortical depth dependent. B) Directed graph representation for between-module subnetworks from source area LM of the same example mouse as in A). C) Directed graph representation for between-module subnetworks from source area AM of the same example mouse in A). D, E, F) Directed graph of overlaid subnetworks of another example mouse.



Supplemental Figure 8 Testing the relationship between PSTH shape and CCG sharp peak. Two simulations were performed to test the relationship between the PSTH temporal dynamics of two neurons and their fast-timescale correlations, as reflected in the jitter-corrected CCG. The negative control was done by generating Poisson spike times with correlated PSTH temporal profiles; then, we tested whether there was a sharp peak in the jitter-corrected CCG with jitter window=25ms. A) PSTHs of two neurons (1 ms bin) with temporal Pearson's correlation equal to 0.35. Neuron 1 has an onset earlier than neuron 2. B) Simulated Poisson

spike trains based on the PSTHs in A) for the two neurons across repeats. C) Raw CCG (red line) and jittered CCG (calculated by jittering spike times large than 25 ms; black line) between neuron 1 and 2 (the probability of neuron 1 generating a spike relative to each spike in neuron 2) calculated based on the simulated Poisson spike times B). Both CCGs showed a clear peak at a negative time lag, reflecting the fact that over long timescales, neuron 1 fires earlier than neuron 2, consistent with the temporal profile of the PSTHs. D) Jitter-corrected CCG calculated by subtracting jittered CCG from raw CCG in C). Because there is no fast timescale correlation within 25ms, predetermined by Poisson process, we didn't observe any sharp peak in jitter-corrected CCG. This negative control indicates that a temporal correlation in the PSTH profile does not necessarily lead to a sharp peak in the CCG.

The positive control was simulated by generating Poisson spike times given uncorrelated PSTH temporal profiles but adding additional synchronized spikes to the two simulated spike trains. Then, we tested whether the two spike trains showed any brief timescale correlation with jitter-corrected CCG. E) Uncorrelated PSTHs (Pearson's r=0.01) of two neurons. F) Simulated Poisson spike train based on the PSTHs in E) with synchronized spikes (10% of total spikes; highlighted in red) added to both neuron 1 and 2. G) Raw and jittered CCGs between simulated spike trains F). H) Jitter-corrected CCG calculated by subtracting jittered CCG from raw CCG. The jitter-corrected CCG showed a sharp peak at 0 time lag, indicating synchronization of spike times between the two neurons. This result is consistent with the injection of synchronized spikes, despite the fact that their PSTH profiles are uncorrelated.