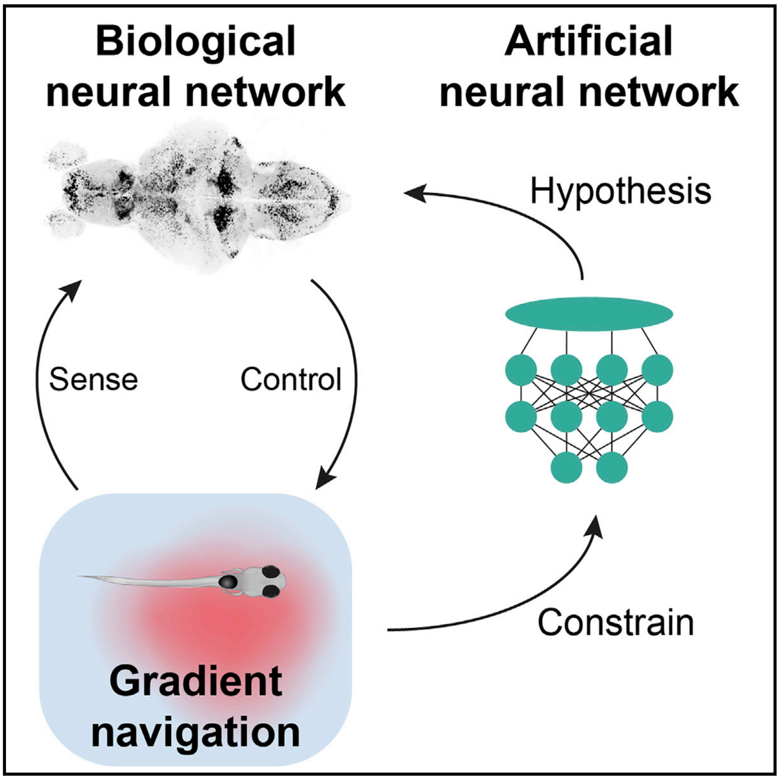
Neuron

Article

Convergent Temperature Representations in Artificial and Biological Neural Networks

Graphical Abstract



Authors

Martin Haesemeyer,
Alexander F. Schier, Florian Engert

Correspondence

haesemeyer@fas.harvard.edu (M.H.),
florian@mcb.harvard.edu (F.E.)

In Brief

Haesemeyer et al. train convolutional neural networks to navigate temperature gradients to reveal shared representations and processing in artificial and biological networks. Constrained by zebrafish behavior, artificial networks critically rely on fish-like units and make testable predictions about the brain.

Highlights

- Artificial neural networks (ANNs) were trained to navigate heat gradients
- ANN temperature representations were compared to a zebrafish whole-brain dataset
- Temperature representations in the ANN bear striking similarities to zebrafish
- ANN responses were used to identify a novel temperature response type in zebrafish

Haesemeyer et al., 2019, Neuron *103*, 1123–1134
September 25, 2019 © 2019 Elsevier Inc.
https://doi.org/10.1016/j.neuron.2019.07.003

Convergent Temperature Representations in Artificial and Biological Neural Networks

Martin Haesemeyer,[1,*] Alexander F. Schier,[1,2,3,4,5] and Florian Engert[1,2,6,*]
[1]Department of Molecular and Cellular Biology, Harvard University, Cambridge, MA 02138, USA
[2]Center for Brain Science, Harvard University, Cambridge, MA 02138, USA
[3]Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA
[4]Harvard Stem Cell Institute, Cambridge, MA 02138, USA
[5]Biozentrum, University of Basel, 4056 Basel, Switzerland
[6]Lead Contact
*Correspondence: haesemeyer@fas.harvard.edu (M.H.), florian@mcb.harvard.edu (F.E.)
https://doi.org/10.1016/j.neuron.2019.07.003

SUMMARY

Discoveries in biological neural networks (BNNs) shaped artificial neural networks (ANNs) and computational parallels between ANNs and BNNs have recently been discovered. However, it is unclear to what extent discoveries in ANNs can give insight into BNN function. Here, we designed and trained an ANN to perform heat gradient navigation and found striking similarities in computation and heat representation to a known zebrafish BNN. This included shared ON- and OFF-type representations of absolute temperature and rates of change. Importantly, ANN function critically relied on zebrafish-like units. We furthermore used the accessibility of the ANN to discover a new temperature-responsive cell type in the zebrafish cerebellum. Finally, constraining the ANN by the *C. elegans* motor repertoire retuned sensory representations indicating that our approach generalizes. Together, these results emphasize convergence of ANNs and BNNs on stereotypical representations and that ANNs form a powerful tool to understand their biological counterparts.

INTRODUCTION

Neural network models such as the perceptron (Rosenblatt, 1962) and parallel distributed processing models (Rumelhart et al., 1987) have been used to derive potential implementations of cognitive processes, such as word perception (McClelland and Rumelhart, 1981; Rumelhart and McClelland, 1982) or attentional control (Cohen et al., 1990). These models demonstrated how complex computations could emerge from networks of simple units suggesting that cognitive processes could be implicitly realized in connectivity weights rather than relying on a diversity of computational units. Indeed, artificial neural networks (ANNs) are increasingly successful in solving tasks long considered hallmarks of cognition in biological neural networks (BNNs). This includes visual discrimination tasks, playing chess and Go, as well as spatial navigation (Banino et al., 2018; Cueva and Wei, 2018; Krizhevsky et al., 2012; Logothetis and Sheinberg, 1996; Moser et al., 2008; Silver et al., 2016; Trullier et al., 1997).

While ANN design principles have been inspired by discoveries in BNNs (Hassabis et al., 2017), it is controversial whether both network types utilize the same fundamental principles and hence to what extent ANNs can serve as models of animal cognition (Lake et al., 2017). However, if representations and algorithms are shared between BNNs and ANNs, then ANN models could be used to guide the analysis of large-scale biological datasets that are becoming prevalent in modern neuroscience (Engert, 2014). And indeed, studies comparing visual processing with image classification ANNs (Khaligh-Razavi and Kriegeskorte, 2014; Yamins and DiCarlo, 2016) as wells as studies on networks performing spatial navigation (Banino et al., 2018; Cueva and Wei, 2018) uncovered parallels in representation between ANNs and BNNs.

We recently used whole-brain calcium imaging and modeling to characterize how larval zebrafish process temperature information to generate behavioral output underlying heat gradient navigation (Haesemeyer et al., 2018). This uncovered critical neural temperature response types in the larval zebrafish hindbrain that broadly fall into two classes, ON and OFF cells that represent changes with opposing sign. Within these classes, a set of neurons reports temperature levels, while another set encodes the rate of change. Since larval zebrafish readily navigate thermal gradients (Gau et al., 2013; Haesemeyer et al., 2015), we now generated and trained deep convolutional neural networks to solve a heat gradient navigation task using a larval zebrafish behavioral repertoire. This approach allowed us to compare stimulus representation and processing in biological and artificial networks that solve the same behavioral task using our rich biological dataset. We found that these behavioral constraints led to striking similarities in temperature processing and representation in the ANN with the zebrafish biological circuits. Namely, the model parallels temperature representation in ON and OFF-type units as well as ANN units showing sustained and adapting responses, effectively encoding both temperature levels and rates of change. Importantly, ANN performance critically relied on units representing temperature in a fish-like manner, while other nodes were dispensable for network function.

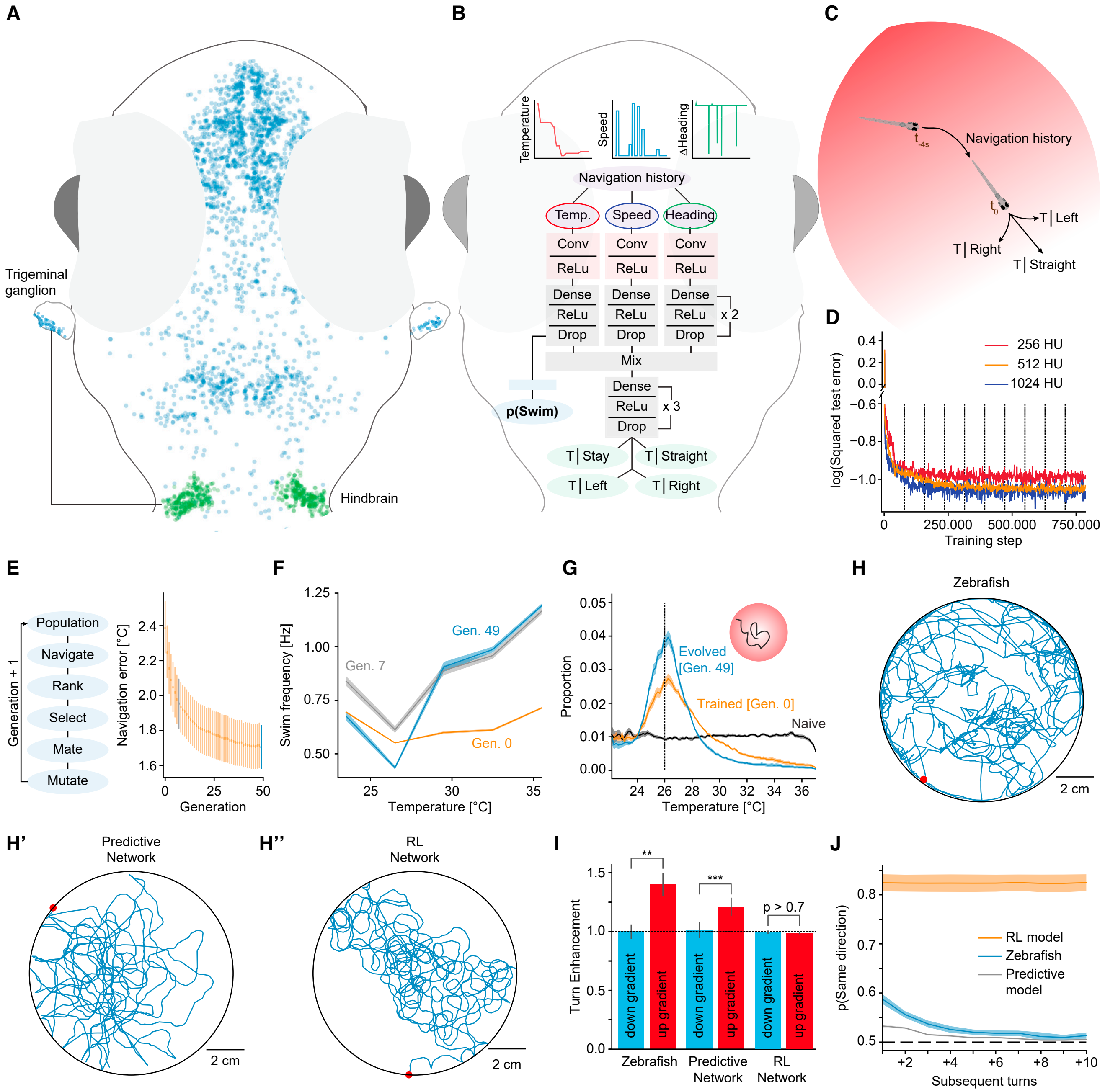


Figure 1. A Convolutional Network for Gradient Navigation

(A) Location of temperature modulated neurons (blue) in the zebrafish brain and sensory trigeminal ganglia. Temperature modulated neurons in a hindbrain processing area are in green.

(B) Structure of the convolutional network for zebrafish temperature prediction. Curves on top depict example network input of the training dataset. Conv, convolutional layer; ReLu, network uses rectifying linear units; Drop, dropout during training.

(C) Schematic of the network task. Given temperature and movement history in a heat gradient, the network predicts the resting temperature after different behavior selections (stay, move straight, turn left, or turn right).

(D) Log of the squared error in temperature predictions on a test dataset after the indicated number of training steps (dashed vertical lines demarcate training epochs).

(E) Evolutionary algorithm to learn p(Swim) weights (left panel) and progression of heat gradient navigation error across generations (right panel). Error bars are bootstrap SE across 20 evolved networks. Generation 7 in gray and last generation in blue for comparison with (F) and (G).

(F) For fully trained predictive network in generation 0 (orange), evolved generation 7 (gray) and network after completed evolution (blue) the average swim frequency by temperature in the gradient.

(legend continued on next page)

We next used the accessibility of the ANN to uncover new features of the zebrafish BNN. This allowed identification of a novel neuronal response type in the zebrafish brain that was predicted by the ANN but escaped detection in previous calcium imaging experiments. Finally, our approach generalizes since training the same ANN constrained by the *C. elegans* motor repertoire resulted in distinct neural representations, indicating that the behavioral means of attaining a goal can influence sensory representations.

These results indicate that behavioral constraints can lead to convergence on stereotypical stimulus representations in ANNs and BNNs, and they demonstrate the utility of ANNs to gain insight into processing in biological networks.

RESULTS

ANN Models for Heat Gradient Navigation

We sought comparing temperature processing in zebrafish with that of ANNs that share the same behavioral goal, namely, heat gradient navigation. We designed two convolutional neural networks with rectifying linear units; the first explicitly predicts the consequences of behaviors within a gradient (Figures 1A–1C), while the other is a reinforcement learning model enacting a behavioral policy (Figure S1B). Four seconds of sensory and behavioral history experienced during virtual gradient navigation served as inputs to the networks since this timescale matches convolutional timescales in a zebrafish circuit model (Haesemeyer et al., 2018). We, however, did not match ANN connectivity to zebrafish circuitry to avoid constraining network representations by anatomy, but instead limit constraints as much as possible to the behavioral goal and the available motor repertoire.

Previously, we observed heat-responsive neurons encoding the rate and direction of temperature change in the larval zebrafish hindbrain and that therefore allow for a simple form of prediction (Haesemeyer et al., 2018). Inspired by this finding, we designed our first network to predict the temperature reached after enacting one of four zebrafish behavioral elements: stay in place, swim straight, turn left, or turn right (Figure 1B). Importantly, this design choice is biologically plausible given the ubiquitous importance of predictive forward models in decision making across animal phyla (Ahrens et al., 2012; Mehta and Schaal, 2002; Wolpert and Miall, 1996; Mischiati et al., 2015; Portugues and Engert, 2011).

We subsequently trained this predictive ANN using backpropagation on training data that were generated from a random walk through a radial heat gradient by drawing swims from distributions observed during zebrafish heat gradient navigation (Haesemeyer et al., 2015). We used drop-out (Srivastava et al., 2014) during training to mimic redundancy and noise generally observed in BNNs. Comparing hidden layer sizes (256, 512, or 1,024 units per layer), performance on a test dataset saturated with 512 hidden units per layer (Figure 1D). We therefore chose this size for our ANN, but we did not fully explore the architectural space of convolutional ANNs, and it could therefore be that much simpler architectures solve the task just as efficiently. Trained networks successfully predicted temperatures reached after typical swims with average errors <0.1°C. To use this prediction for gradient navigation, we implemented a behavioral rule that favors those actions that bring the network closer to a set target temperature (Figure S1A). Invocation of this rule after training led to efficient gradient navigation with an average distance from the target temperature of 2.4°C (SD 0.6°C) compared to an average distance of 4.6°C (SD 0.4°C) in naive networks.

While larval zebrafish swim in discrete bouts occurring on average at a frequency of 1 Hz, they modulate swim frequency by temperature (Haesemeyer et al., 2015), which could be a useful feature in the context of gradient navigation. We therefore extended the network with a module controlling swim probability independent of the already trained predictive function (Figure 1B; p(Swim)). To accomplish this task, the p(Swim) module uses a set of weights to transform the output of the temperature processing branch of the network into a swim probability similar to larval zebrafish where temperature-responsive hindbrain neurons control swim probability (Haesemeyer et al., 2018). We trained these weights using an evolutionary algorithm. Such algorithms can efficiently minimize empirical costs such as heat gradient navigation errors without needing an explicitly defined cost function as required for backpropagation. This approach led to convergence on a specific set of weights for each of 20 trained networks within 50 generations (Figure 1E; Figures S1C–S1E). Importantly, these weights led to increases in swim frequency as temperature departs from preferred values (Figure 1F), which is also observed in larval zebrafish (Gau et al., 2013; Haesemeyer et al., 2015; Prober et al., 2008). This enhanced navigation performance of the network (Figure 1G) reducing the average distance to the preferred temperature from 2.4°C (SD 0.6°C) to 1.7°C (SD 0.6°C).

As an alternative model, we designed an ANN that directly enacts a behavioral policy consisting again of the four behavioral elements "stay in place," "swim straight," "turn left," or "turn right" (Figure S1B). We trained this model via a reinforcement learning strategy that rewarded movements in a virtual heat gradient based on how they influenced the distance to the target temperature (see STAR Methods). Such networks learned to reduce navigational errors within a heat gradient (Figure S1F) and closely matched the performance of our predictive model (Figures S1F and S1G).

(G) Radial heat gradient occupancy of naive (black), trained (orange), and evolved (blue) networks. Dashed vertical line at 26°C indicates desired temperature.

(H–H″) Example trajectory for 30 min of navigation in a circular gradient for larval zebrafish (H), the predictive ANN (H′), and the reinforcement learning ANN (H″).

(I) Average turn angles of zebrafish or indicated networks when the last swim bout was going toward the preferred temperature (blue bars) or away (red bars) relative to the same direction in a non-gradient condition. **Wilcoxon test across n = 25 fish, p = 0.0027; ***Wilcoxon test across n = 20 networks, $p = 8.857 \times 10^{-5}$; p > 0.7: Wilcoxon test across n = 20 networks, p = 0.7938.

(J) Plot of turn coherence for larval zebrafish (blue), predictive ANN (gray), and reinforcement learning ANN (orange) during heat gradient navigation. For successive turns, the probability of turning into the same direction as turn 0 is plotted. Dashed line indicates chance level.

Shading in all panels indicates bootstrap SE across 25 fish or 20 networks, respectively. See also Figure S1.

To gain further insight into the behavioral strategy employed by the trained predictive and reinforcement learning ANN, we compared gradient navigation behavior in larval zebrafish to that of the two ANN types (Figures 1H–1H′′). Larval zebrafish not only modulate their bout frequency based on current temperature but also modulate turning based on temperature changes experienced during the last swim bout. Specifically, turning is enhanced when moving up a heat gradient, effectively reorienting larval zebrafish when departing from desired temperatures (Figure 1I). Interestingly, this strategy is learned by the predictive network but absent in the reinforcement learning network (Figure 1I). The reinforcement learning network instead learned a different strategy becoming either left or right biased (cf. Figures S1I–S1I′′) and thereby circling near the target temperature (Figure 1H′′). While not biasing turning overall (Figures S1I and S1I′), both larval zebrafish and the predictive network string together turns of the same direction within the heat gradient. Namely, the probability of maintaining direction across following turns was initially greater than chance but decayed back to chance level in zebrafish and the predictive ANN, while no such decay was observed for the reinforcement learning ANN (Figure 1J). Such turn coherence has been reported for zebrafish spontaneous behavior as well (Dunn et al., 2016), and within a gradient it likely allows for persistent reorientation without an overall turn bias. We had previously presented time varying heat stimuli to larval zebrafish and now compared behavior generation in response to these stimuli between zebrafish and our predictive ANN. While rates of swimming in response to these stimuli only correlated with a Pearson coefficient of 0.44 between zebrafish and the ANN (Figure S1J), both zebrafish and the ANN increased turning during the rising phase of the stimulus compared to the falling phase (Figure S1K). This argues that also for faster stimulus timescales zebrafish and the predictive ANN generate comparable behavior.

In summary, we designed and trained ANNs to navigate a heat gradient using both a supervised and a reinforcement learning strategy. Behavioral comparisons suggest that a network designed to predict the consequences of movements within a heat gradient mimics important elements of the behavioral strategy displayed by zebrafish when navigating temperature gradients.

Representation in the ANN Parallels the Zebrafish Brain

We next sought to compare temperature representation between the ANN and the zebrafish BNN, which contains a critical set of temperature encoding cells within the hindbrain. These consist of ON and OFF-type cells sensitive to temperature levels on the one hand (named “Slow ON” and “Slow OFF”) and changes in temperature on the other (named “Fast ON” and “Fast OFF” cells) (Haesemeyer et al., 2018). Spectral clustering across units revealed a very similar representation in the temperature navigation ANN (Figures S2A and S2B). Correlation analysis between ANN response types and zebrafish cell types revealed clear response similarities, with a corresponding cell type within the ANN for each zebrafish cell type with correlations r > 0.6 (Figure 2A). Comparing responses of the matching types revealed that indeed two mimicked Fast ON and Slow ON activity found in the larval zebrafish hindbrain (Figure 2B), while another two paralleled Fast OFF and Slow OFF activity (Figure 2C). This similarity in stimulus encoding highlights convergence in representation and information processing between larval zebrafish and the ANN.

Using a linear model relating ANN units to calcium activity in every zebrafish neuron (Yamins et al., 2014; see STAR Methods) revealed a striking overlap between originally identified zebrafish heat cells across the brain (Figure S2C, left) and those cells that were well predicted by ANN activity (Figure S2C, right). Importantly, the overlap is much larger than expected by chance (Figure S2C′). Overall the ANN correspondence recovered 31% of all heat encoding cells in the zebrafish brain. This recovery is expected given trial-to-trial variability in heat cell calcium responses (Haesemeyer et al., 2018).

Encouraged by these similarities, we tried to use other prominent response types found in the ANN to identify heat processing cells in the larval zebrafish brain that may have been missed by previous clustering approaches. In particular, the ANN contained two abundant response types that were quite different from cell types previously described in zebrafish: a group of units responding to both stimulus on- and off-set (Figure S2D) as well as a type that we termed “integrating OFF” as it was most active at low temperatures and integrated over successive temperature decreases during the sine period of the stimulus (Figure 2D). We used these novel cell types as regressors (Miri et al., 2011) to search the larval zebrafish brain data for cells with highly correlated responses. While we couldn’t identify cells that properly matched the response properties of the ON-OFF type (Figure S2D), there was a group of cells with activity closely resembling the “integrating OFF” type (Figure 2D). Importantly, these cells clustered spatially in the larval zebrafish brain, where most of them were located in a tight band in the rostro-dorsal cerebellum (Figure 2D′, arrow). This anatomical clustering strongly supports the idea that these cells indeed form a bona-fide heat-responsive cell type.

By design the ANN connectivity was not matched to the BNN of larval zebrafish. In particular, temporal convolution occurs at both the sensory neuron level as well as in the hindbrain in larval zebrafish (Haesemeyer et al., 2018), while our ANN only has one convolutional layer. Accordingly, all ANN response types already arise at the first layer of the network, while their abundance changes between layers (Figure S2B). Nonetheless, analysis of connectivity weights between the hidden layers in the temperature branch of the network revealed that zebrafish-like types receive on average stronger inputs from other zebrafish-like types than from non-fish types (Figure 2E). This suggests that zebrafish-like response types form a sub-network within the ANN.

Comparing stimulus representations between the reinforcement learning network and larval zebrafish following the same correlation strategy revealed matching representations as well (Figure S2E). However, this network lacked a response type with a correlation >0.6 to Fast OFF neurons (Figure S2E). In fact, the unit type best correlated to zebrafish Fast OFF cells was equally well correlated to zebrafish Slow ON cells.

The observed representations could be a general solution to navigational tasks that use time varying inputs, or they could be specific to spatial gradient stimuli. To disambiguate these hypotheses, we designed a network with a behavioral goal akin to phototaxis (Chen and Engert, 2014; Huang et al., 2013; Wolf

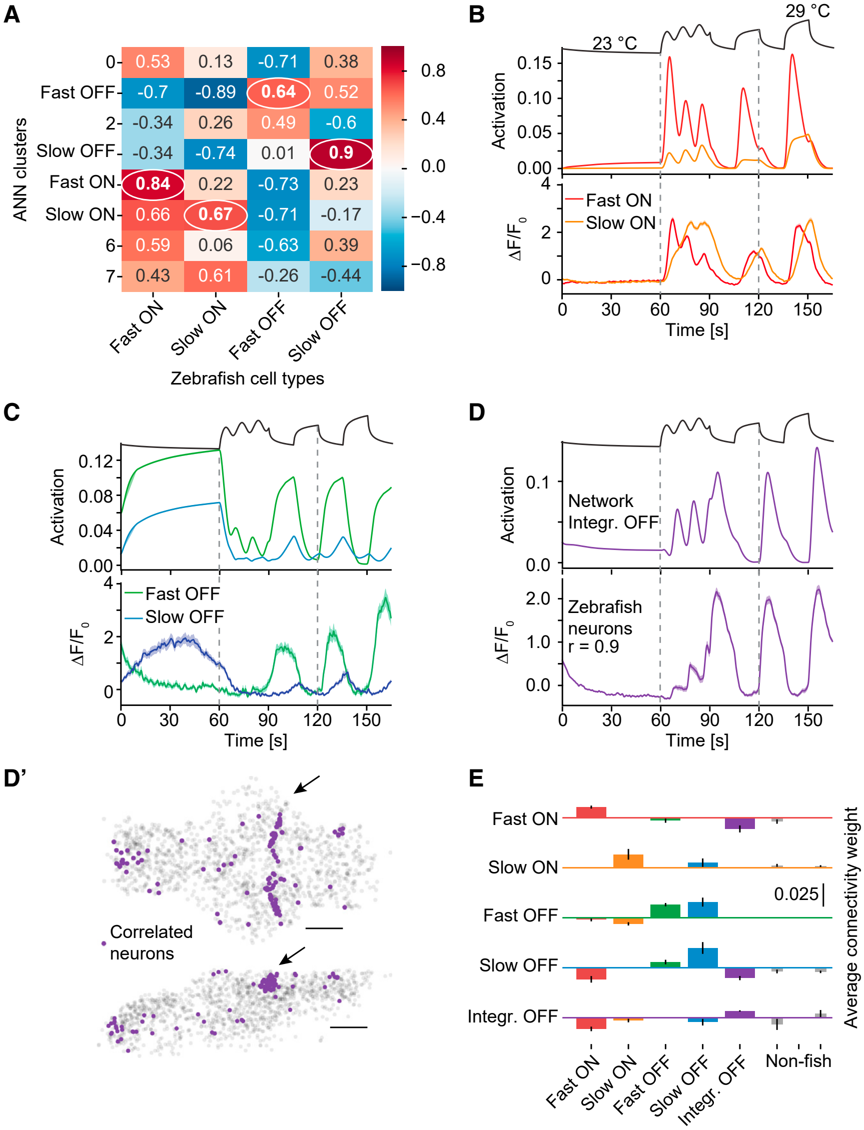


Figure 2. The Network Learns a Zebrafish-like Neural Representation

(A) Activity correlation between previously described zebrafish hindbrain heat response types and all identified response types in the predictive ANN given the same stimulus. The naming refers to the names given to the cell types in Haesemeyer et al. (2018). Circles indicate matched types with a Pearson correlation > 0.6.

(B) Responses of fish-like ON cell types assigned by the correlation in (A). Top panel: network responses of adapting "Fast ON" cells (red) and non-adapting "Slow ON" cells (orange) in the network. Bottom panel shows corresponding zebrafish calcium responses for comparison. Stimulus is depicted in gray on top for reference; vertical dashed lines indicate example rising and falling phase starts.

(C) Responses of fish-like OFF cell types assigned by the correlation in (A). Top panel: network responses of adapting "Fast OFF" cells (green) and non-adapting "Slow OFF" cells (blue) in the network. Bottom panel shows corresponding zebrafish calcium responses for comparison.

(D) "Integrating OFF cell" type identified in the network (purple, top panel) was used as a regressor to identify the same, previously unidentified, cell type in zebrafish data by probing the dataset for cells that have calcium responses with a correlation r > 0.6 to the regressor (bottom panel, shading indicates bootstrap SE across 146 zebrafish neurons).

(D′) The newly identified zebrafish cells cluster spatially, especially in a tight rostral band of the cerebellum (arrow). Top panel: dorsal view of the brain (anterior left, left side bottom). Bottom panel: side view of the brain, anterior left, dorsal top). Scale bars, 100 μm.

(E) Connectivity weights between layer 1 neuron types in the temperature branch (columns) feeding into the indicated types of layer 2 neurons (rows). Fish-like types are indicated by corresponding colored bars, and remaining non-fish-like clusters are indicated by thinner gray bars on the right side. Error bars indicate SD.

Shading indicates bootstrap SE across 20 networks in all panels. See also Figures S2 and S3.

et al., 2017). This network variant receives as input a history of angles to a light source and has the task of predicting its angular position after the same swim types used in the thermotaxis network (Figures S3A and S3B). We found that this ANN efficiently learned to fixate a light source (Figures S3C and S3D). However, comparing cell responses between the thermotaxis and phototaxis networks revealed a much simpler stimulus representation in the latter (Figure S3E). This argues that the observed stimulus representations are not emergent features of networks trained to perform navigation but rather depend on the specific task at hand.

White Noise Stimuli Reveal Shared Processing Strategies

Previously, we characterized the computations underlying behavior generation during heat perception in larval zebrafish using white noise temperature stimuli (Haesemeyer et al., 2015). This approach allowed us to derive "filter kernels" that describe how larval zebrafish integrate temperature information to generate swim bouts (inset Figure 3A). These filter kernels revealed that larval zebrafish integrate temperature information for 500 ms to decide on the next swim and that they extract a derivative of the temperature stimulus as reflected in the 0-crossing of the filter (Figure 3A, inset). The filter kernels furthermore indicated that swims were in part controlled by a strong OFF response just before the start of a bout. For comparison, we now presented white noise temperature stimuli to the ANN and computed filter kernels for straight swims and turns (Figure 3A). These bear striking resemblances to the larval zebrafish filter kernels. Namely, even though no explicit constraints on integration timescales were given to the ANN, the filter kernels reveal that most information is integrated over timescales less than a

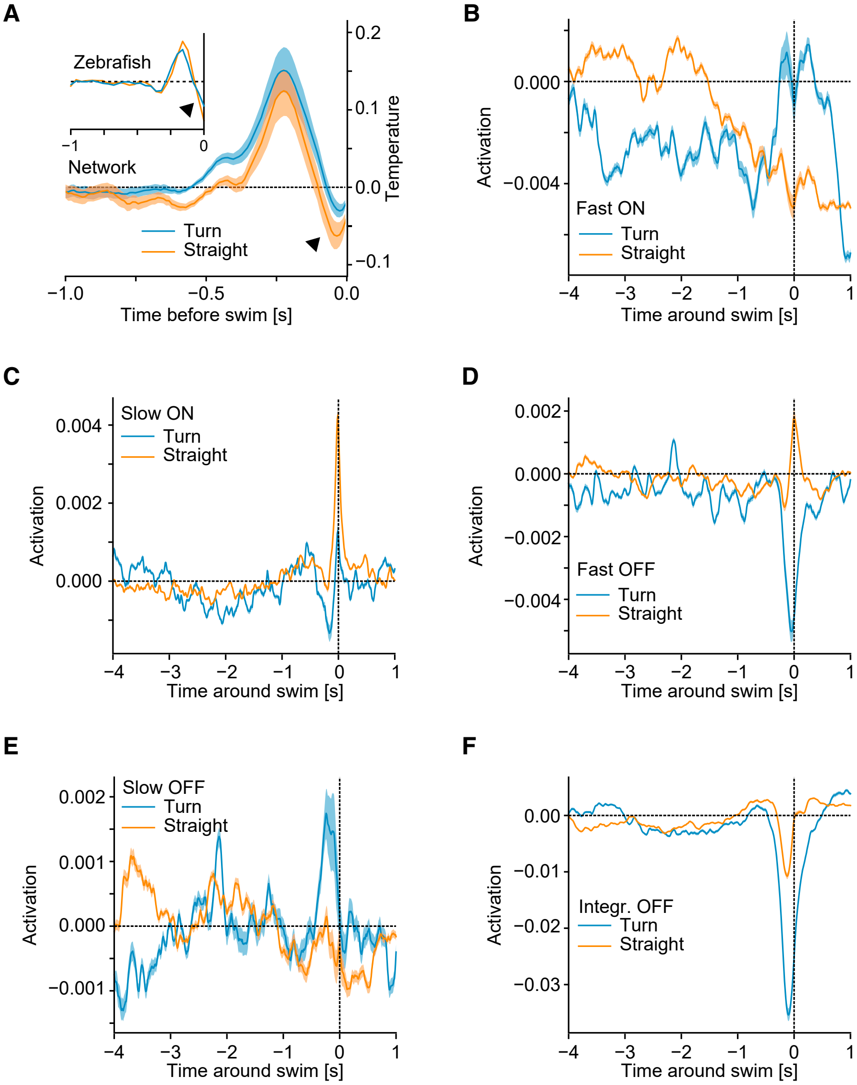


Figure 3. White Noise Analysis Reveals ANN Processing

(A) White noise analysis of behavior induced by the network depicting the average stimulus in the last second preceding a swim. Straight swim kernel is orange, and turn kernel is blue. Inset shows zebrafish kernels for comparison with straight bout kernel in orange and turn kernel in blue. Arrowhead indicates OFF response just before swim start in zebrafish and networks.

(B–F) During the same white noise stimulation paradigm used in (A), the behavior-triggered average activity of the indicated cell types. Orange lines depict behavior-triggered average activity before straight swims; blue lines depict before turns.

(B) Behavior-triggered average activity of “Fast ON” units.

(C) Behavior-triggered average activity of “Slow ON” units.

(D) Behavior-triggered average activity of “Fast OFF” units.

(E) Behavior-triggered average activity of “Slow OFF” units.

(F) Behavior-triggered average activity of “Integrating OFF” units.

Shading indicates bootstrap SE across 20 networks in all panels. See also Figure S4.

second, akin to larval zebrafish integration timescales (Figure S4A). This is likely the result of the ANN adapting processing timescales to the average swim frequency used in the training data, a feature that has previously been suggested in bacterial chemotaxis (Block et al., 1982). Supporting this idea, a reduction of training data baseline swim frequency to 0.5 Hz elongates the filters, while an increase to 2 Hz heightens the filter peaks close to the swim start (Figure S4B). The ANN furthermore computes both a derivative and an integral of temperature, and, notably, behavior is also influenced by a strong OFF response right before the start of a swim (arrowhead in Figure 3A). These are all hallmarks of larval zebrafish temperature processing. Furthermore, as in zebrafish, the OFF response before the swim-start is more strongly modulated for straight swims than turns (Figure 3A), a strategy that likely underlies the favoring of straight swims over turns when temperature approaches cooler, more favorable conditions (Figure 1I). While there are also differences in the filter kernels such as a different modulation in peak height between swims and turns the overall correspondence suggests similar processing strategies. As expected, the swim-triggered averages are completely unstructured in naive networks (Figure S4C).

To gain insight into the contribution of ANN response types to behavior, we analyzed their behavior-triggered average activity during white noise stimulation (Figures 3B–3F). These averages reveal that Slow-ON and Slow-OFF types mostly enhance straight swims or turns respectively without affecting the opposing behavior very much (Figures 3C and 3E). The behavior-triggered activity of Fast-ON cells appears modulated with very slow dynamics (Figure 3B) making the role of this response type harder to interpret. However, as expected from the modulation of turning behavior (Figure 1I), high activity in Fast-ON units seems to strongly suppress straight swims (Figure 3B). Fast-OFF units have the greatest discrimination between straight swims, which are enhanced by activity in this response type and turns that are suppressed (Figure 3D). This is consistent with the observed behavioral modulation based on gradient direction. Integrating OFF cells generally suppress behavior but do so more strongly for turns than straight swims (Figure 3F).

Overall, the behavior-triggered activity suggests contributions by both ON and OFF cells, with especially the rate encoding Fast-OFF cells discriminating turns versus swims. Interestingly, ON and corresponding OFF types don’t affect behaviors in a mirror symmetric fashion but seem to transmit different information. Likewise, as in larval zebrafish, we observed a clear asymmetry between encoding in ON and OFF-type cells such that

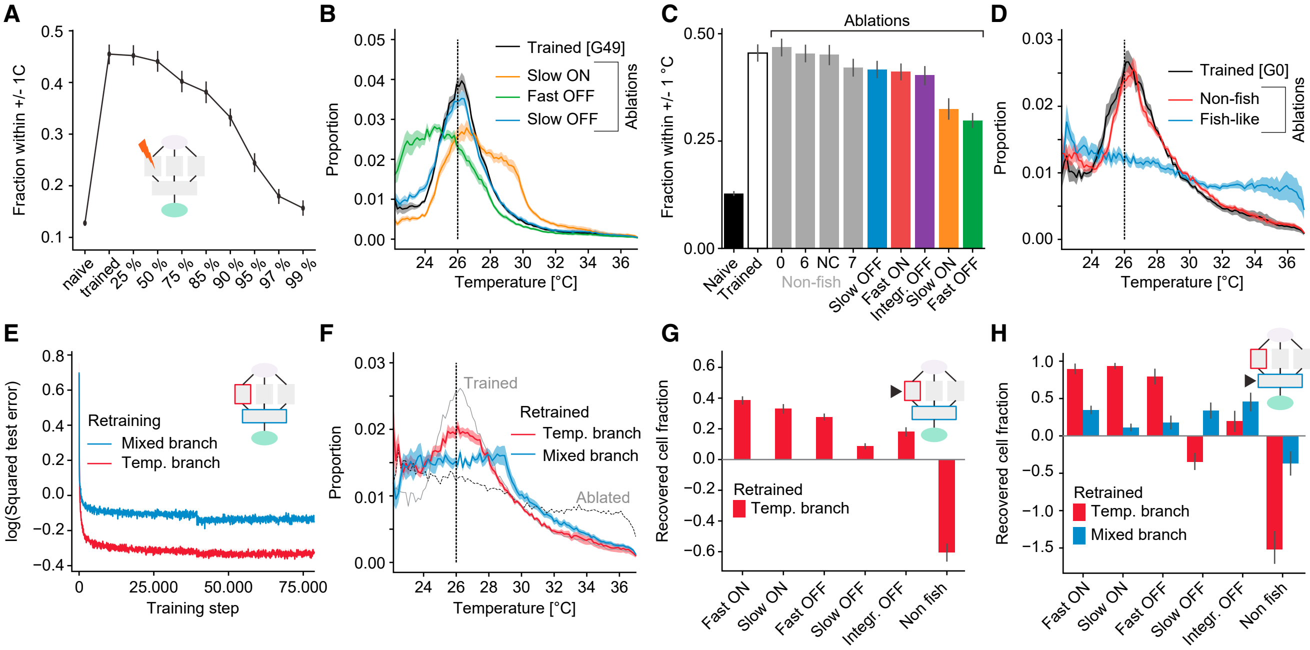


Figure 4. Ablations Reveal Importance of Zebrafish-like Cell Types

(A) Effect of random unit ablations on gradient navigation performance as fraction within 1°C of desired temperature. Shown is performance for naive, fully trained and for random ablations of the indicated fraction of units in the temperature branch for zebrafish networks. Inset depicts location for all ablations.

(B) Occupancy in radial heat gradient for trained zebrafish networks (black) and after ablations of the indicated cell types (colored lines).

(C) Quantification of gradient navigation performance as fraction within 1°C of desired temperature for naive and trained zebrafish networks as well as after ablations of the indicated cell types identified in larval zebrafish (colored bars) and those types not identified in fish ("Non-fish"; gray bars). Ablations are ordered according to severity of phenotype.

(D) Effect on gradient navigation of ablating all types identified in zebrafish (blue line) or all non-fish types (red line). Note that these are non-evolved networks to allow retraining analysis. Trained performance shown in black for reference. The number of ablated units was matched in both conditions (see STAR Methods).

(E) Log of the squared error in temperature predictions of networks on the test dataset after ablating all fish-like types in the temperature branch when retraining weights either in the temperature branch (red line) or in the mixed branch (blue line). Inset indicates retraining locations.

(F) Effect of retraining networks after ablating all zebrafish-like neurons. Retraining was limited to either the temperature branch (red line) or the mixed branch (blue line). Solid gray line visualizes trained and dotted gray line visualizes ablated performance.

(G and H) Recovered fraction of individual cell types after retraining the temperature branch (red bars) or after retraining the mixed branch (blue bars). Insets depict retraining locations.

(G) Cell-type fractions in temperature branch.

(H) Cell-type fractions in mixed branch.

Shading and error bars in all panels indicate bootstrap SE across 20 networks.

OFF cells were not the simple inverse of their ON cell counterparts (Figures 2B and 2C). Since our ANN used rectifying linear units that like biological neurons cannot encode negative firing rates, we wondered whether this constraint caused this asymmetry. We therefore trained an ANN in which we exchanged the activation function for the hyperbolic tangent function resulting in an encoding that is symmetric around 0 (Figure S4D). These networks learned to navigate heat gradients just as well as networks with rectifying linear units (Figures S4E and S4F), but remarkably they represented heat stimuli in a very different manner. Namely, OFF units in this network type were the exact mirror image of ON units (correlation < −0.99 for all pairs), which resulted in an overall simpler representation (Figures S4G and S4H). This notion was supported by the fact that the first 4 principal components explained 99% of the response variance across all cells in hyperbolic tangent networks, while 7 principal components were needed in rectifying linear networks (Figure S4I). This suggests that the biological constraint of only transmitting positive neural responses shapes asymmetries in representations in ON and OFF-type channels and thereby increases required network complexity.

Zebrafish-like Types Form a Critical Core of ANN Function

After discovering clear parallels in representation and computation between ANNs and BNNs, we tested the importance of the common response types for ANN function. As expected for ANNs, random removal of as much as 85% of all units in the temperature processing branch had only a small influence on gradient navigation performance (Figure 4A). However, specific deletion of all Slow ON or Fast OFF-like cells in the network, contrary to Fast ON, Slow OFF, and Integrating OFF deletions, had a strong effect on temperature navigation (Figures 4B and 4C). Indeed, the Slow ON and Fast OFF types also have the highest predictive power on heat-induced behaviors in the larval zebrafish hindbrain (Haesemeyer et al., 2018). Overall, deletion of any zebrafish-like type in the network had a larger effect on navigation performance than deleting individual types not found in larval zebrafish (Figure 4C) indicating a relatively higher importance of fish-like types. Strikingly, deleting all fish-like types in the temperature branch of the ANN nearly abolished gradient navigation, while performance was hardly affected when deleting all non-fish types (Figure 4D). This demonstrates that fish-like response types are of critical importance for gradient navigation.

To test whether the network could adjust to the absence of fish-like representations, we locally retrained the heat-navigation ANN, restricting updates to either the temperature branch of the network or the mixed branch that integrates temperature and movement information. Retraining improved network performance in both cases, but retraining the temperature branch led to considerably better performance compared with retraining the mixed branch (Figures 4E and 4F). This indicates that, while the temperature branch still transmits some usable information after ablation of all fish-like types, the resulting representation is lacking information required for efficient navigation. To gain better insight into the consequences of retraining, we analyzed the distribution of response types in the temperature and mixed branch in retrained networks. When retraining the temperature branch, fish-like types emerged at the expense of non-fish types giving further credence to their importance for temperature prediction and navigation (Figures 4G and 4H). Retraining the mixed branch, however, failed to re-generate most of the fish-like types indicating that these cannot be re-synthesized from information carried by non-fish types (Figure 4H). The only exceptions to this were Slow-OFF and Integrating-OFF cells, which are the two cell types that receive fairly strong inputs from non-fish-like types to begin with (Figure 2E).

Changes in Motor Repertoire Tune Sensory Representations

To test the influence of the behavioral repertoire on sensory representations, we created a network variant using behaviors displayed during *C. elegans* heat gradient navigation (Ryu and Samuel, 2002). To limit changes to the motor repertoire, this network had the same structure and task as the original network (Figures 5A and 5B) but was trained on a random walk through a heat gradient employing *C. elegans* behaviors. While this design choice minimized architectural changes, we note that the resulting network failed to approximate the vastly smaller number of neurons found in *C. elegans* compared to larval zebrafish.

Just like the zebrafish heat navigation ANN, the *C. elegans* ANN learned to predict temperatures (Figure 5C) and hence was able to effectively navigate a heat gradient (Figure 5D). Navigation performance was in fact better than for the zebrafish ANN (cf. Figures 1H and 5D), which likely is a consequence of the ability of trajectory reversals by executing pirouettes. We did not add an evolutionary algorithm to train changes in crawl frequency or speed since such behavioral modulation by temperature is not observed in the worm (Ryu and Samuel, 2002). Comparing temperature responses between the zebrafish and *C. elegans* ANNs using principal-component analysis revealed overlapping as well as divergent responses (Figure 5E). Namely, some types show near identical responses, while other response types are exclusive to one of the two ANNs (Figures S5C–S5I). Overall, there is a large overlap in response dynamics between the zebrafish and *C. elegans* network in spite of vastly different behavioral timescales. This might indicate that adaptations of processing to behavioral timescales as observed by white noise analysis of the zebrafish network (Figure S4B) can be accomplished by small changes in response dynamics of individual units. Importantly, we could identify response types that mimic responses of cells previously described in *C. elegans* (Figures S5A and S5B). This included a strongly adapting cell type that was most sensitive to changes in temperature similar to the *C. elegans* AFD neuron (Figure 5F; Clark et al., 2006, 2007; Kimura et al., 2004). Another cell type largely reported absolute temperature levels as has been suggested for the AWC and AIY neurons (Figure 5F; Kuhara et al., 2008). We do note, however, that correspondences between *C. elegans* neurons and reported ANN responses are weaker than for zebrafish; the AFD sensory neuron preferentially encodes rates of temperature change but displays a larger response to changes in temperature levels than our network unit, while AWD has slower response kinetics than the corresponding network unit. Intriguingly, while the *C. elegans* ANN was as robust to random unit deletions as the zebrafish ANN (Figure S5K), it was considerably more sensitive to single-cell-type ablations. Removal of AFD-like units severely reduced gradient navigation performance and especially affected cryophilic bias (Figure 5G), as reported for *C. elegans* itself (Chung et al., 2006). This is particularly interesting since ablating the closest matching cell type in the zebrafish network (Fast ON, Figure S5D) only has a very small effect on navigation performance (Figure 4C). This indicates a shift in unit requirement concomitant to the change in available motor repertoire. A weaker phenotype was observed when ablating AWC/AIY-like neurons (Figures 5G and 5H) whose role in *C. elegans* thermotaxis is less well established (Garrity et al., 2010). The overall stronger dependence of network performance on individual cell types suggests a less distributed representation in the *C. elegans* ANN compared to the zebrafish ANN, which was also mirrored in sparser inter-type connectivity (Figure S5I). This may well be reflected in the animals themselves, and in fact a recent paper applied a control theory paradigm to suggest links between *C. elegans* body bends and single motor neurons (Yan et al., 2017).

DISCUSSION

ANNs and BNNs are very successful in solving problems of various complexities, but how these networks accomplish such tasks is still largely unclear. Uncovering the fundamental principles that govern these operations is particularly daunting in the case of BNNs because experimental access is limited, and the underlying implementation is not necessarily aligned with human intuition. The principles underlying the operation of ANNs on the other hand are likely easier to dissect because they are made by man and because activity states in such networks can be readily queried. However, it is largely unclear to what extent the responses of units in the models correspond to neuronal responses observed in biological brains; in other

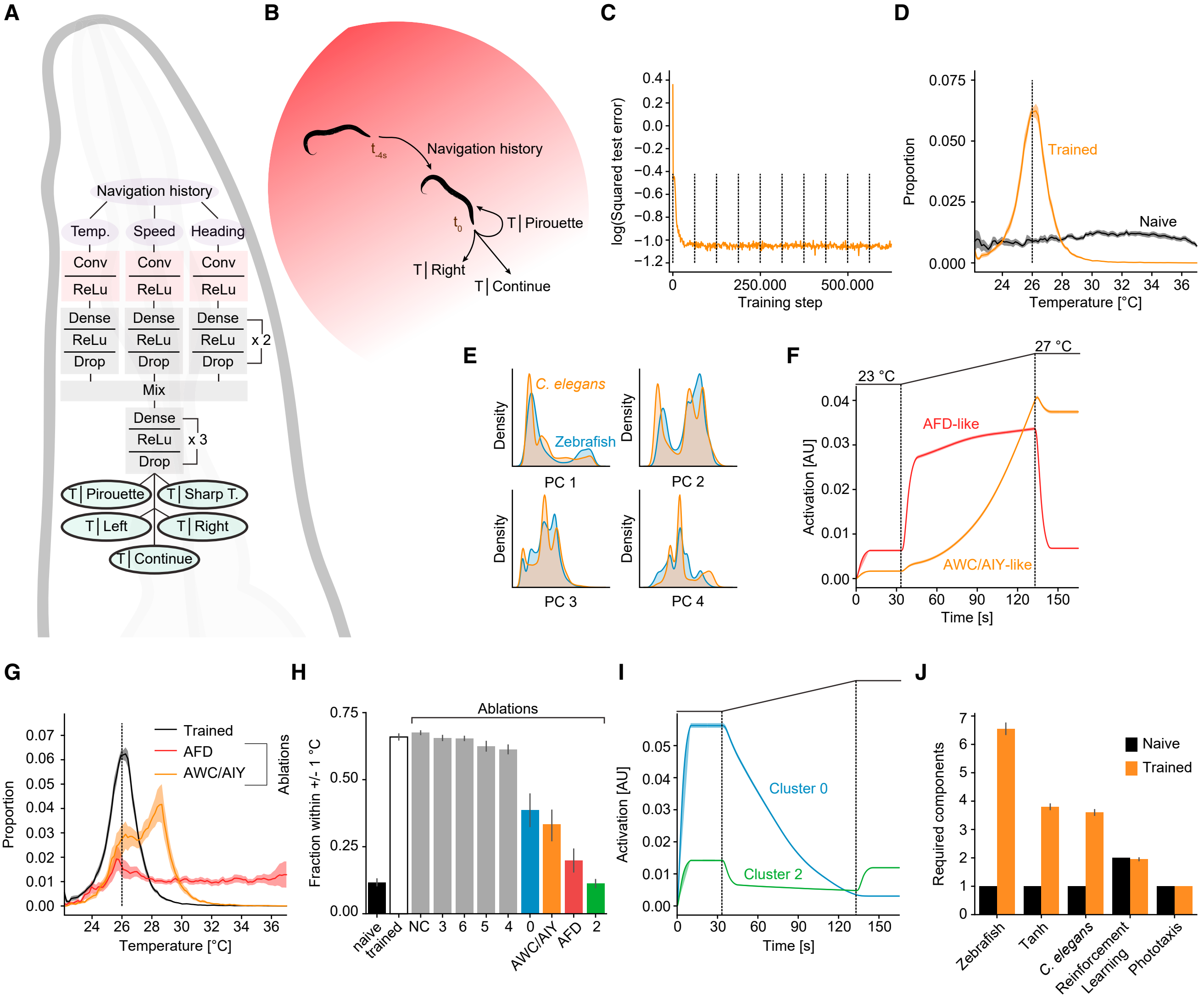


Figure 5. A Network for *C. elegans* Thermotaxis

(A) Architecture of the *C. elegans* convolutional network. Note that the architecture is the same as in Figure 1D except for the predictive output, which is matched to the behavioral repertoire of *C. elegans*.

(B) Schematic of the task of the *C. elegans* ANN: the network uses a 4 s history of experienced temperature and generated behaviors to predict the resting temperature after a *C. elegans* behavior.

(C) Log-squared error of temperature predictions on test dataset during training.

(D) Occupancy in a radial heat gradient of naive (black) and trained (orange) *C. elegans* networks. Dashed vertical line at 26°C indicates desired temperature.

(E) Comparison of all unit responses in the temperature branch of the zebrafish and *C. elegans* heat gradient ANN in principal-component analysis (PCA) space when presenting the same time varying stimulus used in Figure 2B to both networks. The first four principal components capture >95% of the variance. Plots show occupational density along each PC for the zebrafish network (blue) and the *C. elegans* network (orange).

(F) Responses of two *C. elegans*-like cell types when presenting a temperature ramp depicted in black on top. The red type shows adapting responses like the AFD neuron (cf. Clark et al., 2007; Kotera et al., 2016), while the orange type reports temperature level as suggested for the AWC/AIY neurons (cf. Kuhara et al., 2008).

(G) Occupancy in radial heat gradient for trained *C. elegans* networks (black) and after ablations of the indicated cell types (colored lines).

(H) Quantification of gradient navigation performance as fraction within 1°C of desired temperature for naive and trained *C. elegans* networks as well as after ablations of the indicated cell types. Ablations are ordered by severity of phenotype.

(I) Responses of two *C. elegans* cell types with strong gradient navigation phenotypes in (H) to the same temperature ramp presented in (F).

(J) For each network type the number of principal components needed to explain at least 99% of the total network unit variance when the stimulus depicted in Figure 2B is presented to the network. Naive networks, black; fully trained, orange. Note, that naive reinforcement learning networks already require 2 components since these networks have fewer units overall and therefore have a noisier representation in the naive state.

Shading and error bars in all panels indicate bootstrap SE across 20 networks. See also Figure S5.

words, to what extent biological brains and ANNs converge on similar solutions of algorithmic implementation.

Recent work compared image classification ANNs and visual processing in the primate ventral visual stream to show that ANNs can indeed share representations and processing strategies with BNNs (Khaligh-Razavi and Kriegeskorte, 2014; Yamins and DiCarlo, 2016; Yamins et al., 2014). Furthermore, networks that have been trained to learn a place-cell-like encoding give rise to units mimicking entorhinal cell types such as grid cells (Banino et al., 2018; Cueva and Wei, 2018). This apparent convergence in processing strategies suggests that ANNs can give insight into BNN function. This strategy helped gain additional insights into retinal processing by constraining an ANN model to predict retinal ganglion cell responses to natural scenes (Maheswaranathan et al., 2018; McIntosh et al., 2016) and to study the role of interneurons in leech withdrawal reflexes (Lockery et al., 1989). However, it is unclear how generalizable such results are across stimulus modalities or if they generalize to behavioral task constraints.

Here, we extended these approaches by asking whether constraining ANNs by the behavioral task of heat gradient navigation using a species-specific behavioral repertoire would lead to sensory representations observed in a larval zebrafish BNN. In other words, to what extent these ANNs arrive at a solution through training that is similar to the solution that evolved in zebrafish to perform efficient navigation of heat gradients. We followed two complementary modeling strategies: one ANN implements a predictive forward model of the world and was trained to predict the consequences of a behavioral action within a heat gradient; the other network type was trained via reinforcement learning to implement a successful behavioral policy. Importantly, both of these strategies are biologically plausible, but it is currently unclear whether larval zebrafish encode an explicit prediction of behavioral consequences as assumed by the predictive ANN or if such a prediction is rather performed implicitly as in the reinforcement learning model.

Comparing behavioral navigation strategies revealed clear similarities between the predictive ANN and zebrafish, while the reinforcement learning model relies on a different strategy. This suggests that on a behavioral level the predictive model is closer to larval zebrafish. However, since strategies employed by reinforcement learning models are dependent on the actual reward strategy, this does not rule out that a more comparable reinforcement learning model could be found. Since thermal gradient navigation is an innate larval zebrafish behavior the poorer correspondence of the reinforcement learning model could on the other hand also suggest that such training approaches are less well suited to model innate circuits.

Concomitant with learning to navigate heat gradients, stimulus representations in the ANNs changed as well. In naive networks, one principal component can explain >99% of the variance in stimulus responses (Figure 5J). During training, representational complexity increased in all predictive ANN types as evidenced by an increase in the number of principal components necessary to explain response variance (Figure 5J). And post-training, on a neural level we could show that processing and representation in the predictive thermotaxis ANN bears striking similarities to BNN representations in zebrafish. This is apparent by a clear parallel in stimulus representation between zebrafish hindbrain neurons and the predictive ANN, including splitting of information into ON and OFF channels as well as cell types representing the derivative of the stimulus across time. Such representations are also prominent in other modalities such as the visual system. Here, cells compute the spatial derivative of the stimulus, and image classification ANNs represent this information in their initial layers as well (Yamins and DiCarlo, 2016). This indicates a reuse of prominent computational motifs across sensory modalities.

Interestingly, while the reinforcement learning network learns a similar representation in spite of a very different modeling strategy, the Fast OFF cell type that reports rates of temperature decrease is conspicuously absent from this network (cf. Figures 2A with S2E). Given that this cell type most strongly discriminates turns and straight swims in the predictive ANN (Figure 3D) and that both zebrafish and the predictive ANN modulate turning based on gradient direction (Figure 1I), it is interesting to speculate that this cell type is absent because of the different behavioral strategy employed by the reinforcement learning network.

Interestingly, sensory representations get tuned when changing the available motor repertoire from that of zebrafish to that of *C. elegans*. This is apparent for the role of the response type most closely matching the *C. elegans* AFD neuron. This response type is highly similar to the zebrafish Fast ON type (Figure S5D). But while ablations of the Fast ON type have very little effect on zebrafish ANN navigation performance (Figure 4C), it is absolutely critical for *C. elegans* ANN function (Figures 5G and 5H) mirroring its importance for navigation behavior in the worm (Chung et al., 2006). Taken together these results strongly argue that stimulus representations in BNNs and ANNs converged on a stereotypical solution throughout evolution and training respectively and that stimulus representations are likely constrained by behavioral goals and motor repertoires (Rosenblatt, 1962).

Our architectural design choice for the ANN restricted temporal convolutions to the input layer of the ANN, which is different from the zebrafish BNN. These differences are reflected in differences of inter-type connectivity as well, resulting in connections that are different from connectivity suggested by a zebrafish circuit model. Also, in *C. elegans* AFD is connected to AIY, but this connection is absent in our model. The fact that we still observed similarities in representations suggests that while connectivity can constrain representations similar representations can arise in spite of differing connectivity. A similar observation has recently been made in a *Drosophila* visual processing model in which connectivity constrains *where* representations arise within the network but not *if* they arise (Tschopp et al., 2018).

Constraining an ANN by a heat gradient navigation task specifically allowed us to form testable predictions about the larval zebrafish and *C. elegans* BNN. This led to the identification of a novel heat response type, ‘‘Integrating OFF’’ cells, in the larval zebrafish cerebellum. Although ablation of this response type in the model suggests that its role is minor, one can speculate that the integrating properties here might add stability to the navigation behavior, very much like integrating components in technical control systems. In our case, the role of the Integrating OFF type in providing stability to the system may well be reflected in its anatomical location within the cerebellum since it is thought that this structure likely tunes and stabilizes motor output rather than being involved in the generation of the motor commands themselves.

At the same time, the differential effects of deleting fish-like types on navigation performance allows for the generation of testable hypotheses about the relative importance of individual cell types in the zebrafish BNN especially since the two most important response types in the ANN (Slow ON and Fast OFF) are also most strongly implicated in temperature processing in larval zebrafish (Haesemeyer et al., 2018). Virtual ablations also make strong predictions about the role of two OFF cell types in thermal navigation of *C. elegans* (Figure 5I). OFF type cells have so far not been implicated in *C. elegans* thermal navigation, but recent whole-brain imaging data in response to thermal stimuli suggest that thermosensitive OFF types do exist in *C. elegans* (Kotera et al., 2016).

In summary, the strong parallel between ANNs and BNNs implies that artificial networks with their greater amenability to analysis and manipulation over BNNs can serve as powerful tools to derive neural principles underlying cognition in biological brains.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- LEAD CONTACT AND MATERIALS AVAILABILITY
- EXPERIMENTAL MODEL AND SUBJECT DETAILS
- METHOD DETAILS
    - ○ Software framework
    - ○ Behavior generation
    - ○ Artificial neural networks
    - ○ Data analysis
    - ○ Behavior analysis
    - ○ White noise analysis
    - ○ Unit response clustering
    - ○ Connectivity
    - ○ Ablations and retraining of neural networks
    - ○ Comparison of representations by PCA
    - ○ Matching of ANN and zebrafish heat response types
    - ○ Identification and mapping of novel zebrafish response types
    - ○ Linear model based mapping of zebrafish temperature response types
- QUANTIFICATION AND STATISTICAL ANALYSIS
- DATA AND CODE AVAILABILITY

SUPPLEMENTAL INFORMATION

Supplemental Information can be found online at https://doi.org/10.1016/j.neuron.2019.07.003.

ACKNOWLEDGMENTS

M.H. was supported by an EMBO Long Term Postdoctoral fellowship (ALTF 1056-10) and a fellowship by the Jane Coffin Childs Fund for Biomedical Research (61-1468). Research was funded by NIH grants 1U19NS104653 and 5R24NS086601 as well as a Simons Collaboration on the Global Brain Research Award (542973) to F.E. and NIH grant DP1HD094764 to A.F.S. We thank Hanna Zwaka for the *C. elegans* drawing in Figure 5A. We thank Armin Bahl, Andrew Bolton, James Fitzgerald, and Aravi Samuel for helpful discussions and critical comments on the manuscript.

AUTHOR CONTRIBUTIONS

M.H. conceived the project in discussion with F.E. and A.F.S. and carried out all experiments, programming, and data analysis. M.H., A.F.S., and F.E. interpreted the data and wrote the manuscript.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: August 29, 2018
Revised: May 6, 2019
Accepted: July 1, 2019
Published: July 31, 2019

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: a system for large-scale machine learning. In Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, K. Keeton and T. Roscoe, eds. (USENIX Association), pp. 265–283.

Ahrens, M.B., Li, J.M., Orger, M.B., Robson, D.N., Schier, A.F., Engert, F., and Portugues, R. (2012). Brain-wide neuronal dynamics during motor adaptation in zebrafish. Nature *485*, 471–477.

Banino, A., Barry, C., Uria, B., Blundell, C., Lillicrap, T., Mirowski, P., Pritzel, A., Chadwick, M.J., Degris, T., Modayil, J., et al. (2018). Vector-based navigation using grid-like representations in artificial agents. Nature *557*, 429–433.

Block, S.M., Segall, J.E., and Berg, H.C. (1982). Impulse responses in bacterial chemotaxis. Cell *31*, 215–226.

Chen, X., and Engert, F. (2014). Navigational strategies underlying phototaxis in larval zebrafish. Front. Syst. Neurosci. *8*, 39.

Chung, S.H., Clark, D.A., Gabel, C.V., Mazur, E., and Samuel, A.D.T. (2006). The role of the AFD neuron in C. elegans thermotaxis analyzed using femtosecond laser ablation. BMC Neurosci. *7*, 30.

Clark, D.A., Biron, D., Sengupta, P., and Samuel, A.D.T. (2006). The AFD sensory neurons encode multiple functions underlying thermotactic behavior in Caenorhabditis elegans. J. Neurosci. *26*, 7444–7451.

Clark, D.A., Gabel, C.V., Gabel, H., and Samuel, A.D.T. (2007). Temporal activity patterns in thermosensory neurons of freely moving Caenorhabditis elegans encode spatial thermal gradients. J. Neurosci. *27*, 6083–6090.

Cohen, J.D., Dunbar, K., and McClelland, J.L. (1990). On the control of automatic processes: a parallel distributed processing account of the Stroop effect. Psychol. Rev. *97*, 332–361.

Cueva, C.J., and Wei, X.-X. (2018). Emergence of grid-like representations by training recurrent neural networks to perform spatial localization. arXiv, arXiv:1803.07770.

Dunn, T.W., Mu, Y., Narayan, S., Randlett, O., Naumann, E.A., Yang, C.-T., Schier, A.F., Freeman, J., Engert, F., and Ahrens, M.B. (2016). Brain-wide mapping of neural activity controlling zebrafish exploratory locomotion. eLife *5*, e12741.

Engert, F. (2014). The big data problem: turning maps into knowledge. Neuron *83*, 1246–1248.

Garrity, P.A., Goodman, M.B., Samuel, A.D., and Sengupta, P. (2010). Running hot and cold: behavioral strategies, neural circuits, and the molecular machinery for thermotaxis in C. elegans and Drosophila. Genes Dev. *24*, 2365–2382.

Gau, P., Poon, J., Ufret-Vincenty, C., Snelson, C.D., Gordon, S.E., Raible, D.W., and Dhaka, A. (2013). The zebrafish ortholog of TRPV1 is required for heat-induced locomotion. J. Neurosci. *33*, 5249–5260.

Glorot, X., and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Y. Whye The and M. Titterington, eds. (PMLR), pp. 249–256.

Haesemeyer, M., Robson, D.N., Li, J.M., Schier, A.F., and Engert, F. (2015). The structure and timescales of heat perception in larval zebrafish. Cell Syst. *1*, 338–348.

Haesemeyer, M., Robson, D.N., Li, J.M., Schier, A.F., and Engert, F. (2018). A brain-wide circuit model of heat-evoked swimming behavior in larval zebrafish. Neuron *98*, 817–831.

Hassabis, D., Kumaran, D., Summerfield, C., and Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. Neuron *95*, 245–258.

Huang, K.-H., Ahrens, M.B., Dunn, T.W., and Engert, F. (2013). Spinal projection neurons control turning behaviors in zebrafish. Curr. Biol. *23*, 1566–1573.

Khaligh-Razavi, S.-M., and Kriegeskorte, N. (2014). Deep supervised, but not unsupervised, models may explain IT cortical representation. PLoS Comput. Biol. *10*, e1003915.

Kimura, K.D., Miyawaki, A., Matsumoto, K., and Mori, I. (2004). The C. elegans thermosensory neuron AFD responds to warming. Curr. Biol. *14*, 1291–1295.

Kingma, D.P., and Ba, J. (2014). Adam: a method for stochastic optimization. arXiv, arXiv:1412.6980.

Kotera, I., Tran, N.A., Fu, D., Kim, J.H., Byrne Rodgers, J., and Ryu, W.S. (2016). Pan-neuronal screening in *Caenorhabditis elegans* reveals asymmetric dynamics of AWC neurons is critical for thermal avoidance behavior. eLife. Published online November 16, 2016. https://doi.org/10.7554/eLife.19021.

Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems 25, F. Pereira, C.J.C. Burges, L. Bottou, and K.Q. Weinberger, eds. (Curran Associates), pp. 1097–1105.

Kuhara, A., Okumura, M., Kimata, T., Tanizawa, Y., Takano, R., Kimura, K.D., Inada, H., Matsumoto, K., and Mori, I. (2008). Temperature sensing by an olfactory neuron in a circuit controlling behavior of C. elegans. Science *320*, 803–807.

Lake, B.M., Ullman, T.D., Tenenbaum, J.B., and Gershman, S.J. (2017). Building machines that learn and think like people. Behav. Brain Sci. *40*, e253.

Lockery, S.R., Wittenberg, G., Kristan, W.B., Jr., and Cottrell, G.W. (1989). Function of identified interneurons in the leech elucidated using neural networks trained by back-propagation. Nature *340*, 468–471.

Logothetis, N.K., and Sheinberg, D.L. (1996). Visual object recognition. Annu. Rev. Neurosci. *19*, 577–621.

Maheswaranathan, N., McIntosh, L.T., and Kastner, D.B. (2018). Deep learning models reveal internal structure and diverse computations in the retina under natural scenes. bioRxiv. https://doi.org/10.1101/340943.

McClelland, J.L., and Rumelhart, D.E. (1981). An interactive activation model of context effects in letter perception: I. An account of basic findings. Psychol. Rev. *88*, 375.

McIntosh, L.T., Maheswaranathan, N., Nayebi, A., Ganguli, S., and Baccus, S.A. (2016). Deep learning models of the retinal response to natural scenes. Adv. Neural Inf. Process. Syst. *29*, 1369–1377.

Mehta, B., and Schaal, S. (2002). Forward models in visuomotor control. J. Neurophysiol. *88*, 942–953.

Miri, A., Daie, K., Burdine, R.D., Aksay, E., and Tank, D.W. (2011). Regression-based identification of behavior-encoding neurons during large-scale optical imaging of neural activity at cellular resolution. J. Neurophysiol. *105*, 964–980.

Mischiati, M., Lin, H.-T., Herold, P., Imler, E., Olberg, R., and Leonardo, A. (2015). Internal models direct dragonfly interception steering. Nature *517*, 333–338.

Moser, E.I., Kropff, E., and Moser, M.-B. (2008). Place cells, grid cells, and the brain's spatial representation system. Annu. Rev. Neurosci. *31*, 69–89.

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

Portugues, R., and Engert, F. (2011). Adaptive locomotor behavior in larval zebrafish. Front. Syst. Neurosci. *5*, 72.

Prober, D.A., Zimmerman, S., Myers, B.R., McDermott, B.M., Jr., Kim, S.-H., Caron, S., Rihel, J., Solnica-Krezel, L., Julius, D., Hudspeth, A.J., and Schier, A.F. (2008). Zebrafish TRPA1 channels are required for chemosensation but not for thermosensation or mechanosensory hair cell function. J. Neurosci. *28*, 10102–10110.

Rosenblatt, F. (1962). Principles of Neurodynamics (Spartan).

Rumelhart, D.E., and McClelland, J.L. (1982). An interactive activation model of context effects in letter perception: Part 2. The contextual enhancement effect and some tests and extensions of the model. Psychol. Rev. *89*, 60–94.

Rumelhart, D.E., and McClelland, J.L.; Pdp Research Group (1987). Parallel Distributed Processing (MIT Press).

Ryu, W.S., and Samuel, A.D.T. (2002). Thermotaxis in Caenorhabditis elegans analyzed by measuring responses to defined thermal stimuli. J. Neurosci. *22*, 5727–5733.

Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of Go with deep neural networks and tree search. Nature *529*, 484–489.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. *15*, 1929–1958.

Trullier, O., Wiener, S.I., Berthoz, A., and Meyer, J.A. (1997). Biologically based artificial navigation systems: review and prospects. Prog. Neurobiol. *51*, 483–544.

Tschopp, F.D., Reiser, M.B., and Turaga, S.C. (2018). A connectome based hexagonal lattice convolutional network model of the Drosophila visual system. arXiv, arXiv:1806.04793.

Wolf, S., Dubreuil, A.M., Bertoni, T., Böhm, U.L., Bormuth, V., Candelier, R., Karpenko, S., Hildebrand, D.G.C., Bianco, I.H., Monasson, R., and Debrégeas, G. (2017). Sensorimotor computation underlying phototaxis in zebrafish. Nat. Commun. *8*, 651.

Wolpert, D.M., and Miall, R.C. (1996). Forward models for physiological motor control. Neural Netw. *9*, 1265–1279.

Yamins, D.L.K., and DiCarlo, J.J. (2016). Using goal-driven deep learning models to understand sensory cortex. Nat. Neurosci. *19*, 356–365.

Yamins, D.L.K., Hong, H., Cadieu, C.F., Solomon, E.A., Seibert, D., and DiCarlo, J.J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. Proc. Natl. Acad. Sci. USA *111*, 8619–8624.

Yan, G., Vértes, P.E., Towlson, E.K., Chew, Y.L., Walker, D.S., Schafer, W.R., and Barabási, A.-L. (2017). Network control principles predict neuron function in the Caenorhabditis elegans connectome. Nature *550*, 519–523.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited Data		
Model training data and training checkpoints	This paper	https://doi.org/10.5281/zenodo.3258831
Software and Algorithms		
Training, modeling and analysis framework – Python, Tensorflow	This paper	https://github.com/haesemeyer/GradientPrediction

LEAD CONTACT AND MATERIALS AVAILABILITY

Further information and requests for resources and reagents should be directed to and will be fulfilled by the Lead Contact, Florian Engert (florian@mcb.harvard.edu).

EXPERIMENTAL MODEL AND SUBJECT DETAILS

No animal experiments were performed for this study. Experimental data used is from Haesemeyer et al. (2015, 2018).

METHOD DETAILS

All analysis in this study was performed in an automated manner. All sample sizes were fixed before the start of analysis. No architectural adjustments on the networks were performed. The structure of the zebrafish predictive heat gradient network as presented was simply the first tried architecture that was trainable and through the branched structure had the property of containing units that are exclusive to one input modality. The other networks were then created such that they use the exact same structure to keep architectural changes to a minimum, essentially so that only changes to the outputs were necessary.

We note that we observed that the complexity of the five layer networks presented in the manuscript is not necessary and in fact two-layer architectures suffice and arrive at the same representation (data not shown).

Software framework

All neural networks were implemented in Tensorflow (Abadi et al., 2016) using Python 3.6. All data analysis was performed in Python 3.6 using numpy, scipy, and scikit learn (Pedregosa et al., 2011) as well as matplotlib and seaborn for plotting.

Behavior generation

All simulations were run at an update frequency of 100 Hz. Since zebrafish swim in discrete bouts, network predictions were only evaluated whenever swim bouts were instantiated. This occurred at a baseline frequency of 1 Hz (i.e., with a probability of 0.01 given the update frequency). Even though *C. elegans* moves continuously, behavioral modules are only selected with slow dynamics (Ryu and Samuel, 2002). Hence to reduce computational load, models and behavior selections were only evaluated with a frequency of 0.1 Hz (i.e., with a probability of 10^{-4} given the update frequency).

Zebrafish

Zebrafish behavioral parameters were based on swim bouts enacted by freely swimming larval zebrafish during heat gradient navigation (Haesemeyer et al., 2015). When the selected behavior was “stay” the virtual fish stayed in place for one update cycle. For all other choices, displacements were drawn at random in mm from a gamma distribution according to:

$$d \sim \Gamma(2.63,\ 7.25)/9$$

Turn angles (heading changes) for the three swim types were drawn at random in degrees from normal distributions according to:

$$\Delta a_{Straight} \sim \mathcal{N}(0,\ 2)$$

$$\Delta a_{Left} \sim \mathcal{N}(-30,\ 5)$$

$$\Delta a_{Right} \sim \mathcal{N}(30,\ 5)$$

Each behavior was implemented such that each swim lasted 200 ms (20 timesteps), the average length of zebrafish swim bouts. The heading change was implemented within the first frame while the displacement was evenly divided over all frames. This is a simplification over true zebrafish behavior, where heading changes precede displacements as well but where both occur with distinct acceleration and deceleration phases.

The goal of these choices was to approximate larval zebrafish behavior rather than faithfully capture all different swim types.

C. elegans

C. elegans behavioral parameters were based on freely crawling worms navigating temperature gradients (Ryu and Samuel, 2002). When the selected behavior was "continue" or while no behavior was selected, the virtual worm was crawling on a straight line with heading jitter. The per-timestep displacement was drawn in mm according to:

$$d \sim \mathcal{N}(16,\ 5.3)/(60*100)$$

The per timestep heading jitter (random walk in heading direction space) was drawn in degrees according to:

$$\Delta a_{jitter} \sim \mathcal{N}(0,\ 0.1)/100$$

The other behaviors, pirouettes, sharp turns and shallow left or right turns were implemented as heading angle changes together with displacement drawn from the distribution above enacted over a total time of 1 s. The heading angle changes were drawn in degrees at random as follows:

$$\Delta a_{Pirouette} \sim \mathcal{N}(180,\ 10)$$

$$\Delta a_{Sharp\ turn} \sim \mathcal{N}(\pm 45,\ 5)$$

$$\Delta a_{Left} \sim \mathcal{N}(-10,\ 1)$$

$$\Delta a_{Right} \sim \mathcal{N}(10,\ 1)$$

Again, the goal of these choices was to approximate *C. elegans* movement statistics during heat gradient navigation rather than faithfully recapitulating the full behavioral repertoire.

Artificial neural networks

Architecture

All networks were designed with the same basic architecture. Initial branches separately perform convolution on and then process the three inputs, temperature history, speed history and delta-heading history through two hidden layers. The outputs of these separate branches were subsequently concatenated and processed within another set of three hidden layers before being transformed to the desired output (prediction in case of predictive networks, action probability in the case of the reinforcement learning network). This general architecture was both biologically motivated and at the same time had the advantage that single-modality units could be analyzed within the initial branches.

Network inputs consisted of a 2D matrix, with 4 s of temperature, speed, and delta-heading history at a simulation rate of 100 Hz. The first network operation was a mean-pooling, binning the inputs to a model rate of 5 Hz, the same rate at which larval zebrafish imaging data during heat stimulation was previously analyzed. After pooling, the inputs were split into the temperature, speed, and delta-heading components and each component was fed into its corresponding processing branch.

For predictive networks input branches were designed with 40 linear rectifying convolutional layers each, that each learned one 1D filter over the whole 4 s of history (20 weights). Convolution was performed such that only one output datapoint per filter (the dot-product of the filter with the input) was obtained. Hidden layers within the network had 512 rectifying linear units (or 256 or 1024 for the initial tests in Figure 1D). The output layer of the predictive networks consisted either of 4 linear units (for zebrafish networks) or 5 linear units (for *C. elegans* networks). The purpose of the output layer was to compute the temperature (or angle to a light source) 500 ms after enacting the chosen swim type in the case of zebrafish networks or 1 min of straight continuous movement after enacting the chosen behavior in the case of *C. elegans* networks.

The zebrafish reinforcement learning network had the same architecture. However, since training involved the generation of >10^7 swims within a virtual gradient, the network complexity was reduced to reduce training time. This meant that only 20 convolutional layers were use per input branch and each hidden layer only consisted of 128 units. With these adjustments, training of a reinforcement learning model took about the same time as training the predictive model. The output layer of the reinforcement learning network consisted of a softmax layer with 4 units, computing the desired probability of the 4 zebrafish behaviors, p('stay'), p('swim straight'), p('turn left') or p('turn right').

Training data generation and supervised network training for predictive networks

For both zebrafish and *C. elegans*, training data were generated by randomly choosing behavioral modules according to the statistics given above without any influence of temperature on behavioral choices. For training data generation the virtual animals explored two types of circular arenas with a radius of 100 mm each: In one arena, temperature increased linearly from 22°C at the center to 37°C at the periphery, while in the other arena the gradient was reversed. Training datasets were generated from the random walks through these arenas by simulating forward from each timestep for each possible behavioral choice. This way the true temperatures resulting from each behavioral choice were obtained together with the experienced temperature and behavioral history. For the zebrafish phototaxis ANN, the same strategy was employed but instead of recording temperature history and prediction, the angle to a light-source in the center of one arena with a radius of 100 mm was calculated. For each network type a test-dataset was generated in the same manner to be able to evaluate prediction performance.

Networks were trained using stochastic gradient descent on mini-batches consisting of 32 random samples each. Notably, training was not successful when randomly mixing training data from both arena types. Every training epoch was therefore split into two halves during each of which only batches from one arena type training dataset were presented. We used an Adam optimizer [learning rate = 10^{-4}, $\beta 1 = 0.9$, $\beta 2 = 0.999$, $\varepsilon = 10^{-8}$] during training (Kingma and Ba, 2014), optimizing the squared loss between the network predicted temperature and the true temperature in the training dataset. Test batches were larger consisting of 128 samples each, drawn at random from both arena types.

Network weights were initialized such that gradient scales were kept constant across layers according to Glorot and Bengio (2010) by being drawn from a uniform distribution on the interval:

$$\left[-\sqrt{6/(N_{in} + N_{out})},\ \sqrt{6/(N_{in} + N_{out})}\right]$$

where N_{in} is the number of units in the previous and N_{out} the number of units in the current layer.

For training regularization, we applied drop-out in all hidden layers, with a probability of 0.5 as well as weight decay, penalizing the squared sum of all weights in the network ($\alpha = 10^{-4}$). Networks were trained for 10 full epochs.

Training of zebrafish reinforcement learning networks

Reinforcement learning networks learned to navigate temperature gradients by alternating episodes of exploring two circular arenas with a radius of 100 mm: In one arena, temperature increased linearly from 22°C at the center to 37°C at the periphery, while in the other arena temperature increased linearly from 14°C at the center to 29°C at the periphery. The second arena type was necessary as the networks otherwise only learned high temperature avoidance but did not learn to navigate away from temperatures below the preferred temperature, $T_{preferred} = 26°C$.

At each timestep behavior evaluation occurred with a probability of 0.01 leading to a baseline movement frequency of 1 Hz. With a probability $p_{explore} = 0.25$ a behavior was chosen at random (exploration) and such moves were never rewarded. Otherwise the network was fed the navigation history over the last four seconds as input and subsequently the probability outputs of the network were used to select an action. After implementing an action, a reward for that action was calculated according to:

$$R(stay)_t = -\left|T - T_{preferred}\right|_{pre} \Big/ 1000 - (t - t_{last\ move})\ /300$$

$$R(straight,\ left,\ right)_t = \left|T - T_{preferred}\right|_{pre} - \left|T - T_{preferred}\right|_{post}$$

Essentially the reward for movement actions reflected how much closer the agent got to the preferred temperature, while the punishment for staying was dictated by the current distance from the preferred temperature and was additionally punished for the time the network stayed in the current position - this was done to avoid learning the simple strategy of resting at the preferred temperature which is never observed in larval zebrafish. After calculating all rewards for one navigation episode, the discounted return for each move was calculated according to:

$$G_t = \sum_{k=0}^{M-t} 0.63^k R_{t+k+1},$$

where M is the total number of selected behaviors within the epoch. The discounting above effectively implemented rewarding streaks of 10 bouts since the reward influence decayed to 1% after 10 movements. In all equations above, timesteps t refer to actual behavior selections, not simulation timesteps.

At the end of each episode the calculated rewards G_t and their respective history inputs were shuffled and hence used for training of the network in pseudo-random order. This shuffling was done to break high autocorrelations within the navigation data which negatively impacted training. We used an Adam optimizer [learning rate = 10^{-5}, $\beta 1 = 0.9$, $\beta 2 = 0.999$, $\varepsilon = 10^{-8}$] during training (Kingma and Ba, 2014), minimizing

$$L = log\left(p(chosen)_t\right) * G_t$$

Training was performed using backpropagation minimizing the log-loss of the selected action scaled by the delivered discounted return.

Weights in the reinforcement learning networks were initialized as above for the predictive networks. In this case for training regularization we applied drop-out in all hidden layers, with a probability of 0.9 as well as weight decay as above. Networks were trained for 1500 navigation epochs with the number of steps N in epoch i equal to

$$N = 10^6 + 1000 * i$$

This schedule of increasing epoch length was chosen so that networks would initially be exposed to more starting positions within the virtual gradient.

Predictive network navigation

The networks were used for heat gradient (or light) navigation in the following manner: For trained zebrafish networks each timestep had a probability of 0.01 of triggering a behavior (baseline movement frequency of 1 Hz - after evolution this probability depended on the output of p(Move), see below). For *C. elegans* networks the probability of triggering a behavior was set to 10^{-4}, resulting in a frequency of 0.1 Hz. If a timestep was not selected, zebrafish networks stayed in place while *C. elegans* networks continued to move as per the statistics above.

At each behavior evaluation, the preceding 4 s of sensory history as well as speed and delta-heading history were passed as inputs to the network. The network was subsequently used to predict the temperatures resulting from each possible movement choice (or the light angle in case of the phototaxis network). The goal temperature was set to be 26°C and behaviors were ranked according to the absolute deviation of the predicted temperature from the goal temperature. For zebrafish networks, the behavior with the smallest deviation was chosen with a probability of 0.5, the 2nd ranked with a probability of 0.25 and the last two each with a probability of 0.125. For *C. elegans* networks, the highest ranked behavior was chosen with a probability of 0.5 the second highest with probability 0.2 and the remaining three behaviors with probability 0.1 each.

The chosen behavior was subsequently implemented according to the statistics above. Evaluations only resumed after a behavioral module was completed. Whenever a behavior would move a virtual animal outside the circular arena, the behavioral trajectory was reflected at the boundary.

Reinforcement learning network navigation

These networks operated as the predictive networks above. The only difference is that with $p_{explore} = 0.25$ a behavior was chosen at random, while in all other cases the network itself set the probability of each behavior as its output and this probability was subsequently used to pick a behavior. We note that we kept $p_{explore}$ at the same value as used during training and that while its value influences navigation performance, the effect is relatively small in the range from 0.25 to 0.5 and that as expected the behavioral strategy itself remains unchanged.

Evolutionary algorithm to optimize control of swim frequency

A set of 512 weights was used to give the zebrafish networks control over swim frequency. A dot-product between the activations a of the last layer of the temperature branch of the network and these weights w was transformed by a sigmoid function to yield a swim frequency between 0.5 Hz and 2 Hz, by computing swim probabilities between 0.005 and 0.02 according to:

$$p(Swim) = \frac{0.015}{1 + e^{\vec{a}\,\vec{w}}} + 0.005$$

To learn a set of weights w that would optimize gradient navigation performance an evolutionary algorithm was used as follows:

1. Initialize 512 weight vectors, $w \sim N(0,\ 1)$
2. For each weight vector run gradient simulation using it to control p(Swim)
3. Rank weight vectors according to average deviation from desired temperature
4. Pick 10 networks with lowest error and 6 networks at random
5. Form 16*16 mating pairs, mating each network with each other and with itself
6. For each mating pair generate two child vectors by randomly picking each vector element with probability 0.5 from either parent
7. Add random noise to each child vector, $\varepsilon \sim N(0, 0.1)$
8. The 512 created child vectors form the next generation. Repeat from step 1.

Evolution was performed for 50 generations. The average across all 512 weight vectors in the final generation was used to control swim frequency during gradient navigation.

Data analysis

For all network types except the phototaxis network 20 networks were initialized and trained. This number was decided at the beginning, before any analysis was performed. For the phototaxis network a total of 14 networks was trained after observing the small variation across other network types. All presented data are an average across networks which is usually plotted alongside the bootstrap standard error as indicated in figure legends.

Behavior analysis

For zebrafish and networks, swim frequency within the heat gradient was determined by dividing the number of swim starts within temperature bins by the total time spent within that bin during actual or virtual navigation.

To determine turn modulation within the heat gradient (Figure 1I) swims were divided based on the temperature change caused by the preceding swim. Temperature changes > 0.5°C were considered "up gradient," < −0.5°C "down gradient." To normalize for arena effects independent of temperature, the same values were computed during uniform conditions by analyzing swims as if a gradient was present during that time. Subsequently, average turning angles for "up gradient" and "down gradient" bouts for each fish and network during gradient conditions were divided by the average turning angles during uniform conditions.

To determine the swim frequency during the time varying stimulus (Figure S1J) 5000 repetitions of presenting the stimulus to each predictive network were performed and the swim probability (i.e., the probability that the network generated either a "straight swim," a "left turn" or a "right turn") as well as the probability of the network generating any behavior (including "stay") was calculated. Fish data are replot from Haesemeyer et al. (2015). To analyze turn angles (Figure S1K) average turn magnitudes were calculated for times were the stimulus was rising $> 0.5°C/s$, "rising phase," or falling $< -0.5°C/s$, "falling phase" for each fish and network.

White noise analysis

For white noise analysis zebrafish predictive networks were presented with randomly fluctuating temperature stimuli. As during navigation simulations, networks enacted behaviors based on temperature prediction at a frequency controlled by p(Move). The stimulus used for white noise presentation was modeled after the stimulus used previously in freely swimming larval zebrafish (Haesemeyer et al., 2015); however, since there was no water which buffers changes in temperature, the stimulus was switched at shorter intervals and the probed temperature space was larger as well. This allowed for fewer samples overall to result in well-structured filters. Stimulus temperatures in °C were drawn from a Gaussian distribution:

$$T \sim \mathcal{N}(32,\ 6.8)$$

Temperatures drawn from this distribution are almost always above $T_{preferred}$, and the filters hence reflect the behavioral reaction underlying heat avoidance as in Haesemeyer et al. (2015).

Temperature values were switched at random times with intervals in ms drawn from a Gaussian distribution as well:

$$\Delta t \sim \mathcal{N}(50,\ 10)$$

As during navigation simulations, the executed behaviors were used to derive the behavioral history input to the network during the simulations. For each network 10^7 timesteps were simulated and the average temperatures in the 4 s preceding each turn or straight swim were computed.

During the same stimulus scheme cluster average unit activations (Figures 3B–3F) were computed as well by triggering the activity of each unit on a given behavior ("straight" or "turn") and subsequently for each fish averaging the activations across all swims and all units within a given cluster.

Unit response clustering

All clustering was performed on the temperature branch of the networks. To cluster artificial neural network units into response types, the same temperature stimulus was presented to the networks that was previously used to analyze temperature representation in larval zebrafish and the same clustering approach was subsequently employed as well (Haesemeyer et al., 2018). Specifically, the pairwise correlations between all unit responses across all networks of a given type were calculated. Subsequently spectral clustering was performed with the correlation matrix as similarity matrix asking for 8 clusters as this number already resulted in some clusters with very weak responses, likely carrying noise and since PCA across network units furthermore suggested an effective dimensionality of seven of the response space. The cluster means were subsequently used as regressors and cells were assigned to the best-correlated cluster with a minimal correlation cutoff of 0.6. Cells that did not correlate with any cluster averages above threshold were not assigned to any cluster.

For *C. elegans* ANN units the same stimulus was used for clustering and temperature ramp responses were displayed for these obtained clusters as these are more generally used for *C. elegans* characterizations.

To assign units in the mixed branch to clusters in the analysis of the retraining experiments, the same temperature stimulus was presented to the zebrafish ANN while speed and delta-heading inputs were clamped at 0. Correlation to cluster means of the temperature branch, again with a cut-off of 0.6, was subsequently used to assign these units to types.

Connectivity

Connectivity was analyzed between the first and second hidden layer of the temperature branch. Specifically, the average input weight of a clustered type in the first layer to a clustered type in the second layer was determined. The average weight as well as standard deviation across all networks and units was determined. If the standard deviation of a connection was larger than the average weight, the weight was set to 0.

Ablations and retraining of neural networks

Network ablations were performed by setting the activations of ablated units to 0 irrespective of their input. Since fish-like deletions would remove more units from the network overall than non-fish like deletions (63 versus 29% of all units) the number of ablated units in non-fish ablations was matched by additionally removing a random subset of units.

Retraining was performed using the same training data used to originally train the networks, and evaluating predictions using the same test dataset. During retraining, activations of ablated units were kept at 0 and weight and bias updates of units were restricted to either the hidden layers in the temperature branch or in the mixed branch.

To identify unit types in the temperature or mixed branch after ablation or retraining, correlations to the corresponding cluster averages were used while presenting the same temperature stimulus used for clustering to the temperature branch and clamping the speed and delta-heading branch to 0.

Comparison of representations by PCA

To compare stimulus representations across networks, our standard temperature stimulus was presented to all networks. Units from all networks of the types to compare (zebrafish thermal navigation versus zebrafish phototaxis or zebrafish versus *C. elegans* thermal navigation) were pooled and principal component analysis was performed across units. The first four principal components captured more than 95% of the variance in all cases and were therefore used for comparison by evaluating the density along these principal components across network types.

To analyze the representational complexity of a network (type) the same temperature stimulus was presented and principal component analysis was performed across all units in a given network (type). It was subsequently determined how many principal components cumulatively explained at least 99% of the total variance across all units.

Matching of ANN and zebrafish heat response types

To correspond clusters in the predictive and reinforcement learning zebrafish heat navigation networks, correlation analysis between the average response cluster activity and the average activity of zebrafish response clusters in the Rh 5/6 region of the hindbrain (Haesemeyer et al., 2018) was performed. Since the temperature stimulus varies with slow dynamics, leading to large correlations even in the case of spurious matches, correlations were limited to the sinewave and immediately following temperature step period (60-105 s of the stimulus). To be considered a matching type a minimum Pearson correlation of 0.6 was required.

Identification and mapping of novel zebrafish response types

To identify potential novel zebrafish response types, ANN cluster average responses were used for regressor analysis (Miri et al., 2011) by probing the zebrafish whole brain imaging dataset. The cluster average response was correlated to every cell in the zebrafish dataset. All neurons with a response correlated > 0.6 to these cluster averages was considered part of the response type (but we note that even when increasing or decreasing this threshold no neurons could be identified that matched the ON-OFF type in the sense that no neurons in the zebrafish dataset showed both an ON and an OFF response).

To map these identified neurons back into the zebrafish brain, we used the neuron’s coordinates in the reference brain generated in the imaging study.

Linear model based mapping of zebrafish temperature response types

To test similarities in temperature encoding between the zebrafish temperature ANN and BNN a linear model was used to relate unit activity in the temperature branch of one ANN (ANN 0, chosen arbitrarily) to activity across all cells in the zebrafish BNN. For this purpose ridge regression ($\alpha = 0.1$) was used to fit a multiple regression model relating the first two trials of stimulus presentation in the 1024 units present in the ANN to the calcium activity during the first two trials of stimulus presentation in 699840 neurons across the zebrafish brain dataset. Subsequently, the model was applied to the ANN responses of the third stimulus presentation trial and the quality of prediction (R^2) of the third trial of zebrafish neuron responses was scored. At the same time a self-prediction score was calculated as the R^2 of the correlation between the sum of the first two trial responses for each cell and the third trial response of the same cell.

For the presented data, a neuron was considered identified positively by this method if R2 $\geq$ 0.25, i.e., if the linear prediction explained at least 25% of the variance in the neuron’s response during the third trial provided that the prediction score was not higher than the self-prediction score. The rationale behind this filtering step was that a model fit on the first two trials should not be better in predicting the third trial than the first two trials themselves.

QUANTIFICATION AND STATISTICAL ANALYSIS

The number of samples (n) generally refers to comparisons across networks or fish and is indicated in the figure legends.

Displayed errors (bars or shadings) were computed as bootstrap standard errors in all plots.

For turn strength comparisons a non-parametric Wilcoxon test across networks and fish was used.

To compute the expected overlap of ANN prediction and zebrafish clustering analysis (Figure S2C′) 1000 shuffles of the positive prediction labels were performed and the average overlaps across these shuffles are given in the figure.

DATA AND CODE AVAILABILITY

The full source code of this project is available on Github (https://github.com/haesemeyer/GradientPrediction) and all relevant data are deposited on Zenodo at (https://doi.org/10.5281/zenodo.3258831).