The development, recognition, and learning mechanisms of animal-like neural network

Feng Qi[1]

[1]Oxford Centre for Functional MRI of the Brain, Wellcome Centre for Integrative Neuroimaging, Nuffield Department of Clinical Neurosciences, University of Oxford, UK

How animal neural system addresses the object identity-preserving recognition problem is largely unknown[1]. Artificial neural network such as convolution network[2] (CNN) has reached human level performance in recognition tasks, however, animal neural system does not support such kernel scanning operation across retinal neurons, and thus the neuronal responses do not match that of CNN units. Here, we used an alternative recognition-reconstruction network (RRN) architecture as an analogy to animal-like system, and the resulting neural characteristics agreed fairly well with electrophysiological measurements in monkey studies[3,4]. First, in network development study, the RRN also experienced critical developmental stages characterized by specificities in neuronal types, connectivity strength and firing pattern, from early stage of coarse salience map recognition to mature stage of fine structure recognition. In digit recognition study, we witnessed that the RRN could maintain object invariance representation under various viewing conditions by coordinated adjustment of responses of population neurons. And such concerted population responses contained untangled object identity and properties information that could be accurately extracted via a simple weighted summation decoder. In the learning and forgetting study, novel structure recognition was implemented by adjusting entire synapses in low magnitude while pattern specificities of original synaptic connectivity were preserved, which guaranteed a learning process without disrupting the existing functionalities. This work benefits the understanding of human neural mechanism and the development of humane-like intelligence.

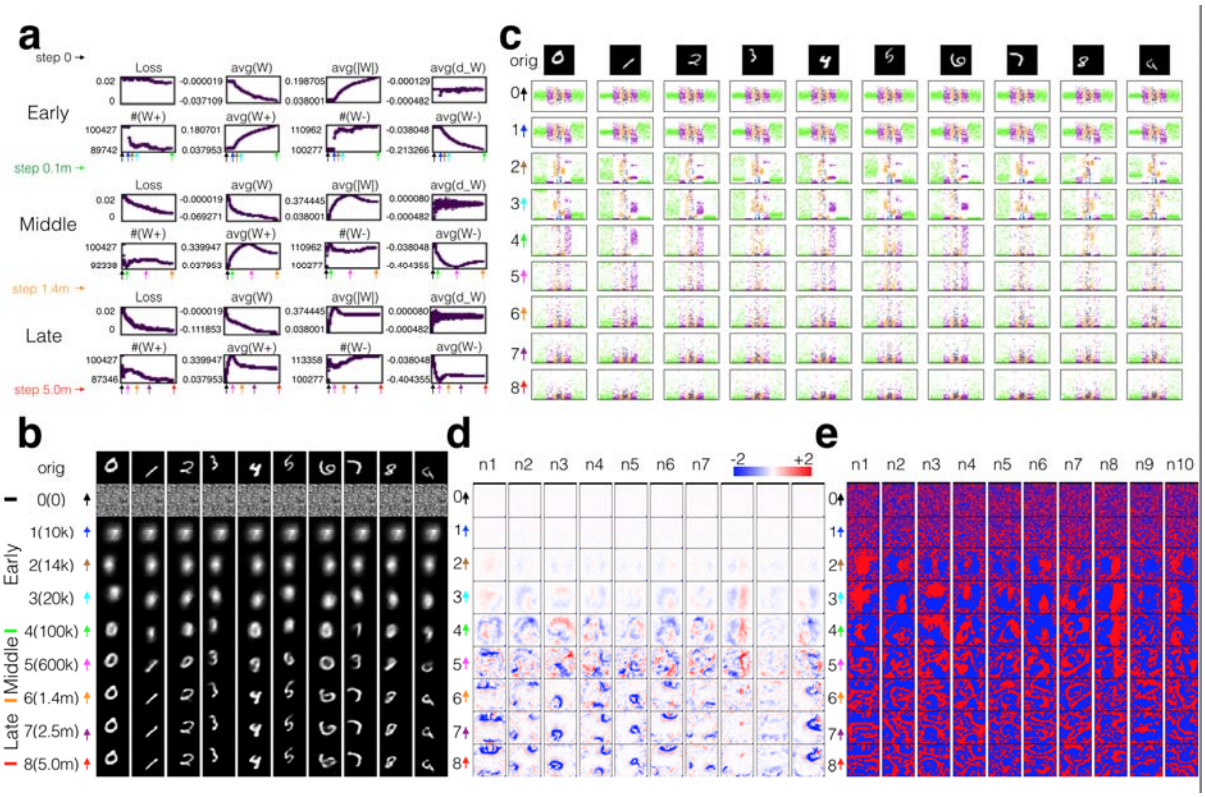


Fig.1| Development process. **a**, Network parameters evolution over the whole development process (Extended Data Table 1). The development process contains three stages (the early, middle and late stage), and 9 critical time points (ctp) according to the key changes in synaptic weights, neuronal firing patterns, and digit reconstruction performance. **b,** Digit reconstructions across the development process, where RRN first figured out the digit's salience map with correct retinal location, size and orientation (early stage), then engraved the basic structure based on the salience map (middle stage), finally refined the structural details in a longer duration (late stage). **c,** Evolution of neuronal response maps for 10 different digit instances, ranging from initial non-specific firing to final object-related firing. Neurons are presented in order, and layers are indicated by colors. **d,** Evolution of synaptic weights from retina to the first layer (Extended Data Fig.2 for all neurons). The increase of weight strength of both excitatory (positive) and inhibitive (negative) synapses facilitated to capture the basic characteristics of digits (ctp0-5), whereas the subsequent drop of weight strength, while keeping synaptic pattern sharpen, enabled to recognize structure details (ctp5-8). **e.** Evolution of distribution of excitatory (red) and inhibitive (blue) synapses (Extended Data Fig.3 for all neurons). At the beginning, it shows random distributed patterns (ctp0-1), followed by cluster-like patterns (ctp2-4), and finally stripe-like patterns(ctp5-8). Note, k represents 1,000 steps and m represents 1,000,000 steps.

Experimental neuroscience[5] discovers that ventral stream in primate brain has been used for core object recognition with information transmission through a feedforward, reflexive, hierarchical neural network to rapidly untangle the object identity manifold[6,7]. Inspired from this, we build a simple recognition and reconstruction network (RRN) (Extended Data Fig.1) to provide an analogy to animal neural system, and expect similar traits could be derived independently in response to similar recognition challenges.

As primates have distinct development stages (e.g. baby, adolescence and adult etc.) characterized with significant difference in neural anatomy, physiology and cognition, the RRN also experienced similar development process (i.e. the Early, Middle, and Late stages) characterized by 9 critical time points (ctp) each with unique firing pattern, synaptic connectivity pattern, and recognition performance (Fig.1). In the Early stage (ctp0-4), RRN took relatively few steps, 100k only, to learn to capture the salience map of objects in retina. The RRN started with random assigned synaptic weights (ctp0) that possessed no power for recognition though digits were presented in retina. Despite there was only slight synaptic plasticity in initial stage (ctp1), the RRN started to realize that those digits were likely to appear in the center of retina. The initial activity changes occurred only in higher layer, indicating the importance of early development of other cortices (like cortices for audition, proprioception and association etc.) and their feedback signals in shaping those V1-like synapses. The ctp2 witnessed a sudden drop in number of excitatory synapses (so was the increase of inhibitive synapses), and ctp2 is also the starting point of strength growth of both excitatory and inhibitory synapses. These changes allowed the RRN to identify whether the digits were presented on left or right part of the retina, which was clearly reflected by the horizontal segmentation of positive and negative synapses (Fig1.e). Following the same trend of synaptic alterations, the RRN could identify the vertical position and orientation of objects in ctp3, with specific cluster-like distributions gradually formed. The units also showed remarkable firing specificity with respect to digit's retinal position. In ctp4, the RRN could capture the object scale information, and this was also the start point when RRN came to identify digit's morphology, namely, entering the Middle stage (ctp4-6). In the Middle stage, the most striking changes occurred in ctp5, where both excitatory and inhibitory synapses stopped strengthening and began to shrink to enhance the recognition, quite similar to the prune mechanism in animal brain to improve the neural network efficiency[8]. At ctp6 the prune effect was quite clearly demonstrated in both activity maps with significantly sparser firing patterns and synaptic weight maps with fewer but sharper distributions. At the same time, V1-like units formed specific visual field in retinal region. From ctp6, the RRN could basically recognize and reconstruct a correct digit, and the further refinement of the structural details was developed in the Late stage (ctp6-8) on a longer time scale. The excitatory to inhibitory ratio of synapse number continually declined in a moderate speed till cpt7 where the network's recognition performance reached a peak level. And the subsequent synaptic plasticity (in ctp8 and further) appeared to stabilize and so were other parameters. The entire process matches the development[9,10] of animal and human.

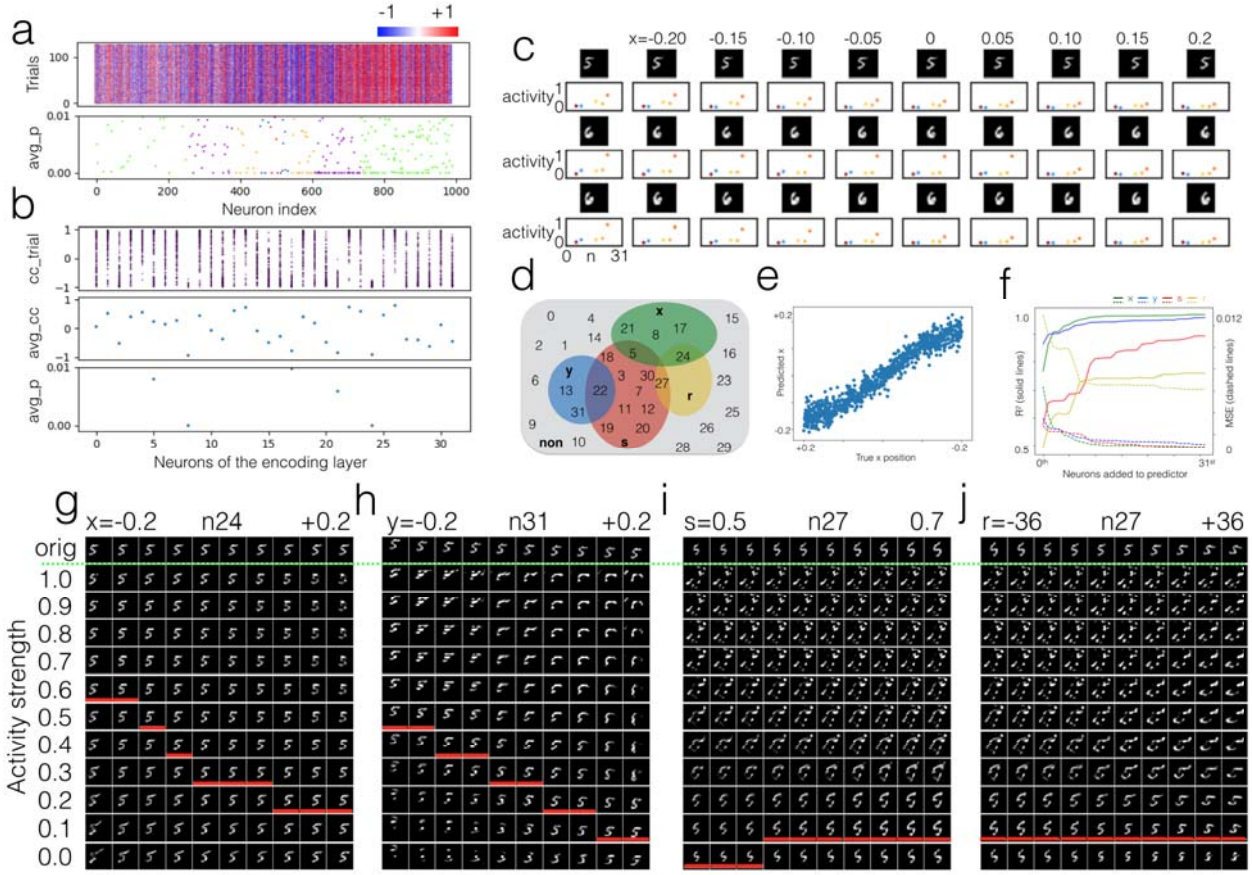


Fig. 2 | Neuronal responses to digit translation, scaling, and in-plane rotation. **a.** Neuronal responses to 128 trials of horizontal translation x (top), and significantly correlated x-neurons (bottom). Note that all layers contained certain x-neurons. **b.** The correlation coefficient (cc) between neuronal activity and x position for 128 trials (top), where each point represents one trial, and only 32 neurons in the Encoding layer are presented. The middle and bottom panels show the averaged cc across 128 trials and 5 significant ($P < 0.01$) x-neurons, respectively. **c.** x translation, and corresponding responses of x-neurons. Note, activities of neuron 24 (n24, orange point, the most significantly x-related) were strongly modulated by digits' x positions, identities, and even morphologies. **d.** Neuronal categorization according to digits' properties. The non-specific and x, y, s, r specific neurons took proportion of 46.9%, 15.6%, 9.4%, 34.4%, 6.3% respectively with certain overlaps. **e.** Prediction of position x. The encoding layer contained untangled x information that were easily extracted by a simple linear decoder implemented by a downstream layer. Similar studies were performed for vertical translation (y), scaling (s), and rotation (r) in the Extended Data Fig. 4-7. **f.** Prediction performance (coefficient of determination R^2 and mean square error) for digit properties. The retinal position (x, y) were predicted with

relatively high accuracy, while scaling and orientation (r, s) were with relatively low accuracy, primarily due to the innate property variations in the training set. **g.** The influence of n24 on digit reconstructions regarding to x position. From left column to right, x position increases from -0.2 to +0.2 as illustrated on top row of original digit images. All other rows are images reconstructed with modified neuronal activities of n24. Red bars highlight the optimal activities derived from RRN. **h-j** similar plots for y, s, and r. Neurons were selected by lowest p-value during correlation studies.

After 5 million steps training, the RRN was mature in recognition of both digit properties and structure details. Then, we quantitatively characterized the information represented in the pivot Encoding layer to investigate how the digit properties were encoded by these 32 neurons. Fig. 2a shows the correlation analysis between retinal position x and neuronal activity across 128 horizontal translation trials (Method). We witness that all layers contain certain number of x-related ($P < 0.01$) neurons, which is reasonable and even necessary, because precise digit reconstruction requires retinal position information being carried throughout the whole network. Five out of 32 neurons in the Encoding layer are x-neurons in fig. 2b-c, in which activity of neuron 24 (n24, the most significantly related) demonstrates three characteristics: (1) digit position modulated activity of n24; (2) the modulation relied on digit identity and morphology; (3) continuous position alteration resulted in continuous change of neuronal response. The characteristics of diffeomorphic mapping of RRN can guarantee the tolerance to position variation. All these are consistent with the characteristics of property preserving representation of IT neurons[1,11] in monkey studies.

Similar correlation studies were performed on properties of y, s and r (Extended Data Fig. 4-7), and the 32 neurons were functionally categorized into 5 classes (Fig. 2d), which demonstrate the population coding nature[12] of these neurons. To test whether these population responses represent the digit properties, a simple linear summation decoder of downstream circuit (Extended Data Fig. 1) was implemented (method) to predict position x from population responses of the Encoding neurons. Fig. 2e demonstrates that downstream layer could perceive the objective (predicted) x value with a simple linear circuit. Fig. 2f shows the predictive performance in term of R^2 and mean square error (MSE), demonstrating that perceptions of retinal position x and y were most accurate, while that of digit size was worse, and in-plane orientation was the worst. Interestingly, although we didn't train the RRN with any digit properties, it could automatically untangle the manifolds of these properties.

To investigate what individual neuronal response represents, we modulated each neuronal activity while traced the alteration of image reconstruction. Fig. 2g demonstrates how n24 response influences reconstruction of digit 5 in various x positions, where we have three observations: (1) for each column, though n24 significantly changed activity, reconstructions of digit 5 were always placed in correct x positions, which suggests that neighbor neuronal activities together also contained the position x information, and n24 activity alone could not

move reconstruction position. (2) For each column, different n24 activity did severely change the reconstructed morphology, however, without communicating with neighbor neurons in the same layer, n24 could readily obtain accurate activity from feedforward inputs of previous layer, which suggests the response of n24 (the most significant x-related neuron) was induced by features in specific retinal regions. (3) across all blocks, there was no sudden change occurring no matter slight move of digit or change of n24 activity, which suggested the RRN reconstruction possessed noise tolerance to digit translation and neuronal activity. Quite similar is the influence of n31 activity on digit's y position, but the reconstruction quality of morphology was more sensitive to the variation of n31 activity. Fig. 2i-j shows that both digit scaling and rotating were significantly correlated to n27 activity, which maintained fairly low but extremely important activity magnitude around 0.1, and disturbing such magnitude would substantially worsen the reconstruction quality, which suggests both big and small responses are necessary parts of the population coding[13]. In sum, in the population coding scheme, what exactly a single neuronal activity represents can hardly be described, because the interpretation may dramatically alter when neighboring neurons change firing patterns.

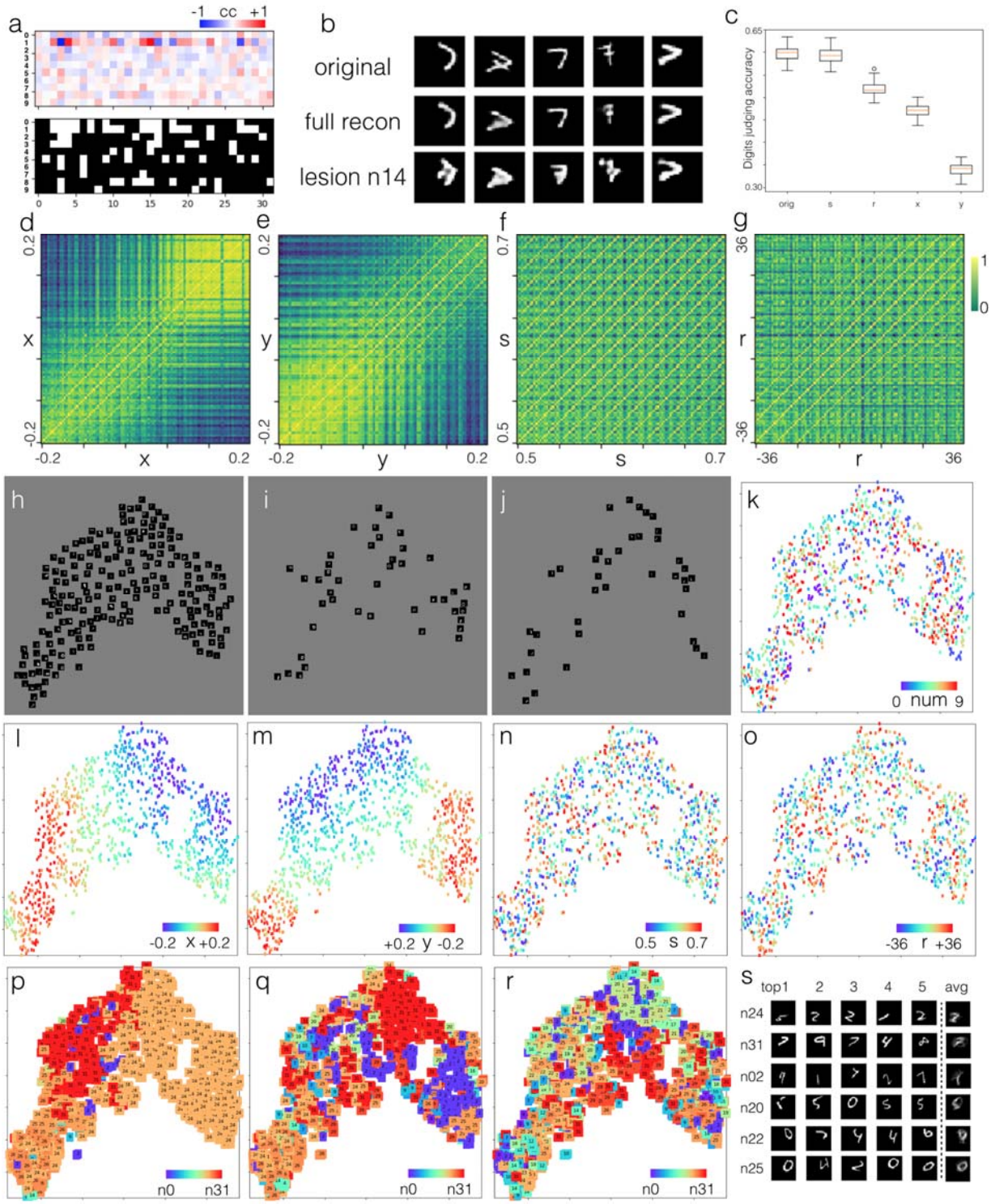

a
-1 cc +1
b
original
full recon
lesion n14
c
Digits judging accuracy
orig
s
r
x
y
d
e
f
g
x
y
s
r
h
i
j
k
0 num 9
l
-0.2 x +0.2
m
+0.2 y -0.2
n
0.5 s 0.7
o
-36 r +36
p
q
r
n0 n31
s
top1
2
3
4
5
avg
n24
n31
n02
n20
n22
n25

Figure 3: Digit identity recognition. a, Correlation coefficient matrix between neuronal activity and digit identity, and significantly ($P < 0.01$) identity related neurons highlighted in white. **b,** Lesion of n14 (whose activity was highly correlated to occurrence of digit 7) damaged the reconstruction of various digit 7 in different degree. **c,** Digits judging accuracy without or with perturbation of x, y, s, r properties. Though training was conducted with no digit labels, a simple logistic classifier could be used to predict digit identity from population responses of the Encoding layer. Note, accuracy of natural guess is 10%. **d,** Population similarity matrices[4] for x translation. The 100 by 100 correlation coefficient matrix was computed between population responses across 32 Encoding neurons to each of 100 stimuli, which were constituted by ten x-position blocks, each contained ten digits from 0 to 9 (method). Note that the paradiagonal stripes indicate property-invariant digit selectivity. **e-f** are similar matrices for properties of y, s, and r. **h,** Non-overlapping view of digit instances in 2D tSNE space. It clearly demonstrates that these images were primarily arranged according to retinal positions, while digit identities were intertwined. **i-j,** Distribution of digit 0 and 1 in 2D tSNE space, respectively. **k-o,** Distribution of image identities and properties of x, y, s, r in 2D tSNE space. Retinal position x and y clearly dominated the global distribution of images, while digit identity, scaling and orientation showed different locally clustering characteristics. **p-r,** The first 3 highest responsive neurons for each image in 2D tSNE space. n24 and n31 fired strongest for most digit images, which seemed to encode features in specific retinal regions, while other neurons managed to add or subtract structural details in these regions. **s,** top5 favorite and average (of top20) images. These neurons fired in favor of some features in specific regions, not necessarily pointing to specific digit identity.

To tackle whether neuronal activities encode digit identity, correlation coefficients between neuronal activities and digit identities were computed (fig.3a), where nearly all neurons were involved in digit identity coding. The Extended Data Fig.8 shows that though variations existed in digit morphologies, these corresponding digit-related neurons had conservative response pattern regarding to each identity. Nevertheless, no single identity neuron found in the Encoding layer. For instance, n14 (the most correlated neuron to digit 7) showed diverse influence on reconstruction of digit 7 in different conditions (fig.3b). Then, we used a simple logistic regression layer (Extended Data Fig.1) to assess whether neurons ensemble contain explicit digit identity information. As compared with the baseline accuracy of 10%, this simple logistic regression layer could predict digit with accuracy of 60% (fig. 3c), though no labels were provided during the training process, which indicates that the Encoding neurons are capable of untangling digit identity which allows downstream layers to easily associate category or name to such neurons. In addition, we found that if the digits were placed in the retinal center, certain rotating or scaling would not drastically deteriorate the recognition accuracy, however, placing digits to peripheral regions especially on top or bottom of retina would significantly damage the digit recognition, that’s why eye movement is necessary for animal neural network.

Visual processing needs to handle identity preserving image transformation problem, or to tolerate property variations during identity recognition. To investigate this, we introduced similarity matrices[4] between population responses to digits with gradual changing properties (method). One remarkable feature in Fig. 3d is the presence of paradiagonal stripes, which indicates same digit with decent horizontal displacement will not severely disturb population response pattern. The other feature is the high correlation region on top right, which indicates that digits placed on right part of retina will elicit similar neuronal activities regardless their identities. Vertical variation of digit position has resembled similarity matrix (fig.2e). Nevertheless, variation of digit size demonstrates remarkably more and stronger paradiagonal stripes than others, which suggests that for scaling effect, digits with same morphology were encoded by similar population response pattern, and only slight modulation on population activities was needed for recognizing the size alteration. In contrast, relatively large adjustment of coordinated population response was needed to recognize retinal position changes[14], which is consistent with Extended Data Fig. 9. So the natural neural mechanism in handling translational invariance is not based on max pooling[15] but on coordinated adjustment of population responses.

t-Distributed Stochastic Neighbor Embedding (tSNE)[16] could provide visible insights into how these digit instances organized in a 2D state space. Fig. 3h-j illustrate two main findings: (1) retinal position x and y dominated the global distribution of digit in the tSNE space; (2) digit identities had a tendency to cluster locally. So alteration of retinal position meant a big move in tSNE space, which required a large scale adjustment of coordinated responses to achieve it; while alteration of digit size meant a small nudge (due to unchanged x and y) in the tSNE space, which required little adjustment in population responses and showed remarkably more and stronger paradiagonal strips. From fig.3p-r, we witness that activities of n24, n31, n2 and n25 might provide location guidance information for image reconstruction, while other neurons provided relatively small activity to refine the structure in such regions. However, both big and small responses were integral parts of population coding. Fig. 3s (Extended Data Fig. 10) shows that neurons did fire in favor of features in specific retinal region, instead of digit identities.

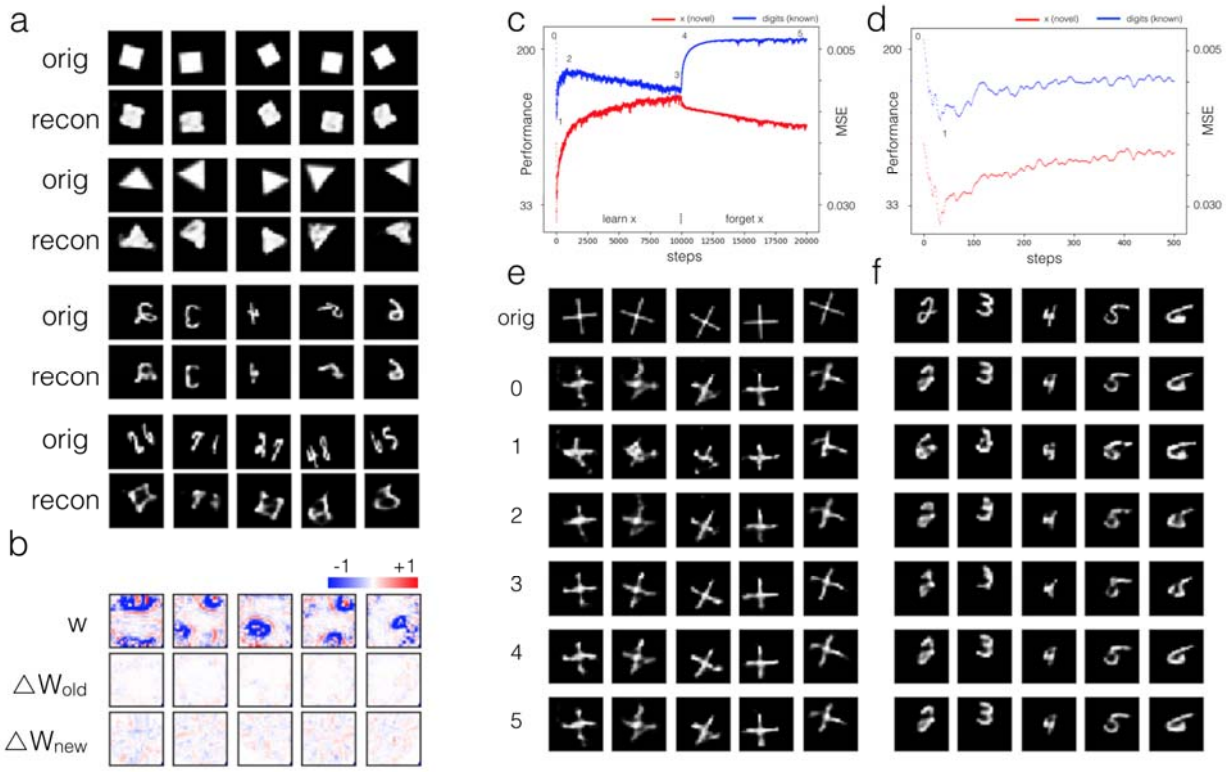


Fig4. Learning and forgetting. **a**, Reconstruction of new structures including solid square, triangle, mirrored digits and double digits. **b,** Synaptic weights distribution and changes after 10,000 steps training of known and novel (symbol x) structures. **c,** Performance evolution during training and forgetting processes. The RRN was trained with novel symbol x for the first 10,000 steps, followed by known digits in the next 10,000 steps. Red represents test performance (or MSE in upside-down y-axis) of novel structure x, while blue represents known digits. Six distinguished ctps (0-5) were observed across the entire process. **d,** enlarged view of the first 500 steps which clearly demonstrates the initial drop and subsequent rise of reconstruction performance. **e-f,** Illustrate the reconstruction of symbol x and digits for 6 ctps, which reflected the performance evolution.

Learning new and forgetting old skills (or knowledge) are basic characteristics of animal brains. Here, we will reveal how new recognition skills are added into the working network while previous skills are still preserved without big disruption. It is not surprising, when novel structures are fed into RRN, it treats the input as known digits and reconstructs output accordingly. In fig.4a, the RRN treated the square as compacted digit 8, and double digit as a wide single digit. Fig.4b demonstrates that novel structures (symbol x) induced significantly (2.5

times, P = 2.43e-50) stronger synaptic plasticity than known structures (digits). More importantly, these changes occurred on all synapses in relatively small magnitude, so that previous sharp synaptic patterns could be retained. This guarantees that those known structures could still elicit similar neuronal responses, which enable downstream layers to process data properly and robustly without big modulation. This also guarantees that any novel structure is learnt and recognized by population coding scheme.

The population rewiring process of learning and forgetting involved several key stages (fig. 4c-f). At ctp-0, the RRN could recognize and reconstruct both novel and known structures with relatively good quality. Surprisingly, the first 30 steps training of symbol x made sudden drop of performance for both structures until ctp-1. Novel experience made RRN unstable, but after transition 0-1 the RRN found a new attractor to settle on in the state space[17], thenceforth a new round of adjustment started. In transition 1-2, RRN rapidly improved the performance for symbol x because it was the only input to the network. Unexpectedly, the performance for known digits also got improved, which might be due to the synaptic adjustment for better reuse of previous connectivity. After sufficient recovery of RRN capacity, performance for known structure began to decline again until ctp-3. After that, the RRN began to see only digits again, the network soon recovered the digit recognition performance to original level, while forgetting of symbol x took effect rapidly. For transition 4-5 and further, input of digits continued to enhance the digit recognition while decrease that of symbol x. Note that real neural network is much larger, the performance alteration (blue curve from ctp-0 to ctp-3) would not change such dramatically. Here, we want to differentiate between recognition learning and knowledge learning, where the former (like tasting spicy will gradually lower the perception of spicy taste) is a long term process requiring plasticity of sensory synapses along processing stream, while the later (like learning a new foreign word) could be a short process requiring certain new association connectivity formed between sensory and downstream regions.

Under the same principle of minimizing free energy of system[16], animal and RRN independently evolved similar neural network, with matched observations in developments, neuronal response, and learning characteristics etc. The work provides insights into studies of both neuroscience and AI.

Reference

1. Dicarlo, J. J., Zoccolan, D. & Rust, N. C. How Does the Brain Solve Visual Object Recognition□? *Neuron* **73,** 415–434 (2012).
2. Lecun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521,** 436–444 (2015).
3. Li, N., Cox, D. D., Zoccolan, D. & Dicarlo, J. J. What Response Properties Do Individual Neurons Need to Underlie Position and Clutter " Invariant " Object Recognition□? *J. Neurosci.* **102,** 360–376 (2009).
4. Winrich A. Freiwald, D. Y. T. Functional Compartmentalization and Viewpoint Generalization Within the Macaque Face-Processing System. *Science (80-.).* **330,** 845–852 (2010).
5. Felleman, D. J. & Essen, D. C. Van. Distributed Hierarchical Processing in the Primate Cerebral Cortex. *Cereb. Cortex* **1,** 1–47 (1991).
6. Dicarlo, J. J. & Cox, D. D. Untangling invariant object recognition. *Trends Cogn. Sci.* **11,** 333–341 (2007).
7. Cadieu, C. F. *et al.* Deep Neural Networks Rival the Representation of Primate IT Cortex for Core Visual Object Recognition. *PLoS Comput. Biol.* **10,** e1003963 (2014).
8. Chechik, G. Neuronal Regulation□: A Mechanism for Synaptic Pruning During Brain Maturation. *neural Comput.* **11,** 2061–2080 (1999).
9. Paolicelli, R. C. *et al.* Synaptic Pruning by Microglia Is Necessary for Normal Brain Development. *Science (80-.).* **333,** 1456–1458 (2011).
10. Magueresse, C. Le & Monyer, H. Review GABAergic Interneurons Shape the Functional Maturation of the Cortex. *Neuron* **77,** 388–405 (2013).
11. Dicarlo, J. J. & Maunsell, J. H. R. Anterior Inferotemporal Neurons of Monkeys Engaged in Object Recognition Can be Highly Sensitive to Object Retinal Position. *J neurophysiol* **89,** 3264–3278 (2002).
12. Meyers, E. M. *et al.* Dynamic Population Coding of Category Information in Inferior Temporal and Prefrontal Cortex. *J neurophysiol* **100,** 1407–1419 (2019).
13. Haxby, J. V *et al.* Distributed and Overlapping Representations of Faces and Objects in Ventral Temporal Cortex. *Science (80-.).* **293,** 2425–2431 (2001).
14. Cox, D. D. & Dicarlo, J. J. Does Learned Shape Selectivity in Inferior Temporal Cortex Automatically Generalize Across Retinal Position□? *J. Neurosci.* **28,** 10045–10055 (2008).
15. Nagi, J. *et al.* Max-Pooling Convolutional Neural Networks for Vision-based Hand Gesture Recognition. *2011 IEEE Int. Conf. Signal Image Process. Appl.* 342–347 (2011).
16. G.E. Hinton, R. R. S. Reducing the Dimensionality of Data with Neural Networks. *Science (80-.).* **313,** 504–508 (2006).
17. Friston, K. The free-energy principle□: a unified brain theory□? *Nat. Rev. Neurosci.* **11,** 127–138 (2010).

Method

RRN architecture. The RRN contains ten densely-connected layers, that process the data by means of Outputs = sigmoid (inputs * W +b), where W is synaptic weights matrix, b is bias vector, and sigmoid is the activation function. The serial processing architecture (Extended Data Fig. 1) contains two components of Recognition and Reconstruction, where the former can convert 28*28 image into low dimensional features represented by 32 Encoding neuronal responses, and the later can convert such responses into reconstructed image. After that, the image reconstruction error is calculated and back propagated, and synaptic weights and biases are updated by Adam optimizer. For digit identity and properties prediction, we branch the Encoding layer to a simple logistic and linear layer, respectively. The network was implemented by Tensorflow r.1.12.0 and sklearn 0.20.2 under python3.6. Note, we don't differentiate between neuron and unit, between response and activity. Training parameters were selected as, learning rate = 0.001, Batch size = 128, training steps = 5m, and all other parameters were default values.

Image processing. The digit images were from mnist (http://yann.lecun.com/exdb/mnist/), which had a training set of 60,000 examples and a test set of 10,000 examples. We used imgaug 0.2.7 to further expand the mnist set with variation of translation, scaling and in-plane rotation. Other object images like square, triangle, mirrored digit, double digits, and symbol x, were created by custom python codes. All figures were created by matplotlib 3.0.2. Properties were randomly selected in ranges of retinal position $x \in [-0.2, 0.2]$, position $y \in [-0.2, 0.2]$, scaling $s \in [0.5, 0.7]$, rotation $r \in [-36°, 36°]$, so as to keep the transformed images in frame.

Correlation study of digit properties. Considering retinal position x as an example, we performed 128 trials of simulation on the mature (5million steps trained) RRN network to compute the correlation coefficients between neuronal activity and x value for each neuron. Each trail contained 100 images of same random digit instance but varying position x from -0.2 to +0.2. So for each trial, with the 1*100 x-vector and 1*100 neuronal activity vector, one pair of cc and p-value could be obtained for each neuron. And then the averaged cc and p-value could be computed by averaging across all 128 trials, and the significant x-neurons were selected with threshold 0.01. The same operations were performed for digit property y, s, and r, whose results are presented in Extended Data Fig. 4-7. Correlation computation was implemented with scipy.stats.pearsonr.

Digit properties prediction. Considering retinal position x as example, we first took 1024 digits from mnist test set, and then transformed each digit with random properties while documented the x value. Next, by feeding digits into the RRN, the activities array of the Encoding neurons could be obtained. After that, the sklearn.linear_model.LinearRegression was trained with the documented x values and corresponding activity arrays. In prediction phase, we created another 1024 unseen digits transformed with random properties while position x documented. By feeding these new digits into RRN, a new set of activity arrays were obtained. Taking these new activity

arrays as input to the sklearn.linear_model.LinearRegression, we obtained the predicted position x. Finally, r2_score and mean_squared_error functions from sklearn.metrics were used to compute the prediction performance in term of R square and MSE, respectively. Predictions of properties y, s and r were processed in similar way.

Correlation study of digit identity. 1024 mnist digits were sequentially processed with random operation of scaling [0.5,0.7], rotating [-36°, 36°], x translation [-0.2, 0.2], and y translation [-0.2, 0.2] to produce digit instances with various properties, while identities (mnist labels in one-hot mode) were documented. By feeding these digit instances into (5million steps trained) RRN network, population responses were obtained for each instance. Finally, the scipy.stats.pearsonr was used to compute correlation coefficient and p-value for each pair of digit identity and neuronal activity (Fig. 3a).

Digit identity prediction under property perturbation. Considering perturbation of x position as an example, two sets (one for training, one for testing) of 1024 mnist digits were sequentially processed by shrinking s = 0.6, and then random $x \in [-0.2, 0.2]$ to produce digit instances with x perturbation, while documented digit identities. By feeding the training set of transformed digit instances into the (5million steps trained) RRN network, population responses were obtained. Then, we trained the sklearn.linear_model.LogisticRegression model with population responses (32 neurons in Encoding layer) and corresponding digit labels. For the test set of transformed digits, RRN output another set of population responses, by feeding which, the trained LogisticRegression model produced digit labels predictions. By comparing the predicted and original labels, the statistics under x-perturbation were displayed as boxplot in Fig. 3c. Similar processes were conducted for other properties.

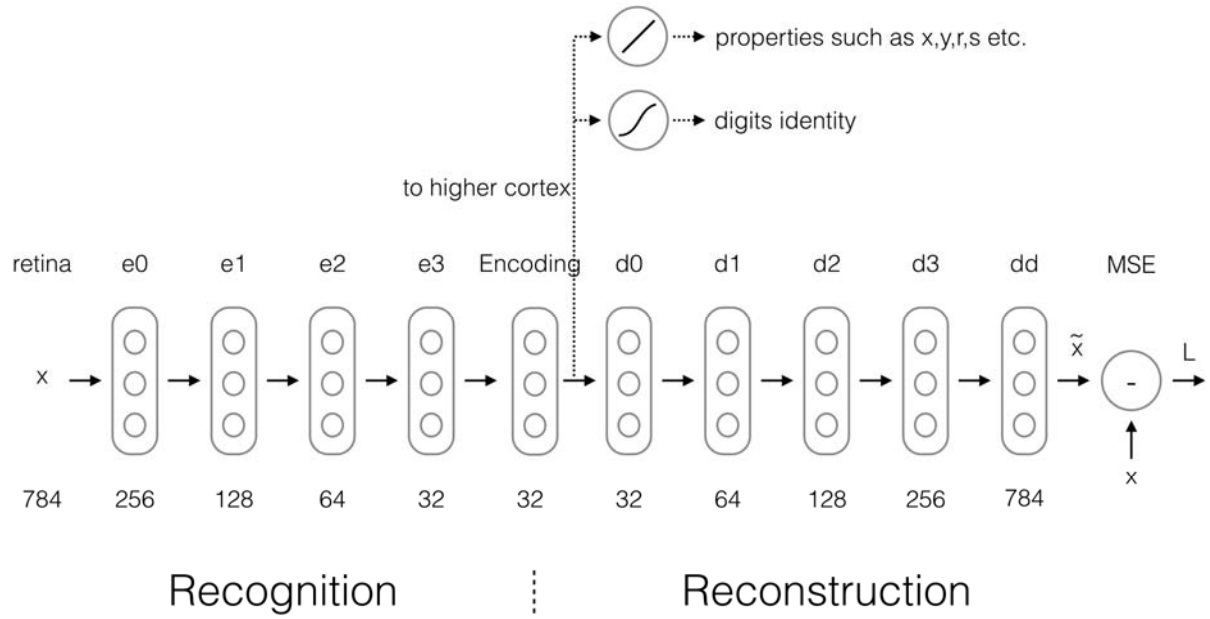
Similarity matrix. Considering translation x as an example, ten mnist digits (0-9, one for each identity) were first shrank with s = 0.6, then duplicated 10 times with each copy being transformed by a x-value from -0.2 to 0.2. Totally 100 images were created and arranged in order, where 10 blocks represented 10 x values, while each block contained 10 different digit identities. By feeding these 100 images into the (5million steps trained) RRN network, we obtained a 100 by 32 response matrix, where each row was a population response vector from 32 neurons with respect to one image. Finally, the correlation coefficient between any two images were computed by the scipy.stats.pearsonr model. Putting all 100 by 100 correlation coefficients together, we constructed the similarity matrix of fig. 3d. Similar processes were conducted for other properties (fig. 3e-g).

tSNE space representation. 1024 mnist digits were transformed with random properties of s, r, x and y to produce the non-biased dataset. By feeding these digit instances into the (5million steps trained) RRN network, the population responses were obtained in form of a 1024 by32 matrix (32 neurons in the Encoding layer). sklearn.manifold.TSNE (perplexity = 30, n_components = 2, init = 'pca', n_iter = 1000) was used to reduce the 32 dimensional

representation into 2D. Taking these two components as x-axis and y-axis, and property as color, the property distribution in 2D tSNE space was presented (fig. 3k-r).

Top-5 and average (top-20) image instances. 1024 mnist digits were transformed with random properties of s, r, x and y to produce the non-biased dataset. By feeding these digit instances into the (5million steps trained) RRN network, the population responses were obtained. For each neuron, rearrange the image instances in a descending order, according to its activity strength. After that, it output the first five images that elicited the highest neuronal responses (Fig. 3s and Extended Data Fig. 10). The 'avg' image was averaged across the top-20 images, and the 'sigf' images contained pixels whose values (over these top-20 images) were significantly ($P < 0.01$) different from zero.

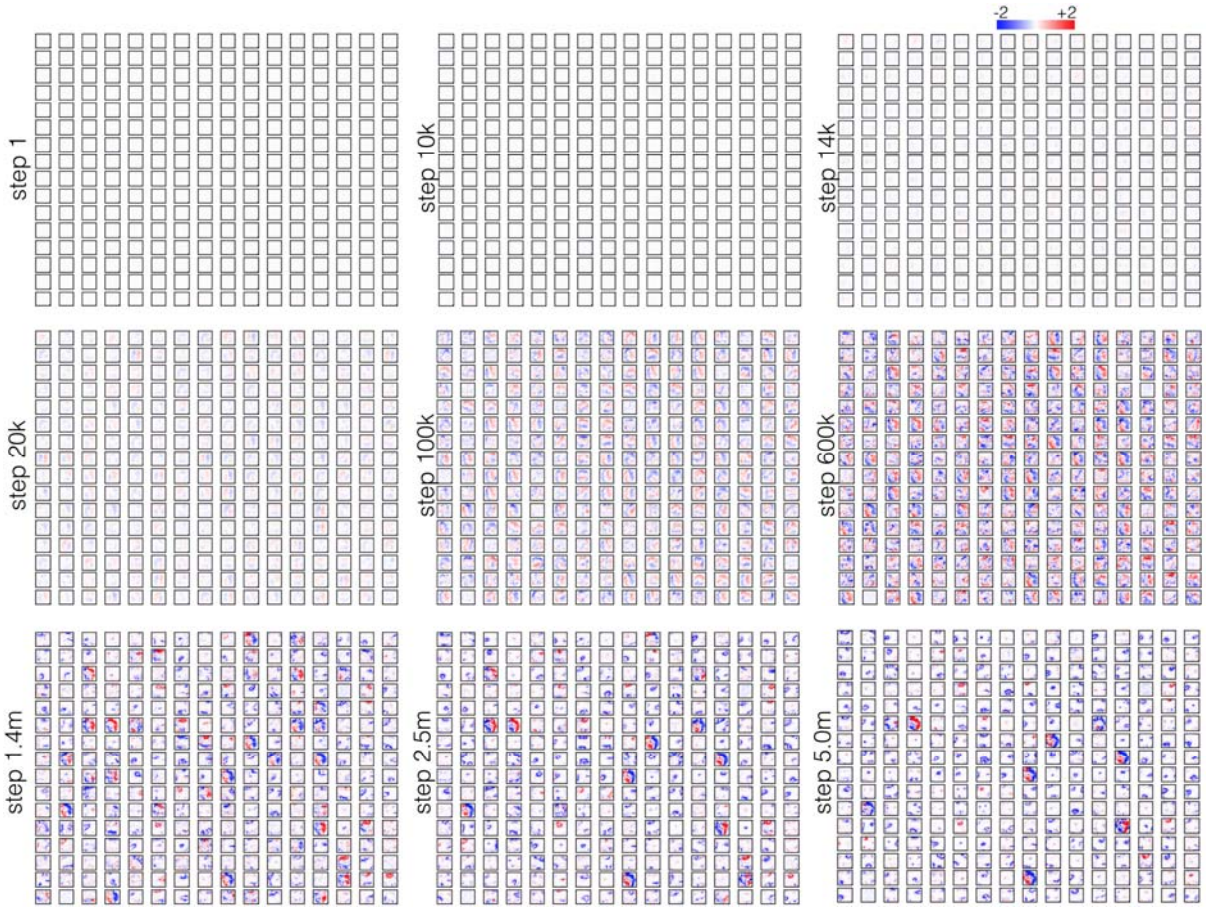
Learning and forgetting. Based on the (5million steps trained) RRN network, we first trained it with additional 10,000 steps of novel symbol x images (with above mentioned random affine transformation to diversify the training set), and then trained it with another 10,000 steps of known digit images. For symbol x recognition, the former was a learning process, and the latter was a forgetting process. The reciprocal of mean square error (MSE) between original and reconstructed images was used as a measure of performance.



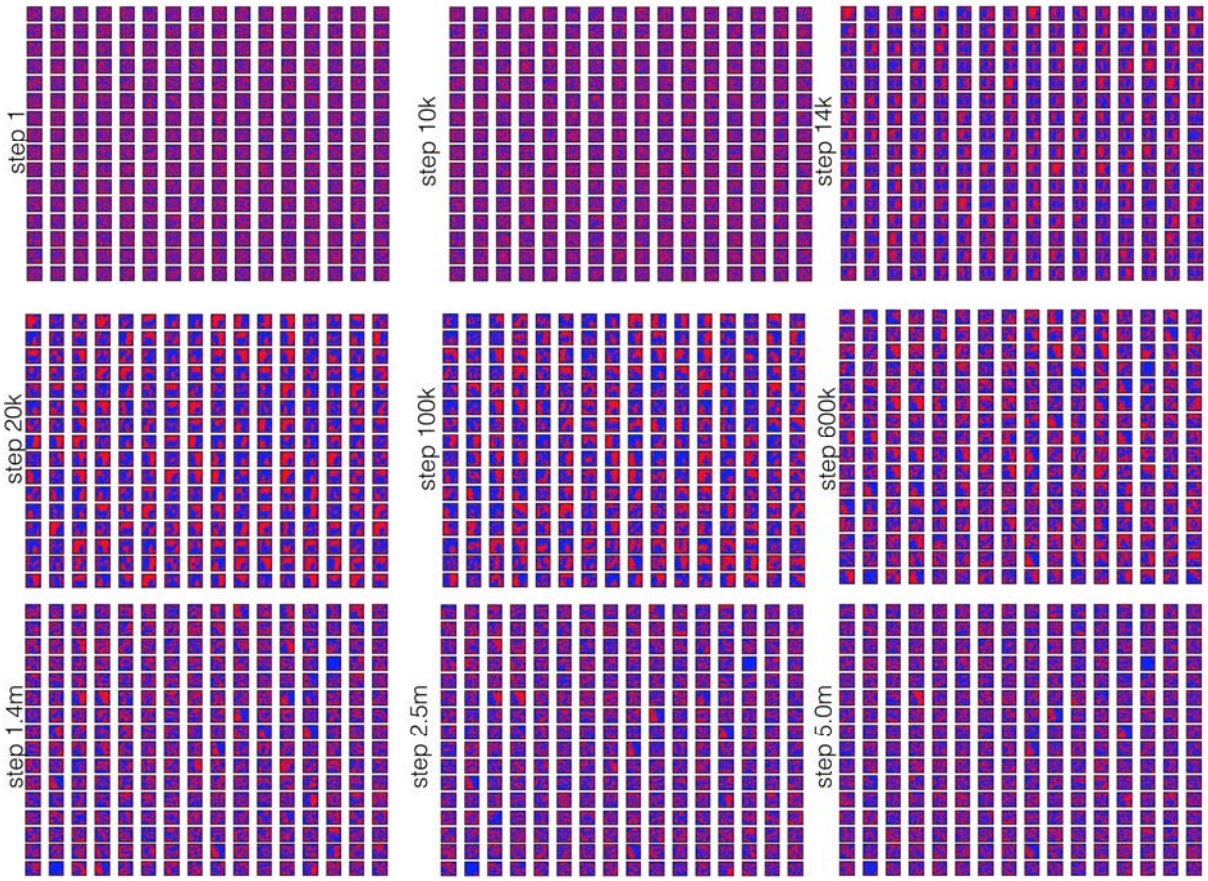
Extended Data Fig. 1 | The RRN architecture. It is a hierarchical modular network containing recognition and reconstruction components, each comprising several layers of homogeneous units with layer name and number of units labeled. Images are fed into retina (28*28), followed by a serial feedforward processing throughout the whole network till layer dd, where reconstructed image is compared with original image to formed the mean square error (MSE), then the synaptic weights is updated via error back propagation process. By taking population responses of the Encoding layer as input, a linear decoder could be used to predict digit properties such as x, y, r and s, and a logistic regression decoder could be used to predict digit identity.

stage	steps	loss	avg(W)	avg(|W|)	avg(d_W)	#(W+)	avg(W+)	#(W-)	avg(W-)
0	0	0.2234	-1.9E+05	0.0380	0.0000	100427	0.0379	100277	-0.0380
1	10k	0.0195	-5.7E+05	0.0381	4.0E-05	100338	0.0380	100366	-0.0381
2	14k	0.0179	-0.0073	0.0479	-0.0033	93034	0.0437	107670	-0.0514
3	20k	0.0182	-0.0121	0.0792	-0.0012	91865	0.0733	108839	-0.0842
4	100k	0.0154	-0.0371	0.1987	-0.0004	89742	0.1807	110962	-0.2132
5	600k	0.0102	-0.0587	0.3744	-7.7E-05	93293	0.3399	107501	-0.4043
6	1.4m	0.0057	-0.0725	0.2754	3.2E-05	91328	0.2229	109376	-0.3192
7	2.5m	0.0041	-0.0941	0.2695	4.2E-04	88642	0.1984	112062	-0.3256
8	5.0m	0.0027	-0.1118	0.2714	-3.2E-05	87346	0.1834	113358	-0.3393

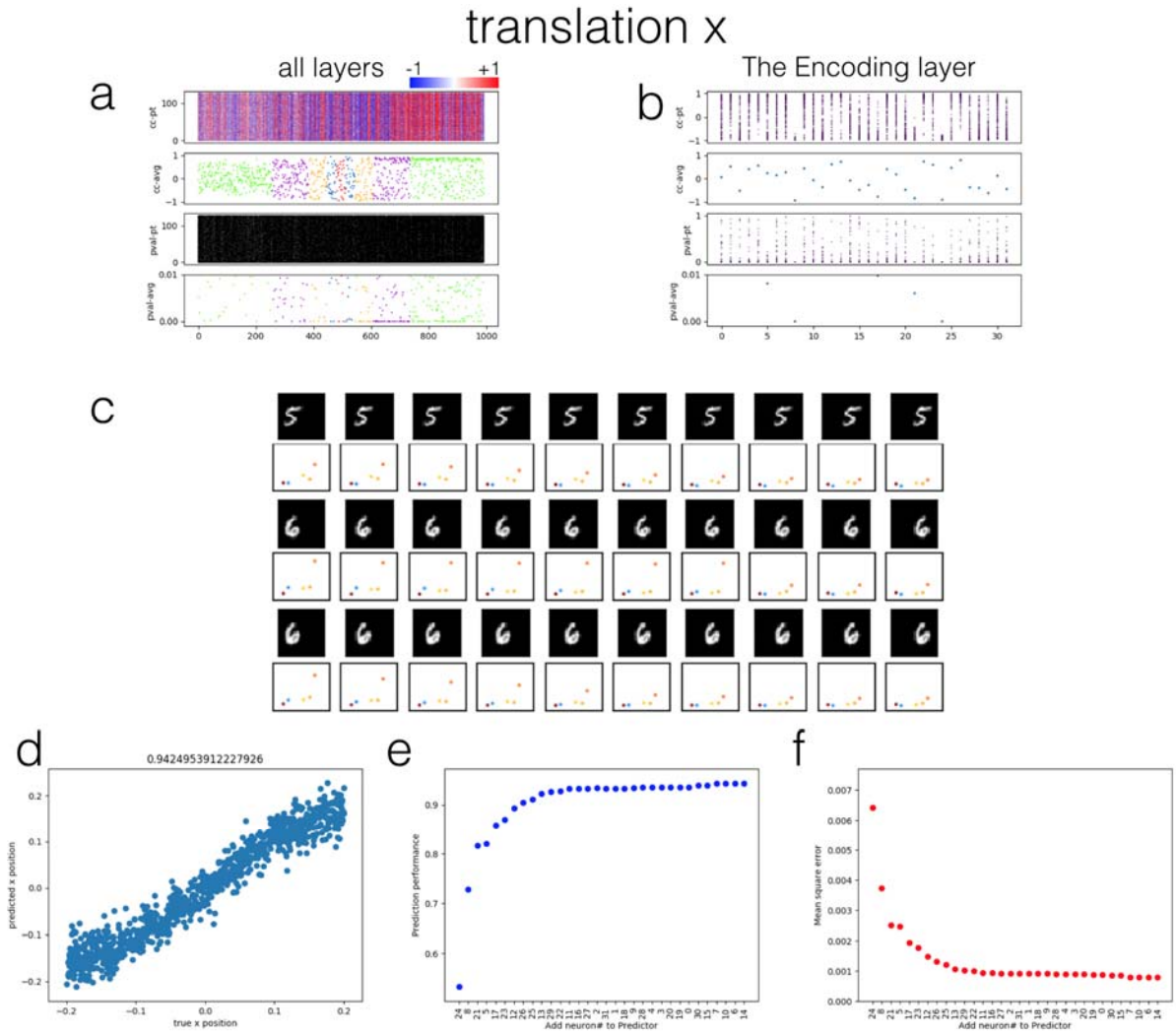
Extended Data Table 1 | parameters alteration across 9 developmental stages. In each column, the table presents the stage, step, value of loss function, average of connectivity strength, average of absolute connectivity strength, average of connectivity changes, number of positive (excitatory) connectivity, average of positive connectivity strength, number of negative (inhibitive) connectivity, and average of negative connectivity strength. Note, k represents 1,000 and m represents 1,000,000.



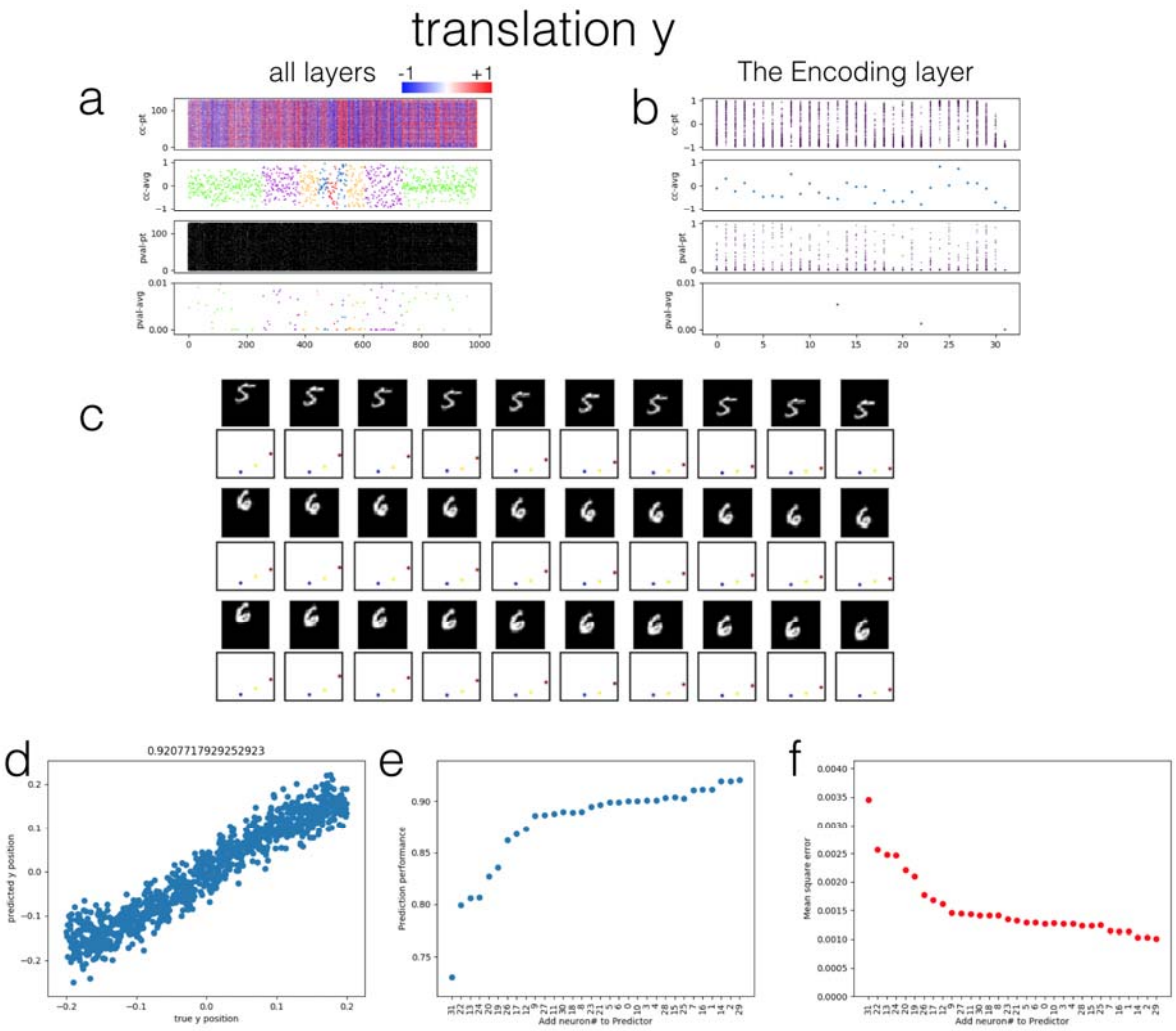
Extended Data Fig.2 | synapse weight evolution over 9 critical time points (ctp). Here we present the synaptic connection pattern from retina to the first layer, because only these synapses have retinotopic patterns. For each ctp we have 16*16 blocks (each block represents a neuron in the first layer), and each block contains 28*28 pixel (each pixel represents a neuron in retina). Each connectivity strength is rendered by pixel color intensity, where red represents excitatory while blue represents inhibitive.



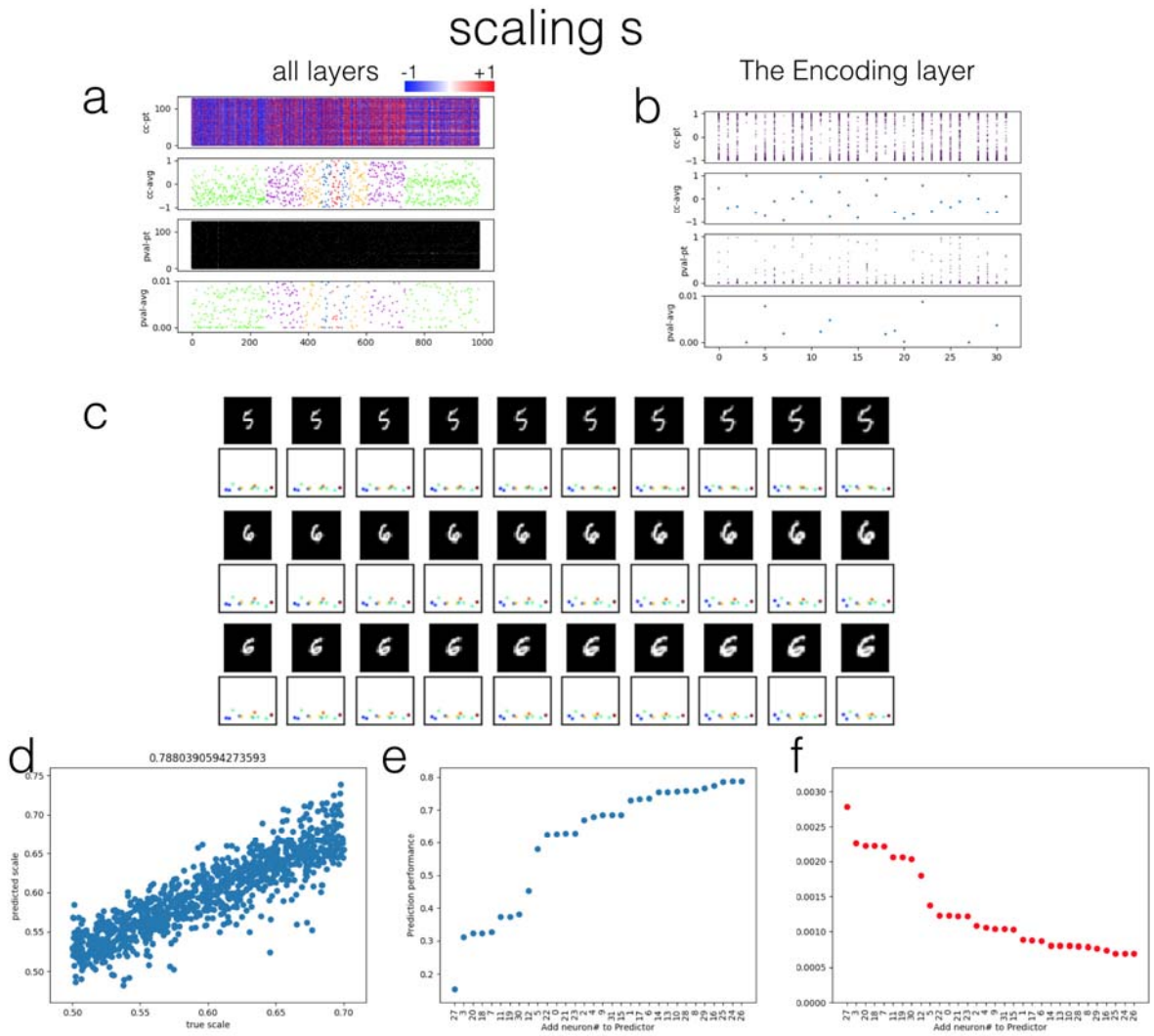
Extended Data Fig.3 | evolution of excitatory and inhibitive synapse over 9 ctps. Similar plots as Extended Data Fig.2. Here, we present a binary connectivity patterns, where we can see it evolved from random distributed pattern, to cluster-like distribution, finally strip-like distribution.



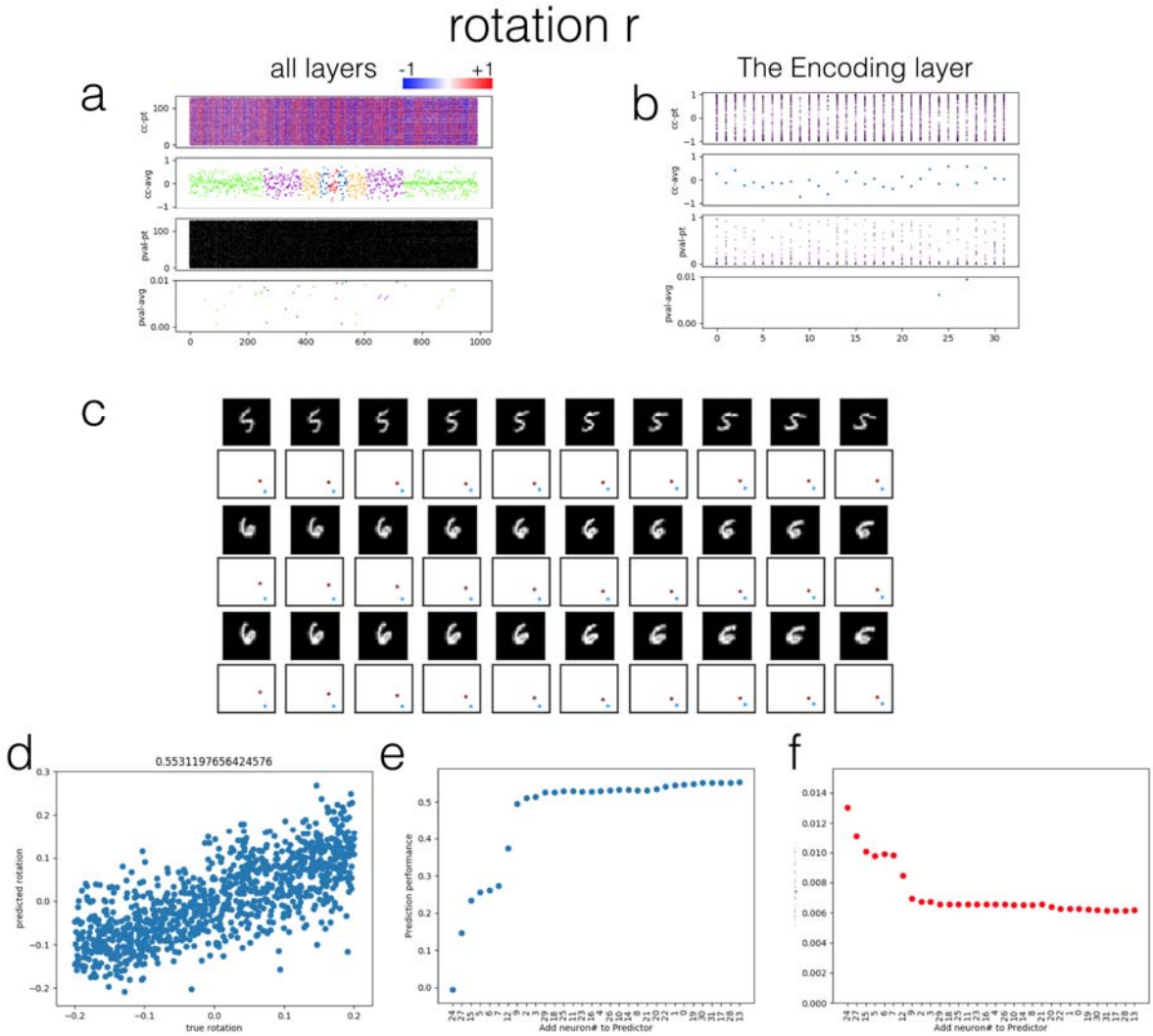
Extended Data Fig. 4 | study of translation x. a, Correlation studies between neuronal activity and x value for all neurons (x-axis). (Top panel) correlation coefficients for each trial (y-axis), with cc value rendered by bwr color map. (Second panel) averaged correlation coefficient for each neuron, with neuron layers rendered by colors. (Third panel) significant ($P < 0.01$) correlation for 128 trials highlighted with white points. (Forth panel) significant neurons across all layers, with y-axis represents p-value in range (0, 0.01). **b,** Correlation studies for 32 neurons in the Encoding layer. (Top panel) correlation coefficient for 128 trials (each point represents cc value of one trial). Note, n8 and n24 had all 128 trial cc values close to -1. (Second panel) averaged cc value across 128 trials. (Third panel) p-value for 128 trials (each point represents p-value of one trial). Note, n8 and n24 had all trial p-values close to 0. (Fourth Panel) 5 significant (averaged $P < 0.01$) x-neurons of n5, n8, n17, n21 and n24. **c,** Neuronal activities vary with respect to x position of digits. **d,** Comparison between true x position and linearly predicted x position. R square value is highlighted on top. **e,** Prediction performance (in term of R square values) improved as more neuronal activity were added into predictor. Note, x-axis presents neurons ordered by its significance (small p-value on the left). **f,** Similar to **e,** but y-axis presents MSE.



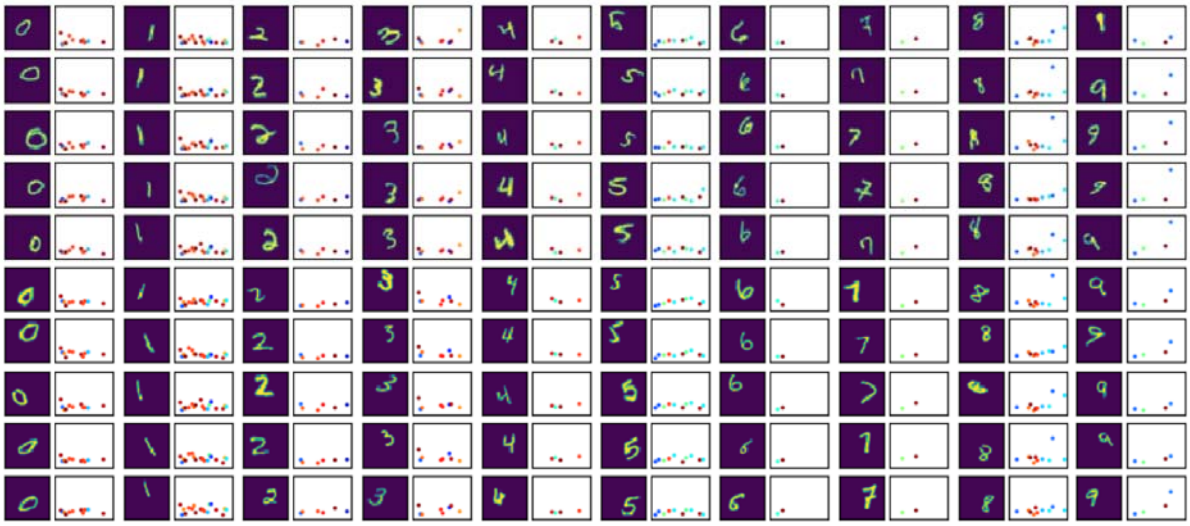
Extended Data Fig. 5 | study of translation y. a, Correlation studies between neuronal activity and y value for all neurons (x-axis). (Top panel) correlation coefficients for each trial (y-axis), with cc value rendered by bwr color map. (Second panel) averaged correlation coefficient for each neuron, with neuron layers rendered by colors. (Third panel) significant ($P < 0.01$) correlation for 128 trials highlighted with white points. (Forth panel) significant neurons across all layers, with y-axis represents p-value in range (0, 0.01). **b,** correlation studies for 32 neurons in the Encoding layer. (Top panel) correlation coefficient for 128 trials (each point represents cc value of one trial). Note, n31 had all 128 trial cc values close to -1. (Second panel) averaged cc value across 128 trials. (Third panel) p-value for 128 trials (each point represents p-value of one trial). Note, n31 had all trial p-values close to 0. (Fourth Panel) 3 significant (averaged $P < 0.01$) y-neurons of n13, n22, and n31. **c,** Neuronal activities vary with respect to y position of digits. **d,** Comparison between true y position and linearly predicted y position. R square value is highlighted on top. **e,** Prediction performance (in term of R square values) improved as more neuronal activity were added into predictor. Note, x-axis presents neurons ordered by its significance (small p-value on the left). **f,** Similar to **e,** but y-axis presents MSE.



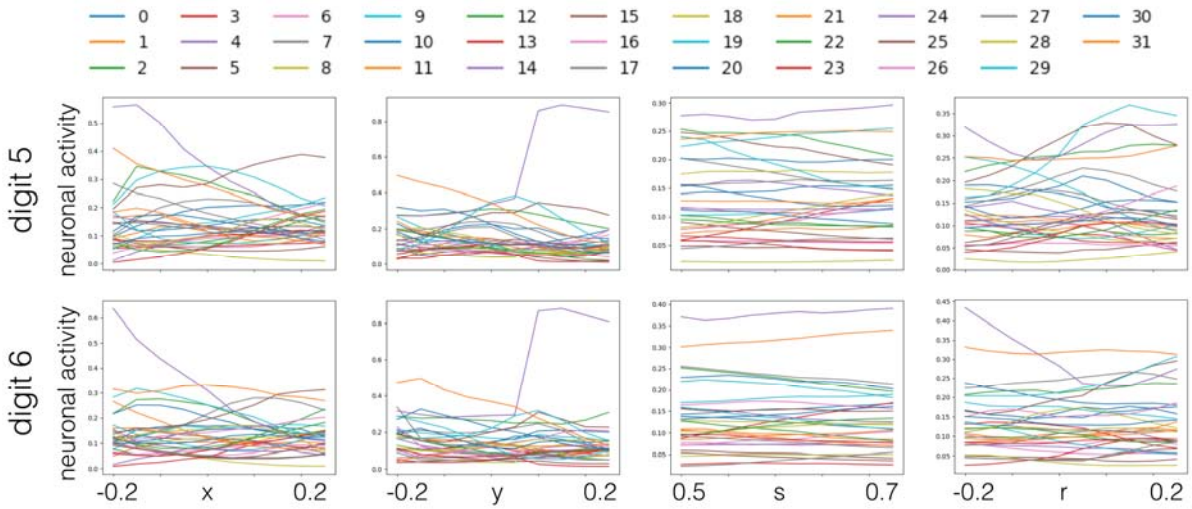
Extended Data Fig. 6 | study of scaling s. a, Correlation studies between neuronal activity and s value for all neurons (x-axis). (Top panel) correlation coefficients for each trial (y-axis), with cc value rendered by bwr color map. (Second panel) averaged correlation coefficient for each neuron, with neuron layers rendered by colors. (Third panel) significant ($P < 0.01$) correlation for 128 trials highlighted with white points. (Forth panel) significant neurons across all layers, with y-axis represents p-value in range (0, 0.01). **b,** correlation studies for 32 neurons in the Encoding layer. (Top panel) correlation coefficient for 128 trials (each point represents cc value of one trial). Note, n27 had all 128 trial cc values close to +1. (Second panel) averaged cc value across 128 trials. (Third panel) p-value for 128 trials (each point represents p-value of one trial). Note, n27 had all trial p-values close to 0. (Fourth Panel) 11 significant (averaged $P < 0.01$) s-neurons. **c,** Neuronal activities vary with respect to scaling s of digits. **d,** Comparison between true scaling s and linearly predicted scaling s. R square value is highlighted on top. **e,** Prediction performance (in term of R square values) improved as more neuronal activity were added into predictor. Note, x-axis presents neurons ordered by its significance (small p-value on the left). **f,** Similar to **e,** but y-axis presents MSE.



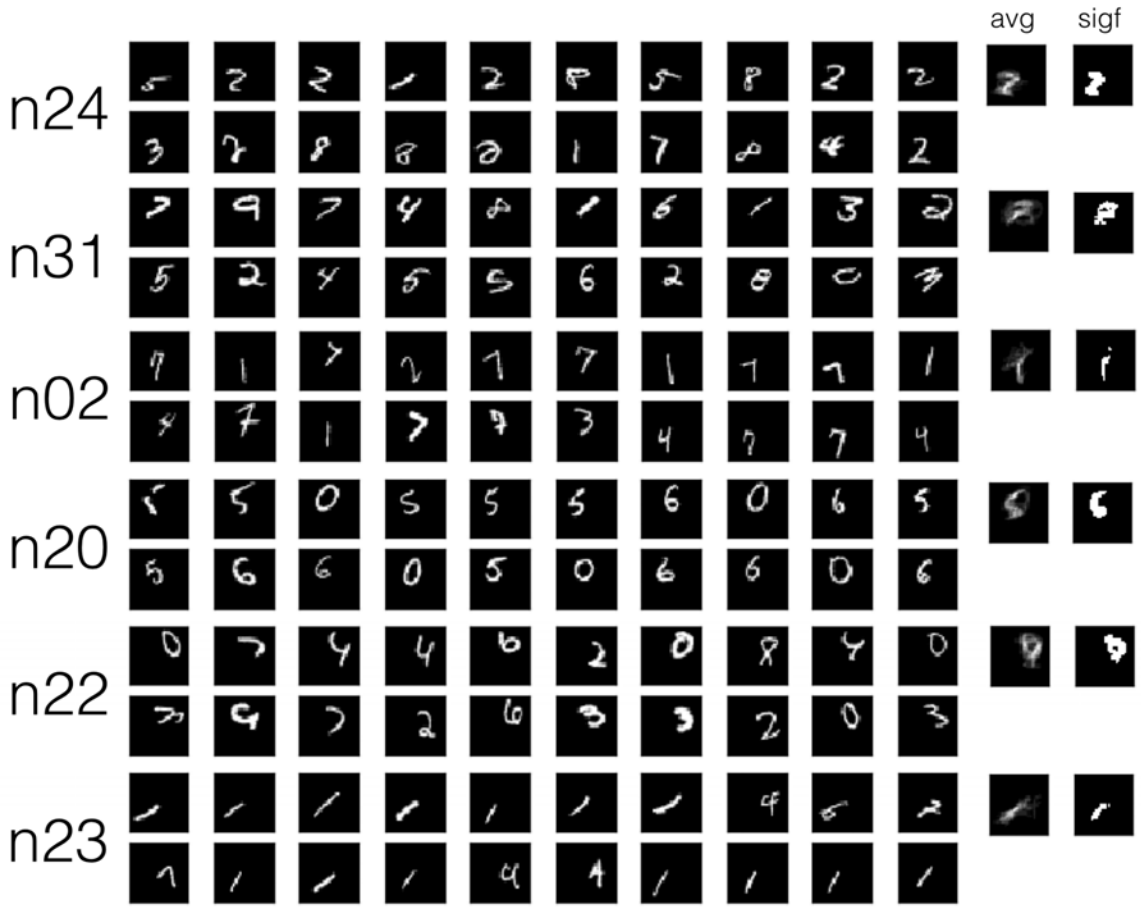
Extended Data Fig. 7 | study of rotation r. a, Correlation studies between neuronal activity and r value for all neurons (x-axis). (Top panel) correlation coefficients for each trial (y-axis), with cc value rendered by bwr color map. (Second panel) averaged correlation coefficient for each neuron, with neuron layers rendered by colors. (Third panel) significant ($P < 0.01$) correlation for 128 trials highlighted with white points. (Forth panel) significant neurons across all layers, with y-axis represents p-value in range (0, 0.01). **b,** correlation studies for 32 neurons in the Encoding layer. (Top panel) correlation coefficient for 128 trials (each point represents cc value of one trial). Note, we cannot find neuron with clustered trial cc values. (Second panel) averaged cc value across 128 trials. (Third panel) p-value for 128 trials (each point represents p-value of one trial). Note, n24 and n27 had relatively concentrated p-values close to 0 though several trials exception. (Fourth Panel) 2 significant (averaged $P < 0.01$) r-neurons of n24 and n27. **c,** Neuronal activities vary with respect to rotation r of digits. **d,** Comparison between true rotating r values and linearly predicted r values. R square value is highlighted on top. **e,** Prediction performance (in term of R square values) improved as more neuronal activity were added into predictor. Note, x-axis presents neurons ordered by its significance (small p-value on the left). **f,** Similar to **e,** but y-axis presents MSE.



Extended Data Fig. 8 | Digit instances and the neuronal activities of identity-related neurons. According to fig. 3a, for different digit identities there are different neurons (in the Encoding layer) whose activities were significantly ($P < 0.01$) correlated to the digit identities. We can clearly witness that constant response pattern of these neurons were generally kept for their identities over changes in morphologies or properties.



Extended Data Fig. 9 | Changes of neuronal activities according to alteration of properties. It demonstrates how 32 neurons (in the Encoding layer) adjust their activities in different viewing conditions of one digit 5 instance (top) and one digit 6 instance (bottom). x-axis represents the variation of properties of x, y, s and r in 4 columns, y-axis represents the neuronal activity strength, and color represents different neurons. We witness that population neurons need a relatively high magnitude of concerted adjustment of ensemble activities to handle position change, while relatively low adjustment for scaling change. The relationship between changes of properties and neuronal responses follows diffeomorphic mapping due to the continuity nature of the RRN network.



Extended Data Fig. 10 | Top-20, average and significant images for 6 neurons. For the first 6 strongest firing neurons (fig. 3p-r) including n24, n31, n02, n20, n22 and n23, we display their top-20 image instances (among 1024 instances) that elicited strongest neuronal activity for each neuron. The 'avg' image was averaged across the top 20 images, and the 'sigf' image contained pixels whose values (over these top-20 images) were significantly ($P < 0.01$) different from zero. These images show that these neuron fires in favor of specific features in specific retinal regions.