Recurrent neural networks that learn multi-step visual routines with reinforcement learning

Sami Mollard[1], Catherine Wacongne[2], Sander M. Bothe[3,4], Pieter R. Roelfsema[1,5,6]

Affiliations:
[1]Department of Vision & Cognition, Netherlands Institute for Neuroscience, Meibergdreef 47, 1105 BA, Amsterdam, The Netherlands.

[2]AnotherBrain, Paris, France. Work done while employed at the Netherlands Institute for Neuroscience

[3]Machine Learning Group, Centrum Wiskunde & Informatica, Science Park 123, 1098XG, Amsterdam, The Netherlands

[4]Swammerdam Institute for Life Sciences, University of Amsterdam, Science Park 904, 1090GE Amsterdam, Netherlands

[5]Department of Integrative Neurophysiology, Center for Neurogenomics and Cognitive Research, VU University, Amsterdam, The Netherlands.

[6]Department of Psychiatry, Academic Medical Center, Amsterdam, The Netherlands

Abstract

Many problems can be decomposed into series of subproblems that are solved sequentially. When the subproblems are solved, relevant intermediate results have to be stored and propagated from one subproblem to the next, until the overarching goal has been completed. The same applies for visual tasks that can be decomposed into sequences of elemental visual operations: experimental evidence suggests that for visual tasks, intermediate results are stored in working memory as an enhancement of neural activity in the visual cortex. The focus of enhanced activity is then available for subsequent subroutines to act upon. However, it remains unknown how those dynamics can emerge in neural networks that are trained with only rewards, as animals are.

Here, we propose a new recurrent architecture capable of solving composite visual tasks in a reinforcement learning context. We trained neural networks on three visual tasks for which electrophysiological recordings of monkeys' visual cortex are available and updated weights following RELEARNN, a biologically plausible four-factor Hebbian learning rule, which is local both in time and space. The networks learned an abstract, general rule and selected the appropriate sequence of elemental operations, solely based on the characteristics of the visual stimuli and the reward structure.

The activity of the units of the neural network resembled the activity of neurons in the visual cortex of monkeys solving the same tasks. Relevant information that needed to be exchanged between subroutines was maintained as a focus of enhanced activity and passed on to the subsequent subroutines. Our results demonstrate how a biologically plausible learning rule can train a recurrent neural network on multistep visual tasks.

Introduction

Many everyday tasks can be decomposed into subroutines that are executed sequentially. If you want to find your way on a map for instance, you first have to search where your goal is, where your actual position is, and then you have to trace a path between those two positions. In 1984 Ullman (Ullman, 1984) proposed that such complex visual tasks can be solved using visual routines, neuronal programs consisting of a sequence of atomic mental steps that he called elemental operations. Curve tracing, visual search, cuing, matching or contour grouping are examples of elemental operations (Roelfsema, 2005; Roelfsema et al., 2000; Zylberberg et al., 2011).

There are two critical aspects for the successful implementation of visual routines: the correct elemental operations must be chosen and applied at the correct time, and information must flow between subroutines so that the output of one subroutine provides the input for the next one. Those two conditions are straightforward to implement in a computer program but it remains unclear how variables are bound to subroutines in the brain.

Several studies provided insights into the neural basis of elemental operations and visual routines. (Jolicoeur et al., 1986) focused on the tracing elemental operation. Here, tracing refers to a mental operation: while participants (humans or monkeys) keep their eye on a fixation point, they have to report if two elements belong to the same curve or not. In the case of the curve tracing operation done by monkeys, the reporting is done by making an eye movement toward the end of the curve that is connected to the fixation point, while another curve (the distractor) is also present in the stimulus (Fig 1.A). (Jolicoeur et al., 1986) showed that the time it takes to trace a curve (one of the elemental operations hypothesized by Ullman) was proportional to the length of the curve to trace, indicating a serial process. Later, Roelfsema et al., 1998 recorded the activity of neurons in the visual cortex of monkeys

performing a curve tracing task. They found that neurons whose receptive field fell on the curve to be traced had their activity enhanced compared to neurons whose receptive field fell on distractor curves. Fig. 1C shows the activity of a neuron when its receptive field is on the target curve (orange curve) or the distractor curve (blue curve). After a delay of 130ms, the activity of the neuron is enhanced when it is on the target curve. Moreover, this enhancement appeared later for neurons further from the beginning of the curve, confirming the serial nature of the process in which the enhanced activity gradually spreads from the start of the curve towards its end (Pooresmaeili & Roelfsema, 2014). Going from simple elemental routines to visual routines, (Moro et al., 2010) and (Roelfsema et al., 2003) recorded neurons in the visual cortex of monkeys learning two visual routines composed of two elemental operation in different orders: a search-then-trace task and a trace-then-search task. In the search-then-trace task (Fig. 1D) the fixation point could be either one of two colours. The monkey had to trace the curve that was connected to a disk that had the same colour as the fixation point. This task hence consists in the composition of two elemental operations: first a search operation to identify the beginning of the curve to trace, and then a trace operation. In the trace-then-search task (Fig. 1G) the order of the two subroutines was inversed: the monkey first had to trace the curve connected to the fixation point, register the colour of the end of that curve and make an eye movement toward the disk that had the same colour. They showed that the outcome of an elemental operation was tagged by a focus of enhanced activity. For example, in the search-then-trace task, the beginning of the target curve, which is the outcome of the search operation, is enhanced first in the visual cortex (159ms Fig. 1F), and then the tracing operation can begin (229ms Fig. 1F). In the trace-then-search task, the outcome of the trace operation, the colour of the disk to search for is also enhanced first in the visual cortex (180ms Fig. 1I), before the search operation can take place (267ms Fig. 1I). This tagging process implemented by an enhancement of activity makes the outcome of an elemental operation available in working memory to be read as the input for the next elemental operation. The sequence of mental operations is associated with a sequence of attention shifts at a psychological level of description. Visual search for a particular target shape or target color in an array of stimuli leads to a shift of attention to the target item that needs to be found (Kim & Cave, 1995). Similarly, when subjects mentally trace a curve, they gradually spread attention across all of its contour elements (Houtkamp et al., 2003).

The dynamics of neural activity in the visual cortex during elemental operations and visual routines can be divided in several phases (Lamme & Roelfsema, 2000; Roelfsema et al., 2007). The first phase is dominated by feedforward processing. Basic features are registered in low level areas and more complex features, such as object category are registered in higher visual areas (Roelfsema, 2006; Vanrullen, 2007). The later phases are dominated by recurrent processes carried out by feedback and horizontal connections and are necessary for more complex visual tasks such as visual routines. During the later phases, neuronal activity is not only influenced by visual stimuli, but also by feedback from higher visual areas and horizontal influences from neurons in the same area. This recurrent processing phase also allows the visual cortex to act as a cognitive blackboard. Features that are extracted during the feedforward phase and that form a base representation can be enhanced or suppressed by higher areas according to their relevance to the task at hand, which thereby write to the blackboard. Other cortical areas can read these selection signals and act accordingly (Roelfsema & de Lange, 2016). Reading and writing to the cognitive blackboard in early visual areas permits the exchange of information between elemental operations that are part of a visual routine. For example, in the search-then-trace task, when the location of the marker with the same colour as the fixation point is found (end of the search operation), it is enhanced in the primary visual cortex, which corresponds to a write operation on the cognitive blackboard. This marks the beginning of the tracing operation, during which the target curve is iteratively labelled with enhanced activity. This illustrates how one

operation, in this case the search one, denotes a location of space as its output, which is maintained as enhanced neuronal activity. This location is then transferred to the next operation, in our example the tracing operation.

Hence, there is substantial evidence that visual routines are implemented in the visual cortex through the propagation of enhanced activity during a recurrent processing phase. However, it remains unknown what architectures and learning rules can learn such routines. To investigate the learning of elemental operations, (Brosch et al., 2015) developed RELEARNN, a biologically plausible learning rule inspired from the Almeida-Pineda algorithm (Almeida, 1987; Pineda, 1987) and used it to train a recurrent neural network on a contour grouping task and a curve tracing task. Interestingly, the strategy that the network learnt using only reward was similar to what was observed in monkey's visual cortex. The target curve was tagged in an iterative manner by the spreading of enhanced neuronal activity. However, they trained their network only on individual elemental operations and it remains unclear if neural networks can learn to stitch elemental visual operations into visual routines using a biologically plausible learning rule. To examine learning of visual routines, we tested neural networks with feedforward, feedback and lateral connections. We asked whether the network (1) would learn multiple elemental operations, (2) whether it would learn to execute the elemental operations sequentially and in the correct order and (3) how the network ensured the transfer of information of one elemental operations to the next.

We trained the networks on three visual routines for which electrophysiological recording of monkeys performing the same routines exist: a trace task, a search-then-trace and a trace-then-search task (Moro et al., 2010). We used RELEARNN (Brosch et al., 2015), which is a biologically plausible learning rule that supplies synapses with the information, local both in time and space, to change strength. We found that the networks were able to learn all three tasks. Interestingly, the activation of units of the networks was similar to that of neurons in the visual cortex of monkeys. The networks learnt to propagate enhanced activity along curves and to enhance activity in retinotopic areas to propagate information between subroutines. Our model shows how a recurrent neural network trained with a biologically plausible learning rule can learn to propagate enhanced activity to solve complex visual tasks.

The tasks

We trained networks on three tasks for which electrophysiological recording of monkeys are available (Moro et al., 2010; Roelfsema et al., 2003). The first task is a curve tracing task (Fig 1A,B). In this task, the monkey had to locate the end of the curve that was connected to a fixation point and make an eye-movement to this position when the fixation point disappeared. This task requires only one of the elemental operations, which is curve tracing (Ullman, 1984). To solve it, an agent must learn to group elements connected to each other. In the version of the task that monkeys performed, they started the trial by directing gaze to a fixation point (red circle in Fig. 1A). They then mentally traced from the fixation point to the blue circle at the other end, which was the target for an eye movement. In the version tailored for the neural network, we used a grid that was 15 by 15 pixels and a depth of four features, one for each colour the pixels could take (red, orange, blue and green). We trained the model to start tracing at the red pixel and to report the blue pixel at the other end of the curve, defined by a set of connected squares, by selecting this location for an eye movement response.

The second task was a search-then-trace task (Fig. 1D,E). Here the monkey had to first register the colour of the fixation point (orange in Fig. 1D). It then had to search for a circular marker with the same colour as the fixation point, which demarcated the start of the target curve. The monkey then traced the target curve to its other end and planned an eye

movement to a circle at that position. Again, the search and trace operations were performed mentally, while the monkey kept its gaze on the fixation point. The eye movement was made only at the end of the trial. In the version of the search-then-trace task for the model, we provided the colour that was the target of search by activating an additional input unit, which could be either yellow or red (Fig. 1D). The model had to locate the pixel with the same colour in the array and trace the curve starting at this pixel to its other end to locate a blue pixel that was the target for eye movement.

The third task was the trace-then-search task (Fig 1G,H). The monkey first traced a curve connected to the fixation point to identify the colour of a marker at the end of the curve. It then searched for another circle with the same colour, which was the target of an eye movement. In the version of the task that was used for the model, the target curve started with a blue pixel. The model had to identify the colour at the other end of the target curve and to select the same colour at a location outside the pixel frame. Specifically, it could choose between a red and yellow response pixel shown to the left of Fig. 1H.

We note that the search-then-trace and the trace-then-search tasks require visual routines, involving the sequential application of visual search and curve tracing as elemental operations, in different orders. It can also be seen how the output of the first operation has to be carried over as input of the next operation. For example, the output of the search operation of the search-then-trace task is the location where the marker with the right colour is found. This location is the starting point for the subsequent tracing operation.

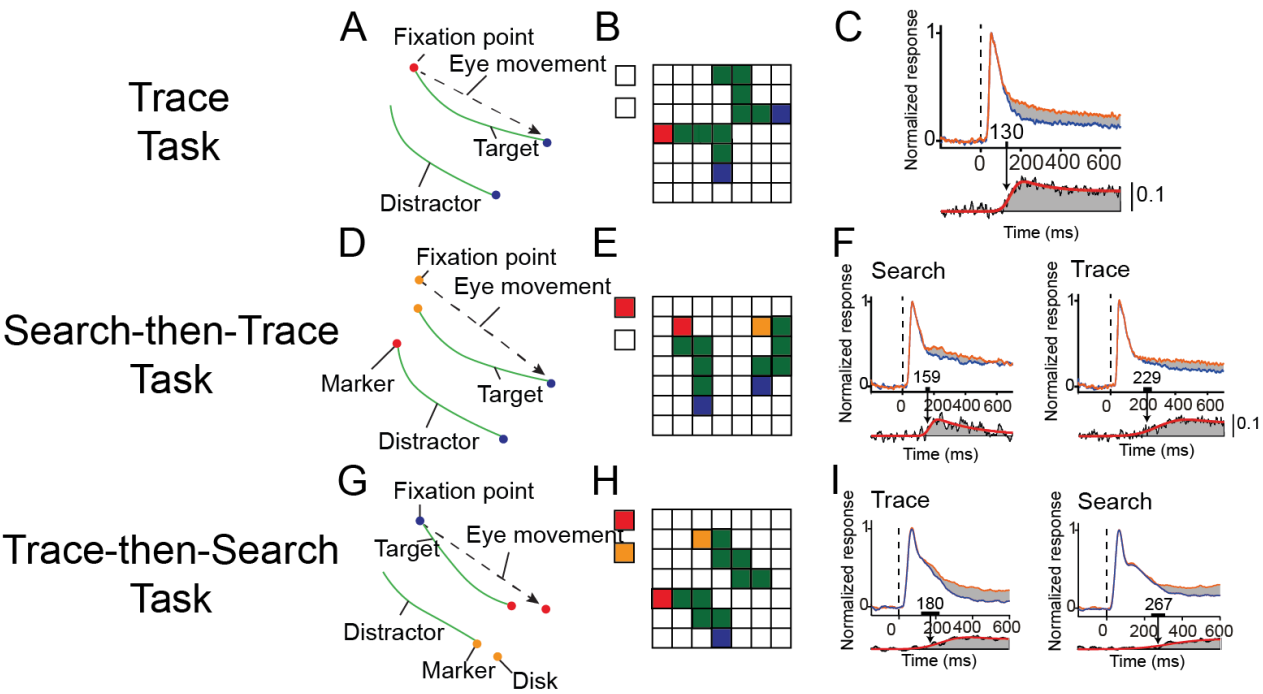


Fig 1. Example stimuli presented to the monkeys (left) and to the network (middle) and electrophysiological recordings from monkeys' visual cortex. A,B,C Trace task. In this task, the goal of the agent is to make an eye movement toward the blue dot connected to the red fixation point. Recordings from monkeys' visual cortex show that the target curve is identified by a tag corresponding to an enhancement of activity of neurons whose receptive field falls on the target curve (C, orange), compared to neurons whose receptive field falls on the distractor curve (C, blue). **D,E,F** Search-then-trace task. Here, the agent has to make a saccade toward the blue dot connected to the curve whose beginning is the same colour as the fixation point. This task can be broken down into two subtasks. First, a search operation to identify the beginning of the correct curve, and then the tracing of this curve. In the artificial stimuli, the fixation point can be either one of the two pixels outside of the grid. To perform those two subroutines and transfer information between them, a tag of focused attention labels the output of the search operation as enhanced activity (F, search) after ~159ms. Then, the tracing operation begins from there, with the tag propagating sequentially along the curve. **G,H,I** Trace-then-search task. In this task, the search and the trace subroutines are inverted: first the agent has to trace the curve connected to the fixation point, then make a saccade toward the marker that has the same colour as the end of the traced curve. In the artificial stimulus, the two coloured disks that can be the target of the eye movement are represented by the two pixels outside of the grid. In the visual cortex, the label of enhanced activity first propagates through the curve, then on the coloured marker to perform the search subroutine. Panels C, F and I adapted from (Moro et al., 2010)

Model

We will first describe the network architecture and the biologically plausible reinforcement learning rule. We will then compare the network simulations with neuronal activity in the visual cortex of monkeys that were trained on the same tasks.

Structure of the network

We trained convolutional neural networks with an input layer, two hidden layers and an output layer. In each layer there was a group of feedforward units that only propagated information to the next layer and to neurons with the same receptive field and a recurrent group that propagated information to the next or previous layer and to units with different receptive fields. As a result, neurons in the feedback group could be modulated by activity outside the pixel that defined their receptive field. However, the activity of neurons in the feedback group was gated by neurons in the feedforward group that had the same receptive field (Fig. 2) so that this modulatory effect could not occur if the feedforward unit with the same receptive field was not active.

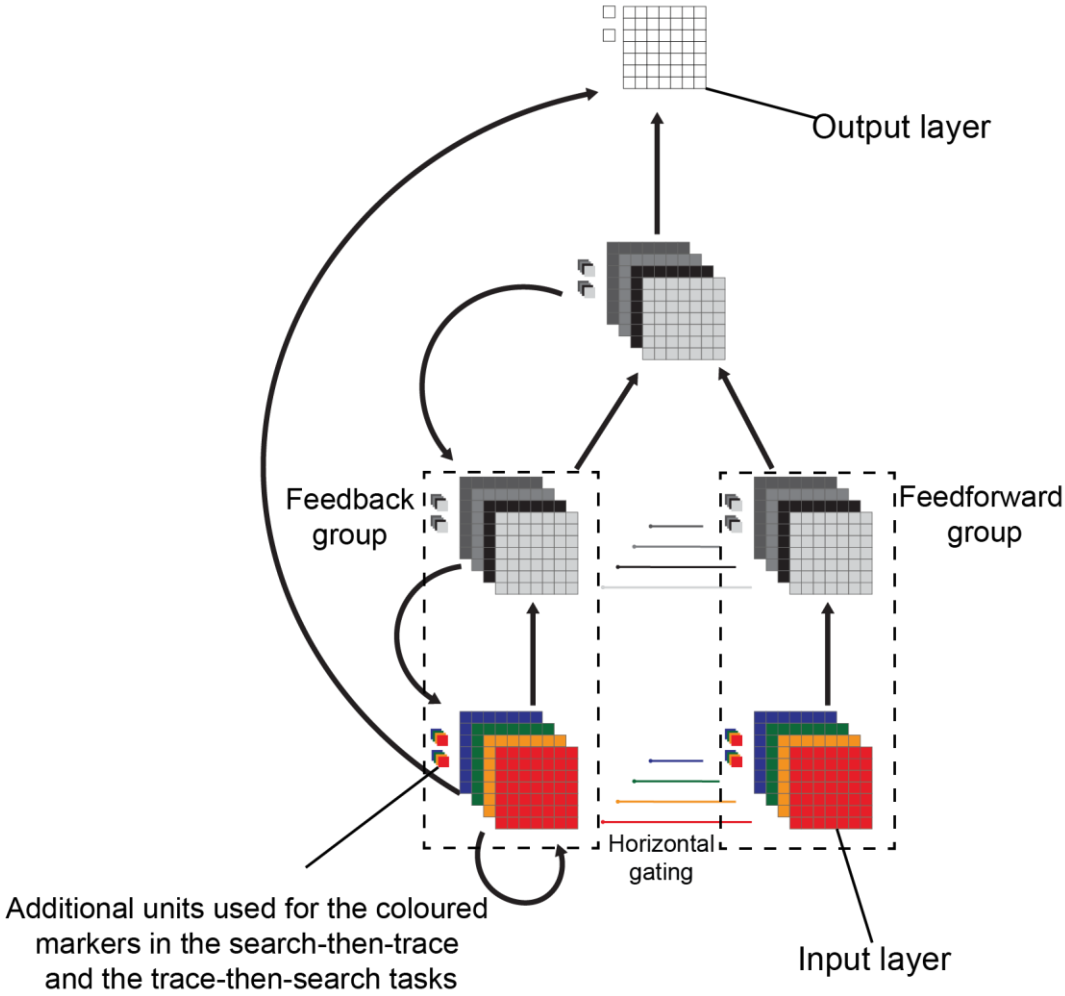


Fig 1. Architecture of the network. The network comprises one input layer, two hidden layers and one output layer. The input and hidden layers have 4 features each. In each layer, neurons can belong either to the feedforward or to the feedback group. Neurons in the feedback group are gated horizontally by neurons in the feedforward group.

The activities of neurons in the feedforward group in layer l at timestep t were determined by activities of feedforward neurons in the lower layer at the same timestep. If X represents the activity of neurons in the feedforward group, Y the activity of neurons in the feedback group, l the layer index, 0 being the input layer, then the activity of neurons was determined by:

$$X^l(t) = \sigma\left(V \otimes X^{l-1}(t)\right)$$

$$\mathbf{Y}^{\mathbf{l}}(t) = \begin{cases} \sigma\left(\phi(\mathbf{X}^{\mathbf{l}}(t)) \odot (U \otimes \mathbf{Y}^{\mathbf{l+1}}(t-1) + H \otimes \mathbf{Y}^{\mathbf{0}}(\mathbf{t}-\mathbf{1}))\right), & \text{if } l = 0 \\ \sigma\left(\phi(\mathbf{X}^{\mathbf{l}}(t)) \odot (U \otimes \mathbf{Y}^{\mathbf{l+1}}(t-1) + T \otimes \mathbf{Y}^{\mathbf{1-1}}(t))\right), & \text{otherwise} \end{cases}$$

Here σ is a nonlinear activation function and φ a gating function (see bellow), ⊙ represents an element-wise product between feedback and feedforward elements and ⊗ indicates convolution between weights and neurons. For connections between hidden layers, we used 3x3 convolution kernel, and enforced that neurons could only make connections to their Von Neumann neighbourhood (4- neighbourhood). The last hidden layer and the output layer were fully connected with weights linking two neurons with the same receptive field initialized with positive values, whereas weights linking neurons with different receptive fields initialized with negative values. This ensured a winner-takes-all mechanism between the last layers that stabilized learning.
The activity of neurons in the output layer was determined as:

$$Q = W \otimes X^3(T)$$

Where T is the timestep at which the network reaches a stable state.

The output layer is retinopically organised, each neuron is associated with one pixel of the input grid and represents the expected reward if the networks select this pixel to make an eye movement, also called Q-value (Sutton & Barto, 2018).

During training, with probability $1 - \epsilon$ the action with the highest Q-value was chosen, which meant that the corresponding pixel was selected for an eye movement. With probability ϵ the network explored other actions. In this case a random action was sampled from the the Boltzmann distribution P_B:

$$P_B(a) = \frac{exp(q_a)}{\sum_k exp(q_k)}$$

ϵ was set to 0.05 across tasks and networks.
We trained the networks with RELEARNN (Brosch et al., 2015), which is a biologically plausible implementation of Q-learning that can be broken down in three phases. Upon presentation of the stimulus, the activity of the input neurons remains constant and activity propagates through the recurrent connections of the network. If the activity of neurons between two consecutive timesteps was constant, we considered that a stable state was reached and the action was selected, but we did not wait longer than 50 timesteps if the activity was not yet stable. When an action is selected, an “attentional” signal originating from the winning action is propagated through an accessory network to determine the influence of each neuron on the selected action. The network then gets a reward r from the environment and computes a reward prediction error *d* that is determined by:

$$\delta = r - Q_a$$

Where Q_a is the state-action value of the selected action.
This reward prediction error *d* is broadcasted in the whole network by a neuromodulatory signal, such as dopamine (Schultz et al., 1997) and is thus available at all synapses. *d* and the "attentional" feedback signal are then combined to update the weights of the network. See bellow for a complete description of the learning rule.

Activation and gating functions

A popular activation function to train recurrent neural networks is the ReLU function because it alleviates the problem of the vanishing gradient (Glorot et al., 2011). However, ReLU functions offer no guarantees that the network will reach a stable state (Simard et al., 1989), which is a necessary condition for learning rules such as RELEARNN. On the other hand, squashing nonlinearities like sigmoid or tanh functions guarantee that the network reaches a stable state (Liao et al., 2018), but may suffer from the vanishing gradient problem, which might hurt performance (Linsley et al., 2020). To guarantee that our network reaches a stable state while mitigating the vanishing gradient problem, we used a ReLU-like function that only saturates after a threshold:

$$\sigma(x) = \begin{cases} max(0, x), & \text{if } x \leq 6 \\ log(1.5 \cdot x + 1) + c, & \text{otherwise} \end{cases}$$

Where c is set to $6 - log(10)$ to ensure that *s* is continuous. We found empirically that if the activation function had a finite limit toward infinity the tasks couldn't be properly learnt: for longer curves all the pixels would eventually reach the limit and the network couldn't properly differentiate between the target and the distractor curve. To prevent this we used a log function in the activation function to ensure that even for longer curves, the network would still be able to differentiate between the two curves.

We also used a non-linearity for the horizontal gating between the feedforward and the feedback group. As we want the feedforward group to encode the base-representation only, with no information about the feature or the position of the corresponding pixel, the activity of neurons of the feedforward group was passed through a continuous step-function before gating activity of neurons in the feedback group. The gating function was defined as:

$$\phi(x) = \frac{100.x}{\sqrt{1 + (100.x)^2}}$$

RELEARNN

One of the most popular algorithms to train recurrent neural network is backpropagation trough time (Werbos, 1988, 1990). During the forward pass, the consecutive states of each neuron are kept in memory and are used to update the weights during the feedback pass. This algorithm is non-biological because it is neither local in time nor in space (Lillicrap & Santoro, 2019): the information needed to update synapses is not available at the relevant synapses and each neuron would need access to the history of previous activation levels. We therefore used RELEARNN (Brosch et al., 2015), which is a learning rule that is local in time and in space and equivalent to backpropagation (Almeida, 1987; Pineda, 1987). RELEARNN uses a separate "attentional" network with accessory units, one unit for every

regular network unit (Fig. 3). The accessory units represent the influence of the regular units on the action that was chosen by the network. Activation in the accessory network starts at the unit of the output layer that represents the chosen action and the activity is then propagated through the rest of the accessory network. The strength of a connection from unit A to unit B in the accessory network is proportional to the strength of the connection from unit B to unit A in the regular network and the connections in the accessory network (dotted arrows in Fig.3) corresponds to the transposed and linearized connections in the main network (plain arrows in Fig. 3, see Brosch et al., 2015 for details). When the accessory network has converged to a stable state (Fig. 3, third phase), the activity of accessory units is proportional to the influence of the corresponding regular units on the chosen action value: accessory units that correspond to regular units with a strong influence on the Q-value of the chosen action are active and those that correspond to regular units that did not influence this Q-value are silent. The network of accessory units can thereby assign credit (or blame) to the regular units that are responsible for the outcome of the action. Changing the synapses between the regular units with a strongly active accessory unit is the best way to influence the Q-value of the chosen action on later trials.

It is useful to distinguish between three phases on each trial (Fig 3). In the first phase, the stimulus is presented and input neurons are activated by it. Activity propagates through the network of regular units for several timesteps until it converges to a stable state. At the end of this phase a presynaptic neuron *i* and a postsynaptic neuron *j* will have activity x_i^{∞} and y_j^{∞}. In the second phase, the accessory network is activated. Activity in this network is initiated by the selected action and propagates from there eventually reaching a stable state although in our experience stopping this second phase after an arbitrary fixed number of timesteps suffices (Liao et al., 2018). At the end of the second phase, neurons *i* and *j* in the accessory network have activity x_i^{acc} and y_j^{acc}, which is proportional to the influence of the corresponding regular units on the chosen action value (see Brosch et al., 2015 for a proof):

$$x_i^{acc} \propto \frac{\partial Q_a}{\partial x_i^{\infty}}$$

In the third phase, the network receives a reward if the correct action was selected, and no reward in case of an erroneous response. The reward prediction error is broadcasted trough the network using a neuromodulator and weights are modified according to a four-factor Hebbian learning rule:

$$\Delta w_{ij} = \delta \cdot x_i^{\infty} \cdot y_j^{acc} \cdot \sigma'\left(y_j^{\infty}\right)$$

As can be seen in the weight update, RELEARNN is local in space, as weights are modified using only pre and postsynaptic information, and local in time, as it's not necessary to keep a history of all the subsequent activities of a neuron, only the activity at the last timestep.

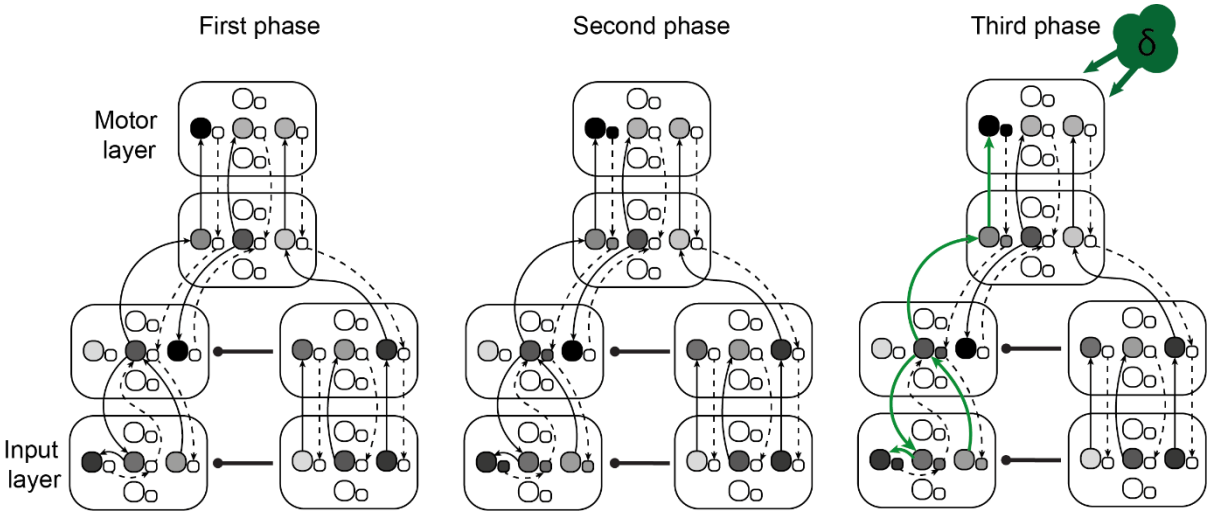


Fig 3. RELEARNN in our architecture. In the first phase, activity propagates in the main network (medium squares and plain arrows) until it reaches a stable state. In the second phase, a winning unit is selected (black square in the motor layer) and current is injected in the corresponding unit in the accessory network (small black square in the motor layer). From there, activity propagates in the accessory network (small squares and dotted arrows) to tag the connections that were responsible for the outcome in the first phase. After a few timesteps, activity in the accessory unit x_i^{acc} is proportional to the influence of the activity of the corresponding regular unit x_i^{∞} on the chosen output unit Q_a. In the third phase, a reward is administered or not depending on the outcome, a neuromodulator globally available broadcast the reward prediction error to the network and weights are changed according to a four-factors Hebbian learning rule.

Results

Curve-tracing task

We trained 24 networks on the curve tracing task and considered the task learnt when a network achieved at least 85% accuracy for 9 pixels long curves. 100% of the networks learnt the task in an average of 19 000 trials. We also trained networks on the trace task with no weight sharing. Again, 100% of the networks learnt the task, but in an average of 140000 trials. Given the much bigger computational burden associated with training networks with no weight sharing, we decided to use weight sharing to train the networks for the three visual routines.

To prevent the networks from getting stuck in a local minimum, we trained them with a curriculum, a strategy that was also used to train monkeys. To train the networks, we first presented a stimulus of one blue pixel and one red pixel randomly placed on the grid. This first step of the curriculum ensured that the network learnt quickly and reliably to select blue pixels only. If the network achieved 85% accuracy during a 500 trials test phase with weight fixed and no exploration, we presented it with a target and a distractor curves that were three pixels each. If the network achieved 85% accuracy during the test phase, we added one pixel to the curves and repeated the operation until the curves were 9 pixels long.

The curves were drawn randomly at each trial (see Fig. 4A for a few examples of stimuli shown). Since the number of curves grows exponentially with their length, at the end of the training the number of curves possible vastly exceeds the number of trials needed to learn the task and there is little chance that the network could memorize all the inputs presented. Instead, the networks likely learn an abstract, general rule that consists in applying elemental operations in different orders irrelevant of the characteristics of the stimuli shown like the shape of the curves. To draw on the computer program analogy, the characteristics of the stimuli, such as the length of the curves, are parameters, and the networks learn an abstract solution that can be applied to any set of parameters. To disambiguate between the two possibilities, we tested networks during the course of training on curve longer than what they had been trained on: when a network achieved at least 85% percent accuracy for a curve of

length N, before resuming learning, we tested the network without exploration on curves with lengths ranging from N+1 to N+4. At the end of training, networks could perform with above chance accuracy even for curves that were 4 pixels longer than what the network was trained on (Fig. 4B, red curve, p<10e-6, at the end of training for 13 pixels long curves, Wilcoxon signed-rank test). This generalization ability was not present at the beginning of training but was learnt during the course of training, indicating that networks progressively learn to apply an abstract rule that can be used with any stimulus (Roelfsema et al., 1999).

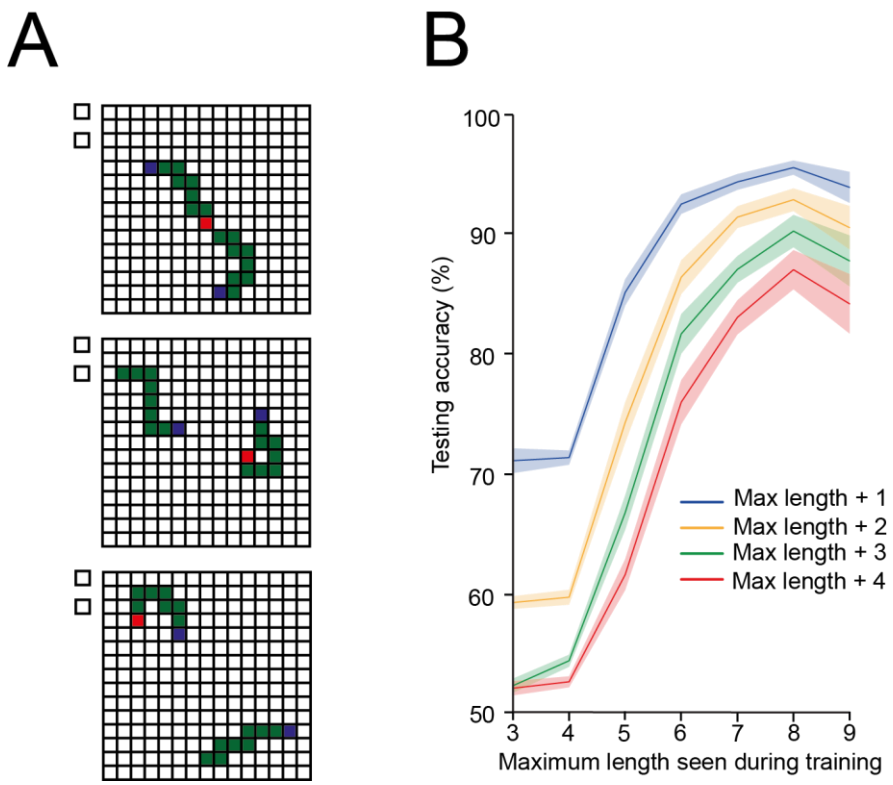


Fig 4. Generalization accuracy. A. Example stimuli that were presented to the network for the trace task. The number of curves possible is too big for the network to learn a one-to-one input-output mapping for every possible curve. **B**. Testing accuracy for curves of length up to N+4 pixels where N is the maximum length that has been seen during training. At the beginning of training, the agent cannot generalise to longer curve. At the end of training, even though the agent has been trained on curves up to 9 pixels long, it could solve the tasks with curves up to 13 pixels with above chance accuracy. The agent has learnt an abstract, general rule that can be applied irrespectively of the parameters used (p<10e-6, at the end of training for 13 pixels long curves, Wilcoxon signed-rank test).

Monkeys were trained on a curve tracing task similar to the one used in the artificial networks. They first had to keep their gaze on a fixation point. After 300ms two curves ending in blue circles appeared on the screen. One of the curves was connected to the fixation point. After an additional 600ms the fixation point disappeared from the screen and the monkey had to make an eye movement toward the end of the curve connected to the fixation point to get some juice as reward. The input presented to the artificial network corresponds to the 600ms where curves and fixation point were on the screen. When the monkeys had learnt the task, 40-50 multiunit electrodes were chronically implemented in the visual cortex to record activity of V1 neurons during the task (Roelfsema et al., 1998).

We first compared the activity of networks units whose receptive field fell on the target curve to the activity of units whose receptive field fell on the distractor curve. To quantify the magnitude of the response enhancement we defined a modulation index (Roelfsema et al., 1998s) as the difference between the response evoked by the target and distractor curve divided by the average response:

$$MI = \frac{Target - Distractor}{\frac{Target + Distractor}{2}}$$

The modulation index was defined for every neuron and every position they could take on the curve. A positive modulation index signifies an enhancement of activity evoked by the target curve as compared to the distractor curve. We observed that artificial neurons representing target pixels had a higher activity than those representing distractor pixels. The mean modulation index was 1.18, corresponding to a rate enhancement of 292% (Fig. 5A). This result is qualitatively similar to what has been observed in monkeys (Fig 5B, see also Roelfsema et al., 1998) where the activity of the target elements is also positively modulated compared to the activity of the distractor element. In both monkeys and artificial neural networks, this modulation takes the form of an enhancement of activity that ultimately tag the elements that have to be grouped together.

We next sought to examine the time course of this modulation of the target elements. To this effect, we looked at the normalized difference across time between the activity of neurons elicited by pixels of the target and distractor curve at the same distance from the beginning of the curves. For each position on the curves, we looked at the difference between the blue and the orange curve on Fig. 5C. When this difference becomes positive, that means that the tag of enhanced activity has spread up until this point and that the corresponding element can be grouped with the previous ones. Note that for pixels in the feedforward group, the activity stays constant, as they are not modulated in time and only represent the base representation. We found that the tag of enhanced activity was spreading iteratively from the beginning of the curve one pixel after the other. As can be seen on Fig. 5D, pixels on the target curve that were closer to the beginning of the curve had their activity enhanced first compared to pixels further away. This result is qualitatively similar to what was observed in neurons in the visual cortex of monkeys: neurons whose receptive fields were far from the fixation point saw their activity enhanced later than neurons whose receptive fields were close from it. We found that the time course of modulation was similar in the output layer, i.e. for units that make the eye movement, (Supplementary figure S1) in line with behavioural results by (Jolicoeur et al., 1986) showing that the longer a curve is, the longer it will take for subjects to make a decision. We also evaluated modulation latencies at individual sites: for each individual neuron, we compared the activity when the neuron's receptive field fell on the target curve and when the same neuron's receptive field fell on the distractor curve, and looked at the time of the onset of modulation, defined as the timestep when the difference between the activity when the neuron is on the target curve and the activity when the neuron is on the distractor curve is greater than 70% of its maximum (Fig. 5E). The distribution of modulation latencies for every neuron confirms that when a neuron's receptive field is close to the fixation point, the modulation onset will be earlier than when it is further from the fixation point. Those distributions are qualitatively similar to the ones obtained in monkey's visual cortex (Fig 5F): in both monkeys and artificial neural networks, the further a neuron's receptive field is from the fixation point, the longer it will take for the tag of enhanced activity to reach it. This suggests that, as in monkey's visual cortex, the artificial network has learned to solve the curve-tracing task by spreading enhanced neuronal activity along the target curve, starting at the fixation point. This spreading is iterative and highly reliant on horizontal and feedback connections, which makes it adaptive to any length, and not bound to the network depth.

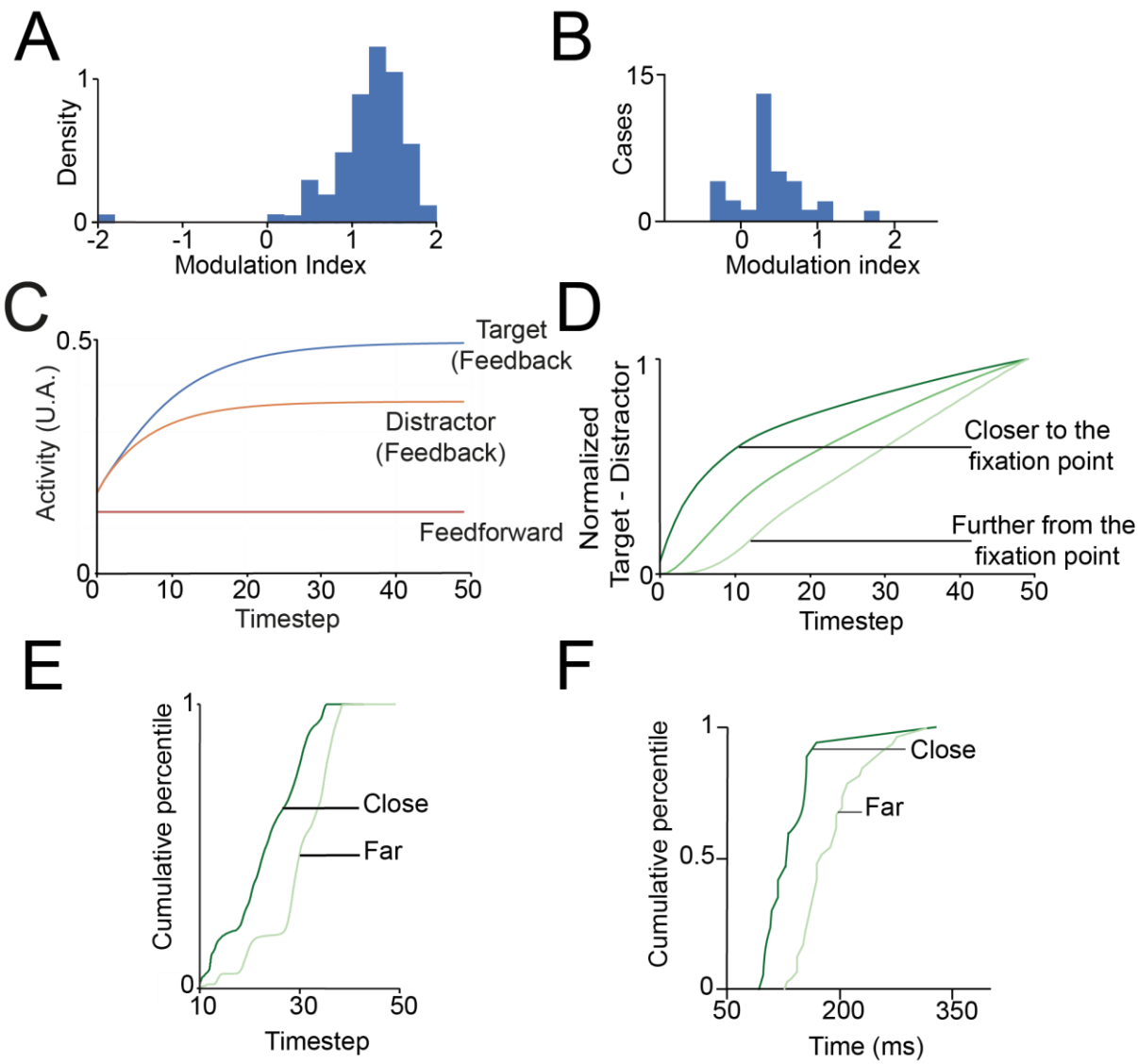


Fig 5. The target curve is identified by the iterative propagation of a tag of enhanced activity. A. Distribution of modulation index of all neurons in the neural networks. A positive modulation index indicates an enhanced response to the target curve. **B**. Distribution of modulation index of recorded neurons in monkey's visual cortex, adapted from (Roelfsema et al., 1998). **C**. Activity of a single neurons when its receptive field is on the target curve (blue) or the distractor curve (orange). When the neuron is on the distractor curve, its activity is enhanced compared to when it's on the distractor curve, similarly to what has been observed in monkeys' visual cortex. **D**. Normalized difference of activity between elements on the target curve and elements on the distractor curve averaged across recording sites. The tag of enhanced activity is propagated iteratively from the fixation point to the end of the curve. The further a pixel is from the beginning of the curve, the later it is modulated. **E**. Distribution of the modulation latency across artificial neurons. When a neuron is further from the beginning of the curve, it is modulated later ($p<10e-15$, Maxx-Whitney U test). **F**. Distribution of the modulation latency across recording sites in monkeys solving the trace task, adapted from Pooresmaeili & Roelfsema, 2014.

How can the network learn to group elements together when the teaching signal given after each trial is only informative about the end of the curve selected? To understand how learning is done, we looked at the activity of the units in the accessory network. As can be seen in Fig. 6A, when a target is chosen, the teaching signal in the curve connected to it first reach pixels that are closer to the target of the eye movement and then propagates to earlier elements of the curve (Fig. 6A, blue curves, plain and dotted respectively). This ensure that horizontal and feedback connections between elements connected to the target of the eye movement are iteratively tagged, and can be updated according to the reward received. Conversely, elements on the curve that was not selected all get the teaching signal at the same time: there is no propagation from the end of the curve, as the end was not selected

(Fig. 6A, orange curves). Thus, in learning how to group pixels into a curve by the propagation of extra activity through the recurrent network, e.g., starting from the fixation point, the accessory network and the learning rule used are essentials. It propagates activity from the pixel that was selected for an eye movement, i.e. the chosen action, back to the neurons that represent pixels that were connected to it. If the eye movement is correct, these recurrent connections are strengthened by the neuromodulator that encodes *d* so that the propagation will be stronger when the stimulus reappears later. In contrast, in case of an erroneous eye movement, these recurrent connections are weakened.

Looking at the accessory network also helps understand why the network needs to be trained with a curriculum. If the network has been trained up to a certain length N, and it is presented with curves N+1 pixels long, the teaching signal, as measured by activity of neurons in the accessory network, will reach the fixation point and the corresponding connections can be trained (Fig. 6B, left, the teaching signal received at the first pixel of the selected curve and the non selected one are different). However, if the network is presented with curves N+5 pixels long, the signal received at the beginning of the curve is the same for the selected and the non selected curve: no teaching signal reaches the beginning of the curve and learning is not possible (Fig. 6B, right, the teaching signal received on the selected curve and the non selected one are identical). That is why training is a lot more efficient when the length of the curve is increased step by step.

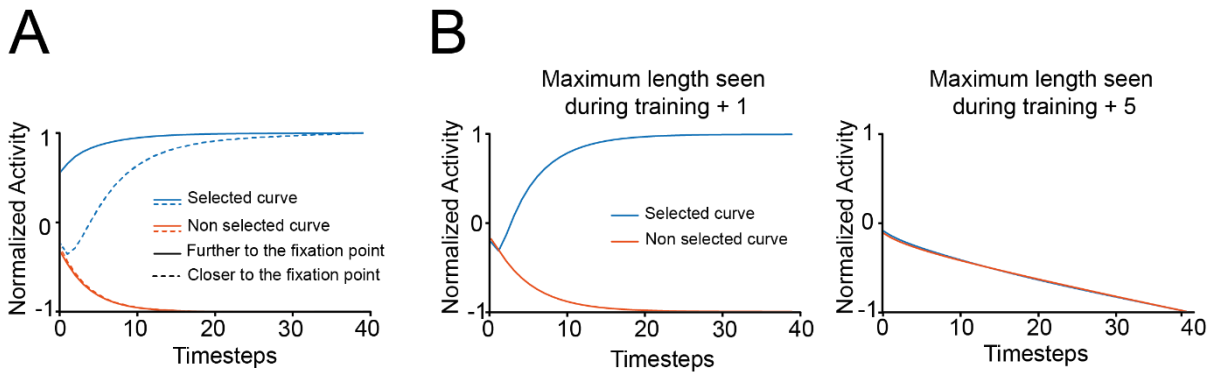


Fig 6. Activity in the accessory network. A. Activity of neurons in the accessory network whose receptive field fall on the selected curve (blue lines) or the non-selected one (orange lines), at different distances from the target of the eye movement (dotted, close to the fixation point, plain, further to the fixation point). The teaching signal propagates along the curve connected to the target of the eye-movement and reach further elements later. For the non-selected curve however, no teaching signal propagates and all the elements are tagged identically. **B.** Activity of neurons corresponding to the beginning of the selected and non selected curves in the accessory network, for a stimuli one pixel longer than what the longer was trained on (left), or 5 pixels longer than what the network was trained on (right). If the stimulus is close to what was seen during training, activity can propagate to the beginning of the curves and those connection can be trained. However, if the stimulus is much longer than what was seen during training, the teaching signal can't link all the elements of the curves together and training fails, emphasizing the importance of a curriculum during training.

To summarize what happens in the visual cortex of monkeys and the artificial networks, when the stimulus is presented feedforward computations first take place. After the feedforward sweep, neurons whose receptive fields fall on the target curve or the distractor curve have the same activity level: they form what has been called a 'base representation' (Roelfsema, 2006). As time unfolds, neurons that respond to the target curve enhance their response compared to neurons that respond to the distractor curve. This enhancement of activity is selectively propagated along the base representation and doesn't spill over from one curve to another one. The reason is that feedforward neurons that respond to image locations that are not occupied by a pixel are silent. The activation of feedforward neurons is necessary before feedback neurons can activate. In other words, feedforward units gate the activity of

feedback neurons so that the enhanced neuronal activity cannot spread across image locations that are not occupied by a pixel.

Search-then-trace task

After making sure that our networks could learn the simple curve tracing task and that it did so in a way analogous as what has been observed in the visual cortex of monkeys, we sought to explore more complex multistep visual routines that require the stitching of different elemental operations and the propagation of information between them, like the search-then-trace task. In this task, the fixation point (one of the two pixels outside of the grid, Fig. 1B, chosen randomly for each trial) could be either orange or red. The network had to make an eye movement toward the blue dot connected to the curve starting with a marker the same colour as the fixation point. This task thus requires the execution of two elemental operations one after the other: first a search operation to identify the position of the correct marker, and then a trace operation along the curve connected to this marker.

We trained the network using a curriculum analogous to the one used for the trace task. First, we presented a stimulus of one blue pixel and one red pixel randomly placed on the grid. If the network achieved 85% accuracy during a 500 trials test phase with weights fixed and no exploration, we presented it with a target and a distractor curves that were three pixels each. If the network achieved 85% accuracy during the test phase, we added one pixel to the curves and repeated the operation until the curves were 9 pixels long.

We trained 24 networks on this task and 79% completed it in an average of 40000 trials. Similarly, as in the trace task, networks learnt a general solution during the course of the training, independent of the position and colour of the fixation point and the characteristics of the curve. At the end of training, networks could perform with above chance accuracy even for curves that were 4 pixels longer than what the network was trained on (Fig. 7A, red curve, $p<10e-4$, at the end of training for 13 pixels long curves, Wilcoxon signed-rank test).

How does the network learn to transmit information between elemental operations such that the output of an elemental operation becomes the input of the next one? In the visual cortex of monkeys realising the same task, electrophysiological recordings show that this is done by tagging, with of focus of enhanced activity, the neurons coding for the information to be transmitted. For example, in the search-then-trace task, the activity of the marker with the same colour as the fixation point will first be enhanced compared to the other marker. This corresponds to a write operation to identify the output of the search operation. From here, a read operation can retrieve the focus of enhanced activity that can now be used as the input for the trace operation, where it can serve as the starting point.

As visual routines consist in the stitching of elemental operations, a difficulty in implementing them is to transmit information between the elemental operations. In the visual cortex of monkeys, this is done by tagging with a focus of enhanced activity the neurons coding for the information to be transmitted. In this way, the output of one elemental operation is clearly marked, identified and can become the input of the next one. Following Moro et al., 2010, we looked at the latency of modulation of neurons tuned to colour (search operation) or spatially selective (trace operation), first by looking at the average neuron activity. In Fig. 7B, we looked at the average normalized difference between neurons whose receptive field fall on the target elements (marker or curve) or the distractor ones. The modulation of activity that acts as a label first arrives on the marker (red curve on Fig. 7B), which corresponds to the output of the search operation being identified. From there, the modulation of activity propagating on the target curve can start, corresponding to the tracing operation (green curves, Fig. 7B). The time course of modulation was again similar in the output layer (Supplementary figure S1). Similarly, as we did for the trace task, we also looked at the

modulation of activity at a population level. For each neuron, we looked at the time of the onset of modulation when the neuron receptive field was on a target element (marker with the same colour as the fixation point or first pixel of the target curve), as compared to when it was on a distractor element, and then plotted the distribution of the times of modulation onset for the marker (red curve, search operation, Fig. 7D), and for the first pixel of the curve (green curve, trace operation, Fig. 7D). Fig. 7D shows results for our artificial agent and Fig. 7E shows results for monkeys' visual cortex (adapted from (Moro et al., 2010)). In both cases, the modulation of activity first tags the neurons corresponding to the marker (the output of the search operation which is also the input of the trace operation), and then the neurons corresponding to the first pixel of the curve, corresponding to the beginning of the trace operation, after a delay.

Those results suggest that the networks spontaneously learn a solution analogous to the one learnt in the visual cortex of monkeys to stitch together different elemental operations into one coherent visual routine: the elemental operations are solved sequentially and a focus of enhanced activity is used to identify the output of each elemental operation so that it can be used as the input for the next one.

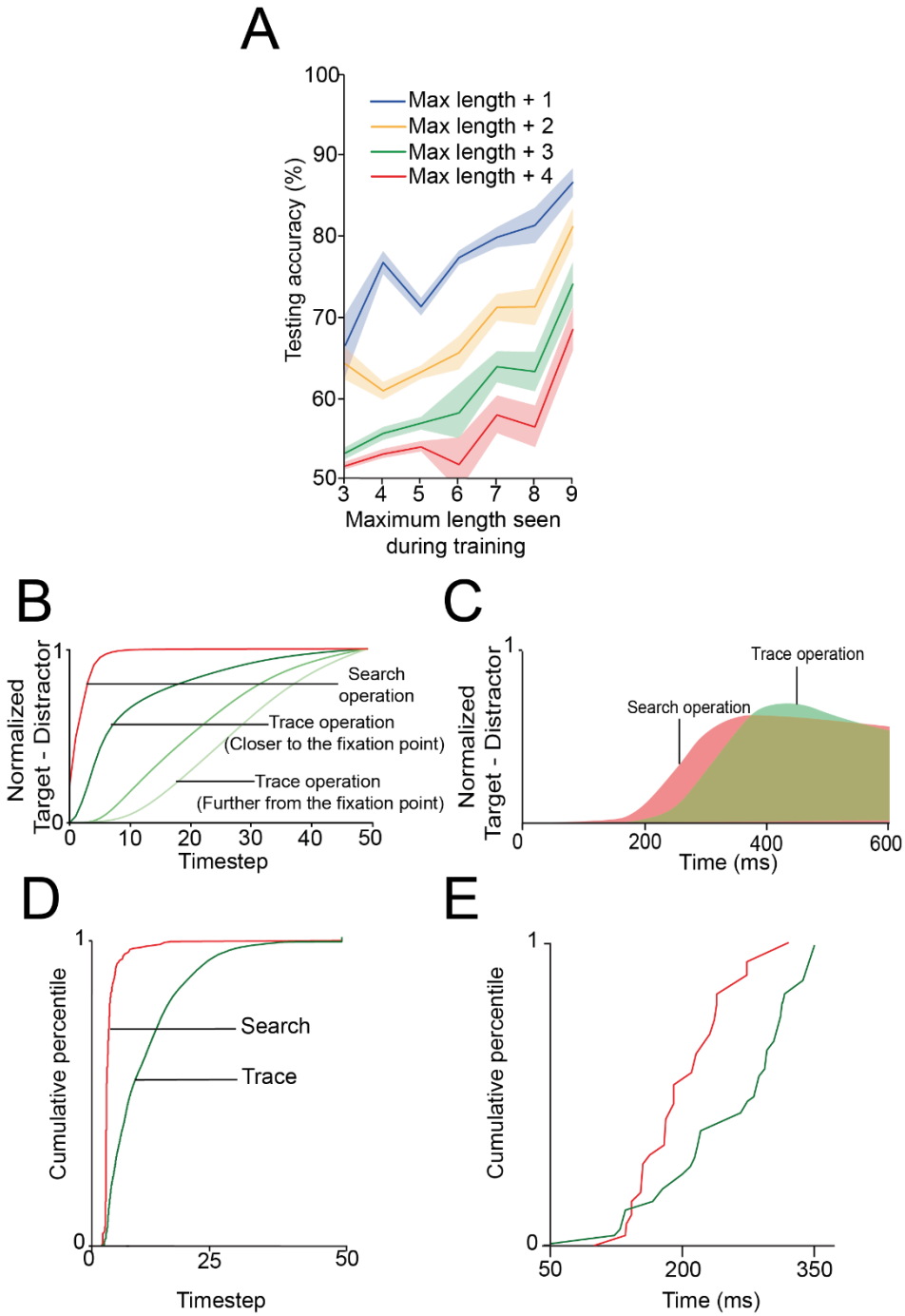


Fig 7. A tag of enhanced activity is used to identify information to transmit between subroutines. A. Testing accuracy for curves of length up to N+4 pixels where N is the maximum length that has been seen during training. At the beginning of training, the agent cannot generalise to longer curve. At the end of training, even though the agent has been trained on curves up to 9 pixels long, it could solve the search-then-trace task with curves up to 13 pixels with above chance accuracy. The agent has learnt an abstract, general rule that can be applied irrespectively of the parameters used (p<10e-6, at the end of training for 13 pixels long curves, Wilcoxon signed-rank test). **B**. Normalized difference of activity between target and distractor elements averaged across recording sites. First the output of the search operation is tagged (search operation, red curve), then the tag is propagated sequentially along the curve connected to it (trace operation, green curves). **C**. In the visual cortex of monkeys, modulation of activity also enhances the output of the search operation (red curve) before labelling the elements of the curve during the trace operation (green curve), adapted from Moro et al. 2010. **D**. Distribution of the modulation latency across artificial neurons. The output of the search operation is modulated first, before the trace operation can begin (p<10e-15, Maxx-Whitney U test). **F**. Distribution of the modulation latency across recording sites in monkeys solving the search-then-trace task, adapted from Pooresmaeili & Roelfsema, 2014.

Trace-then-search task

To make sure that the solution learned by the network could be applied to different visual routines, we then trained networks on the trace-then-search task. In this task, the elemental operations used are the same as in the search-then-trace task but applied in the opposite order. First, the agent has to trace the curve connected to the fixation point. This curve ends with a marker that can be either red or orange. The agent then has to make an eye movement toward the disk that has the same colour as the marker (Fig. 1G,H). All those operations are done mentally, while the gaze is fixated on the fixation point.

For this task, the curriculum we used was different than for the two previous ones. We first trained the networks to perform a simple curve tracing tasks, with the target curve starting with a blue pixel, and the two curves ending with a red and an orange marker. When the networks had learnt to trace 9 pixels long curves, we added the two coloured disks outside of the grid, with the target of the eye movement being the coloured disk matching the colour at the end of the target curve.

We trained 24 networks on this task and 66% completed it in an average of 37000 trials. As for the previous tasks, networks learnt a general solution during the course of the training, independent of the position and colour of the marker and the characteristics of the curve. At the end of training, networks could perform with above chance accuracy even for curves that were 4 pixels longer than what the network was trained on (Fig. 8A, red curve, p=1.4e-4, at the end of training for 13 pixels long curves, Wilcoxon signed-rank test).

In this task, the order of the elemental operations is reversed compared to the search-then-trace task. Thus we expect to see an enhancement of activity corresponding to the trace operation first, followed by the enhancement of activity corresponding to the search operation.

After training the networks on the trace-then-search task, we applied the same analysis as for the search-then-trace task. We first looked at the latency of modulation of spatially selective neurons (trace operation), or of neurons tuned to colour (search operation). Fig. 8B shows the average difference of activity between neurons whose receptive field fall on the target or distractor elements. The enhancement of activity first spreads along the target curve until its end (Fig. 8b green curves), when the correct colour is identified. The tag is then propagated to the marker with the same colour (Fig. 8B red curve). Hence the network first performs a trace operation, followed by a search operation. In the output layer, the time course of modulation followed a similar pattern. (Supplementary figure S1). Looking at the modulation of activity at the population level (Fig. 8D) shows the same order of operation. The last pixel of the curve is enhanced, which correspond to the output of the trace operation being written (green curve), and then the coloured marker is enhanced, corresponding to the realisation of the search operation (red curve). Those results are qualitatively similar to population results in the visual cortex of monkeys, where the modulation of activity first spreads along the curve (Fig. 8E green curve), then to the coloured marker that is the target of the eye movement (Fig. 8F red curve).

Those results suggest that elemental operations can be applied in different orders depending on the task at hand, as has been observed in monkeys' visual cortex.

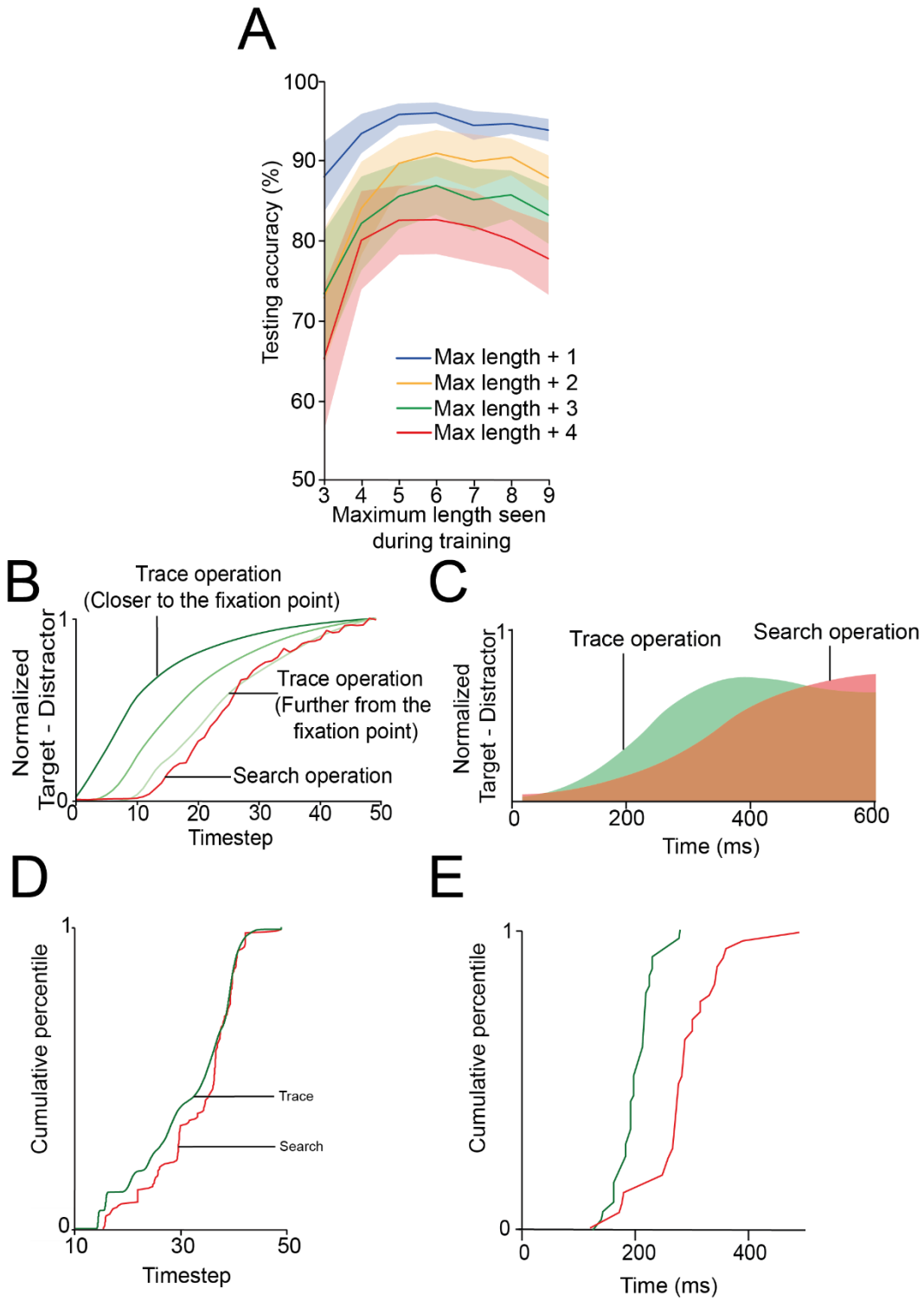


Fig 8. In the trace-then-search task, elemental operations are applied in the opposite order as in the search-then-trace task. A. Testing accuracy for curves of length up to N+4 pixels where N is the maximum length that has been seen during training. At the beginning of training, the agent cannot generalise to longer curve. At the end of training, even though the agent has been trained on curves up to 9 pixels long, it could solve the search-then-trace task with curves up to 13 pixels with above chance accuracy. The agent has learnt an abstract, general rule that can be applied irrespectively of the parameters used (p=1.5e-4, at the end of training for 13 pixels long curves, Wilcoxon signed-rank test). **B**. Normalized difference of activity between target and distractor elements averaged across recording sites. First the curve connected to the fixation point is tagged iteratively (trace operation, green curves), then the tag is propagated to the appropriate coloured marker (search operation, red curve). **C**. In the visual cortex of monkeys, modulation of activity also labels the elements of the curve during the trace operation (green curve), before enhancing the output of the search operation (red curve), adapted from Moro et al., 2010. **D**. Distribution of the modulation latency across artificial neurons. The output of the trace operation is modulated first, before the search operation can begin (p=1.5e-5, Maxx-Whitney U test). **F**. Distribution of the modulation latency across recording sites in monkeys solving the search-then-trace task, adapted from Pooresmaeili & Roelfsema, 2014.

Discussion

In this study, we aimed to understand how complex visual reasoning tasks such as a trace task, a trace-then-search task or a search-then-trace task are learnt. As they consist in complex visual routines, those tasks have to be broken down into multiple, easier to solve elemental operations to be solved. In each elemental operation, like the trace or the search elemental operations, the target elements have to be labelled to be discriminated from the distractor elements. As the tasks consist of multiple elemental operations stitched together, information also has to be transmitted between the subroutines. It has been shown that the elemental operations used here, the trace and the search operations, are solved in the visual cortex by spreading a tag of enhanced activity along a curve, i.e. between spatially selective neurons (trace operation), or between spatially selective and unselective neurons that are tuned to colour (search operation). This label tags target element that are useful inside one subroutine, but also enables the transfer of information between subroutines: a labelled element at the output of a subroutine can then be read out as the input of the next one, akin to the transfer of parameters between subroutines in a computer program (Fig. 9A)

To understand how such strategies are learnt, we trained a neural network on the same tasks monkeys were trained on. A critical feature of our model is the presence of two groups of neurons, a feedforward group that receives inputs only from lower layer, and a feedback group that receives recurrent inputs from higher layers as well as from horizontal connections between neurons in the same layer. Neurons in the feedforward group can only be active if there is a non-zero element in their receptive field. They are then used in gating, via multiplicative connections, neurons in the feedback group. This prevent recurrent activity to spill to neurons that are not driven by feedforward activity. Recording made simultaneously in all layers of the primary visual cortex during a figure-ground segregation task showed that the base representation, corresponding to the initial feedforward sweep, started in layers 4 and 6. After a delay, strong neuronal responses were detected in upper layer 4, layer 1, 2 and 5, corresponding to boundary detection and filling-in due to horizontal and feedback connection (Self et al., 2013). Thus, in our model the feedforward group could correspond to neurons in layers 4 and 6 while the feedback group could correspond to neurons in upper layer 4, layer 1,2 and 5. There are also experimental evidences for the existence of separate channels for feedforward and feedback processing, so that feedback activity do not directly influence feedforward activity (Semedo et al., 2022)

To train the network, we used RELEARNN, a biologically plausible learning rule inspired from the Almeida-Pineda algorithm and that is local both in space and in time. This learning rule relies on the presence of two networks, the main network used during inference, where activity propagates during the presentation of the input, and an accessory network, used for learning, where activity propagates after an action has been selected. Weights are then updated using the activities of the neurons in the main and accessory network, which are all available locally at the synapse. This learning rule was previously used to train networks to learn simple elemental operations, but we show that it is powerful enough to train networks on more complex, multistep tasks.

After training our network, we can look at the activation of spatially selective neurons, and of neurons tuned to colour and compare it the activity of neurons in the visual cortex. Critically, we found that the artificial neural networks spontaneously learnt to develop the same strategies as what as been observed in the visual cortex of monkeys. During the trace elemental operations, neurons whose receptive field are on the target curve have their activity sequentially enhanced, starting from the beginning of the curve while during the search operation, neurons representing the target colour have their activity enhanced. Information is

also transferred between the subroutines as a focus of enhanced activity, so that the output of an elemental operation can be read as the input of the next one (Fig. 9B,C)

Previous studies used artificial neural network models to solve complex, sequential visual tasks. (Linsley et al., 2019), trained recurrent neural network on pathfinder, a dataset consisting of images with curves of different length. Curves could have dots on one, two or none of their extremities and the goal of the network was to indicate the curve with dots on each of its extremities. This task correspond to a trace task. (Linsley et al., 2019) were able to show that the networks performed better than state-of-the-art convolutional neural networks and to stress the importance of recurrent connections in such sequential tasks. The architecture they developed, that they called h-GRU, is inspired by modelling work from (Mély et al., 2018) and the visual cortex. However, if their work shows the importance of recurrence, they trained their network using supervised learning and backpropagation through time, which are both biologically implausible. Moreover, neural activity in their network is not analysed after training so it remains unknown how the network solve the tracing problem and what strategies emerges. (Marić & Domijan, 2019) also tackled the issue of tracing curves. Their networks can trace tasks that have any length and they do so in a similar fashion as what is seen in monkey's visual cortex: by spreading a tag of enhanced activity. However, their network is not trained on such tasks but weights between the neurons are hardcoded and it remains unclear if the architecture and the rules they come up with can emerge through learning with just a reward. Our work is the first one to study complex multistep visual operations both during and after learning.

Even though the correspondence between our results and neurophysiological data is excellent, our model falls short in accounting for other general properties of visual routines like scale-invariance: the time needed to trace a curve doesn't depend on the viewing distance. If you are closer to a curve that needs to be traced, the total length, measured in degrees of visual angle, increase but the distance between the curves increase proportionally (Jolicoeur et al., 1991; Pooresmaeili & Roelfsema, 2014). As each neuron in our model has a receptive field of one pixel, and activity can spread only from one neuron to the next, the time to trace curves is linearly dependant on the viewing distance. To alleviate this issue, future models could take into account the fact that the visual cortex is hierarchically organized, such that neurons in higher areas have bigger receptive fields. When the viewing angle is close, or the curves are far apart, activity can propagate at a higher level, trough neurons with bigger receptive field. If the viewing angle gets farther or the two curves get closer, they would fall on the same receptive field of neurons in higher areas so that the selectivity of the propagation of enhanced activity might be lost. In that situation, activity will need to propagate at a lower level, so that propagating activity over a certain distance in the visual field takes more time (Roelfsema & Houtkamp, 2011).

In real life, if tracing along curves is a task that happens, it is much more common to look at objects with a definite surface. It has been shown that the same tag of enhanced activity that propagates along a curve to bind all of its element together can propagate on the surface on an object. In this way, elements of the object are bound together and the object is recognized as a whole. The same principle that enables our network to trace curves could be expanded to enable a network to perform more real-life and useful scenarios like semantic segmentation.

We have presented a new architecture trained with a biologically plausible learning rule on three tasks for which electrophysiological recording of monkey's visual cortex are available. The qualitative similarity between the activity of artificial neurons and biological ones suggest that the artificial networks learnt, using only reward, the same strategies used by monkeys and that feedforward, feedback, horizontal and lateral connections can interact to learn to spread enhanced activity to group elements belonging to a coherent whole and to transmit information between different subroutines.

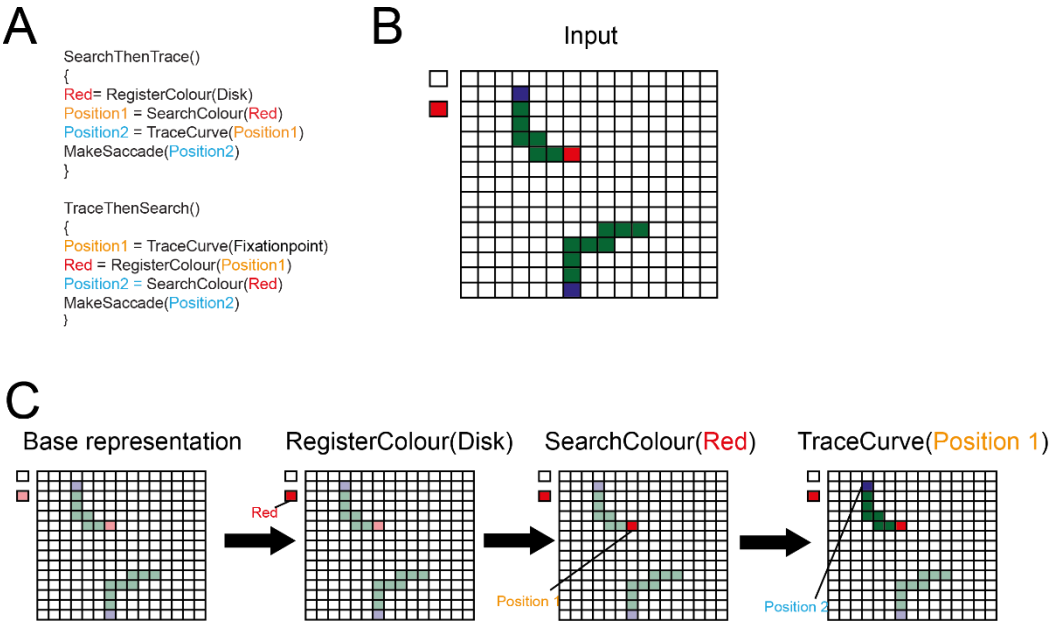


Fig 9. **Visual routines as the sequential execution of elemental operations. A.** Example pseudocode for the search then trace and the trace then search task to illustrate the importance of parameter transfer to solve those tasks. The output of a given subroutine has to be stored in a variable to be used as the input of the next subroutine. In a computer, variables correspond to memory location. In the brains they correspond to a focus of enhanced activity. **B,C.** Example input and representation in the artificial network or the visual cortex of what happens during the execution of the search-then-trace task. First, all elements belonging to a curve, or a marker are identified in the base representation. The marker is then tagged, and the correct colour (red here) is identified. The search operation then identifies the position of the red pixel on the grid and tags it as its output (Position 1). From here, the trace operation can begin and iteratively tag elements belonging to the target curve, until the target for the saccade is identified (Position 2).

Methods

Data Analysis

We trained 24 networks for each task. For the analysis, we only kept networks that learnt the tasks for curves up to 9 pixels long. After the training was done the weights were frozen and the exploration rate ϵ was set to 0. Each network was presented 15000 stimuli generated randomly and the neurons activations for those 15000 stimuli were kept for analysis. We only included correct trials in the analysis.

Parameters

All networks were initialized and trained with the same parameters. They are summarized in the following table.

Parameter	Value
Learning rate δ	0.02
Exploration rate ϵ	0.05
Weights between hidden layers	Sampled from U(0,0.1)

Weights between the last hidden and the output layers, same receptive field	Sampled from U(0,0.1)
Weights between the last hidden and the output layers, different receptive field	Sampled from U(-0.01,0)

Code availability

Code used to train the networks and to analyze the data is available on the following github address: https://github.com/samimol/Visual-Routines

References

Almeida, L. B. (1987). A learning rule for asynchronous perceptrons with feedback in a combinatorial environment. *Proceedings of the IEEE First International Conference on Neural Networks*, *II*, 609–618.

Brosch, T., Neumann, H., & Roelfsema, P. R. (2015). Reinforcement Learning of Linking and Tracing Contours in Recurrent Neural Networks. *PLoS Computational Biology*, *11*(10), e1004489. https://doi.org/10.1371/journal.pcbi.1004489

Glorot, X., Bordes, A., & Bengio, Y. (2011). *Deep Sparse Rectifier Neural Networks* (pp. 315–323). JMLR Workshop and Conference Proceedings. https://proceedings.mlr.press/v15/glorot11a.html

Houtkamp, R., Spekreijse, H., & Roelfsema, P. R. (2003). A gradual spread of attention during mental curve tracing. *Perception & Psychophysics*, *65*(7), 1136–1144. https://doi.org/10.3758/BF03194840

Jolicoeur, P., Ullman, S., & Mackay, M. (1986). Curve tracing: A possible basic operation in the perception of spatial relations. *Memory & Cognition*, *14*(2), 129–140. https://doi.org/10.3758/BF03198373

Jolicoeur, P., Ullman, S., & Mackay, M. (1991). Visual Curve Tracing Properties. *Journal of Experimental Psychology: Human Perception and Performance*, *17*(4), 997–1022. https://doi.org/10.1037/0096-1523.17.4.997

Kim, M.-S., & Cave, K. R. (1995). Spatial Attention in Visual Search for Features and Feature Conjunctions. *Source: Psychological Science*, *6*(6), 376–380.

Lamme, V. A. F., & Roelfsema, P. R. (2000). The distinct modes of vision offered by feedforward and recurrent processing. *Trends in Neurosciences*, *23*(11), 571–579. https://doi.org/10.1016/S0166-2236(00)01657-X

Liao, R., Xiong, Y., Fetaya, E., Zhang, L., Yoon, K., Pitkow, X., Urtasun, R., & Zemel, R. (2018). *Reviving and Improving Recurrent Back-Propagation*. https://github.com/

Lillicrap, T. P., & Santoro, A. (2019). Backpropagation through time and the brain. In *Current Opinion in Neurobiology* (Vol. 55, pp. 82–89). Elsevier Ltd. https://doi.org/10.1016/j.conb.2019.01.011

Linsley, D., Ashok, A. K., Govindarajan, L. N., Liu, R., & Serre, T. (2020). *Stable and expressive recurrent vision models*. https://github.com/c-rbp.

Linsley, D., Kim, J., Veerabadran, V., Windolf, C., & Serre, T. (2019). *Learning long-range*

spatial dependencies with horizontal gated recurrent units.

Marić, M., & Domijan, D. (2019). Neural dynamics of spreading attentional labels in mental contour tracing. *Neural Networks*, *119*, 113–138. https://doi.org/10.1016/J.NEUNET.2019.07.016

Mély, D. A., Linsley, D., & Serre, T. (2018). Complementary surrounds explain diverse contextual phenomena across visual modalities. *Psychological Review*, *125*(5), 769–784. https://doi.org/10.1037/REV0000109

Moro, S. I., Tolboom, M., Khayat, P. S., & Roelfsema, P. R. (2010). Neuronal activity in the visual cortex reveals the temporal order of cognitive operations. *Journal of Neuroscience*, *30*(48), 16293–16303. https://doi.org/10.1523/JNEUROSCI.1256-10.2010

Pineda, F. J. (1987). Generalization of back-propagation to recurrent neural networks. *Physical Review Letters*, *59*(19), 2229–2232. https://doi.org/10.1103/PhysRevLett.59.2229

Pooresmaeili, A., & Roelfsema, P. R. (2014). A growth-cone model for the spread of object-based attention during contour grouping. *Current Biology*, *24*(24), 2869–2877. https://doi.org/10.1016/j.cub.2014.10.007

Roelfsema, P. R. (2005). Elemental operations in vision. *Trends in Cognitive Sciences*, *9*(5), 226–233. https://doi.org/10.1016/j.tics.2005.03.012

Roelfsema, P. R. (2006). Cortical algorithms for perceptual grouping. *Annual Review of Neuroscience*, *29*, 203–227. https://doi.org/10.1146/ANNUREV.NEURO.29.051605.112939

Roelfsema, P. R., Bohte, S., & Spekreijse, H. (1999). *Algorithms for the detection of connectedness and their neural implementation*.

Roelfsema, P. R., & de Lange, F. P. (2016). Early Visual Cortex as a Multiscale Cognitive Blackboard. In *Annual review of vision science* (Vol. 2, Issue 1, pp. 131–151). https://doi.org/10.1146/annurev-vision-111815-114443

Roelfsema, P. R., & Houtkamp, R. (2011). Incremental grouping of image elements in vision. *Attention, Perception, and Psychophysics*, *73*(8), 2542–2572. https://doi.org/10.3758/s13414-011-0200-0

Roelfsema, P. R., Khaya, P. S., & Spekreijse, H. (2003). Subtask sequencing in the primary visual cortex. *Proceedings of the National Academy of Sciences of the United States of America*, *100*(9), 5467–5472. https://doi.org/10.1073/pnas.0431051100

Roelfsema, P. R., Lamme, V. A. F., & Spekreijse, H. (1998). Object-based attention in the primary visual cortex of the macaque monkey. *Nature 1998 395:6700*, *395*(6700), 376–381. https://doi.org/10.1038/26475

Roelfsema, P. R., Lamme, V. A. F., & Spekreijse, H. (2000). The implementation of visual routines. *Vision Research*, *40*(10–12), 1385–1411. https://doi.org/10.1016/S0042-6989(00)00004-3

Roelfsema, P. R., Tolboom, M., & Khayat, P. S. (2007). Different Processing Phases for Features, Figures, and Selective Attention in the Primary Visual Cortex. *Neuron*, *56*(5), 785–792. https://doi.org/10.1016/J.NEURON.2007.10.006/ATTACHMENT/9D72C394-0631-4A87-B220-8A8887E0869E/MMC2.AVI

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*(5306), 1593–1599.

https://doi.org/10.1126/science.275.5306.1593

Self, M. W., van Kerkoerle, T., Supèr, H., & Roelfsema, P. R. (2013). Distinct roles of the cortical layers of area V1 in figure-ground segregation. *Current Biology : CB*, *23*(21), 2121–2129. https://doi.org/10.1016/J.CUB.2013.09.013

Semedo, J. D., Jasper, A. I., Zandvakili, A., Krishna, A., Aschner, A., Machens, C. K., Kohn, A., & Yu, B. M. (2022). Feedforward and feedback interactions between visual cortical areas use different population activity patterns. *Nature Communications 2022 13:1*, *13*(1), 1–14. https://doi.org/10.1038/s41467-022-28552-w

Simard, P. Y., Ottaway, M. B., & Ballard, D. H. (1989). Analysis of Recurrent Backpropagation. *UR Research*, 103–112. https://urresearch.rochester.edu/institutionalPublicationPublicView.action?institutionalItemId=5516

Sutton, R. S., & Barto, A. G. (2018). Reinforcement Learning: An Introduction, Second Edition. In *The Lancet* (Vol. 258, Issue 6685). https://doi.org/10.1016/S0140-6736(51)92942-X

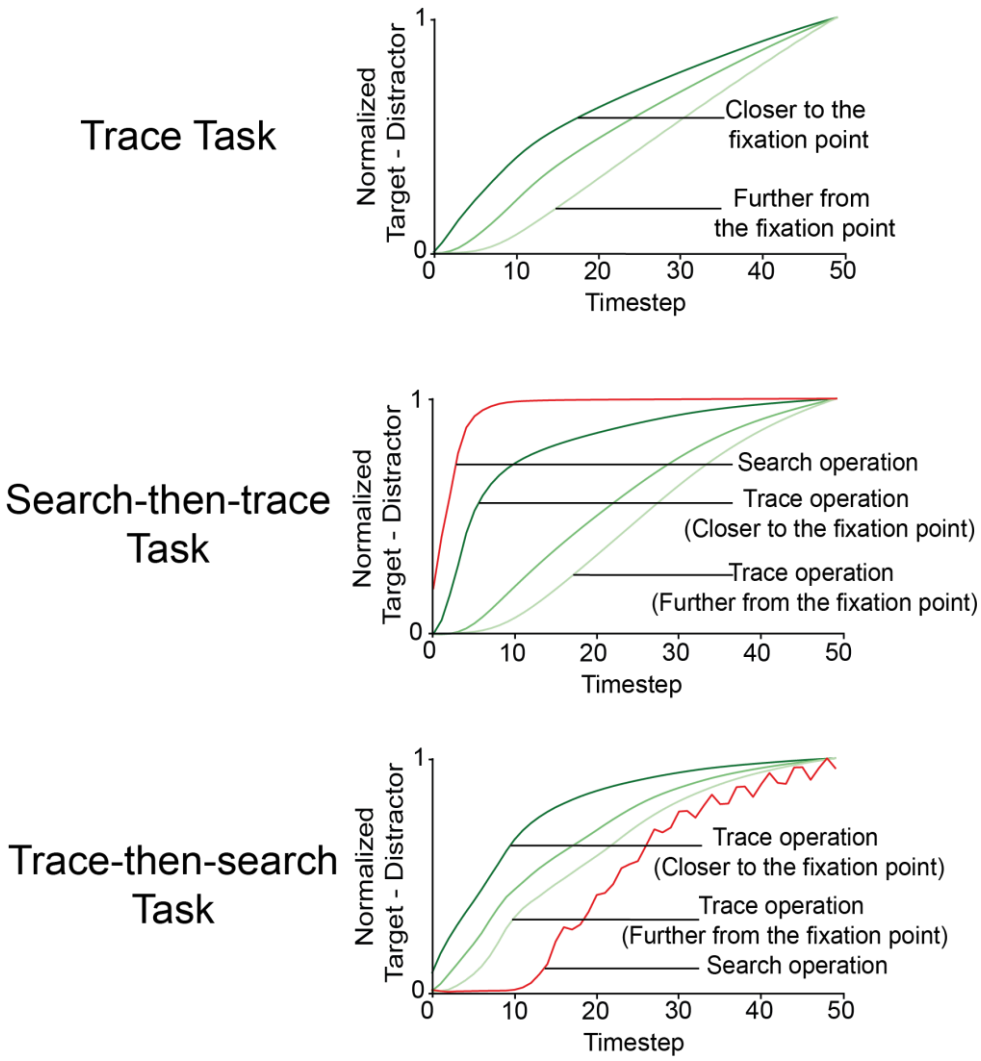
Ullman, S. (1984). Visual routines. *Cognition*, *18*(1–3), 97–159. https://doi.org/10.1016/0010-0277(84)90023-4

Vanrullen, R. (2007). The power of the feed-forward sweep. *Advances in Cognitive Psychology*, *3*(1–2), 167–176. https://doi.org/10.2478/V10053-008-0022-3

Werbos, P. J. (1988). Generalization of backpropagation with application to a recurrent gas market model. *Neural Networks*, *1*(4), 339–356. https://doi.org/10.1016/0893-6080(88)90007-X

Werbos, P. J. (1990). Backpropagation Through Time: What It Does and How to Do It. *Proceedings of the IEEE*, *78*(10), 1550–1560. https://doi.org/10.1109/5.58337

Zylberberg, A., Dehaene, S., Roelfsema, P. R., & Sigman, M. (2011). The human Turing machine: A neural framework for mental programs. In *Trends in Cognitive Sciences* (Vol. 15, Issue 7, pp. 293–300). https://doi.org/10.1016/j.tics.2011.05.007



Supplementary figure S1. **Time course of modulation in the output layer.** In the output layer, the modulation of activity of neurons whose receptive field fall on the target curve follows the same time course as neurons in the hidden layers. This means that the longer modulation will take to propagate in the visual area, the longer the output area will be to select a target for the eye-movement, in line with behavioural results by (Jolicoeur et al., 1986), showing that the longer a curve is, the longer it will take for subjects to make a decision.