A New Approach towards Vision suggested by Biologically Realistic Neural Microcircuit Models[1]

Wolfgang Maass, Robert Legenstein, and Henry Markram

Wolfgang Maass, Robert Legenstein
Institute for Theoretical Computer Science
Technische Universitaet Graz
Inffeldgasse 16b, A-8010 Graz, Austria
E-mail: {maass, legi}@ igi.tu-graz.ac.at
http://www.igi.tugraz.at/maass/
http://www.igi.tugraz.at/legi/

Henry Markram
Brain Mind Institute
Ecole Polytechnique Federale de Lausanne
CE-Ecublens
CH- 1015 Lausanne
E-mail: henry.markram@epfl.ch
http://www.weizmann.ac.il/neurobiology/labs/markram/
markram.html

Abstract

We propose an alternative paradigm for processing time-varying visual inputs, in particular for tasks involving temporal and spatial integration, which is inspired by hypotheses about the computational role of cortical microcircuits. Instead of storing discrete frames of past visual inputs in a suitable datastructure and then applying a time-consuming algorithm to extract salient information about motion from these frames, we demonstrate through computer simulations that the following approach is in principle also feasible. Continuously time-varying visual inputs $u(\cdot)$ are directly injected into a sufficiently large and complex dynamical system (such as for example a cortical microcircuit). Simple readout devices can be trained to extract from the current state $x(t)$ of such dynamical system at any time t in real-time salient information about the trajectory of recent inputs $u(s)$ for $s < t$. Since detailed knowledge about the precise structure of the dynamical system is not needed for that, it can in principle also be implemented with partially unknown or faulty analog hardware. In addition, this approach supports parallel real-time processing of time-varying visual inputs for diverse tasks, since different readouts can be trained to extract concurrently from the same dynamical system completely different information components.

Keywords: cortical microcircuits, recurrent connections, spiking neurons, dynamic synapses, dynamical systems, movement prediction, direction of motion, novelty detection, parallel computing.

---

[1] The work was partially supported by the Austrian Science Fond FWF, project # P15386.

1. Introduction

Cortical neural circuits differ in several essential aspects from circuit models arising in the context of computer science or artificial neural networks. The computational units of cortical circuits (neurons and synapses) are quite heterogeneous, being endowed with a multitude of different dynamical processes on several time scales (see for example [Gupta et al., 2000] and [Markram et al., 1998]). Furthermore these units are highly recurrent ("loops within loops") but sparsely connected, with accidental features in the connectivity patterns. It appears to be virtually impossible to implement any previously suggested approach for real-time processing of dynamically changing visual inputs on such circuits.

We show that nevertheless biologically realistic neural microcircuit models can be used for analyzing complex time-varying visual inputs once they are viewed from a suitable organisational perspective. Instead of collecting information about preceding input frames in a suitable datastructure, and then applying time consuming algorithms in order to extract the desired information, we demonstrate through computer simulations that a quite different strategy is in principle feasible, and may in fact be biologically more realistic. One can inject continuously time-varying visual stimuli $u(\cdot)$ directly into a simulated generic neural microcircuit. Realistic models for cortical microcircuits are sparse but highly recurrently connected, and involve diverse dynamical components. Therefore the current spike output $x(t)$ of such circuit, which is an n-dimensional vector if the circuit consists of n neurons, contains information about the trajectory of preceding inputs $u(s)$ for $s < t$, although in a highly nonlinear encoding. Nevertheless simple algorithms for static pattern classification and regression can be applied to extract at any time t from the current circuit output $x(t)$ information about recent movements $u(\cdot)$. The decoding of this information from the current circuit output $x(t)$ presents a virtually unsolvable task for a human observer, or any explicit decoding algorithm, especially if this circuit output is high dimensional (for example about 10.000-dimensional in the case of a cortical column). But the nonlinearity and high dimensionality of the "code" $x(t)$ enhance the chances for being able to train simple linear readout devices to extract the desired information – without the danger of getting stuck in local minima of the error function. The reason is that the transformation of the time-varying inputs $u(\cdot)$ into a subsequent high dimensional circuit state $x(t)$ tends to act like a kernel for support vector machines that facilitates linear readout.

In order to demonstrate the possible contribution of generic cortical circuit structures for typical visual processing tasks, no attempt has been made in the simulations discussed in this article to endow the neural microcircuit model in addition with specialized feature detectors such as orientation- or direction selective neurons, or neurons with particular receptive fields, or a more specialized connectivity structure. These features are likely to enhance the performance of such microcircuit models for a range of visual tasks. In fact, the generic recurrent sparse connectivity and heterogeneity of our microcircuit model may provide the previously missing organisational scheme for combining the outputs of such specialized feature detectors for a wide variety of tasks that require real-time processing of rapidly varying visual information, even tasks that are not well-matched to the sizes of receptive fields or other details of specialized feature detectors. Thus the approach discussed in this article may provide new concepts and ideas for investigating the role of neural circuits in the visual cortex: by viewing them as wetware implementations of kernels that are particularly useful for processing time-varying visual inputs (analogously to the kernels discussed in [Schölkopf and Smola, 2002], which are particularly useful for other types of tasks).

Earlier neural circuit models for visual processing, such as for example those of [Abbott and Blum, 1996] and [Rao and Sejnowski, 2000], focused on neural circuits that had essentially "no hidden neurons". They consisted of arrays of neurons where each excitatory neuron in the network was directly targeted by exactly one "sensory" input, with excitatory or inhibitory (mediated by point-to-point interneurons) lateral connections between these directly input-driven neurons. Consequently these earlier articles focused on predictions of input movements of *pointwise* objects whose direction and speed was aligned with the specific pattern of connections and transmission delays of the lateral connections between the array of input-driven neurons. In contrast, we consider in this article generic randomly connected networks of integrate-and-fire neurons with (biologically realistic) very short synaptic delays, into which the input is mapped by random connections that favor a topographic map. We demonstrate that the computational advantage of this biologically more realistic architecture lies in its capability to support analyzes of a much larger diversity of shapes, directions and speeds of moving objects that could possibly be anticipated in the pattern of connections and transmission delays of an explicitly constructed circuit.

The idea to train just the neurons that read out information from a recurrent neural circuit had apparently first been proposed by [Buonomano and Merzenich, 1995]. The computational power of the resulting model has previously been analyzed in [Maass et al., 2001]. A related model for artificial neural networks was proposed in [Jaeger, 2001].

2. **The Generic Neural Microcircuit Model**

We employed circuit models that reflect empirical data from microcircuits in rat somatosensory cortex (see [Gupta et al., 2000] and [Markram et al., 1998]) with regard to the types of dynamic synapses and their relationship to the type of presynaptic and postsynaptic neuron (excitatory or inhibitory). The biological models for neurons and dynamic synapses that were used were rather realistic (in contrast to the neurons and synapses used in artificial neural network models). Connections between neurons were chosen randomly, but with a biologically realistic bias towards local connections. The resulting circuits, each consisting of 768 leaky integrate-and-fire neurons (of which 20 % were randomly chosen to be inhibitory), were sparsely but highly recurrently connected. We refer to the Appendix and [Maass et al., 2001] for details.

If one excites such complex and highly dynamic recurrent circuit with a continuous input stream $u(s)$, and looks at a later time $t > s$ at the current output $x(t)$ of the circuit, then $x(t)$ is likely to hold a substantial amount of information about recent inputs. We as human observers may not be able to understand the "code" by which this information about $u(s)$ is encoded in $x(t)$, but that is obviously not essential. Essential is whether a readout neuron that has to extract such information at time t for a specific task can accomplish this. But this amounts to a classical (static) pattern recognition or regression problem, since the temporal dynamics of the input stream $u(s)$ has been transformed by the recurrent circuit into a single high dimensional spatial pattern $x(t)$. This pattern classification or regression task tends to be relatively easy to learn, even by a memoryless readout, provided the desired information is present in the circuit output $x(t)$. We demonstrate that a single readout can be trained to accomplish this task for many different time points t.

If the recurrent neural circuit is sufficiently large, it may support this learning task by acting like a kernel for support vector machines (see [Vapnik, 1998]), which presents a large number of nonlinear combinations of components of the preceding input stream to the readout. Such nonlinear projection of the original input stream $u(\cdot)$ into a high dimensional space tends to facilitate the extraction of information about this input stream at later times t, since it boosts the power of linear readouts for classification and regression tasks. Linear readouts are not only better models for the readout capabilities of a biological neuron than for example multi-layer-perceptrons, but their training is much easier and robust because it cannot get stuck in local minima of the error function (see [Vapnik, 1998] and [Haykin, 1999]). These considerations suggest new hypotheses regarding the computational function of generic neural circuits in the visual cortex: to serve as general-purpose temporal integrators, and simultaneously as kernels (i.e., nonlinear projections into a higher dimensional space) to facilitate subsequent linear readout of information whenever it is needed. Note that in all experiments described in this article only the readouts were trained for specific tasks, whereas always the same recurrent circuit (with a randomly chosen fixed setting of synaptic "weights" and other parameters) was used for generating $x(t)$.

Input to this recurrent circuit was provided by 64 simulated sensors that were arranged in an 8×8 2D array (see Figure 1). The receptive field of each sensor was modeled as a square of unit size. The sensor output (with range [0, 1]), sampled every 5 ms, reflected at any moment the fraction of the corresponding unit square that was currently covered by a simulated moving object. The outputs of the 64 sensors were projected as analog input to the circuit in a topographic manner.[2]

Neural readouts from this randomly connected recurrent circuit of leaky integrate-and-fire neurons were simulated as in [Maass et al., 2001] by pools of 50 neurons (without lateral connections) that received postsynaptic currents from all neurons in the recurrent circuit, caused by their firing. For simplicity we assumed that each readout neuron outputs a value of 1 at time *t* if its membrane potential is above its threshold at that time

---

[2] The $16 \times 16 \times 3$ neuronal sheet was divided into 64 $2 \times 2 \times 3$ *input regions*, and each sensor from the 8×8 sensor array projected to one such input region in a topographic manner, i.e., neighboring sensors projected onto neighboring input regions. Each sensor output was injected into a randomly chosen subset of the neurons in the corresponding input region (selection probability 0.6) in the form of additional input current (added to their background input current). One could just as well provide this input in the form of Poisson spike trains with a corresponding time-varying firing rate, with a slight loss in performance of the system.

t, and otherwise a value of 0. A piecewise linear squashing function applied to the fraction of these 50 perceptrons that currently output a 1 was interpreted as a time-varying analog output of that readout pool with values in [0, 1] (sampled every 25 ms). This corresponds to a space rate code if the perceptrons are replaced by leaky integrate-and-fire neurons, which can be done with a 5-10% performance loss of the system. The synapses of the neurons in each readout pool were adapted according to the p-delta rule of [Auer et al., 2002]. But in contrast to [Maass et al., 2001], this learning rule was used here in an unsupervised mode, where target output values provided by a supervisor were replaced by the actual later activations of the sensors which they predicted (with prediction spans of 25 and 50 ms into the future). Other readouts were trained in the same unsupervised manner to predict whether a sensor on the perimeter was going to be activated by more than 50 % when the moving object finally left the sensor field. These neurons needed to predict farer into the future (100 – 150 ms, depending on the speed of the moving object, since they were trained to produce their prediction while the object was still in the mid-area of the sensor field). The latter readouts only needed to predict a binary variable, and therefore the corresponding readout pools could be replaced by a single perceptron (or a single integrate-and-fire neuron), at a cost of about 5 % in prediction accuracy.

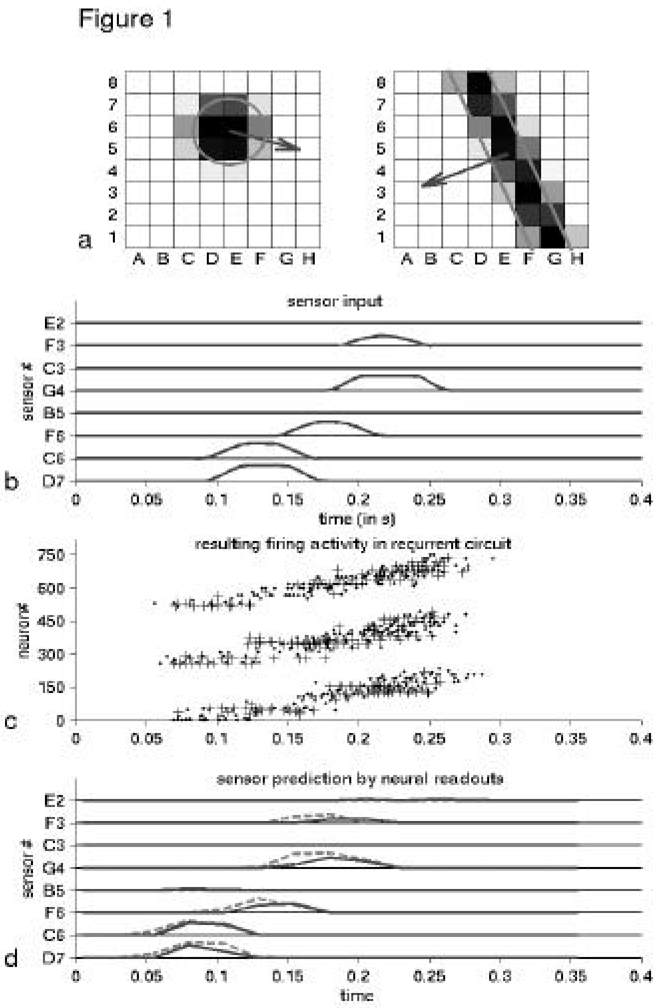

Figure 1 : The prediction task.

a) Typical movements of objects over a 8 x 8 sensor field.

b) Time course of activation of 8 randomly selected sensors caused by the movement of the ball indicated on the l.h.s. of panel a.

c) Resulting firing times of 768 integrate-and-fire neurons in the recurrent circuit of integrate-and-fire neurons (firing of inhibitory neurons marked by +). The neurons in the 16 x 16 x 3 array were numbered layer by layer. Hence the 3 clusters in the spike raster result from concurrent activity in the 3 layers of the circuit.

d) Prediction targets (dashed lines) and actual predictions (solid lines) for the 8 sensors from panel b. (Predictions were sampled every 25 ms, solid curves result from linear interpolation.)

We wanted to demonstrate that the same microcircuit model can support a large number of different vision tasks. Hence in our simulations 102 readout pools received their input from the same recurrent circuit consisting of 768 leaky integrate-and-fire neurons. 36 of them were trained to predict subsequent sensor activation 25 ms later in the interior 6 × 6 subarray of the 8 × 8 sensor array, 36 other ones were trained for a 50 ms prediction of the same sensors (note that prediction for those sensors on the perimeter where the object enters the field is impossible, hence we have not tried to predict all 64 sensors). 28 readout pools were trained to predict which sensors on the perimeter of the 8 × 8 array were later going to be activated when the moving object left the sensor field. All these 100 readout pools were trained in an unsupervised manner by movements of two different objects, a ball and a bar, over the sensor field. In order to examine the claim that other readout pools could be trained simultaneously for completely different tasks, we trained one further readout pool in a supervised manner by the p-delta rule to classify the object that moved (ball or bar), and another readout pool to estimate the speed of the moving object.

3. Demonstration that this New Approach towards Visual Processing is in Principle Feasible

The general setup of the prediction task is illustrated in Figure 1. Moving objects, a ball or a bar, are presented to an 8×8 array of sensors (panel a). The time course of activations of 8 randomly selected sensors, resulting from a typical movement of the ball, is shown in panel b. Corresponding functions of time, but for all 64 sensors, are projected as 64 dimensional input by a topographic map into a generic recurrent circuit of spiking neurons (see Section 2). The resulting firing activity of all 768 integrate-and-fire neurons in the recurrent circuit is shown in panel c. Panel d of Figure 1 shows the target output for 8 of the 102 readout pools. These 8 readout pools have the task to predict the output that the 8 sensors shown in panel b will produce 50 ms later. Hence their target output (dashed line) is formally the same function as shown in panel b, but shifted by 50 ms to the left. The solid lines in panel d show the actual output of the corresponding readout pools after unsupervised learning. Thus in each row of panel d the difference between the dashed and predicted line is the prediction error of the corresponding readout pool.

The diversity of object movements that are presented to the 64 sensors is indicated in Figure 2. Any straight line that crosses the marked horizontal or vertical line segments of length 4 in the middle of the 8x8 field may occur as trajectory for the center of an object. Training and test examples are drawn randomly from this – in principle infinite – set of trajectories, each with a movement speed that was drawn independently from a uniform distribution over the interval from 30 to 50 units per second (unit = side length of a unit square). Shown in Figure 2 are 20 trajectories that were randomly drawn from this distribution. Any such movement is carried out by an independently drawn object type (ball or bar), where bars were assumed to be oriented vertically to their direction of movement. Besides movements on straight lines one could train the same circuit just as well for predicting nonlinear movements, since nothing in the circuit was specialized for predicting linear movements.

36 readout pools were trained to predict for any such object movement the sensor activations of the 6×6 sensors in the interior of the 8×8 array 25 ms into the future. Further 36 readout pools were independently trained to predict their activation 50 ms into the future, showing that the prediction span can basically be chosen arbitrarily. At any time t (sampled every 25 ms from 0 to 400 ms) one uses for each of the 72 readout pools that predict sensory input ΔT into the future the actual activation of the corresponding sensor at time $t + \Delta T$ as target value ("correction") for the learning rule. The 72 readout pools for short-term movement prediction were trained by 1500 randomly drawn examples of object movements. More precisely, they were trained to predict future sensor activation at any time (sampled every 25 ms) during the 400 ms time interval while the object (ball or bar) moved over the sensory field, each with another trajectory and speed.

Among the predictions of the 72 different readout pools on 300 novel test inputs there were for the 25 ms prediction 8.5 % false alarms (sensory activity erroneously predicted) and 4.8 % missed predictions of subsequent sensor activity. For those cases where a readout pool correctly predicted that a sensor will become active, the mean of the time period of its activation was predicted with an average error of 10.1 ms. For the 50 ms prediction there were for 300 novel test inputs 16.5 % false alarms, 4.6 % missed predictions of sensory activations, and an average 14.5 ms error in the prediction of the mean of the time interval of sensory activity.

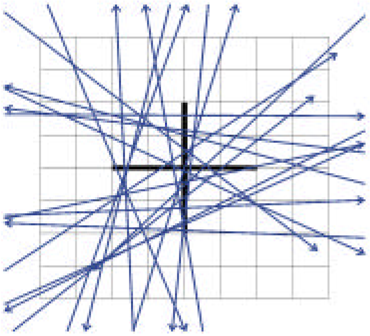

Figure 2: 20 typical trajectories of movements of the center of an object (ball or bar).

One should keep in mind that movement prediction is actually a computationally quite difficult task, especially for a moving ball, since it requires context-dependent integration of information from past inputs over time and space. This computational problem is often referred to as the "aperture problem": from the perspective of a single sensor (or a small group of sensors) that is currently partially activated because the moving ball is covering part of its associated unit square (i.e., its "receptive field") it is impossible to predict whether this sensor will become more or less activated at the next movement (see [Mallot, 2000]). In order to decide that question, one has to know whether the center of the ball is moving towards its receptive field, or is just passing it tangentially. To predict whether a sensor that is currently not even activated will be activated 25 or 50 ms later, poses an even more difficult problem that requires not only information about the direction of the moving object, but also about its speed and shape. Since there exists in this experiment no preprocessor that extracts these features, which are vital for a successful prediction, each readout pool that carries out predictions for a particular sensor has to extract on its own these relevant pieces of information from the raw and unfiltered information about the recent history of sensor activities, which are still "reverberating" in the recurrent circuit.

28 further readout pools were trained in a similar unsupervised manner (with 1000 training examples) to predict *where* the moving object is going to leave the sensor field. More precisely, they predict which of the 28 sensors on the perimeter are going to be activated by more than 50 % when the moving object leaves the 8 × 8 sensor field. This requires a prediction for a context-dependent time span into the future that varies by 66 % between instances of the task, due to the varying speeds of moving objects. We arranged that this prediction had to be made while the object crossed the central region of the 8 × 8 field, hence at a time when the current position of the moving object provided hardly any information about the location where it will leave the field, since all movements go through the mid area of the field. Therefore the tasks of these 28 readout neurons require the computation of the direction of movement of the object, and hence a computationally difficult disambiguation of the current sensory input. We refer to the discussion of the disambiguation problem of sequence prediction in [Levy, 1996] and [Abbott and Blum, 1996]. The latter article discusses difficulties of disambiguation of movement prediction that arise already if one has just pointwise objects moving at a fixed speed, and just 2 of their possible trajectories cross. Obviously the disambiguation problem is substantially more severe in our case, since a virtually unlimited number of trajectories (see Figure 2) of different extended objects, moving at different speeds, crosses in the mid area of the sensor field. The disambiguation is provided in our case simply through the "context" established inside the recurrent circuit through the traces (or "reverberations") left by preceding sensor activations. Figure 3 shows in panel a a typical current position of the moving ball, as well as the sensors on the perimeter that are going to be active by ≥ 50 % when the object will finally leave the sensory field. In panel b the predictions of the corresponding 28 readout neurons (at the time when the object crosses the mid-area of the sensory field) is also indicated (striped squares). The prediction performance of these 28 readout neurons was evaluated as follows. We considered for each movement the line from that point on the opposite part of the perimeter, where the center of the ball had entered the sensory field, to the midpoint of the group of those sensors on the perimeter that were activated when the ball left the sensory field (dashed line). We compared this line with the line that started at the same point, but went to the midpoint of those sensor positions which were predicted by the 28 readout neurons to be activated when the ball left the sensory field (solid line). The angle between these two lines had an average value of 4.9 degrees for 100 randomly drawn novel test movements of the ball (each with an independently drawn trajectory and speed).

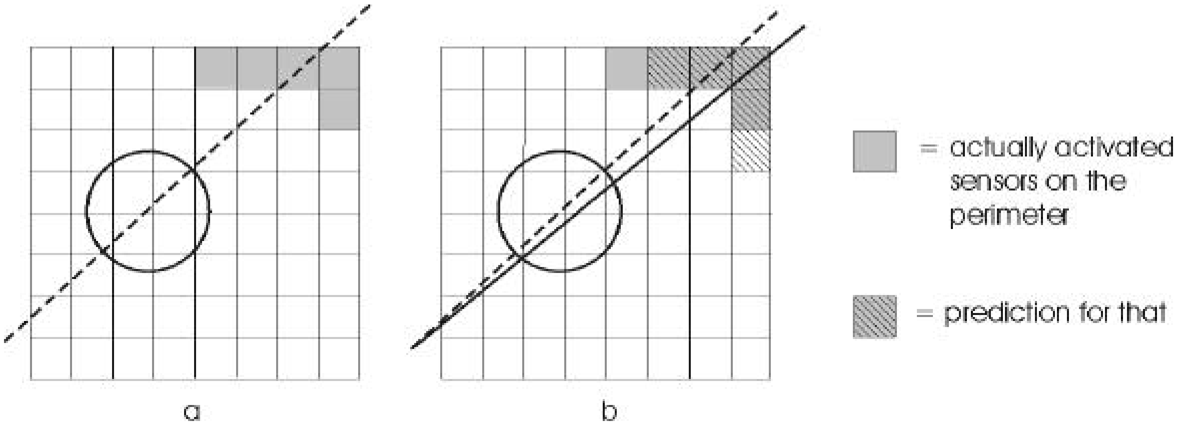

Figure 3 : Computation of movement direction. Dashed line is the trajectory of a moving ball. Sensors on the perimeter that will be activated by ≥ 50 % when the moving ball leaves the sensor field are marked in panel a. Sensors marked by stripes in panel b indicate a typical prediction of sensors on the perimeter that are going to be activated by ≥ 50 %, when the ball will leave the sensor field (time span into the future varies for this prediction between 100 and 150 ms, depending on the speed and angle of the object movement). Solid line in panel b represents the estimated direction of ball movement resulting from this prediction (its right end point is the average of sensors positions on the perimeter that are predicted to become ≥ 50 % activated). The angle between the dashed and solid line (average value 4.9° for test movements) is the error of this particular computation of movement direction by the simulated neural circuit.

Another readout pool was independently trained in a supervised manner to classify the moving object (ball or bar). It had an error of 0 % on 300 test examples of moving objects. The other readout pool that was trained in a supervised manner to estimate the speed of the moving bars and balls, which ranged from 30 to 50 units per second, made an average error of 1.48 units per second on 300 test examples. This shows that the same recurrent circuit that provides the input for the movement prediction can be used simultaneously by a basically unlimited number of other readouts, that are trained to extract completely different information about the visual input.

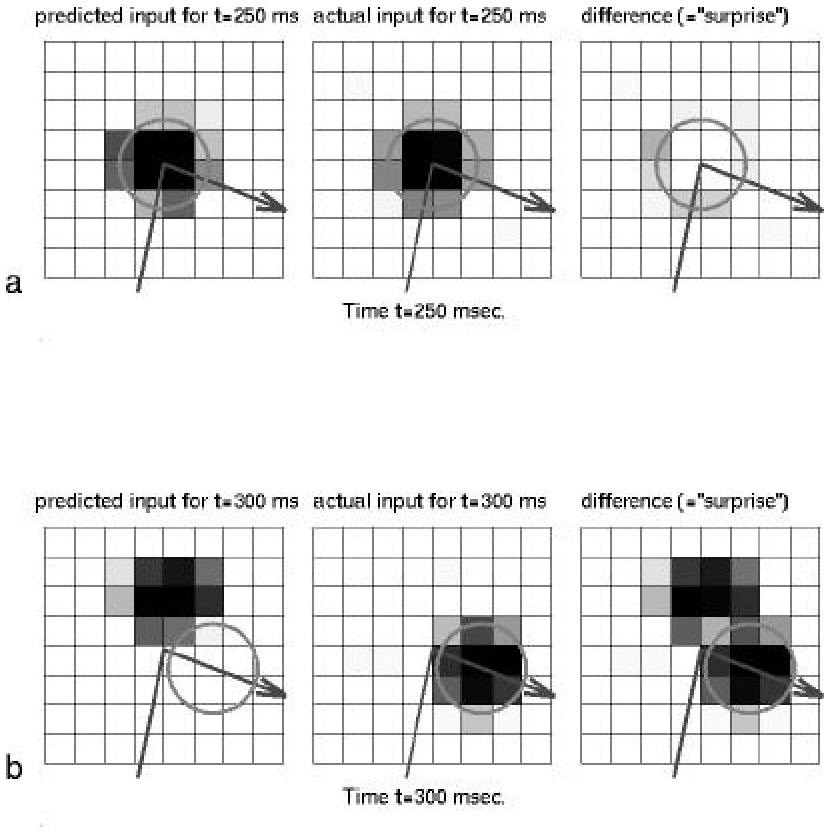

Figure 4: Novelty detection. Shown in panel a is the 50 ms prediction of future sensor activity made at time $t = 200$ ms, the actual sensor activation at time $t = 250$ ms, and for each sensor the absolute value of the prediction error. The ball changed its direction of movement at time $t = 250$ ms, resulting in a large error of the 50 ms prediction made at time 250 ms, which is shown on the l.h.s. of panel b. The comparison of this prediction with the actual sensor activation at time $t = 300$ ms (difference shown on the r.h.s. of panel b) provides all salient input needed for an analysis of this surprising event.

We also evaluated the 72 readout pools that had been trained to predict future sensor activity for a completely different movement, that was unrelated to those shown during training. We considered the response to object movements when the object changed its direction of movement while crossing the sensor field. Figure 4 shows the prediction error of the 36 neurons that predict sensor activation $\Delta T = 50$ ms later when the object changes its direction of movement at time $t = 250$ ms. The weights of these readout neurons resulted from a training procedure described before (see Figure 2), where only linear movement paths were shown. Panel b indicates that at time $t = 300$ ms, thus 50 ms after the object has changed its direction of movement, the prediction error is very high, much higher than during any straight movement. Hence one can easily discern that something unusual has happened. Furthermore the spatial distribution of prediction errors (r.h.s. of panel b in Figure 4) provides all the relevant information for analyzing this event. Note that the time point $t = 250$ ms where the moving object changed its direction had been picked arbitrarily in this experiment. Hence any-time novelty detection and analysis is supported by our neural circuit that had been trained in an unsupervised manner, although it had never been trained for novelty detection.

Finally, we show in Figure 5 what happens if from some arbitrarily chosen time point on (here $t = 125$ ms) the sensor input to the recurrent circuit is removed, and replaced by predictions of future inputs by the readout pools. More precisely, the time series of inputs (sampled every 5 ms) was replaced for each sensor after $t = 125$ ms by the preceding prediction of the corresponding readout pool (that had been trained for this prediction in an unsupervised manner as described before). Hence further predictions after time $t = 125$ ms are made based on an increasing portion of imagined rather than real inputs to the recurrent circuit. The resulting autonomously "imagined" continuation of the object movement is shown in panels b – d. It turned out that this imagined movement proceeded by 87.5 % faster than the initial "real" part of the movement. Panel e shows the firing activity of 100 neurons in the recurrent circuit for the case where the input arises from the "real" object movement, and panel f shows the firing activity of the same neurons when the "real" input is replaced after $t = 125$ ms by imagined (predicted) inputs.

4. **Discussion**

We have demonstrated through computer simulations that a radically different paradigm for processing dynamically changing visual input is in principle feasible. Instead of storing discrete frames in a suitably constructed datastructure, and then applying a time-consuming algorithm for extracting movement information from these frames, we have injected the time-varying (simulated) visual input continuously into a high dimensional dynamical system consisting of heterogeneous dynamic components. As dynamical system we took a generic cortical microcircuit model, with biologically realistic diverse dynamic components. Then we trained neural readouts to extract at any time t from the current output $x(t)$ of the dynamical system information about the continuous input stream that had recently entered the dynamical system, not in order to reconstruct that earlier input, but to output directly the computational targets which required that information. The theoretical potential of this general approach had previously been explored in [Maass et al., 2001] and [Haeusler et al., 2002].

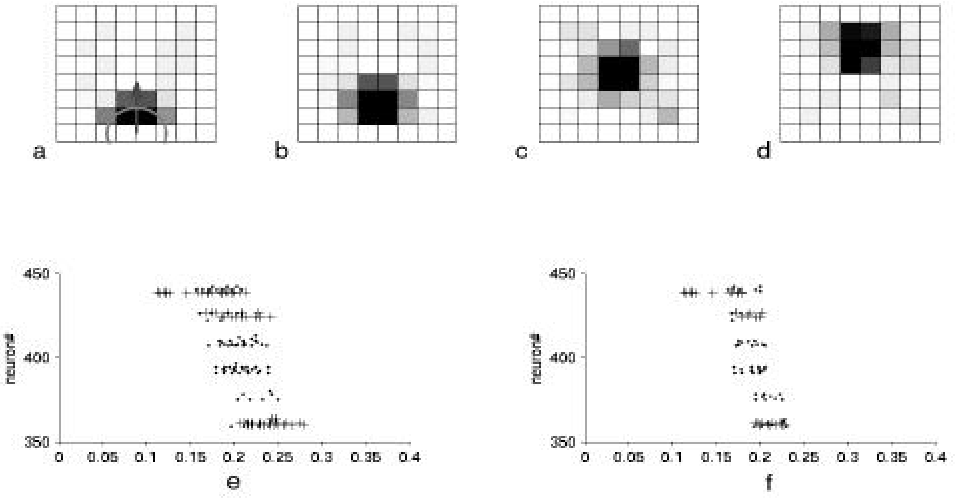

Figure 5: Imagined movement generated by the neural circuit. Panels a-d show the predictions of object positions at times $t = 130$ ms, 155 ms, 180 ms, 205 ms. Only the first prediction, shown in panel a, is based on sensor input. The predictions in panels b-d are primarily based on preceding input predictions, that were fed back as input into the recurrent neural circuit. This imagined movement happens to proceed faster than the actual movement which it continues, demonstrating that it is not constrained to have the same speed. Panel f shows the firing activity of a subset of 100 neurons in the recurrent neural circuit during this "imagined movement". Compared with the firing pattern of these neurons during a continuation of sensory input from the actual object movement after $t = 125$ ms (if it would continue on the same trajectory, with the same speed as at the beginning), shown in panel e, the firing pattern is very similar, but contracted in time.

In this way we have shown that generic neural microcircuits are in principle capable of learning in an autonomous manner to augment and structure the complex visual input stream that they receive: They can learn to predict individual components of the subsequent frames of typical "input movies", thereby allowing the system to focus both on more abstract and on surprising aspects of the input. For example, they can autonomously learn to extract the direction of movement of an object, which requires integration of information from many sensors ("pixels") and many frames of the input movie. Because of the diversity of moving objects, movement speeds, movement angles, and spatial offsets that occurred, it appears to be very difficult to construct explicitly any circuit of the same size that could achieve the same performance. Furthermore the prediction errors of our approach can be reduced by simply employing a larger generic recurrent circuit. On the other hand, given the complexity of this prediction task (for two different objects and a large diversity in movement directions and movement speeds), the recurrent circuit consisting of 768 neurons that we employed – which had not been constructed for this type of task – was doing already quite well. Its performance provides an interesting comparison to the analog VLSI circuit for motion analysis on a 7 × 7 array of sensors discussed in [Stocker and Douglas, 1999].

Whereas a circuit that would have been constructed for this particular task is likely to be specialized to a particular range of moving objects and movement speeds, the circuit that we have employed in our simulations is a completely generic circuit, consisting of randomly connected integrate-and-fire neurons, that has not at all been specialized for this task. Hence the same circuit could be used by other readouts for predicting completely different movements, for example curved trajectories. We also have demonstrated that it can at the same time be used by other readout neurons for completely different tasks, such as for example object classification. Furthermore we have shown that a generic neural circuit that has been trained in an unsupervised manner to predict future inputs automatically supports novelty detection when being exposed to new types of input movements. Finally we have demonstrated that if from some time point on the circuit input is replaced by input predictions that are fed back from neural readouts, the emerging sequence of further predictions on the basis of preceding predictions may generate a fast imagined continuation of a movement, triggered by the initial sequence of inputs from the beginning of that movement.

The results of this article are quite stable, and they work for a large variety of recurrent neural circuits and learning algorithms. In particular they can be implemented with the most realistic computer models for neural microcircuits that are currently known. Hence one could view them as starting points for building biologically realistic models of parts of the visual system which are not just conceptually interesting or which produce orientation selective cells, but which can really carry out a multitude of complex visual processing tasks. In our current work this paradigm is applied to real-time processing of actual visual input and input from infrared sensors of a mobile robot.

Other current work focuses on the combination of top down processing of expectations with bottom up processing of visual information – which makes biologically vision systems so powerful. Obviously our circuit models are ideally suited for such investigations, because in contrast to virtually all other circuits that have been constructed for solving vision tasks, the circuits considered in this article have not been chosen with a bias towards any particular direction of processing.

References

[Abbott and Blum, 1996] Abbott, L. F., and Blum, K. I. (1996) Functional significance of long-term potentiation for sequence learning and prediction, *Cerebral Cortex*, vol. 6, 406-416.

[Auer et al., 2002] Auer, P., Burgsteiner, H., and Maass, W. (2002) Reducing communication for distributed learning in neural networks. In *Proc. ICANN'2002*, 2002. Springer-Verlag.

[Buonomano and Merzenich, 1995] Buonomano, D. V., and Merzenich, M. M. (1995) Temporal information transformed into a spatial code by a neural network with realistic properties, *Science*, vol. 267, Feb. 1995, 1028-1030.

[Gupta et al., 2000] Gupta, A., Wang, Y., and Markram, H. (2000) Organizing principles for a diversity of GABAergic interneurons and synapses in the neocortex, *Science* 287, 2000, 273-278.

[Haeusler et al., 2002] Haeusler, S., Markram, H., and Maass, W. (2002) Observations on low dimensional readouts from the complex high dimensional dynamics of neural microcircuits, submitted for publication. Online available as # 137 on http://www.igi.tugraz.at/maass/publications.html.

[Haykin, 1999] Haykin, S. (1999) *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 2. ed.

[Jaeger, 2001] Jaeger, H. (2001) The “echo state” approach to analyzing and training recurrent neural networks, submitted for publication.

[Levy, 1996] Levy, W. B. (1996) A sequence predicting CA3 is a flexible associator that learns and uses context to solve hippocampal-like tasks, *Hippocampus*, vol. 6, 579-590.

[Maass et al., 2001] Maass, W., Natschlaeger, T., and Markram, H. (2001) Real-time computing without stable states: A new framework for neural computation based on perturbations, *Neural Computation* (in press). Online available as # 130 on http://www.igi.tugraz.at/maass/publications.html.

[Mallot, 2000] Mallot, H. A. (2000) *Computational Vision*, MIT-Press (Cambridge, MA).

[Markram et al., 1998] Markram, H., Wang, Y., and Tsodyks, M. (1998) Differential signaling via the same axon of neocortical pyramidal neurons, *Proc. Natl. Acad. Sci*., 95, 5323-5328.

[Rao and Sejnowski, 2000] Rao, R. P. N., and Sejnowski, T. J. (2000) Predictive sequence learning in recurrent neocortical circuits, *Advances in Neural Information Processing Systems* 12, (NIPS*99), 164-170, S. A. Solla, T. K. Leen, and K. R. Muller (Eds.), MIT Press.

[Schölkopf and Smola, 2002] Schölkopf, B., and Smola, A. J. (2002) *Learning with Kernels*, MIT-Press (Cambridge, MA).

[Stocker and Douglas, 1999] Stocker, A., and Douglas, R. (1999) Computation of smooth optical flow in a feedback connected analog network. *Advances in Neural Information Processing Systems* 11, (NIPS*98), 706-712.

[Tsodyks et al., 2000] Tsodyks, M., Uziel, A., and Markram, H. (2000) Synchrony generation in recurrent networks with frequency-dependent synapses, *J. Neuroscience*, Vol. 20, RC50.

[Vapnik, 1998] Vapnik, V. N. (1998) *Statistical Learning Theory*, John Wiley (New York).

Appendix

We employed a generic randomly drawn recurrent circuit consisting of 768 integrate-and-fire neurons. 20 % of these neurons were randomly chosen to be inhibitory. Connection probabilities and other parameters were chosen to reflect empirical data from neurobiology. Neuron parameters were chosen as in [Tsodyks et al., 2000]: membrane time constant 30ms, absolute refractory period 3ms (excitatory neurons), 2ms (inhibitory neurons). The threshold of each neuron was chosen to be -55 mV, the resting membrane potential and the reset voltage had a value of -56.5 mV. A value of 1 MΩ was chosen for the input resistance. In addition, each neuron received a background current of 13.5 nA.

The probability of a synaptic connection from neuron *a* to neuron *b* (as well as that of a synaptic connection from neuron *b* to neuron *a*) was defined as $C \cdot e^{-(D(a,b)/l)^2}$, where l is a parameter which controls both the average number of connections and the average distance between neurons that are synaptically connected. We assumed that the 768 neurons were located on the integer points of a $16 \times 16 \times 3$ cube in space, where $D(a,b)$ is the Euclidean distance between neurons a and b. Depending on whether a and b were excitatory (*E*) or inhibitory (*I*), the value of C was 0.4 (EE), 0.2 (EI), 0.5 (IE), 0.1 (II).

In the case of a synaptic connection from a to b we modeled the synaptic dynamics according to the model proposed in [Markram et al., 1998], with the synaptic parameters *U* (use), *D* (time constant for depression), *F* (time constant for facilitation) randomly chosen from Gaussian distributions that were based on empirical data reported for such connections in [Gupta et al., 2000] and [Markram et al., 1998]. Depending on whether a, b were excitatory (*E*) or inhibitory (*I*), the mean values of these three parameters (with *D*, *F* expressed in second, s) were chosen to be 0.5, 1.1, 0.05 (EE), 0.05, 0.125, 1.2 (EI), 0.25, 0.7, 0.02 (IE), 0.32, 0.144, 0.06 (II). The maximal amplitude *A* (in nA) for postsynaptic currents was chosen to be 30 (EE), 60 (EI), -19 (IE), -19 (II). The SD of each synaptic parameter was chosen to be 50 % of its mean (with negative values replaced by values chosen from an appropriate uniform distribution). Synaptic connections from sensors (inputs) to neurons in the circuit were static, with their amplitude randomly drawn from a Gaussian distribution, the mean for their maximal amplitude *A* had a value of 3 nA, and the SD was 60 % of the mean.

The temporal evolution of postsynaptic currents was modeled by an exponential decay exp(-t/τs) with τs=3ms (τs=6ms) for excitatory (inhibitory) synapses. The transmission delays between neurons were chosen uniformly to be 1.5 ms (EE), 0.7 ms (EI), and 0.8 for the other connections (in order to demonstrate that large and diverse delays are not necessary).