Synchronization, Coherence-Detection and Three-Dimensional Vision

Rolf D. Henkel
henkel@physik.uni-bremen.de
http://axon.physik.uni-bremen.de

University of Bremen
Institute for Theoretical Physics
D-28334 Bremen, Germany

20th November 2000

Abstract

A new functional role for spiking neurons is proposed, considered necessary to convert noisy sensory data into meaningful and stable perceptions. Percept creation and validation is performed by a dynamical process of coherence detection between neural signals. The crucial operational step of the network is the interaction of neural oscillators in the weak-coupling limit, which realizes coherence detection dynamically by selective synchronization of neural responses. A robust estimate of the incoming signals is transmitted as modulation frequency of the output current of the coherence-detecting layer, and a validation measure is given by the modulation depth of this current. As a real world example of these ideas, a neural network is presented solving the task of stereo vision with real image data. It combines operations of time-averaging, rate-coding neurons with integrate-and-fire-neurons, which calculate a disparity map of a scene by partially synchronizing their spike trains.

Keywords: Synchronization, Coherence, Neural Code, Neural Computations, Robust Estimators, Three-dimensional Vision, Integrate-And-Fire-Neurons.

1 Introduction

Despite extensive ongoing research, there is no general agreement about how information is transmitted and processed in biological neural networks. Many neurobiological experiments suggest that there is a strong correlation between the rate of action potential production and the stimulus strength, at least in early sensory areas. In addition, one also finds changing temporal correlations between spike trains of neurons by varying stimuli [1, 2, 3]. Thus, there are indications that the average frequency of spike trains as well as their temporal correlations carry important information within the neural systems. In this paper, these two coding concepts are combined and attributed to two complementary modes of network operations: one mode utilizing the firing rates of neurons for continuous representation and calculation of stimulus properties, the other mode relying on the temporal timing of spike trains to construct and mark stable percepts by a dynamical process of coherence-detection.

As a worked-out example, a network composed of rate- and spike-coding neurons is presented. The network performs disparity calculations between two stereo images, solving in effect the task of depth perception. As shown here, both operational modes are necessary for the successful operation of the network.

2 Construction by Coherence-Detection

2.1 Ambiguity of Single Neural Responses

One of the most important objectives any animal faces is the faithful representation of at least some partial aspects of its external environment. Only if faithful, stable and, if possible, validated representations are available, purposeful reactions can be deduced in an ever changing environment.

By itself, the response of a single neuron is not a good carrier of information. There are two main reasons for this: the limited coding range any single neuron has, and the mixing of independent feature dimensions within single neurons.

The range of stimulus values which a single neuron can represent or process is restricted for two different reasons: (1) even though the connection numbers in cortex may be rather high, any single neuron can sample information only from a limited number of sensory receptors or other neurons. So by design, neurons have a limited access to available information, restricting trivially the “field of view” of neurons; (2) a further limiting factor is the input-output relationship realizable by a neuron, i.e, the neuronal transfer function. At one end of the scale, resource restrictions lead to saturation effects limiting neural responses in the range of high neural activity. On the other end of the scale, noise and threshold characteristics restrict the range of representable values from below.

To make things even more complicated, most neurons not only respond to variations along a single, fixed stimulus quality, but usually to many, rather different stimulus characteristics [4]. For example, a single neuron might change its firing rate if the orientation of the stimulus, or the texture, or the contrast, or some combination of these stimulus properties is varied.

In contrast to most lab experiments, all these stimulus qualities are available and will vary independently in normal visual scenes. Summarising, no single neuron can give an accurate representation of any specific stimulus aspect of the external world. For this, signals from several neurons must be appropriately combined.

2.2 Grouping by Coherence-Detection

A priori it is not clear which neural signals out of a given pool should be combined into a good estimate for a specific stimulus property.

A simple weighted average, or any similar linear combination, of neural signals will *not* work under most circumstances, because such an estimate will be subject to noisy interference by all neurons which are processing data not related to the stimulus property in question.

For example, the signal of a neuron which codes mainly stimulus orientation, but is also sensitive to colour changes, might correctly reflect the orientation of an edge in some cases, but might show a mixture of orientation and colour readings in other cases. Clearly, one would want to include the data of this neuron in orientation estimates only if the signal is not disturbed by colour.

A simple weighted average of neural signals will fail also if some of the neurons are driven out of their working range. Any estimate based partially on neural signals which are clamped by saturation- or threshold-effects will inevitably be biased.

In summary, any good estimation process should be highly selective concerning the question of which neural responses should be combined and which should be dropped in an estimate.

There is problem surfacing here: The decision of which specific subgroup of neurons should be included in an stable estimate, and which should not, varies greatly with the actual stimulus situation. It does not seem to be an easy task for a neural network to discriminate between noise and signal in spike trains coming from a pool of neurons.

However, there is a rather simple solution to this problem, which uses the fact that many neurons have receptive fields overlapping with each other, i.e. are looking at nearly the *same* aspect of the external world. This allows the external world to be used as reference.

Any specific stimulus situation will split a given pool of neurons into two disjunct classes: one class which codes the stimulus value more or less confidently (class $\mathcal{A}$), and the rest of the neurons, which display either no relation or a wrong relationship between their neural signals and the stimulus value in question (class $\mathcal{B}$). Overlapping receptive fields ensure that class $\mathcal{A}$ will consist of more than a single neuron, and this makes it possible to detect this class by a simple "neural voting" process.

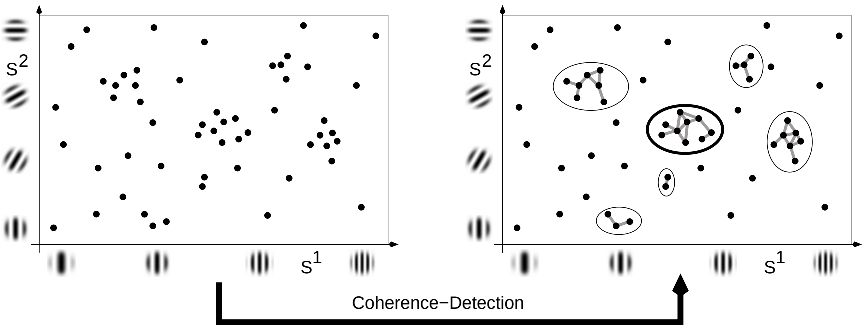

Figure 1: Neural signals coding incoming data and noise in a two-dimensional feature space. Coherence detection connects signals similar enough with each other, forming several distinct coherence clusters in the example. A signal average over the largest cluster will result in a stable estimate derived from the set of noisy signals.

More formally, if s_i stands for the neural response of the i^{th} neuron, and if one searches within the whole pool of neurons for coherence clusters $\mathcal{C}$, defined as all sets having pairwise similar responses,

$$\mathcal{C} = \{i, j \;:\; |s_i - s_j| < \epsilon\} \;, \tag{1}$$

the largest cluster found by this operation will be class $\mathcal{A}$. A simple average over this coherence cluster,

$$s = \frac{1}{n_\mathcal{A}} \sum_{i \in \mathcal{A}} s_i \;, \tag{2}$$

where $n_\mathcal{A}$ is the number of neurons in class $\mathcal{A}$, gives a stable estimate of the neural signals coming from the coherence cluster.

In this process, the coherence threshold ϵ defines the amount of noise tolerated within the coherence cluster – neuronal signals deviating more than ϵ from the signals in class $\mathcal{A}$ will be regarded as outliers coding noise or other information.

The actual size of the coherence cluster depends on the choice of ϵ. For $\epsilon \to 0$, the number of neurons able to participate in the coherence cluster decreases. In this case, the estimate will be based on fewer units and will become less reliable. For $\epsilon \to \infty$, all neurons, including the noisy ones, can participate in the coherence cluster. In this limit, equation (2) computes simply the average of all signals.

The coherence threshold ϵ might be chosen adaptively, depending on actual stimulus and noise statistics. Normally however, classes $\mathcal{A}$ (signal) and $\mathcal{B}$ (noise) are clearly separated, so the actual value of ϵ is not critical (compare results in Sections 2.6, 3.5).

For a fixed coherence threshold ϵ, the number of neurons participating in the coherence cluster, $n_\mathcal{A}$, compared to the total number of neurons in the pool, n_{total}, is an indication of the quality of the coherence estimate and can be used as a validation measure (compare Section 2.6).

So, the simple answer to the selection problem posed above turns out to be: Let the neurons do the selection themself via the process of coherence detection, which is itself driven by the coherence of the external world.

2.3 Coherence-Detection by Spike-Synchronization

In operational terms, the process of coherence detection defined above is complicated, and the question arises how this process might be realized with available neural hardware. One answer to this question is simple, and closely connected to the behaviour of weakly coupled neural oscillators.

It is well known that a network of identical oscillators, coupled together by appropriate links, will synchronize under a wide variety of conditions [5, 6, 7, 8, 9, 10]. The argument rests: *1)* on the possibility of reducing most oscillator models under appropriate conditions to an equivalent model of phase oscillators; and *2)* a generic condensation process leading towards synchronization for this type of oscillator.

Irrespective of the internal complexity of an oscillator, the temporal development of any isolated oscillatory unit can be described by a phasevariable, increasing from 0 to 2π in one oscillation of the unit. If several oscillators are coupled together, the description of the dynamics by a set of phasevariables might not be appropriate. However, if the coupling between the oscillatory units is weak enough, the dynamic of any single unit will only be slightly disturbed, compared to the uncoupled case. In this case a description of the whole system by a set of phasevariables is still acceptable.

If coupled phaseoscillators have the same oscillation frequency, they easily synchronize. However, the situation becomes more complicated if the oscillatory units have slightly different frequencies. The tendency for synchronization between any two oscillators is then counteracted by the difference between them. For a given weak coupling, there exist a maximum possible deviation between two oscillators to synchronize [5, 6, 7, 11, 12, 13, 14, 15, 16, 17].

In summary, neural oscillators connected by weak coupling will display selective locking behaviour, depending on the magnitude of differences between them. Oscillators with minor differences in frequency will lock their signals, while oscillators with larger differences will not. Of course, such a selective locking behaviour is an exact replica of the search for the coherence cluster in the coherence-detection process defined above in equation (1).

While many neural circuits display oscillatory behaviour and might be used as basic computing elements for coherence detection, single neurons are used in this paper. The neurons are modeled as standard leaky integrate-and-fire neurons [see Appendix A.2.1 for details]. If supplied with an external current, these neurons spike regularly, with a rate depending on the strength of the input current.

The neurons are assumed to drive other neurons with an exponentially decaying current, i.e., each spike supplies an input current $\sim \exp(-t/\tau_s)$ to all postsynaptic neurons. Fig. 2 shows the temporal dynamics of a network of such integrate-and-fire neurons, with input currents reflecting the actual estimates of a group of disparity estimators (details in the Appendix). When left uncoupled (Fig. 2A), no obvious structure is visible in the spike trains of the neurons. By introducing a global, but *weak* synaptic coupling between the neurons, the network switches to a distinctively different behaviour (Fig. 2B). The pool of neurons quickly split into two separate sets, one a synchronous spiking cluster, which clearly shows up as sequence of vertical traces in the spike time diagram, and the rest of the pool, which stays asynchronous with the coherent cluster.

The attainment of synchronization under the weak-coupling paradigm is rapid, in this simulation within about 10 spike cycles of the coherence pool. Most important is that the decision of which neuron participates in the coherence cluster and which does not is stimulus-dependent: another stimulus situation changes the spike rates of the neurons, and in turn redistributes the synchronous cluster to other neurons within the pool (Fig. 2B, C).

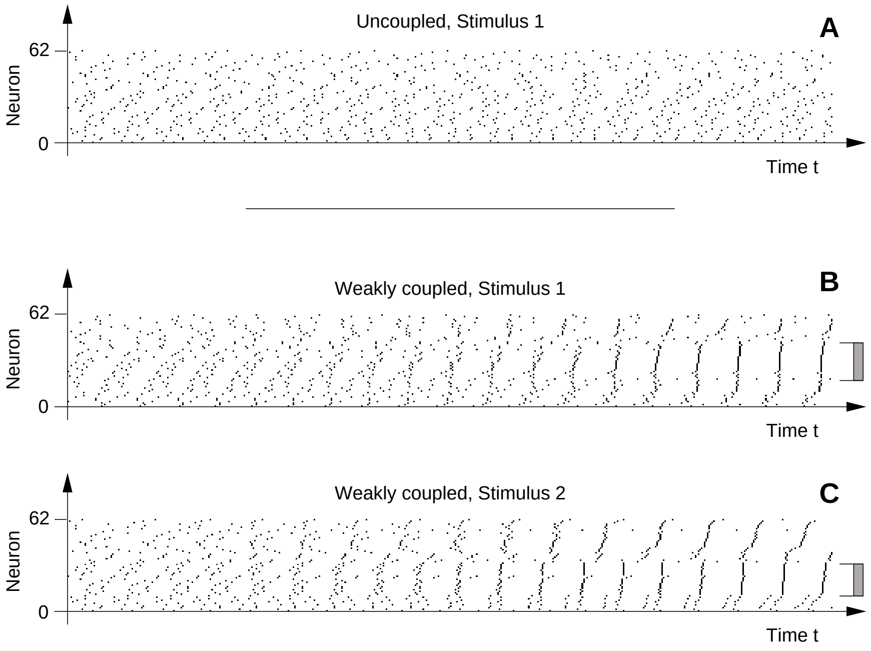

Figure 2: Spike events in a pool of a 62 integrate-and-fire neurons coding disparity estimates along a common view direction. Spike trains are sorted according to their firing rate. If left uncoupled (A), no clear structure is visible in the data. With weak coupling (B, C), the neurons belonging to the coherence cluster quickly synchronize and appear as vertical blocks in the spike train display. For different stimuli, different groups of neurons are forming.

The instantaneous stimulus value detected by the network is assumed to be coded by the spike frequency of the coherence cluster. To a first approximation, this frequency equals the average of the original, uncoupled spike frequencies of the units participating in the cluster, i.e., the frequencies before synchronization [6]. Thus, formula (2) of the coherence detection scheme is realized dynamically through the process of synchronization within the coherence cluster.

2.4 Marking by a synchronous code

As soon as the neurons within the coherence cluster fire synchronously, the characteristics of the total output current of the network changes (Fig. 3). With weak coupling switched off, the fluctuating amplitude of the output current only reflects random coincidences of spikes (time < 20 in Fig. 3). Turning the weak coupling on between neurons (at times $>$ 20) the synchronization of the coherence cluster leads to a strong periodic modulation of the output current, easily detectable by subsequent layers. Since the depth of this modulation depends on the relative percentage of neurons in the coherent cluster, it is a good indication of the validity of the estimate.

The marking of neural groups by a synchronous time code is an old idea [18, 19] and has been subject to various modeling attempts over time. But the dynamical coherence detection scheme proposed here differs in important aspects from previous ideas.

Most of the other schemes use a much stronger coupling between neurons, compared to the weak-coupling paradigm developed here. But with stronger coupling, the *whole* network of oscillators quickly synchronizes, and all information about the stimulus value it codes is lost.

Avoidance of full synchronization defines an upper limit for the term “weak-coupling” introduced here. Dynamical coherence detection is only possible within a range of inter-

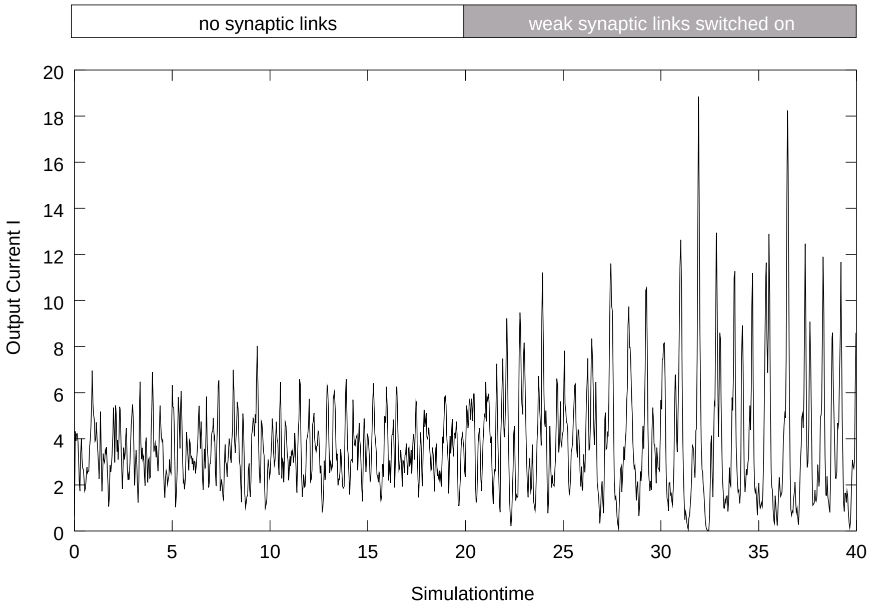

Figure 3: The total current injected into higher processing layers from a pool of integrate-and-fire-neurons. Starting the simulation without coupling between the neurons, the output current randomly fluctuates. At time $t = 20$ weak synaptic links are switched on, the output current of the neural pool soon displays a strong oscillatory component coming from the coherence cluster. The modulation depth of this current is an indirect indication of the size of the coherence cluster.

mediate coupling strengths, located between the two extremes of “independent oscillators” and “fully synchronized network”.

Too strong coupling between oscillatory units can further result in complicated network behaviour, including chaotic dynamics and oscillator death [7, 20]. With strong coupling the internal characteristics of the oscillatory units become important – and they can no longer be described as interacting phaseoscillators.

Many of the network models proposed in the literature use oscillatory elements with the *same* frequency of oscillation, facilitating locking of all oscillators. Stimulus values are coded in all these models not by intrinsic oscillator properties, but by the *place* of the activated elements in the networks. Most of the time, a binary valued stimulus is used as input, to switch the oscillatory elements “on” or “off”. Connected by preset [21, 22, 23, 24, 25, 26, 27, 28] or prelearned [29, 30] synaptic couplings, the activated and connected oscillators quickly synchronize; the non-activated units of the network remain in the quiescent state.

Thus, synchronization is used in these approaches not as a computational tool, but only to mark a preselected group of oscillatory elements. Other efforts to model feature-binding [19, 31, 32, 33, 34, 35, 36] rely on fast synaptic links, modified by activity-driven learning rules. In effect, these learning rules couple only neural units with a common high activity together. Thus, from a computational point of view, models utilizing fast synaptic links are similar to models utilizing oscillatory units being switched “on” or “off”. Primary computational units are the dynamical synapses, and synchronization is used only to mark the group of synaptically coupled neurons.

Dynamical coherence detection uses quite different concepts. Here, stimulus values are continously represented as an internal property of the oscillatory units, namely the frequency of oscillations, and synchronization is used as computational tool *to detect* the coherence cluster. Synaptic links are necessarily weak, barely changing the normal, stimulus-driven activity of single units, and fixed, at least on short timescales.

2.5 Preknowledge coded in synaptic links

Within the context of dynamical coherence detection, the synaptic link structure has a different computational purpose. In any given cortical area, specific neural units will be sometimes co-activated by incoming stimuli, while others will never be. Which units will be co-activated and which not will depend on the statistical nature of the input signals. A synaptic link structure can adapt to this statistical structure of the incoming neural signals by increasing the connection strength between units which are often co-active. With such an adapting link structure, new, but similar stimuli will be able to recruit more neural units into a coherence cluster, which can stabilize the estimates derived by the coherence detection process.

For example, in the auditory cortex co-activation of neurons responding to higher harmonics of a single frequency is to be expected quite often, since most auditory stimuli in the real world are composed of a base frequency plus higher harmonics. Neurons responding to non-harmonics might also be sometimes co-active – but only by chance, and not in general. In the same way, disparity sensitive neurons in the visual cortex will sometimes be co-activate if they cover the same view-direction with their receptive fields. For neurons responding to different view-directions, co-activation will be purely coincidental.

In this way, an adaptive synaptic link structure can adopt to the main characteristics of the input signals. The important point is that under the weak-coupling paradigm, a given synaptic link is *not enough* to group neurons together. The synaptic link structure of a network presents only *possibilities of interaction*, but at any given moment most of these links will not be dynamically utilized. Usually, only *one* of all possible groupings supported by the synaptic link structure will be instantiated by an actual stimulus.

Fig. 2 shows an example of such a stimulus-dependent utilization of synaptic links. In this simulation, all neurons of the network have synaptic links with each other. However, for a given stimulus, only a small subgroup of all the neurons synchronize; which one depends on the current stimulus value.

Conceptually, dynamical coherence detection and synaptic link modifications operate on two different time scales. On a short time scale, coherence detection analyses the current stimulus situation; on a much longer time scale, synaptic link modifications record the long-time statistics of the incoming signals.

2.6 Robust Estimation

If confronted with severely corrupted data, the behaviour of an estimator can be described by its breakdown point β. This is the smallest fraction of outliers, i.e. of data not obeying the assumed noise model, which can cause the estimator to produce arbitrarily bad results.

As a simple example, consider n measurements $x_i = s + \eta_i$ of a signal s corrupted by additive noise η_i. We further assume the noise to be Gaussian noise, i.e., $P(\eta) \sim \exp(-\eta^2/2\sigma^2)$. The maximum likelihood estimate s^* of the signal is than given by a least-square fit, which in our example yields the mean of the measurements, $s^* = 1/n \sum_i x_i$, as estimation formula.

However, the widely used assumption of signals corrupted by additive Gaussian noise is slightly questionable. Since extremely low probability values are assigned to large values of noise, these values distort the estimate if they occur. Indeed, in our example, a single large deviation in one the measurements, say x_k, will cause the mean s^* to deviate arbitrarily far from the true value. It follows that this estimator has a breakdown point of $\beta = 1/n$ and asymptotically, i.e. for $n \to \infty$, a breakdown point of 0. This is a property common to all least-square-based estimators [37].

There exist other classes of estimators, called robust estimators, which can tolerate a non-zero percentage of outliers. They are typically non-linear estimation schemes and therefore hard to implement. A classical example of a robust estimator is the median of n data points, which is insensitive to a few large outliers in its set of measurements. In fact, this estimator has a breakdown point of 0.5, i.e., as much as 50% of the data can be corrupted before this estimator fails.

Dynamical coherence detection also realizes a robust estimation scheme. This is because the coherence detection process connects only a small subset of the whole neural population to the coherence cluster, namely the ones which agree in their estimates with each other. All other neurons, representing the outliers, stay asynchronous with the coherence cluster.

Since the neurons coding outliers stay asynchronous, the contribution of the outliers to the total output current will be a current which fluctuates approximately with an amplitude of $1/\sqrt{n}$, with n being the number of incoherently firing neurons. This part of the total output current has no detectable time structure, i.e. it fluctuates randomly in time. Instead, the coherence cluster contributes an oscillating current with an amplitude $\sim n_{\mathcal{C}}$, with $n_{\mathcal{C}}$ being the number of neurons in the coherence cluster. The output current of the coherence network is thus composed of an oscillatory component, reflecting the correlated dynamics of the coherence cluster, plus a small random component, caused by the noisy rest of the pool (compare Fig. 3).

It is rather easy for an appropriately designed output layer to lock exclusively onto the oscillatory component, discarding the data from all non-coherent neurons. This behaviour is highly advantageous for the process of percept creation – probably only robust estimators are able to extract stable perceptions out of noisy data, which is recorded, transmitted and analyzed by unreliable neural hardware.

Fig. 4, A-D, compares the performance of a coherence-based network to a neural network transmitting only the average of the incoming signals to subsequent layers. In the experiment, a varying percentage of the neurons in the input layer is allocated to transmit the signal, a noisy sinusoidal. The rest of the neural population in the input layer is driven by random signals, much larger than the sinusoidal signal variation.

In the experiment, the size of the signal-transmitting group of neurons is varied between 0 and 100% of the total number of neurons in the input layer. The output layer of the network simply responds to the average current coming from the input layer [compare equations (11, 13) in the Appendix]. Thus, by switching off the interlayer synaptic couplings of the input layer, the network calculates the mean of the incoming signals, realizing the maximum likelihood estimator for Gaussian noise.

As expected, the network without interlayer links (Fig. 4B) can not follow the incoming signal correctly if the percentage of neurons carrying the signal is too low compared with the percentage of noisy neurons. In addition, the modulation of the output current gives no hint of the varying signal quality which the maximum likelihood estimator for Gaussian noise is delivering to subsequent network layers. The modulation depth stays largely constant (Fig. 4D).

This situation changes drastically if one switches on the interlayer synaptic links in the coherence layer. As can be seen by referring to Fig. 4A, the network acts now as a robust estimator: it locks onto the signal, even if the number of outliers exceeds the number of signal-carrying neurons. In addition, the modulation depth of the output current (Fig 4C) rises monotonically with the number of signal transmitting neurons, thus indicating increasing confidence in the estimate.

The decision of which neurons can participate in the coherence cluster and which neuron will be assigned as "coding an outlier" depends on the coherence threshold ϵ in equation (1). In the dynamical coherence detection scheme realized with neural oscillators, the coherence threshold ϵ is replaced the interlayer coupling constant $w_{\mathcal{CC}}$ [compare equations (11, 12) in the Appendix]. In (Fig. 4E) network behaviour is analyzed while this coupling constant is varied. The number of signal carrying neurons was fixed in this simulation to 33% of the total neuron population.

With $w_{\mathcal{CC}} \approx 0$, the sinusoidal signal variation is not transmitted faithfully through the network, as expected. Around $w_{\mathcal{CC}} \approx 0.2$, the system starts to lock onto the signal, and at $w_{\mathcal{CC}} \approx 0.7$ the modulation depth rises sharply to a higher value (Fig 4F). This indicates

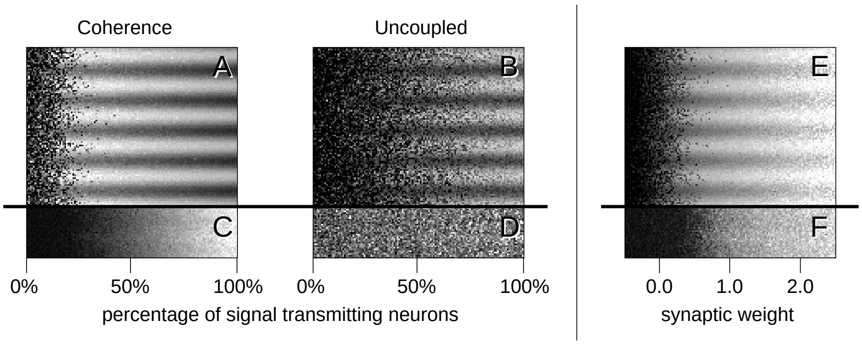

Figure 4: Signaltransmission through a coherence layer (A) and a layer of uncoupled neurons (B), in a noisy environment. The increasing modulation depth of the output current of the coherence layer (C) reflects the increasing signal fidelity. In the uncoupled case (D), the modulation depth carries no information. Coherence detection is moderated by the coupling constant in the coherence layer; there exists a broad range of values where the coherence detection layer can lock onto the signal (E). As more and more neurons are drawn into the coherence cluster, the modulation depth of the output current increases. For details, refer to main text.

that at this value of the coupling constant, all neurons belonging to the coherence cluster are able to synchronize. At values $w_{CC} > 1.5$, neurons coding noise are beginning to be drawn into the coherence cluster. This causes a more noisy estimate of the coherence network for these values of w_{CC} and indicates the limit of the weak-coupling regime.

In summary, the introduction of weak synaptic links between neurons allows for the dynamical computation of coherence between neural signals, realizing a robust estimator of incoming sensory stimuli. In this context, is interesting to note that the experimental data in [38] suggests the use of robust estimation in a task related to stereo vision.

3 A Worked-Out Example: Depth-Perception

As an example of the concepts discussed in Section 2, in this section a neural network is presented which solves the task of depth perception from stereoscopic images.

Depth perception by animals through stereo vision requires overlapping, binocular visual fields. With both eyes looking at the same part of a scene, but from two slightly different perspectives, objects in the left retinal image will be shifted relative to their position in the right retinal image. This shift will depend on the distance of the objects from the observer. If one fixates a point at infinity, the amount of shift turns out to be inversely proportional to the distance of objects. Objects at infinity will not be shifted, while nearer objects will have larger horizontal shifts. Estimating these shifts (disparities) between the two stereo views makes it possible to recover the three-dimensional structure of the scene.

The network developed in this section performs the task of disparity estimation by a combination of rate- and spike-coding neurons. It consists of two parts: (1) layers of simple and complex cells, calculating raw disparity estimates out of the input images; and (2) a coherence detecting network, combining the raw, but noisy estimates from the initial network layers into a final, stable disparity map. The layers of simple and complex cells are modeled by rate-coding neurons, but the coherence-network needs exact spike timing to function properly.

3.1 Stereovision, Optical Flow and Textures

In visual perception, many seemingly unrelated tasks are, in fact, very similar in their structure. For example, stereo vision, optical flow estimation and texture analysis are all closely related to each other. This can be seen as follows. If one assumes, for simplicity, a static scene, the slightly disparate images recorded by the two eyes of a binocular animal can also be sampled by a camera moving slowly from the position of the left eye to the position of the right eye (Fig. 5). In the process, the scene creates a characteristic space-time intensity pattern in the moving camera, with the texture pattern reflecting the arrangement of objects in the scene. More precisely, in this setup, the local orientation of the recorded space-time texture has a direct correlation with the observed distances of objects.

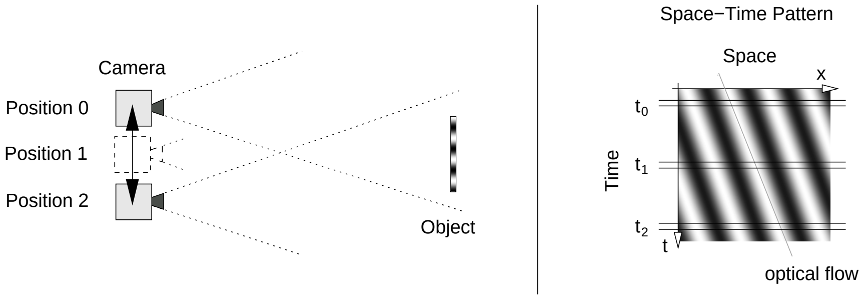

Figure 5: The two stereo images which a binocular animal sees are sample slices (t_0, t_2) out of a full space-time texture, which would be recorded by a camera moving between the positions of the two eyes.

The local space-time texture orientation is usually called “optical flow”. Thus, stereo vision can be regarded as a slightly more complicated optical flow estimation problem, having available only two image slices out of the full space-time texture. Further, optical flow estimation is, in turn, nothing more than the analysis of local texture directions in space-time patterns. So in retrospect, it is not too suprising that neural models proposed for neural texture analysis [39, 40], optical flow estimation [41] and stereo vision [42] are very similar in structure.

3.2 Input Layers of the Network

Textures seem to be analyzed by humans mainly along the dimensions “direction” and “granularity” [43]. Granularity refers here to textures with no prominent direction, but similar spatial texture variation. In the case of stereo vision, we are only interested in texture directions, indicating local image shifts, i.e., distances of objects.

Texture directions can be analyzed quite simply with neural hardware. Transforming a texture with one or several prominent texture directions into Fourier space, the texture directions show up as distinct spikes in the energy spectrum of the signal (Fig. 6).

Spikes in the spectrum can easily be detected by local measurements in Fourier space. For example, one might sample the local energy available in blobs placed around a circle. The blob with the largest energy content will indicate the main texture-direction (Fig. 6, bottom left).

Local texture energy can also be measured directly in the original signal space. Since the Fourier transformation of a Gaussian is a Gabor function and vice versa, the masking of signal energy in Fourier space by a Gaussian is equivalent to the convolution of the original signal with two Gabor filters, each related to each other by a phase shift of $\pi/2$. Filter functions of this type are called quadrature-filters (or Hilbert transform pairs). Squaring and adding together the resulting filter amplitudes gives a local measure of signal energy (filtered signals in Fig. 6).

The important point is that these filter kernels and nonlinear point operations easily map to receptive field profiles and transfer functions of simple and complex cells in the visual cortex. In disparity space only two slices out of the full space-time texture of the moving camera are available (compare Fig. 5), so the two-dimensional filter kernels used

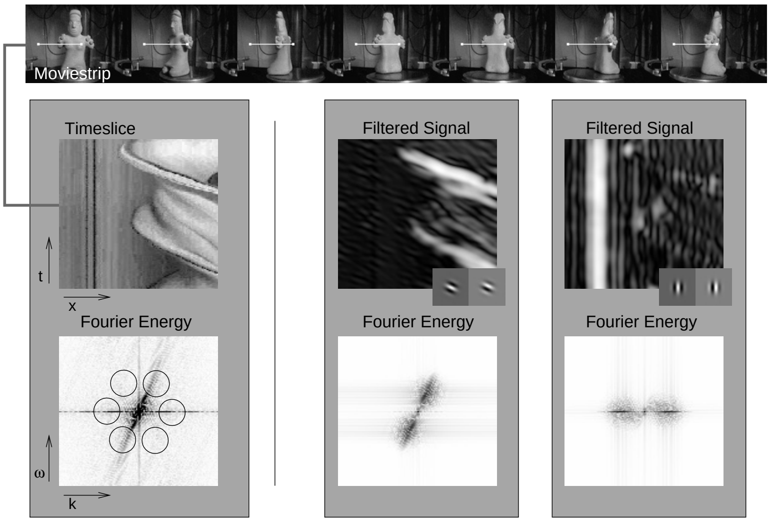

Figure 6: Cutting space-time slices out of a moviestrip (top line) creates characteristic flow patterns (timeslice), which show up as spikes in Fourier space. Local measurements of the Fourier energy can be used to detect the presence of these spikes. A simple way to measure local energies is the filtering of the timeslice by matched filters (right displays). Using Gabor filter patches in quadrature (small insets) is equivalent to masking Fourier energy with blob-like gaussian envelopes in Fourier space. Note that the filter ed signals depend also on the local image contrast.

in Fig. 6 are reduced to two simple one-dimensional filter profiles, convolving data either from the left or from the right eye. To compute the local energy, the signals coming from the quadrature-paired filters have to be squared and summed. This results in a circuit identical in structure to the one sketched in Fig. 7: units S with Gabor-like receptive fields sample data from the left and right retinae, and the squared output of these units is summed by a energy unit C, giving finally local texture energy (only this time in disparity-space). Interestingly, a circuit structurally equivalent to the one in Fig. 7 was proposed in [42] to account for experimental data measured from complex cell recordings in the visual cortex, with units S representing simple cells, and unit C representing a complex cell.

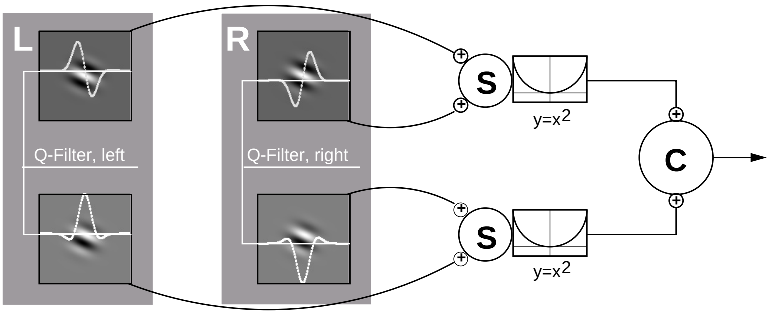

Figure 7: Spikes in Fourier space can be detected by local filter operations, having a close relationship with circuits proposed for the first stages of visual cortex. Simple cells S sample data from the left and right retinae, with filter profile which are essentially slices out of a two-dimensional Gabor function (white traces). Two simple cell in quadrature configuration fed their squared signals into the complex cell C, which calculates local Fourier energy (compare [42]).

Note that the raw local energies calculated by any neural circuit similar to the one in Fig. 7 can not be used directly for texture analysis, since the estimated energies depend also on local image contrast (compare Fig. 6). To deduce texture direction, one has to perform either a maximum detection around a circle in Fourier space, as already discussed, or some kind of contrast normalization. Adapting these two possibilities to the task of disparity estimation, we recover either the approaches by Qian (disparity estimation via maximum-detection, in [44]) or Adelson & Bergen (normalization, formulated in the context of optical flow estimation, in [41]).

In the network simulation presented here, disparity estimators derived from the original optical flow estimator of Adelson & Bergen [41] are used. This means that the difference between the output of two complex cells (estimating left and right disparity energies) is normalized by the output of a corresponding complex cell (measuring local contrast). For details, see the Appendix.

3.3 Coherence-Detection

Combining complex cell responses as described in the previous section, one obtains neural units sensitive only to local disparity. However, as discussed, any single disparity unit on its own is rather useless, since there is no way to decide whether a coded value is a true disparity estimate or just a random reading.

Actually, one can show that *any conceivable* disparity estimator, not only the ones described above, will fail to estimate disparity values correctly most of the time [45]! In the worst-case scenario, the range of disparity values any basic estimator can faithfully calculate is *less than the receptor spacing* in the retina.

At this point, the need for a special process arises which is able to derive a valid and stable estimate from the noisy data of a pool of disparity units. As we have argued, any weighted linear combination of unit outputs will fail, since such an estimator would require that the number of correctly coding units is high compared to the number of units with invalid estimates.

3.4 Network-Structure

Disparity estimators in real biological networks can and will vary in various properties, notably in the separation of the center of their receptive fields in the left and right eye. Other possible parameters of interest are the spatial orientation, the scale or the phase of the Gabor filter patches.

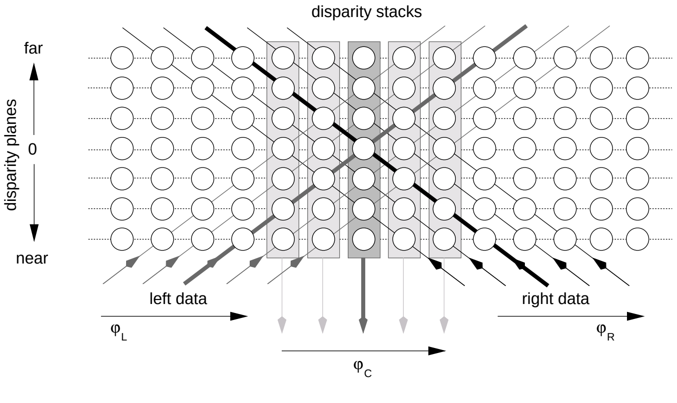

Figure 8: The main structure of the stereo vision network. Disparity estimators are arranged in horizontal sheets responding to a common depth. Units looking into common view directions are collected together, into vertically aligned disparity stacks. Within these stacks, coherence detection takes place, facilitated by appropriate weak coupling between the units in a stack.

In the simplified network used here, the orientation of the Gabor filter patches is fixed to the vertical, and three to four different spatial scales are used. At a single scale, image data is fed diagonally into layers of identical disparity units. This creates layers of units having a common and fixed separation of receptive fields in the left and right eye (Fig. 8). Disparity units stacked vertically above each other sample space in a common view direction, and it is here where the coherence detection scheme sets in: all units with a common view direction are connected with each other through weak synaptic links. This coupling creates the vertical disparity stacks marked by the rectangular outlines in Fig 8. To simplify the simulations, disparity units operating at different scales are not treated separately, but are simply included in the appropriate disparity stacks at the finest spatial resolution.

3.5 Results

In the first few layers of the stereo network standard rate-coded neurons are used for computations. Starting with the disparity stacks, network operations are realized in detail by using leaky integrate-and-fire neurons (technical details can be found in the Appendix). Real image data is supplied to the network, and pre-processing in the retina is simulated by a logarithmic nonlinearity, followed by convolving the data with a Mexican-hat filter.

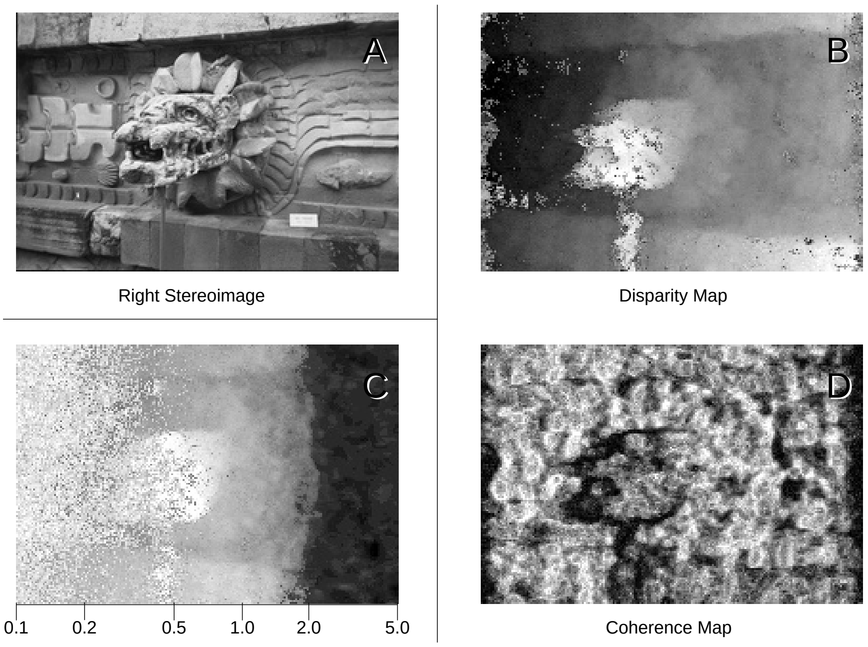

Figure 9: The performance of the coherence network with a real stereo image as input (right image of stereo pair is shown in A). Brighter image areas in the disparity map (B) are estimated as nearer to the viewer. The modulation depth of the output current of the coherence layer, here named coherence map (D), gives a hint on how secure the neural network is of its estimate. Network operation depends on the strength of the interlayer synaptic links; in the range $w_{CC} \approx 0.5 - 2.0$, a useable disparity map is calculated (C).

Fig. 9A shows one of the stereoimages used for testing and the disparity map estimated by the network (Fig. 9B). Distance is coded in the disparity map as brightness, with brighter image areas estimated as nearer, darker image areas estimated further away from the viewer. For nearly all view directions, disparity is calculated correctly out of the raw stereo pair.

During network operations, all disparity stacks operate independently of each other and in parallel. The lower traces (B, C) in Fig. 2 show examples of the dynamics of different disparity stacks during such a simulation.

The coupling within each disparity stack leads quickly to the development and synchronization of the coherence cluster, which in turn shows up as a prominent modulation of the total output current (see Fig. 3). This modulation of the output current coming from the coherence detection layer can be picked up by read-out neurons with appropriately adjusted weights. It is the firing rate of these read-out neurons which is displayed in Fig. 9B as the disparity estimate.

Of course, more elaborate read-out schemes are conceivable, like phase-locked loops which will lock exclusively on the oscillatory component of the output current coming from the coherence layer. Also, the inclusion of additional synchronizing links between the independently operating disparity stacks would further enhance network performance.

Switching off the synaptic links responsible for the coherence detection process, i. e., the links internal to the disparity stacks, results in an unusable disparity map. This can be seen in Fig. 9C, which displays an overview of network operations with changing coupling constant w_{CC} in the disparity stacks. This coupling constant regulates the coherence detection process; in Fig. 9C w_{CC} is varied along the horizontal image dimension for an overview (note the logarithmic scale).

At low values of w_{CC}, no synchronization is possible between the neurons and the readout-layer computes an average of the signals coming from the disparity stacks. Due the large amount of noise within the stacks, no reliable estimates are obtained. In the range of $w_{CC} \approx 0.5 - 2.0$, synchronization becomes possible, and the coherence detection delivers correct disparity estimates, somewhat decreasing in fidelity with increasing w_{CC}. For values larger than $w_{CC} \approx 2.0$, all the units in a single disparity stack synchronize, resulting again in unusable estimates.

The coherence detection process gives also a hint about the validity of the estimate. The modulation depth of the total output current of the coherence layer reflects the number of neural units participating in an estimate, so this number can be used as validation measure for the estimate.

In the example of Fig. 9, image areas where the disparity estimate might be in doubt can be found at the borders around the dragon head. No distance can be calculated here, due to occlusion effects: image areas visible in the right image are not visible in the left image and vice versa. Looking at the map which records the modulation depth of the output current, (Fig. 9D), one sees that the critical areas are indeed marked by low coherence measure.

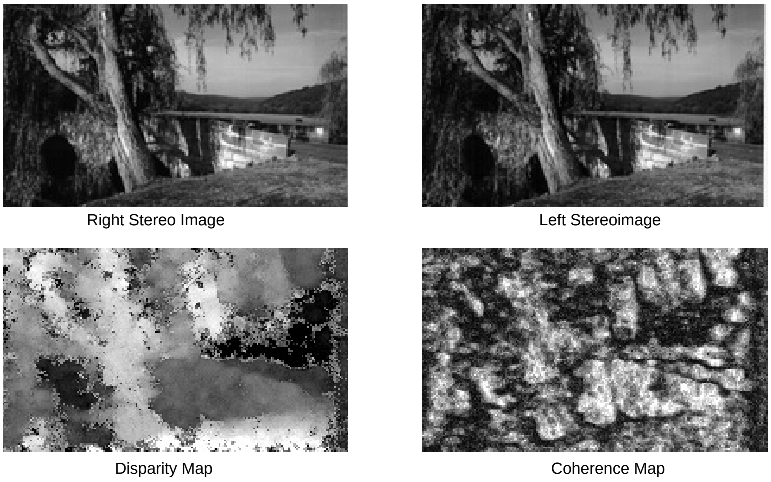

Figure 10: Another example of disparity estimation by the coherence network. The stereo pair (top images) is arranged for crossed-eye viewing. The estimates of the disparity map can be validated by the corresponding coherence map.

The reliability of the coherence map for validating the disparity estimates can also be inferred from a second example (Fig. 10), where all prominent object borders in the scene show up as dark lines of low confidence in the network coherence map. The structure-less sky area and some additional image areas where no stable disparity estimate could be obtained are also marked by a low coherence count.

Coherence detection in combination with several parallel processing streams has the interesting property of opportunistic selection of data sources. The neural coherence detection process recruits every piece of information which is usable, but discards the rest. This can lead to the effect of "filling-in" in areas of low or no texture of a stereogram.

Fig 11 displays results obtained with a sparse random-dot stereogram, where only 3% of the image pixels are set to black. The rest of the images is a constant background, so in most image areas, there is no data available to estimate depth.

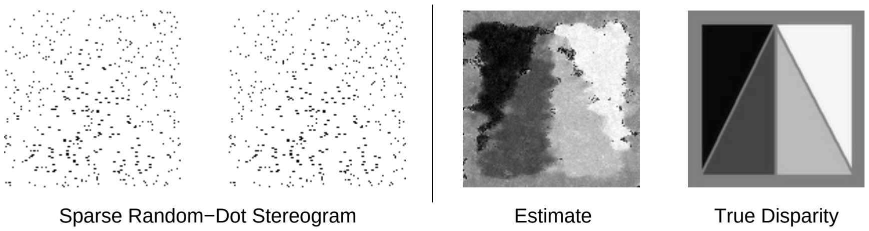

Figure 11: The coherence-based stereo network "fills-in" missing data; a sparse stereogram, where only 3% of the pixels are set, still leads to the perception of planar surfaces (from left: stereopair, estimated disparity, true disparity).

Despite this sparse information, humans perceive this stereograms as an arrangement of several flat surfaces, and this is also what happens in the stereo network. This is caused essentially by the network operating at three different spatial scales. For image areas where no data is available in the fine resolution channels, the coarser scales can still give a guess about the disparity present in these areas. The coherence-detection process locks onto this partial information, and discards the non-valid or absent estimates of the finer resolution channels. The same kind of opportunistic signal selection might also be responsable for the perception of transparency, where several depth planes are seen in a single view direction.

4 Summary

A dynamical coherence-detection process has been proposed as primary basis for stable percept creation. This process of coherence detection between neural signals can be realized with plausible neural circuitry, by synchronizing neural oscillators connected through weak synaptic links.

The synchronization process in pools of interconnected neurons calculates a robust estimate from incoming noisy signals and an additional measure of validation. The robust estimate is coded as modulation frequency of the output current of the coherence-detecting layer, and the validation measure is given by the modulation depth of this current. As a worked-out example of the concepts, a stereo vision network was presented, calculating disparity estimates out of real image data by coherence detection.

Acknowledgments

The author would like to thank Helmut Schwegler and Svenja Falk, for interesting discussions and encouraging support. Many thanks go also to Mike Mackey, whose insightful comments greatly improved a preliminary, "noisy" version of the manuscript.

A Appendix - Details of Network Operation

A.1 The Input Layers

Network operations in the first layers of the stereo vision network are simulated with simple rate-coding neurons, i.e., only the average firing frequency s_i of a neuron is used for coding signals. The operation of a generic rate-coding neuron with index i can be described by the equation

$$s_i = f\left(\sum_j w_{ij} s_j\right) , \tag{3}$$

where w_{ij} is the synaptic link from cell j to cell i and $f(\ldots)$ the transfer function of the neuron.

A.1.1 The Retina

Processing in the retina was modeled by filtering the logarithm of the input image $\ln(I_{(i,j)})$, where $I_{(i,j)}$ represents image intensity at the retinal position (i,j), with the two-dimensional Mexican-hat function

$$\begin{aligned} m_{(x,y)} &= \left[\partial_x^2 + \partial_y^2\right] \exp(-0.5 * (x^2+y^2)/\sigma^2) \\ &= \frac{1}{\sigma^4}(x^2+y^2-2*\sigma^2) \exp(-0.5*(x^2+y^2)/\sigma^2) \, . \end{aligned} \tag{4}$$

The filter operation

$$e_{(k,l)} = \sum_{i,j} m_{(i,j)} \cdot \ln(I_{(k-i,l-j)}) \tag{5}$$

is quite effective in removing relative contrast variations between the two stereo images.

In the simulations, $\sigma^2 = 2.0$ is used, and the result $e_{(k,l)}$ of this retinal filter operation is fed forward into the simple cell layer.

A.1.2 Simple Cells

The combined layer of simple and complex cells is practically identical to the model proposed in [42] for modeling responses of binocular cortical neurons. For simplicity, neurons in our simulation are allowed to code negative as well as positive signal values; in [42], a push-pull configuration of neurons was used instead.

The receptive fields of the simple cells are modeled by a generic Gabor function of the form

$$g(x;\omega,\varphi) = \exp(-0.5*(x/\sigma)^2) \cdot \cos(\omega x + \varphi) \, , \tag{6}$$

with phase-parameter φ and spatial frequency ω. The filter profiles resulting from sampling this filter function at the discrete receptor positions $x = i$ are normalized and adjusted so that no DC-component is present in the final filter coefficients.

Simple cells sample data from both left and right input streams, which we denote here by $e^L_{(k,l)}$ and $e^R_{(k,l)}$. As transfer function for simple cells, a squaring nonlinearity is used. With the normalized filter coefficients given by $\tilde{g}(k;\omega,\varphi)$, the output of a simple cell can be written as

$$s^a_{(i,j)}(\varphi,s) = \left[\sum_k \tilde{g}(k-s/2;\omega,+\varphi)\cdot e^L_{(i-k,j)} + \tilde{g}(k+s/2;\omega,-\varphi)\cdot e^R_{(i-k,j)}\right]^2 . \tag{7}$$

Here s is a small relative shift of the centers of the two receptive fields in the left and right retina. Varying s creates the layered structure displayed in Fig. 8; in the simulations, s runs from -8 to $+8$ in steps of $1/3$. The spatial frequency ω is set to $\omega = 1.5$. The phase-shift φ of the simple cells depends on the type of complex cell to which the simple cell is feeding data and can have the values $-\pi/4$, 0 and $+\pi/4$ (compare below).

For quadrature filtering, the output of any simple cell having a phase-shift φ has to be paired with another simple cell, having a phase-shift of $\varphi + \pi/2$. This means that for every simple cell $s^a_{(i,j)}(\varphi, s)$ with a receptive field defined by (7) there exists another, paired one, calculating

$$\begin{aligned} s^b_{(i,j)}(\varphi, s) &= \Bigg[\sum_k g(k - s/2; \omega, \pi/2 + \varphi) e^L_{(i-k,j)} \\ &+ \sum_k g(k + s/2; \omega, \pi/2 - \varphi) e^R_{(i-k,j)}\Bigg]^2 \end{aligned} \tag{8}$$

A.1.3 Complex Cells

Complex cells $c_{(i,j)}(\varphi, s)$ sample input from the simple cell layer by adding the responses of paired cells. In the reported simulations, data is sampled from a small patch of neighboring simple cells, so a complex cell has the response

$$c_{(i,j)}(\varphi, s) = \sum_{k,l} b_k b_l [s^a_{(i-k,j-l)}(\varphi, s) + s^b_{(i-k,j-l)}(\varphi, s)]\,, \tag{9}$$

with the filter kernel $b_k = (0.0625, 0.25, 0.375, 0.25, 0.0625)$. Depending on the value of φ, a complex cells responds to different relative shifts of images patches in left or right retinal image; we use the values $\varphi = -\pi/4$, $+\pi/4$, and 0. This corresponds to “left”, “right” and “no” image motion. Thus, for $\varphi = 0$, a complex cell measures local image contrast.

A.1.4 Disparity Estimators

The output of any complex cell depends also on the local image contrast and has to be normalized in order to achieve contrast-independent performance. The necessary contrast normalization, which might be realized in cortex by shunting inhibition [40, 46, 47], was modeled here simply by dividing the signals of complex cells responding to left and right image motion with the response of a matching complex cell measuring local contrast. Following Adelson, we construct disparity estimators out of the response of three complex cells by computing

$$d_{(i,j)}(s) = \frac{c_{(i,j)}(+\pi/4, s) - c_{(i,j)}(-\pi/4, s)}{c_{(i,j)}(0, s)}\,.$$

In effect, this gives a set of noisy local disparity estimates $d_{(i,j)}(s)$ in each view direction (i, j). The shift parameter s becomes a this point merely an index, numbering different disparity units in a single disparity stack $\mathcal{D}(i, j)$ [45].

A.2 The Coherence Network

The noisy disparity estimates from the first three network layers are combined into a stable percept by coherence networks for every view direction (i, j). The coherence network consists of two layers, a primary coherence layer $\mathcal{C}(i, j)$ doing coherence detection and a final read-out layer $\mathcal{R}(i, j)$. A single coherence layer $\mathcal{C}(i, j)$ with corresponding read-out layer $\mathcal{R}(i, j)$ realizes one of the disparity stacks $\mathcal{D}(i, j)$ of Fig. 8. All disparity stacks $\mathcal{D}(i, j)$ operate independently from each other.

A.2.1 The Model Neuron

Starting with the coherence layer, network operations are simulated in detail, with spiking integrate-and-fire neurons described by

$$C\,\frac{dV}{dt} = -V/R + I(t)\ . \tag{10}$$

Here, the parameter C models the membrane capacity of the neuron, and R is the corresponding membrane resistance. In the simulations, the membrane capacity is fixed to $C = 0.00625$, and $R = 40$ for all neurons. This gives a membrane time constant of $\tau = 0.25$.

In an integrate-and-fire neuron, the membrane potential $V(t)$ develops according to equation (10) as long as $V < \theta$, the threshold potential. If $\theta = 16.8$ is reached, an actionpotential or spike is emitted, and the membrane potential is reset to $V = 0$. After a short refractory period, $t_r = 0.2$, where $V(t)$ stays at its resting level $V_0 = 0$, the membrane potential develops again according to Equation 10. If a neuron k fired at times t_k^F, it supplies a current of exponentially decaying pulses, $i_k(t;\tau) = \sum_{t^F} \exp(-(t - t_k^F)/\tau)$, to all postsynaptic neurons.

A.2.2 The Coherence Layer

All neurons in a single coherence layer $\mathcal{C}(i,j)$ of a disparity stack $\mathcal{D}(i,j)$ are numbered by the shift parameter s (compare A.1.4). The total input current of a neuron $s \in \mathcal{C}(i,j)$ in the coherence layer is given by

$$I_s(t) = \sum_{\bar{s}\in\mathcal{C}(i,j)} w^{\mathcal{C}}_{s\bar{s}} \cdot i_{\bar{s}}(t;\tau_{\mathcal{C}}) + S(d_{(i,j)}(s))\,, \tag{11}$$

where the raw disparity estimates $d_{(i,j)}(s)$ are converted to actual input currents $S(d_{(i,j)}(s))$ by a linear function, $S(x) = 3.0 + 0.65 * (x + 10)$. The synaptic decay constant of the input currents is set to $\tau_{\mathcal{C}} = 0.01$.

The interlayer coupling between the neurons in a coherence layer is an all-to-all coupling without self-interaction, i.e., the synaptic links are given by

$$w^{\mathcal{C}}_{s\bar{s}} = \begin{cases} w_{\mathcal{C}\mathcal{C}}/N_{\mathcal{C}} & \text{if } s \neq \bar{s} \\ 0 & \text{else} \end{cases}\ . \tag{12}$$

Here $N_{\mathcal{C}}$ is the number of neurons in a single disparity stack, and varies between 20 and 100 neurons in the reported simulations.

The value of the interlayer coupling $w_{\mathcal{C}\mathcal{C}}$ is the most important parameter of the whole system; it is this coupling constant which corresponds to the coherence threshold ϵ in the abstract coherence detection scheme, defined in equation (1). To ensure that network operations realize the dynamical coherence detection scheme, $w_{\mathcal{C}\mathcal{C}}$ can vary between $w_{\mathcal{C}\mathcal{C}} \approx 0.5 - 2.0$ (compare Fig. 4E, 4F and Fig. 9D). In the simulations reported here, a value of $w_{\mathcal{C}\mathcal{C}} = 0.7$ was used.

A.2.3 The Read-Out Layer

During dynamical coherence detection, the total output current of the coherence layer is composed of a small random component plus an oscillatory component coming from the synchronous coherence cluster. The oscillatory component has to be detected by the read-out layer $\mathcal{R}(i,j)$.

In the simulations, the read-out neurons $r \in \mathcal{R}(i,j)$ are driven by the total output current of the coherence layer (the first term in equation (13)), plus a synchronizing interaction term:

$$I_r(t) = w_{\mathcal{C}\mathcal{R}}/N_{\mathcal{C}} \cdot \sum_{s\in\mathcal{C}(i,j)} i_s(t;\tau_{\mathcal{C}\mathcal{R}}) + \sum_{\bar{r}\in\mathcal{R}(i,j)} w^{\mathcal{R}}_{r\bar{r}} \cdot i_{\bar{r}}(t;\tau_{\mathcal{R}}) \tag{13}$$

The interlayer connections,

$$w_{r\bar{r}}^{\mathcal{R}} = \begin{cases} w_{\mathcal{R}\mathcal{R}}/N_{\mathcal{R}} & \text{if } r \neq \bar{r} \\ 0 & \text{else} \end{cases}, \tag{14}$$

include no self-interaction term, as in the coherence layer.

The number of read-out neurons $N_{\mathcal{R}}$ ranges between 2 and 16, and the synaptic decay time constant $\tau_{\mathcal{R}}$ is chosen to equal $\tau_{\mathcal{C}}$ in the simulations. Network performance turns out to be insensitive to the precise value of $w_{\mathcal{R}\mathcal{R}}$, and $w_{\mathcal{R}\mathcal{R}}$ was set to $w_{\mathcal{R}\mathcal{R}} = 0.85$ in all simulations.

The coupling between coherence- and read-out layer, $w_{\mathcal{C}\mathcal{R}}$, modifies the capture range of the read-out neurons; in the simulations, it was fixed to $w_{\mathcal{C}\mathcal{R}} = 4.4$, and the synaptic time constant to $\tau_{\mathcal{C}\mathcal{R}} = 0.01$. The results displayed in the maps of the main text were obtained by measuring the spike frequencies f_r of the neurons in the read-out layer $\mathcal{R}(i,j)$ and averaging: $\overline{f_{(i,j)}} = 1/N_{\mathcal{R}} \sum_{r \in \mathcal{R}(i,j)} f_r$.

References

[1] R. Eckhorn, R. Bauer, W. Jordan, M. Brosch, W. Kruse, M. Munk, H. J. Reitboeck, *Coherent Oscillations: A Mechanism of Feature Linking in the Visual Cortex?*, Biol. Cybern. 60, 1988, 121–130.

[2] C. M. Gray, W. Singer, *Stimulus-specific Neuronal Oscillations in Orientation Columns of Cat Visual Cortex*, Proc. Natl. Acad. Sci. USA 86, 1989, 1698–1702.

[3] A. Engel, P. König, W. Singer, *Direct physiological Evidence for Scene Segmentation by Temporal Coding*, Proc. Nat. Acad. Sci. USA 88, 1991, 9136–9140.

[4] P. Lennie, *Single Units and Visual Cortical Organization*, Perception 27, 1998, 889–935.

[5] A. Winfree, *The Geometry of Biological Time*, Springer-Verlag, New York, 1980.

[6] Y. Kuramoto, *Chemical Oscillations, Waves, and Turbulence*, Springer-Verlag, New York, 1984.

[7] L. Glass, M. C. Mackey, *From Clocks to Chaos – The Rhythms of Life*, Princton University Press, Princeton, 1988.

[8] J. D. Murray, *Mathematical Biology*, Springer-Verlag, Berlin, 1989.

[9] R. E. Mirollo, S. H. Strogatz, *Synchronization of Pulse-Coupled Biological Oscillators*, SIAM Journ. of Appl. Math. 50, 1990, 1645–1662.

[10] N. Kopell, G.B. Ermentrout, *Phase Transitions and other Phenomena in Chains of Coupled Oscillators*, SIAM Journ. of Appl. Math. 50, 1990, 1014–1052.

[11] H. Daido, *Lower Critical Dimension for Populations of Oscillators with Random Distributed Frequencies: A Renormalization-Group Analysis*, Phys. Rev. Lett. 61(2), 1988, 231–234.

[12] E. Niebur, H. G. Schuster, D. M. Kammen, C. Koch, *Oscillator-Phase Coupling for Different Two-Dimensional Network Connectivities*, Phys. Rev A 44(10), 1991, 6895–6904.

[13] E. D. Lumer, B. A. Huberman, *Binding Hierarchies: A Basis for Dynamical Perceptual Grouping*, Neural Computation 4, 1992, 341–355.

[14] J. L. van Hemmen, W. F. Wreszinski, *Lyapunov Function for the Kuramoto Model of Nonlinearly Coupled Oscillators*, J. Stat. Phys. 72, 1993, 145–166.

[15] M. Tsodyks, I. Mitkov, H. Sompolinski, *Pattern of Synchronity in Inhomogenous Networks of Oscillators with Pulse Interactions*, Phys. Rev. Lett. 71(8), 1993, 1280–1283.

[16] S. Bottani, *Synchronization of Integrate and Fire oscillators with Global Coupling*, Phys. Rev. E 54(3), 1996, 2334–2350.

[17] E. M. Izhikevich, *Weakly Connected Quasi-Periodic Oscillators, FM-Interactions, and Multiplexing in the Brain*, Siam J. Appl. Math. 59(6), 1999, 2193–2223.

[18] P. M. Milner, *A Model for Visual Shape Perception*, Psych. Re. 81(6), 1974, 521–535.

[19] C. von der Malsburg, W. Schneider, *A Neural Cocktail-Party Processor*, Biol. Cybern. 54, 1986, 20–40.

[20] D. V. Ramana Reddy, A. Sen, G. L. Johnson, *Time Delay Induced Death in Coupled Limit Cycle Oscillators*, Phys. Rev. Lett. 80(23), 1998, 5109–5112.

[21] H. G. Schuster, P. Wagner, *A Model for Neuronal Oscillations in the Visual Cortex*, Biol. Cybern. 64, 1990, 77–85.

[22] C. von der Malsburg, J. Buhmann, *Sensory Segmentation with Coupled Neural Oscillators*, Biol. Cybern. 67, 1992, 233-242.

[23] T. Chawanya, T. Aoyagi, I. Nishikawa, K. Okuda, Y. Kuramoto, *A Model for Feature Linking via Collective Oscillations in the Primary Visual Cortex*, Biol. Cybern. 68, 1993, 483–490.

[24] E. R. Grannan, D. Kleinfeld, H. Sompolinsky, *Stimulus-Dependent Synchronization of Neuronal Assemblies*, Neural Computation, 5(4), 1993, 550–569.

[25] T. B. Schillen, P. König, *Binding by Temporal Structure in Multiple Feature Domains of an Oscillatory Neuronal Network*, Biol. Cybern. 70, 1994, 397–405.

[26] D. Wang, D. Terman, *Locally Excitatory Globally Inhibitory Oscillator Networks*, IEEE Trans. on Neural Networks, 6(1), 1995, 283–286.

[27] S. Sinha, J. Basak, *Response of an Excitatory-Inhibitory Neural Network to External Stimulation: An Application to Image Segmentation*, preprint 1998, `http://xxx.lanl.gov/abs/cond-mat/9811403.`

[28] S. R. Campbell, D. L. Wang, C. Jayaprakash, *Synchrony and Desychrony in Integrate-and-Fire-Oscillators*, Neural Computation 11, 1999, 1595–1619.

[29] H. Neven, A. Aersten, *Rate Coherence and Event Coherence in the Visual Cortex: a Neuronal Model of Object Recognition*, Biol. Cybern. 67, 1992, 309–322.

[30] R. Ritz, W. Gerstner, U. Fuentes, J. L. van Hemmen, *A Biological Motivated and Analytically Soluble Model of Collective Oscillations in the Cortex*, Biol. Cybern. 71, 1994, 349–358.

[31] P. Baldi, R. Meir, *Computing with Arrays of Coupled Oscillators: An Application to Preattentive Texture Discrimination*, Neural Computation 2, 1990, 458–471.

[32] O. Sporns, G. Tononi, G. M. Edelman, *Modeling Perceptual Grouping and Figure-Ground Separation by Means of Active Reentrant Connections*, Proc. Natl. Acad. Sci. USA 88, 1991,129–133.

[33] H. Sompolinsky, D. Golomb, D. Kleinfeld, *Cooperative Dynamics in Visual Processing*, Phys. Rev. A 43(12), 1991, 6990–7011.

[34] T. Murata, H. Shimizu, *Oscillatory Binocular System and Temporal Segmentation of Stereoscopic Depth Surfaces*, Biol. Cybern. 68, 1993, 381–391.

[35] S. Champbell, D. Wang, *Synchronization and Desynchronization in a Network of Locally Coupled Wilson-Cowan Oscillators*, IEEE Trans. on Neural Networks 7(3), 1996, 541–552.

[36] D. Wang, D. Terman, *Image Segmentation Based on Oscillatory Correlation*, Neural Computation 9(4), 1997, 805–836.

[37] P. Meer, *Robust high breakdown estimation and consensus*, in "Advances in Machine Vision: Strategies and Applications", C. Archibald, E. Petriu (Eds.), World Scientific Publishing Company, Singapore, 1992, 23–34.

[38] J. Porrill, J. P. Frisby, W. J. Adams, D. Buckley, *Robust and Optimal Use of Information in Stereo Vision*, Nature 397, 1999, 63–66.

[39] M. Concetta Morrone, D. C. Burr, *Feature Detection in Human Vision: a Phase-Dependent Energy Model*, Proc. R. Soc. Lond. B 235, 1988, 221–245.

[40] D. J. Heeger, E. P. Simonelli, J. A. Movshon, *Computational Models of Cortical Processing*, Proc. Natl. Acad. Sci. USA 93, 1996, 623–627.

[41] E. H. Adelson, J. R. Bergen, *Spatiotemporal Energy Models for the Perception of Motion*, J. Opt. Soc. Am. A 2(2), 1985, 284–299.

[42] I. Ohzawa, G. C. DeAngelis, R. D. Freeman, *Stereoscopic Depth Discrimination in the Visual Cortex: Neurons Ideally Suited as Disparity Detectors*, Science, 1990, 1037–1041.

[43] L. A. Olzak, J. P. Thomas, *Neural Recoding in Human Pattern Vision: Models and Mechanism*, Vis. Res. 39, 1999, 231–256.

[44] N. Qian, *Computing Stereo Disparity and Motion with Known Binocular Cell Properties*, Neural Comp. 6, 1994, 390–404.

[45] R. D. Henkel, *A Simple and Fast Neural Network Approach to Stereo Vision,* Proc. of NIPS'97 in Denver, MIT Press, Cambridge 1998, 808–814.

[46] M. Carandini, D. J. Heeger, J. A. Movshon, *Linearity and Normalization in Simple Cells of Macaque Primary Visual Cortex*, J. of Neuroscience 17(21), 1997, 8621–8644.

[47] D. J. Fleet, D. J. Heeger, H. Wagner, *Modelling Binocular Neurons in the Primary Visual Cortex,* in "Computational and Biological Mechanisms of Visual Coding", M. Jenkin, K. Harris (Eds.), Cambrigde University Press, 1997, 103–130.