Biol Cybern (2008) 98:195–211
DOI 10.1007/s00422-007-0203-z

ORIGINAL PAPER

Biological Cybernetics

Cooperation in self-organizing map networks enhances information transmission in the presence of input background activity

Maxim Raginsky · Thomas J. Anastasio

Received: 23 April 2007 / Accepted: 21 November 2007 / Published online: 12 December 2007

Abstract The self-organizing map (SOM) algorithm produces artificial neural maps by simulating competition and cooperation among neurons. We study the consequences of input background activity on simulated self-organization, using the SOM, of the retinotopic map in the superior colliculus. The colliculus not only represents its inputs but also uses them to localize saccadic targets. Using the colliculus as a test-bed enables us to quantify the results of self-organization both descriptively, in terms of input–output mutual information, and functionally, in terms of the probability of error (expected distortion) in localizing targets. We find that mutual information is low, and distortion is high, when the SOM operates in the presence of input background activity but without the cooperative component (no neighbor training). Cooperation (training neighbors) greatly increases mutual information and greatly decreases expected distortion. Our simulation results extend theoretical work suggesting that cooperative mechanisms are needed to increase the information content of neural representations. They also identify input background activity as a factor affecting the self-organization of information-transmitting channels in the nervous system.

This work was funded by Beckman Institute Fellowship to MR, and by Office of Naval Research Grant N00014-01-1-0249 to TJA.

M. Raginsky (✉)
Beckman Institute for Advanced Science and Technology,
University of Illinois, 405 N Mathews Ave,
Urbana, IL 61801, USA
e-mail: maxim@uiuc.edu

Present Address:
M. Raginsky
Department of Electrical and Computer Engineering,
Duke University,
Durham, NC 27708, USA
e-mail: m.raginsky@duke.edu

T. J. Anastasio
Beckman Institute for Advanced Science and Technology,
Department of Molecular and Integrative Physiology,
University of Illinois, Urbana, IL 61801, USA
e-mail: tja@uiuc.edu

1 Introduction

Topographic sensory maps are ubiquitous in the brain. Classic examples are the retinotopic maps in the lateral geniculate nucleus and visual cortex, tonotopic maps in the cochlear nucleus and auditory cortex, and the body-surface map (homunculus) in the primary somatosensory cortex (Daniel and Whitteridge 1961; Bishop et al. 1962; Malpeli and Baker 1975; Merzenich et al. 1975; Tusa et al. 1978; Bourk et al. 1981; Woolsey 1981). While the basic layout of topographic maps is established by activity-independent mechanisms during development (Tessier-Lavigne 1995; Tessier-Lavigne and Goodman 1996; O'Leary et al. 1999), subsequent map refinement requires activity-dependent mechanisms (Schmidt 1985; Constantine-Paton et al. 1990; Cline 1991, 1998; Zhang et al. 1998). The self-organizing map (SOM) algorithm is a well-accepted model of this refinement process. By simulating competition and cooperation among neurons, the SOM trains an array of output units so that units near each other in the array respond to inputs with similar features. The competitive mechanism causes different output units to become selective for different input features. It is the cooperative mechanism, implemented over a local neighborhood of output units during SOM training, which causes the spatial ordering of the outputs that leads to map formation. The SOM produces artificial maps with properties similar to those of real neural maps (e.g., Willshaw and von der Malsburg 1976; Kohonen 1982, 1988; Obermayer et al. 1990; Obermayer et al. 1992).

From the time of the classic map studies it has been known that neurons in the sensory structures that provide input to sensory maps have spontaneous background activity (e.g., Kuffler 1953; Hubel and Wiesel 1960; Katsuki et al. 1962; Mountcastle et al. 1963; Kiang 1965; Cleland et al. 1971; Aitkin and Webster 1972; Tsumato and Nakamura 1974; Hayward 1975). The background activity depends on many factors including brain region, anesthetic level, and species (cat and monkey are the most common subjects), but its level is a sizable fraction of the driven activity of these neurons. Despite the omnipresence of spontaneous neural activity, the effects of input background activity on networks trained using the SOM have not been explored. Here, we show through computational modeling that map formation may be the result of a mechanism that is needed to increase information transmission from inputs to outputs, when the inputs have background activity.

The function of the activity-dependent mechanisms that shape sensory representations may be to increase the amount of sensory information contained by the neurons in the representation. Algorithms have been developed that explicitly maximize information transmission from inputs to outputs (see Baddeley et al. 2000 for recent examples). Some of these InfoMax algorithms make use of map formation processes (e.g., Linsker 1989; Luttrell 1989, 1994; Van Hulle 1996, 1997). Theoretical work shows that the SOM itself can increase information transmission in networks. The so-called "magnification factor" of the SOM, which relates the proportion of the output representing a given input to the input probability density, is explicitly related to the size of the neighborhood of cooperation and controls the amount of transmitted information (Ritter and Schulten 1986; Ritter 1991; Dersch and Tavan 1995; Villmann and Claussen 2006). This theory extends the results on asymptotic level density in vector quantizers (Zador 1982; Gersho and Gray 1992; Graf and Luschgy 2000) to the case of training algorithms which combine competitive and cooperative mechanisms. These theoretical results were derived using continuous input representations and lead to the conclusion that information transmission should increase as neighborhood size increases. However, already in the case of vector quantization (i.e., for purely competitive networks) the theory for discrete inputs and finitely many quantizer codepoints (weights) is qualitatively different from the asymptotic theory for continuous-valued inputs. For discrete inputs and for finite-size networks, the distribution of the quantizer codepoints is discrete, rather than continuous, and its relation to the input distribution may be much more complicated than a simple power law (Berger 1971; Berger and Gibson 1998). Here, we show that, for networks with discrete-valued inputs, SOM training over local output neighborhoods can increase the amount of transmitted information, just like in the extensively studied asymptotic continuous case, but the optimal neighborhood size (leading to maximum information transmission) is finite and small. We also show that the presence of input background activity can adversely affect information transmission in a purely competitive self-organizing network, but that cooperative mechanisms can increase information transmission despite the presence of input background activity. To give behavioral relevance to the information, we study self-organization in the context of a model of a sensorimotor structure whose function is to detect and localize the targets of orienting movements.

The vertebrate optic tectum, which is called the superior colliculus in mammals, is a midbrain structure involved in generating orienting responses, especially shifts of gaze, toward the sources of sensory stimuli in the environment (Vanegas 1984). It receives sensory input of multiple modalities and is topographically organized (Middlebrooks and Knudsen 1984; Meredith and Stein 1990; Meredith et al. 1991; Wallace et al. 1996). The colliculus receives input from numerous brain regions (Edwards et al. 1979), and neurons in these regions can have substantial spontaneous activity (e.g., Bock et al. 1971; Brownell 1975; Guinan et al. 1972; Schmidt 1996). The SOM has been used to model the self-organization and registration of multiple sensory maps in the barn owl optic tectum (Gelfand et al. 1988; Ferrell 1996), and the multisensory representation in the mammalian colliculus (Anastasio and Patton 2003). The motor output of the colliculus (tectum) is also topographically organized (Robinson 1972; Hepp et al. 1993) and in register with the sensory maps. Thus, the function of the colliculus is to use its sensory inputs to detect and localize targets, and to initiate orienting responses toward them.

The spatial target localization function of the superior colliculus allows the behavioral significance of information transmission to be assessed in terms of the probability of error in determining target location. Here, we characterize networks trained using the SOM not only in terms of the amount of information transmitted from input to output, but also according to the "meaning" of that information in terms of its behavioral relevance. We use the SOM in the context of a model of the superior colliculus to show that competitive mechanisms alone are inadequate to increase information transmission, and to decrease the probability of error in target localization, in the presence of input background activity. Combining competition with cooperation dramatically alters that situation, and it is the cooperative mechanism that causes map formation with the SOM.

In order to demonstrate as clearly as possible the usefulness of cooperative mechanisms in forming information-preserving output representations in the presence of input background activity, we use simplified, stripped-down models of the colliculus. The models have two layers (input and output), with discrete input spatial tuning functions and output neighborhoods. The output self-organizes to represent

target location only, and target localization by the model colliculus is implemented by finding the output unit with the maximal response to a given input. The SOM learning rule can be seen as a simplified version of the activity-dependent processes that shape output representations in the real brain. The SOM is essentially Hebbian, in that increases in a synaptic weight depend on the correlation between pre-synaptic and post-synaptic activities. The effects of input background activity, and the usefulness of cooperative mechanisms that we demonstrate using the SOM, would apply to more complicated algorithms that are also based on Hebbian forms of learning. Model simplification facilitates mathematical description and analysis, and presents the effects at issue in the clearest possible terms.

The present paper makes three novel contributions. First, we confirm empirically that SOM training can increase information transmission, but show that the optimal neighborhood size for discrete-input networks is small and finite, rather than very large and potentially infinite, as specified by the previous theory based on a continuous approximation. Second, we demonstrate computationally that the benefit of cooperative mechanisms that cause map formation may be to increase information transmission when the inputs have background activity. Third, we argue that the functional significance of these mechanisms cannot be understood merely in terms of maximizing mutual information. Instead, one must first state an *operational objective*, which in the case of the colliculus is meaningfully defined as the probability of correctly localizing the target. Information-theoretic quantities, such as entropy or mutual information, can then be used to assess the effects both of the structural characteristics of the model (such as the degree of cooperation in the output layer) and its statistical parameters (such as the relative values of the mean driven and background activity) on the optimal performance achievable by the system in terms of that operational objective.

2 Methods

The benefit of cooperative mechanisms on information transmission is illustrated using a series of neural network simulations. The networks have two layers of processing units, input and output. The input units encode the location of a sensory target. The amount of input background activity, relative to the activity driven by the target, varies between simulations. The output is winner-take-all, and the function of the network is to infer target location according to the identity of the winning output unit. The networks are trained using the SOM algorithm, and the size of the output neighborhood varies between simulations. To determine the effectiveness of training, we determine the amount of target information transmitted from the inputs to the outputs and use it to assess the minimum achievable probability of error in determining target location. The output responses are also used to determine the presence or absence of map formation. The results are used to draw conclusions concerning the benefit of cooperative mechanisms on information transmission in the presence of input background activity.

2.1 Notation

We adhere to the following notational conventions. All random variables are denoted by uppercase letters, e.g., X, and their specific realizations are denoted by the corresponding lowercase letters, e.g., x. Deterministic variables are often denoted by uppercase letters too, but it should be clear from the context whether a particular variable is random or deterministic. Vector-valued quantities are written in boldface, e.g., $\mathbf{X}$ for a random vector and $\mathbf{x}$ for its specific realization. Matrices are denoted by sans-serif uppercase letters, e.g., V. Probability distributions (mass functions) are denoted by lowercase letters, e.g., p, q, etc., while conditional probability distributions are denoted by uppercase letters, e.g., P, Q, etc. All logarithms are taken to the base two.

2.2 Network structure, function, and adaptation

We consider separately two models of the colliculus: a simple one-dimensional deterministic model, whose aim is to illustrate the basic principles involved with the minimum of detail, and a slightly more detailed two-dimensional stochastic model. We describe each model in turn.

Deterministic model. The input and output units are arranged in one-dimensional arrays of size M and N, respectively, where $M < N$ (see Discussion). The activity of input unit j is denoted by x_j, while the activity of output unit i is denoted by y_i. The target can appear randomly at the location of any input unit. The input unit at that location is activated, along with a set of adjacent input units, according to the input spatial tuning parameter s. When s is set to zero, only the single input unit at the location of the target is activated. When s is set to one, the input at the target location and its nearest neighbors are activated. Using a discrete, discontinuous spatial tuning function allows us to unambiguously set the activity levels of input units that are, and are not, driven by the target. (This is especially important in the stochastic case; see next section.) The boundary conditions are open, so that when the target appears at the locations of the input units at the ends of the array, only the one nearest neighbor is activated. All input units activated by the target take value one, while all input units not activated by the target take the background

value $b < 1$. For the purpose of finding nearest neighbors, the distance between the units is computed according to

$$\Delta(j, j') = |a(j) - a(j')|, \tag{1}$$

where $a(j)$ denotes the position of unit j along the input array. The distance between the units in the output layer is computed analogously.

The input units connect to the output units over connection weights v_{ij}. The response of the output unit i is computed as the weighted sum over its inputs

$$y_i = \sum_j v_{ij} x_j = \mathsf{V}_i^{\mathrm{T}} \mathbf{x}, \tag{2}$$

where $\mathbf{x}$ is the column vector with elements x_j, V is the weight matrix with nonnegative elements v_{ij}, V_i is the ith row of V, and T is the transposition operator. The output is winner-take-all, with the index of the winner given by

$$i^* = i^*(\mathbf{x}) = \arg\max_i y_i. \tag{3}$$

In case of a tie (i.e., two or more output units sharing the maximal response) during SOM training or operation, the winner is taken as the unit with the maximal response and the lowest index. Prior to training, the elements of the weight matrix V are drawn randomly from the uniform distribution on [0, 1], and its rows are normalized to unit norm. During training, the weight vector of the winning output unit i^* is adjusted, along with a set $\mathcal{N}(i^*)$ of adjacent output units, according to the neighborhood parameter $h \in \{0, 1, \ldots, 10\}$

$$\mathcal{N}(i^*) = \{\text{all } i : \Delta(i^*, i) \le h\}. \tag{4}$$

In other words, when h is set to zero, only the weight vector of the winning output unit i^* is modified. For the sake of consistency we shall slightly abuse terminology and refer to the $h = 0$ case as a (purely competitive) SOM, even though it is, strictly speaking, merely an online version of the well-known k-means algorithm (MacQueen 1967; Haykin 1999). When h is set to one, the weight vectors of the winning output unit and its two nearest neighbors (one on each side) are modified, and so on for larger neighborhood sizes. Weights are updated according to the SOM update rule of Kohonen (1982)

$$\mathsf{V}_i(\tau + 1) = \frac{\mathsf{V}_i(\tau) + \alpha \mathbf{x}^{\mathrm{T}}(\tau + 1)}{\|\mathsf{V}_i(\tau) + \alpha \mathbf{x}^{\mathrm{T}}(\tau + 1)\|}, \tag{5}$$

where $\|\cdot\|$ is the vector norm, α is the learning rate, and $\tau = 1, 2, \ldots,$ indexes the training cycles. Although output unit activity is not represented explicitly in Eq. (5), this weight update is essentially Hebbian, where the winner and its neighbors take value one, and all other output units take value zero.

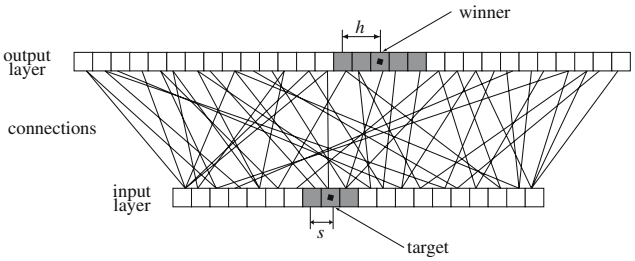

Fig. 1 The one-dimensional SOM network. Input units (*bottom*) are connected to output units (*top*) via synaptic weights. Locations of the target and the winning output unit are marked with a *black square*. The *shaded units* in the input layer are driven by the target; the remaining input units fire spontaneously. The *shaded units* in the output layer are contained in the neighborhood of the winner. Also shown are the spatial tuning s (here equal to 1) and the neighborhood size h (here equal to 2)

Typically, when using the SOM to create artificial maps, the neighborhood size and the learning rate are both decremented as SOM training proceeds (Haykin 1999). Because we are interested specifically in cooperative (neighborhood-based) mechanisms, we fix the neighborhood size h at various preset values throughout SOM training (see Discussion). For simplicity in the deterministic case, we fix the learning rate α at 1 throughout training. The architecture of the model is displayed in Fig. 1.

Stochastic model. In the stochastic model, the input and output layers of the network are both two-dimensional. The input units are located on an $M \times M$ regular grid, while the output units are located on an $N \times N$ regular grid, where $M < N$. The input units are modeled as though they receive information about the target at each grid location from multiple independent sources (e.g., from sensory sources of multiple modalities, see Introduction). Thus, the activity x_j of input unit j has k components $u_i^{(j)}$, $1 \le i \le k$, and is given by

$$x_j = \sum_{i=1}^{k} u_i^{(j)} + \beta, \tag{6}$$

where $u_1^{(j)}, \ldots, u_k^{(j)}$ are the k variable, multiple components of the jth input unit, while $\beta > 0$ is a constant bias which is independent of j. We set $\beta = 1$.

The target can appear randomly at the location of any input unit. Thus, the number of possible target states is equal to M^2, the number of the input units. The activation pattern induced by the target is governed by the spatial tuning parameter s. When the target appears at the location of input unit j, it drives that unit along with all units j' satisfying

$$\Delta(j, j') = \max_{m=1,2} |a(j, m) - a(j', m)| \le s, \tag{7}$$

where $a(j,1)$ and $a(j,2)$ are the coordinates of unit j in the input array. The components $\{U_i^{(j)} : 1 \le j \le M^2, 1 \le i \le k\}$ are Bernoulli random variables which are conditionally independent given the (random) target location. If unit j is activated by the target, then $U_1^{(j)}, \ldots, U_k^{(j)}$ each take value 1 with probability p_1 and value 0 with probability $1 - p_1$ independently of one another. If unit j is not activated by the target, then $U_1^{(j)}, \ldots, U_k^{(j)}$ each take value 1 with probability p_0 and value 0 with probability $1 - p_0$ independently of one another. Thus, the activity X_j of the jth input unit is a random variable taking values in the set $\{1, 2, \ldots, k+1\}$. Any unit activated by the target exhibits *driven activity* distributed according to

$$p_1(x) = \binom{k}{x-1} p_1^{x-1}(1-p_1)^{k-x+1}, \tag{8}$$

while the *spontaneous activity* of a unit not activated by the target is distributed according to

$$p_0(x) = \binom{k}{x-1} p_0^{x-1}(1-p_0)^{k-x+1}, \tag{9}$$

where $\binom{m}{n} \equiv m!/(n!(m-n)!)$ is the binomial coefficient. Since the average driven activity should be higher than the average spontaneous activity, we require that $p_1 > p_0$. Thus, both the driven and the spontaneous activities are described by binomial random variables, but with an additional shift by 1 to ensure that all input units are active at all times. The binomial distribution is a simple and convenient discrete distribution to use for modeling integration of independent inputs from multiple sources, and has been used previously in models of collicular self-organization (Anastasio and Patton 2003).

Just as in the deterministic case, the vector $\mathbf{x}$ of the input activities determines the output activities y_i via the weight matrix V. Prior to traning, the entries v_{ij} of the weight matrix are drawn from the uniform distribution on [0, 1], and each row of V is normalized to unit norm. The training takes place in a manner similar to the deterministic case, except that the learning rate decays exponentially starting from some initial value α_0, i.e., $\alpha(u+1) = \alpha_0 r^u$ for some $0 < r < 1$. The weights of the winning unit $i^* = \arg\max_i y_i$ and those of its neighbors $\mathcal{N}(i^*)$ undergo the Kohonen update (5). Following training, the network operates exactly as in the deterministic case.

2.3 Evaluating map formation and information transmission

Following training, the networks are qualitatively evaluated to determine whether or not a map is formed. To evaluate map formation, the target is positioned at each input location in turn, and the input is activated according to the spatial tuning parameter s. In the deterministic case, the activated input units take value 1 and the other units take value b. In the stochastic case the activated units take discrete values at random according to the driven distribution in Eq. (8), while the other units take values according to the spontaneous distribution in Eq. (9). Then the output response is computed. In both the deterministic and stochastic cases, the winning output unit is identified as the output unit with the maximal response. In case of a tie, the winner is taken as the output unit with the maximal response and the lowest index. Finally, each position in the array of target locations is marked by the index of the output unit that produced the winning response to the input activated by the target at that location. This is the reverse of the usual map display format, in which each unit in the array of output units is marked by the input pattern to which it produces its best response. The target location representation is more convenient here because there are fewer target locations than output units, and the output is winner-take-all, so that some output units may never win for any of the target locations (see Discussion). The maps that form are often fractured. In a fractured map, a smooth spatial progression in the input representation is interrupted by a sharp break, after which there appears another smooth progression. Lack of map formation is identified by the lack of any spatial ordering of the output units by target location.

The trained networks are also evaluated to determine the amount of information about the target they are capable of transmitting from input to output. This evaluation is based on the information-theoretic measures of entropy and mutual information (Cover and Thomas 1991). The state (location) of the target is a random variable T. The information content of the target is given by the entropy of T

$$H(T) = -\sum_t p(t) \log p(t), \tag{10}$$

where the summation is over all possible states (locations) t of the target, $p(t)$ is the probability that the target T is in state t, and $\log(\cdot)$ is the logarithm to the base two. Given the state of the target $T = t$, the input to the network is determined according to the conditional probability distribution $P(\mathbf{x}|t)$. This covers both the deterministic and the stochastic cases. In the former case $P(\mathbf{x}|t)$ takes value 1 for a particular input pattern $\mathbf{x}(t)$ and takes value 0 for all other $\mathbf{x}$. In the latter case, each input pattern $\mathbf{x}$ occurs for a given target state $T = t$ with probability

$$P(\mathbf{x}|t) = \prod_{j=1}^{L} p_{A(j,t)}(x_j), \tag{11}$$

where $L = M^2$ is the total number of input units, and $A(j,t)$ is either 1 or 0, depending on whether input unit j is activated

by the target in state t. Note that, in both the deterministic and the stochastic cases, the conditional distribution of $\mathbf{X}$ given $T = t$ factorizes into a product over the components of $\mathbf{X}$. However, because the Kohonen algorithm is an *unsupervised* learning procedure and so does not have access to the target state that has actually produced each training example, the relevant distribution, as far as SOM training is concerned, is the *marginal* distribution of the network input $\mathbf{X}$. It is obtained by averaging out the state of the target, and has the mixture form

$$p(\mathbf{x}) = \sum_{t=1}^{T} p(t) P(\mathbf{x}|t), \tag{12}$$

which no longer factorizes into a product over individual input units. That is, the target-averaged activities of different input units are, in general, statistically dependent. This dependence is what makes analysis of information transmission in the SOM especially difficult, especially in the stochastic case. Information transmission in SOM networks with continuous inputs drawn from nonproduct probability distributions was recently analyzed computationally in the framework of magnification factors (Merényi et al. 2007). Here, however, the inputs are discrete and are themselves noisy representations of a discrete random variable of interest (namely, the target state), where the noise is due to the background activity.

We are interested in the amount of information that the index of the winning output unit, which we denote by W, contains about target state T. This is given by the mutual information, one of whose equivalent definitions useful for the purpose of computing information transmission in neural networks is

$$I(T; W) = \sum_{t} p(t) \sum_{i} Q(i|t) \log \frac{Q(i|t)}{q(i)}, \tag{13}$$

where the sum is over all possible states t of the target and over all possible indices i of the winner. $Q(i|t)$ is the conditional probability that unit i is the winner given $T = t$

$$Q(i|t) = \sum_{\mathbf{x}} P'(i|\mathbf{x}) P(\mathbf{x}|t), \tag{14}$$

and $q(i)$ is the unconditional probability of output unit i being the winner:

$$q(i) = \sum_{t} Q(i|t) p(t). \tag{15}$$

Note that, because the winner selection is deterministic, the conditional probabilities $P'(i|\mathbf{x})$ can only take values 0 or 1, depending on whether output unit i would win the competition when the input activation pattern is $\mathbf{x}$, that is,

$$P'(i|\mathbf{x}) = \begin{cases} 1, & \text{if } i = i^*(\mathbf{x}) \\ 0, & \text{if } i \neq i^*(\mathbf{x}) \end{cases}. \tag{16}$$

Therefore, we can write Eq. (14) in the following equivalent form:

$$Q(i|t) = \sum_{\mathbf{x}: i = i^*(\mathbf{x})} P(\mathbf{x}|t). \tag{17}$$

The mutual information $I(T; W)$ is bounded from above by $\min\{H(T), H(W)\}$, where $H(T)$ is the entropy of the target and

$$H(W) = -\sum_{i} q(i) \log q(i) \tag{18}$$

is the entropy (information content) of the location of the winning output unit. Because we use binary logarithms, the units of entropy and mutual information are bits. The procedures for computing input–output information transmission (input–output mutual information) are necessarily different for deterministic and stochastic inputs.

Deterministic model. Because the output is winner-take-all, the number of possible output states is equal to the number of output units. When the input is deterministic, the number of input states (activity vectors) equals the number of possible target locations. These restrictions on the numbers of target (input) and output states make it practical to explicitly determine the conditional probabilities $Q(i|t)$ in the deterministic case. The conditional probability distribution $Q(i|t)$ is determined for a deterministic network by finding the output unit that produces the winning response to each target (input) state. Thus, $Q(i|t)$ is one when output unit i is the winner for target state t, and zero otherwise. In the case of a tie, the winner is taken as the output unit with the maximal response and the lowest index.

The random target can appear uniformly at any allowed location. Thus, the target probability $p(t)$ is simply $1/K$, where K is the number of target states, which is equal to M in the one-dimensional case and to M^2 in the two-dimensional case. Multiplication of $Q(i|t)$ by $p(t)$ for each t converts the conditional probability distribution $Q(i|t)$ to the joint distribution $p(i, t)$. The output probability distribution $q(i)$ is found by marginalizing the joint distribution over t, as in Eq. (15). The distributions $p(t)$, $q(i)$, and $Q(i|t)$ can then be used to compute $I(T; W)$ using Eq. (13). Note that, for deterministic input and output units, the mutual information $I(T; W)$ is equal to the output entropy $H(W)$. The target entropy $H(T)$ is simply equal to $\log K$, where K is the number of target (input) states. The amount of target information

transmitted by the network can then be assessed by comparing the information content of the target $H(T) = \log K$ with the information $I(T; W) = H(W)$ that the winner's index contains about the target.

Stochastic model. In the stochastic case, there are k^{M^2} possible input patterns for each target state $T = t$, hence for large values of M it is impractical to explicitly determine the conditional probability distribution of the winning index W given the input $\mathbf{X}$. Instead, we have to resort to Monte Carlo simulation techniques. For each target state t, we generate a large number n of independent samples $\mathbf{X}^{(l)}$, $l = 1, \ldots, n$, according to the conditional distribution $P(\mathbf{x}|t)$ and estimate the conditional probability $Q(i|t)$ by

$$\tilde{Q}(i|t) = \frac{1}{n} \sum_{l=1}^{n} P'(i|\mathbf{X}^{(l)}), \tag{19}$$

where the $\{0, 1\}$-valued conditional probabilities $P'(i|\mathbf{X}^{(l)})$ are computed according to Eq. (16). We then use the estimated conditional probabilities $\tilde{Q}(i|t)$ to compute mutual information using Eq. (13).

2.4 Rate-distortion theory

We are interested in assessing the effect of training neighbors on the sensorimotor performance of the network as measured by the probability of correctly localizing the target T following an observation of the index W of the winning output neuron. Towards this end, we draw on rate-distortion theory (Berger 1971) which relates information-theoretic quantities like entropies and mutual informations to *operational* characteristics of data-processing systems, such as probabilities of error in data transmission over a noisy communication channel. Below, we give a brief synopsis of the fundamental notions of rate-distortion theory.

The basic object there is the rate-distortion function of an information source. Let Z be a random variable taking values in some finite set $\mathscr{Z}$ according to a probability distribution $p(z)$. Define the *distortion measure* by

$$d(z, z') = \begin{cases} 0, & z = z' \\ 1, & z \neq z' \end{cases}. \tag{20}$$

Let $\hat{Z}$ be another random variable taking values in $\mathscr{Z}$ and related to Z via a conditional probability distribution $P(\hat{z}|z)$. Then the expected value of $d(Z, \hat{Z})$ is precisely the probability that $Z \neq \hat{Z}$

$$\mathrm{E}_{Z,\hat{Z}}[d(Z, \hat{Z})] = \sum_{z} \sum_{\hat{z}} p(z) P(\hat{z}|z) d(z, \hat{z})$$

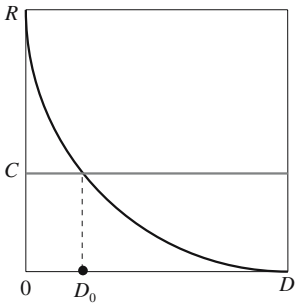

Fig. 2 A typical rate-distortion curve. The *horizontal line* corresponds to the capacity of the channel whose output is used for estimation of the random variable of interest. The distortion level D_0 at which $R(D_0) = C$ is the minimum probability of estimation error achievable by any possible processing of the channel output

$$= \sum_{z} \sum_{\hat{z} \neq z} p(z) P(\hat{z}|z)$$
$$\equiv \Pr(Z \neq \hat{Z}), \tag{21}$$

where $\mathrm{E}_{Z,\hat{Z}}[\cdot]$ denotes the expectation operator with respect to the joint distribution of Z and $\hat{Z}$. The rate-distortion function $R(D)$ of Z with respect to the distortion measure of Eq. (20) is defined as the *minimum* amount of mutual information between Z and $\hat{Z}$ required to reproduce Z by $\hat{Z}$ with the probability of error at most D:

$$R(D) = \min_{P(\hat{z}|z)} \left\{ I(Z; \hat{Z}) : \mathrm{E}_{Z,\hat{Z}}[d(Z, \hat{Z})] \leq D \right\}, \tag{22}$$

where the minimum is over all conditional probability distributions $P(\hat{z}|z)$ satisfying $\mathrm{E}_{Z,\hat{Z}}[d(Z, \hat{Z})] \leq D$. Given a distortion level D, $R(D)$ is the smallest amount of mutual information between Z and $\hat{Z}$ required for $\hat{Z}$ to represent Z correctly with probability at least $1 - D$. The rate-distortion function is convex and strictly decreasing for all $D \in (0, D_{\max})$, where

$$D_{\max} = \min_{\hat{z}} \sum_{z} p(z) d(z, \hat{z}) = \min_{\hat{z}} \sum_{z \neq \hat{z}} p(z), \tag{23}$$

and $R(D) = 0$ for all $D \geq D_{\max}$. A typical rate-distortion curve is shown in Fig. 2.

Now suppose that we do not have direct access to Z, but instead observe another random variable U which is related to Z via a conditional probability distribution $Q(u|z)$. We can picture this situation in terms of a noisy *communication channel*, which accepts z as input and emits an output u with probability $Q(u|z)$. We must then process U to obtain an estimate $\hat{Z}$ of Z. According to rate-distortion theory, the quality of our estimate is limited by the maximum amount of information that can be transmitted over the channel Q regardless of the statistics of the input Z. This is given by the *channel capacity*

$$C(Q) = \max_{p(z)} I(Z; U), \tag{24}$$

where the maximum is over all distributions of the input random variable. Then, according to Shannon's Converse

Information Transmission Theorem (Berger 1971), no amount of processing can recover Z from U with the probability of error less than D_0 determined by $R(D_0) = C$. Since the rate-distortion function is convex and strictly decreasing in the range $(0, D_{\max})$, there can be at most one such point D_0. These ideas are illustrated in Fig. 2.

We now apply the foregoing theory to the models in this paper. The rate-distortion function of a uniformly distributed source, such as the target state T with probabilities $p(t) = 1/K$ for all $t = 1, \ldots, K$ is given by

$$R(D) = \begin{cases} \log K - h(D) - D\log(K-1), \\ \quad \text{if } 0 \le D \le 1 - 1/K \\ 0, \quad \text{if } D > 1 - 1/K \end{cases}, \tag{25}$$

where $h(D) = -D\log D - (1-D)\log(1-D)$ is the binary entropy function (Berger 1971). The target state is then passed through two successive stages of processing. First, the vector of input activities $\mathbf{X}$ is generated according to the conditional probability model $P(\mathbf{x}|t)$, and then the winning output unit W is determined via the conditional probability model $P'(i|\mathbf{x})$. The overall stochastic mapping Q from the target state to the winning neuron's index is given by the composition of the two channels, as in Eq. (14). Under this setup, the winner's location W and the target state T are conditionally independent given the input activity vector $\mathbf{X}$, i.e.,

$$\begin{aligned} &\Pr(W = i, T = t|\mathbf{X} = \mathbf{x}) \\ &\quad = \Pr(W = i|\mathbf{X} = \mathbf{x}) \Pr(T = t|\mathbf{X} = \mathbf{x}). \end{aligned} \tag{26}$$

In probabilistic terminology, the random variables $T, \mathbf{X}, W$ form a *Markov chain* in that order. Hence by the Data Processing Inequality (Cover and Thomas 1991), the amount of target information contained in W cannot be larger than the amount of target information contained in $\mathbf{X}$: $I(T;W) \le I(T;\mathbf{X})$. In other words, some target information is inevitably lost by the network. The mutual information $I(T;W)$ is a lower bound on the capacity $C(Q)$ of the channel Q.

Now suppose that the location of the winning output unit W is used to estimate the location of the target T. Any such estimation procedure can be represented as a conditional probability model $Q'(\hat{t}|i)$, where $\hat{T}$ is an estimate of the target location. This includes all deterministic decision rules formed by dividing the output array of the network into K disjoint regions and estimating the target state as $\hat{T} = \hat{t}$ whenever the winning unit is a member of decision region $\hat{t}$, as well as all randomized decision rules that make "soft" decisions. The main attractive feature of rate-distortion theory is that it allows us to determine the best performance achievable by *any* decision rule without precise knowledge of the rule. Namely, we determine the capacity $C(Q)$ of the channel that takes T to W and then compute the distortion level D_0 satisfying $R(D_0) = C(Q)$, where $R(D)$ is given by Eq. (25). The channel capacity and the target-state distribution achieving it can be determined approximately by means of the well-known Blahut–Arimoto algorithm (Arimoto 1972; Blahut 1972).

Because the target, and hence the input, uniquely determines the output in the deterministic case, the channel Q is implemented by a deterministic function f, which associates to each input z a definite output $u \equiv f(z)$. The capacity $C(Q)$ is equal to $\log \|f\|$, where $\|f\|$ is the number of the elements in the range of f. A capacity-achieving distribution at the output can be constructed by simply taking each u in the range of f, finding some z such that $f(z) = u$, and then assigning to that z the probability $1/\|f\|$. The remaining elements of $\mathscr{Z}$ are assigned zero probability. In general, the choice of a capacity-achieving distribution is not unique, but the capacity $C(Q)$ and a capacity-achieving distribution can be found essentially by inspection in the deterministic case. For these reasons we calculate $C(Q)$ in the stochastic case only.

3 Results

3.1 Deterministic input

To demonstrate the benefits of cooperative neighbor training on information transmission in SOM networks in the presence of input background activity, a simple case is considered first in which the input and output units are deterministic, and both input and output unit arrays are one-dimensional. This network has $M = 20$ input units. The target can appear at the location of any input unit. Thus there are $K = 20$ possible target states and 20 possible input states. The one-dimensional network also has $N = 30$ output units. Input and neighborhood parameters are varied between simulations. Deterministic input driven activity is fixed at 1. The input background rate b varies between 0 and 0.9. The input spatial tuning parameter s is set at 0 or 1, and the size of the output training neighborhood h varies between 0 and 10. The network is trained for 1,000 iterations using the SOM as described above (see Methods). The presence of an input map is determined qualitatively following training. An example map, with $b = 0.5$, $s = 1$, and $h = 1$, is shown in Table 1. This map covers the target array, rather than the output array as is typically done in map modeling (see Methods). The map indicates the index of the output unit that produces the winning response to the input activated by each of the 20 allowed target locations.

The example map is fractured. Target locations 1 through 10 are represented by output units 17 through 1. Then an interruption occurs, after which target locations 11 through 20 are represented by output units 19 through 30. The representations sometimes skip units but are relatively smooth. The target location representation also allows easy assessment

Table 1 Winning output for each target state with input background $b = 0.5$, tuning $s = 1$, and neighborhood size $h = 1$

Target	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Output	17	15	13	11	9	8	7	5	3	1	19	20	21	23	25	26	27	28	30	30

of the number of states used by the output for encoding the input. In the example map above, output unit 30 is the winner for two different (and adjacent) target locations. Thus the output uses 19 out of a possible 30 states to represent the 20 input states.

Similar results using deterministic inputs (not shown) verify, as expected, that no map forms when the output neighborhood size is zero. Even with a nonzero neighborhood size, no map forms if input spatial tuning is zero. With zero spatial tuning none of the input states (input vectors) overlap one another. With zero overlap, all input vectors are equally distant from one another in the 20-dimensional vector space. No output neighborhood can form if there is no input neighborhood to represent.

Regardless of whether a map forms or not, training with neighbors increases information transmission in the SOM networks. The amount of information the index W of the winning output unit contains about the target T depends in part on the number of distinct winners the output uses in representing the target. SOM networks trained with neighbors sometimes had a different winner for each input state, and transmitted complete target information, while SOM networks trained without neighbors sometimes had as few as one winner for all input states, and transmitted zero target information. In deterministic networks with deterministic inputs, the channel capacity is given simply by the logarithm of the number of winning output units (see Methods, the section on rate-distortion theory), and is not reported.

The amount of information the index W of the winning output unit contains about the target T can be assessed by comparing the information content (entropy) of the target $H(T)$ with information transmission from the target to the output (mutual information) $I(T; W)$. With 20 allowed target locations, each deterministically producing a distinct input state, the entropy of the input $H(X)$ is equal to the entropy of the target $H(T)$, which is $\log 20 \approx 4.32$ bits in all cases. Because the output of the SOM networks is deterministic, the mutual information $I(T; W)$ is equal to output entropy $H(W)$ (see Methods). Thus, the ability of the SOM networks to extract target information from their inputs can be assessed by comparing the entropy of the target $H(T)$ with output entropy $H(W)$.

By interpreting our SOM networks as models of the superior colliculus we can also assess the functional consequences of information transmission. Because the function of the colliculus, in part, is to localize targets, the functional consequences of information transmission can be measured in terms of the minimum achievable probability of error in determining target location from the identity of the winning output unit. That probability is the expected distortion D obtained by solving the equation $R(D) = I(T; W)$ (see Methods). Distortion D endows the target information $I(T; W)$ contained by the output units with functional significance, or "meaning". Because achievable distortion depends on the mutual information, the two measures do not provide independent assessments of information transmission. The distortion measure is included because it indicates the functional significance of the information in the context of the sensorimotor performance of the colliculus in localizing the target.

Cooperation in the SOM involves training neighbors, and the ability of neighbor training to increase information gain and decrease distortion depends on neighborhood size. In the one-dimensional case, the neighborhood size is simply the number of units on either side of the winner that are trained by the SOM on each update cycle (see Methods). The benefit of neighbor training is shown in Fig. 3 for neighborhood sizes from 0 to 10. The target drives the input unit at its corresponding location, and the input spatial tuning parameter s specifies the number of input units on either side of the target that are also driven. One hundred SOM networks are trained for each neighborhood size. Deterministic input driven activity is fixed at 1, and input background activity b is fixed at 0.5. Input spatial tuning is 0 or 1.

The average mutual information $I(T; W)$ (Fig. 3a) and distortion D (Fig. 3b) are computed over all 100 SOM networks trained for each neighborhood size. For input spatial tuning of 1 (solid lines), mutual information is nearly maximal (i.e., nearly equal to target entropy $H(T)$), and distortion is nearly minimal (i.e., 0), for the neighborhood size of 1. Mutual information is lower, and distortion higher, for all neighborhood sizes when input spatial tuning is 0 (dashed lines) rather than 1. However, maximal mutual information and minimal distortion occur for neighborhood size 1 regardless of whether spatial tuning is 1 or 0. Thus, it appears that neighborhood size 1 is optimal for the one-dimensional deterministic input with input spatial tuning of 0 or 1 (see Discussion). Note that no map can form for neighborhood size 0, either for spatial tuning of 0 or 1, or for spatial tuning 0 for any neighborhood size.

To explore the benefit of neighbor training on information transmission in the presence of input background activity, 100 SOM networks are trained using a neighborhood size of 1 or 0, with input spatial tuning of 1 or 0, at a series of input

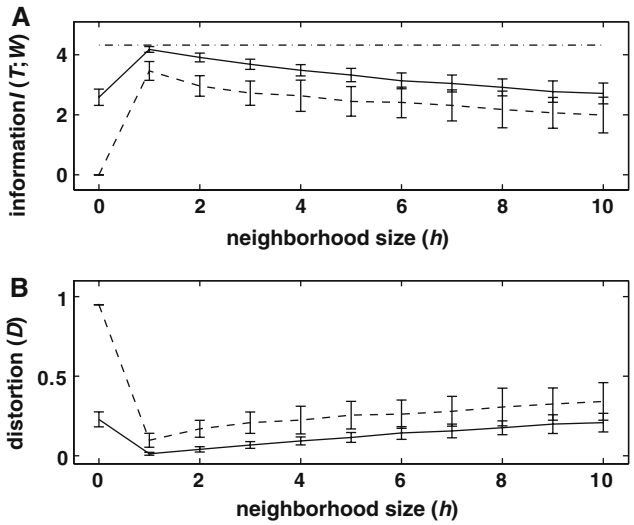

Fig. 3 The optimal neighborhood size is 1 for the one-dimensional, deterministic inputs. The input driven activity is fixed at 1 and the background activity b at 0.5. Input spatial tuning s is either 1 (*solid lines*) or 0 (*dashed lines*). Neighborhood size h is varied from 0 to 10. Average mutual information $I(T; W)$ (**a**) and distortion D (**b**) are computed over 100 SOM networks trained at each neighborhood size. For spatial tuning 1, mutual information is nearly maximal (i.e., nearly equal to target entropy $H(T)$, *dot-dash* line), and distortion is nearly minimal (i.e., 0), for neighborhood size 1. Mutual information is lower and distortion higher at all neighborhood sizes for spatial tuning 0, but again the optimal neighborhood size is 1. The plots are sample averages over 100 independent simulations; the error bars show one sample standard deviation

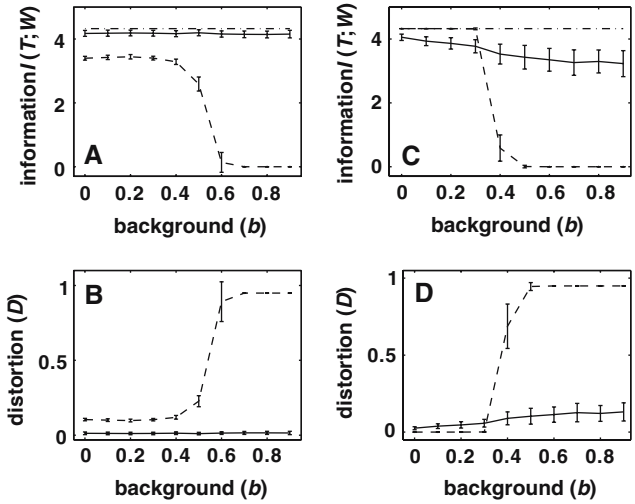

Fig. 4 Neighbor training increases mutual information $I(T; W)$ and decreases distortion D over a broad range of input background activity levels. Input spatial tuning s is either 1 (**a**, **b**) or 0 (**c**, **d**). Neighborhood size h is either 1 (*solid lines*) or 0 (*dashed lines*). With input spatial tuning 1 and neighborhood size 1, mutual information (**a**) is nearly maximal (i.e., nearly equal to $H(T)$, *dot-dashed lines*), and distortion (**b**) is nearly minimal (i.e., 0), over the entire range of input background activities. With neighborhood size 0, mutual information (**a**) is slightly lower, and distortion (**b**) slightly higher for backgrounds less than 0.4, but mutual information rapidly decreases, and distortion rapidly increases, for backgrounds greater than 0.4. With input spatial tuning 0 and neighborhood size 0, mutual information (**c**) is nearly maximal and distortion (**d**) is nearly minimal for backgrounds less than 0.3, but mutual information rapidly decreases, and distortion rapidly increases, for backgrounds greater than 0.3. With neighborhood size 1, mutual information starts out high and decreases gradually, and distortion starts out low and increases gradually, as backgrounds increase over the range. The plots are sample averages over 100 independent simulations. The error bars show one sample standard deviation. Note that the error bars for maximal and minimal mutual information or distortion are essentially zero and cannot be discerned on the plots

background activities b ranging from 0 to 0.9 (Fig. 4). For input spatial tuning of 1 and neighborhood size of 1 (solid lines), mutual information is nearly maximal (i.e., nearly equal to target entropy $H(T)$, dot-dashed line in Fig. 4a), and distortion is nearly minimal (i.e., 0, Fig. 4b), over the entire range of input background activities. Mutual information is slightly less, and distortion slightly greater, for neighborhood size of 0 (dashed lines), but only for input background activities of 0.4 or lower. As input background increases above 0.4, mutual information decreases rapidly to 0 and distortion increases rapidly to 1. Thus, with spatial tuning 1, neighbor training increases information transmission and decreases distortion at all input background activity levels, especially above 0.4, or 40% of the driven activity.

The benefits of neighbor training are essentially the same for input spatial tuning 0. Paradoxically, mutual information is maximal (Fig. 4c) and distortion is minimal (Fig. 4d) for networks trained with neighborhood size 0 (dashed lines), but only for inputs with background activities less than 0.3. As input background increases above 0.3, mutual information decreases rapidly to 0 and distortion increases rapidly to 1 for networks trained without neighbors. The paradoxical benefits of neighborhood size 0 for spatial tuning 0 in the deterministic case for low background rates ($b < 0.3$) do not carry over to the stochastic case at any background rate (see next section). In the deterministic case, for input spatial tuning 0 and neighborhood size 1 (solid lines), mutual information is nearly maximal, and distortion nearly minimal, for background activity of 0, and the deleterious effects of increasing the background are relatively mild. For neighborhood size 1, mutual information falls slightly (Fig. 4c), and distortion rises slightly (Fig. 4d), as input background activity increases from 0 to 90% of driven. Note that no map forms with input spatial tuning 0, regardless of whether or not neighbors are trained. Thus, cooperative neighbor training increases information transmission in SOM networks, and decreases distortion, even when no map forms.

3.2 Stochastic input

We extend the SOM model by making the input stochastic, and by making the input and output arrays two-dimensional. The extended network has an $M \times M$ input array and an

$N \times N$ output array with $M = 10$ and $N = 20$. Thus, the number of possible target states is $K = M^2 = 100$ with the target entropy $H(T) = \log 100 \approx 6.64$ bits. Each input neuron receives target information from $k = 5$ independent binary components. The probability models for the driven and the spontaneous activity are given by Eqs. (8) and (9), respectively. The driven probability p_1 is fixed at 0.9, while the spontaneous probability p_0 is varied from 0 to 0.8 in increments of 0.1. The driven mean is $kp_1 + 1 = 5.5$ and the variance is $kp_1(1 - p_1) = 0.45$. The spontaneous mean and variance cover the ranges from 0 to 5 and from 0 to 0.8, respectively. The input spatial tuning s can take values 0 and 1.

For each value of the spontaneous activation probability p_0 and for spatial tuning $s = 1$ and $s = 0$ we performed a series of simulations over a range of neighborhood sizes $h = 0$ through $h = 10$, 10 simulations for each value of h. In each simulation the network is trained for 1000 iterations with the starting learning rate $\alpha_0 = 1$ decaying exponentially to 0.1 at the end of training. After training, we estimate the 400×100 conditional probability matrix $Q(i|t)$ for each neighborhood size h via the Monte Carlo method. For each target state t in turn, we generate $n = 300$ independent samples $\mathbf{X}^{(l)}$, $l = 1, \ldots, n$, according to the conditional probability distribution $P(\mathbf{x}|t)$ in Eq. (11) and estimate $Q(i|t)$ for each $i = 1, \ldots, 400$ by

$$\tilde{Q}(i|t) = \frac{1}{300} \sum_{l=1}^{300} P'(i|\mathbf{X}^{(l)}). \tag{27}$$

Because the output units are deterministic, the conditional probabilities $P'(i|\mathbf{X}^{(l)})$ take values 0 or 1 according to Eq. (16).

Following the estimation of the conditional probabilities $Q(i|t)$, the networks are evaluated qualitatively for map formation. As in the one-dimensional case, map formation in the two-dimensional case is apparent when contiguous target locations are represented by relatively smooth (but possibly discontinuous) progressions of output neurons. As expected, maps in the two dimensional case do not form when neighborhood size or spatial tuning are zero (not shown). The more critical issues involve channel capacity, information transmission, and average distortion.

We use the estimated conditional probability matrices $\tilde{Q}(i|t)$ for each value of h and for each value of p_0 to determine the channel capacity $C(Q)$ by means of the Blahut–Arimoto algorithm (Arimoto 1972; Blahut 1972), as well as the actual rate of information transmission $I(T; W)$ for uniformly distributed target states using Eq. (13). The results for $p_0 = 0.5$ are plotted in Fig. 5 and show that, in the absence of cooperative interaction between the output units ($h = 0$), the index of the winner contains a negligible amount of target information. Both the channel capacity (solid line) and the actual information rate (dashed line) are increased when cooperative mechanisms are introduced. The plot indicates that there is an optimal neighborhood size (here $h = 1$) for which both the channel capacity and the information rate are the largest; specifically, $C(Q) = 3.42$ bits and $I(T; W) = 3.23$ bits for $s = 1$ (Fig. 5a) and $C(Q) = 0.83$ bits and $I(T; W) = 0.77$ bits for $s = 0$ (Fig. 5b). As the training neighborhood size is increased past $h = 1$, both $C(Q)$ and $I(T; W)$ begin to drop, but remain well above their $h = 0$ values. The curves for other values of p_0 (not shown) exhibit similar behavior: both the channel capacity $C(Q)$ and the mutual information $I(T; W)$ are the lowest when there is no cooperation among the output units ($h = 0$) and the highest when $h = 1$. When h increases past the value of 1, both $C(Q)$ and $I(T; W)$ decrease, but always stay above their $h = 0$ values. Note that in all cases the difference between the channel capacity $C(Q)$ and the actual information rate $I(T; W)$ is small, especially when $h = 1$. This indicates that training the SOM with the Kohonen algorithm enables it to adapt to statistical variations in the input. The same conclusion is reached by the theoretical work on magnification factors of SOM networks operating on continuous inputs (Ritter and Schulten 1986; Ritter 1991; Dersch and Tavan 1995; Villmann and Claussen 2006).

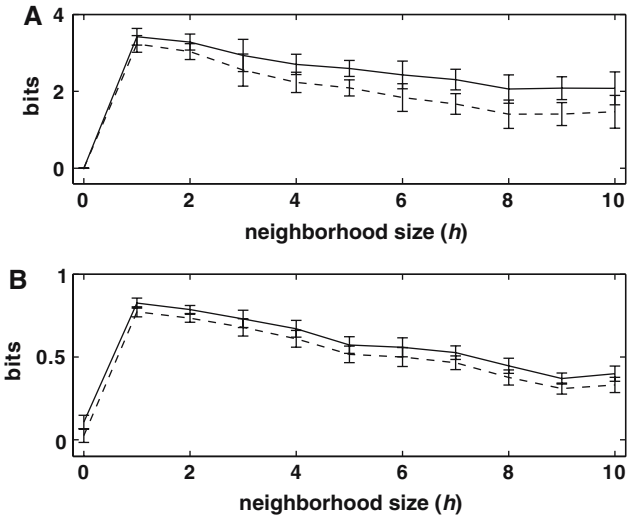

Fig. 5 The optimal neighborhood size is 1 for the two-dimensional, stochastic inputs. The input driven rate p_1 is fixed at 0.9 and the background rate p_0 at 0.5. Input spatial tuning s is either 1 (**a**) or 0 (**b**). Neighborhood size h is varied from 0 to 10. Channel capacity $C(Q)$ (*solid lines*) and mutual information $I(T; W)$ (*dashed lines*) are computed over 10 SOM networks trained at each neighborhood size. For spatial tuning 1 **a**, channel capacity and mutual information are maximal for neighborhood size 1. Channel capacity and mutual information are lower at all neighborhood sizes for spatial tuning 0 **b**, but again the optimal neighborhood size is 1. The plots are sample averages over ten independent simulations; the error bars show one sample standard deviation

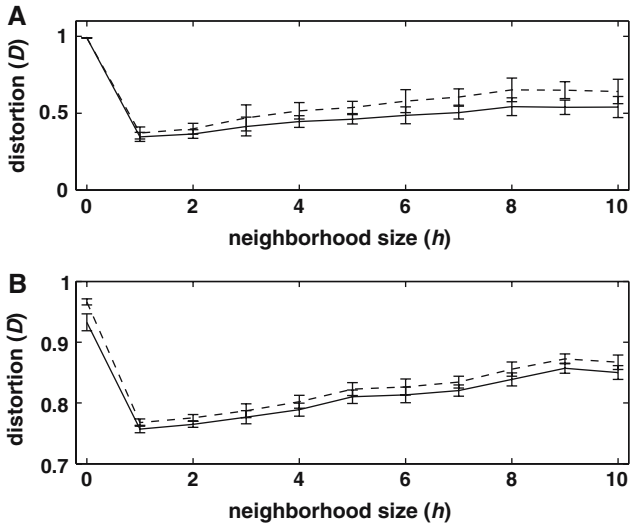

Fig. 6 The optimal neighborhood size is 1 for the two-dimensional, stochastic inputs. The input driven rate p_1 is fixed at 0.9 and the background rate p_0 at 0.5. Input spatial tuning s is either 1 (**a**) or 0 (**b**). Neighborhood size h is varied from 0 to 10. Minimum expected distortion attainable by the capacity-achieving distribution of T (*solid lines*) and by the uniform distribution used in our simulations (*dashed lines*) are computed over ten SOM networks trained at each neighborhood size. For spatial tuning 1 (**a**), distortion is minimal for neighborhood size 1. Distortion is higher at all neighborhood sizes for spatial tuning 0 (**b**), but again the optimal neighborhood size is 1. The plots are sample averages over ten independent simulations; the error bars show one sample standard deviation

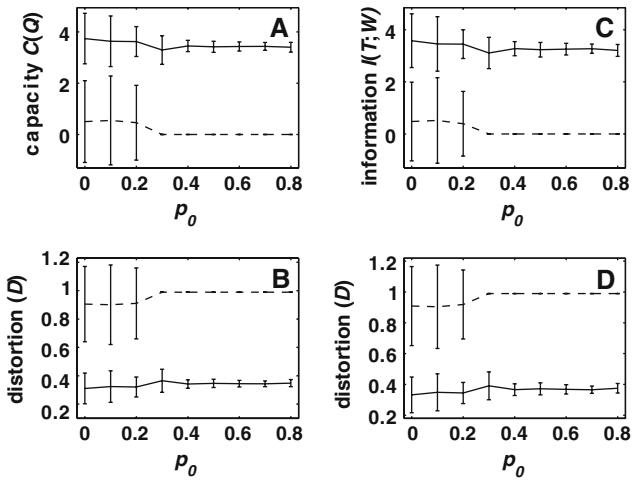

Fig. 7 Neighbor training increases channel capacity $C(Q)$ and mutual information $I(T; W)$, and decreases minimum achievable distortion over a broad range of input background activity levels. Input spatial tuning s is 1 in this figure. Neighborhood size h is either 1 (*solid lines*) or 0 (*dashed lines*). With input spatial tuning 1, the channel capacity (**a**) is much higher, and distortion (**b**) is much lower, for neighborhood size 1 than for neighborhood size 0 over the entire range of input background activities. Similarly, the actual mutual information (**c**) is much higher, and the actual distortion (**d**) is much lower, for neighborhood size 1 than for neighborhood size 0. Neighbor training increases information transmission and decreases distortion at all levels of input background activity for spatial tuning $s = 1$. The plots are sample averages over ten independent simulations; the error bars show one sample standard deviation. Note that for high values of p_0 (i.e., when the background activity is high) the error bars on the *dashed lines* are very small and hence not visible in the plot because the rate of information transmission is essentially zero in all simulations

In order to relate information-theoretic characteristics of our model to its sensorimotor performance in localizing the target, we determine the minimum achievable average distortion (probability of target localization error). To this end, we use the rate-distortion function $R(D)$ for the uniform distribution of target states. For each background activation probability p_0 and for each h, let $C(h, p_0)$ and $I(h, p_0)$ be the channel capacity $C(Q)$ and the information rate $I(T; W)$. Define $D_0(h, p_0)$ and $D_1(h, p_0)$ as the (unique) solutions of $R(D) = C(h, p_0)$ and $R(D) = I(h, p_0)$. Thus, for given values of p_0 and h, $D_0(h, p_0)$ is the fundamental lower limit on the probability of error in target detection achievable by any decision rule using W as input, which in principle can be achieved only by coding arbitrarily long temporal sequences of independent realizations of uniformly distributed target state T into long sequences of random variables that drive the channel almost at capacity (see Discussion), while $D_1(h, p_0)$ is the best target detection performance achievable when the target state is distributed uniformly (as in our actual models). These numbers are plotted in Fig. 6 ($D_0(h, p_0)$, solid lines; $D_1(h, p_1)$, dashed lines) for all values of h, with the background activation probability $p_0 = 0.5$ and spatial tuning $s = 1$ (Fig. 6a) or $s = 0$ (Fig. 6b). Note that when there is no cooperation between output units, the network transfers only a negligible fraction of target information to the output, which in turn implies that the output of such a network is poorly suited for the inference of the location of the target. The target localization performance increases dramatically when the competitive mechanisms are complemented by cooperation among nearest neighbors.

To explore the benefit of neighbor training on information transmission in the presence of stochastic input background activity, ten SOM networks are trained using a neighborhood size of 1 or 0, with input spatial tuning of 1 or 0, at a series of input background rates p_0 ranging from 0 to 0.8. The input driven rate p_1 is fixed at 0.9. For input spatial tuning of 1 (Fig. 7), channel capacity $C(Q)$ (Fig. 7a) is much larger for neighborhood size 1 (solid lines) than for neighborhood size 0 (dashed lines), and the minimum expected distortion attainable by coding the uniformly distributed target state T into channel input with the capacity-achieving distribution (Fig. 7b) is much lower for $h = 1$ than for $h = 0$ at all levels of the background rate p_0. The same holds for mutual information $I(T; W)$ (Fig. 7c) and the minimum expected distortion attainable with the uniformly distributed target state T (Fig. 7d). Thus, with spatial tuning 1, neighbor training increases information transmission and decreases distortion at all input background activity levels. The benefits of

neighbor training are essentially the same for input spatial tuning 0 (Fig. 8), although in absolute terms the performance of the network is much poorer compared to the $s = 1$ case.

Our results also show that the actual mutual information $I(T; W)$ between the uniformly distributed target state T and the winner-take-all output W of the SOM remains close to the information capacity $C(Q)$ of the overall channel $Q(i|t)$ for all neighborhood sizes h and for all values of the background activation probability p_0. In fact, the difference $C(Q) - I(T; W)$ is the smallest for $h = 1$, for all values of p_0. This reinforces the notion that the SOM training is an adaptive procedure which learns to transfer the maximum amount of input information under given constraints on h, the degree of cooperation in its output layer (Ritter and Schulten 1986; Ritter 1991; Dersch and Tavan 1995; Villmann and Claussen 2006), but with the caveat that, for discrete-valued inputs, increasing the neighborhood size past a certain optimal value will lead to a decrease of the rate of information transmission.

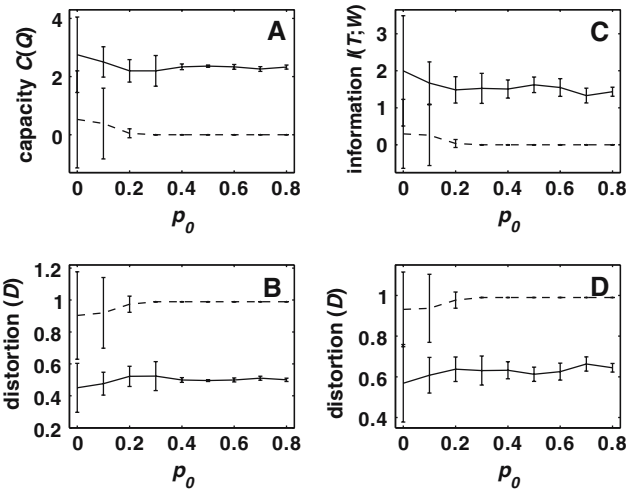

Fig. 8 Neighbor training increases channel capacity $C(Q)$ and mutual information $I(T; W)$, and decreases minimum achievable distortion over a broad range of input background activity levels. Input spatial tuning s is 0 in this figure. Neighborhood size h is either 1 (*solid lines*) or 0 (*dashed lines*). With no input spatial tuning, the channel capacity (**a**) is much higher, and distortion (**b**) is much lower, for neighborhood size 1 than for neighborhood size 0 over the entire range of input background activities. Similarly, the actual mutual information (**c**) is much higher, and the actual distortion (**d**) is much lower, for neighborhood size 1 than for neighborhood size 0. Neighbor training increases information transmission and decreases distortion at all levels of input background activity for spatial tuning $s = 0$. The plots are sample averages over ten independent simulations; the error bars show one sample standard deviation. Note that for high values of p_0 (i.e., when the background activity is high) the error bars on the *dashed lines* are very small and hence not visible in the plot because the rate of information transmission is essentially zero in all simulations

4 Discussion

The results using both deterministic and stochastic inputs, over a broad range of background activities, show that the outputs of SOM networks trained with neighbors contain more target information than networks trained without neighbors, regardless of input spatial tuning and the ultimate formation of a map. This phenomenon can be explained in terms of the structure of the input vectors.

With the background rate close to the driven rate, all input vectors are "flat", in the sense that all of the elements of the input vectors take nearly the same value, regardless of the location of the target. An output unit with a flat initial weight vector (one that has its initially random input weight values nearly equal) will respond well to the input, regardless of target location. The output unit with the flattest initial weight vector, compared with the other output units, will win the competition for the first input, no matter the target location. SOM training without neighbors will cause that output to flatten its weight vector even further (i.e., make the elements of its weight vector even more nearly equal), and this will make the winner even more likely to respond best to any input, regardless of target location. With high input background rate, SOM training without neighbors can result in one (or a few) output units winning the competition for all of the inputs. Information transfer is low because the output uses only one (or only a few) of its potential winner-take-all states to represent the target.

SOM training with neighbors changes this situation. Now the neighbors of the winner flatten their weight vectors as well. Since the initially random weight vectors are unlikely to be the same, training the neighbors makes them more sensitive to inputs with high background rates, and possibly also more sensitive than the winner to inputs with different target locations. The result is that the neighbors will likely be more sensitive than the previous winner to an input encoding a different target location, and the network ends up using more of its potential winner-take-all states to encode target location. Cooperatively training neighbors ensures high output information content even when input background activity is as high as 90% of driven. The benefit of neighbor training occurs regardless of whether or not the inputs have a nonzero spatial tuning function. If the input spatial tuning function is nonzero, then a map is formed, but information transmission is improved by cooperatively training neighbors in either case.

Interestingly, our simulations show that increasing the neighborhood size does not improve information transmission indefinitely. This was somewhat unexpected considering previous theoretical work that suggests that information transmission in networks trained using the SOM should increase as the size of the neighborhood increases (Ritter and Schulten 1986; Ritter 1991; Dersch and Tavan 1995; Villmann and Claussen 2006). Instead, we observe a peak at

a certain neighborhood size, and then the information rate beings to level off. The difference may be due to the fact that the theoretical results were derived using asymptotic continuous approximations while our actual networks are discrete and have finite size. Moreover, whereas previous work treated the network input as a primary object and studied the extent to which the SOM output preserved the input, in our models the network input is an intermediate noisy representation of the (random) state of an external target, where the noise takes the form of input background activity. Therefore, we are interested not so much in the ability of the SOM network to accurately represent its noisy input, but rather in its ability to extract information relevant to the target state from the input. We can adduce the following explanation for the apparently optimal neighborhood size of one in our discrete networks. Given the target location, not every input unit carries information needed for localizing the target. In fact, in order to localize the target, the network needs only to isolate the island of driven activity in the input layer. This can be accomplished by finding the *boundary* between the neurons that are driven and those that fire spontaneously. Thus, apart from the neurons driven by the target, the rest of the input layer has spontaneous activity which is essentially noise. Increasing the neighborhood size essentially forces the network to focus on these noisy, spontaneously firing input units which are more numerous than the driven ones; hence, the information needed to localize the driven-spontaneous boundary is "drowned out" by noise. Note, however, that even past the optimal neighborhood size the rate of information transmission is still substantially higher than that achievable without any neighbor training, because the network trained without neighbors cannot effectively distinguish a driven input unit from a spontaneously firing input unit. This reasoning also applies to smooth neighborhood functions such as a Gaussian, which is more commonly used with SOM's (Haykin 1999).

Our main finding is that input background activity has deleterious effects on information transmission in networks trained using the SOM, and that cooperative mechanisms can reduce these effects. However, information transmission decreases as input background activity increases whether or not neighbors are trained. The deleterious effect of input background activity is most striking in the deterministic case with neighborhood size zero (Fig. 4, dashed lines), but is also clearly a factor for neighborhood size one and input spatial tuning zero (Fig. 4c, solid line). Small but consistent effects are noted in the stochastic case for neighborhoods of one and zero, and input spatial tuning of one and zero (Figs. 7 and 8). Input background activity may be a complicating factor for activity-dependent mechanisms of self-organization in general.

The precise nature of the activity-dependent mechanisms that shape sensory maps in the real brain is not fully known, but they are generally believed to involve a Hebbian increase in synaptic weights to a neuron, based on pre-synaptic and post-synaptic correlation, followed by some sort of weight normalization to prevent weights from growing without bound (Dayan and Abbott 2001). Any such mechanism would be as susceptible to the deleterious effects of input background activity as the SOM. Our simulations based on the SOM illustrate what may be a general problem in the formation of sensory representations in the presence of input background activity. They also indicate a possible solution, in terms of the cooperative mechanisms that can lead to map formation but also increase the information content of self-organizing sensory representations.

Inputs have nonzero spatial tuning functions in the real nervous system, and they also have nonzero background rates. Despite the nonzero input background, a cooperative mechanism operating on these inputs would increase the amount of information contained by the outputs. Because of the nonzero spatial tuning function, the cooperative mechanism would also produce a map of the feature space encoded by the inputs. Our results suggest that the map is incidental to the real function of activity-dependent, cooperative learning mechanisms, which may be to maximize the transmission of target information from the network input to the output in the presence of input background activity.

In training SOM networks, neighborhood size typically starts out large, often encompassing the entire network, and is reduced to zero as training proceeds (Haykin 1999). Here our goal was to compare SOM networks trained with and without neighbors. Training without neighbors is equivalent to fixing the neighborhood size at zero throughout training. To ensure a fair comparison, we compared networks trained without neighbors to networks trained with neighborhoods fixed at various sizes throughout training. In both the deterministic and the stochastic cases, training at neighborhood size one produced the best information transfer and the lowest probability of error in localizing the target. The performance of networks trained with neighborhood size fixed at one throughout training was comparable to that of networks trained with a discrete neighborhood that starts out encompassing the entire network and is decreased to zero during training, or to the classical Gaussian neighborhood with variance that starts out large and is decreased during training (Haykin 1999). The fixed neighborhood size of one provided competition enough to cause specialization of the output units, and cooperation enough to draw most, if not all, of the output units into the representation.

Not all output units were winners. An output unit that was not a winner had the same preferred input state as a nearby winner. Non-winning output units occurred in SOM networks trained using all of the fixed neighborhood sizes, and also occurred in SOM networks trained with a neighborhood that starts out encompassing the entire network and is

decreased during training. Using the winner-take-all decoding scheme, a network would need one winner for each input state to achieve complete information transfer. Because the SOM produced some non-winning output units, we needed to have more output units than input states to allow for the possibility that the SOM networks could achieve complete information transfer.

Even though some output units were non-winners, most of the output units were responsive. Unresponsive output units were rare (one or just a few), and tended to occur in networks trained with smaller fixed neighborhood sizes. The possibility that unresponsive, or "dead", units could arise in SOM (and competitive learning) networks has been noted previously (Grossberg 1976; Rumelhart and Zipser 1985; Ritter et al. 1992). Removal of dead neurons is part of the developmental process in real neural systems (Contestabile 2000).

Fractured (rather than continuous) maps occurred in SOM networks trained with input spatial tuning one and all of the fixed neighborhood sizes, and also occurred with a neighborhood that starts out encompassing the entire network and is decreased during training. Fracturing occurred in our SOMs because of the small amount of overlap of our input patterns; with input spatial tuning of one, only the input unit at the location of the target and its nearest neighbors are activated. While some real brain maps are continuous, many others are fractured (Woolsey 1981). The collicular map is continuous, but its continuity may have more to do with activity-independent than with activity-dependent mechanisms (see Introduction). Our results on information transfer in SOM networks are independent of the particular layout of the maps that may result from training. Indeed, the fractured, discontinuous nature of our maps underscores our main point: that the purpose of cooperative mechanisms is not to form maps, but to increase the information content of output representations that self-organize from inputs that are spontaneously active. We demonstrate the benefits of neighbor training even for the case of zero overlap between input states, in which no map can form (see Results).

We use rate-distortion theory to measure the probability of correctly localizing the target by observing the identity of the winning unit in the model colliculus, thus endowing the mutual information between the target and the winner with "meaning". It is important to point out that in general the optimum average distortion D_0 (see Methods, the section on rate-distortion theory) is achieved by coding arbitrarily long temporal sequences (blocks) of independent realizations of T into longer blocks of channel input symbols, such that repeated transmissions of the encoded input symbols over the channel effectively drive the channel nearly at capacity (Cover and Thomas 1991). It is highly unlikely that such complex block encodings are implemented in the actual nervous system. Instead, it is more reasonable to suppose the following. Let $R_1 = I(T; W)$ be the mutual information between the target location T and the index of the winner W, and let D_1 be the unique solution of $R(D_1) = R_1$. Because $I(T; W) \leq C(Q)$, we have $D_1 \geq D_0$. Now, suppose that there exists a conditional probability distribution $Q'(\hat{t}|i)$, such that

$$R_1 - \varepsilon \leq I(T; \hat{T}) \leq R_1 \tag{28}$$

and

$$D_1 - \varepsilon \leq \sum_{t,\hat{t}} \sum_{i} d(t, \hat{t}\,) Q'(\hat{t}|i) Q(i|t) p(t) \leq D_1 \tag{29}$$

for some small $\varepsilon > 0$. Then, given a single input $T = t$, the output of the channel $Q(i|t)$ can be stochastically decoded in such a way that the end-to-end average performance of the system lies close to the (D_1, R_1) point on the rate-distortion curve. In information-theoretic terms, this is an example of *source-channel matching* for symbol-by-symbol communication (Gastpar et al. 2003). It has been recently suggested that this kind of source-channel matching could have evolved in biological neural systems, enabling the brain to accurately make crucial decisions based on short-duration stochastic inputs (Berger 2003). Note also that if R_1 is close to $C(Q)$ (which is the case when the actual distribution $p(t)$ of T is close to the capacity-achieving distribution), then D_1 is close to D_0 as well, owing to the fact that the rate-distortion function $R(D)$, being convex and strictly decreasing for all $D \in (0, D_{\max})$, has a continuous inverse. If there exists a stochastic decoding Q' satisfying the source-channel matching conditions of Eqs. (28) and (29), then the performance close to the fundamental optimum point $(D_0, C(Q))$ on the rate-distortion curve can be achieved in a symbol-by-symbol manner.

We also stress that information-theoretic quantities, such as entropy or mutual information, by themselves shed no light on the *functional* significance of a given neuronal structure. In order to address the issues of function and behavioral relevance, one has to state an *operational objective*, and then relate this objective to information-theoretic quantities in order to determine the optimal performance achievable by neural networks with given structure and parameters. We have followed this philosophy in our model of the superior colliculus, in which the operational objective is the probability of correctly localizing the randomly located external target. We modeled collicular neurons and their inputs as the output units and the inputs of an SOM network. This model comprised an encoding rule that translates the discrete location of the target into the activation pattern of the input layer of the SOM, followed by the winner-take-all mapping. Training neighbors has a discernible effect on the optimal end-to-end performance of the system, when the output of the SOM is decoded to produce an estimate of the target location. Although, we did not specify a decoding rule, it is reasonable to assume that the actual decoding used in the brain is well-matched to the typical inputs received by the colliculus, and

thus can perform close to the rate-distortion optimum. In this regard, it is important to point out that rate-distortion theory allows us to calculate the *minimum* amount of information needed to achieve a given level of performance. This is of considerable importance to the brain because retaining more information than necessary for achieving a given goal may be metabolically costly. Thus, what matters is not the absolute quantity of target information transmitted to the colliculus, but the extent to which it exceeds the rate-distortion minimum for a given value of the error probability. We plan to address these issues in future work.

It is also worthwhile to ask whether using the *entire* output of the network, rather than just the winner, will result in improved performance. This may potentially be the case, since the entire network output contains more target information than just the winner. In this paper, however, we focus only on winner-take-all because our primary motivation is to elucidate the target selection function of the superior colliculus which, to a first approximation, can be modeled as a winner-take-all (Keller and McPeek 2002; McPeek and Keller 2002). Another reason why we do not consider a more general scenario is computational: the number of possible output states is enormous even for a deterministic one-dimensional network of moderate size, which makes computation of the mutual information infeasible. Collapsing the network output to winner-take-all substantially reduces the number of effective output states, which makes it practical to compute information measures. We leave the question of the benefit of arbitrary SOM outputs for future investigation.

Acknowledgments The authors would like to thank the anonymous reviewers for their constructive criticism, which resulted in several major improvements to the presentation.

References

Aitkin LM, Webster WR (1972) Medial geniculate body of the cat: organization and responses to tonal stimuli of neurons in ventral division. J Neurophysiol 35:365–380

Anastasio TJ, Patton PE (2003) A two-stage unsupervised learning algorithm reproduces multisensory enhancement in a neural network of the corticotectal system. J Neurosci 23:6713–6727

Arimoto S (1972) An algorithm for calculating the capacity of an arbitrary discrete memoryless channel. IEEE Trans Info Theory IT-18:14–20

Baddeley R, Hancock P, Földiák P (eds) (2000) Information theory and the brain. Cambridge University Press, Cambridge

Berger T (1971) Rate distortion theory: a mathematical basis for data compression. Prentice Hall, Englewood Cliffs

Berger T (2003) Living information theory. In: IEEE Info Theory Soc Newsletter, March 2003: http://www.itsoc.org/publications/newsletters.html

Berger T, Gibson JA (1998) Lossy source coding. IEEE Trans Info Theory 44:2693–2723

Bishop PO, Kozak W, Levick WR, Vakkur GJ (1962) The determination of the projection of the visual field on to the lateral geniculate nucleus in the cat. J Physiol Lond 163:503–539

Blahut RE (1972) Computation of channel capacity and rate-distortion functions. IEEE Trans Info Theory IT-18:460–473

Bock GR, Webster WR, Aitkin LM (1971) Discharge patterns of single units in inferior colliculus of the alert cat. J Neurophysiol 35:265–277

Bourk TR, Mielcarz JP, Norris BE (1981) Tonotopic organization of the anteroventral cochlear nucleus of the cat. Hear Res 4:215–241

Brownell WE (1975) Organization of the cat trapezoid body and the discharge characteristics of its fibers. Brain Res 93:413–433

Cleland BG, Dubin MW, Levick WR (1971) Sustained and transient neurons in the cats retina and lateral geniculate nucleus. J Physiol Lond 217:473–496

Cline HT (1991) Activity-dependent plasticity in the visual systems of frogs and fish. Trends Neurosci 14:104–111

Cline HT (1998) Topographic maps: developing roles for synaptic plasticity. Curr Biol 8:R836–839

Constantine-Paton M, Cline HT, Debski E (1990) Patterned activity, synaptic convergence, and the NMDA receptor in developing visual pathways. Annu Rev Neurosci 13:129–154

Contestabile A (2000) Roles of NMDA receptor activity and nitric oxide production in brain development. Brain Res Revs 32:476–509

Cover TM, Thomas JA (1991) Elements of information theory. Wiley, New York

Daniel PM, Whitteridge D (1961) The representation of the visual field on the cerebral cortex in monkeys. J Physiol Lond 159:203–221

Dayan P, Abbott CF (2001) Theoretical neuroscience: computational and mathematical modeling of neural systems. MIT Press, Cambrdige

Dersch DR, Tavan P (1995) Asymptotic level density in topological feature maps. IEEE Trans Neural Netw 6:230–236

Edwards SB, Ginsburgh CL, Henkel CK, Stein BE (1979) Sources of subcortical projections to the superior colliculus in the cat. J Comp Neurol 184:309–329

Ferrell C (1996) Orientation behavior using registered topographic maps. In: Conference on the Simulation of Adaptive Behavior. Cape Cod, MA: http://www.ai.mit.edu/projects/cog/publications.html

Gastpar M, Rimoldi B, Vetterli M (2003) To code, or not to code: lossy source-channel communication revisited. IEEE Trans. Info Theory 49:1147–1158

Gelfand JJ, Pearson JC, Spence CD, Sullivain WE (1988) Multisensor integration in biological systems. In: IEEE international symposium on intelligent control. IEEE Computer Society Press, Arlington, pp 147–153

Gersho A, Gray RM (1992) Vector quantization and signal compression. Kluwer, Boston

Graf S, Luschgy H (2000) Foundations of quantization for probability distributions. Springer-Verlag, Berlin

Guinan JJ, Guinan SS, Norris BE (1972) Single auditory units in the superior olivary complex. I. Responses to sounds and classification based on physiological properties. Int J Neurosci 4:101–120

Grossberg S (1976) Adaptive pattern classification and universal recoding. I. Parallel development and coding of neural feature detectors. Biol Cybern 23:121–134

Haykin S (1999) Neural networks: a comprehensive foundation, 2nd edn. Prentice Hall, Upper Saddle River

Hayward JN (1975) Response of ventrobasal thalamic cells to hair displacement on the face of the waking monkey. J Physiol Lond 250:385–407

Hepp K, Van Opstal AJ, Straumann D, Hess BMJ, Henn V (1993) Monkey superior colliculus represents rapid eye movements in a two-dimensional motor map. J Neurophysiol 69:965–979

Hubel DA, Wiesel TN (1960) Receptive fields of optic nerve fibers in the spider monkey. J Physiol Lond 154:572–580
Katsuki Y, Suga N, Kanno Y (1962) Neural mechanism of the peripheral and central auditory systems in monkeys. J Acoust Soc Am 34:1396–1410
Keller EL, McPeek RM (2002) Neural discharge in the superior colliculus during target search paradigms. Ann NY Acad Sci 956:130–142
Kiang NYS (1965) Stimulus coding in auditory nerve and cochlear nucleus. Acta OtoLaryngol 59:186–200
Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biol Cybern 44:59–69
Kohonen T (1988) Self organization and associative memory, 2nd edn. Springer, Berlin
Kuffler SW (1953) Discharge patterns and functional organization of mammalian retina. J Neurophysiol 16:37–68
Linsker R (1989) How to generate ordered maps by maximizing the mutual information between input and output signals. Neural Comput 1:402–411
Luttrell SP (1989) Self-organization: a derivation from first principles of a class of learning algorithms. IEEE conference on neural networks, Washington DC, pp 495–498
Luttrell SP (1994) A Bayesian analysis of self-organizing maps. Neural Comput 6:767–794
MacQueen J (1967) Some methods for classification and analysis of multivariate observations. In: Proceedings of 5th Berkeley symposium on mathematical statistics and probability, vol 1, pp 281–296
Malpeli JG, Baker FH (1975) The representation of the visual field in the lateral geniculate nucleus of *Macaca mulatta*. J Comp Neurol 161:569–594
McPeek RM, Keller EL (2002) Superior colliculus activity related to concurrent processing of saccade goals in a visual search task. J Neurophysiol 87(4):1805–1815
Meredith MA, Stein BE (1990) The visuotopic component of the multisensory map in the deep laminae of the cat superior colliculus. J Neurosci 10:3727–3742
Meredith MA, Clemo HR, Stein BE (1991) Somatotopic component of the multisensory map in the deep laminae of the cat superior colliculus. J Comp Neurol 312:353–370
Merényi E, Jain A, Villmann T (2007) Explicit magnification control of self-organizing maps for "forbidden" data. IEEE Trans Neural Netw 18:786–797
Merzenich MM, Knight PL, Roth GL (1975) Representation of cochlea within primary auditory cortex in the cat. J Neurophysiol 38:231–249
Middlebrooks JC, Knudsen EI (1984) A neural code for auditory space in the cat's superior colliculus. J Neurosci 4:2621–2634
Mountcastle VB, Pogio GF, Werner G (1963) The relation of thalamic cell response to peripheral stimuli carried over an intensive continuum. J Neurophysiol 26:807–834
Obermayer K, Ritter H, Schulten K (1990) A principle for the formation of the spatial structure of cortical feature maps. Proc Natl Acad Sci USA 87:8345–8349
Obermayer K, Blasdel GG, Schulten K (1992) Statistical–mechanical analysis of self-organization and pattern formation during development of visual maps. Phys Rev A 45:7568–7589
O'Leary DDM, Yates PA, McLaughlin T (1999) Molecular development of sensory maps: Representing sights and smells in the brain. Cell 96:255–269
Ritter H (1991) Asymptotic level density for a class of vector quantization processes. IEEE Trans Neural Netw 2:173–175
Ritter H, Schulten K (1986) On the stationary state of Kohonen's self-organizing sensory mapping. Biol Cybern 54:99–106
Ritter H, Martinetz T, Schulten K (1992) Neural computation and self-organizing maps: an introduction. Addison-Wesley, Reading
Robinson DA (1972) Eye movements evoked by collicular stimulation in the alert monkey. Vis Res 12:179–1808
Rumelhart DE, Zipser D (1985) Feature discovery by competitive learning. Cogn Sci 9:75–112
Schmidt JT (1985) Formation of retinotopic connections: Selective stabilization by an activity dependent mechanism. Cell Mol Neurobiol 5:65–84
Schmidt M (1996) Neurons in the cat pretectum that project to the dorsal lateral geniculate nucleus are activated during saccades. J Neurophysiol 76:2907–2918
Tessier-Lavigne M (1995) Eph receptor tyrosine kinases, axon repulsion, and the development of topographic maps. Cell 82:345–348
Tessier-Lavigne M, Goodman CS (1996) The molecular biology of axon guidance. Science 274:1123–1133
Tsumato T, Nakamura S (1974) Inhibitory organization of the thalamic ventrobasal neurons with different peripheral representations. Exp Brain Res 21:195–210
Tusa RJ, Palmer LA, Rosenquist AC (1978) The retinotopic organization of area 17 (striate cortex) in the cat. J Comp Neurol 177:213–236
Van Hulle MM (1996) Topographic map formation by maximizing unconditional entropy: A plausible strategy for on-line unsupervised competitive learning and nonparametric density estimation. IEEE Trans Neural Netw 7:1299–1305
Van Hulle MM (1997) Nonparametric density estimation and regression achieved with topographic maps maximizing the information-theoretic entropy of their outputs. Biol Cybern 77:49–61
Vanegas H (ed) (1984) Comparative neurology of the optic tectum. Plenum Press, New York
Villmann T, Claussen JC (2006) Magnification control in self-organizing maps and neural gas. Neural Comput 18:446–469
Wallace MT, Wilkinson LK, Stein BE (1996) Representation and integration of multiple sensory inputs in primate superior colliculus. J Neurophysiol 76:1246–1266
Willshaw DJ, von der Malsburg C (1976) How patterned neural connections can be set up by self-organization. Proc R Soc Lond B 194:431–445
Woolsey CN (1981) Cortical sensory organization: multiple somatic areas. Humana Press, Clifton
Zador PL (1982) Asymptotic quantization error of continuous signals and the quantization dimension. IEEE Trans Info Theory IT-28:139–149
Zhang LL, Tao HW, Holt CE, Harris WA, Poo M-M (1998) A critical window for cooperation and competition among developing retinotectal synapses. Nature 395:37–44