Computational Model of the Cerebral Cortex that Performs Sparse Coding Using a Bayesian Network and Self-Organizing Maps

Yuuji Ichisugi[1] and Haruo Hosoya[2]

[1] National Institute of Advanced Industrial Science and Technology(AIST)
Tsukuba Central 2,Tsukuba,Ibaraki 305-8568, Japan
`y-ichisugi@aist.go.jp`
[2] 7th Bld. of Faculty of Science, The University of Tokyo,
Hongo 7-3-1, Bunkyo-ku, Tokyo 113-0033, Japan
`hahosoya@is.s.u-tokyo.ac.jp`

Abstract. The authors have proposed a computational model of the cerebral cortex, called the BESOM model, that combines a Bayesian network and Self-Organizing Maps. In this paper, we add another model of the cerebral cortex, called sparse coding, into our model in a biologically plausible way. In the BESOM model, hyper-columns in the cerebral cortex are interpreted as random variables in a Bayesian network. We extend our model so that random variables can become "inactive." In addition, we apply bias at the time of recognition so that almost all of the random variables may become inactive. This mechanism realizes sparse coding without breaking the theoretical framework of the model based on the Bayesian networks.

Keywords: Sparse coding, Bayesian network, Self-organizing maps, cerebral cortex, BESOM

1 Background

Some computational neuroscientists have begun to understand that the *Bayesian network* [2] is the essential mechanism of the cerebral cortex [6][8][9][11][12][13][14] [15]. Bayesian networks are a technology for knowledge representation that can efficiently express the causal relationships among many random variables. Models based on Bayesian networks can successfully explain the fundamental mechanism of the cerebral cortex, namely, robust pattern recognition using prediction based on context[8]. Furthermore, previous studies strongly suggest that the cerebral cortex is a Bayesian network, according to models that reproduce electrophysiological phenomena[9][15] and models that explain the roles of major anatomical characteristics of the cerebral cortex[11][12]. Moreover, the mechanism of the motor area is explained by using a combination of a Bayesian network and a reinforcement learning mechanism[13]. Another study realizes probabilistic reasoning at the Markov Random Field (a model similar to Bayesian networks) with a biologically plausible neural circuit of spiking neurons[14].

Yuuji Ichisugi, Haruo Hosoya:
"Computational Model of the Cerebral Cortex that Performs Sparse Coding Using a Bayesian Network and Self-Organizing Maps",
In Proc. of 17th International Conference on Neural Information Processing
(ICONIP 2010), Part I, LNCS 6443, pp.33--40, Nov 2010.

Although previous models explain only some parts of the mechanism of the cerebral cortex functions, we believe that, in the not too distant future, these models (maybe including [5]) will become integrated into one universal model based on Bayesian networks.

The authors have also proposed a model of the cerebral cortex using a Bayesian network, called the *BESOM model* [11][13]. This model inherits the basic structure of Neocognitron and its successors[1], which are macroscopic models of the visual area. Moreover, the mechanism of the *Self-Organizing Maps* (SOM)[3], whose origin is the model of the orientation columns of the primary visual area, is adopted as a learning algorithm. BESOM can also be regarded as a novel machine-learning algorithm that uses multiple SOMs, like [10] and [17].

On the other hand, there is another model, called the *sparse coding model* [4], of an aspect of the cerebral cortex. Sparse coding is a kind of unsupervised learning whose goal is to acquire a basis on which to express an input vector by a linear summation of a smaller number of basis vectors. It has been suggested that sparse coding is performed in the primary visual area[4] and primary auditory area[16] of the cerebral cortex. Sparse coding efficiently compresses information and is supposed to conserve both the energy spent by neurons and the cost of maintaining synapses.

In this paper, we propose a biologically plausible computational model that unifies three mechanisms, a Bayesian network, SOM, and sparse coding. The model is an extension of the previous BESOM model.

2 Overview of the Model

2.1 Bayesian Network and MPE

A Bayesian network[2] is a model of knowledge representation that expresses causal relationships between random variables using a directed acyclic graph. Random variables are expressed as *nodes*, and relationships between random variables are expressed as *edges*. Each node has a table of conditional probability, which denotes the degree to which nodes are related to the set of its parent nodes.

In a Bayesian network, an *MPE (most probable explanation)* is the set of values of nodes that most likely explains given observed data. Let $\mathbf{i}$ be a set of values of observed random variables and $\mathbf{h}$ be a set of values of hidden variables (unobserved random variables). MPE $\hat{\mathbf{h}}$ is defined by the following equations.

$$\hat{\mathbf{h}} = \operatorname*{argmax}_{\mathbf{h}} P(\mathbf{h}|\mathbf{i}) = \operatorname*{argmax}_{\mathbf{h}} P(\mathbf{h}, \mathbf{i}) \tag{1}$$

where $P(\mathbf{h}, \mathbf{i})$ is the joint probability of $\mathbf{h}$ and $\mathbf{i}$, which can be calculated by the following formula if a Bayesian network is given:

$$P(\mathbf{h}, \mathbf{i}) = \prod_{x \in \mathbf{h} \cup \mathbf{i}} P(x|parents(x)) \tag{2}$$

where $parents(x)$ denotes the set of values of parent nodes of node X.

2.2 The Structure of Two-Layered BESOM

Figure 1 shows the neural circuit of the two-layered BESOM network used in this paper. The ellipses are *nodes* (random variables), the small circles are *units* (values that random variables can take), and the straight lines are the *connections* (conditional probabilities) between units.

Although every pair of units contained in the two layers (the hidden layer and the input layer) has a connection, most connections are omitted in Fig.1. There is no connection between units in the same layer.

At the time of learning, each node in the hidden layer plays the role of a competitive layer of SOM, learns the *weights of the connections* (conditional probabilities) between units, and compresses the input from its child nodes.

At the time of recognition, all nodes play the role of random variables in a Bayesian network. At this time, all edges of the Bayesian network are from all nodes of the hidden layer to all nodes of the input layer. There is no edge between nodes in the same layer.

The number of nodes, the number of units, and the network structure of nodes are given first, and are not changed by learning.

When BESOM is used as a cerebral cortex model, each node is a hypercolumn, each unit is a column (minicolumn), and each connection weight between a pair of units is the weight of a synapse.

The correspondence of components in BESOM, SOM, a Bayesian network and a cerebral cortex is summarized in Table 1.

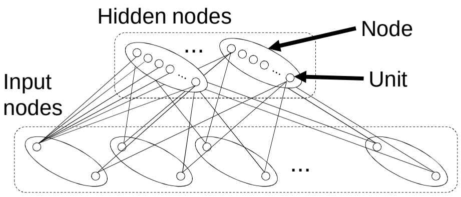

Fig. 1. The structure of two-layered BESOM network.

2.3 Recognition Steps and Learning Steps

When a set of values of input nodes (observed data) is given, BESOM executes a recognition step and a learning step. By repeating this cycle, BESOM acquires an approximated generative model of the outer world.

Let $\mathbf{i}(t)$ be the set of values of input nodes at time t. We assume that each input is generated from i.i.d. (independent identical distribution). The objective of the learning is to maximize the likelihood of the parameter θ (the vector of

Table 1. The correspondence of components in BESOM, SOM, a Bayesian network and a cerebral cortex.

BESOM	SOM	Bayesian network	Cerebral cortex
node	competitive layer	random variable	hyper-column
unit	element of input vector, unit of competitive layer	value of random variables	column
parent node	competitive layer	parent node (cause)	upper area
child node	input layer	child node (result)	lower area
output of unit	the winner of competition	MPE	response of column
weight of connection	element of reference vector	conditional probability	weight of synapse

all the elements of all conditional probability tables).

$$\theta^* = \underset{\theta}{\operatorname{argmax}}(\prod_{i=1}^{t} P(\mathbf{i}(i)|\theta))P(\theta) \tag{3}$$

$$= \underset{\theta}{\operatorname{argmax}}(\prod_{i=1}^{t} \sum_{\mathbf{h}} P(\mathbf{h}, \mathbf{i}(i)|\theta))P(\theta) \tag{4}$$

Although the algorithm described in this paper is somewhat complex, its essence can be simply expressed by the following two equations.

At the recognition steps, the estimated values of hidden variables $\hat{\mathbf{h}}(t)$ (MPE) based on the current parameter $\theta(t)$ are calculated by the following equation.

$$\hat{\mathbf{h}}(t) = \underset{\mathbf{h}}{\operatorname{argmax}}\, P(\mathbf{h}, \mathbf{i}(t)|\theta(t)) \tag{5}$$

At the following learning step, the parameter $\theta(t+1)$ is estimated by Eq. (4), with an approximation that replaces the marginalization of $\mathbf{h}$ with the estimated value $\hat{\mathbf{h}}(i)$.

$$\theta(t+1) = \underset{\theta}{\operatorname{argmax}}(\prod_{i=1}^{t} P(\hat{\mathbf{h}}(i), \mathbf{i}(i)|\theta))P(\theta) \tag{6}$$

The exact calculation of Eq. (5) and Eq. (6) requires enormous amounts of computation. On the other hand, the actual brain should execute both recognition and learning very efficiently with some clever approximation. In addition, the brain should perform online learning; that is, $\theta(t+1)$ should be calculated only by using $\mathbf{i}(t)$, $\hat{\mathbf{h}}(t)$ and $\theta(t)$. We think the algorithm described in this paper satisfies these restrictions and is thus a plausible brain model.

2.4 Basic Idea of Sparse Coding for BESOM

We add the mechanism of sparse coding to BESOM by introducing an “inactive state” into each random variable (i.e., a hyper-column). If the inactive state is introduced to each random variable and large numbers of nodes become inactive at each recognition step, sparse coding will be realized in BESOM. (See Fig. 2.) Actually, the way in which inactive states are introduced into a Bayesian network is not obvious. Moreover, in order for the mechanism to become an appropriate model of the cerebral cortex, it should be implemented in a biologically plausible way. Detailed recognition and learning algorithms that realize the idea of inactive states are described in the next section.

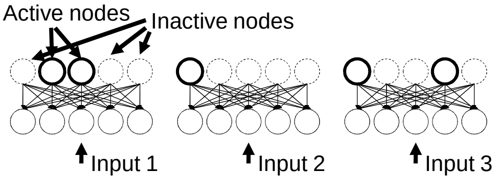

Fig. 2. Basic idea of sparse coding using BESOM. Different sets of hidden nodes become inactive depending on inputs.

3 Algorithm

3.1 Characteristics of the Acquired Bayesian Network

We restrict the Bayesian network acquired by the proposed algorithm to satisfy the following conditions.

1. Let X be a node (a random variable). X's value should be one of the following $s+1$ values.
$$X \in \{x_\phi, x_1, x_2, \cdots, x_{s-1}, x_s\} \tag{7}$$
We call the value x_ϕ the "ϕ-value" and the values other than x_ϕ "non-ϕ-values." The value x_ϕ means the node X is inactive.
2. A conditional probability table $P(x_i|u_1, \cdots, u_m)$ $(i = \phi, 1, \cdots, s)$ should satisfy the following equation:
$$P(x_i|u_1, \cdots, u_m) = \frac{1}{m}\sum_{k=1}^{m} P(x_i|u_k) \tag{8}$$
The constant $1/m$ normalizes so that $\sum_i P(x_i|u_1, \cdots, u_m)$ becomes 1.
3. Let U be a node and X be a child node of U. The ϕ-value of U, u_ϕ, should not have a causal relation to X. That is, the following equation should hold.
$$P(x_i|u_\phi) = P(x_i)\ (i = \phi, 1, \cdots, s) \tag{9}$$
This condition will be satisfied when the learning converges because of the learning rules described in Section 3.3 .

3.2 Recognition Step

In the original sparse coding algorithm[4], sparseness is realized by adding the penalty term for activeness to the objective function. We apply this idea to BESOM.

Eq. (2), which calculates the joint probability of an MPE candidate $\mathbf{h}$ and an input $\mathbf{i}$, is modified as follows, so that a penalty is given depending on the number of activity nodes:

$$P(\mathbf{h}, \mathbf{i}) = e^{-\beta A(\mathbf{h})} \prod_{x \in \mathbf{h} \cup \mathbf{i}} P(x | parents(x)) \tag{10}$$

where β is the parameter that controls the sparseness, $A(\mathbf{h})$ is the number of active nodes in $\mathbf{h}$.

The formula for this joint probability is used at the recognition steps that calculate MPE. We do not calculate strict MPE in the present simulation. The approximate MPE is calculated by the hill-climbing method. In the actual brain, a variation of the belief revision algorithm[2], which seems biologically more plausible[14], may be used.

If MPE is used as a learning step as it is, the learning will be likely to fall into a local minimum. Then, to avoid the local minimums, we add a moderate amount of noise to the recognition results in the early stages of the learning.

3.3 Learning Step

At each a learning step, the weights of connections between units will be updated according to the MPE calculated at the preceding recognition step. In the nodes at the hidden layer, the units that correspond to the values of MPE are regarded as winners for competitive learning.

In this section, we clarify our explanation by explaining a simplified version of the learning rule, which does not include neighborhood-learning rules.

Let us call the units corresponding to the value of MPE *winner units.* In the input layer, winner units represent observed data. In the hidden layer, winner units represent estimated values of hidden variables. Let X be a node at the hidden layer and Y_l $(l = 1, \cdots, n)$ be a child node of X at the input layer.

The connection weight w_{ij}^l between the winner unit x_i and unit y_j^l is updated by the following formulas:

1. When $i = \phi$:

$$w_{\phi j}^l = \begin{cases} \Phi_{Y_l} & (j = \phi) \\ (1 - \Phi_{Y_l})/s & (j \neq \phi) \end{cases} \tag{11}$$

   where Φ_{Y_l} is the frequency that Y_l becomes a ϕ-value in MPE; these Φ_{Y_l} are learned separately.
2. When $i \neq \phi, j \neq \phi$:

$$w_{ij}^l \leftarrow w_{ij}^l + \alpha(v_j^l - w_{ij}^l) \tag{12}$$

   where α is the learning rate, v_j^l is an input from the child node Y_l, whose value is 1 if unit y_j^l is the winner unit, and is 0 otherwise.

3. When $i \neq \phi, j = \phi$:

$$w^l_{i\phi} = 1 - \sum_{j=1}^{s} w^l_{ij} \tag{13}$$

The connection weight w^l_{ij} is learned by the above algorithm. When the estimated values of the hidden variables are regarded as true observed values and the learning rate α is appropriately scheduled, the connection weight w^l_{ij} becomes a maximum likelihood estimator of the conditional probability $P(Y_l = y^l_j | X = x_i)$[11].

The obtained conditional probabilities are used at the next recognition step to calculate joint probability, defined as Eq. (10), assuming the constraints of Eq. (8).

4 Experiment: Sparse Coding of Natural Images

We used images provided by Olshausen (the images filtered with whitening/low-pass as described in [7]) and clipped to the range $[0, 1]$. The images are used as input to a two-layered BESOM, with 4 nodes in the hidden layer and 49 in the input layer. At each step, we extracted a image patch with 7x7=49 pixels from a random position. Then, we gave the pixel intensities in the image patch to the binary input nodes. The value of each input node is set to 1 according to the pixel intensity taken as a probability. For example, for intensity 0.2, the value was set to 1 with probability 0.2. The parameter that determines the sparseness of node activity is set to $\beta = 8$.

Figure 3(a) shows a learning result. It shows the values of the conditional probability tables of the ϕ-value units $P(y^l_1 | x_\phi)$ and non-ϕ-value unit $P(y^l_1 | x_i)$ $(i = 1, \cdots, 9, l = 1, \cdots, 49)$ as the brightness of 7x7 pixels.

It is shown that the non-ϕ-value unit of each node obtained the conditional probability table with orientation selectivity like V1 simple cells[4]. The ϕ-value unit of each node learned the mean of input images. In this experiment, we found that 0-3 nodes are activated according to the inputs.

Figure 3(b) shows a learning result with the sparseness parameter $\beta = 0$. In this case, every base image is close to the mean image of input because all nodes tend to be active. The result shows weak orientation selectivity.

References

1. K. Fukushima, Neural network model for selective attention in visual-pattern recognition and associative recall, APPLIED OPTICS 26 (23): 4985-4992 Dec 1 1987.
2. J. Pearl , Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, 1988.
3. T. Kohonen, Self-Organizing Maps. Springer-Verlag, 1995.
4. Olshausen BA, Field DJ, Emergence of simple-cell receptive field properties by learning a sparse code for natural images, NATURE 381 (6583): 607-609 JUN 13 1996.

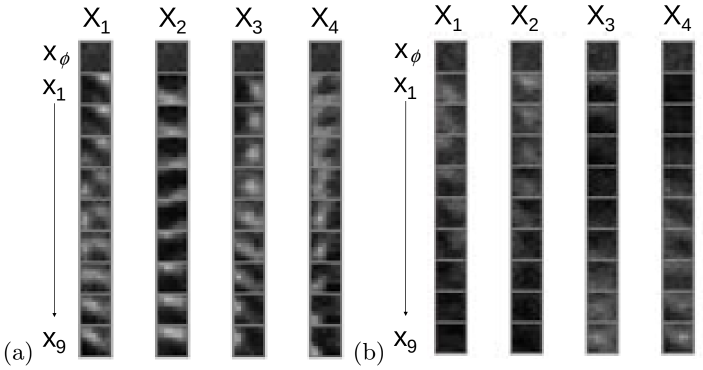

Fig. 3. Learning result of sparse coding of natural images. The sparseness parameter values are $\beta = 8$ (a) and $\beta = 0$ (b) .

5. Coward, L.A., The Recommendation Architecture: lessons from the design of large scale electronic systems for cognitive science. Journal of Cognitive Systems Research 2(2), 111-156, 2001.
6. Lee, T.S., Mumford, D. , Hierarchical Bayesian inference in the visual cortex. Journal of Optical Society of America, A. . 20(7): 1434-1448, 2003.
7. Bruno A. Olshausen and David J. Field, Sparse coding with an overcomplete basis set: A strategy employed by V1? Vision Research, 37(23):3311.3325, 2003.
8. George, D. Hawkins, J., A hierarchical Bayesian model of invariant pattern recognition in the visual cortex, In proc. of IJCNN 2005, vol. 3, pp.1812-1817, 2005.
9. R. Rao., Bayesian inference and attention in the visual cortex. Neuroreport 16(16), 1843-1848, 2005.
10. Oshiro N, Kurata K, Separating visual information into position and direction by two inhibitory-connected SOMs. Artif Life and Robotics 9(2):86.89, 2005.
11. Yuuji ICHISUGI, The cerebral cortex model that self-organizes conditional probability tables and executes belief propagation, In Proc. of International Joint Conference on Neural Networks (IJCNN2007), pp.1065–1070, Aug 2007.
12. Florian Roehrbein, Julian Eggert, and Edgar Koerner, Bayesian Columnar Networks for Grounded Cognitive Systems, In Proc. of the 30th Annual Conference of the Cognitive Science Society, pp.1423–1428, 2008.
13. Haruo Hosoya: A motor learning neural model based on Bayesian network and reinforcement learning, In Proceedings of International Joint Conference on Neural Networks, 2009.
14. Shai Litvak, Shimon Ullman: Cortical Circuitry Implementing Graphical Models, Neural Computation 21, 3010.3056, 2009.
15. Chikkerur, S., T. Serre, C. Tan and T. Poggio, What and Where: A Bayesian Inference Theory of Attention, Vision Research, 2010.
16. Hiroki Terashima and Haruo Hosoya, Sparse codes of harmonic natural sounds and their modulatory interactions. Network: Computation in Neural Systems, 20(4):253-267, 2009.
17. Tetsuo Furukawa, SOM of SOMs, Neural Networks, Vol.22, Issue 4, pp.463-478,May 2009.