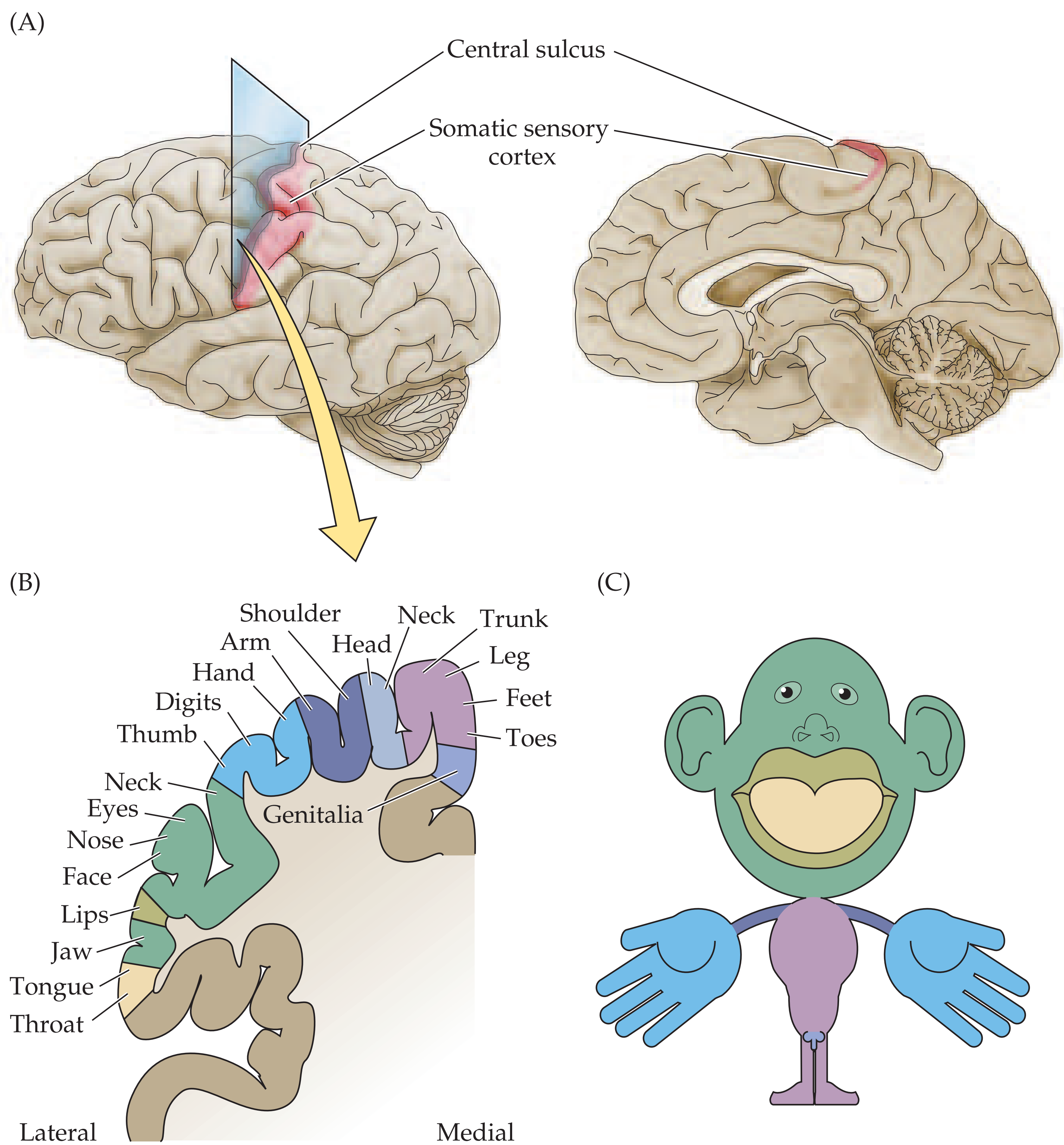
Machine Learning Lab
University of Freiburg

Self-Organizing Map (SOM)

Machine Learning
Summer 2015

Dr. Joschka Boedecker

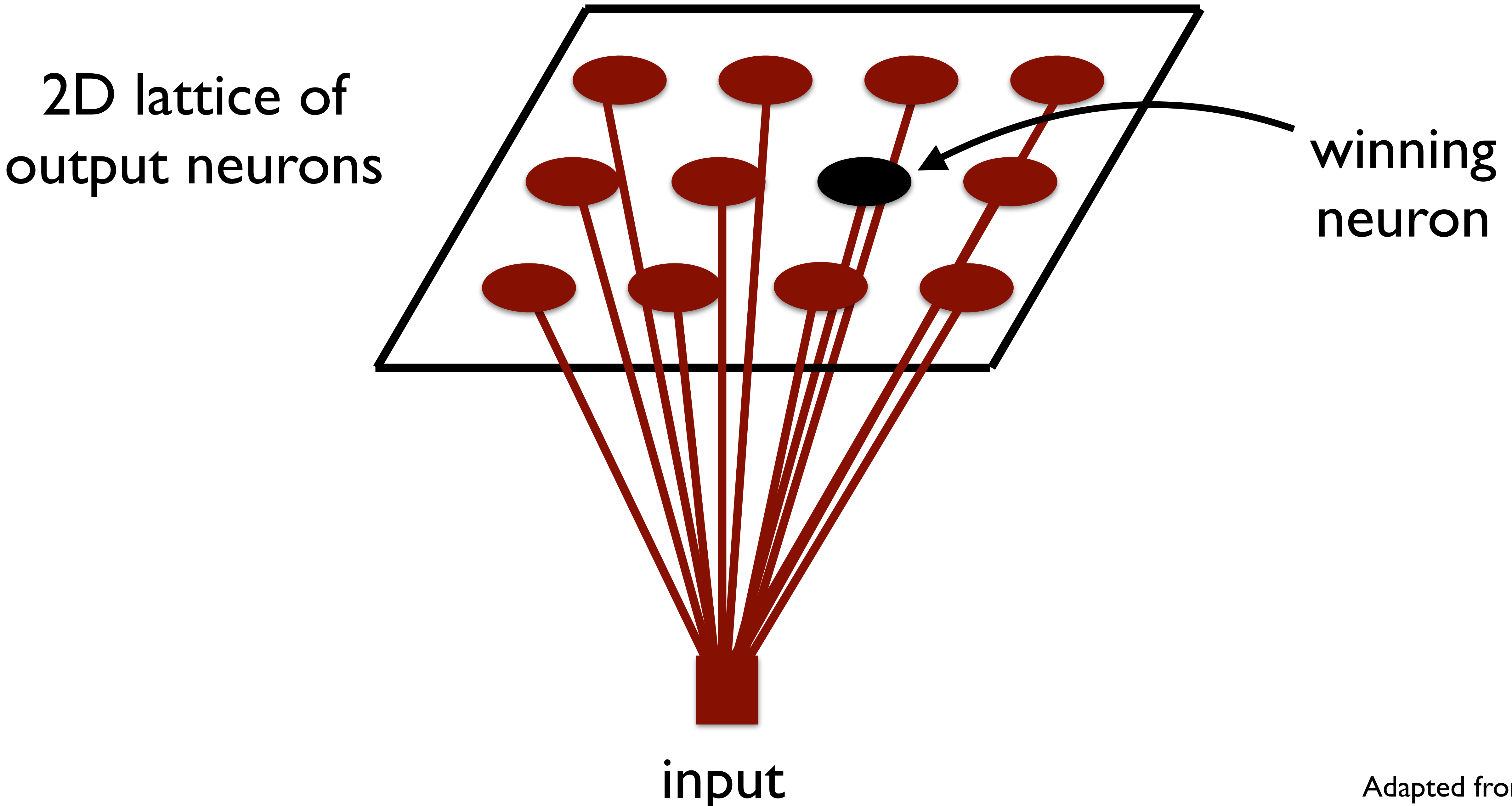
Topological Maps in the Brain



Source:
(Purves et al., 2004)

Self-Organizing Map (SOM)

Kohonen Model



Adapted from:
(Haykin, 2009)

Basic Idea

- Three important processes in the formation of the map:
  1. Competition: Each neuron computes value of a discriminant function. The neuron with the largest value wins the competition. This is reminiscent of *long-range inhibition* in the brain.
  2. Cooperation: The winning neuron determines the spatial location of a topological neighborhood for cooperation of excited neurons. This corresponds to *short-range excitation.*
  3. Synaptic Adaptation: Enable the *excited* neurons to increase their values of the discriminant function in relation to the input patterns.

Competitive Process

- Denote m-dimensional input pattern by:

$$\mathbf{x} = [x_1, x_2, \ldots, x_m]^T$$

- Synaptic weight vector of neuron *j*:

$$\mathbf{w}_j = [w_{j1}, w_{j2}, \ldots, w_{jm}]^T, \quad j = 1, 2, \ldots, l$$

- Discriminant function is the inner product between neuron *j*'s weight vector and the input vector:

$$\mathbf{w}_j^T \mathbf{x}$$

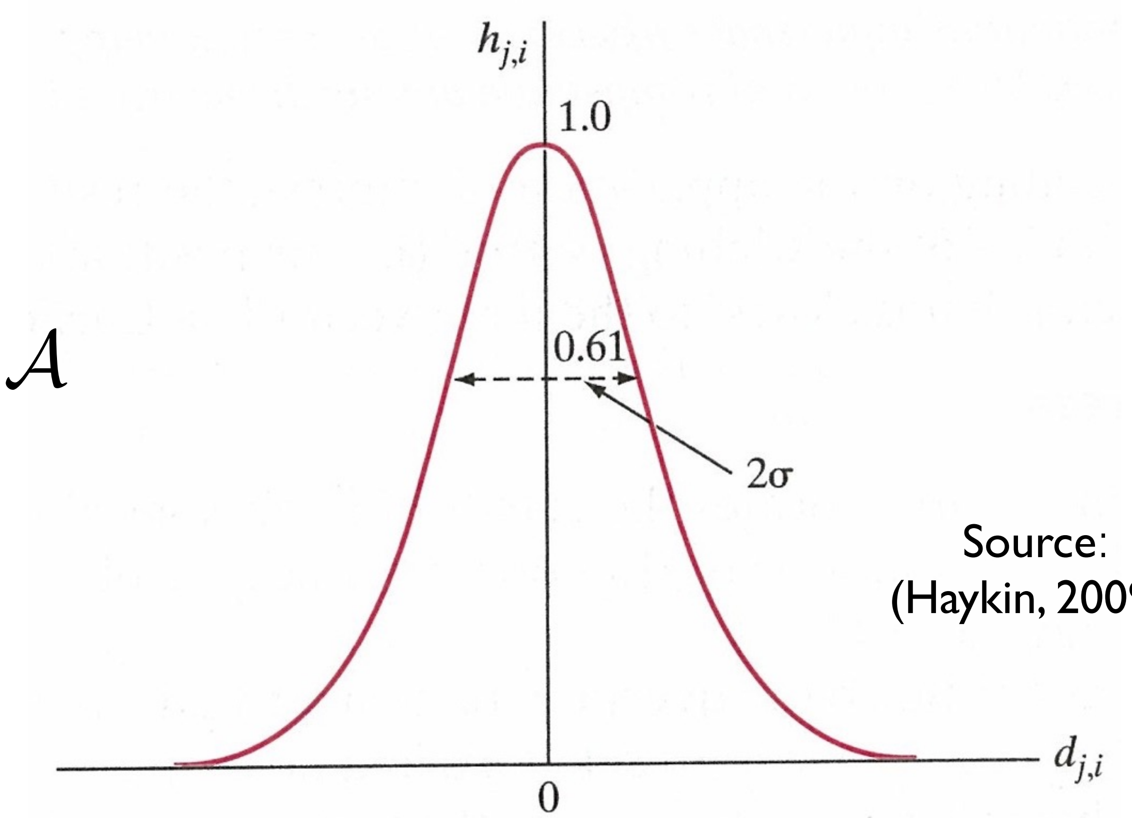
- Determine index i(**x**) as:

$$i(\mathbf{x}) = \arg\min_j \|\mathbf{x} - \mathbf{w}_j\|, \quad j \in \mathcal{A}$$

Cooperative Process

- Neurobiological data suggests that the topological neighborhood should be:
  - symmetric about the winning neuron *i*
  - monotonically decreasing with distance $d_{j,i}$ to the winning neuron
- Good choice for the neighborhood function $h_{j,i}$ given these requirements is a Gaussian:

$$h_{j,i(\mathbf{x})} = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2}\right) \quad j \in \mathcal{A}$$



Source:
(Haykin, 2009)

Cooperative Process

- For 1D lattice: $d_{j,i} = |j - i|$
- For 2D lattice: $d_{j,i}^2 = \|\mathbf{r}_j - \mathbf{r}_i\|^2$

  with $\mathbf{r}_j$ being the position of the excited neuron *j*
  and $\mathbf{r}_i$ the position of the winning neuron

- Topological neighborhood is allowed to shrink with time, e.g. exponentially:

$$\sigma(n) = \sigma_0 \exp\left(-\frac{n}{\tau_1}\right) \quad n = 0, 1, 2, \ldots$$

- Time-varying form of this neighborhood is the topological neighborhood function:

$$h_{j,i(\mathbf{x})}(n) = \exp\left(-\frac{d_{j,i}^2}{2\sigma^2(n)}\right) \quad n = 0, 1, 2, \ldots$$

Adaptive Process

- Hebbian Learning: "Neurons that fire together, wire together", increase synaptic weight with simultaneous occurrence of pre- and postsynaptic spike
- In its basic forms, it drives changes in only one direction, so we introduce a *forgetting term*: $g(y_j)\mathbf{w}_j$
- Weight change is then expressed as:

$$\Delta\mathbf{w}_j = \eta y_i \mathbf{x} - g(y_j)\mathbf{w}_j$$

with $g(y_j) = \eta y_j$ and $y_j = h_{j,i(\mathbf{x})}$

the weight change becomes: $\Delta\mathbf{w}_j = \eta h_{j,i(\mathbf{x})}(\mathbf{x} - \mathbf{w}_j)$

Adaptive Process

- Weight updates performed by:

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n) h_{j,i(\mathbf{x})}(n)(\mathbf{x}(n) - \mathbf{w}_j(n))$$

- Decrease learning-rate over time:

$$\eta(n) = \eta_0 \exp\left(-\frac{n}{\tau_2}\right), \quad n = 0, 1, 2, \ldots$$

Summary of the SOM Algorithm

1. Initialization: Choose small random values for the initial weight vectors $\mathbf{w}_j(0)$ for all neurons j = 1,2,...,l where l is the number of neurons in the lattice

2. Sampling: Draw sample x from the input with a certain probability and apply to the lattice

3. Similarity Matching: Find the best-matching (winning) i(**x**) at time step *n* using:

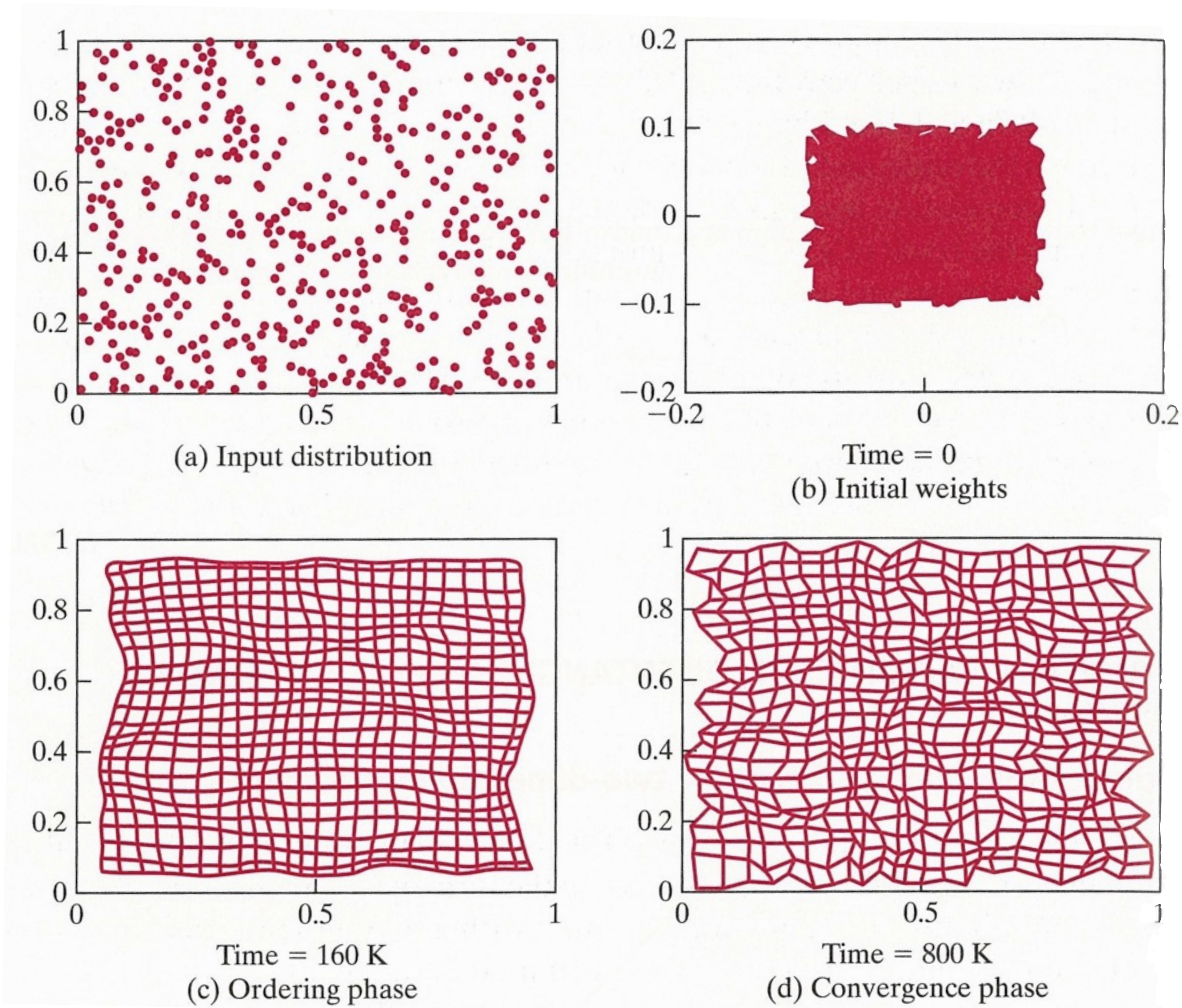
$$i(\mathbf{x}) = \arg\min_j \|\mathbf{x}(n) - \mathbf{w}_j\|, \quad j = 0, 1, 2, \ldots, l$$

4. Updating: Adjust weight vectors of all excited neurons by:

$$\mathbf{w}_j(n+1) = \mathbf{w}_j(n) + \eta(n) h_{j,i(\mathbf{x})}(n)(\mathbf{x}(n) - \mathbf{w}_j(n))$$

5. Continuation: Continue with step 2 until no noticeable changes in the map are observed

Examples



(a) Input distribution

(b) Initial weights

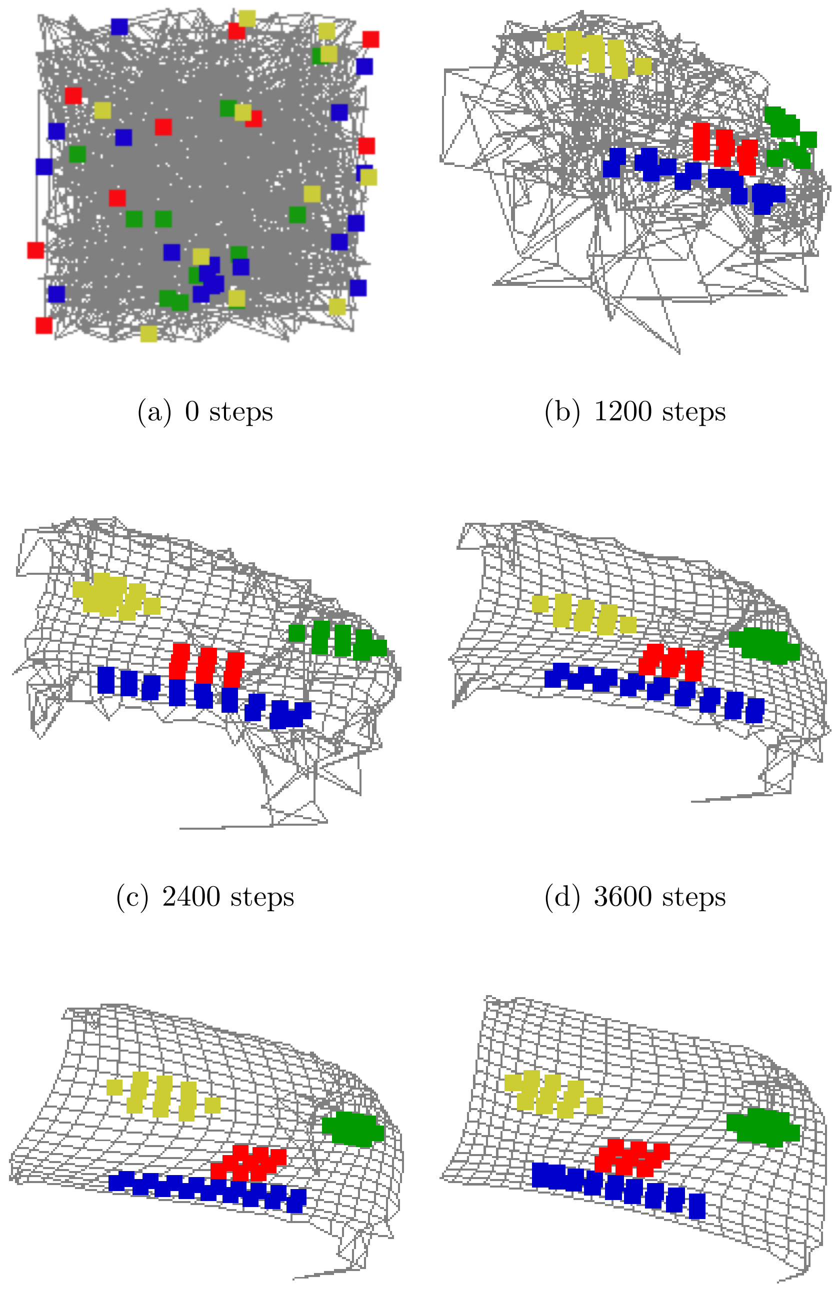
(c) Ordering phase

(d) Convergence phase

Source: (Haykin, 2009)

Application: Robotics

- Organize tactile sensors on a robot's face



(a) 0 steps (b) 1200 steps

(c) 2400 steps (d) 3600 steps

(e) 4800 steps (f) 7200 steps

Source:
(Fuke, 2009)

References

(Haykin, 2009) S. Haykin, Neural Networks and Learning Machines, 3rd edition Prentice Hall, 2009.

(Purves et al., 2004) D. Purves, G.J. Augustine, Neuroscience, D. Fitzpatrick, W.C. Hall, A. S. LaMantia, J.O. McNamara, S.M. Williams, Neuroscience, 3rd edition, Sinauer Associates Inc., 2004.

(Fuke, 2009) S. Fuke, Multimodal body representation based on visuo-proprioceptive association triggered by attention and synchrony, PhD Thesis, Graduate School of Engineering, Osaka University, 2009.

Software

- **SOM**-Toolbox: Matlab toolbox for SOMs and more: http://www.cis.hut.fi/somtoolbox/