The line process model (I)

- Our first example is the classic *line process* model (Geman & Geman, 1984; Blake & Zisserman, 2003; Mumford & Shah, 1989), which was developed as a way to segment images. It has explicit *line process* variables that "break" images into regions where the intensity is piecewise smooth. Our presentation follows the work of Koch et al. (1986), who translated it into neural circuits.
- The model takes intensity values $\vec{I}$ as input, and outputs smoothed intensity values. But this smoothness is broken at places where the intensity changes are too high. The model has continuous variables $\vec{J}$ representing the intensity, and binary-valued variables $\vec{l}$ for the line processes (or edges). The model is formulated as performing *maximum a posteriori* (MAP) estimation. The algorithm for estimating MAP is a neural network model that can be derived from the original Markov model (Geman & Geman, 1984) by mean field theory (Geiger & Yuille, 1991). Note that in this model, the variables do not have to represent intensity. Instead they can represent texture, depth, or any other property that is spatially smooth except at sharp discontinuities.

The line process model (II)

- For simplicity we present the weak membrane model in one dimension. The input is $\vec{I} = \{I(x) : x \in \mathcal{D}\}$; the estimated, or smoothed, image is $\vec{J} = \{J(x) : x \in \mathcal{D}\}$; and the line processes are denoted by $\vec{l} = \{l(x) : x \in \mathcal{D}\}$, where $l(x) \in \{0, 1\}$.
- The model is specified by a posterior probability distribution:

$$P(\vec{J}, \vec{l} | \vec{I}) = \frac{1}{Z} \exp\{-E[\vec{J}, \vec{l} : \vec{I}]/T\},$$

where

$$E[\vec{J}, \vec{l} : \vec{I}] = \sum_x (I(x) - J(x))^2 + A \sum_x (J(x+1) - J(x))^2 (1 - l(x)) + B \sum_x l(x).$$

The line process model (III)

The first term ensures that the estimated intensity $J(x)$ is close to the input intensity $I(x)$. The second encourages the estimated intensity $J(x)$ to be spatially smooth (e.g., $J(x) \approx J(x+1)$), unless a line process is activated by setting $l(x) = 1$. The third pays a penalty for activating a line process. The result encourages the estimated intensity to be piecewise smooth unless the input $I(x)$ changes significantly, in which case a line process is switched on and the smoothness is broken. The parameter T is the variance of the probability distribution and has a default value $T = 1$.

The line process model illustration

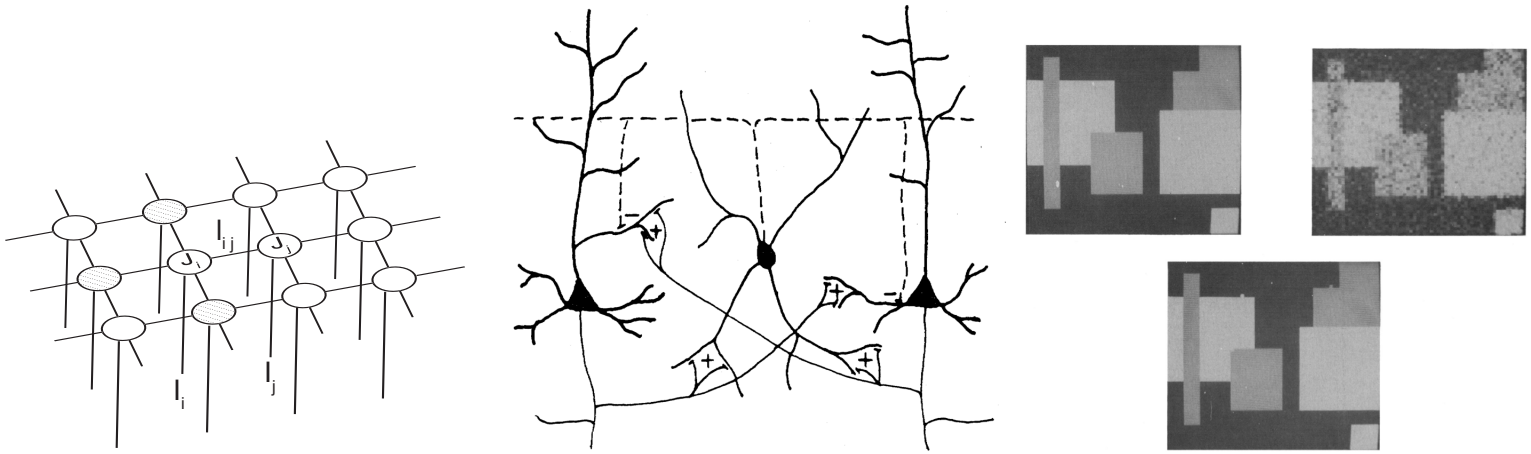

Figure 28 : A representation of the line process model (left) compared to a real neural network (center). On the right, the original image (upper left), the image corrupted with noise (upper right), and the image estimated using the line process model (bottom).

The line process model and neural circuits (I)

- This model can be implemented by a neural circuit (Koch et al., 1986). The connections between these neurons is shown in the previous figure. To implement this model Koch et al., (1986) proposed a neural net model that is equivalent to doing mean field theory on the weak membrane MRF (as discussed earlier) by replacing the binary-valued line process variables $l(x)$ by continuous variables $q(x) \in [0, 1]$ (corresponding roughly to the probability that the line process is switched on).
- This gives an algorithm that updates the regional variables $\vec{J}$ and the line variables $\vec{q}$ in a coupled manner. It is helpful, as before, to introduce a new variable $\vec{u}$ which relates by $q(x) = \frac{1}{1+\exp\{-u(x)/T\}}$ and $u(x) = T \log \frac{q(x)}{1-q(x)}$.

The line process model and neural circuits (II)

$$\frac{dJ(x)}{dt} = -2(J(x) - I(x))$$

$$= -2A\{(1 - q(x))(J(x) - J(x+1)) + (1 - q(x-1))(J(x) - J(x-1))\}, \quad (32)$$

$$\frac{dq(x)}{dt} = \frac{1}{T} q(x)(1 - q(x))\{A(J(x+1) - J(x))^2 - B - T \log \frac{q(x)}{1 - q(x)}\}, \quad (33)$$

$$\frac{du(x)}{dt} = -u(x) + A(J(x+1) - J(x))^2 - B. \quad (34)$$

The update rule for the estimated intensity $\vec{J}$ behaves like nonlinear diffusion, which smooths the intensity while keeping it similar to input $\vec{I}$. The diffusion is modulated by the strength of the edges $\vec{q}$. The update for the lines $\vec{q}$ is driven by the differences between the estimated intensity; if this is small, then the lines are not activated.

The line process model and neural circuits (III)

This algorithm has a Lyapunov function $L(\vec{J}, \vec{q})$ (derived using mean field theory methods) and so will converge to a fixed point, with

$$L(\vec{J}, \vec{q}) = \sum_x (I(x) - J(x))^2 + A \sum_x (J(x+1) - J(x))^2 (1 - q(x)) + B \sum_x q(x) + T \sum_x \{q(x) \log q(x) + (1 - q(x)) \log(1 - q(x))\}. \quad (35)$$

Relations to electrophysiology (I)

- There is some evidence that a generalization of this models roughly matches the electrophysiological findings for those types of stimuli. The generalization is performed by replacing the intensity variables $I(x), J(x)$ by a filterbank of Gabor filters so that the weak membrane model enforces edges at places where the texture properties change (Lee et al., 1992). The experiments, and their relation to the weak membrane models are reviewed in (Lee & Yuille, 2006). The initial responses of the neurons, for the first 80 msec, are consistent with the linear filter models described earlier. But after 80 msec, the activity of the neurons changes and appears to take spatial context into account.
- While the weak membrane model is broadly consistent with the perceptual phenomena of segmentation and "filling in," the types of filling in, their dynamics, and the neural representations of contours and surface are complicated (von der Heydt, 2002; Komatsu, 2006). Exactly how contour and surface information is represented and processed in cortex is an active topic of research (Grossberg & Hong, 2006; Roe et al., 2012).

Relations to electrophysiology (II)

- The findings of the electrophysiological experiments are summarized as follows:
  (1) There are two sets of neurons, with one set encoding regional properties (such as average brightness), and the other set coding boundary location (in agreement with J and I variable in the model, respectively).
  (2) The processes for computing the region and the boundary representations are tightly coupled, with both processes interacting with and constraining each other (as in the dynamical equations above).
  (3) During the iterative process, the regional properties diffuse within each region and tend to become constant, but these regional properties do not cross the region (in agreement with the model).
  (4) The interruption of the spreading of regional information by boundaries results in sharp discontinuities in the responses across two different regions (in agreement with the model). The development of abrupt changes in regional responses also results in a gradual sharpening of the boundary response, reflecting increased confidence in the precise location of the boundary.
- These findings are roughly consistent with neural network implementations of the weak membrane model. But other explanations are possible. For example, the weak membrane model requires lateral (sideways) interaction, and it is possible that the computations are done hierarchically using feedback from V2 to V1.

Relations to electrophysiology illustration

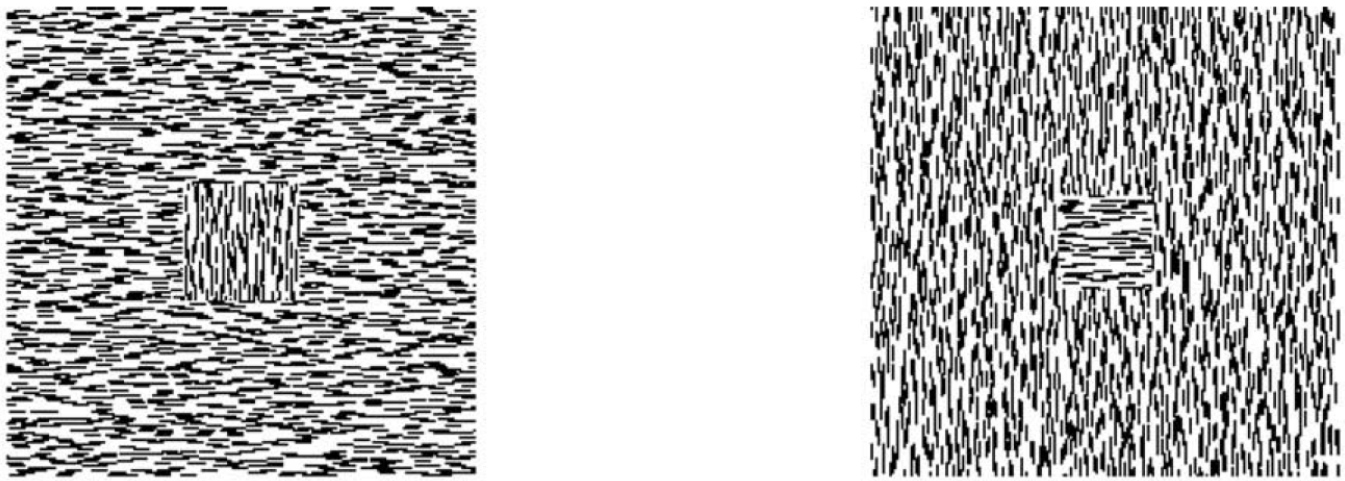

Figure 29 : The stimuli for the experiments by TS Lee and his collaborators (Lee & Yuille, 2006).

Edge detection with spatial context (I)

- Our second example is to develop a model for detecting edges using spatial context. This relates to the phenomena known as association fields, see chapter figure 12.26 (left panel), where Gabor filters that are spatially aligned (in orientation and direction) get grouped into a coherent form.
- For this model, we have a set of neurons at every spatial position x, each tuned to a different angle $\theta_i : i = 1, ..., 8$, and a default cell at angle θ_0. The first cells are designed to detect edges at each orientation – i.e., they can be driven by the log-likelihood ratio of an edge detector at orientation θ_i at this position. The default cell is a dummy that is intended to fire if there is no edge present at this position. This organization forms a population of cells arrayed according to orientation (similar to a hypercolumn in V1).