A computational feature binding model

Eduardo Tusa

et109@hw.ac.uk

March 25, 2013

HERIOT WATT UNIVERSITY

Binding Problem

Overview

Information processing in the human brain is highly parallel. This means that different features of an object are processed in different parts of the brain.

Figure: Brain

Binding Problem

Overview

Information processing in the human brain is highly parallel. This means that different features of an object are processed in different parts of the brain.

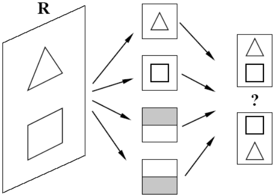


Figure: Binding Problem Illustration [Rosenblatt, 62]

Binding Problem

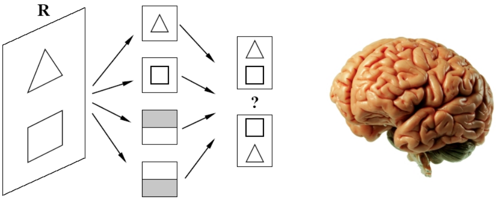


Figure: Binding Problem Illustration [Rosenblatt, 62]

Question?

How features that are processed in parallel are bound to the one unique percept?

The competitive layer model

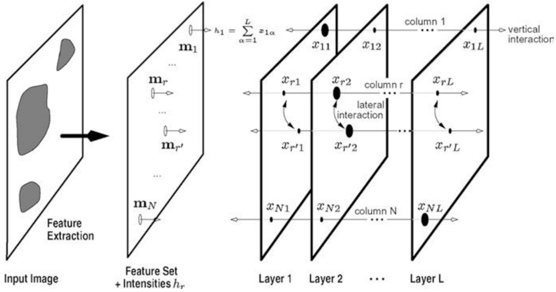
Principles

1. Finite set of features: Representation of the structure of the sensory input.
2. Measure of the mutual compatibility: Gestalt Laws.

Objective

To use these compatibilities to partition the input features into salient groups by the recurrent dynamics in a layered neural network with topographically structured competitive and cooperative interactions.

The competitive layer model



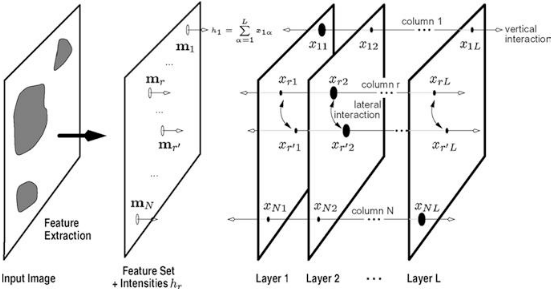
Architecture

Image positions: $r = (x, y) \in I$

Stimulus: m_r - spatial, texture or edge information.

Significance of the detection feature: h_r

The competitive layer model



Architecture

Number of Layers: L

Number of formal neurons with nonnegative activity $x_{r\alpha} \geq 0$: N

Superposition condition: $\Sigma_{\alpha=1}^{L} x_{r\alpha} \approx h_r$

The competitive layer model

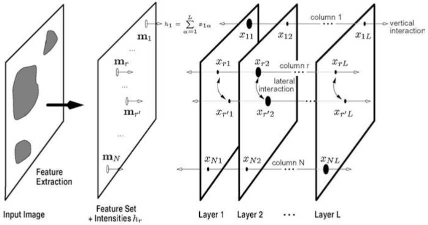
Energy function

$$E = \frac{J}{2}\Sigma_r(\Sigma_\beta x_{r\beta} - h_r)^2 + \frac{1}{2}\Sigma_\alpha \Sigma_{rr'} f_{rr'} x_{r\alpha} x_{r'\alpha}$$

Interaction function

$$\dot{x_{r\alpha}} = -x_{r\alpha} + \sigma(J(h_r - \Sigma_\beta x_{r\beta}) + \Sigma_{r'} f_{rr'} x_{r'\alpha})$$

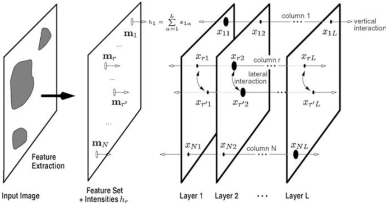
where $\sigma(x) = max(x, 0)$ and $f(m_r, m_{r'} = f_{rr'})$



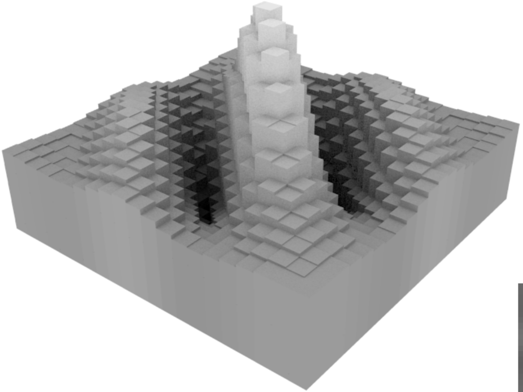
The competitive layer model

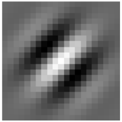
Image Segmentation

1. Initialize all $x_{r\alpha}$ with small random values $x_{r\alpha}(t=0) \in [h_r/L - \epsilon, h_r/L + \epsilon]$
2. Do NL times: choose (r, α) randomly and update $x_{r\alpha} = max(0, \xi)$, where $\xi = \frac{J(h_r - \Sigma_{\beta \neq \alpha} x_{r\beta}(t)) + \Sigma_{r' \neq r} f_{rr'} x_{r'\alpha}(t))}{J - f_{rr'}}$
3. Go to step 2 until convergence



Multiscale Filtering





Multiscale Filtering

2D Gabor Filters

$$g(x,y) = e^{\left(\frac{(x-x_0)^2}{2\sigma_x^2}+\frac{(y-y_0)^2}{2\sigma_y^2}\right)} e^{-ik(x-x_0)}$$

2D Gabor Filters

Center of the receptive field: (x_0, y_0)
Widths of the Gaussian envelope along the x- and y-axes: σ_x and σ_y
Spatial frequency of a complex plane wave with wave normal along the x-axis and wavelength λ: $k = 2\pi/\lambda$

Multiscale Filtering

2D Gabor Filters

$$g(x,y) = e^{\left(\frac{(x-x_0)^2}{2\sigma_x^2} + \frac{(y-y_0)^2}{2\sigma_y^2}\right)} e^{-ik(x-x_0)}$$

Gabor Wavelets

$g_{mn} = g(x', y')$
$x' = a^{-m}(x \cos \Theta_n + y \sin \Theta_n)$ and
$y' = a^{-m}(-x \sin \Theta_n + y \cos \Theta_n)$

Gabor Wavelets

A sparse sampling of the phase space (which is spanned by m, n, x_0 and y_0) is sufficient for a complete representation of arbitrary image data.

Multiscale Filtering

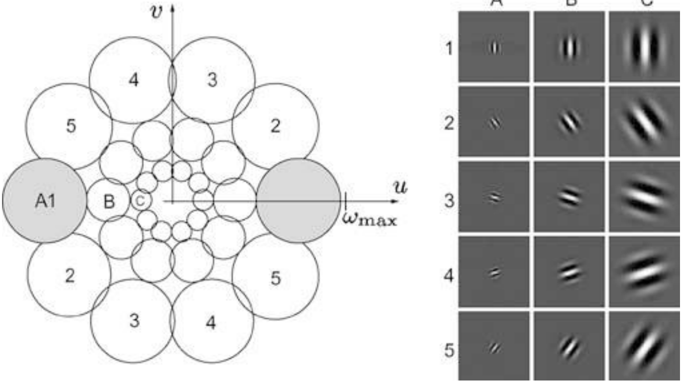


Figure: Set of 2D Gabor Filters for Feature Extraction

Multiscale Filtering

Algorithm

1. Compute a set of 2D Gabor filters tuned to five different orientations and three scales.
2. For each element of the filter bank, compute its response image, or channel, c_{mn} to the given input.
3. Apply the nonlinear contrast transfer function to each channel c_{mn}, yielding c_{mn}^{CTF}.
4. Compute the texture features μ_{mn} and σ_{mn} for each of the 15 channels according to steps (3) and (4).

$$\mu_{mn} = c_{mn}^{CTF} * gs_{mn}(x, y)$$

$$\sigma_{mn} = \sqrt{(c_{mn}^{CTF} - \mu_{mn})^2 * gs_{mn}(x, y)}$$

1. Introduction
2. The competitive layer model
3. Feature Extraction
4. **The CLM texture segmentation model**
5. Results
6. Conclusions

The CLM texture segmentation model

Multi-dimensional scaling

Feature space: 30-dimensional hyperellipsoid t_r
Principal component analysis: 4-dimensional p_r

Feature subsampling

Image 256 x 256 pixels: 65536 features
1024 8 x 8 regions: 1024 4D feature vectors p_r
Nearest neighbour: $|p_r - p_{near}| + |r - r_{near}| = min.$

The CLM texture segmentation model

Interaction matrix

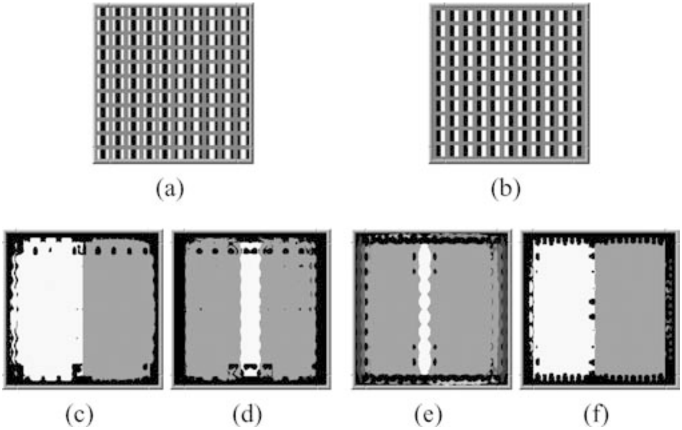
$$f_{rr'} = e^{-\frac{d_{text}(r,r')^2}{R_{sim}^2}} + ce^{-\frac{|x_r - x_{r'}|^2}{R_{prox}^2}} - k$$

$$d_{text}(r,r') = \sqrt[n]{\Sigma_{i=1}^{4}\left(\frac{|p_r^i - p_{r'}^i|}{\sqrt{\sigma(p^i)}}\right)^n}$$

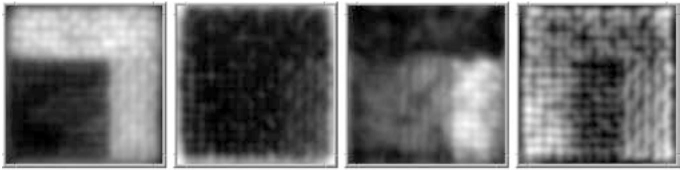
CLM's output

$x_{r\alpha} = h_r + \frac{1}{J_1}\Sigma_{r'} f_{rr'} x_{r'\alpha}$ or $x_{r\alpha} = 0$
Threshold: $1/2h_r$

Artificial Textures

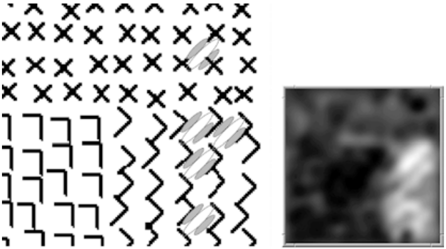


Artificial Textures



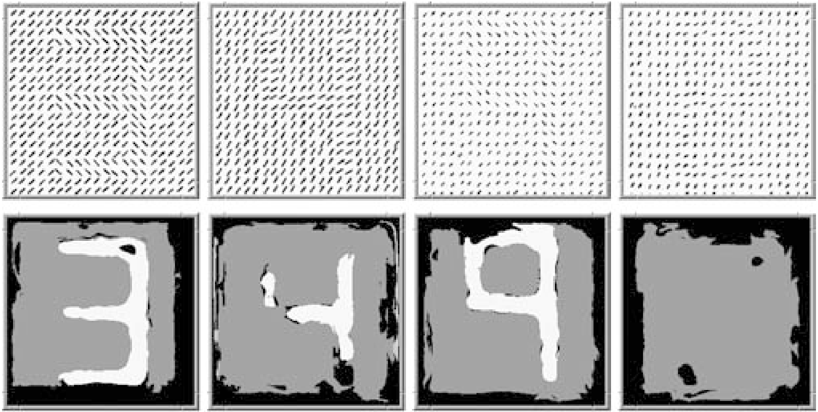
(a) 1st (b) 2nd (c) 3rd (d) 4th

Artificial Textures



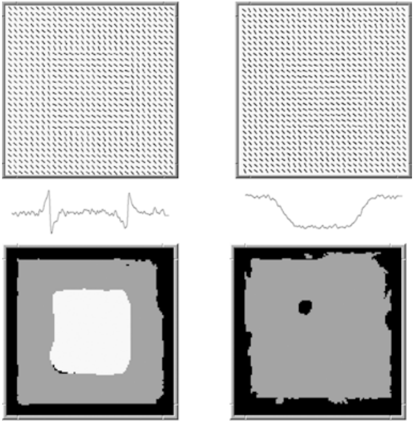
(a) (b)

Edge- and region-based phenomena



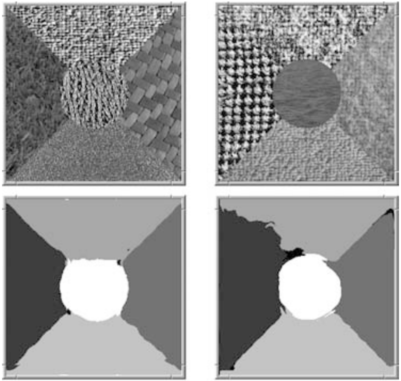
(a) (b) (c) (d)

Edge- and region-based phenomena



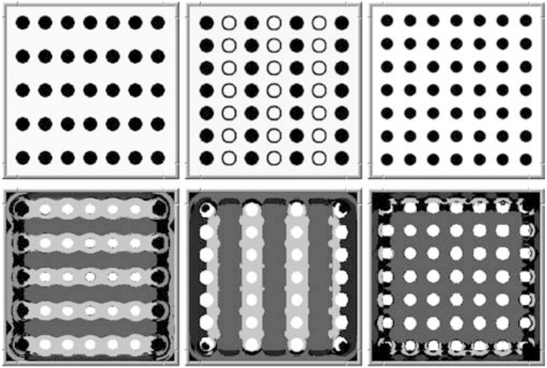
(a) (b)

Natural textures



(a) (b)

Gestalt laws



(a) (b) (c)

Conclusions

1. A feature extraction algorithm was developed to produce meaningful texture descriptor.
2. CLM Model reproduces edge-based as well as region-based phenomena.
3. The incorporation of Gestalt laws offers an interesting perspective to describe saliency groups.
4. Parameters tuning process was not necessary.
5. Model can mimic a large variety of human perception phenomena.

References

1. Jorg Ontrup and Heiko Wersing and Helge J. Ritter. A computational feature binding model of human texture perception. Cognitive Processing. pp. 31-44. 2004.