A Comparison of CNN and LEGION Networks

DeLiang Wang

The Ohio State University

Outline of presentation

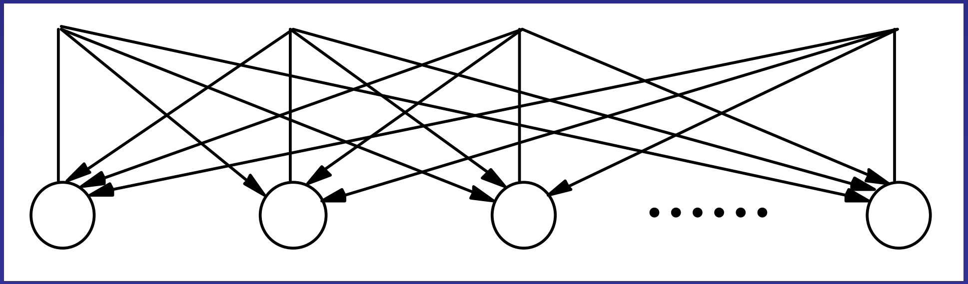
- **Motivation of the comparison**
- **Definitions**
- **Example applications**
  - Early visual processing
  - Connectedness detection
  - Image segmentation
- **Comparisons**
  - Unit dynamics
  - Connectivity
  - Visual processing
  - VLSI implementation
- **Discussion**

Why the comparison?

- **CNN (Chua & Yang, 1988) and LEGION (Terman & Wang, 1995) have been studied extensively in recent years**
- **They share a number of common properties**
  - Continuous-time dynamics
  - Nonlinearity
  - Emphasis on local connectivity
- **Yet, they were developed with different motivations and along different paths**
- **Are there deeper connections between CNN and LEGION?**
  - What are fundamental similarities and distinctions?
  - What are their relative strengths and limitations?
  - Is there synergy between them so that combining them can solve larger information-processing tasks?

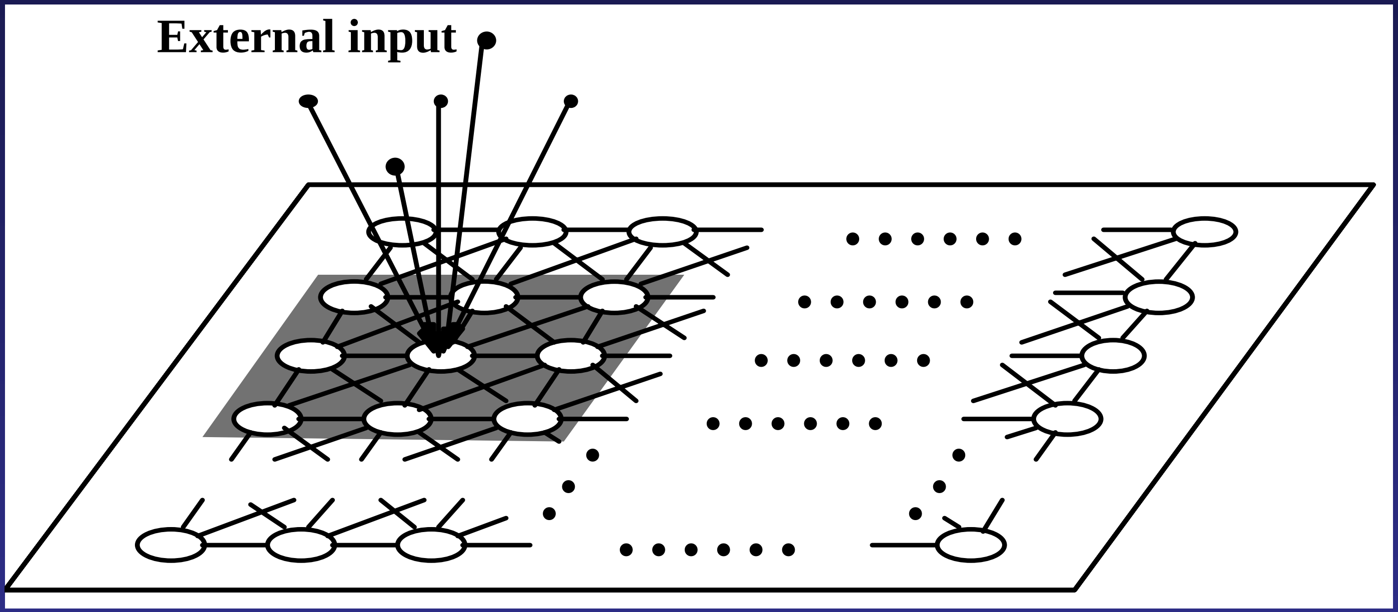
CNN motivation

- **Originally motivated by capabilities of the Hopfield network, Chua and Yang proposed CNN (Cellular Neural Network, later also called Cellular Nonlinear Network) in 1988**
  - To circumvent the full connectivity requirement of the Hopfield net ("curse of interconnecting wires") in order to facilitate VLSI implementation
  - CNN allows only local connectivity, hence forming cellular structure in the sense of cellular automata



Fully connected Hopfield net

CNN architecture



2D architecture with local input and local recurrent connectivity

CNN definition

- **A cell C_{ij} is defined by the following equations**

$$\dot{x}_{ij} = -x_{ij} + I_{ij} + \sum_{kl \in N(i,j)} A_{kl} y_{kl} + \sum_{kl \in N(i,j)} B_{kl} u_{kl}$$

  - $N(i,j)$ is the neighborhood of C_{ij} and I_{ij} is a threshold

- **The output of C_{ij} is given by**

$$y_{ij} = \frac{1}{2}(|x_{ij}+1| - |x_{ij}-1|) = \begin{cases} 1, & x_{ij} \geq 1 \\ x_{ij}, & 1 \geq x_{ij} \geq -1 \\ -1, & x_{ij} \leq -1 \end{cases}$$

  - This piecewise linear activation function is the origin of nonlinearity

CNN properties

- **A CNN network reaches a steady equilibrium under certain conditions, in particular when the *A* template (feedback template) is symmetric with respect to its center**
  - The symmetry of connectivity is akin to the weight symmetry requirement in the original Hopfield net
- **If the center element of *A* is greater than 1, the output of each CNN cell will be bipolar (1 or -1), after the network reaches equilibrium**

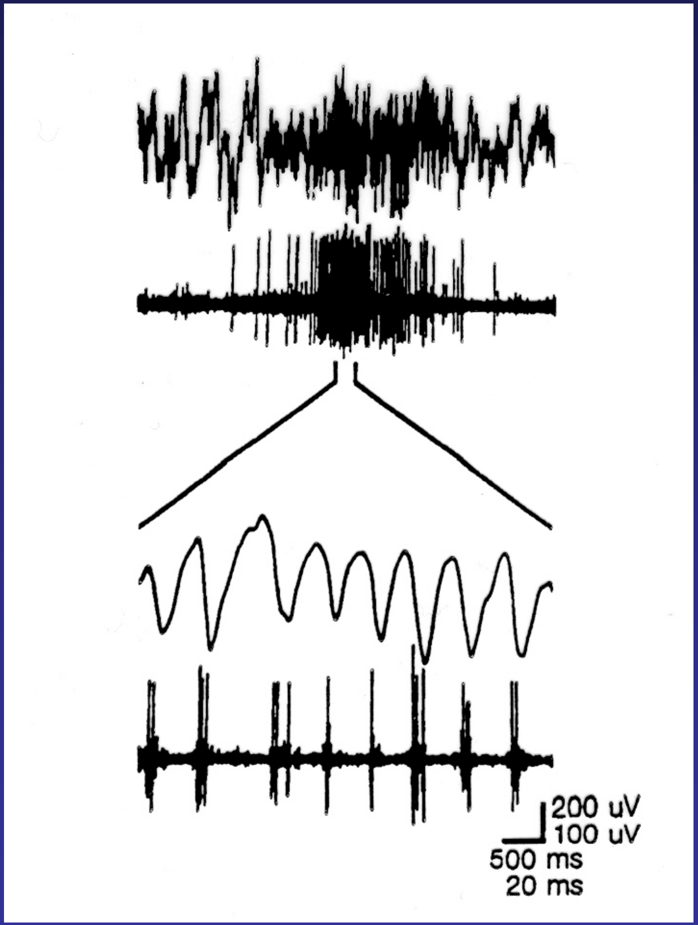
LEGION background: Scene analysis problem

Temporal correlation theory

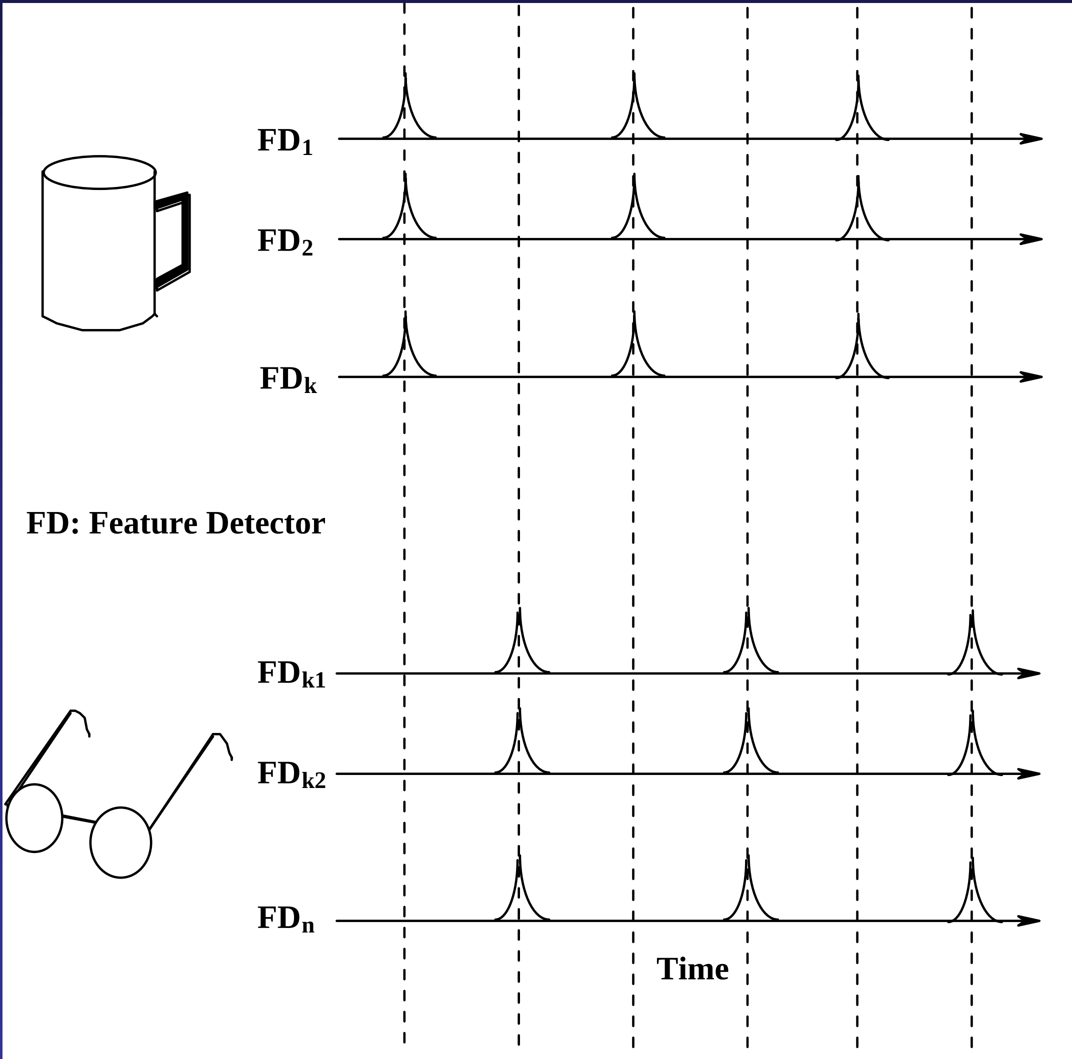
- **Feature binding is a fundamental problem**
  - In neuroscience
  - In perception
- **Temporal correlation as a representation (von der Malsburg, 1981; see also Milner, 1974; Abeles, 1982)**
  - An extra dimension
  - A plausible mechanism

Neurophysiological evidence

- Gray & Singer (1989)

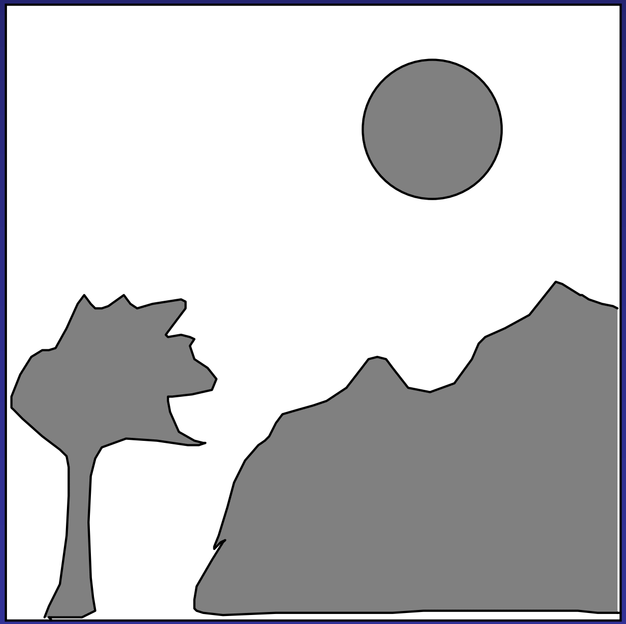


Oscillatory correlation theory



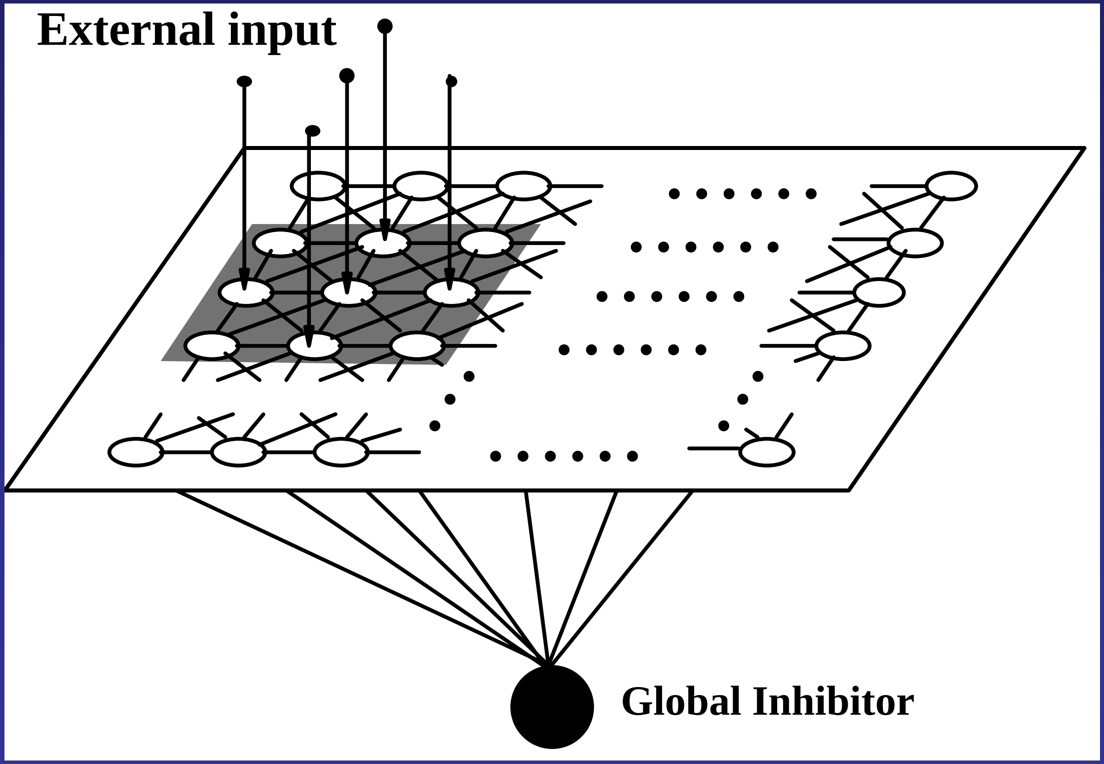
Computational requirements for oscillatory correlation

- **Need to synchronize locally coupled oscillator assembly**
  - Extensive literature in theoretical physics and mathematics on globally coupled oscillator populations
- **Need to desynchronize different assemblies, when facing multiple objects**
- **The above functions must be achieved very rapidly**

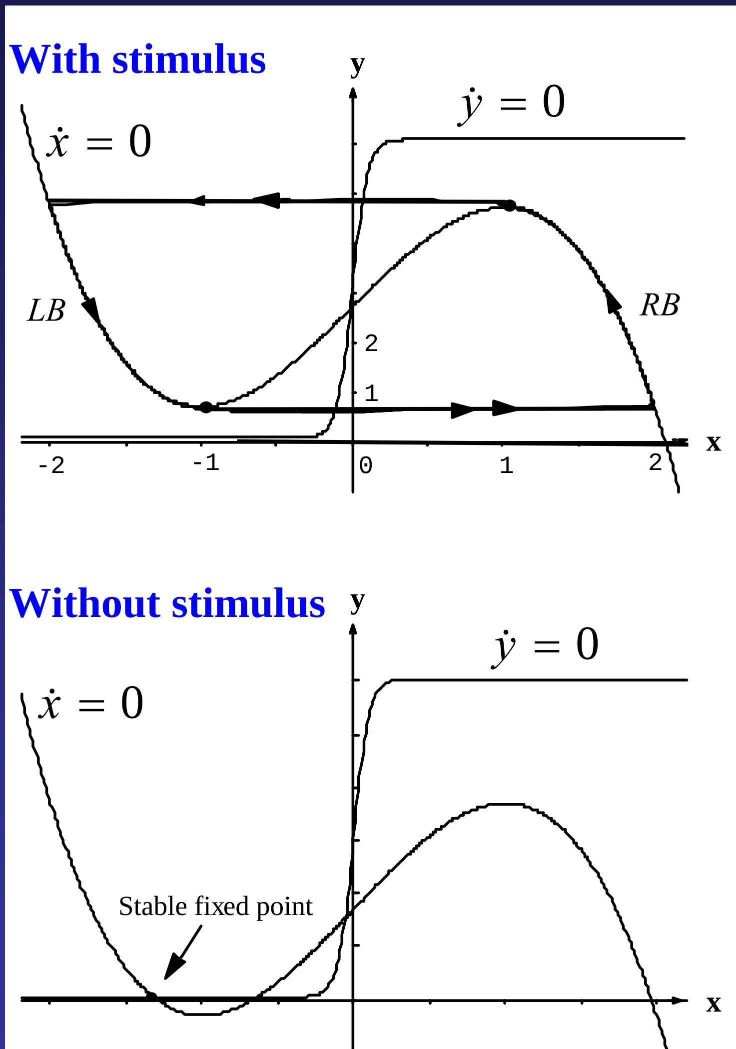


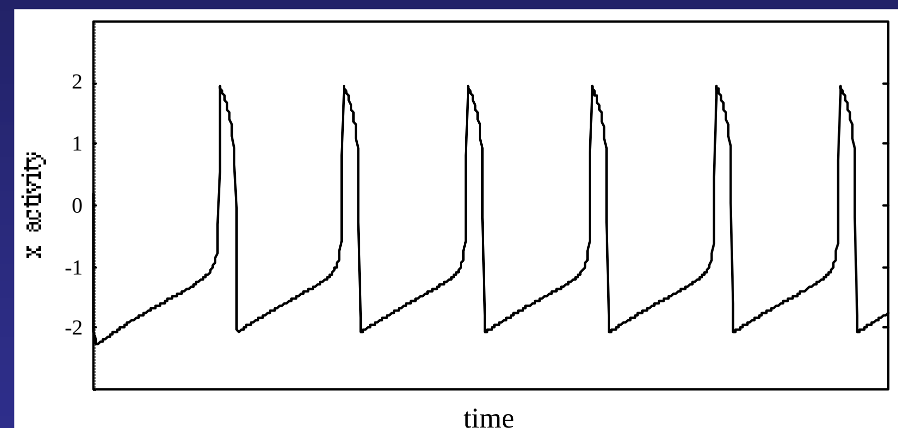
LEGION architecture

- **To provide a mechanism for oscillatory correlation, Terman and Wang proposed LEGION (Locally Excitatory Globally Inhibitory Oscillator Network) in 1995**



Single relaxation oscillator





Typical *x* trace (membrane potential)

LEGION definition

- **Basic oscillator definition**

$$\dot{x}_{ij} = 3x_{ij} - x_{ij}^3 + 2 - y_{ij} + u_{ij} + S_{ij} + \rho$$

$$\dot{y}_{ij} = \varepsilon(\alpha(1 + \tanh(x_{ij} / \beta)) - y_{ij})$$

- **Coupling between oscillators and the global inhibitor *z***

$$S_{ij} = \sum_{kl \in N(i,j)} W_{kl} H(x_{kl}) - W_z H(z - \theta_z)$$

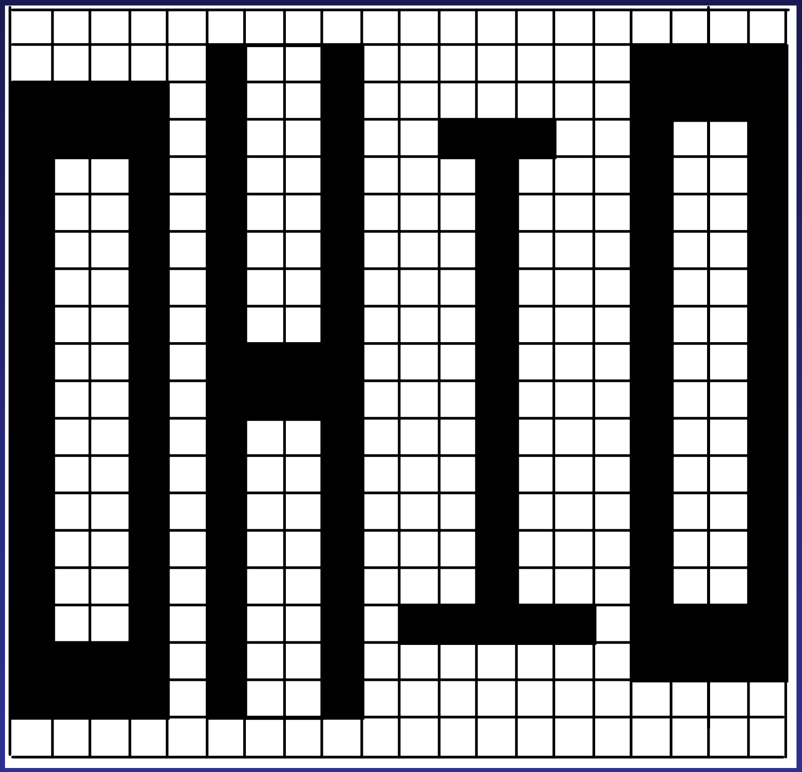
$$\dot{z} = \phi(\sigma_\infty - z)$$

  - H is the Heaviside step function, and σ_∞ is 1 if any oscillator is active and 0 otherwise

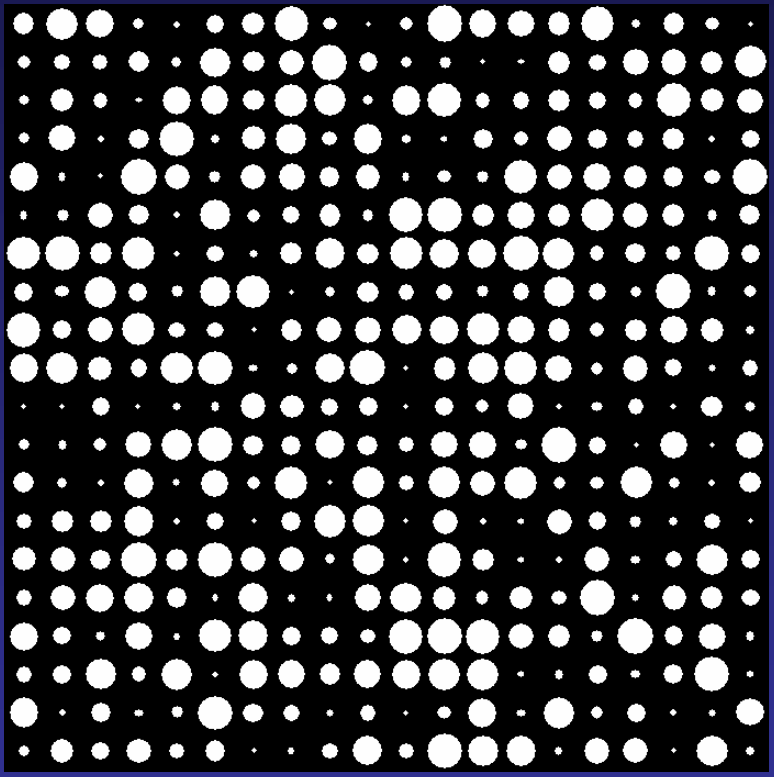
LEGION properties

- **Synchronization**. Under certain parameter conditions, all the oscillators in an assembly, corresponding to a connected pattern, synchronize.
- **Desynchronization**. After a certain time, oscillators belonging to different assemblies will never be in the active phase simultaneously
- **Speed**. The time LEGION takes to reach both synchrony within each assembly and desynchrony between different assemblies is no greater than N cycles, where N is the number of patterns.

LEGION example: Demo



Input image

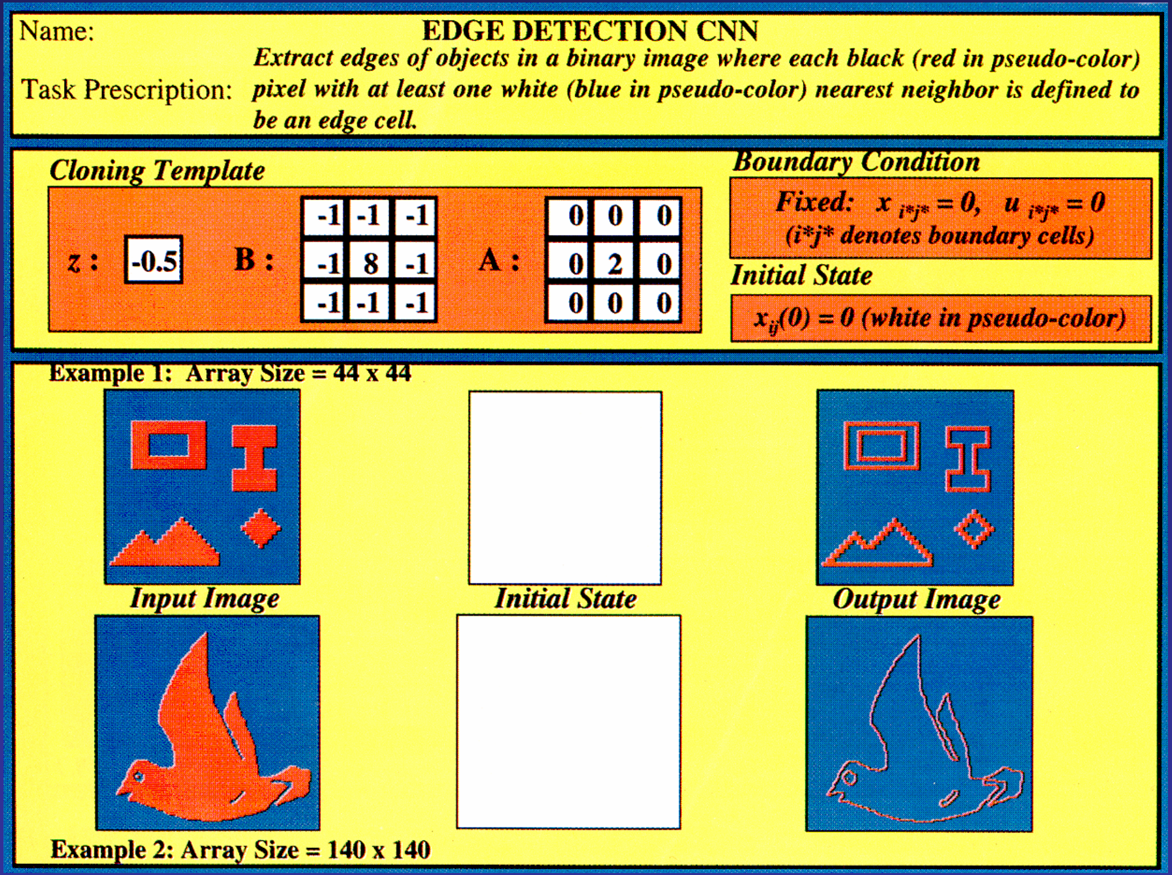


Outline of presentation

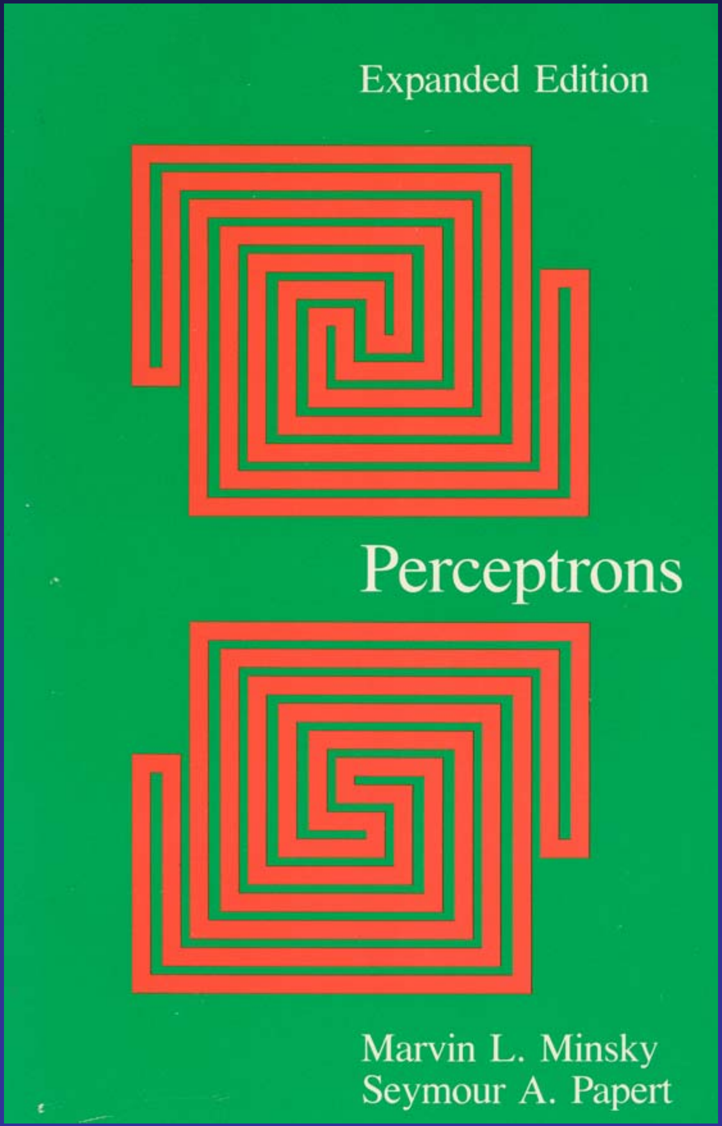
- **Motivation of the comparison**
- **Definitions**
- **Example applications**
  - Early visual processing
  - Connectedness detection
  - Image segmentation
- **Comparisons**
  - Unit dynamics
  - Connectivity
  - Visual processing
  - VLSI implementation
- **Discussion**

Example application: Early visual processing

- **CNN has been used to perform many local visual processing tasks, such as edge detection and image translation**



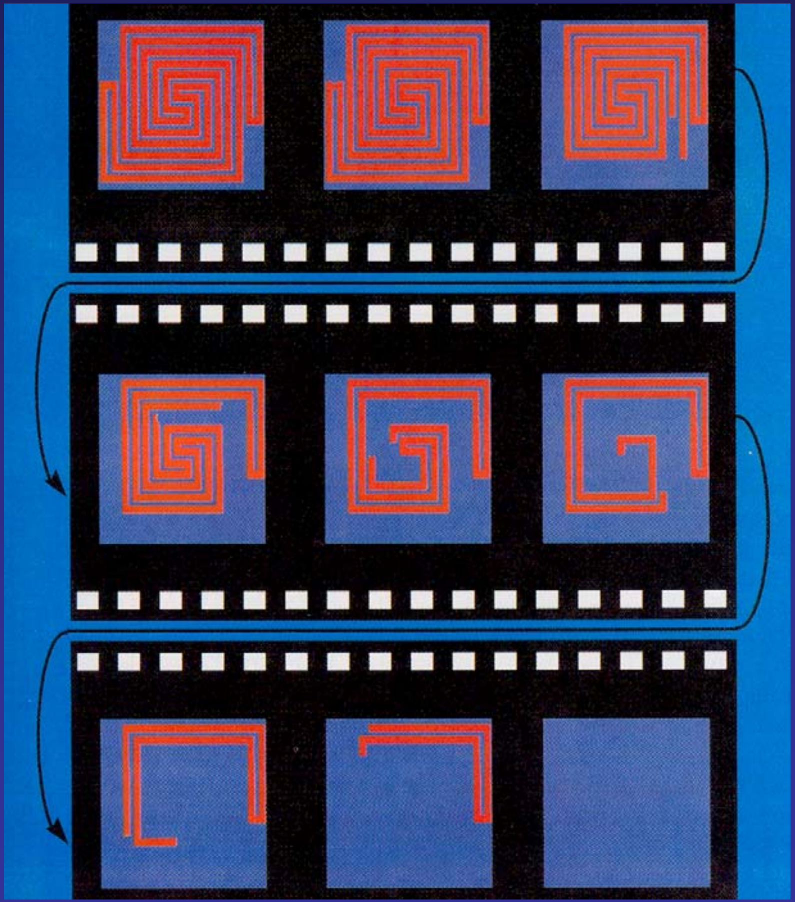
Example application: Connectedness detection

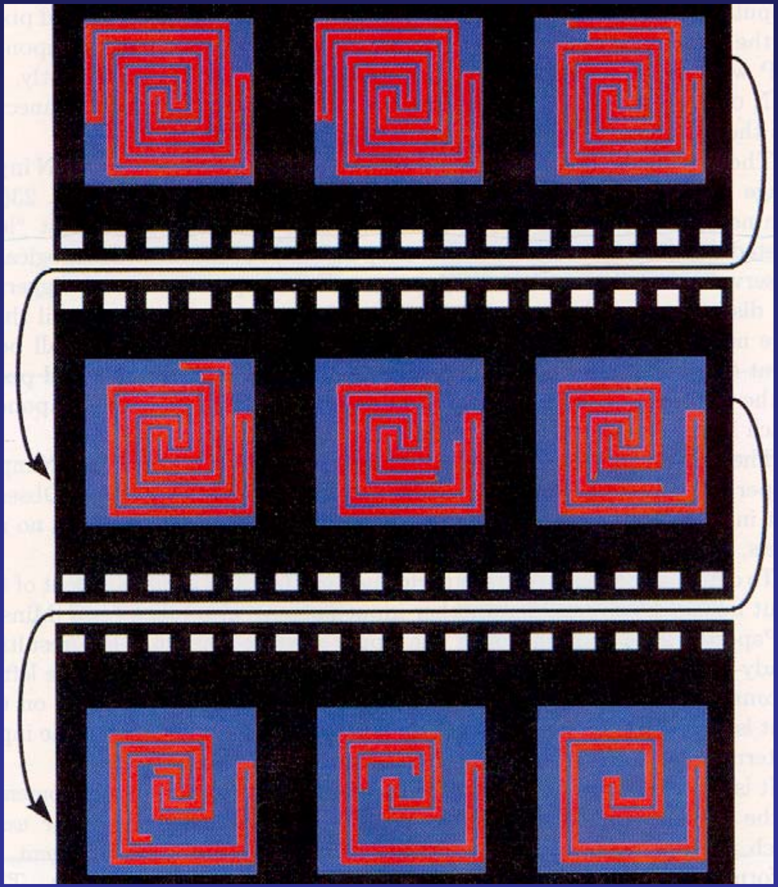


- **Minsky-Papert connectedness problem is a long-standing problem in perceptron learning**
- **The problem exposes the limitations of supervised learning, and illustrates the importance of proper representations**

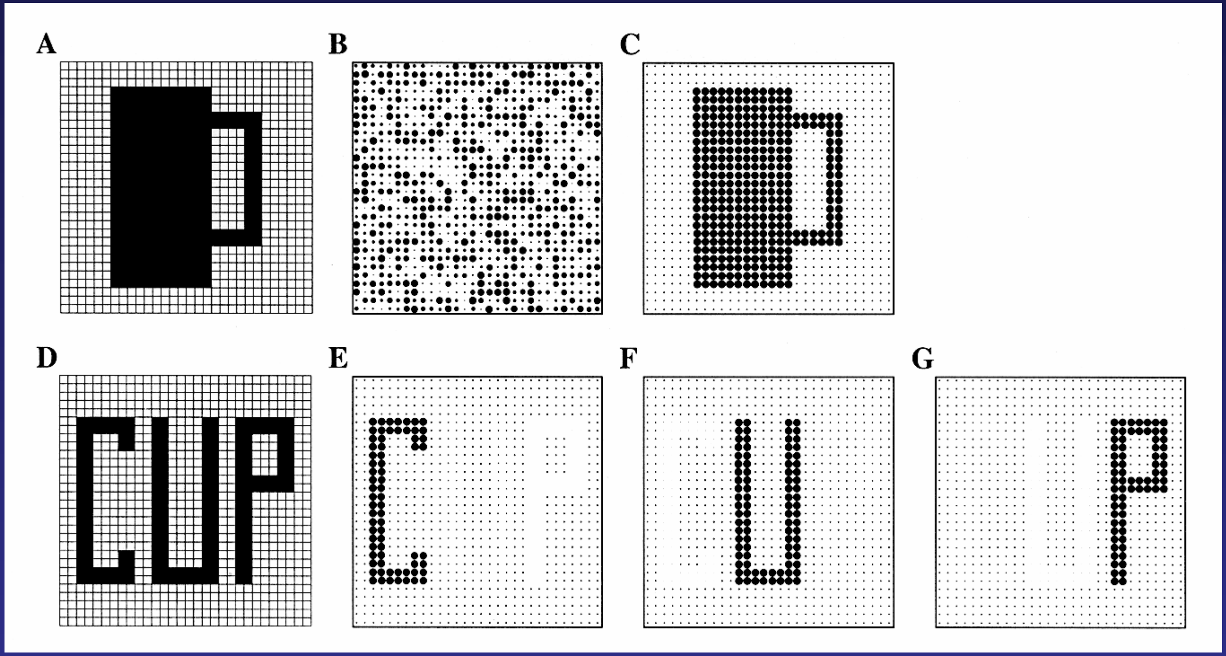
The connectedness problem: CNN solution

- **Basic idea: An activated cell deactivates itself and its neighbors, and this process propagates**

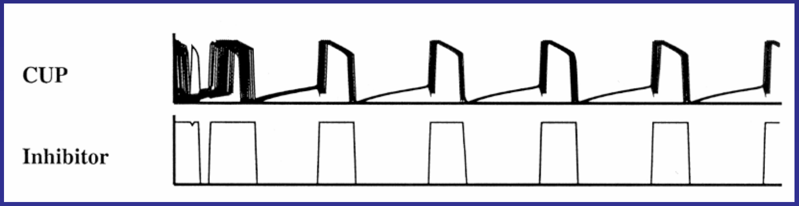


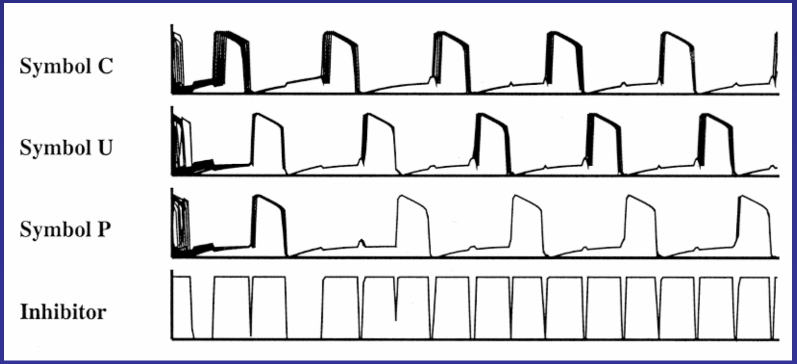


The connectedness problem: LEGION solution



Basic idea: Synchronization within a connected pattern and desynchronization between different ones





CNN image segmentation

- **It decomposes the segmentation problem into different stages performing image smoothing, edge detection, major feature extraction through morphological operations, contour extraction, and hollow filling, etc.**
- **The sequence of the stages results in a collection of closed contours**

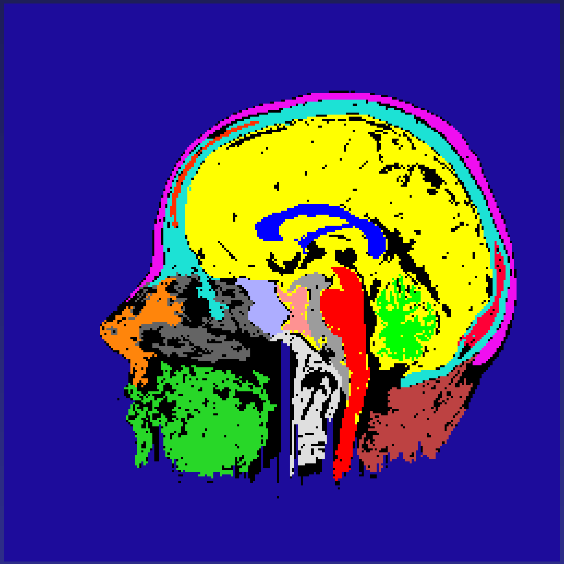
Stoffels et al., 1997

LEGION image segmentation

- **Feature extraction first takes place**
  - An image feature can be pixel intensity, depth, local image patch, texture element, optic flow, etc.
- **Connection weights between two neighboring oscillators are set to be proportional to feature similarity**
- **Global inhibitor controls granularity of segmentation**
  - Larger inhibition results in more and smaller regions
- **Segments pop out from LEGION in time**
- **A main motivation behind LEGION**

LEGION segmentation example

Input image



Segmentation result

Outline of presentation

- **Motivation of the comparison**
- **Definitions**
- **Example applications**
  - Early visual processing
  - Connectedness detection
  - Image segmentation
- **Comparisons**
  - Unit dynamics
  - Connectivity
  - Visual processing
  - VLSI implementation
- **Discussion**

Comparisons: Fundamentals

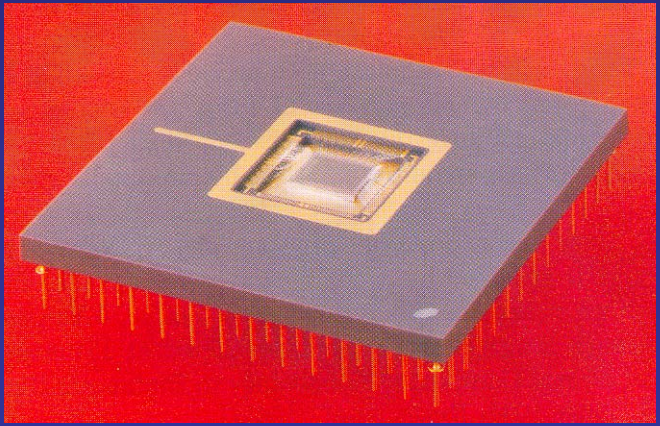
- **Unit dynamics**
  - CNN: Equilibrium dynamics. Hence time (phase) is not an intrinsic dimension in CNN
  - LEGION: Oscillatory dynamics. It can represent both amplitude and phase
  - Implications: Representation of activity versus feature binding
- **Connectivity**
  - Both possess local recurrent connectivity
  - LEGION has, in addition, a single global unit

Comparisons: Visual processing

- **CNN: Well suited for local operations on an image**
  - Corresponding to early visual processing
  - Though capable of decomposing an segmentation task to implementable stages, such decomposition would require human intervention (i.e. high-level programming)
- **LEGION: Well suited for midlevel visual processing, corresponding to perceptual organization and scene segmentation**
  - It needs a separate feature extraction stage, which belongs to early visual processing, in order to perform midlevel processing
- **For visual processing, early and midlevel as well as high-level processing (e.g. recognition) all likely take place**
  - This suggests natural synergy between CNN and LEGION mechanisms

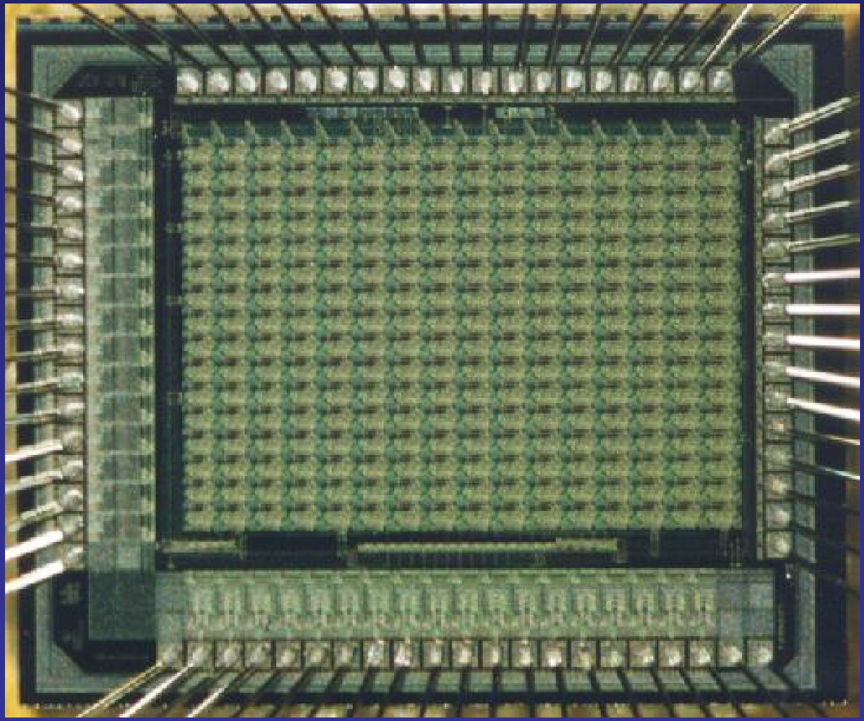
Comparisons: VLSI implementation

- **Direct circuit implementation has advantages of higher processing speed, lower power consumption, and smaller (silicon) area**
- **Circuit implementation is the principal motivation of CNN, and many successful attempts have been made**
  - Most effort is on image processing involving local operations
  - Some recent studies also attempt to perform image segmentation, but need a hybrid of analog and digital circuits



Comparisons: VLSI implementation

- **Effort has also been made in VLSI implementation of LEGION, to a less extent compared to CNN**
- **A 16x16 LEGION chip by Cosp et al. (2003)**



Discussion

- **According to Marr (1982), a complex information processing system must be understood in three levels**
  - Computational theory: Goal, its appropriateness, and basic processing strategy
  - Representation and algorithm: Representations of input and output and transformation algorithms
  - Implementation: Physical realization
- **Neural network research tends to blend the boundary between the algorithmic and implementation levels**
  - Research on LEGION exemplifies this situation
- **However, Marr’s three levels remain a key guide for understanding and accomplishing complex information-processing tasks, such as visual processing**

Discussion – cont.

- **Fundamentally, CNN is an implementation theory. Although LEGION is motivated by representational and algorithmic considerations it, too, is an implementation theory**
  - It is an important level of analysis, but no substitute for computational-theory and algorithmic analyses
- **Chua (1998) proposed the concept of local activity, and asserts that complexity cannot emerge unless the medium is locally active**
- **LEGION research poses the question: Is local activity, although necessary, adequate by itself for complex information processing?**
  - What about attentional processing?
  - How to explain global connectivity in the brain?

Conclusion

- **CNN and LEGION share continuous dynamics and local connectivity, yet they differ in the use of dynamics and the use of global connectivity**
- **CNN is good at early visual processing whereas LEGION is good at midlevel processing**
- **There is natural synergy between the two frameworks**
- **Both CNN and LEGION are fundamentally implementation theories in Marrian information-processing paradigm**