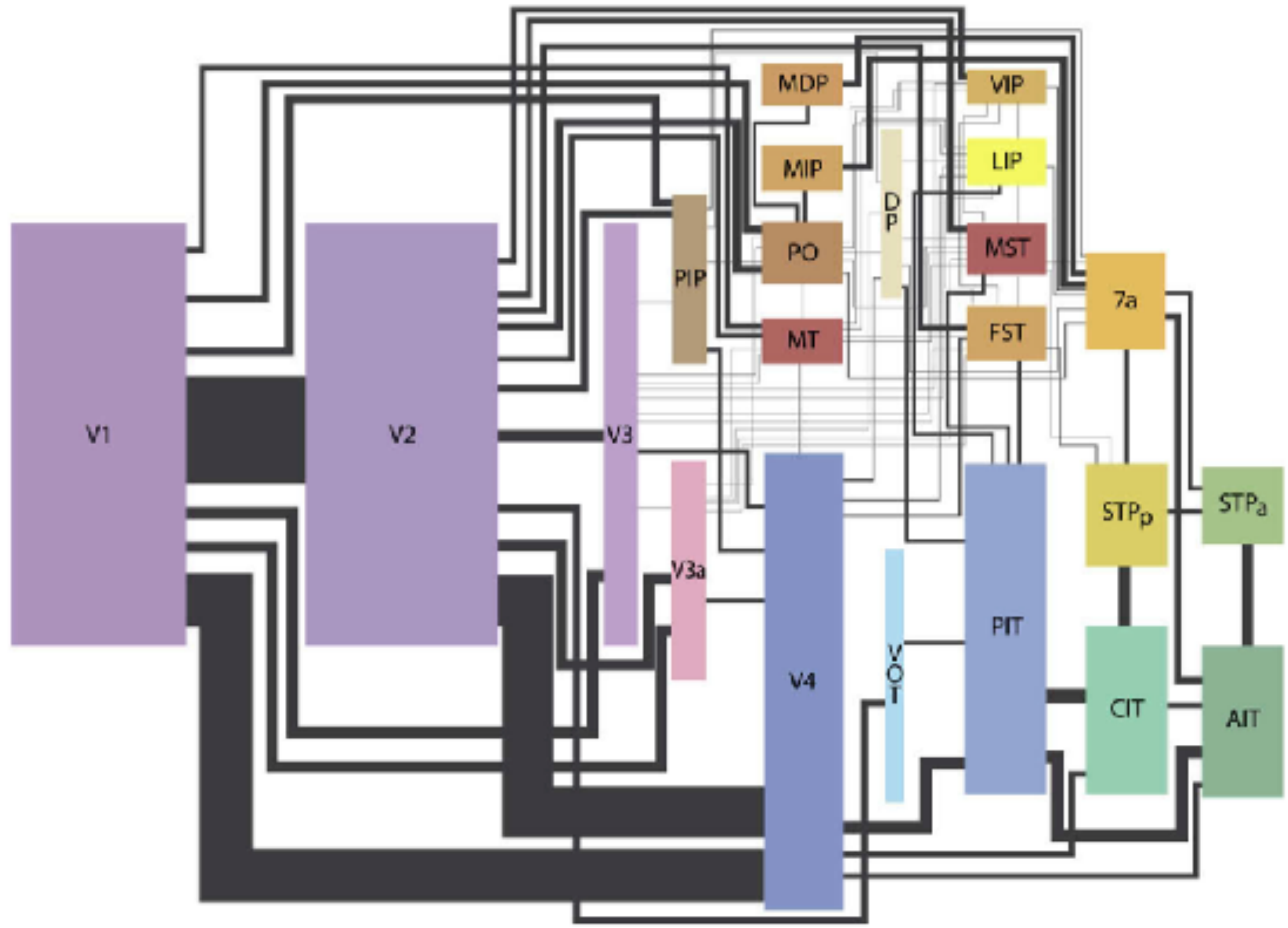
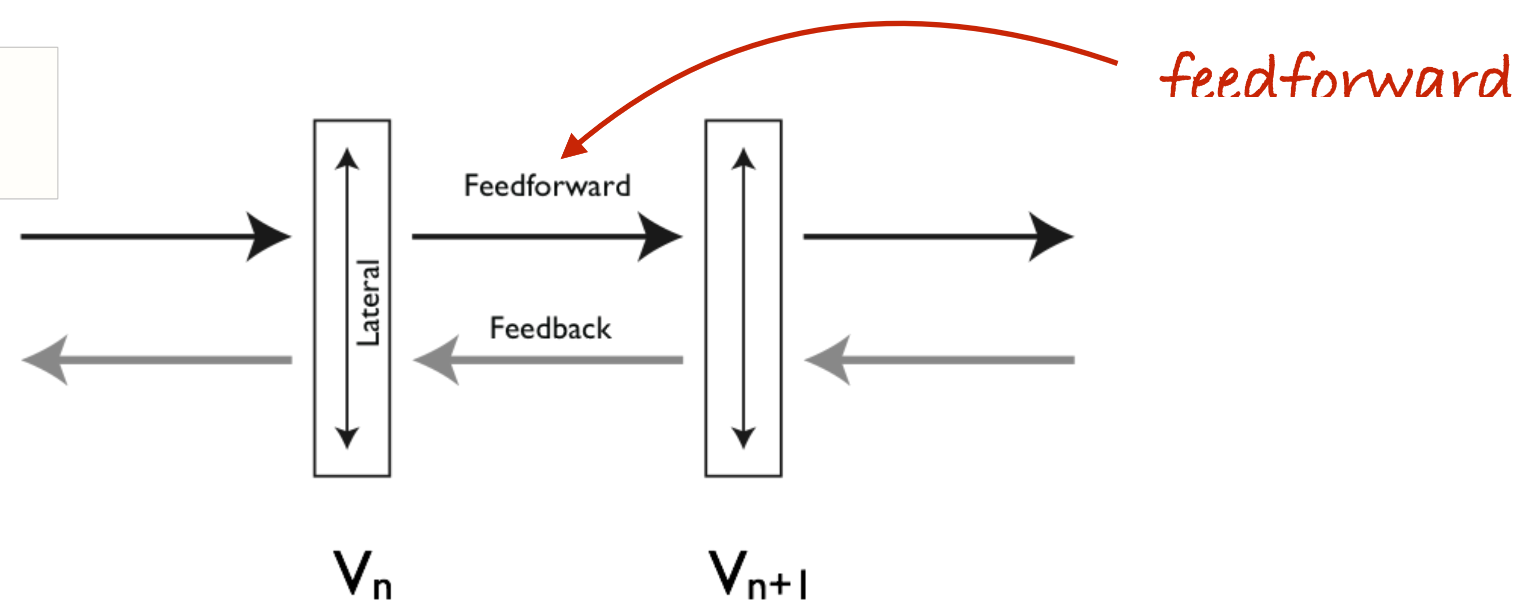
# Bidirectional processing I:

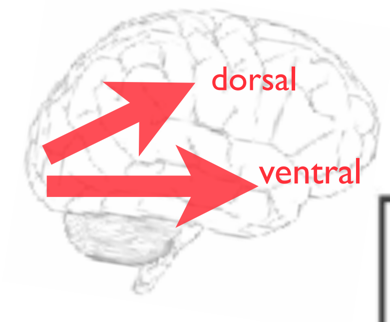
feedforward & feedback networks for recognition

Focus today on feedforward architectures from monkey

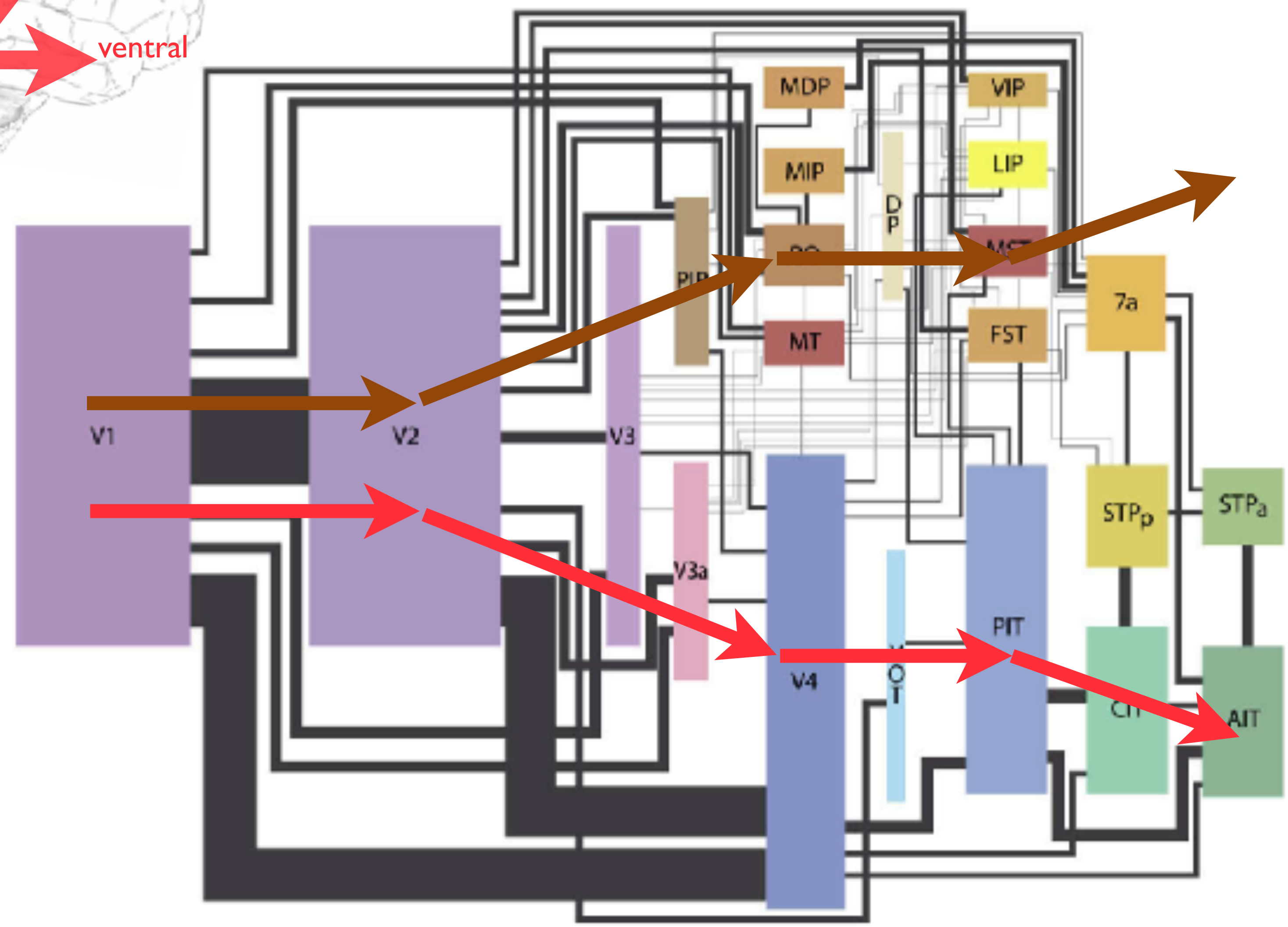



Wallisch, P., & Movshon, J. A. (2008). Structure and Function Come Unglued in the Visual Cortex. *Neuron*, *60*(2), 194–197.




from monkey




dorsal

*object-viewer properties*
*for spatial and motor actions*

ventral

*intrinsic object properties*
*for identification*

feedforward

# What determines the different selectivities for pathways and areas?

*image information required for different basic tasks*
*...but lots of tasks*

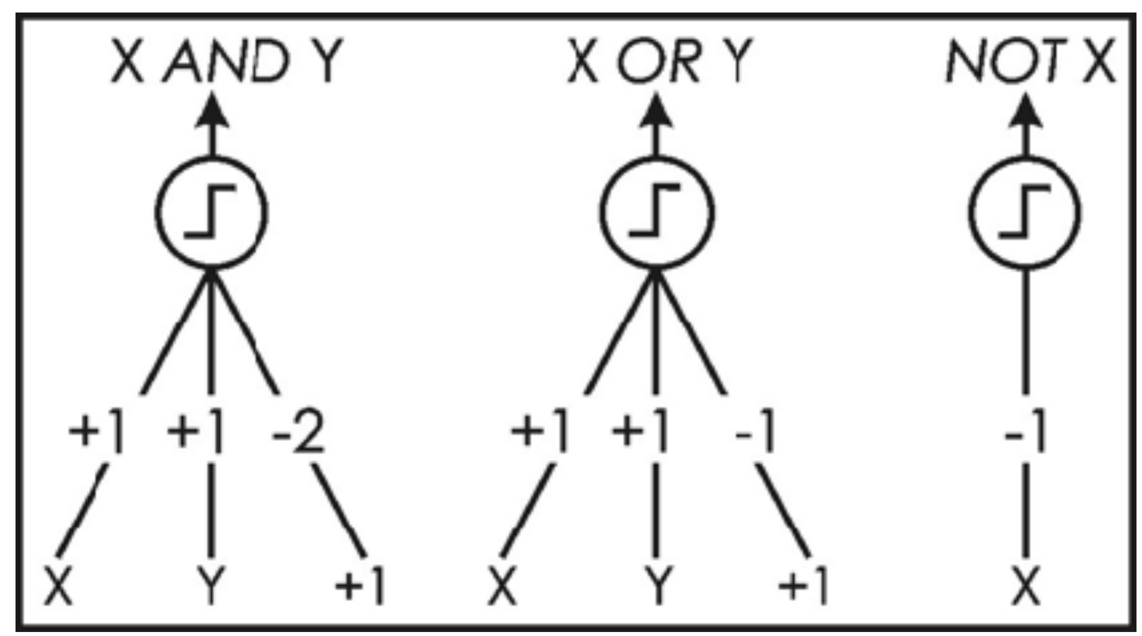
*focus on object categorization*

Wallisch, P., & Movshon, J. A. (2008). Structure and Function Come Unglued in the Visual Cortex. *Neuron*, *60*(2), 194–197.

# A brief history of computational pattern/object recognition

1940s

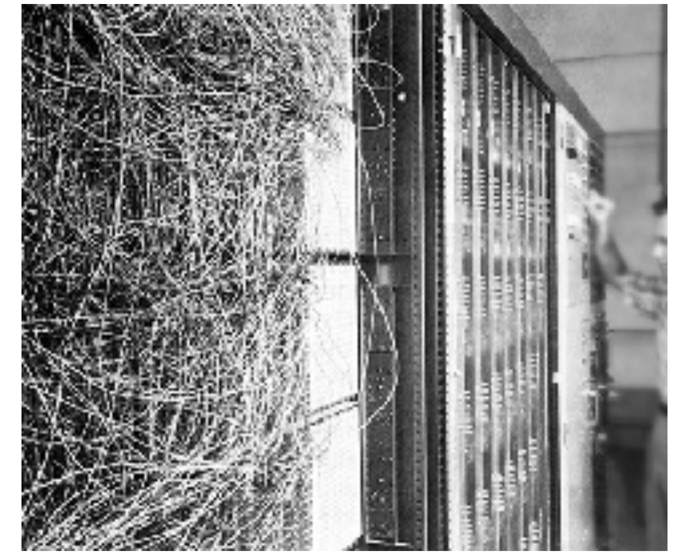
McCulloch and Pitts
threshold logic units

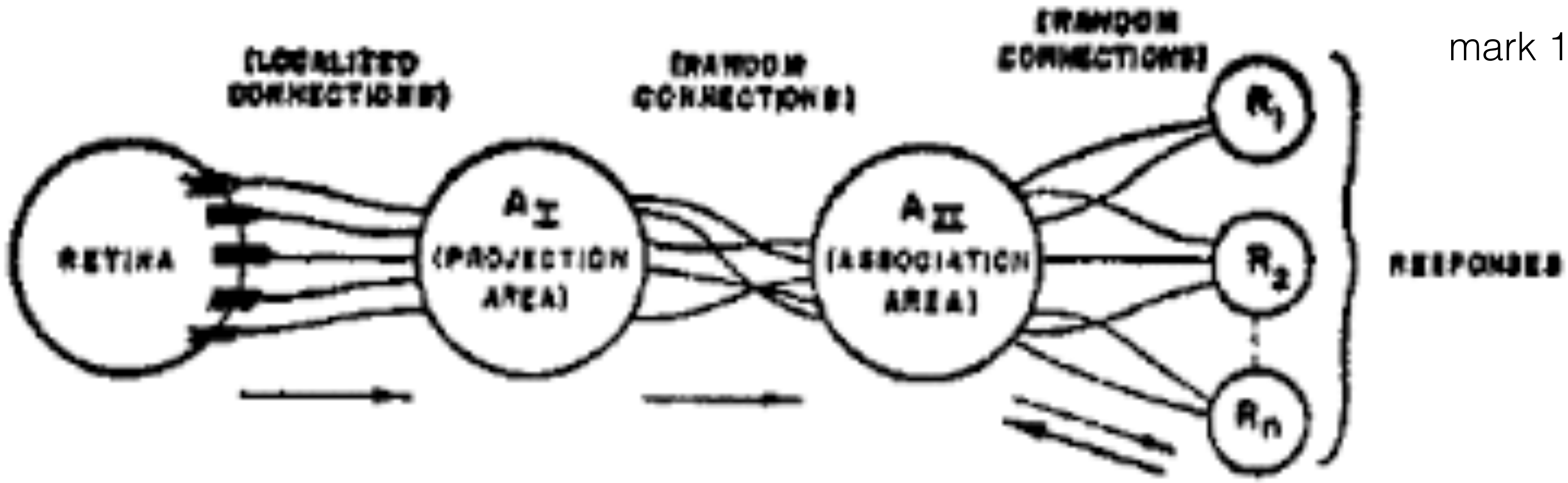



template models, e.g. SDT


# 1950s

# Rosenblatt's perceptron






mark 1

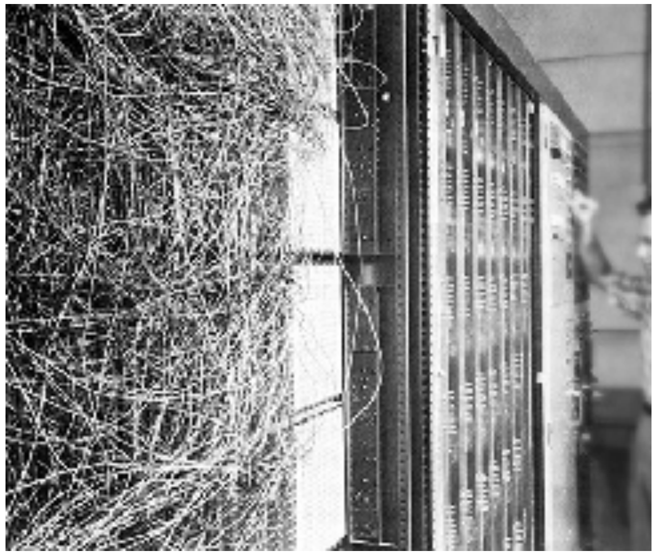
feedback too

random connections.
perceptron learning algorithm enabled it to
to learn to classify

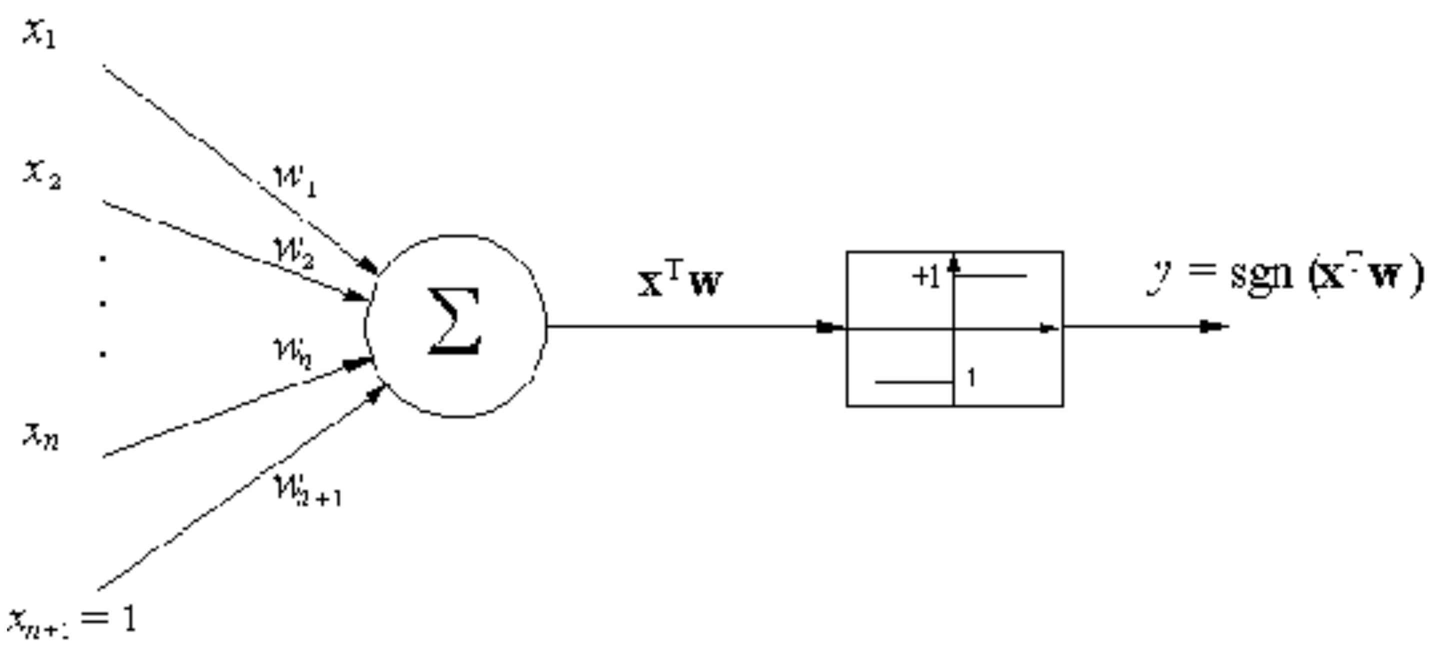
Rosenblatt, F. 'The Perceptron, a Perceiving and Recognizing Automaton', Cornell Aeronautical Laboratory Report No. 85-460-1 (1957);

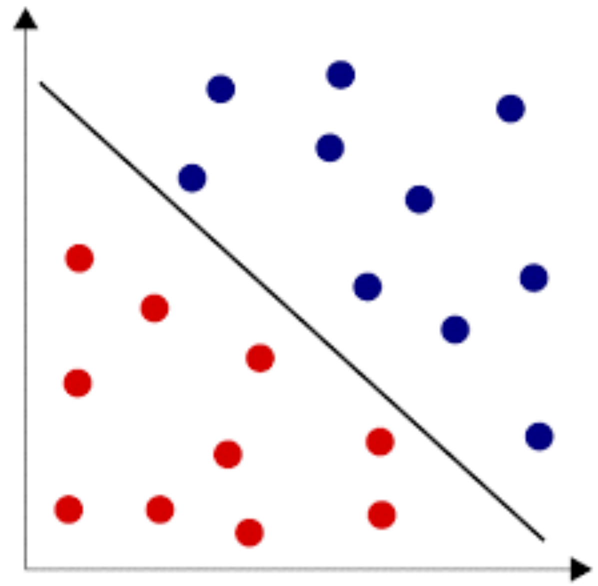
Rosenblatt, F. Principles of Neurodynamics (Washington, D.C.: Spartan, 1962).


# Threshold-logic and the perceptron learning rule



mark 1



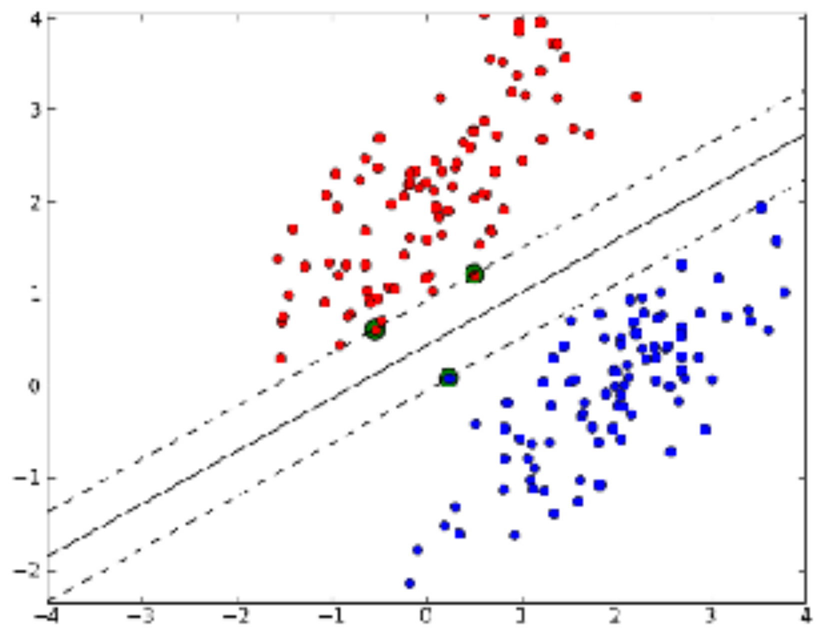



Adjust weights, w, to find separating line.
Limited to linearly separable classes

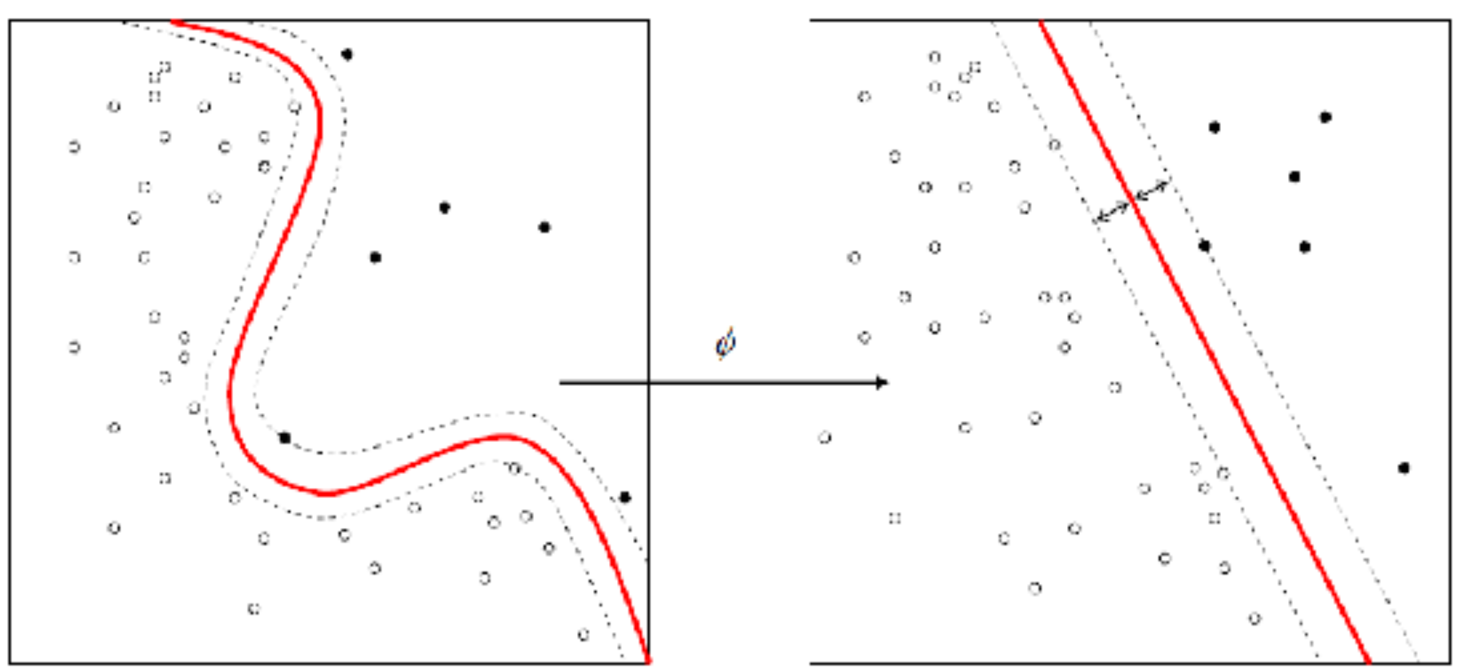
Rosenblatt, F. 'The Perceptron, a Perceiving and Recognizing Automaton', Cornell Aeronautical Laboratory Report No. 85-460-1 (1957);

Rosenblatt, F. Principles of Neurodynamics (Washington, D.C.: Spartan, 1962).

# support vector machines




1963 — linear
1992 — non-linear kernels




$$\hat{y} = \operatorname{sgn} \sum_{i=1}^{n} w_i y_i k(\mathbf{x}_i, \mathbf{x}')$$

https://en.wikipedia.org/wiki/Kernel_method#/media/File:Kernel_Machine.png

# 1980s through 1990s
# getting multi-layer perceptrons to work

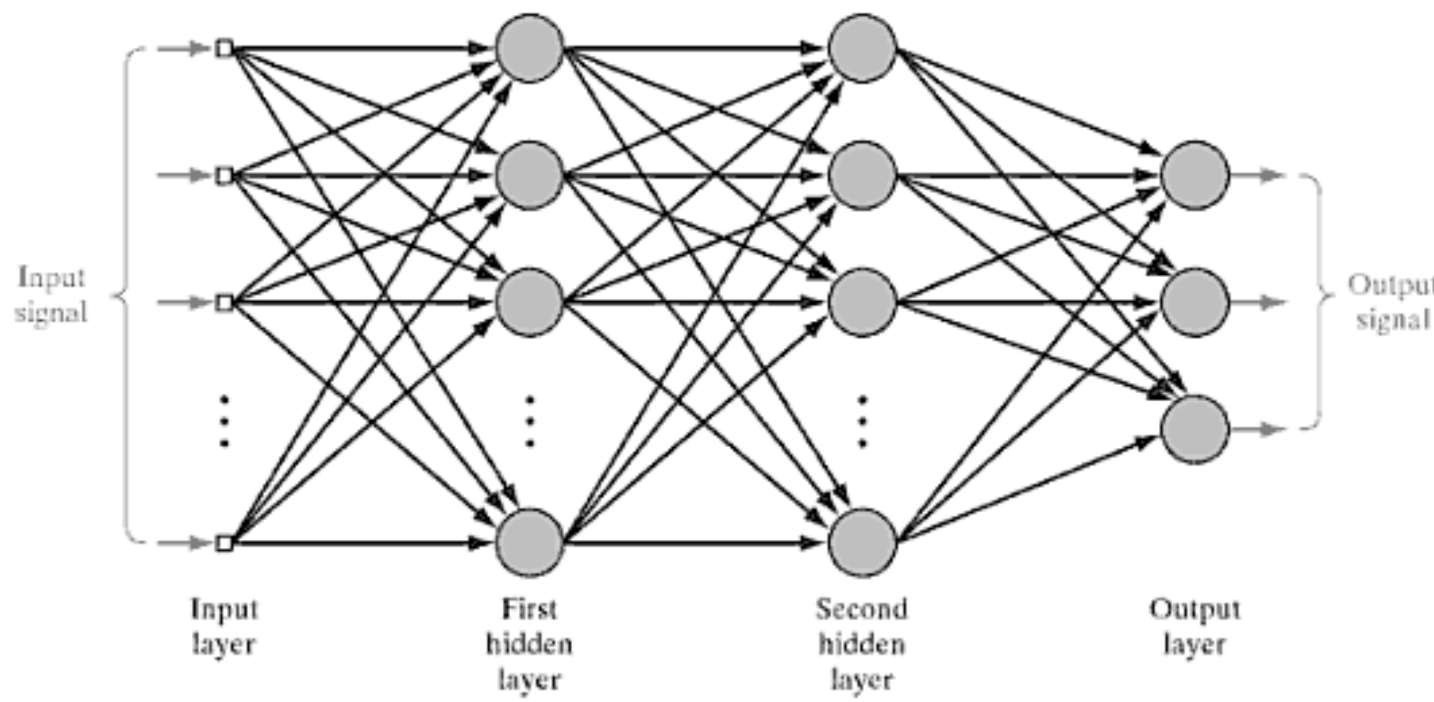



FIGURE 4.1 Architectural graph of a multilayer perceptron with two hidden layers.

solving the supervised learning problem:

error-back propagation for learning weights


Rumelhart, David E.; Hinton, Geoffrey E.; Williams, Ronald J. (8 October 1986). "Learning representations by back-propagating errors". Nature 323 (6088): 533–536

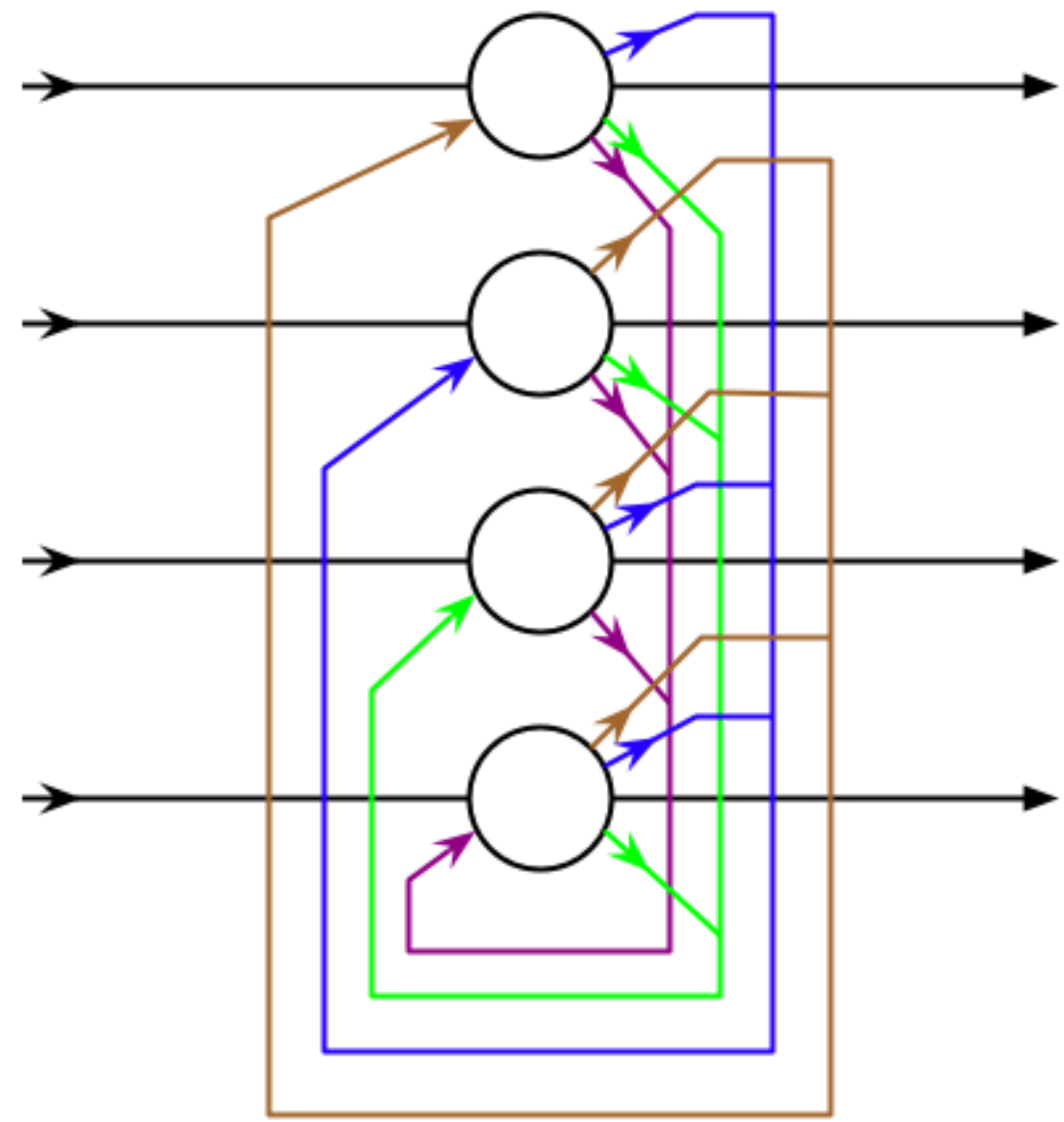
LeCun,Y,Bottou,L,Bengio,Y,andHaffner,P.Gradient-basedlearningappliedtodocumentrecognition. Proceedings of the IEEE, 86(11):2278–2324, November 1998.

*****

Paul J. Werbos. Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD thesis, Harvard University, 1974

Bryson, A.E.; W.F. Denham; S.E. Dreyfus. Optimal programming problems with inequality constraints. I: Necessary conditions for extremal solutions. AIAA J. 1, 11 (1963) 2544-2550

# 1980s



recurrent networks
Hopfield network
Boltzmann machines

- theoretical understanding of what networks were doing
- development of cost (energy) function methods for finding solutions and learning
- very slow convergence, did not scale up
- but no architectural constraints (e.g. hierarchical)

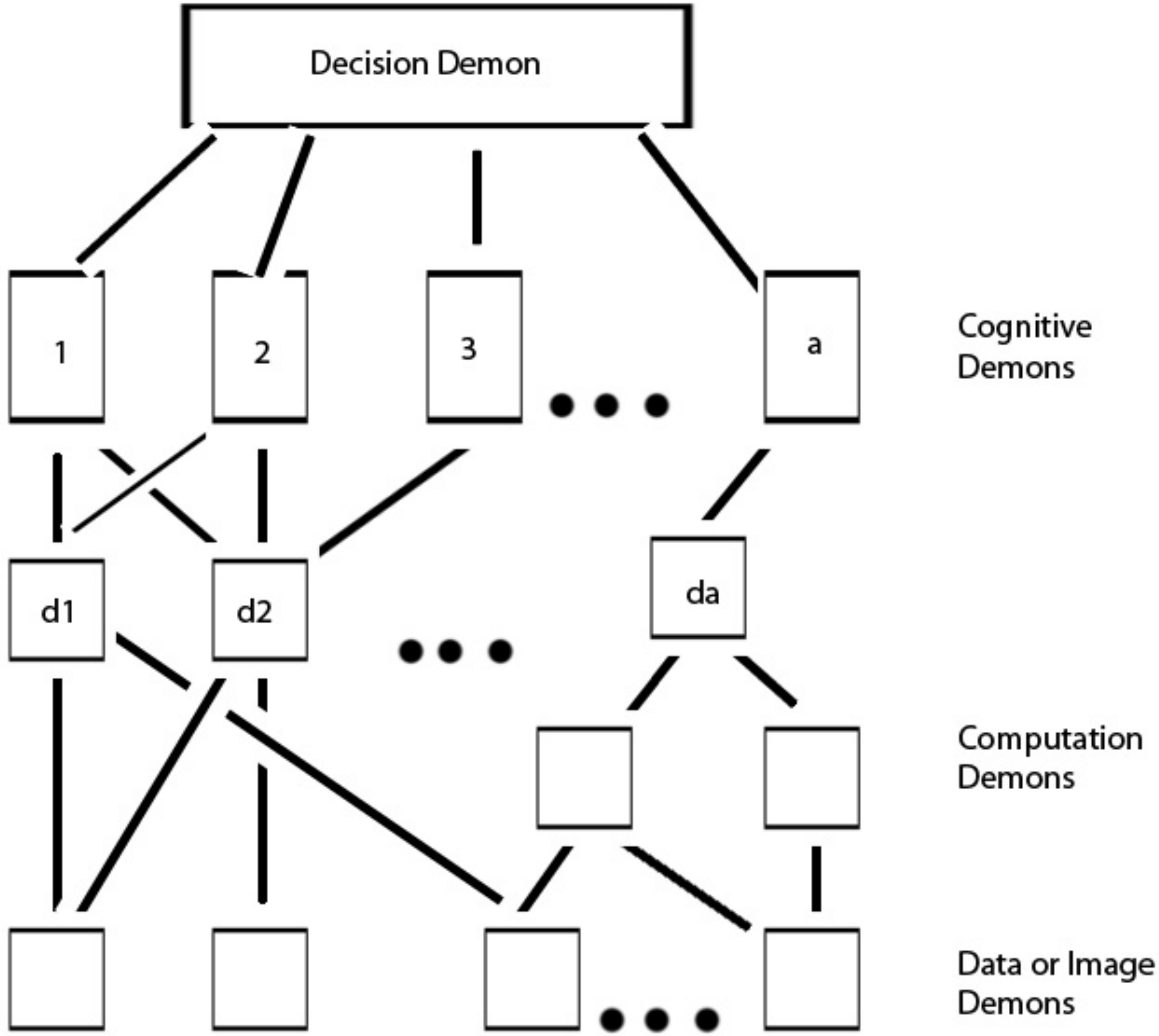
Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, *9*(1), 147–169.

The need for an “architecture” for vision

to manage local uncertainty

and the complexities of real-world images

# Pandemonium 1959




- parallel processing,
- learning
- hill-climbing cost functions

O. G. Selfridge. "Pandemonium: A paradigm for learning." In D. V. Blake and A. M. Uttley, editors, Proceedings of the Symposium on Mechanisation of Thought Processes, pages 511–529, London, 1959.

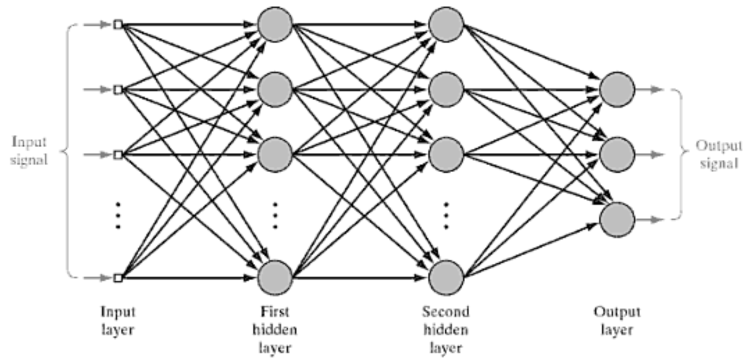
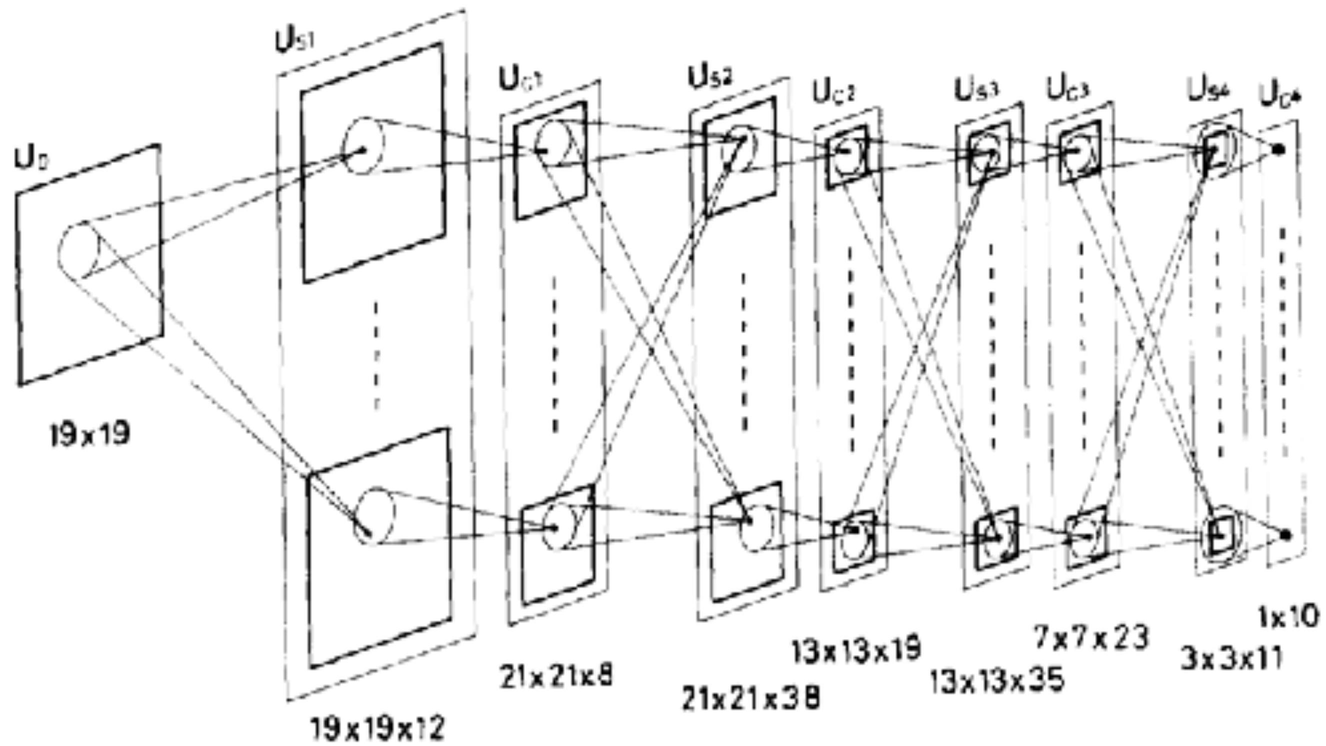
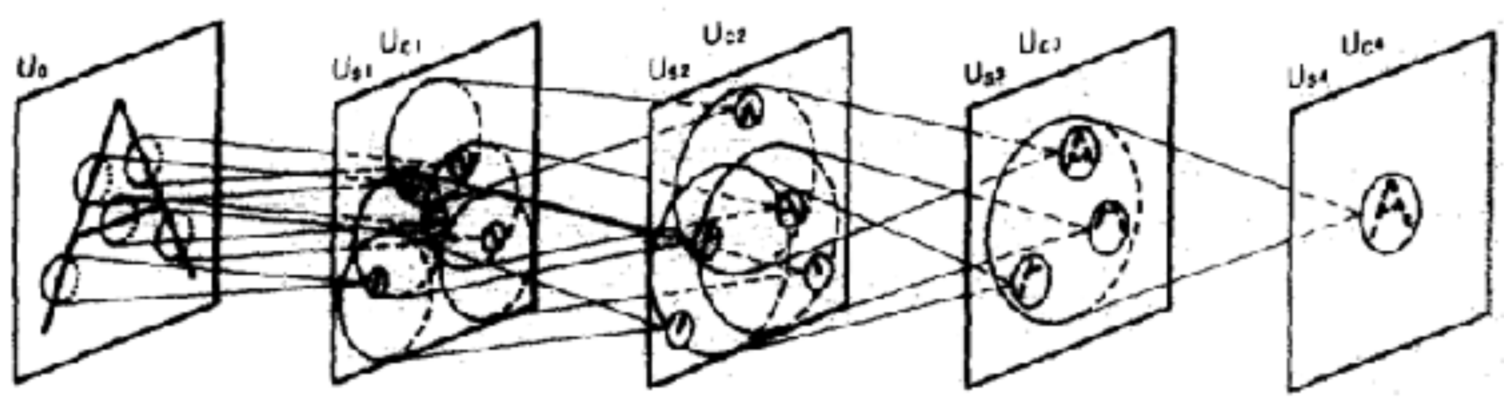



FIGURE 4.1 Architectural graph of a multilayer perceptron with two hidden layers.

# Fukushima 1988




Fukushima, K. (1988). Neocognitron - a Hierarchical Neural Network Capable of Visual-Pattern Recognition. *Neural Networks*, *1*(2), 119–130.

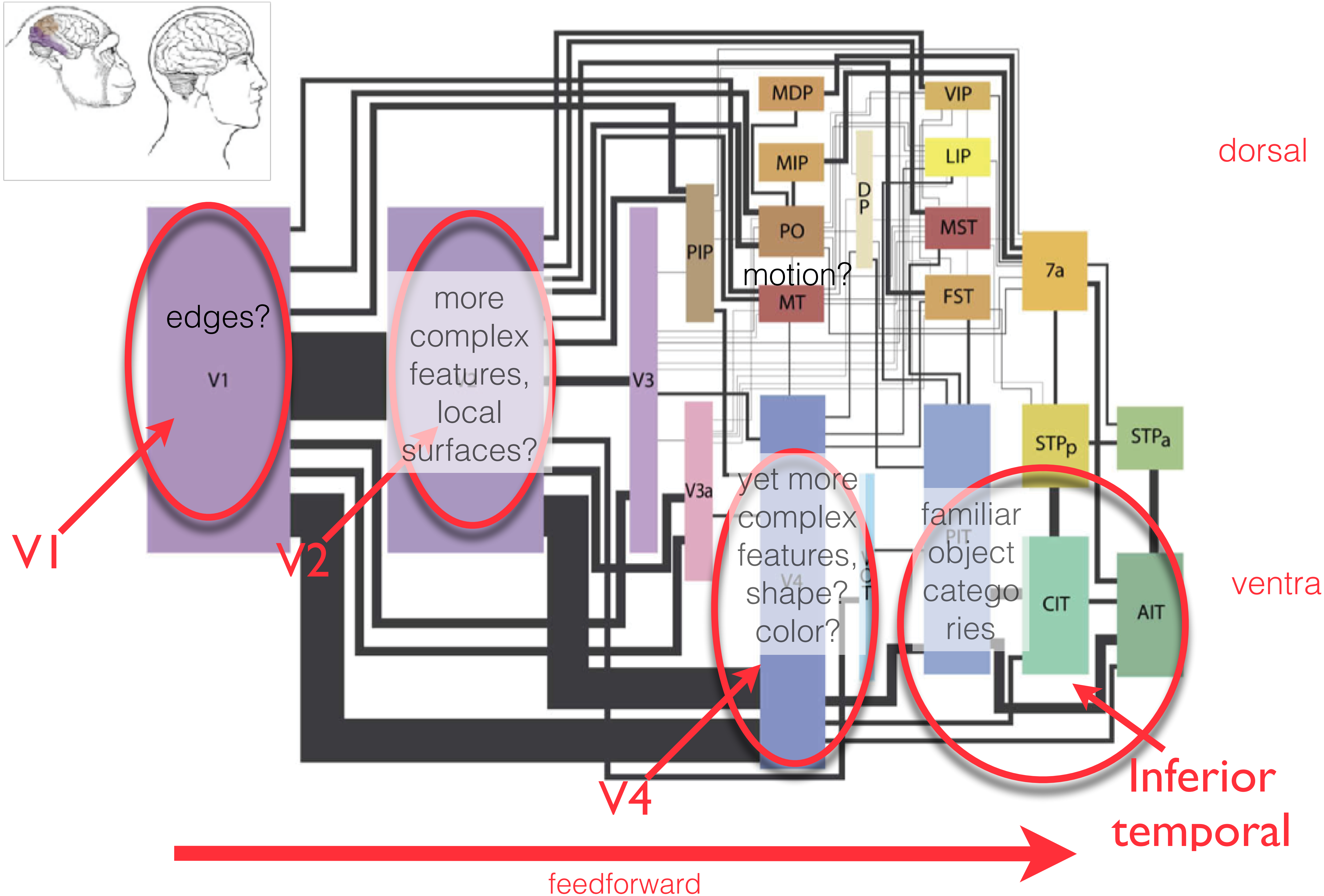
supervised and unsupervised learning

# primate visual hierarchical neuroarchitecture

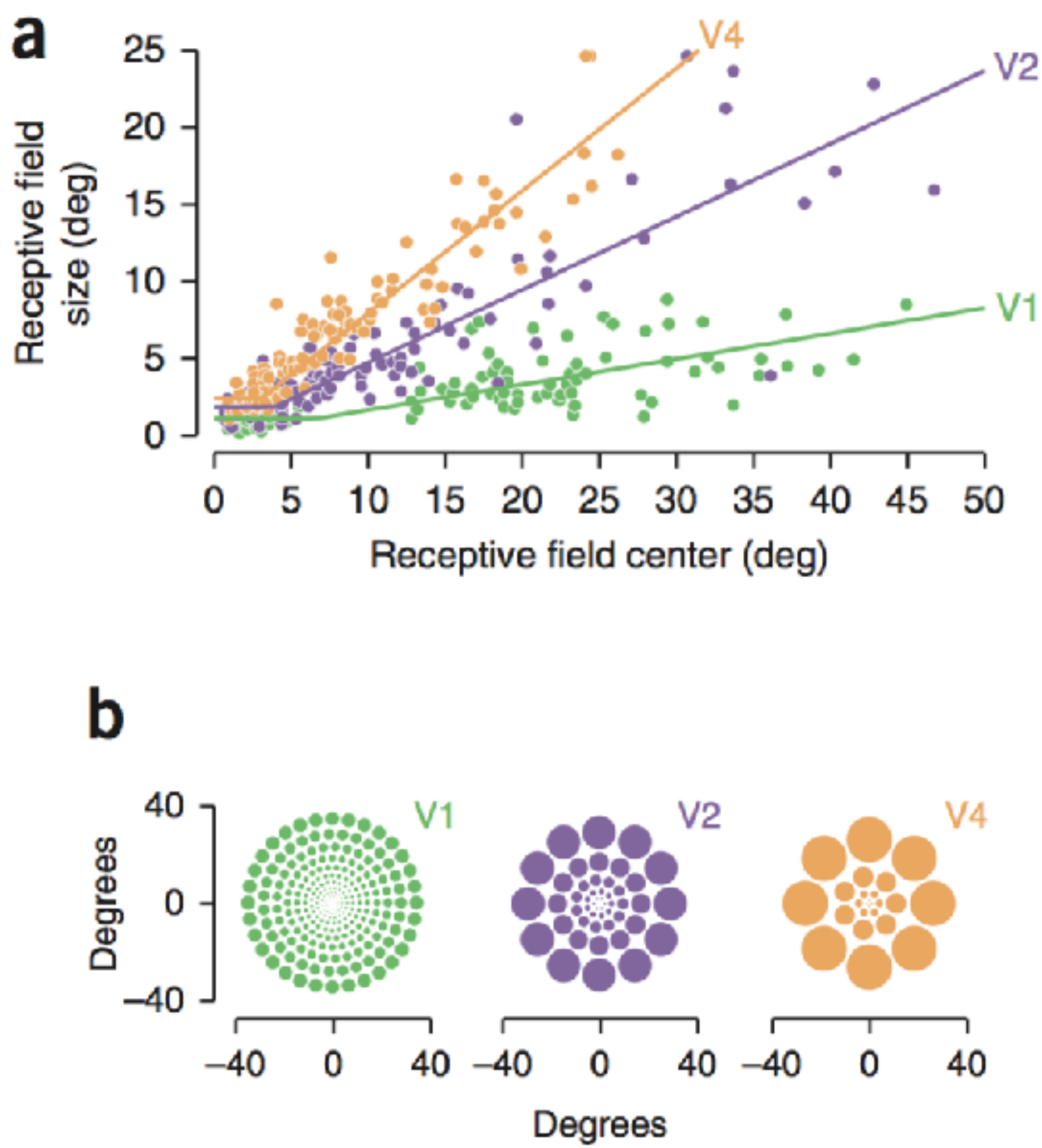
## 1978….1991

Zeki, S. M. (1978). Functional specialisation in the visual cortex of the rhesus monkey. *Nature*, *274*(5670), 423–428.

Felleman, D. J., & Van Essen, D. C. (1991). Distributed hierarchical processing in the primate cerebral cortex. Cerebral Cortex, 1(1), 1–47.

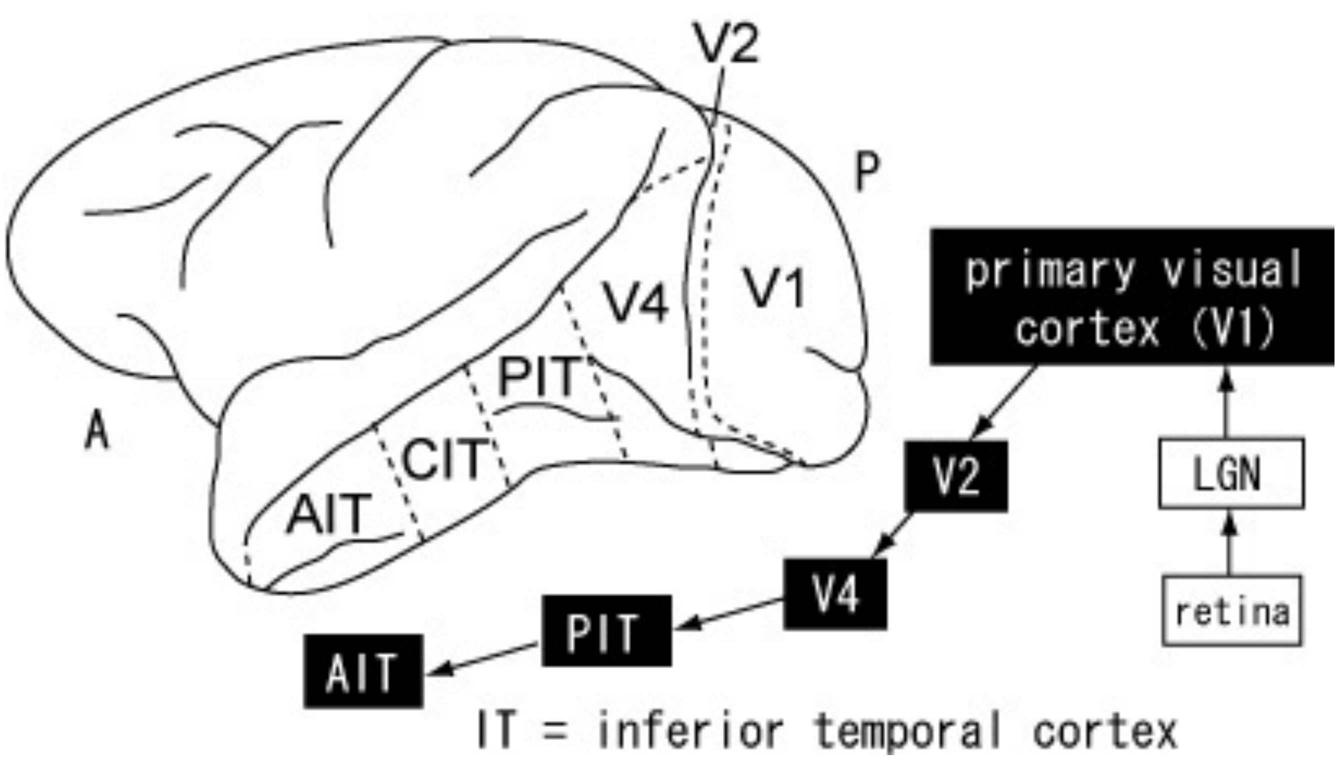


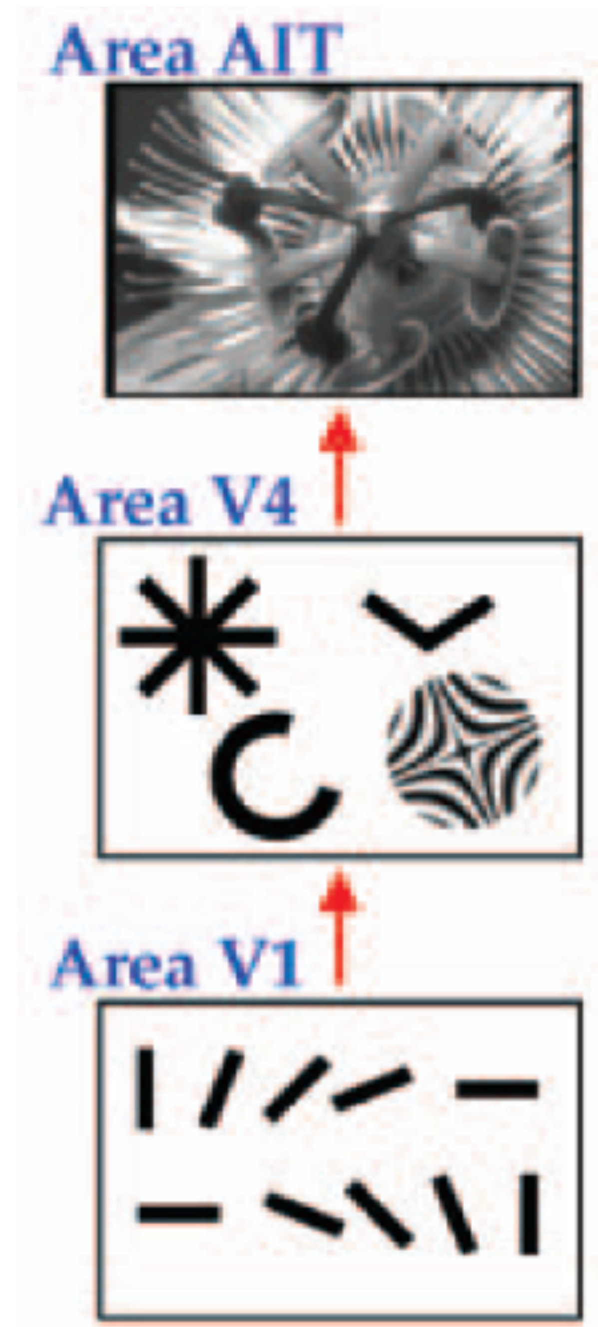

*increasing receptive field sizes, pattern selectivity, invariance to position and scale*




From Freeman & Simoncelli, 2011

# Hierarchical models of object recognition




bread and butter of ventral stream modeling

Hegde and Felleman, 2007

# Hierarchical models for feature extraction for recognition

Local features progressively grouped into more structured representations

- edges => contours => fragments => parts => objects

Selectivity/invariance trade-off

- Increased selectivity for object/pattern type

- Decreased sensitivity to view-dependent variations of translation, scale and illumination

what are the underlying computations
to achieve both selectivity and invariance?

example of recognizing the letter

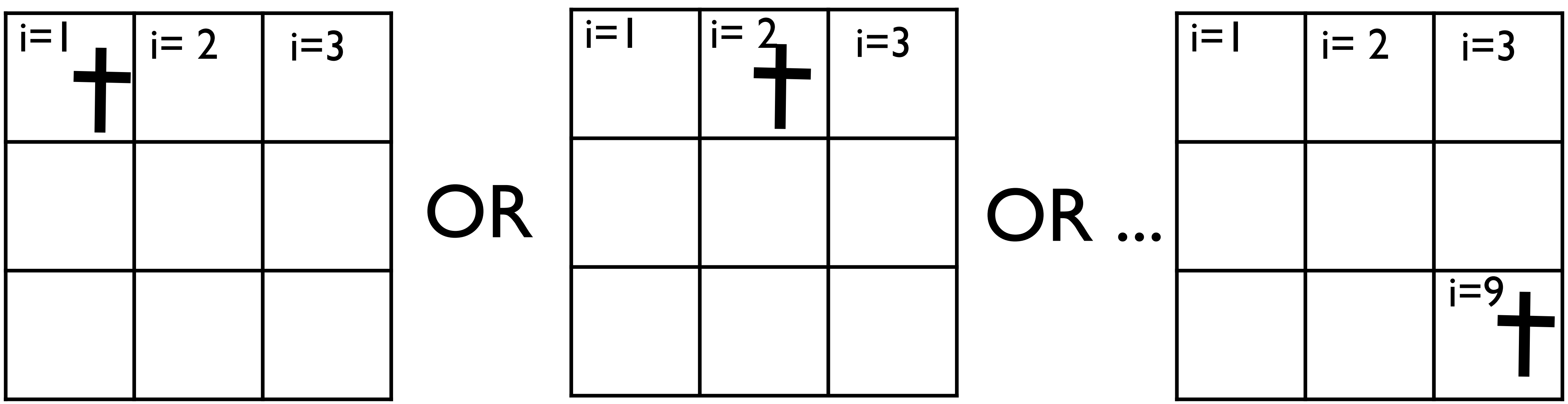
**t**

# ANDs & ORs

## Recognize the letter “t”

“t” is represented by the conjunction of a vertical and horizontal bar:

| AND — = t




which can occur at any one of many locations i

“t”: $h_1 \&\& v_1 \;||\; h_2 \&\& v_2 \;||\; h_3 \&\& v_3 \ldots$

simple and complex cells as AND- and OR-like operations

contributing towards an end-goal of invariant recognition

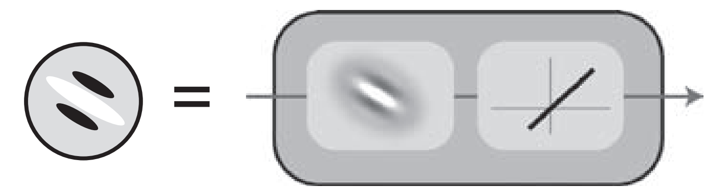
Riesenhuber & Poggio model

- combine the properties of simple- and complex-like cells with hierarchical organization to progressively achieve invariance

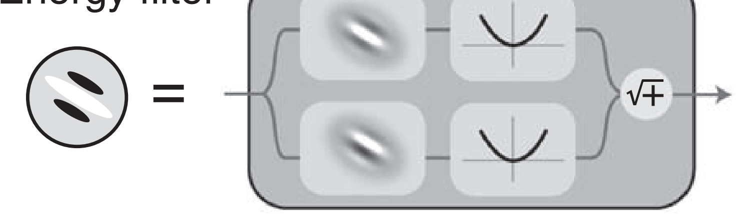
Poggio, T. (2011). The Computational Magic of the Ventral Stream: Towards a Theory. *Nature Precedings*.

# two main classes of V1 cells*

- Simple cells
  - detect conjunctions of inputs
    - similar to a logical AND
    - e.g. of similar pixels to form an edge template
    - “phase sensitive”

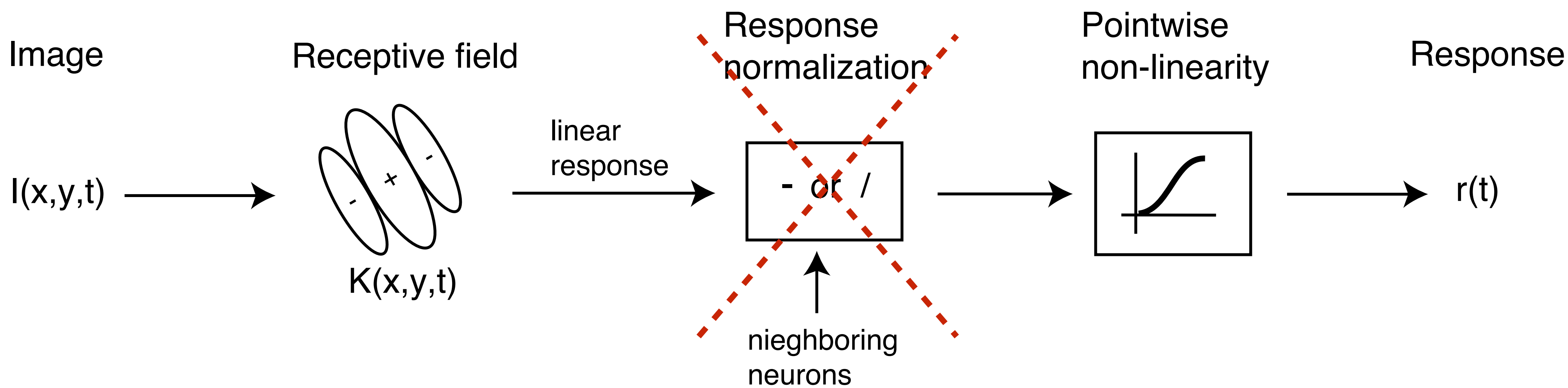


- Complex cells
  - detect disjunctions of inputs
    - similar to a logical OR
    - e.g. any of several similar oriented edges within a region of space will fire cell
    - “phase *in*sensitive”




**The distinction isn’t categorical--i.e. a range of phase sensitivities. And there other types of cells, e.g. end-stopped. See Mechler, F., & Ringach, D. L. (2002). On the classification of simple and complex cells. Vision Research, 42(8), 1017–1033.*

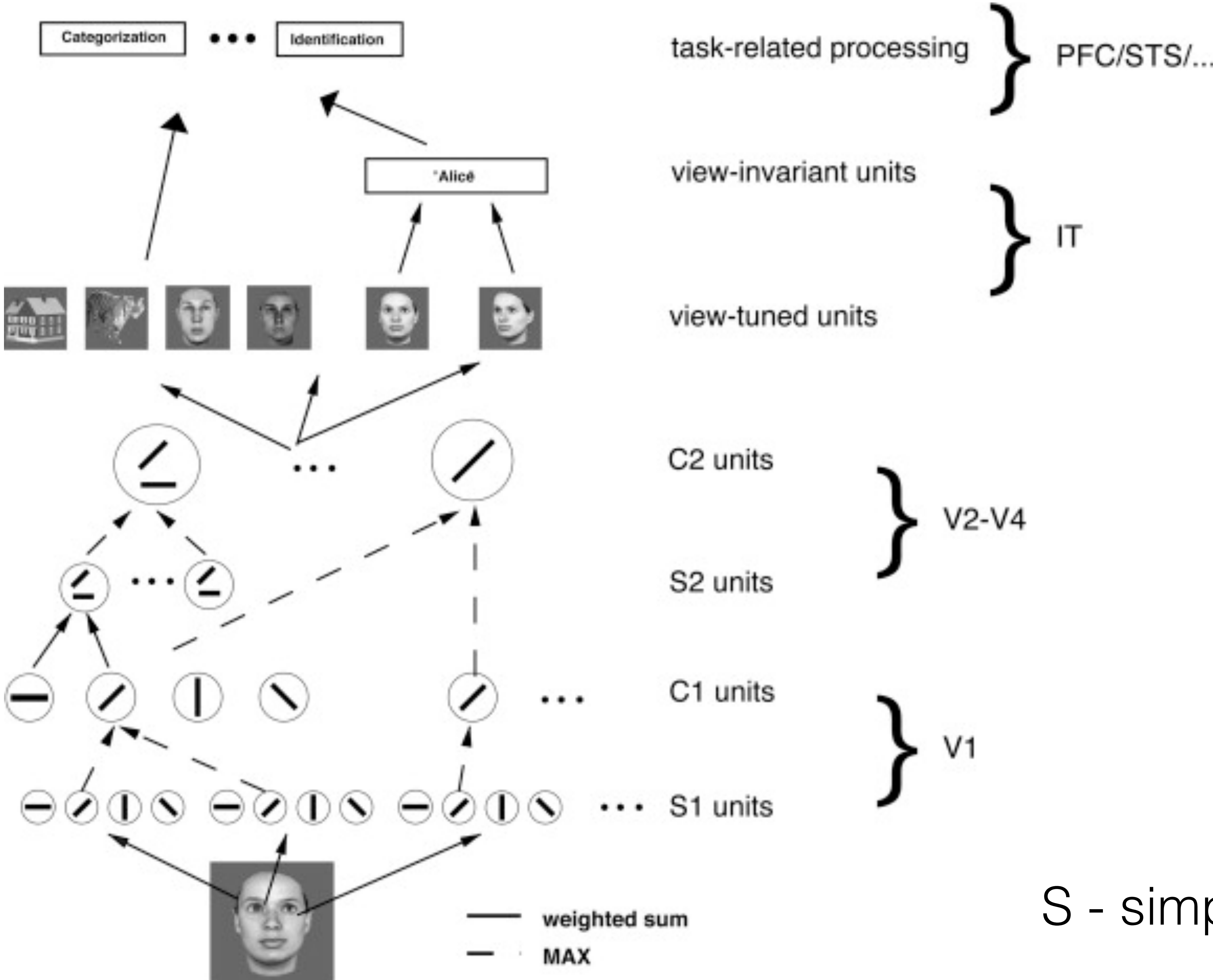
# simple cell feedforward model




convolution — similar filtering operations repeated over space

Similar filtering operations repeated between subsequent levels

$V_n \to V_{n+1}$



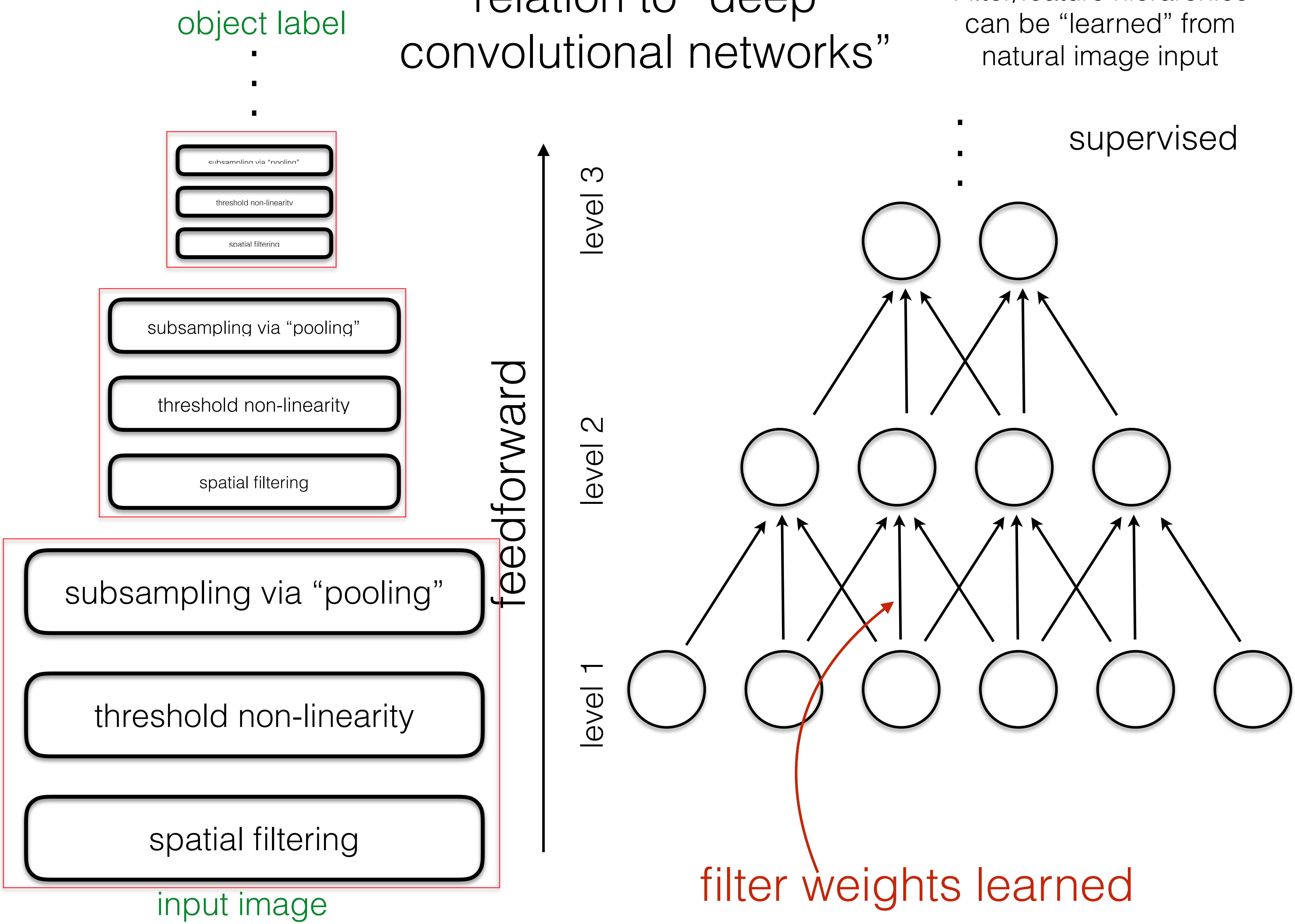

S - simple cell like

C - complex cell like

Riesenhuber & Poggio, 1999

# relation to "deep convolutional networks"

Filter/feature hierarchies can be "learned" from natural image input



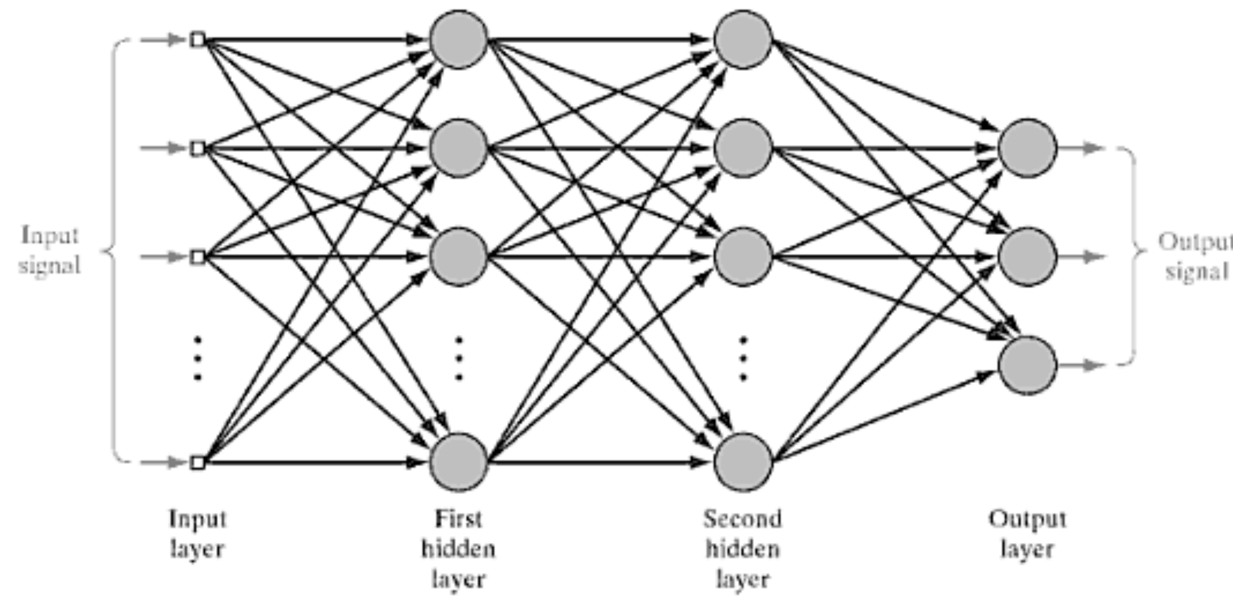


FIGURE 4.1 Architectural graph of a multilayer perceptron with two hidden layers.

Deep convolutional network learning
What's new since the 1980s?

large labelled image datasets
faster computations—GPUs
some tricks to avoid over-fitting

# What determines feature hierarchies?

Grouping to form more abstract features, given image regularities that support tasks

— "hand - wire" based on analysis of computation and neural models

- e.g. Riesenhuber and Poggio, …

— unsupervised learning based on based on successive discovery of image regularities (Barlow)

- detecting "suspicious coincidences":
    - Is p(feature A, feature B) >> p(feature A) p(feature B)
    - if so, recode to remove dependence. E.g. contingent adaptation example
- advantage of general features. but perhaps more useful at lower levels of the hierarchy

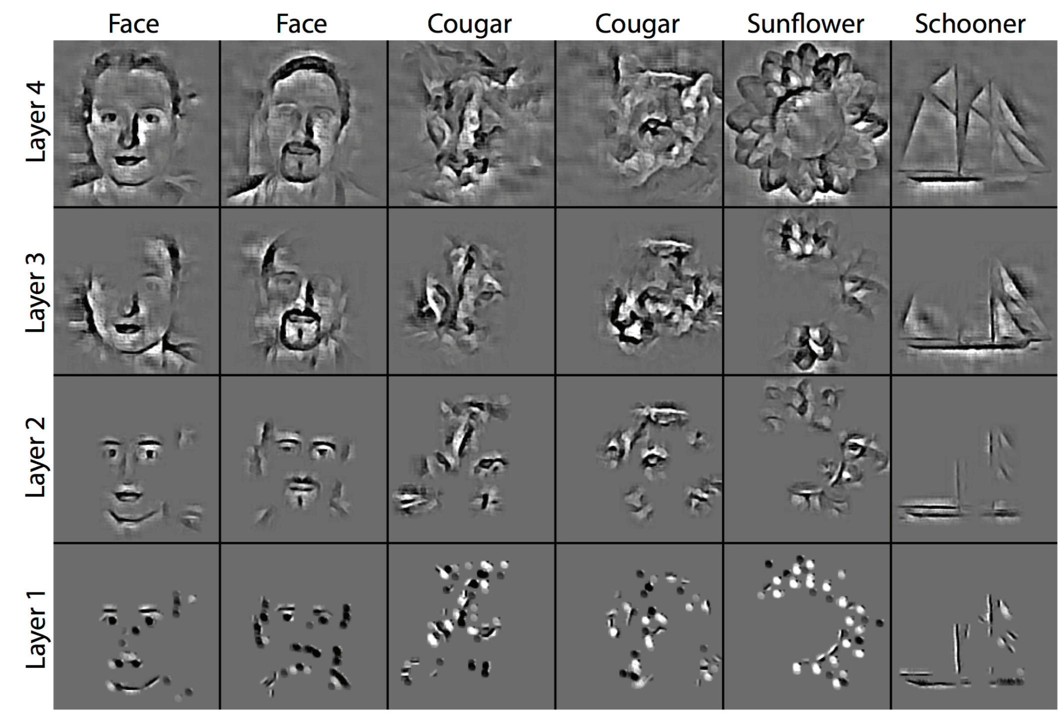
— supervised learning — task dependent

- — "20 questions" approach (Ephstein et al.)
    - find diagnostic features that distinguish the categories for the most important tasks to determine the top level
    - repeat at a lower level of abstract to find sub-features that distinguish the diagnostic features
    - …and so forth
- deep convolutional networks

Filter/feature hierarchies can be “learned” from natural image input

unsupervised


Zhu, L., Chen, Y., Torralba, A., Freeman, W., & Yuille, A. (2011). Part and appearance sharing: Recursive compositional models for multi-view multi-object detection. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1919–1926.




Zeiler, M., Taylor, G., & Fergus, R. (2011). Adaptive deconvolutional networks for mid and high level feature learning. Computer Vision (ICCV), 2011 IEEE International Conference on, 2018- 2025.

“Compositional” constraints:
suspicious coincidences
part-sharing

“Deep belief” networks
learning constrained by generative prediction

Explicit, “symbolic”

Implicit

# What determines feature hierarchies?

Grouping to form more abstract features, given image regularities that support tasks

— “hand - wire” based on analysis of computation and neural models

- e.g. Riesenhuber and Poggio, …

— unsupervised learning based on based on successive discovery of image regularities (Barlow)

- detecting “suspicious coincidences”:
    - Is p(feature A, feature B) >> p(feature A) p(feature B)
    - if so, recode to remove dependence. E.g. contingent adaptation example
- advantage of general features. but perhaps more useful at lower levels of the hierarchy

— supervised learning — task dependent

- — “20 questions” approach (Ephstein et al.)
    - find diagnostic features that distinguish the categories for the most important tasks to determine the top level
    - repeat at a lower level of abstract to find sub-features that distinguish the diagnostic features
    - …and so forth
- deep convolutional networks

# What determines feature hierarchies?
# An example based on task requirements

Need features for rapid, accurate generalization, given a visual task requirement.

*Find features of "intermediate complexity", i.e. image "fragments", that are most informative for category distinctions*

*Ullman, S., Vidal-Naquet, M., & Sali, E. (2002). Visual features of intermediate complexity and their use in classification. Nature Neuroscience*

# Object recognition in the context of a task requirement

What do these scenes have in common?

# “Up” curbs-- requiring a step up

Distinguish from non “up curbs”

...that do not require a step up and require different actions

# Learning based on informative fragments for the task

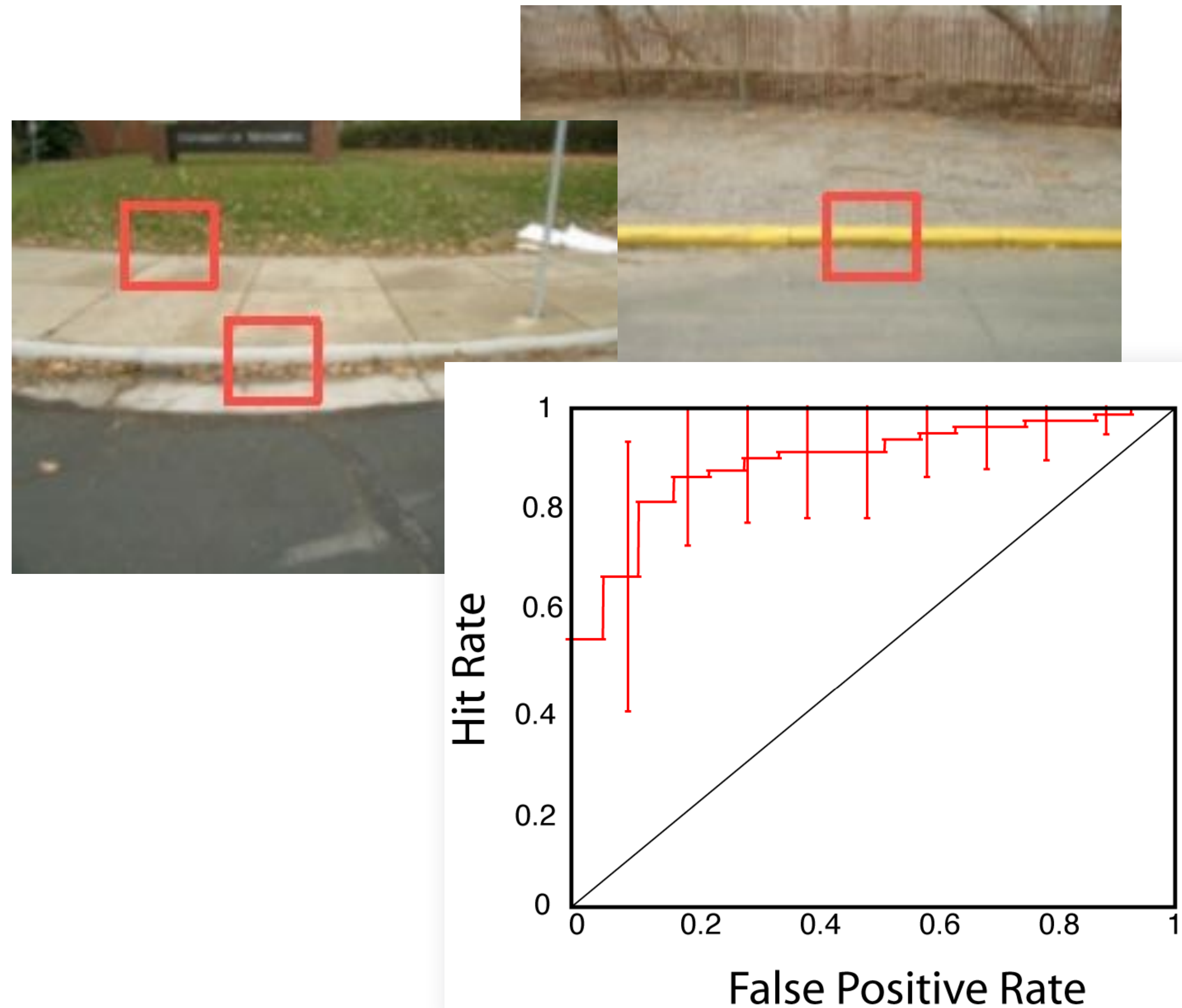
Algorithm finds fragments that maximize mutual information

Detect “up curbs” from an approach angle that requires a step.

View-specific

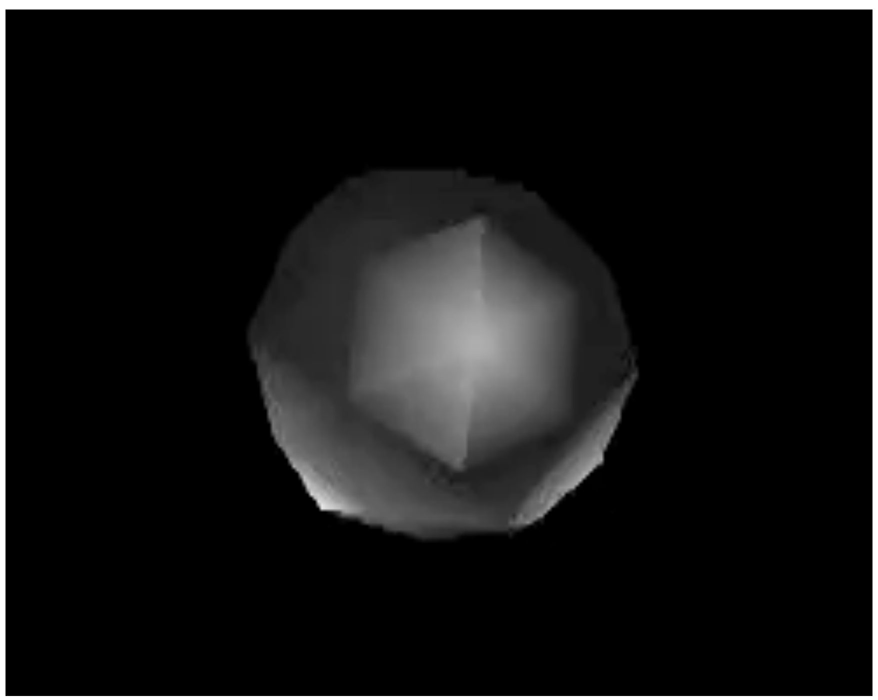
Works well

Experimentally tractable

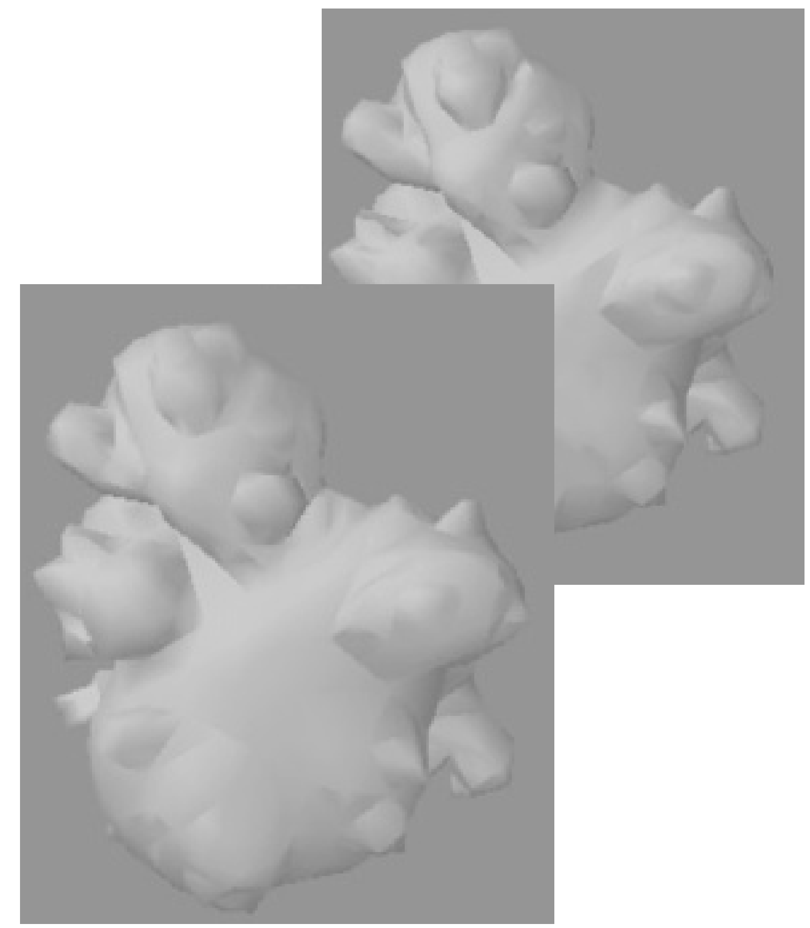



Evgeniy Bart

# Do people learn to use fragments of predicted "intermediate complexity"



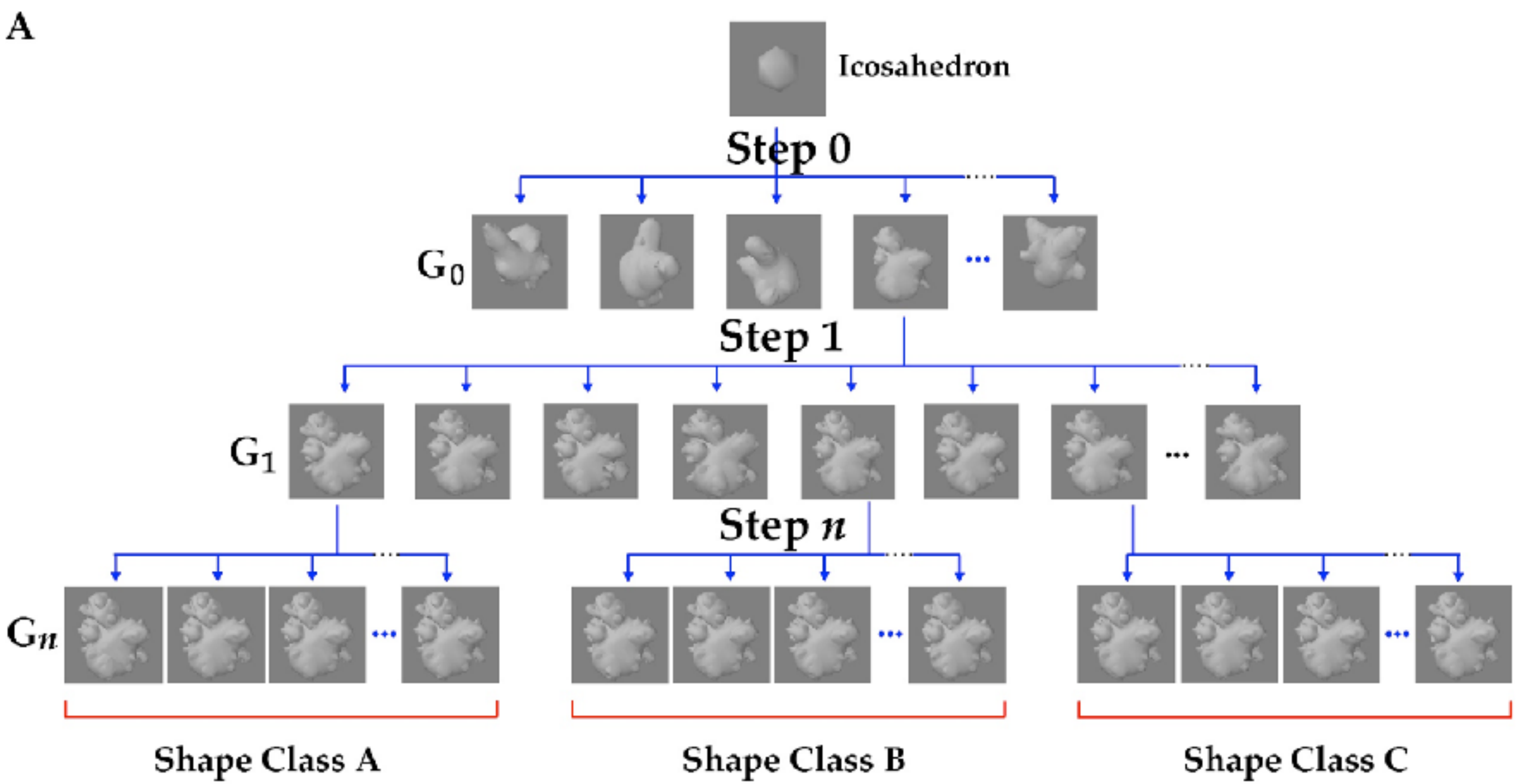
Virtual morphogenesis



Brady, M. J., & Kersten, D. (2003).
Bootstrapped learning of novel objects.
Journal of Vision, 3(6), 413–422.

# Generating naturalistic object classes

## Virtual Phylogenesis



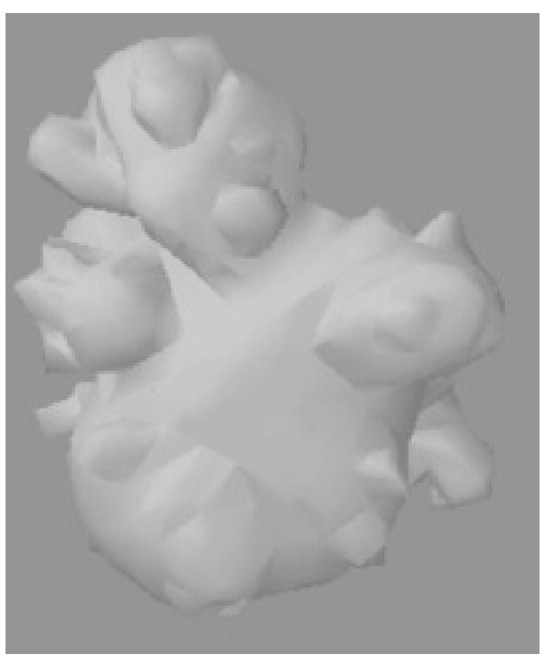

Hegde, J., Bart, E., & Kersten, D. (2008). Fragment-Based Learning of Visual Object Categories. Curr Biol. 18, 597-601

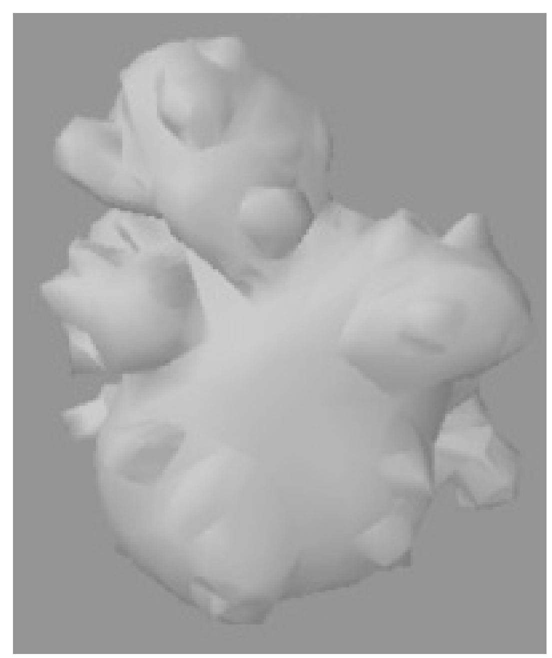
# Training

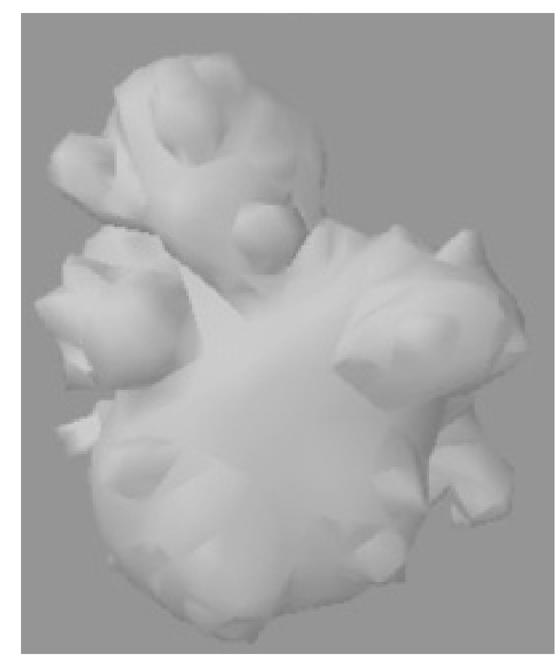
Member of category A or B?

A

B

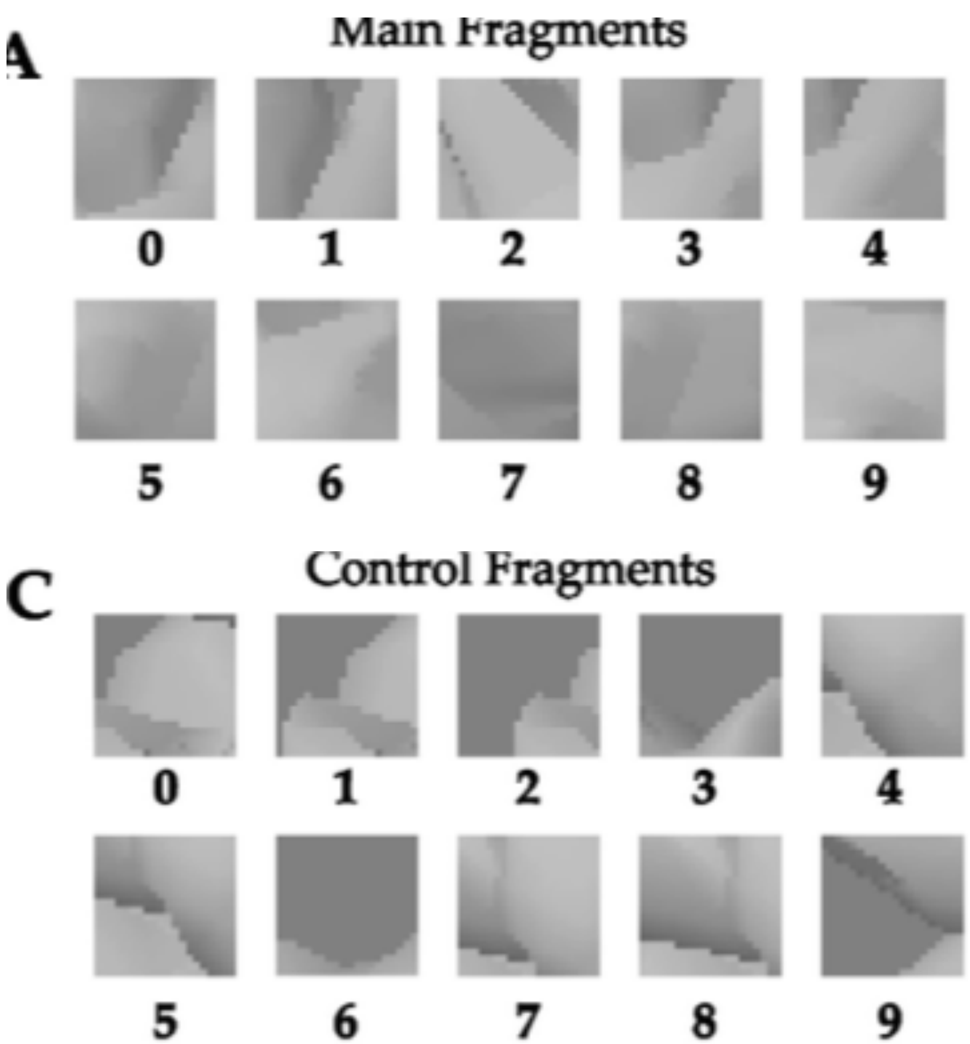


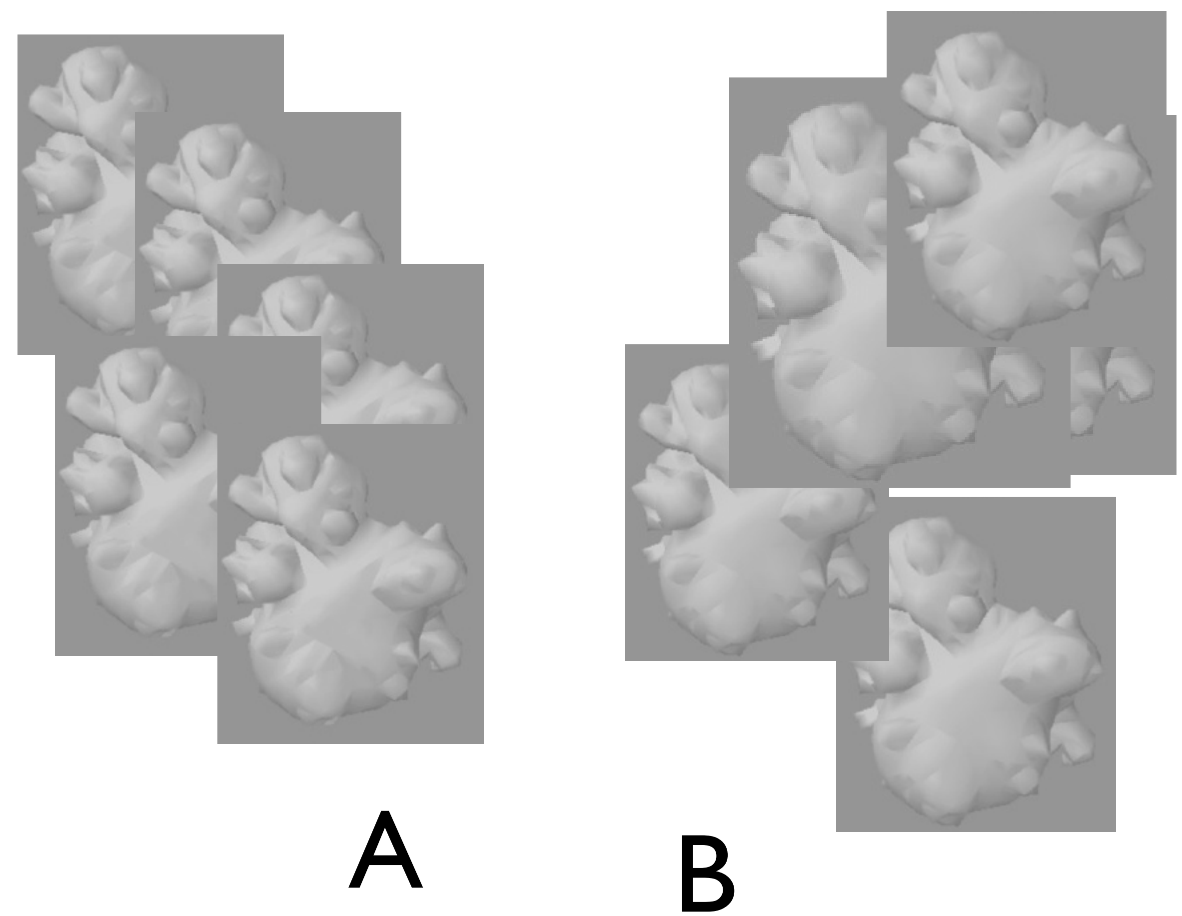




# Results

Features of intermediate complexity (local image patches) predicted human observers ability to classify new objects from learned categories




Hegde, J., Bart, E., & Kersten, D. (2008). Fragment-Based Learning of Visual Object Categories. Curr Biol. 18, 597-601