Feature Hierarchies and Object Recognition

Thomas Breuel

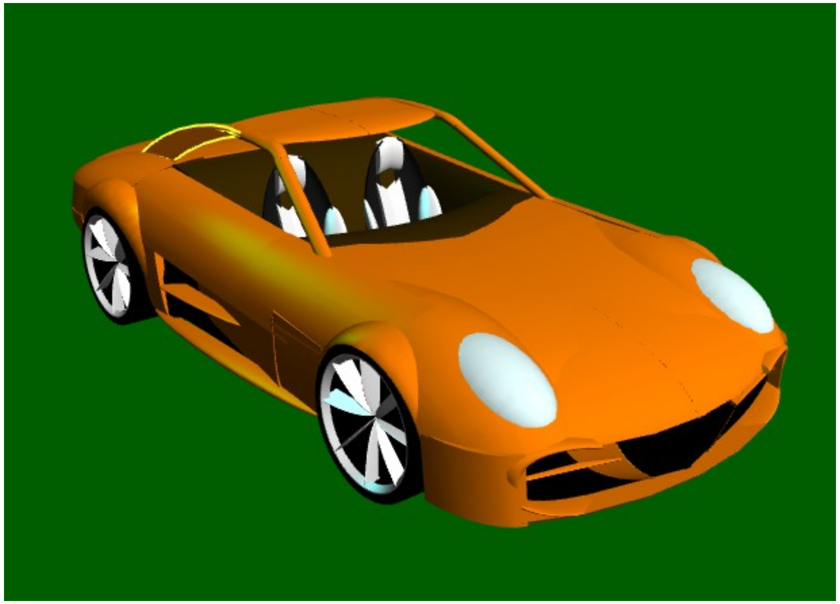
VISUAL OBJECT RECOGNITION

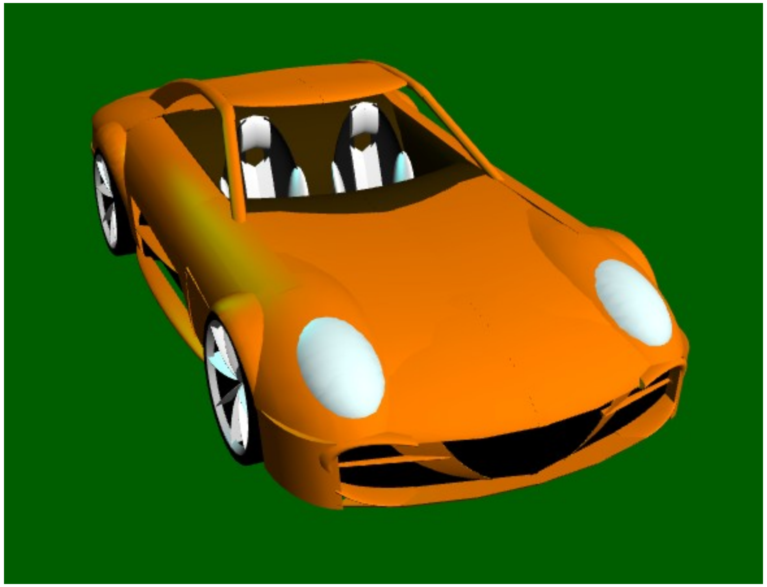
motivation: human recognition

visual recognition in humans + animals:

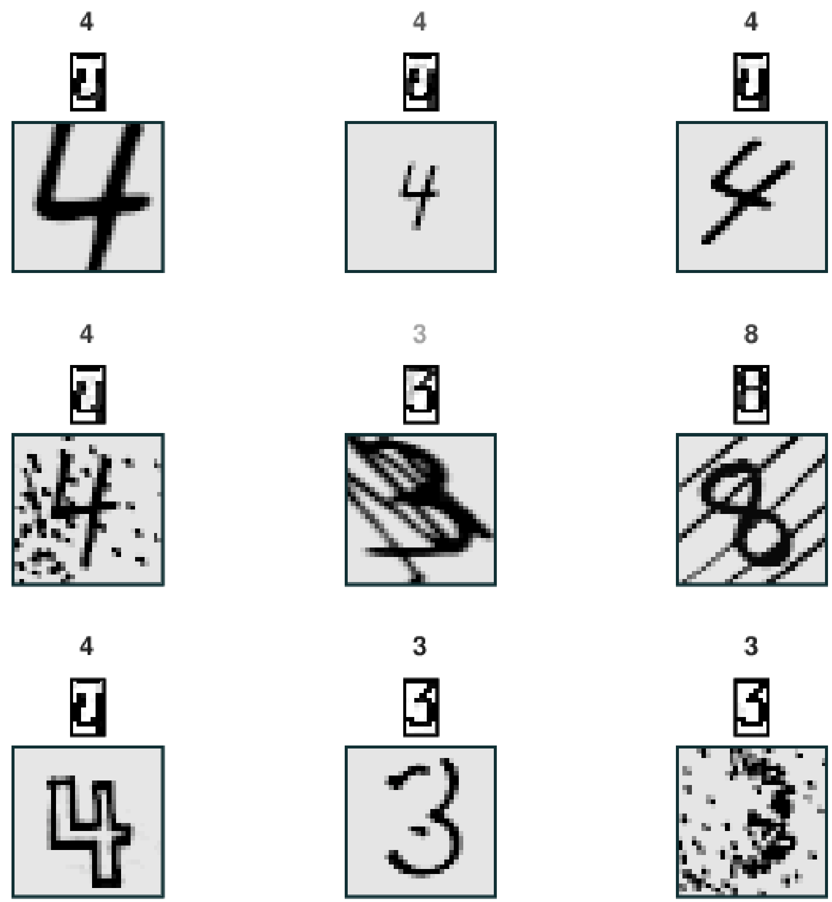
- **2D scale and translation invariant from single presentation**
- **not fully invariant to 3D viewpoint or lighting**
- **recognition is fast (= limited opportunity for feedback)**

invariant object recognition





invariant recognition for digits



engineering approach

- understand the physics and geometry
- develop models and algorithms

- 3D object models
- feature extraction, edge detection
- geometric matching

machine learning approach

- collect a lot of data
- pick a powerful, scalable machine learning algorithm
- train to predict specific object classes

neuromimetic approach

- determine the areas and connections between brain areas
- determine the function and computations of brain areas
- implement equivalent functions in software and use them for recognition

common approaches

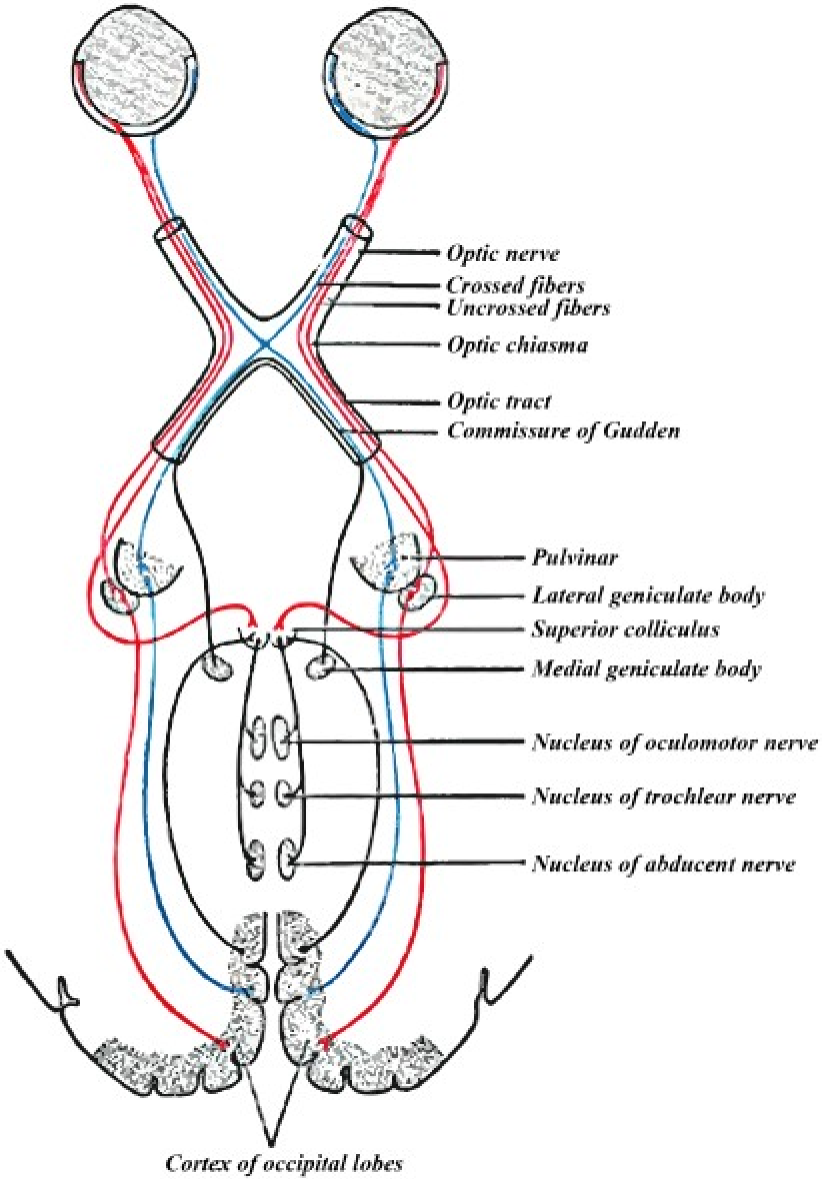
- **no single approach sufficient**
- **therefore...**
  - engineering + machine learning
  - **neuromimetic + machine learning**

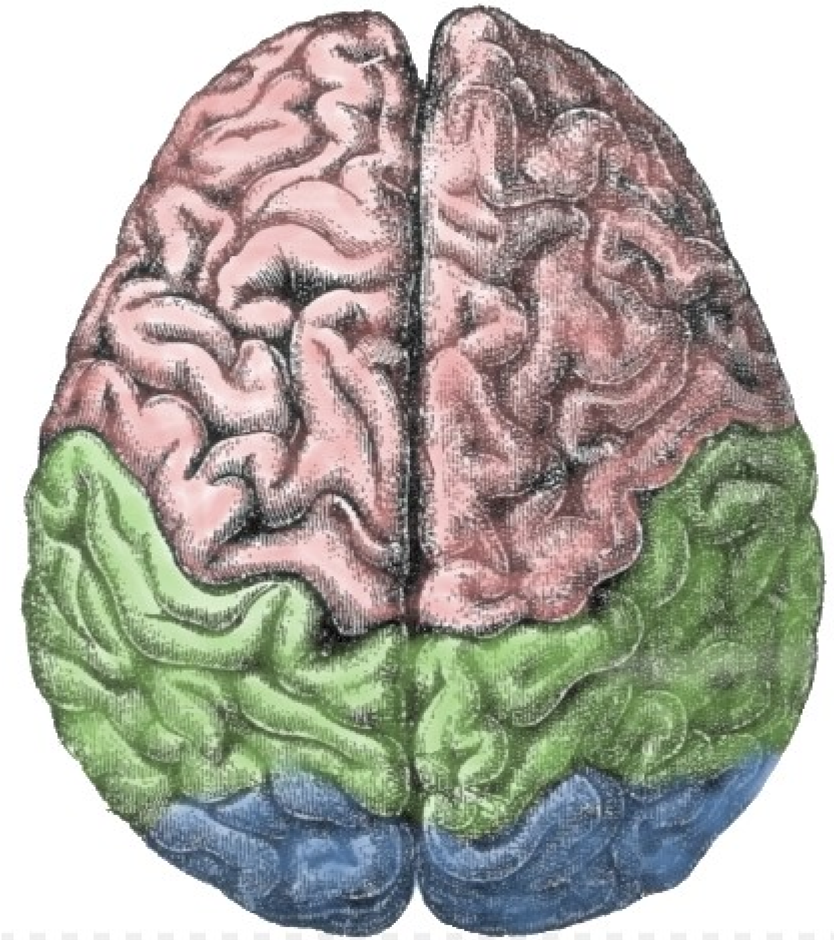
basis for neuromimetic systems

- find the modules making up the visual system
- determine their connectivity
- determine their functions / computations

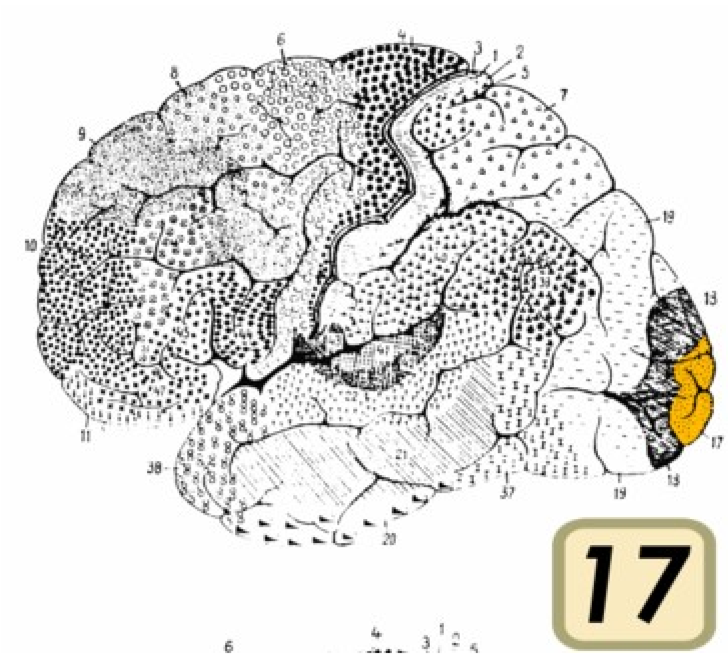
anatomy

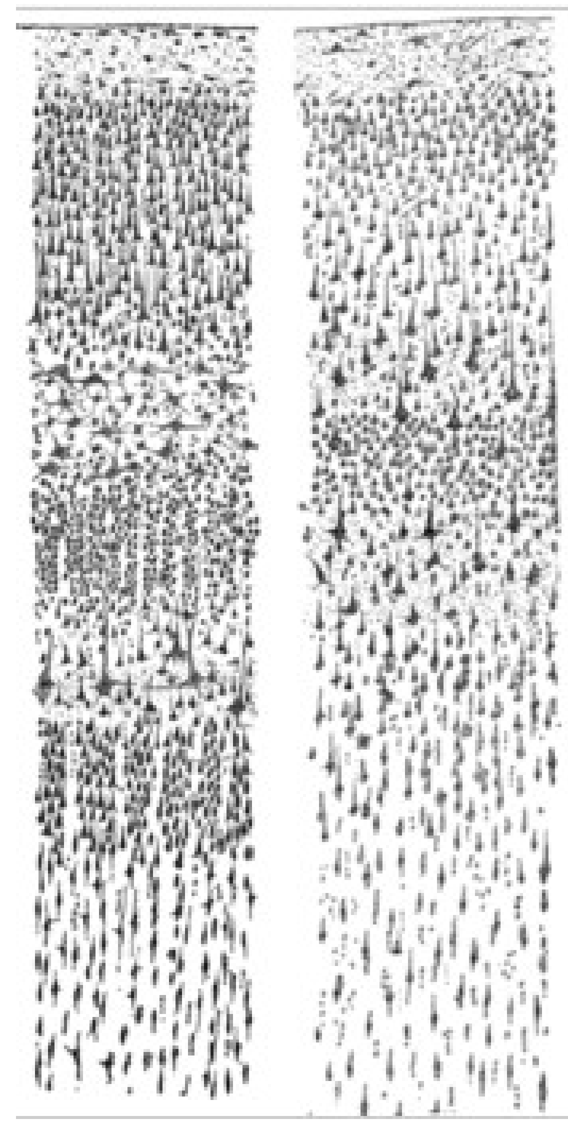
general structure



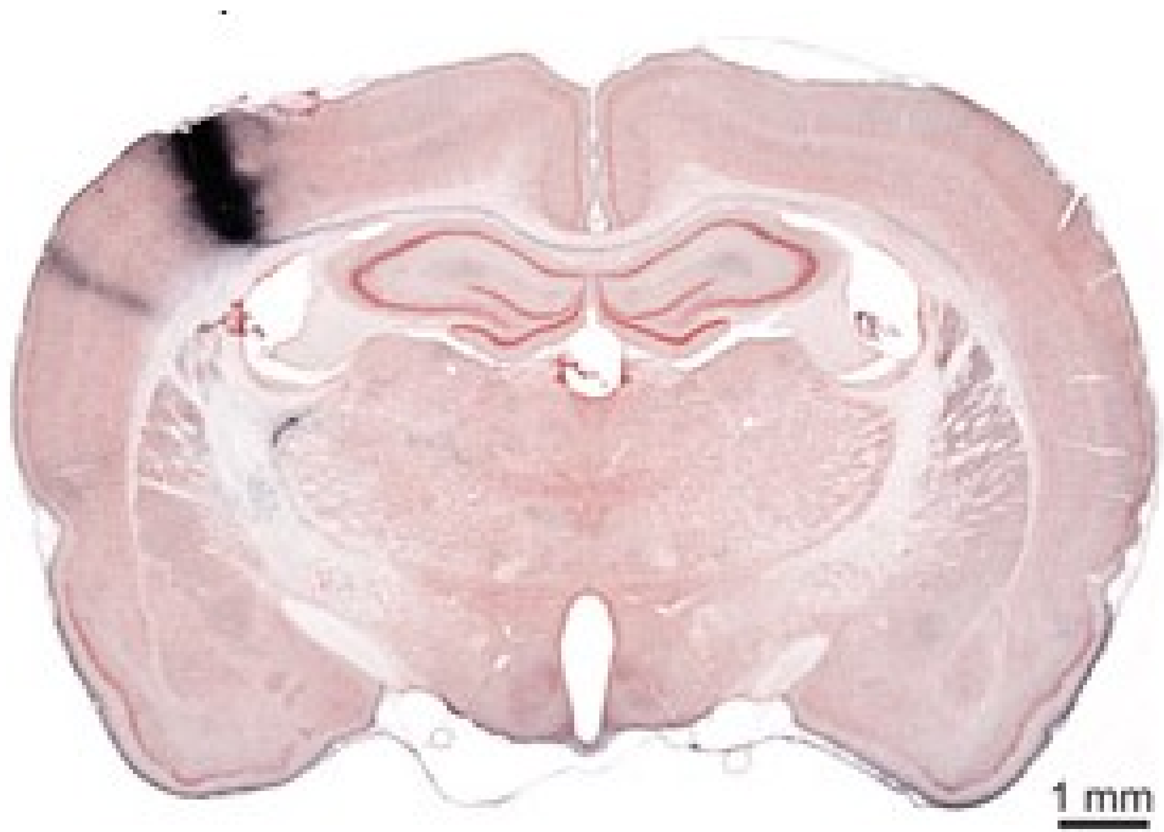


Brodmann's Areas



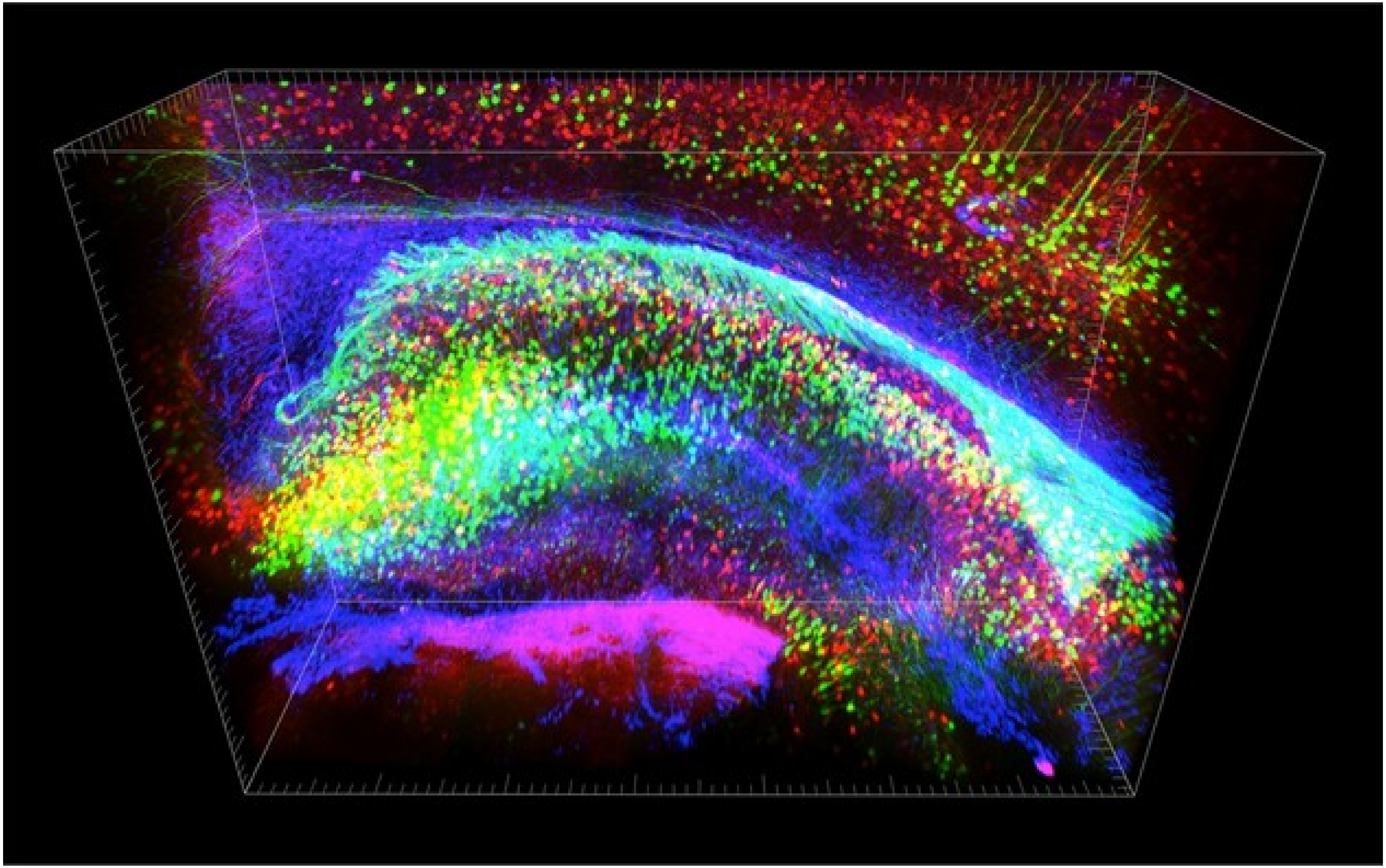


pathway tracing



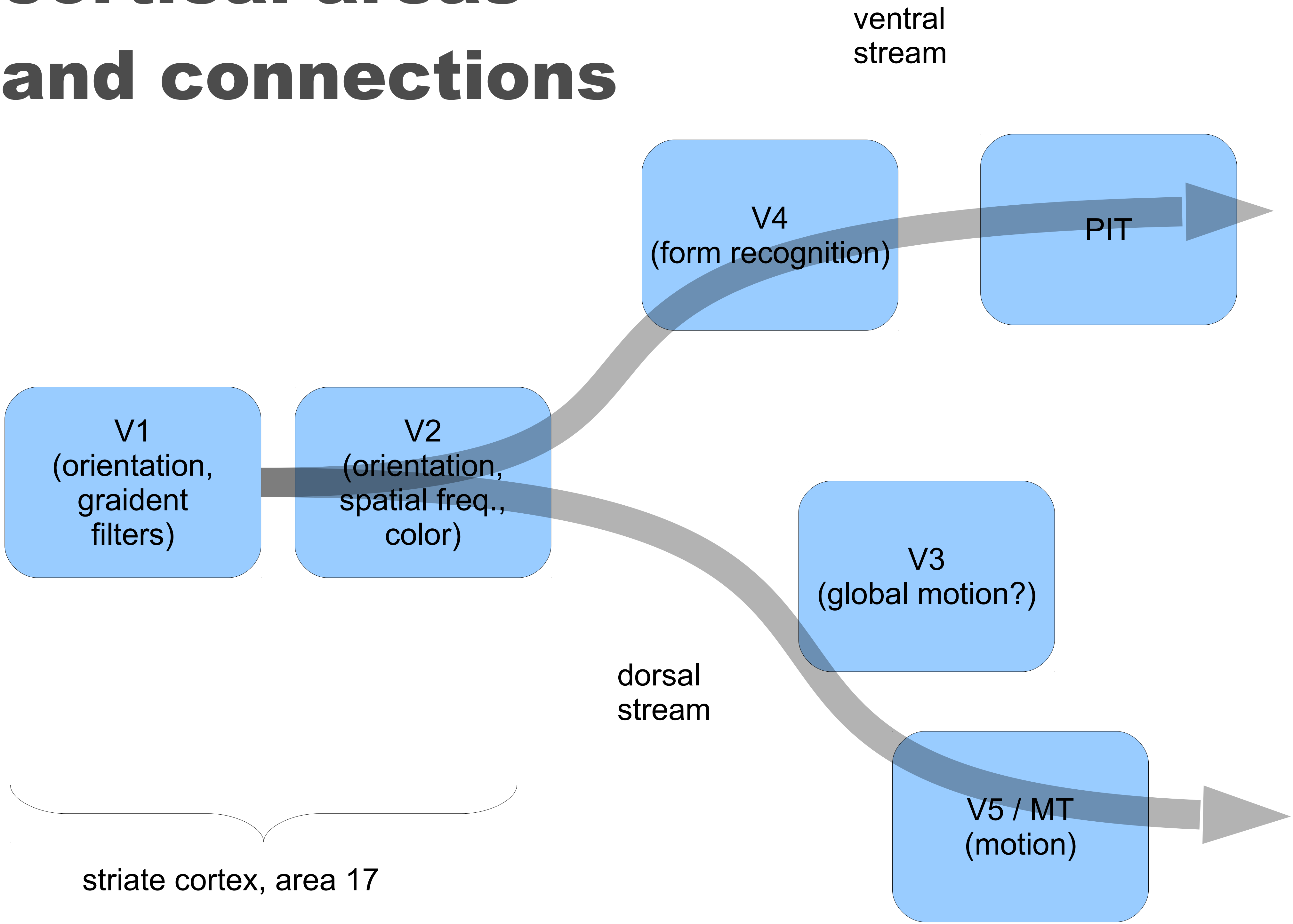
http://cmbn-approd01.uio.no/zoomgen/connectivity/home.do

CLARITY

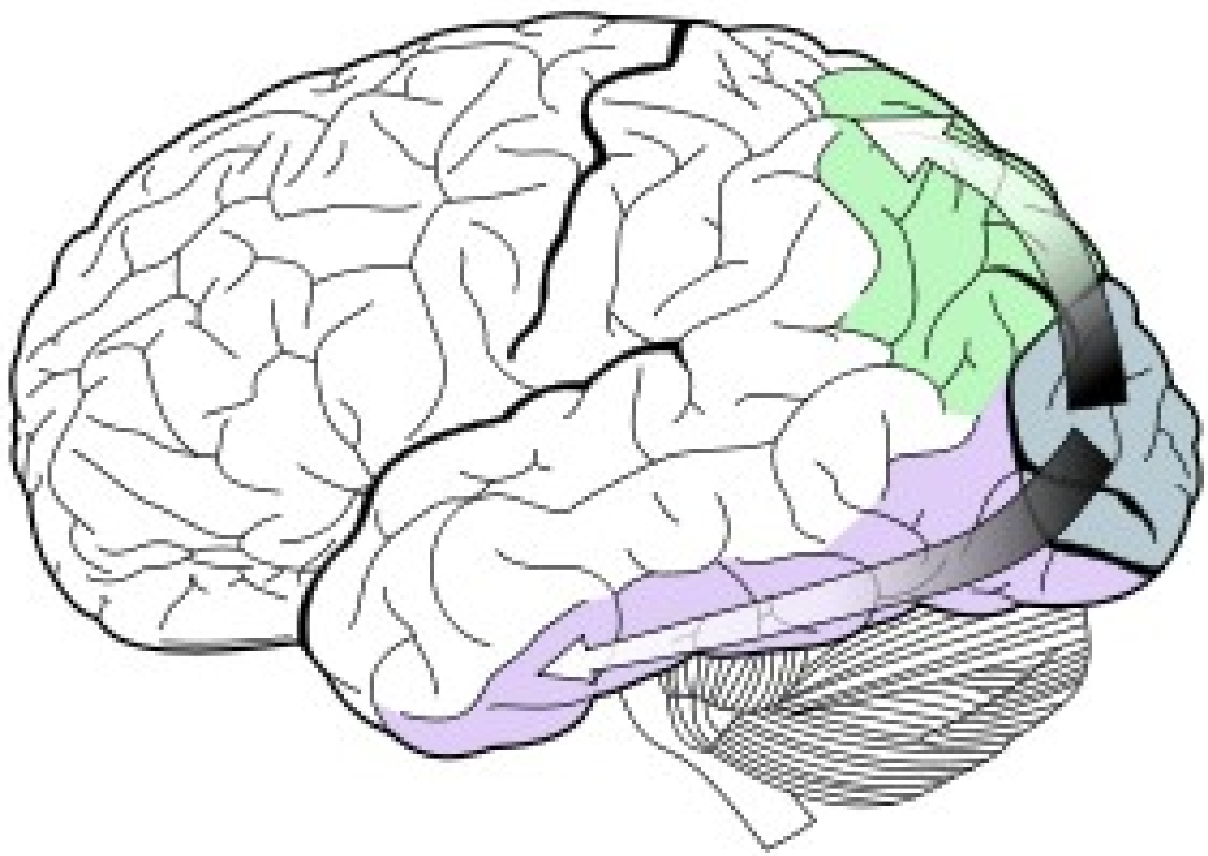


http://www.nature.com/news/see-through-brain-reduced-colours-jpg-7.9883?article=1.12768

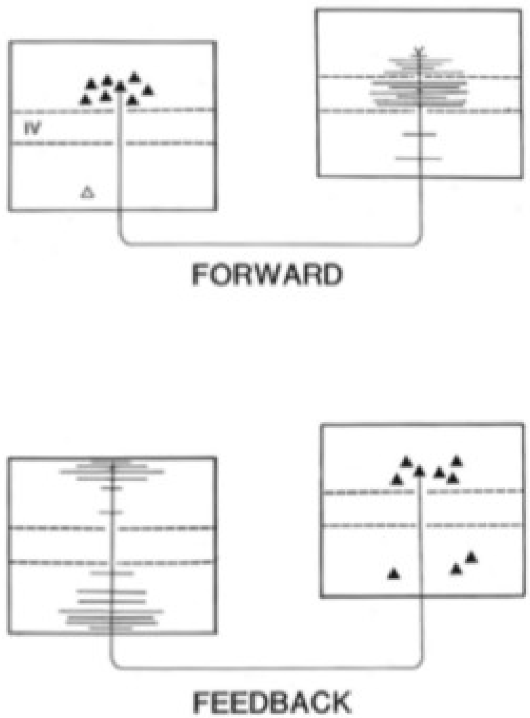
cortical areas and connections



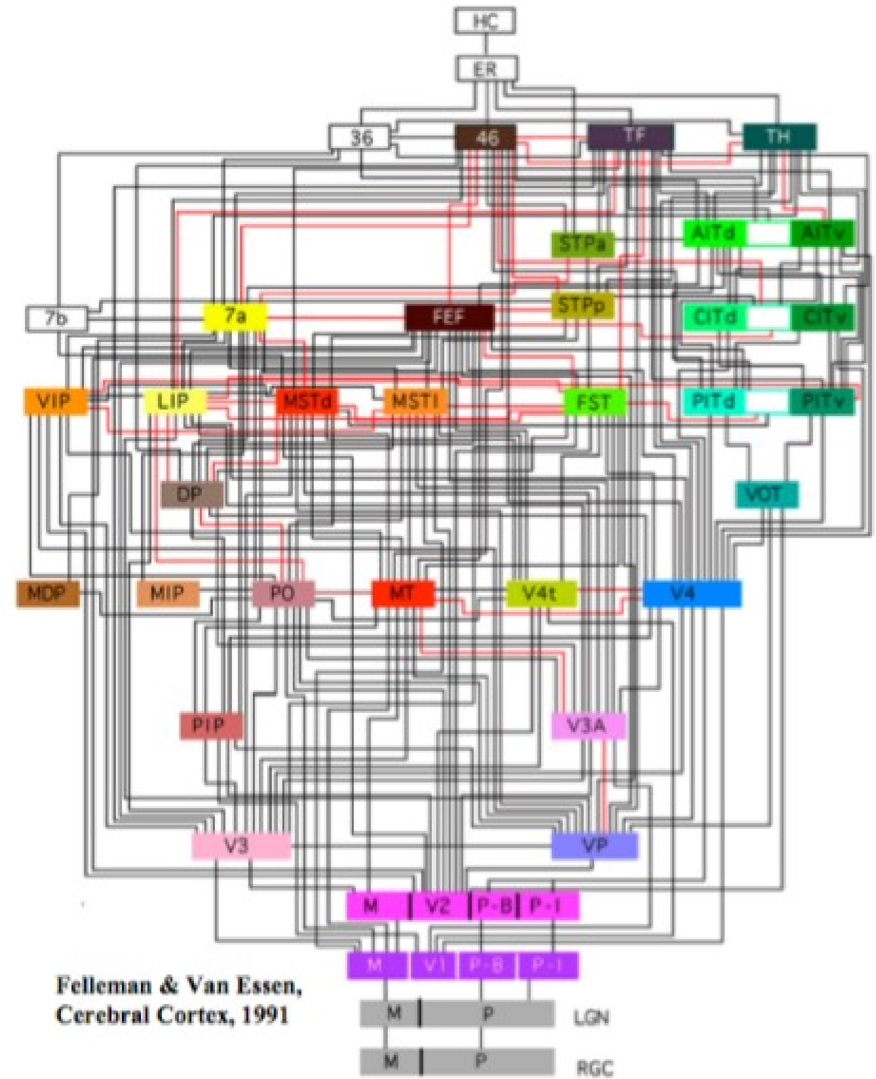
dorsal / ventral stream



detailed maps



Maunsell & Van Essen, J. Neurophysiol., 1983



Felleman & Van Essen, Cerebral Cortex, 1991

function from lesions

- strokes, tumors destroy parts of the brain
- what deficits do we observe as a result?
- this tells us about potential functions

electrophysiology of visual areas

extracellular recording

http://www.youtube.com/watch?v=IOHayh06LJ4

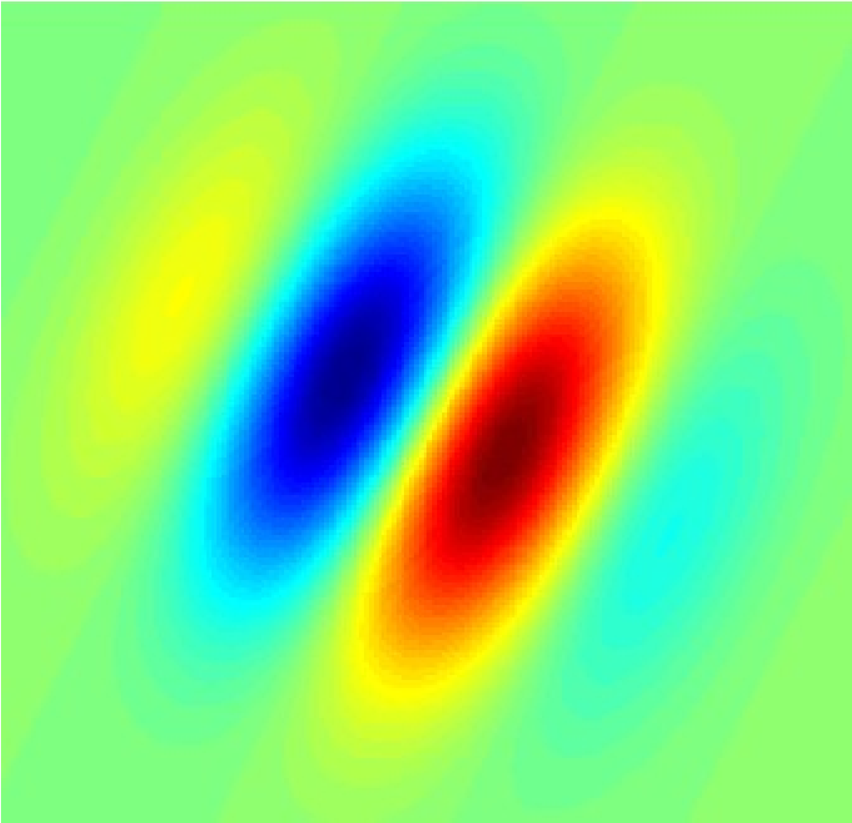
simple cells

- identifiable excitatory / inhibitory regions
- stimuli in these regions are additive
- excitatory / inhibitory regions are antagonistic
- responses can be predicted from the location of the receptive field and stimuli
- found in V1

simple cell model: Gabor filters

- gaussian times oriented sine wave
- linear filter

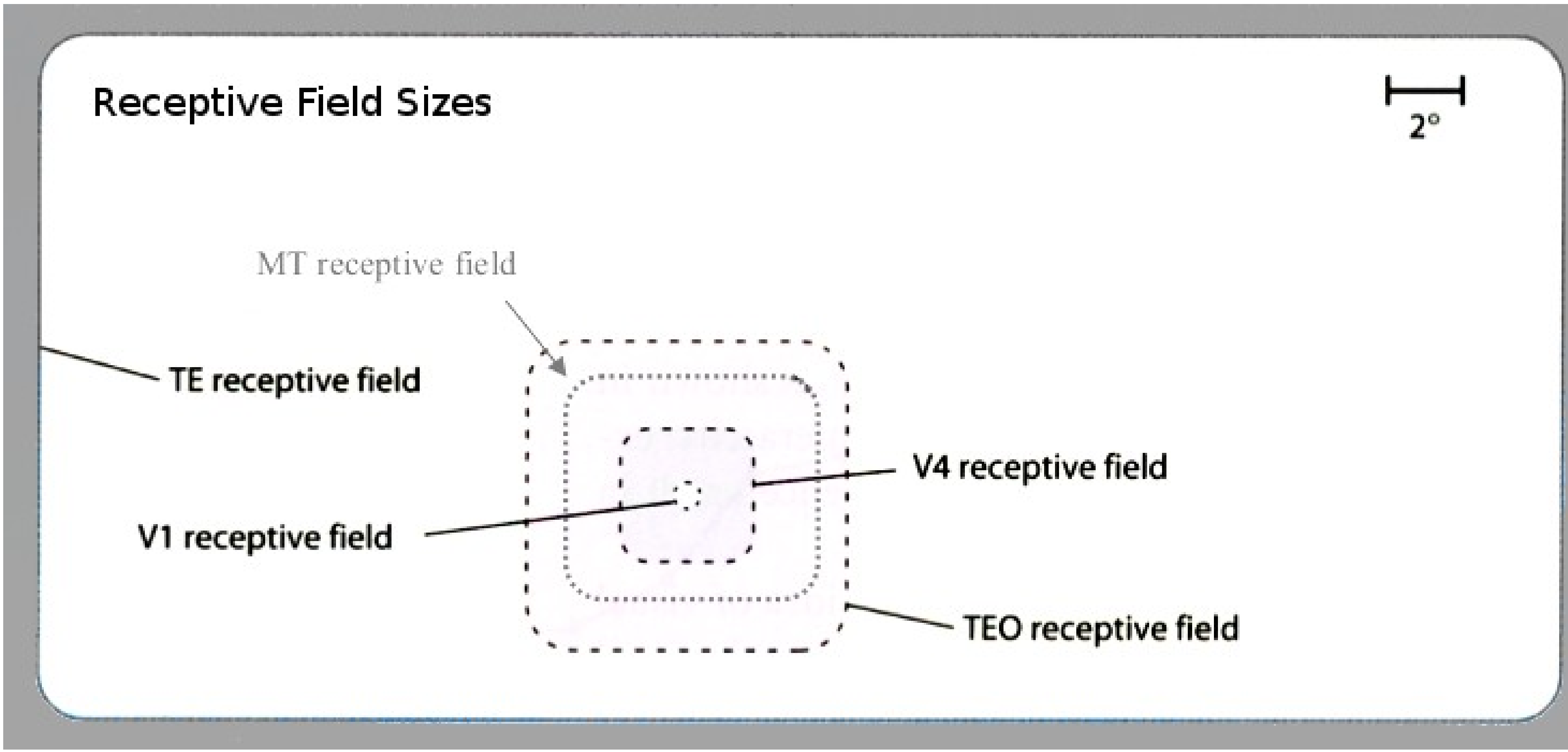
- NB: alternative models have been suggested



complex cells

- responds to oriented edges and gratings (like simple cell)
- response possible anywhere in receptive field
- no excitatory/inhibitory regions
- complex cells receive input from simple cells
- found in V1, V2, and V3

receptive field sizes



questions

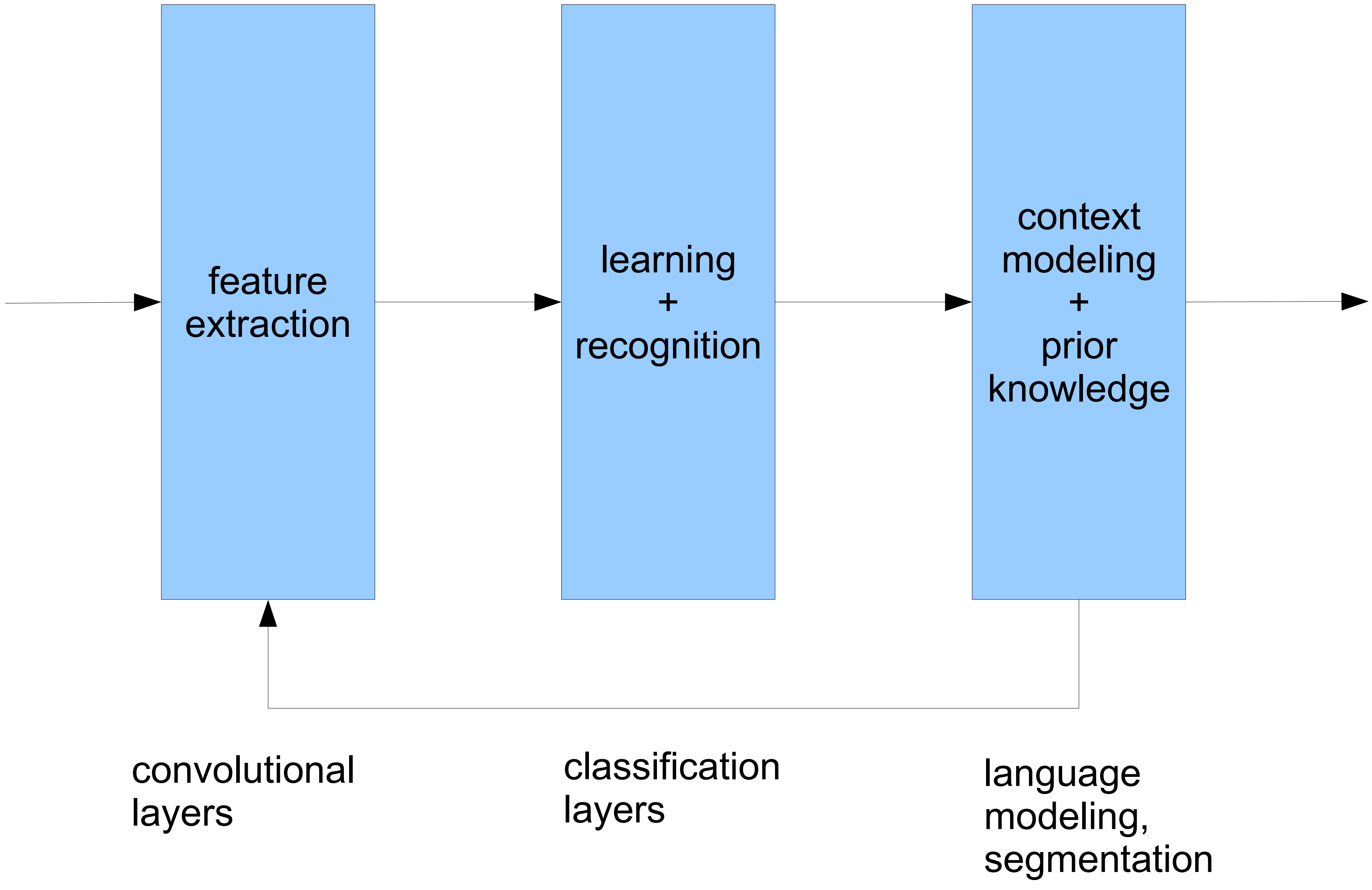
- how do complex cells compute what they do?
- how do simple and complex cells fit in with visual object recognition as a whole?

observation

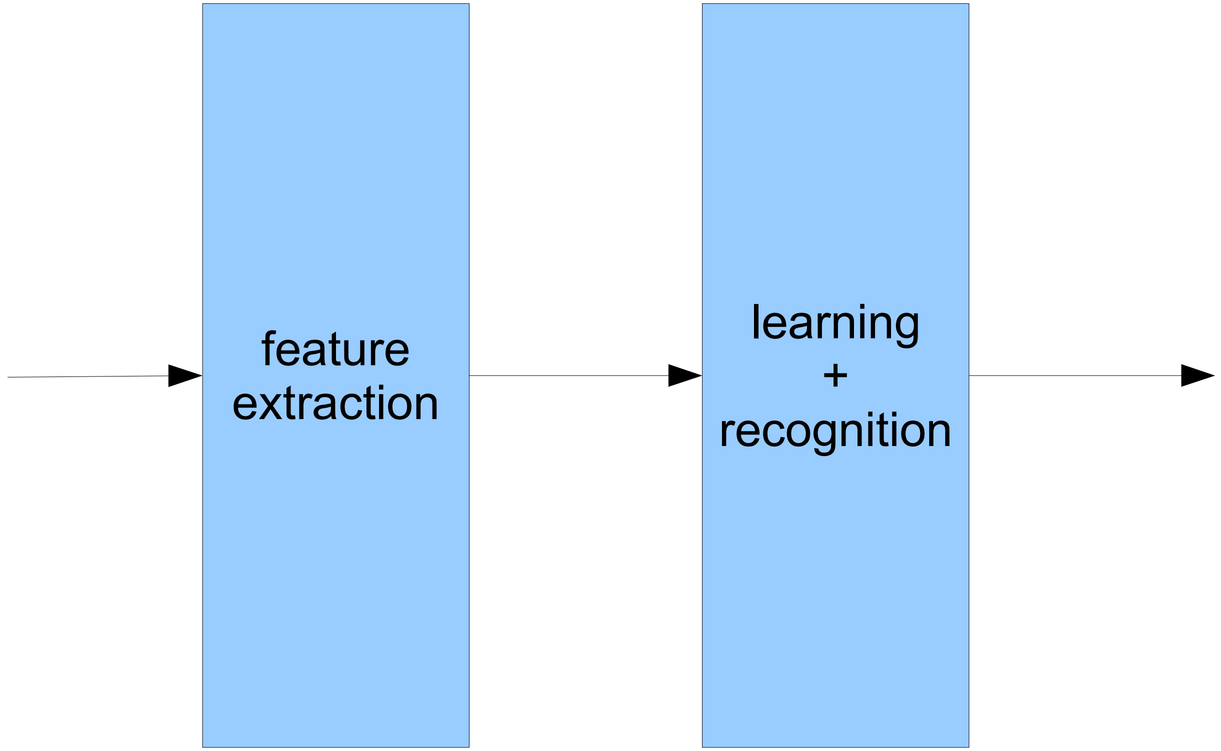
- **there's a huge amount of data (neural data is easy to measure)**
- **there are huge gaps even in the understanding of V1 and V2**
- **computational modeling might help us understand what works and what doesn't better**

ROADMAP

example 1: LeNet



example 2: HMAX



simple and complex
cells in V1 and V2

standard classifiers
(e.g., SVM, neural
network, logistic
regression)

questions

- **mathematical relationship between hierarchies and invariances?**
- **complexity (how many do you need)?**
- **relationship to non-invariance of visual system?**
- **development of neural connections?**
- **degree of agreement between experiments and theory; falsifiability?**

neuromimetic approaches

- **hard-wired**
  - feature hierarchies are hard-wired in the brain
    or at least assumed as somehow given and fixed
  - example: Riesenhuber & Poggio's HMAX model

- **learning**
  - feature hierarchies are learned based on input data
  - example: convolutional neural networks
  - example: unsupervised learning (ICA, sparse coding, ...)

MODELS

CONVOLUTIONAL NEURAL NETWORKS

key paper

Gradient-Based Learning Applied to Document Recognition

Y. LeCun, L. Bottou, Y. Bengio, P. Haffner

main message

"Better pattern recognition systems can be built by relying more on automatic learning, and less on hand-designed heuristics."

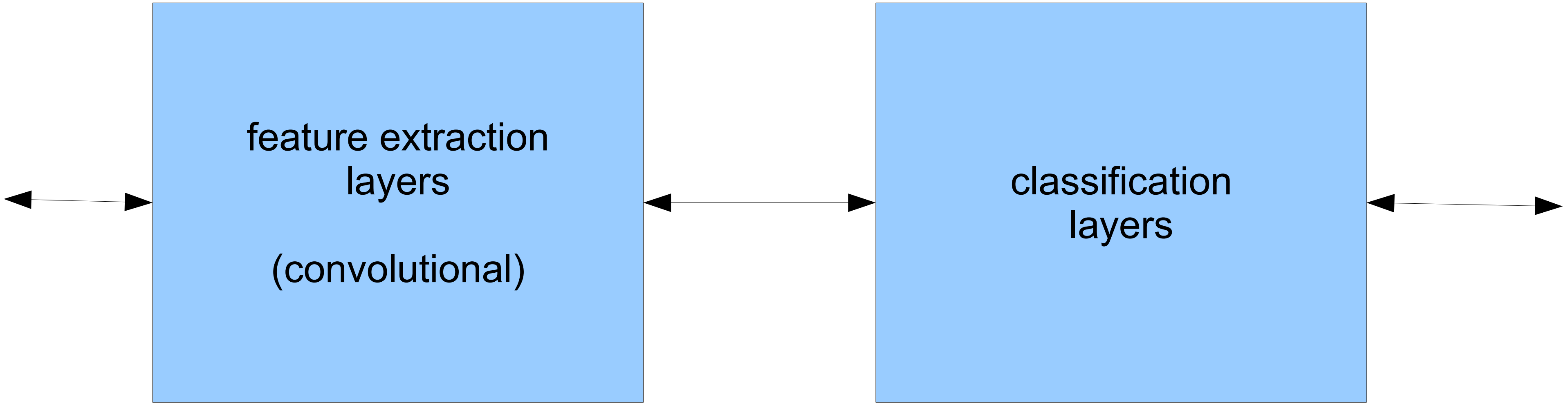
key idea

- **Use MLPs also for feature extraction, not just for classification.**

feature extractors

- **feature extractors**
  - local "receptive fields"
  - uniformly applied over the image
- **convolutional neural networks**
  - each shifted input image → separate input pattern
  - force input weights to be zero outside a small window
  - share weights together for different shifts
  - subsample the outputs
  - eventually, switch from convolutional to global network

training of convolutional neural networks



Activations and deltas propagated as usual, even beteween convolutional and non-convolutional layers.

LeNet-5 architecture

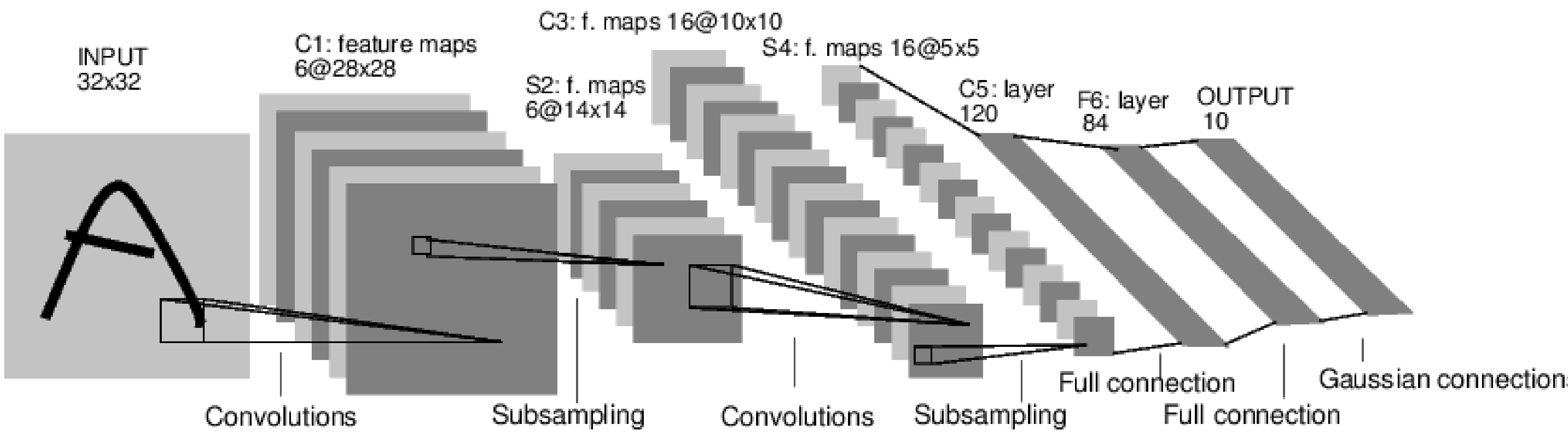


Fig. 2. Architecture of LeNet-5, a Convolutional Neural Network, here for digits recognition. Each plane is a feature map, i.e. a set of units whose weights are constrained to be identical.

Note analogy to visual areas and increasing receptive field sizes.

S2-C3 connectivity

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	X				X	X	X			X	X	X	X		X	X
1	X	X				X	X	X			X	X	X	X		X
2	X	X	X				X	X	X			X		X	X	X
3		X	X	X			X	X	X	X			X		X	X
4			X	X	X			X	X	X	X		X	X		X
5				X	X	X			X	X	X	X		X	X	X

TABLE I

EACH COLUMN INDICATES WHICH FEATURE MAP IN S2 ARE COMBINED BY THE UNITS IN A PARTICULAR FEATURE MAP OF C3.

- **rationale: lack of total connectivity...**
  - breaks symmetry
  - forces multiple different representations
  - reduces # parameters

output layer

- **output layer uses "RBFs"**
  - Euclidean distance between output and weights
  - log of output of unit computing Gaussian
  - fixed covariance matrix

output representation

- **algorithm is trained to return stylized characters as targets**
- **rationale: confusable classes are similar in that representation**

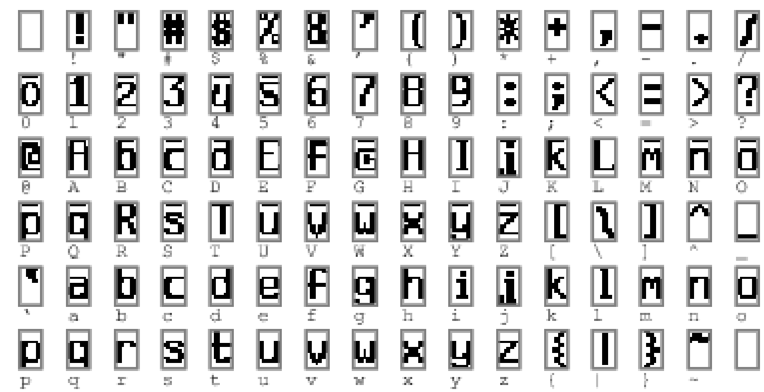


Fig. 3. Initial parameters of the output RBFs for recognizing the full ASCII set.

results

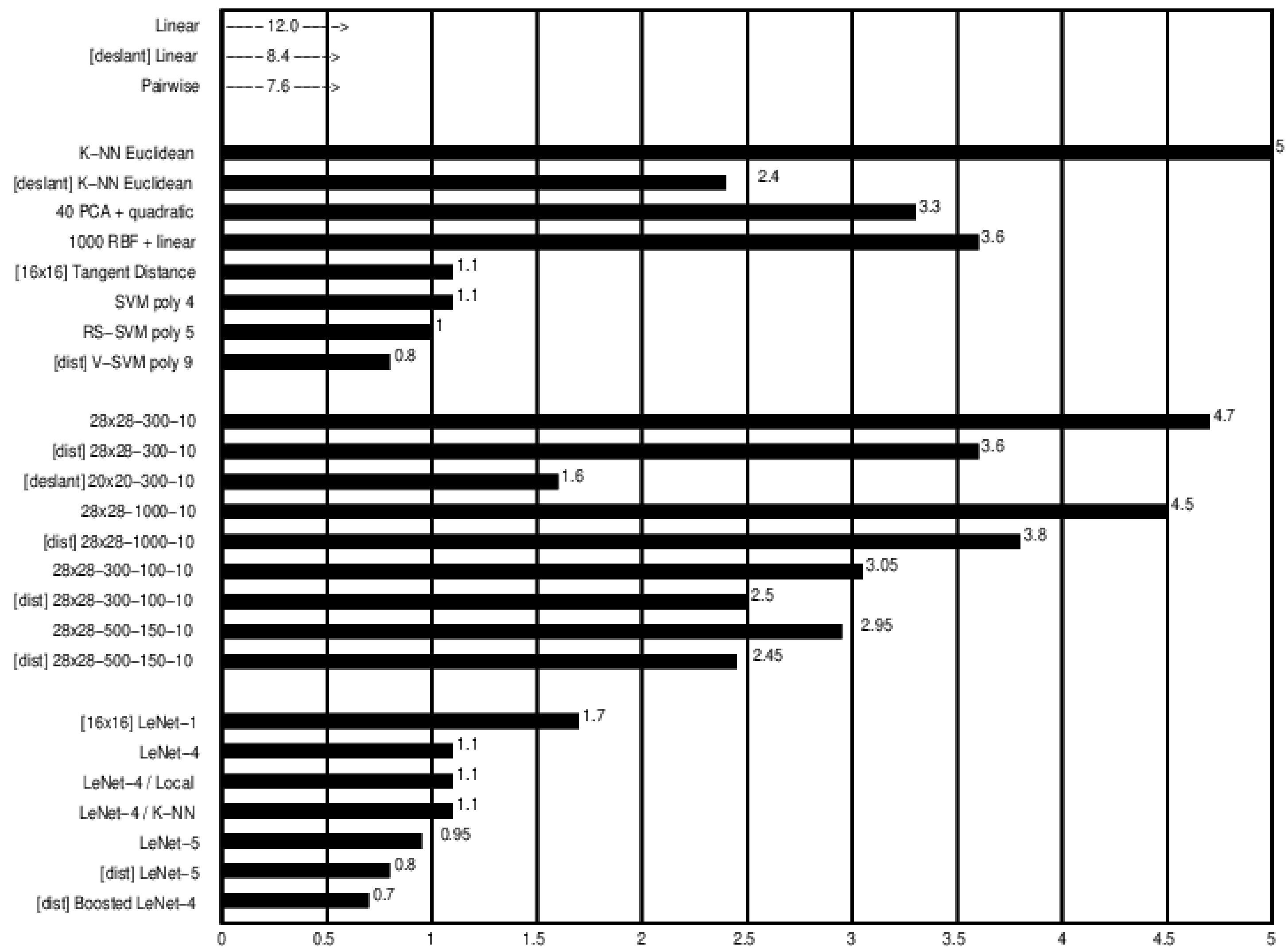


Fig. 9. Error rate on the test set (%) for various classification methods. [deslant] indicates that the classifier was trained and tested on the deslanted version of the database. [dist] indicates that the training set was augmented with artificially distorted examples. [16x16] indicates that the system used the 16x16 pixel images. The uncertainty in the quoted error rates is about 0.1%.

example recognition

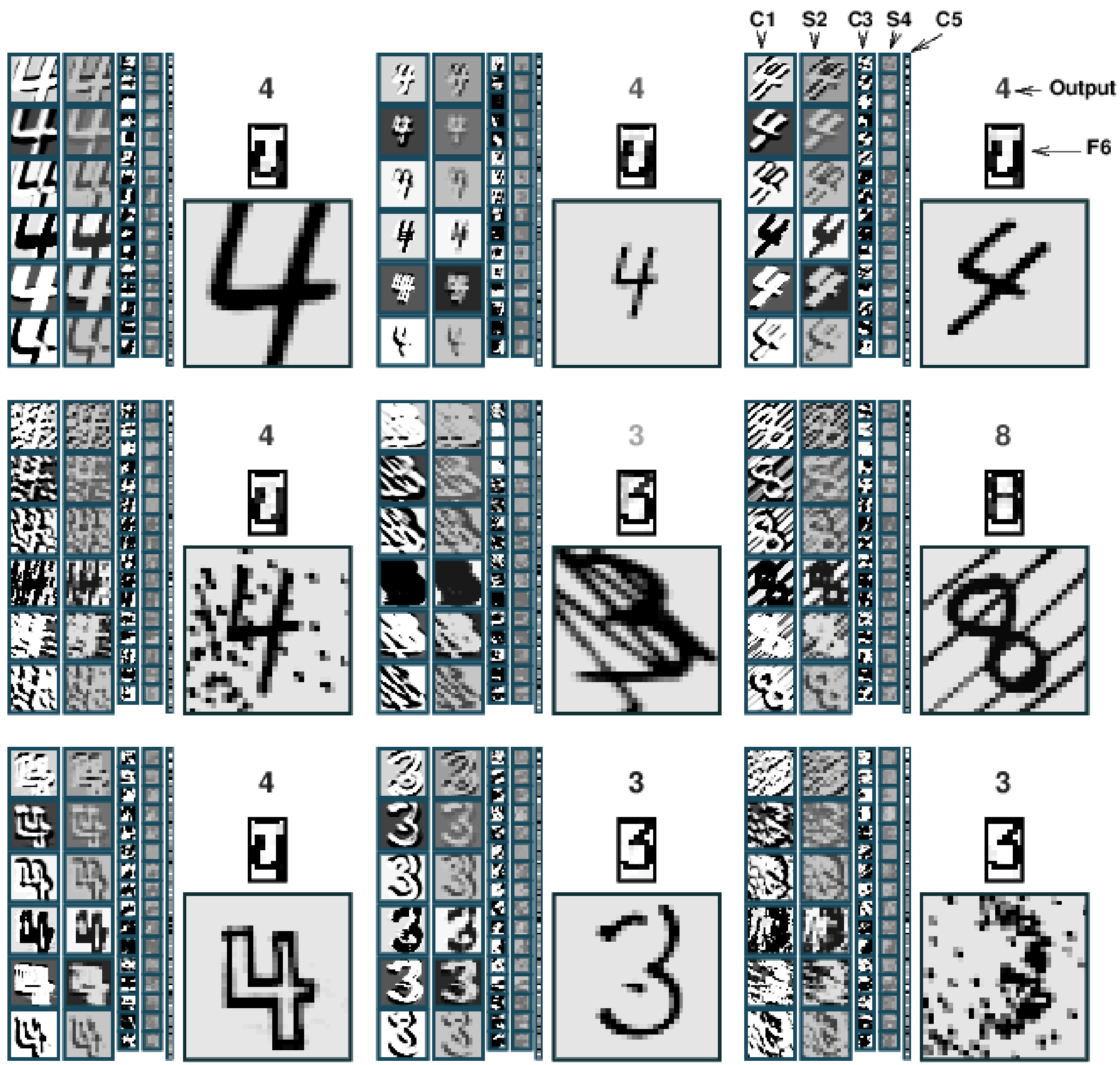
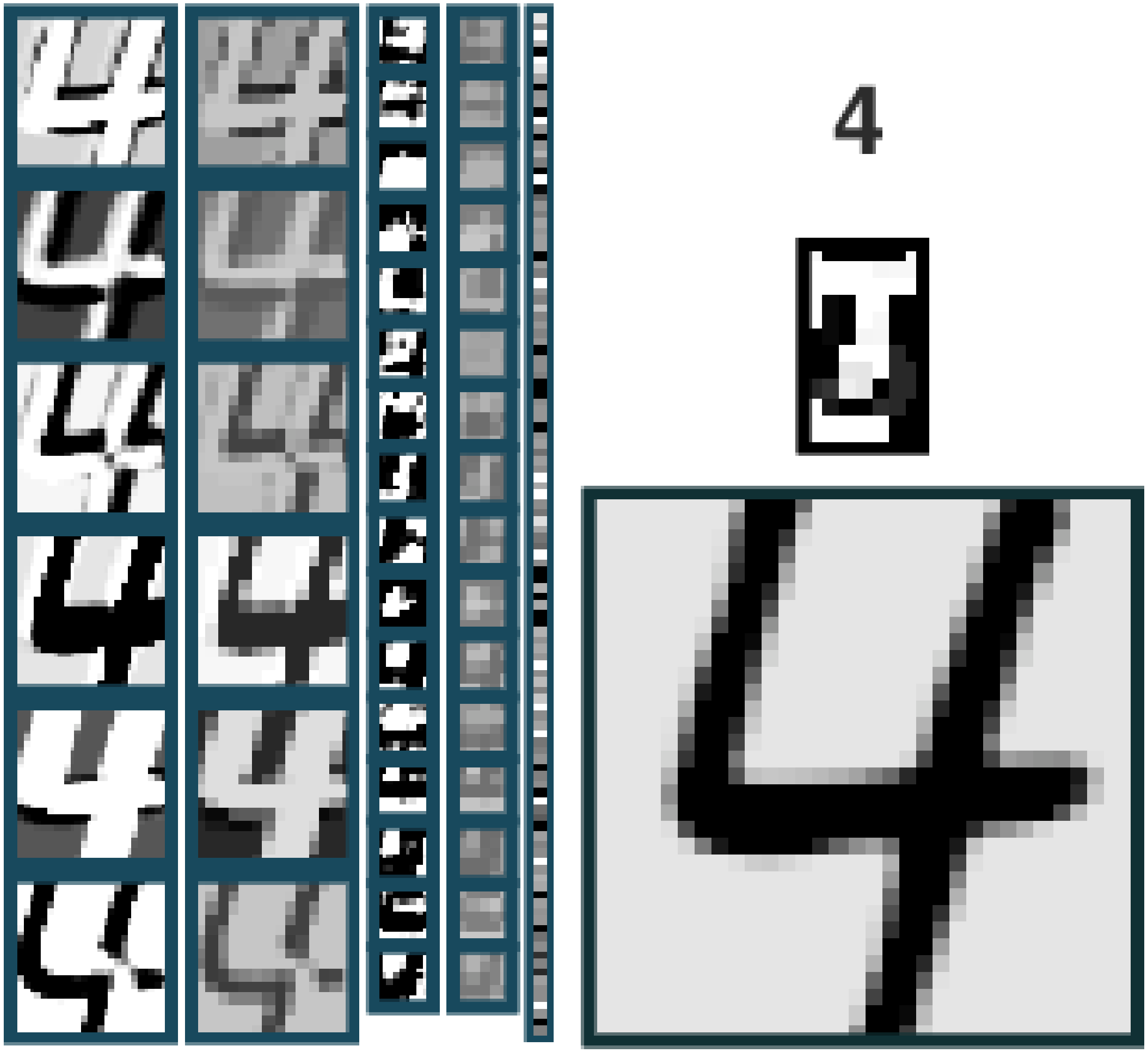


Fig. 13. Examples of unusual, distorted, and noisy characters correctly recognized by LeNet-5. The grey-level of the output label represents the penalty (lighter for higher penalties).

hidden layers



interpretation of hidden layers

- each “hidden layer” consists of multiple feature maps
- each feature map computes some property at each location
- a feature map = collection of neurons with similar receptive fields, but in different locations

multiple outputs?

- each input contains multiple characters in unknown locations
- the output is a string of transcriptions
- how do they get aligned?

graph transformer networks

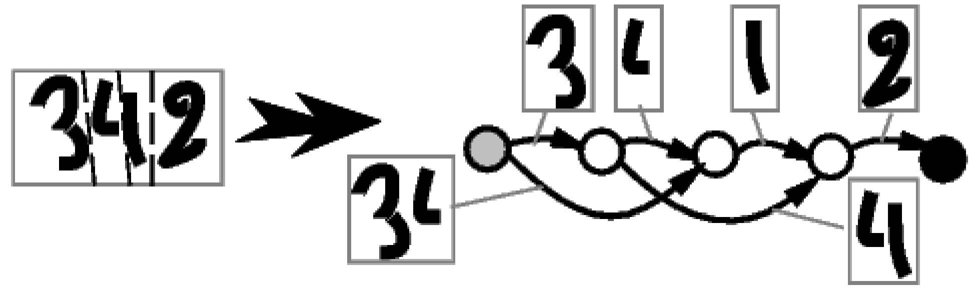


Fig. 16. Building a segmentation graph with Heuristic Over-Segmentation.

- **when recognizing multiple characters...**
  - image needs to be (over-)segmented
  - Viterbi algorithm needs to pick out the best interpretation
- **how do we train?**
  - although the segmentation graph is a discrete structure that depends on the input, we can still back-propagate through it

space displacement neural network

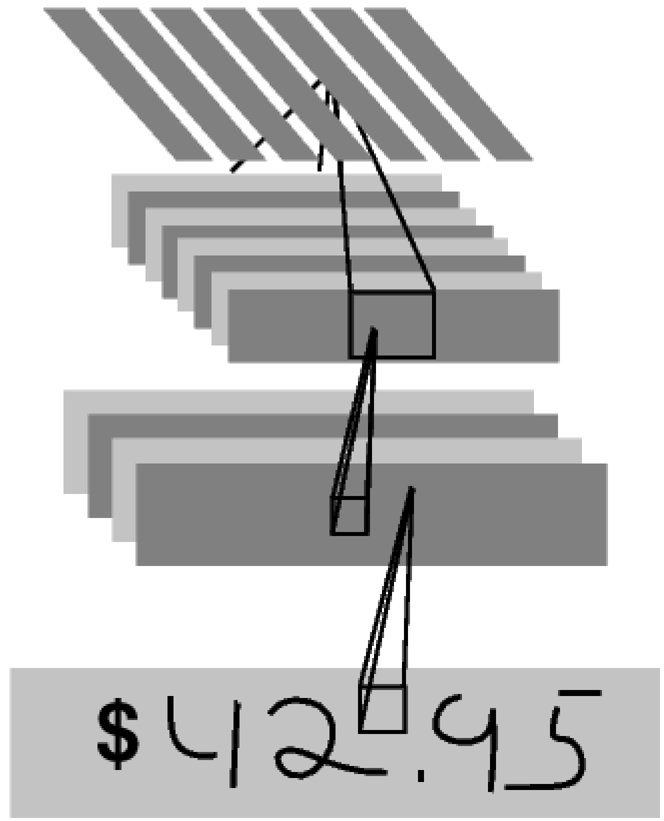


Fig. 23. A Space Displacement Neural Network is a convolutional network that has been replicated over a wide input field.

context + segmentation

- neural systems have extensive backwards connections for “attention” and “modulation”
- we will return to these in a later lecture
- in LeNet-5, graph transformer networks serve some of the same functions

questions

- **"less hand designed heuristics"?**
  - why are there so many layers?
  - what effect do the individual design decisions have?
  - what procedure do you use to apply this to other problems?
- **do the feature detectors do what the authors claim they do? how could you test?**
- **is this a realistic model for recognition?**
- **what properties does/does it not have?**

APPLICATION TO OBJECT RECOGNITION

Learning Methods for Generic Object Recognition with Invariance to Pose and Lighting

Yann LeCun, Fu Jie Huang,
The Courant Institute, New York University
715 Broadway, New York, NY 10003, USA
http://yann.lecun.com

Léon Bottou
NEC Labs America,
4 Independence Way, Princeton, NJ 08540
http://leon.bottou.org

Abstract

We assess the applicability of several popular learning methods for the problem of recognizing generic visual categories with invariance to pose, lighting, and surrounding clutter. A large dataset comprising stereo image pairs of 50 uniform-colored toys under 36 azimuths, 9 elevations, and 6 lighting conditions was collected (for a total of 194,400 individual images). The objects were 10 instances of 5 generic categories: four-legged animals, human figures, airplanes, trucks, and cars. Five instances of each category were used for training, and the other five for testing. Low-resolution grayscale images of the objects with various amounts of variability and surrounding clutter were used for training and testing. Nearest Neighbor methods, Support Vector Machines, and Convolutional Networks, operating on raw pixels or on PCA-derived features were tested. Test error rates for unseen object instances placed on uniform backgrounds were around 13% for SVM and 7% for Convolutional Nets. On a segmentation/recognition task with highly cluttered images, SVM proved impractical, while Convolutional nets yielded 16/7% error. A real-time version of the system was implemented that can detect and classify objects in natural scenes at around 10 frames per second.

idea

- **apply similar techniques as in LeNet-5 to visual object recognition**
- **approach**
  - get a dataset of objects with multiple views of each object
  - train a LeNet-like architecture
  - compare with other approaches
  - implement in real time

dataset

- 50 uniformly colored toys
- 36 azimuths, 9 elevations, 6 light. conditions
- 10 instances of 5 generic categories
- stereo images

- 5 instances for training, 5 for testing

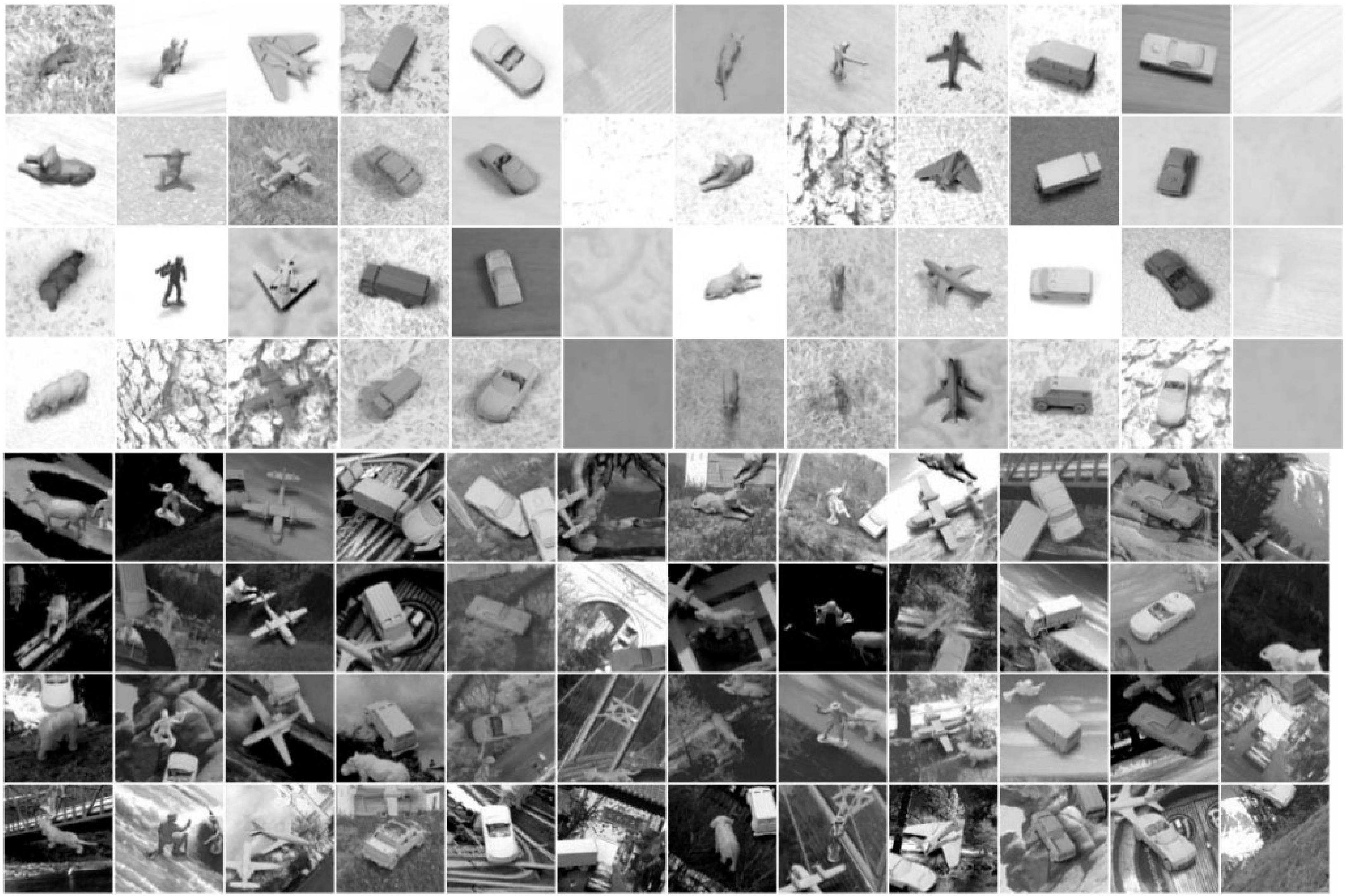
algorithms tested

- nearest neighbor
- support vector machines
- convolutional neural networks

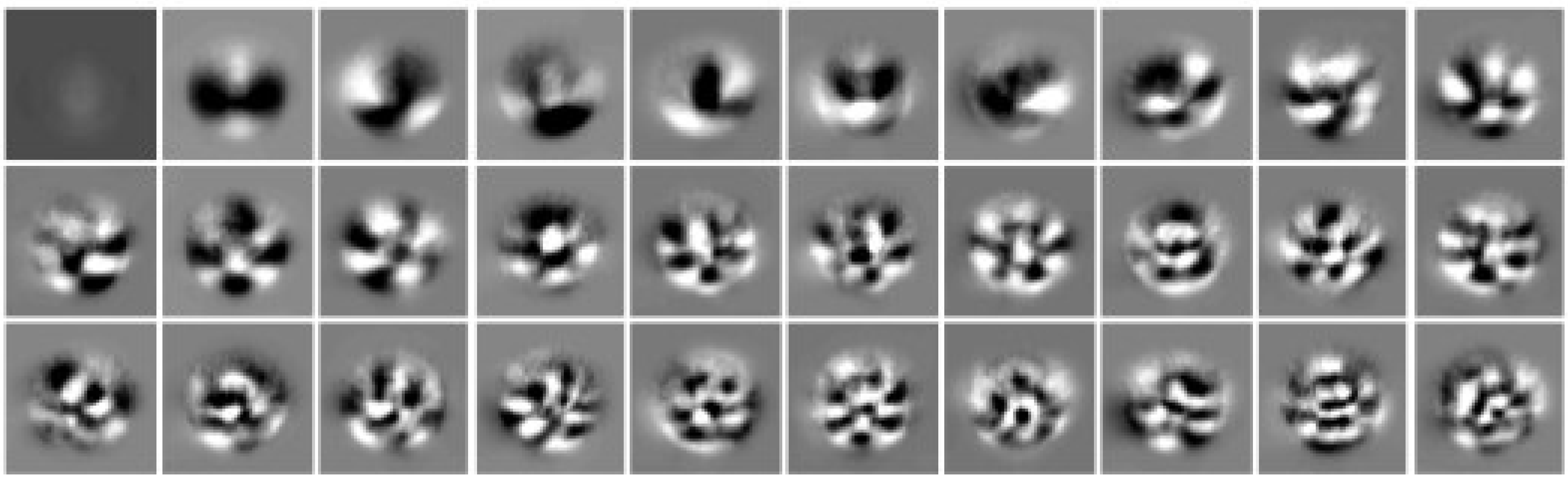
- raw pixels or PCA

data from NORB

jittered – cluttered training set



PCA components



results

Classification				
exp#	Classifier	Input	Dataset	Test Error
1.0	Linear	raw 2x96x96	norm-unif	30.2%
1.1	K-NN (K=1)	raw 2x96x96	norm-unif	18.4 %
1.2	K-NN (K=1)	PCA 95	norm-unif	16.6%
1.3	SVM Gauss	raw 2x96x96	norm-unif	N.C.
1.4	SVM Gauss	raw 1x48x48	norm-unif	13.9%
1.5	SVM Gauss	raw 1x32x32	norm-unif	12.6%
1.6	SVM Gauss	PCA 95	norm-unif	13.3%
1.7	Conv Net 80	raw 2x96x96	norm-unif	6.6%
1.8	Conv Net 100	raw 2x96x96	norm-unif	6.8%
2.0	Linear	raw 2x96x96	jitt-unif	30.6%
2.1	Conv Net 100	raw 2x96x96	jitt-unif	7.1%

Detection/Segmentation/Recognition				
exp#	Classifier	Input	Dataset	Test Error
5.1	Conv Net 100	raw 2x96x96	jitt-text	10.6%
6.0	Conv Net 100	raw 2x96x96	jitt-clutt	16.7%
6.2	Conv Net 100	raw 1x96x96	jitt-clutt	39.9%

sample results

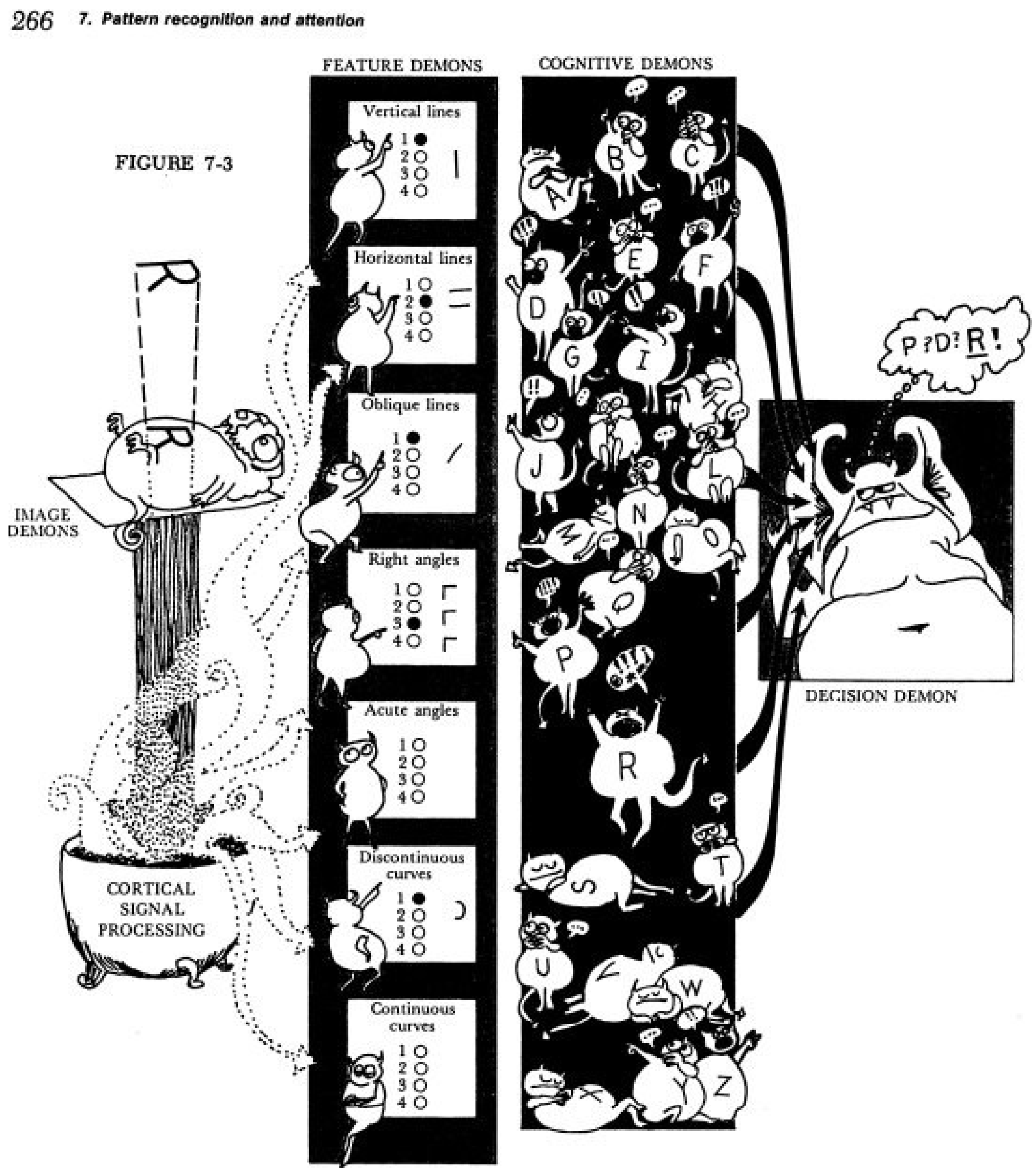
HMAX model

Hierarchical models of object recognition in cortex

Maximilian Riesenhuber and Tomaso Poggio

Visual processing in cortex is classically modeled as a hierarchy of increasingly sophisticated representations, naturally extending the model of simple to complex cells of Hubel and Wiesel. Surprisingly, little quantitative modeling has been done to explore the biological feasibility of this class of models to explain aspects of higher-level visual processing such as object recognition. We describe a new hierarchical model consistent with physiological data from inferotemporal cortex that accounts for this complex visual task and makes testable predictions. The model is based on a MAX-like operation applied to inputs to certain cortical neurons that may have a general role in cortical function.

Pandemonium (Selfridge, 1959)



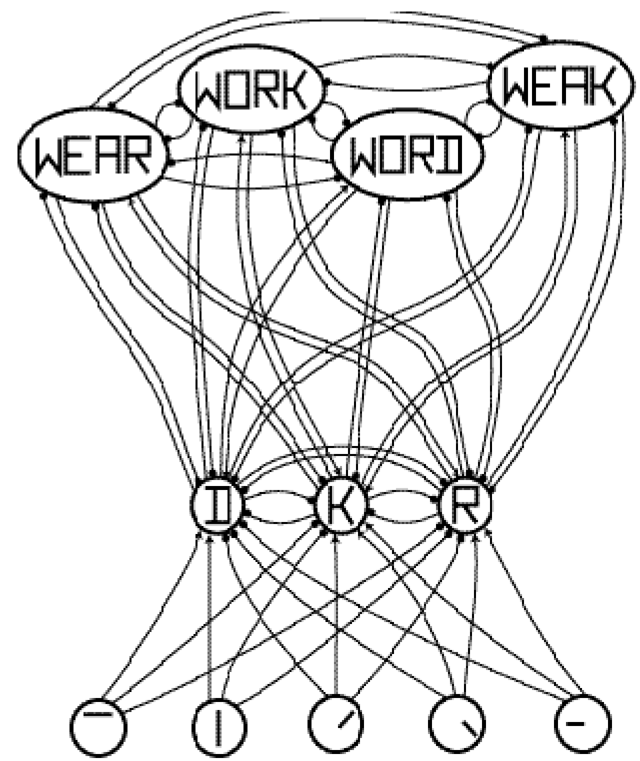
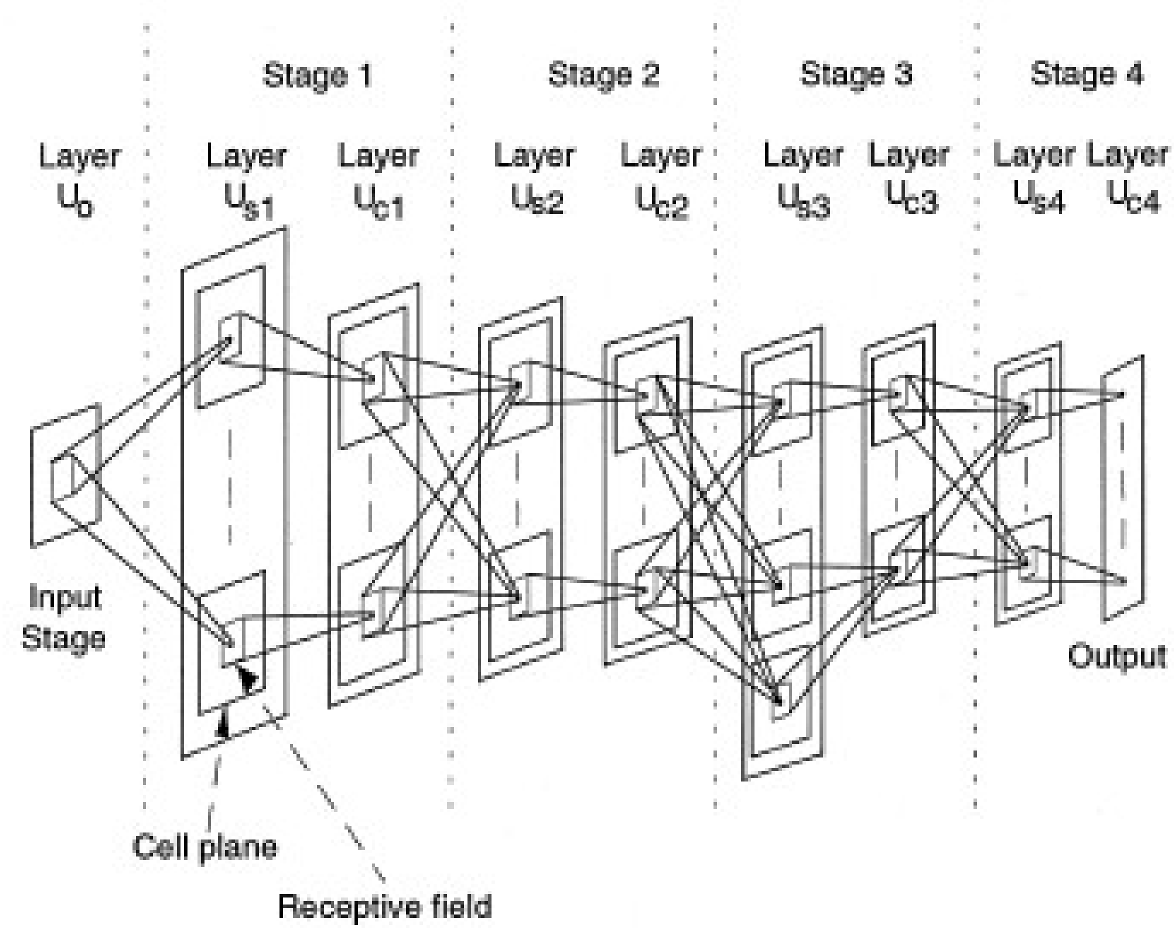


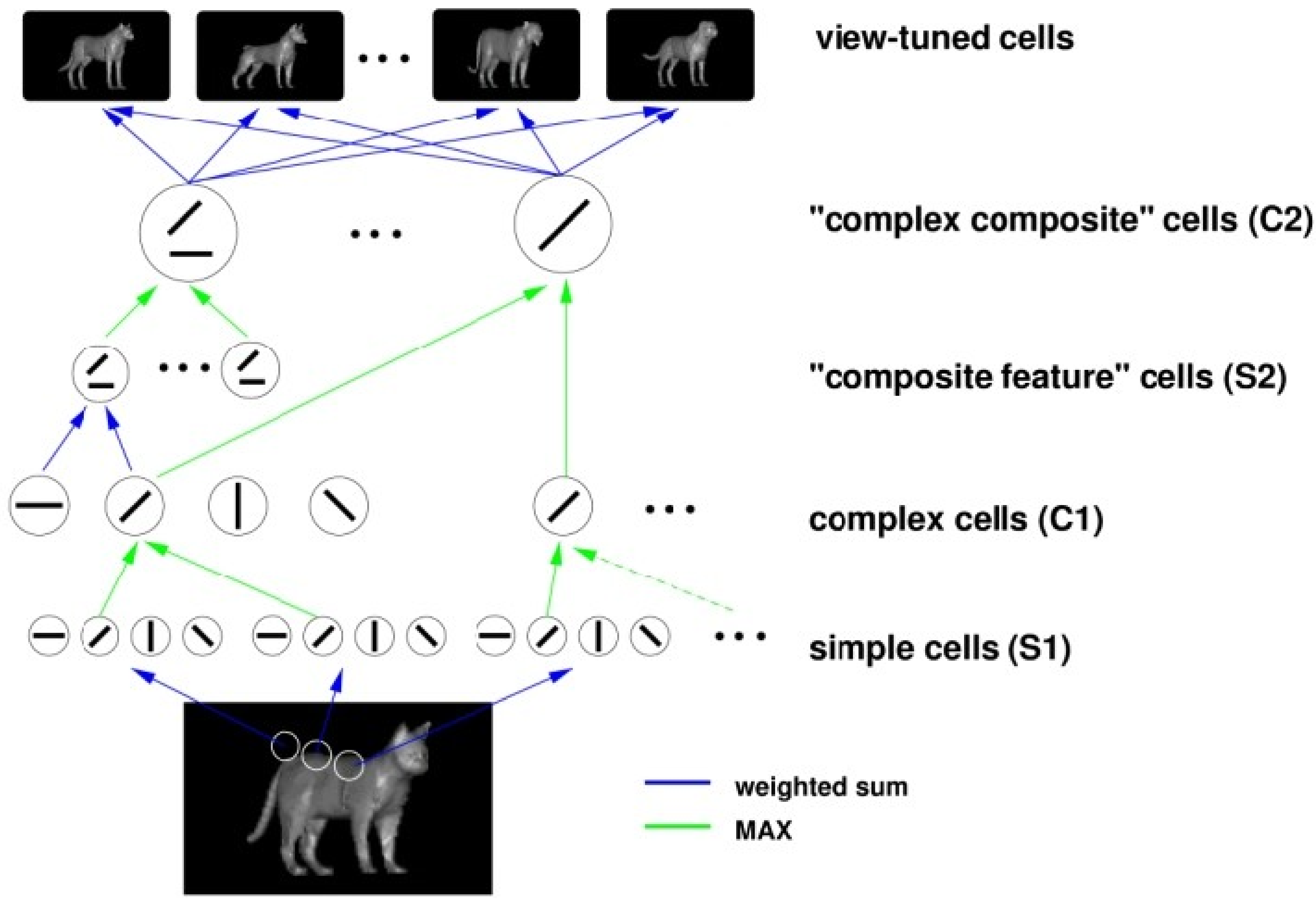
Figure 1. Interactive Activation Network Model (after McClelland and Rumelhart, 1981).

Neocognitron (Fukushima, 1980)

Figure 1
The architecture of Neocognitron



HMAX (Riesenhuber&Poggio, 1999)



notes

- **hierarchy of feature detectors is classical**
- **their paper...**
  - makes the model concrete
  - quantitatively consistent with biology
  - evaluates it

quantitative observations

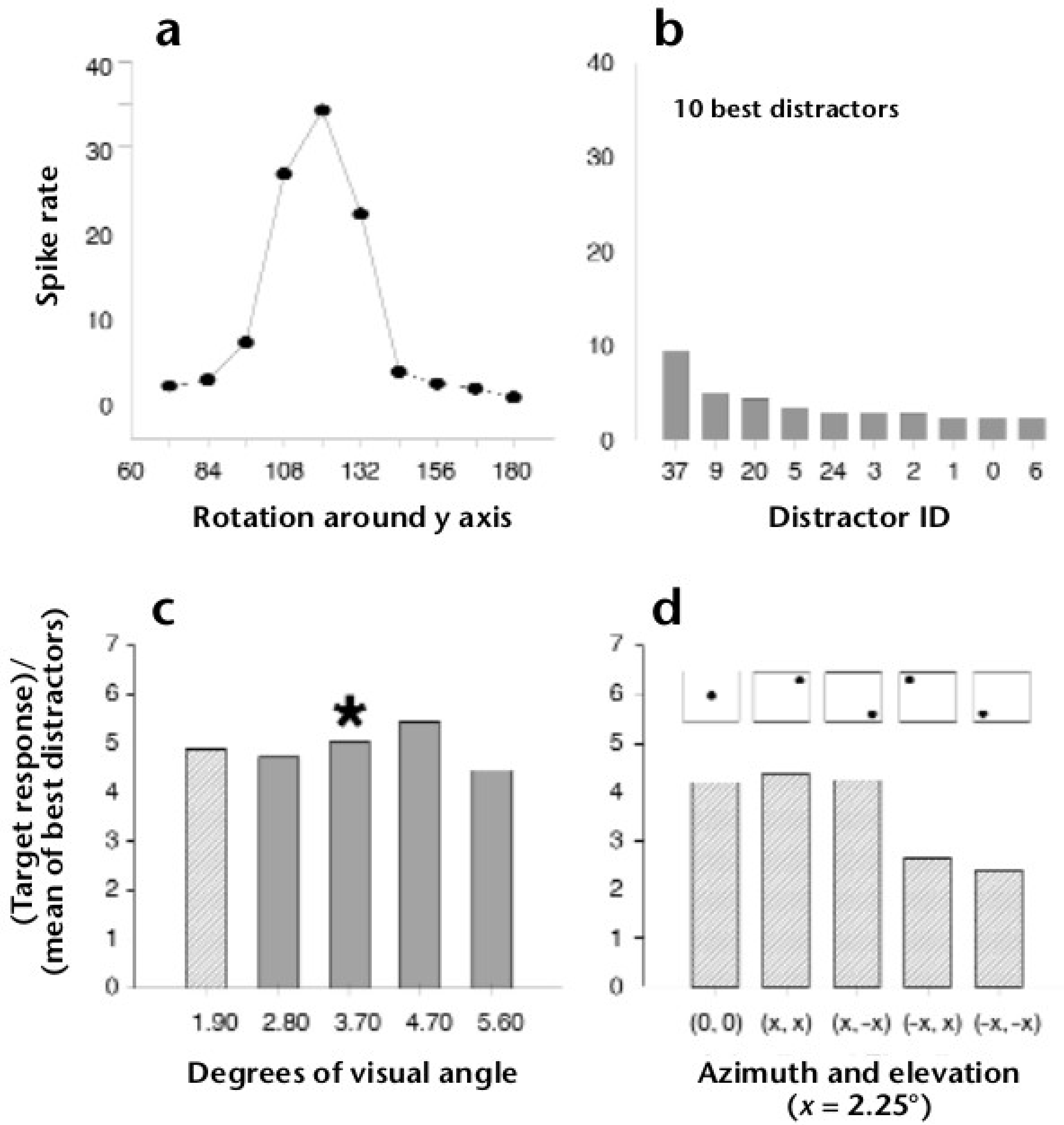


Fig. 1. Invariance properties of one neuron (modified from Logothetis *et al.*[21]). The figure shows the response of a single cell found in anterior IT after training the monkey to recognize paperclip-like objects. The cell responded selectively to one view of a paperclip and showed limited invariance around the training view to rotation in depth, along with significant invariance to translation and size changes, even though the monkey had only seen the stimulus at one position and scale during training. (**a**) Response of the cell to rotation in depth around the preferred view. (**b**) Cell's response to the ten distractor objects (other paperclips) that evoked the strongest responses. The lower plots (**c**, **d**) show the cell's response to changes in stimulus size (asterisk shows the size of the training view) and position (using the 1.9° size), respectively, relative to the mean of the ten best distractors. Defining 'invariance' as yielding a higher response to transformed views of the preferred stimulus than to distractor objects, neurons showed an average rotation invariance of 42° (during training, stimuli were actually rotated by ±15° in depth to provide full 3D information to the monkey; therefore, the invariance obtained from a single view is probably smaller), translation and scale invariance on the order of ±2° and ±1 octave around the training view, respectively (J. Pauls, personal communication).

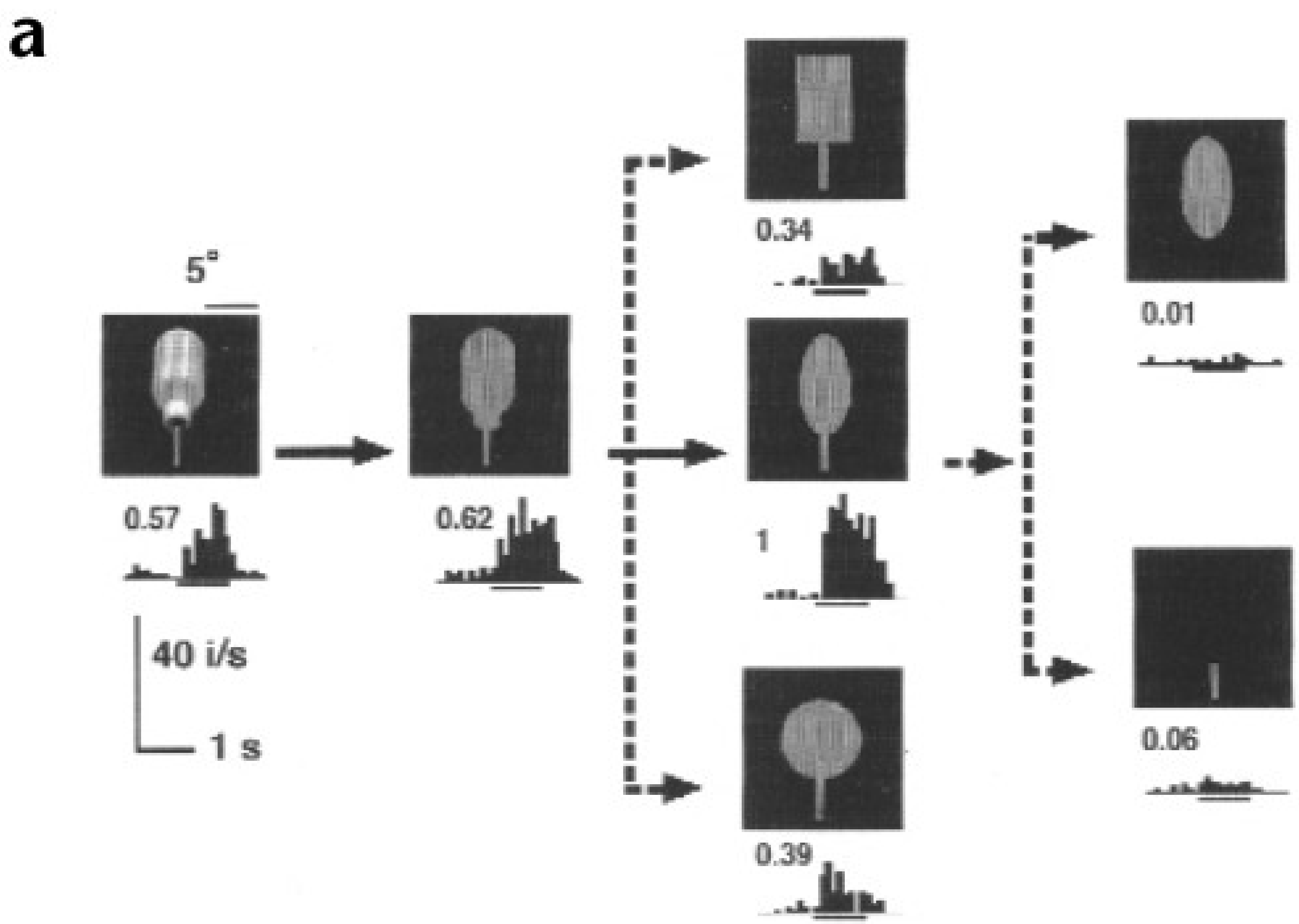
observations

- **separate**
  - feature specificity → template matching to features
  - invariance → "pooling" of responses
- **possible pooling operation**
  - linear SUM with equal weights (isotropic response)
  - non-linear SUM equal weights and thresholds
  - MAX

problems with using SUM

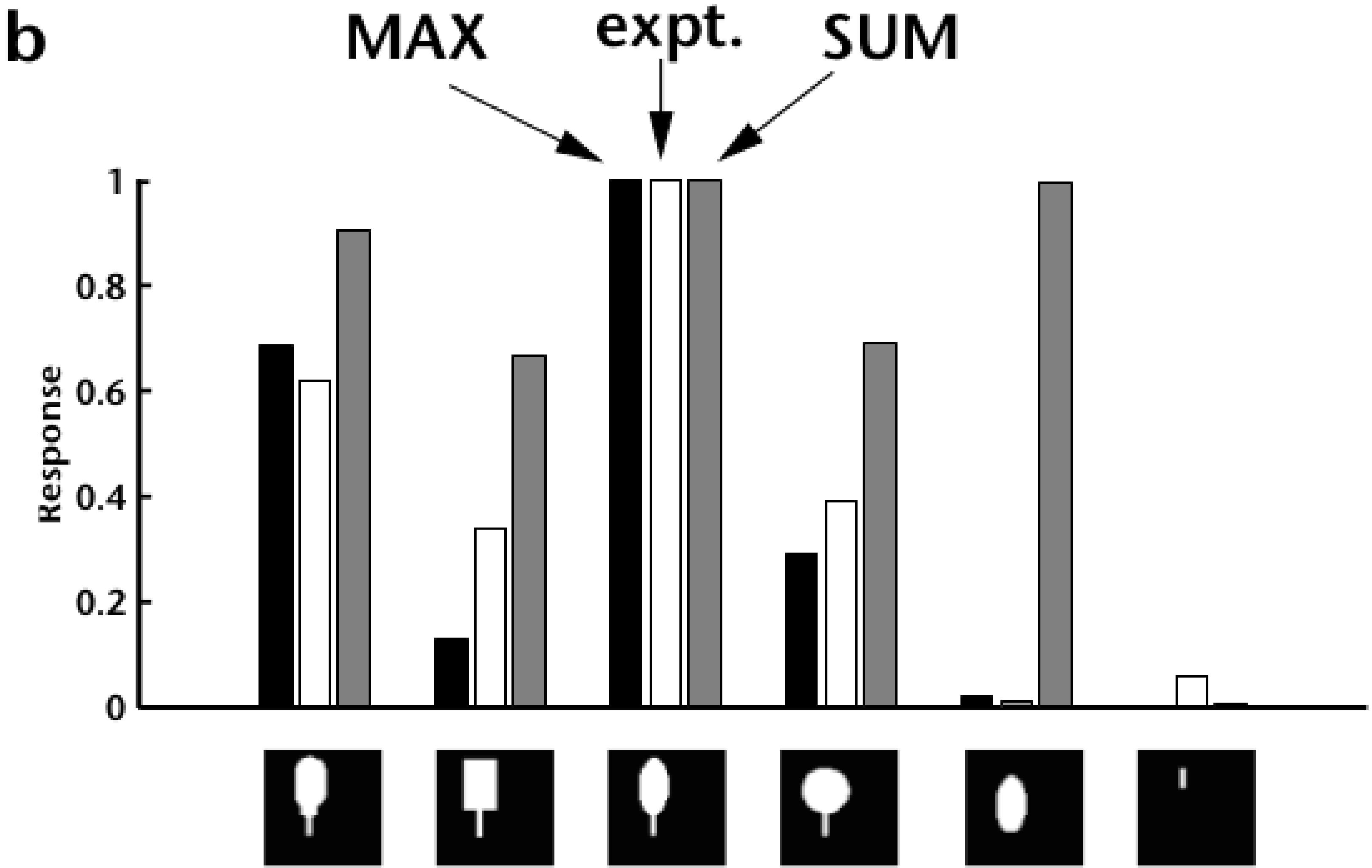
- **simple SUM = problems with size invariance**
- **SUM + nonlinear = need to learn threshold**
  - NB: that's what LeNet does as well

experimental results

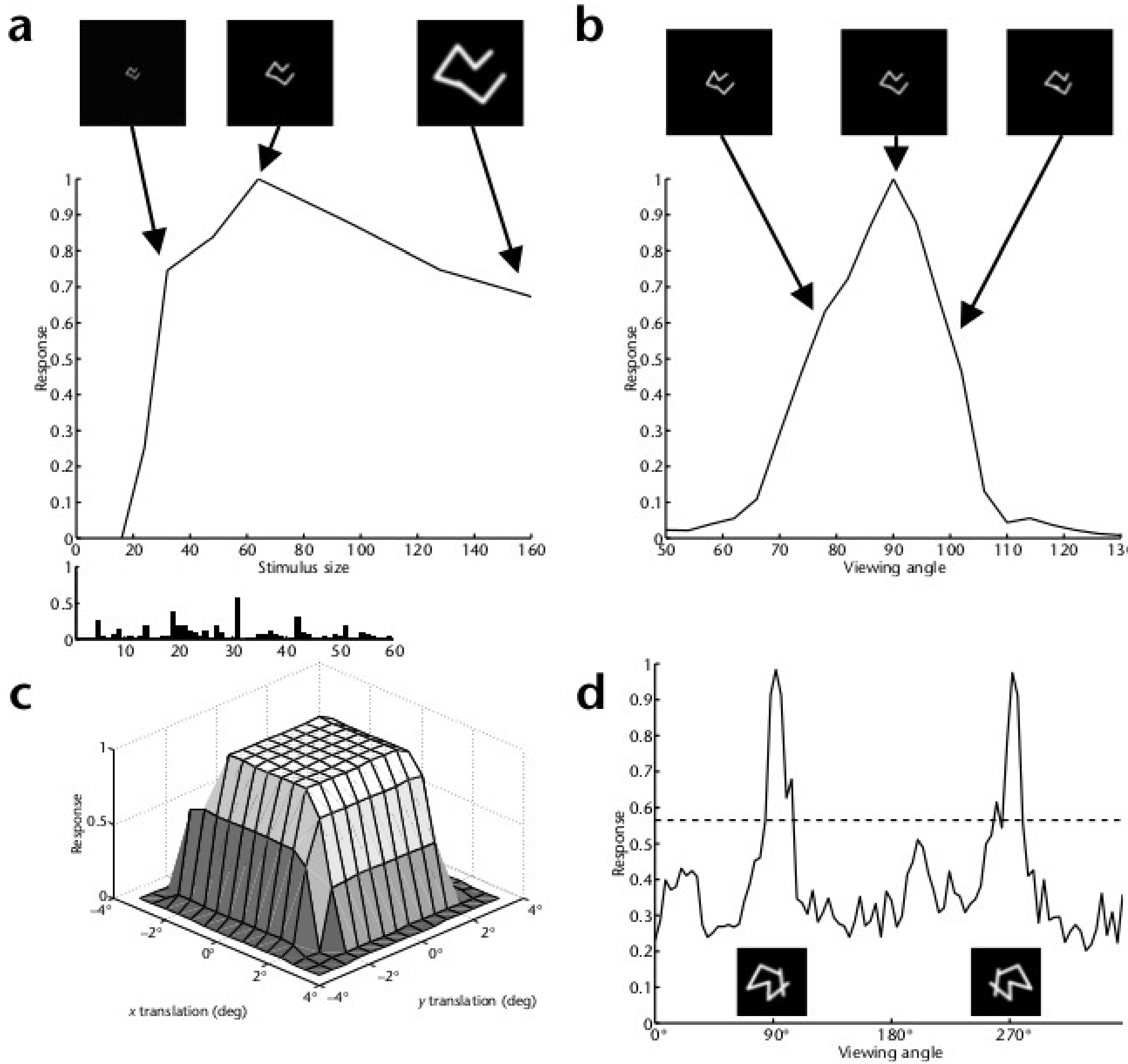


variants of object input generated via "simplification procedure"

simulated and real responses



simulated 3D responses



questions / issues

- **paper gives no comparison...**
  - with LeNet
  - alternative methods
- **i.e. does agreement with experiment show anything?**
- **unrealistic stimuli**
- **no modeling of feedback connections**
- **no clutter, no occlusions**
- **more work needed...**

TESTING PERFORMANCE

Comparing State-of-the-Art Visual Features on Invariant Object Recognition Tasks

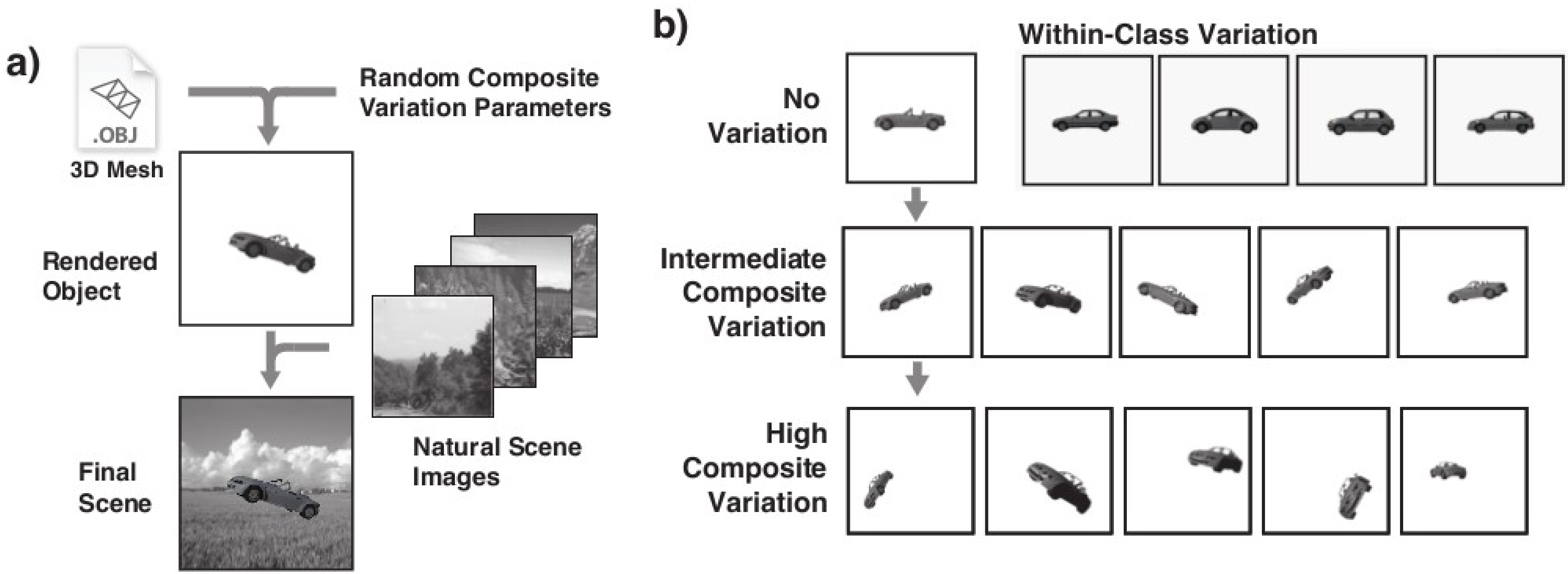
Nicolas Pinto[1], Youssef Barhomi[1], David D. Cox[2], and James J. DiCarlo[1]

[1] *Massachusetts Institute of Technology, Cambridge, MA, U.S.A*
[2] *The Rowland Institute at Harvard, Cambridge, MA, U.S.A*

Abstract

Tolerance ("invariance") to identity-preserving image variation (e.g. variation in position, scale, pose, illumination) is a fundamental problem that any visual object recognition system, biological or engineered, must solve. While standard natural image database benchmarks are useful for guiding progress in computer vision, they can fail to probe the ability of a recognition system to solve the invariance problem [23, 24, 25]. Thus, to understand which computational approaches are making progress on solving the invariance problem, we compared and contrasted a variety of state-of-the-art visual representations using synthetic recognition tasks designed to systematically probe invari-

data set generation



descriptors

- scale invariant feature transform (SIFT)
- pyramid histogram of visual words (PHOW)
- pyramid histogram of gradients (PHOG)
- geometric blur
- sparse localized features (SLF, HMAX++)

classification

- L2-regularized SVM
- Shogun toolbox
- 150 training + 150 testing

performance