9.250 in 2009

Statistical Learning Theory
and
Applications

Lorenzo Rosasco + Jake Bouvrie +
Ryan Rifkin + Charlie Frogner + Tomaso Poggio
McGovern Institute for Brain Research
Center for Biological and Computational Learning
Department of Brain & Cognitive Sciences
Massachusetts Institute of Technology
Cambridge, MA 02139 USA

Learning: Brains and Machines

Learning is the gateway to understanding the brain and to making intelligent machines.

Problem of learning:
a focus for

- modern math
- computer algorithms
- neuroscience

Learning: much more than memory

- Role of **learning (**theory and applications in many different domains) has grown substantially in CS: learning+statistics is becoming a *lingua franca* in CS

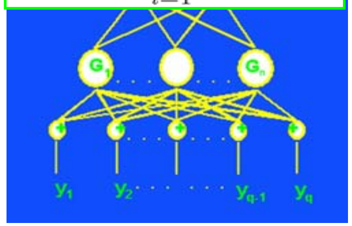
- Plasticity and learning increasingly have a central stage in the neurosciences

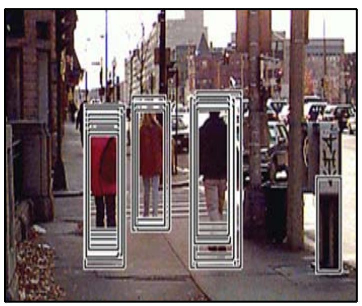
- Until now math and engineering of learning has developed independently of neuroscience…but it may begin to change as we will see in the class.

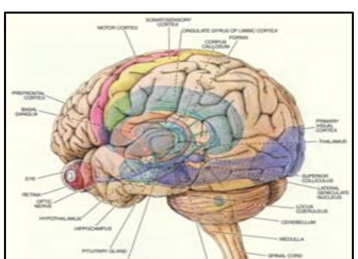
Learning:
math, engineering, neuroscience

$$\min_{f \in H}\left[\frac{1}{\ell}\sum_{i=1}^{\ell} V(y_i, f(x_i)) + \mu \|f\|_K^2\right]$$

$$f(x) = \sum_{i=1}^{l} c_i K(\mathbf{x}_i, \mathbf{x})$$







Learning theory + algorithms

Theorems on foundations of learning:

Predictive algorithms

ENGINEERING APPLICATIONS

- Bioinformatics
- Computer vision
- Computer graphics, speech synthesis, creating a virtual actor

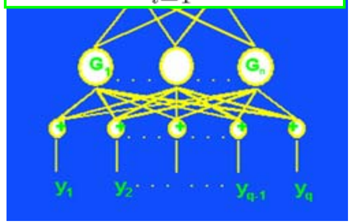
Computational Neuroscience: models+experiments

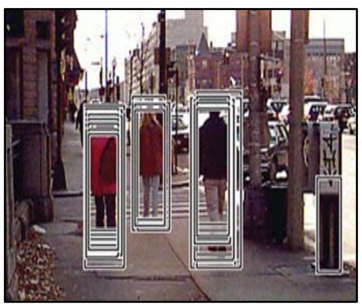
How visual cortex works – and how it may suggest better computer vision systems

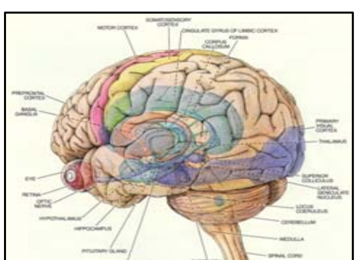
Learning: math, engineering, neuroscience *(now)*

$$\min_{f \in H} \left[\frac{1}{\ell} \sum_{i=1}^{\ell} V(y_i, f(x_i)) + \mu \|f\|_K^2 \right]$$

$$f(x) = \sum_{i=1}^{l} c_i K(\mathbf{x}_i, \mathbf{x})$$







Learning theory + algorithms

Theorems on foundations of learning

Predictive algorithms (Regularization networks ~ SMS…)

ENGINEERING APPLICATIONS

- Bioinformatics
- Computer vision
- Computer graphics, speech synthesis
- Speech recognition

Computational Neuroscience: models+experiments

How visual cortex works:

Deep Learning in Cortex

Math Camp? Look at old Mathcamps on Web site: we will decide on Monday

Functional Analysis:

Linear and Euclidean spaces
scalar product, orthogonality
orthonormal bases, norms and semi-norms,
Cauchy sequence and complete spaces
Hilbert spaces, function spaces
and linear functional, Riesz representation
theorem, convex functions, functional calculus.

Linear Algebra

Basic notion and definitions: matrix and
vectors norms, positive, symmetric,
invertible matrices, linear systems,
condition number.

& Multivariate Calculus:

Extremal problems, differential, gradient.

Probability Theory:

Random Variables (and related
concepts), Law of Large Numbers,
Probabilistic Convergence,
Concentration Inequalities.

9.520 Statistical Learning Theory and Applications (2007)

Class 26: Project presentations (past examples)

10:30 - Simon Laflamme "Online Learning Algorithm for Structural Control using Magnetorheological Actuators"

- Emily Shen "Time series prediction"
- Zak Stone "Facebook project"
- Jeff Miller "Clustering features in the standard model of cortex"
- Manuel Rivas "Learning Age from Gene Expression Data"
- Demba Ba "Sparse Approximation of the Spectrogram via Matching Pursuits: Applications to Speech Analysis"
- Nikon Rasumov "Data mining in controlled environment and real data"

9.520 Statistical Learning Theory and Applications

Class 26: Project presentations (past examples)

2:35-2:50 "Learning card playing strategies with SVMs", David Craft and Timothy Chan

2:50-3:00 "Artificial Markets: Learning to trade using Support Vector Machines", Adlar Kim

3:00-3:10 "Feature selection: literature review and new development'', Wei Wu

3:10—3:25 "Man vs machines: A computational study on face detection" Thomas Serre

Project suggestions

We will provide some in the next few classes and we will speak more about them just before spring break

Overview of overview

- The problem of supervised learning: "real" math behind it
- Examples of engineering applications (from our group)
- Learning and the brain

Reason to learn some learning theory

Applications cannot be carried out by simply using a black box.

What is needed: the right formulation of the problem (which is helped by knowledge of theory): choice of representation (inputs, outputs), choice of examples, validate predictivity, do not datamine

$\ldots\ f(\mathbf{x}) = \mathbf{w}\mathbf{x} + b$

Notes

Two connected and overlapping strands in learning theory:

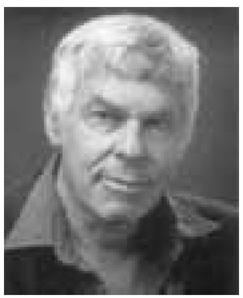
- Bayes, hierarchical models, graphical models…

- Statistical learning theory, regularization (closer to classical math, functional analysis+probability theory+empirical process theory…)

Interesting development:
in the last few years he theoretical foundations of learning have become part of mainstream mathematics:
L. Valiant,V. Vapnik, S. Smale, I. Daubechie <u>et al.</u>

BULLETIN (New Series) OF THE
AMERICAN MATHEMATICAL SOCIETY
Volume 39, Number 1, Pages 1–49
S 0273-0979(01)00923-5
Article electronically published on October 5, 2001

ON THE MATHEMATICAL FOUNDATIONS OF LEARNING



FELIPE CUCKER AND STEVE SMALE

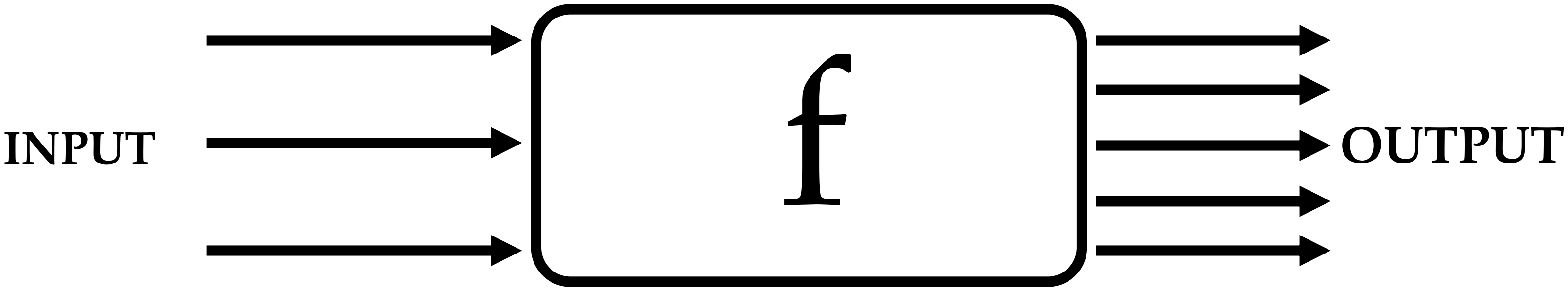
The problem of learning is arguably at the very core of the problem of intelligence, both bi

T. Poggio and C.R. Shelton

Introduction

(1) A main theme of this report is the relationship of approximation to learning and the primary role of sampling (inductive inference). We try to emphasize relations of the theory of learning to the mainstream of mathematics. In particular, there are large roles for probability theory, for algorithms such as *least squares*, and for tools and ideas from linear algebra and linear analysis. An advantage of doing this is that communication is facilitated and the power of core mathematics is more easily brought to bear.

Learning from examples: goal is not to memorize but to generalize, eg *predict*.



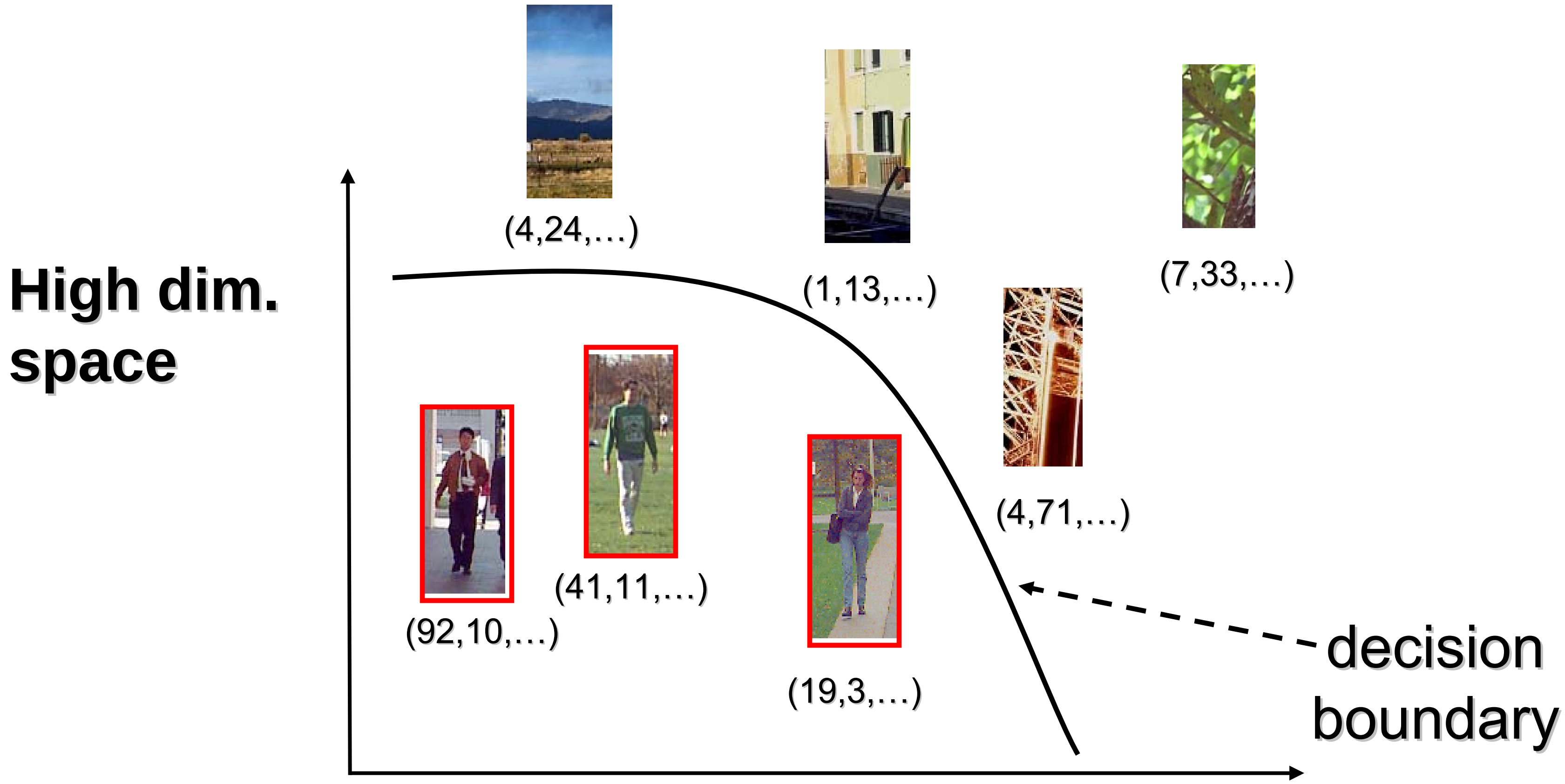
Given a set of ℓ examples (data) $\{(x_1, y_1), (x_2, y_2), \ldots, (x_\ell, y_\ell)\}$

Question: find function *f* such that

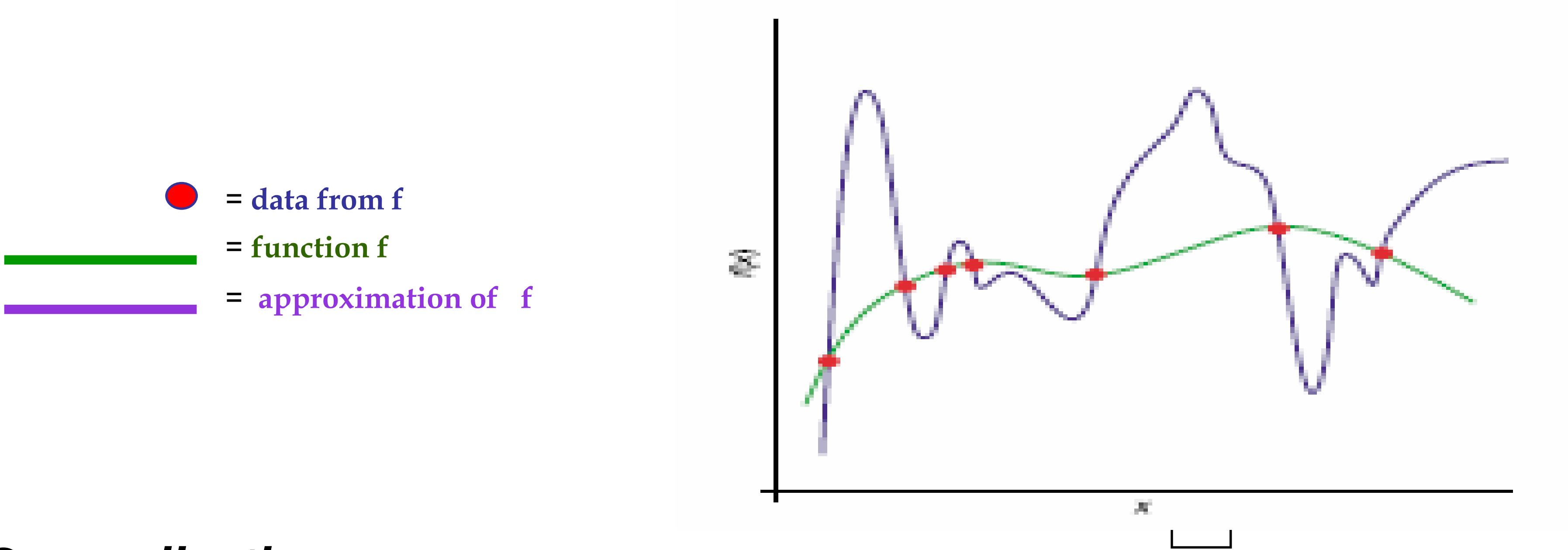
is a ***good predictor*** of *y* for a ***future*** input *x (fitting the data is **not** enough!):*

$$f(x) = \hat{y}$$

Binary classification case



Learning from examples: predictive, multivariate function regression from data (not just curve fitting)



***Generalization*:**
estimating value of function where there are no data (good generalization means predicting the function well; most important is for empirical or validation error to be a good proxy of the prediction error)

***Regression*: function is real valued**

***Classification*: function is binary**

The learning problem

There is an unknown **probability distribution** on the product space $Z = X \times Y$, written $\mu(z) = \mu(x, y)$. We assume that X is a compact domain in Euclidean space and Y a closed subset of $\mathbb{R}$.

The **training set** $S = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\} = \{z_1, ...z_n\}$ consists of n samples drawn i.i.d. from μ.

$\mathcal{H}$ is the **hypothesis space**, a space of functions $f : X \to Y$.

A **learning algorithm** is a map $L : Z^n \to \mathcal{H}$ that looks at S and selects from $\mathcal{H}$ a function $f_S : \mathbf{x} \to y$ such that $f_S(\mathbf{x}) \approx y$ *in a predictive way.*

A key requirement for learning: *generalization*

An informal definition of generalization is to find a model that predicts well **new** data

Example:

A prototypical algorithm is ERM (empirical risk minimization)

$$\min_{f \in \mathcal{H}} \frac{1}{\ell} \sum_{i=1}^{\ell} V(f(x_i), y_i)$$

What are the conditions ensuring generalization?

We need a more formal definition...

Definitions

Given a function f, a loss function V, and a probability distribution μ over Z, the **expected or true error** of f is:

$$I[f] = \mathbb{E}_z V[f, z] = \int_Z V(f, z) d\mu(z) \tag{1}$$

which is the **expected loss** on a new example drawn at random from μ.
The **empirical error** of f is:

$$I_S[f] = \frac{1}{n} \sum V(f, z_i) \tag{2}$$

A very natural requirement for f_S is distribution independent **generalization**

$$\forall \mu, \lim_{n \to \infty} |I_S[f_S] - I[f_S]| = 0 \textit{ in probability} \tag{3}$$

In other words, the training error for the solution must converge to the expected error and thus be a "proxy" for it. Otherwise the solution would not be "predictive".

Control of complexity

The problem does not have a *predictive* solution in general (just fitting the data does not work).

Choosing an appropriate hypothesis space *H* (for instance a compact set of continuous functions) can guarantee generalization (how good depends on the problem and other parameters) (Vapnik,...)

A superficially different requirement for learning is *well-posedness* of the associated optimization problem

J. S. Hadamard, 1865-1963

A problem is well-posed if its solution

exists, unique and

is stable, eg depends continuously on the data (here examples)

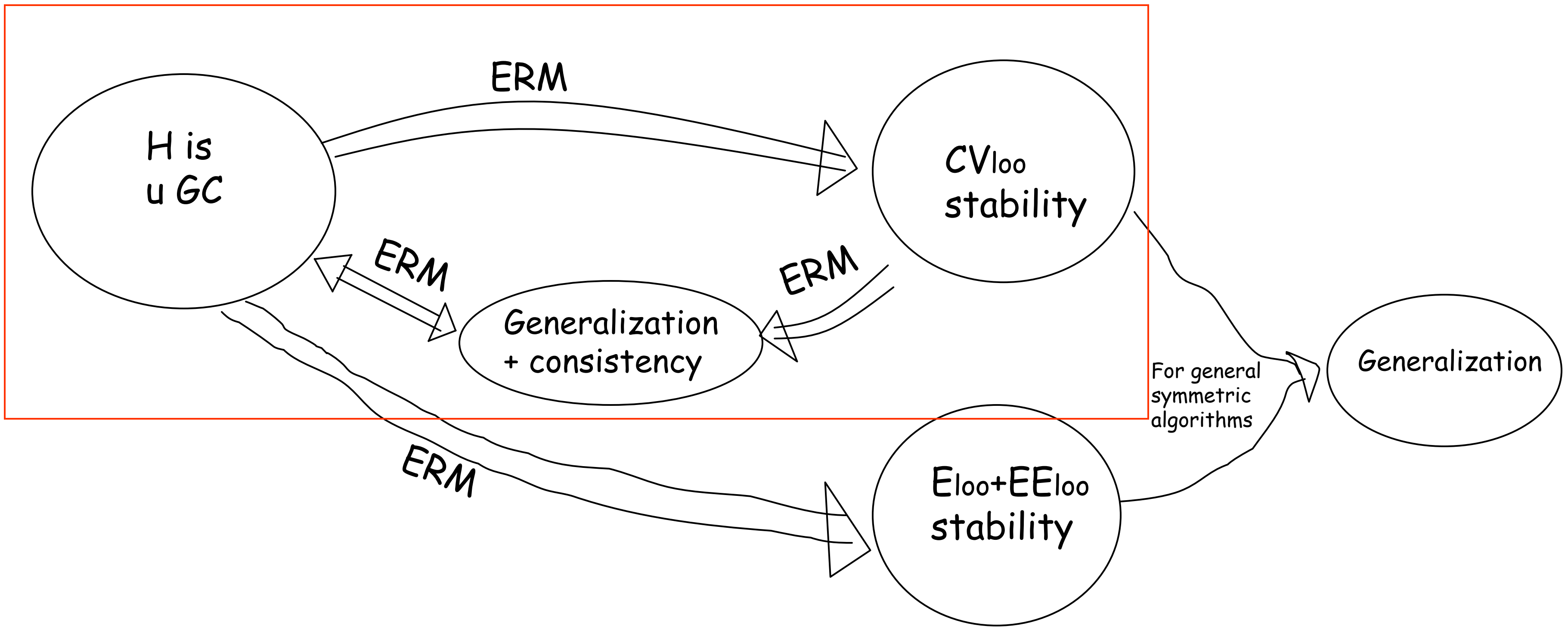
Well-posedness and *generalization: are they related?*

It is possible to show that under quite general assumptions

generalization and well-posedness are *equivalent*, eg one implies the other.

Stability implies generalization:

a stable solution is predictive and (for ERM) also viceversa.



Mukherjee, S., P. Niyogi, T. Poggio and R. Rifkin. Learning Theory: Stability is Sufficient for Generalization and Necessary and Sufficient for Consistency of Empirical Risk Minimization, *Advances in Computational Mathematics*, 25, 161-193, 2006; see also Nature, 2005

This is an example of foundational results
in learning theory...

Learning theory and natural sciences

Conditions for **generalization** in learning theory

have deep, almost philosophical, implications:

they may be regarded as conditions that guarantee a theory to be *predictive* (that is *scientific*)

Regularization in RKHS: a simple algorithm which generalizes well (it is *uniformly stable*) and is computationally tractable

$$\min_{f \in H} \left[\frac{1}{\ell} \sum_{i=1}^{\ell} V(f(x_i) - y_i) + \lambda \, \|f\|_K^2 \right]$$

implies

$$f(\mathbf{x}) = \sum_i^l \alpha_i K(\mathbf{x}, \mathbf{x}_i)$$

Equation includes splines, Radial Basis Functions and Support Vector Machines (depending on choice of V).

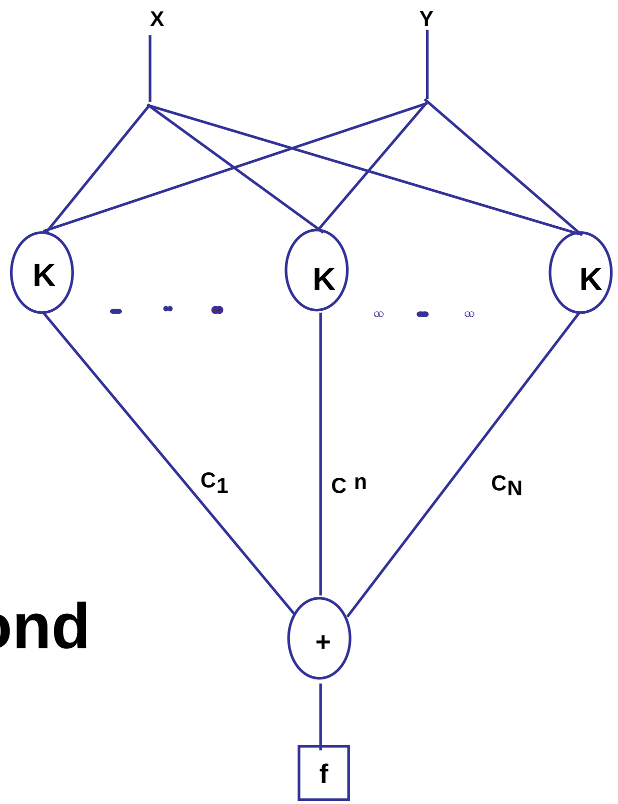
RKHS were explicitly introduced in learning theory by Girosi (1997), Vapnik (1998). Moody and Darken (1989), and Broomhead and Lowe (1988) introduced RBF to learning theory. Poggio and Girosi (1989) introduced Tikhonov regularization in learning theory and worked (implicitly) with RKHS. RKHS were used earlier in approximation theory (eg Parzen, 1952-1970, Wahba, 1990).

*For a review, see Poggio and Smale, **The Mathematics of Learning**, Notices of the AMS, 2003*

“Classical” kernel machines are equivalent to (shallow) networks

Kernel machines…

$$f(\mathbf{x}) = \sum_i^l c_i K(\mathbf{x}, \mathbf{x}_i) + b$$



can be “written” as shallow networks (for many different V): the value of K corresponds to the “activity” of the “unit” for the input and the c_i correspond to “weights”

Winning against the curse of dimensionality: new research directions in learning

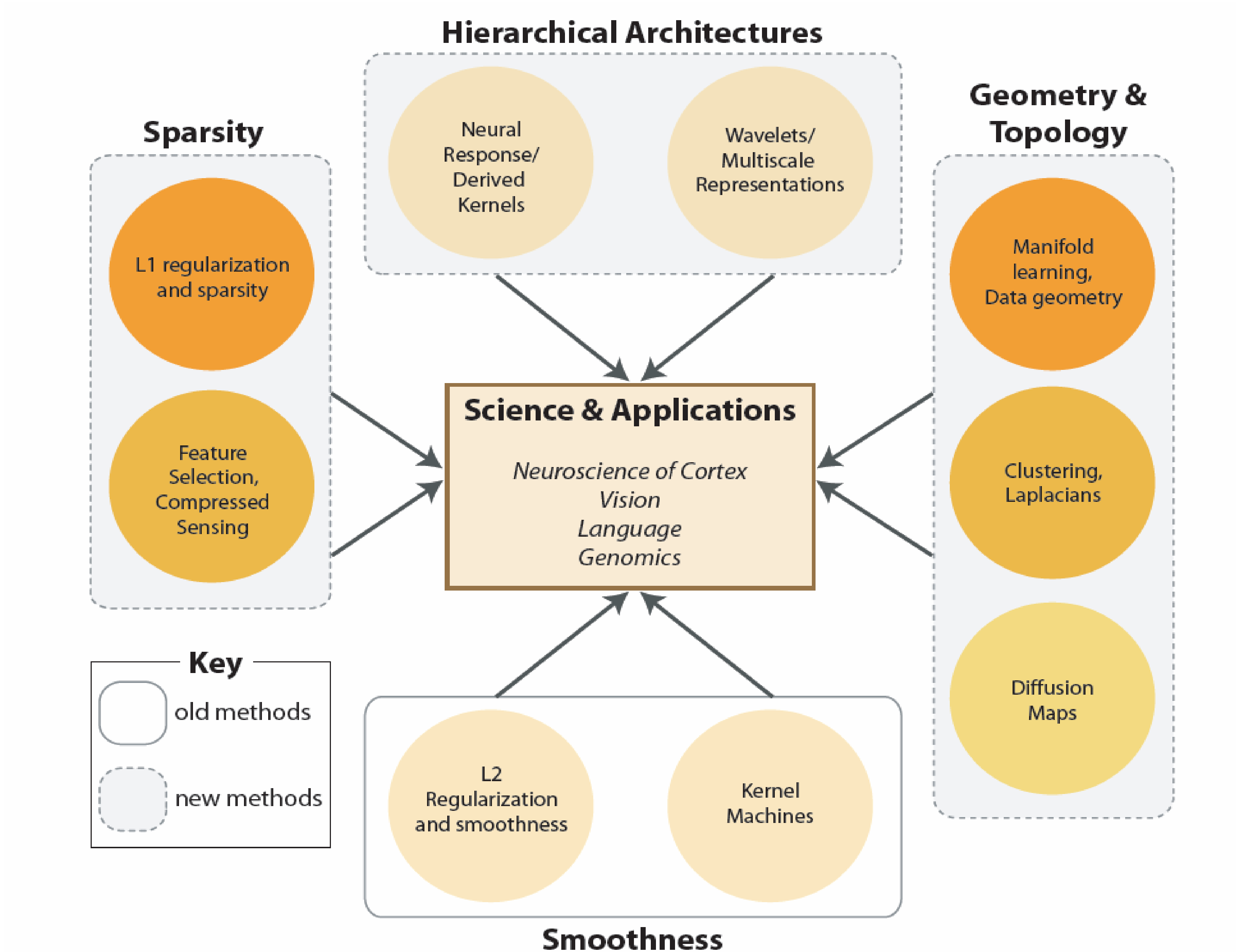
Many processes - physical processes as well as human activities – generate high-dimensional data: *curse of dimensionality or poverty of stimulus*.

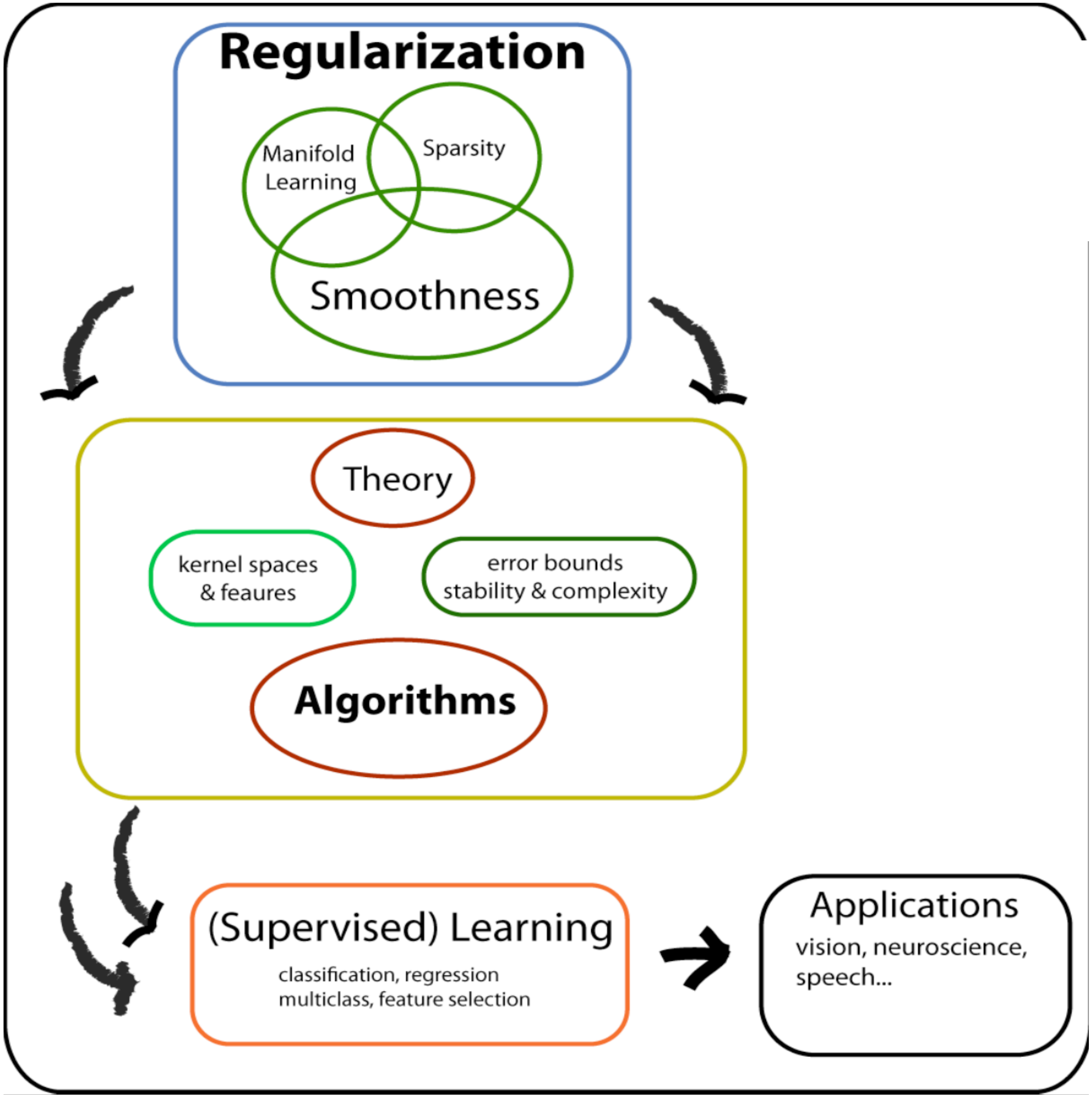
There are, however, basic properties of the data generating process that may allow to circumvent the problem of high dimensionality and make the analysis possible:

- smoothness - exploited by L2 regularization techniques
- sparsity - exploited by L1 regularization techniques
- data geometry - exploited by manifold learning techniques
- hierarchical organization – suggested by the architecture of sensory cortex

$$\min_{f \in H} \left[\frac{1}{\ell} \sum_{i=1}^{\ell} V(f(x_i) - y_i) + \lambda \ pen(f) \right]$$

New Research Directions





Regularization
Manifold Learning
Sparsity
Smoothness
Theory
kernel spaces & feaures
error bounds stability & complexity
Algorithms
(Supervised) Learning
classification, regression multiclass, feature selection
Applications
vision, neuroscience, speech...

Overview

- Supervised learning: real math
- Examples of recent and ongoing in-house engineering applications

Overview of overview

- The problem of supervised learning: "real" math behind it
- Examples of engineering applications (from our group)
- Learning and the brain

Learning from Examples: engineering applications

Computer Vision

- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

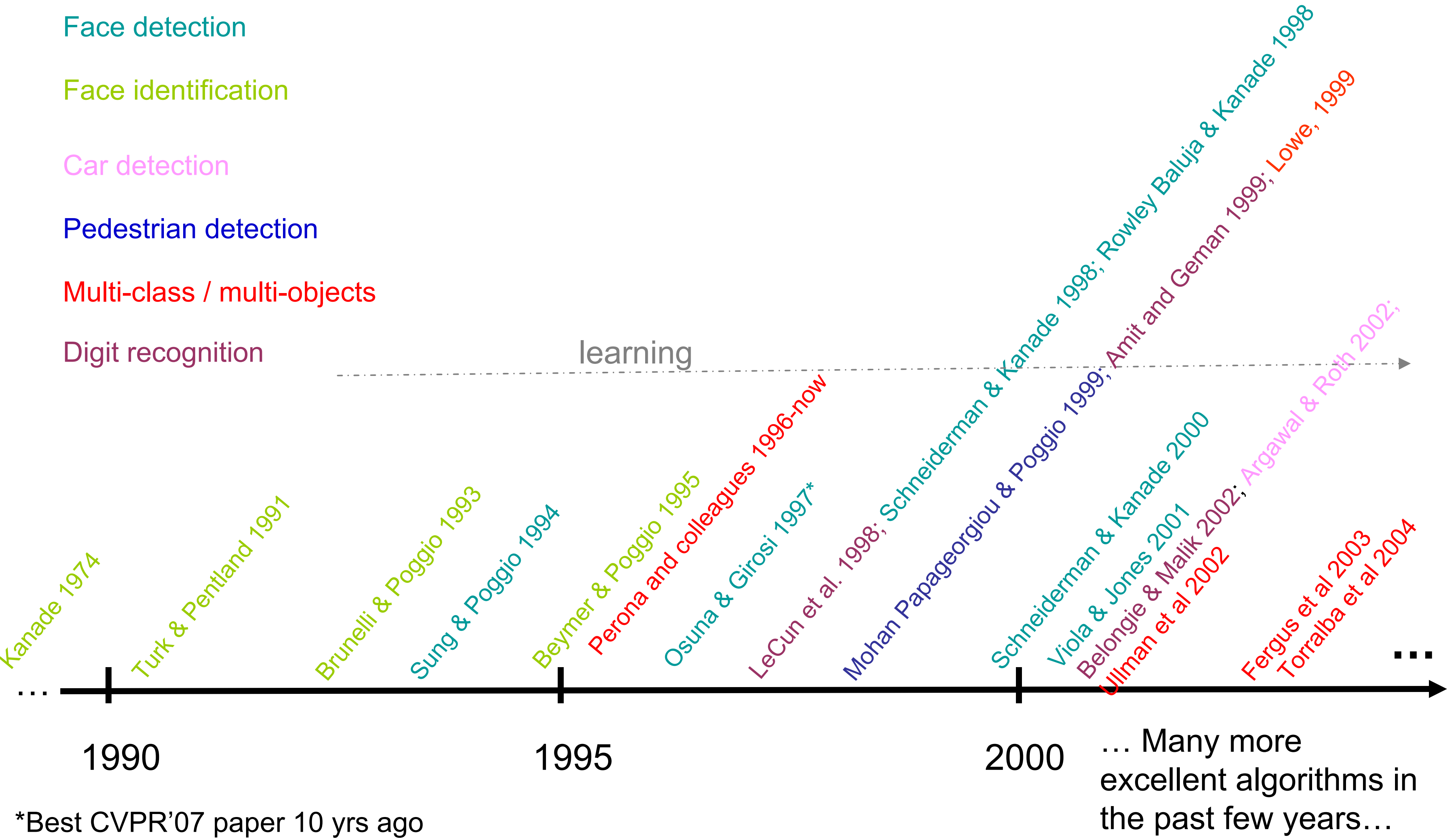
Graphics

Text Classification

Artificial Markets

.....

Object recognition for computer vision: (personal) historical perspective



*Best CVPR'07 paper 10 yrs ago

Learning from Examples: engineering applications

Computer Vision

- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

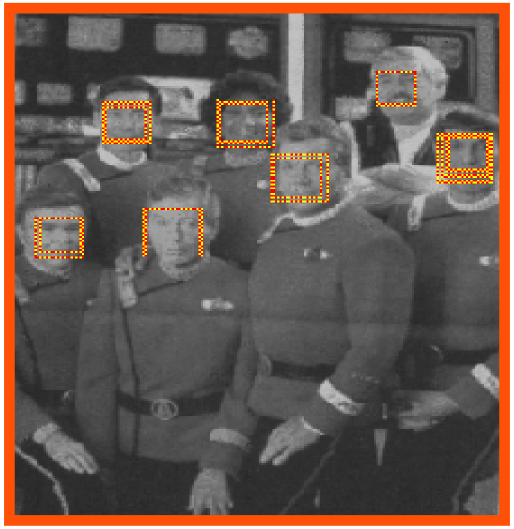
Graphics

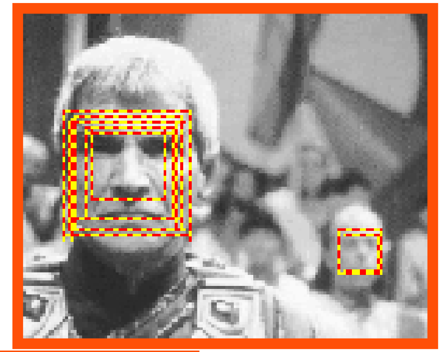
Text Classification

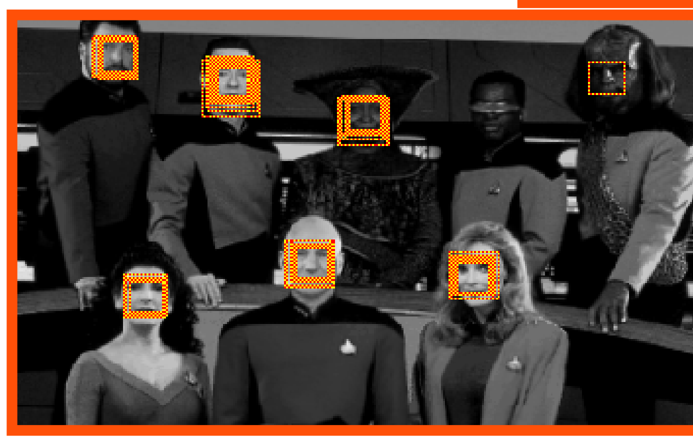
Artificial Markets

.....

Learning Object Detection: Finding Frontal Faces ...



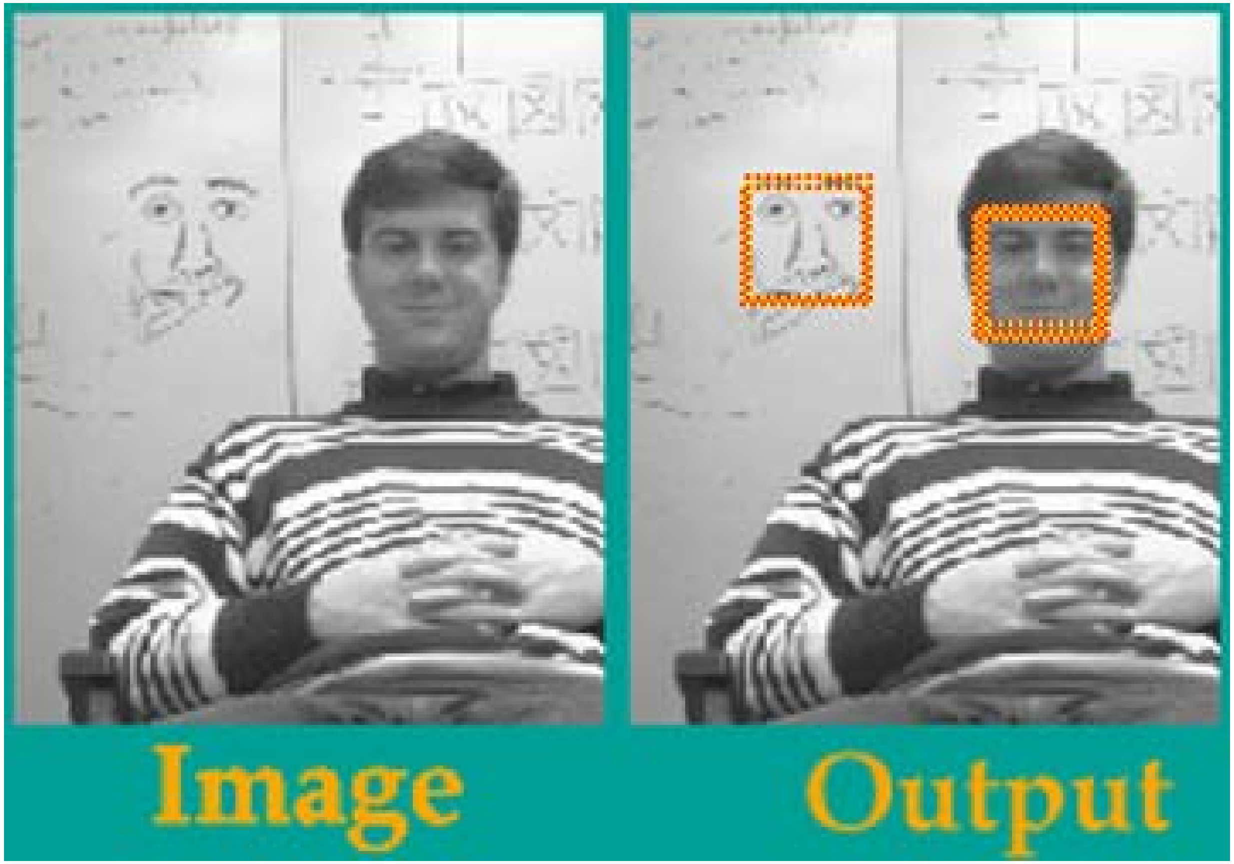




Training Database
1000+ Real, 3000+ *VIRTUAL*
50,0000+ Non-Face Pattern

Sung, Poggio 1995

Learning Face Detection



Sung, Poggio
1994

Face detection:...

Learning from Examples: engineering applications

Computer Vision

- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

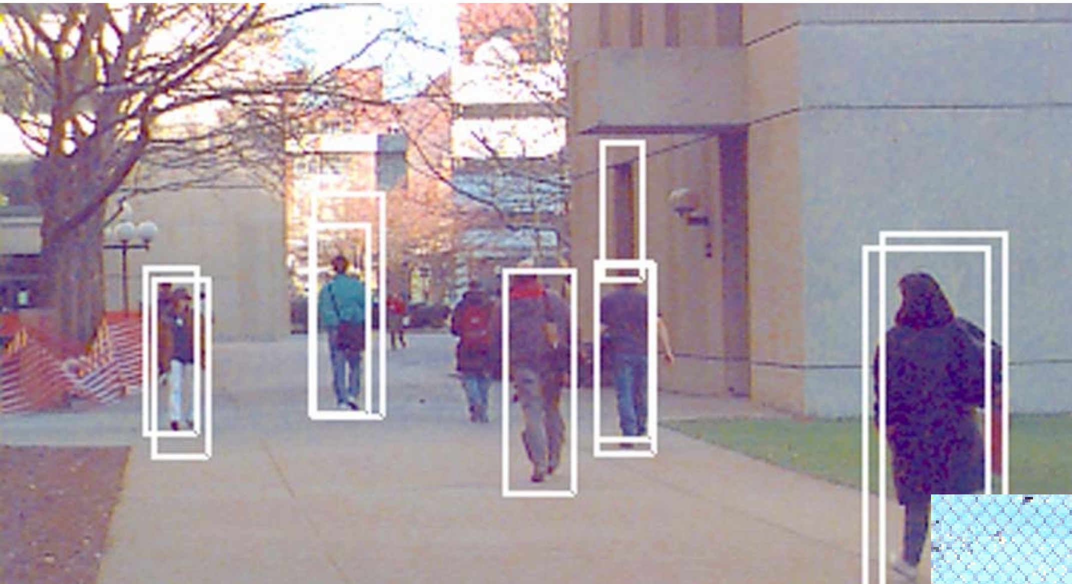
Graphics

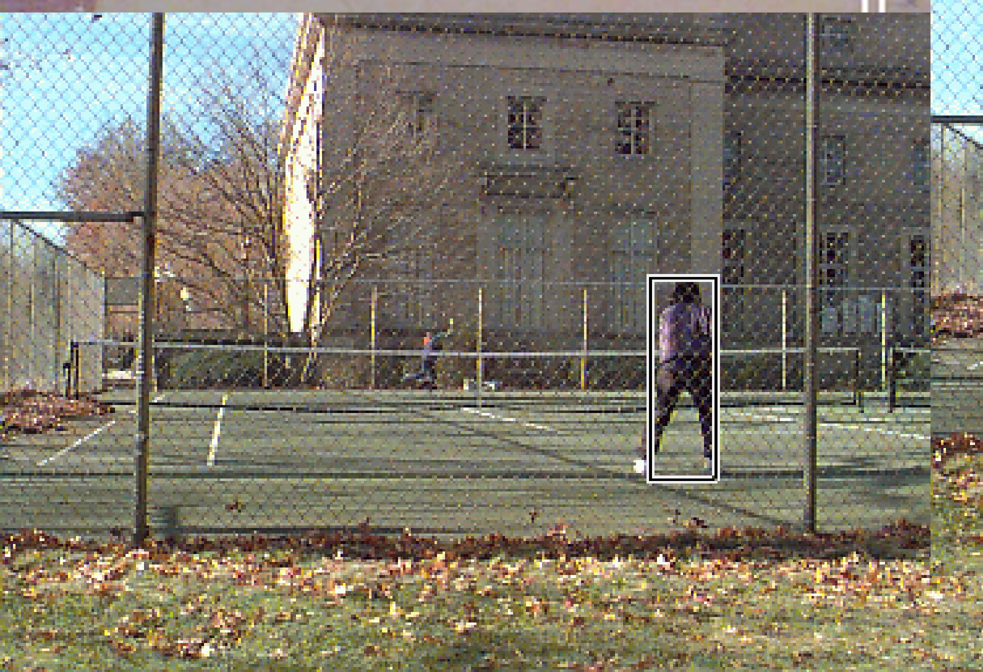
Text Classification

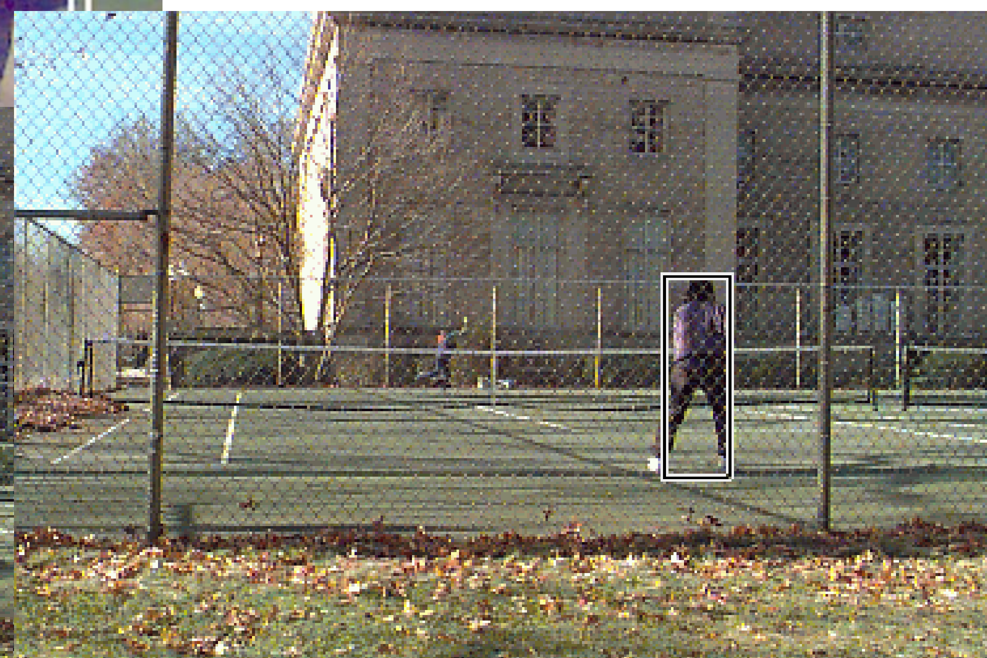
Artificial Markets

.....

Trainable System for Object Detection: Pedestrian detection - Results

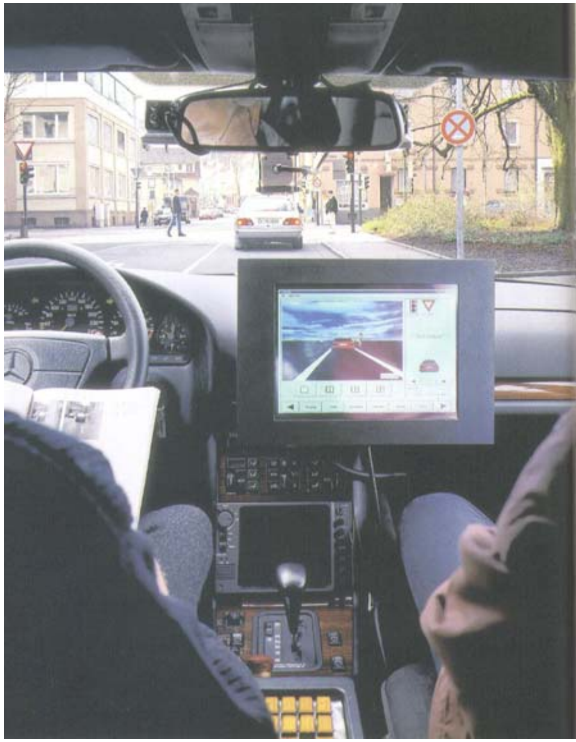


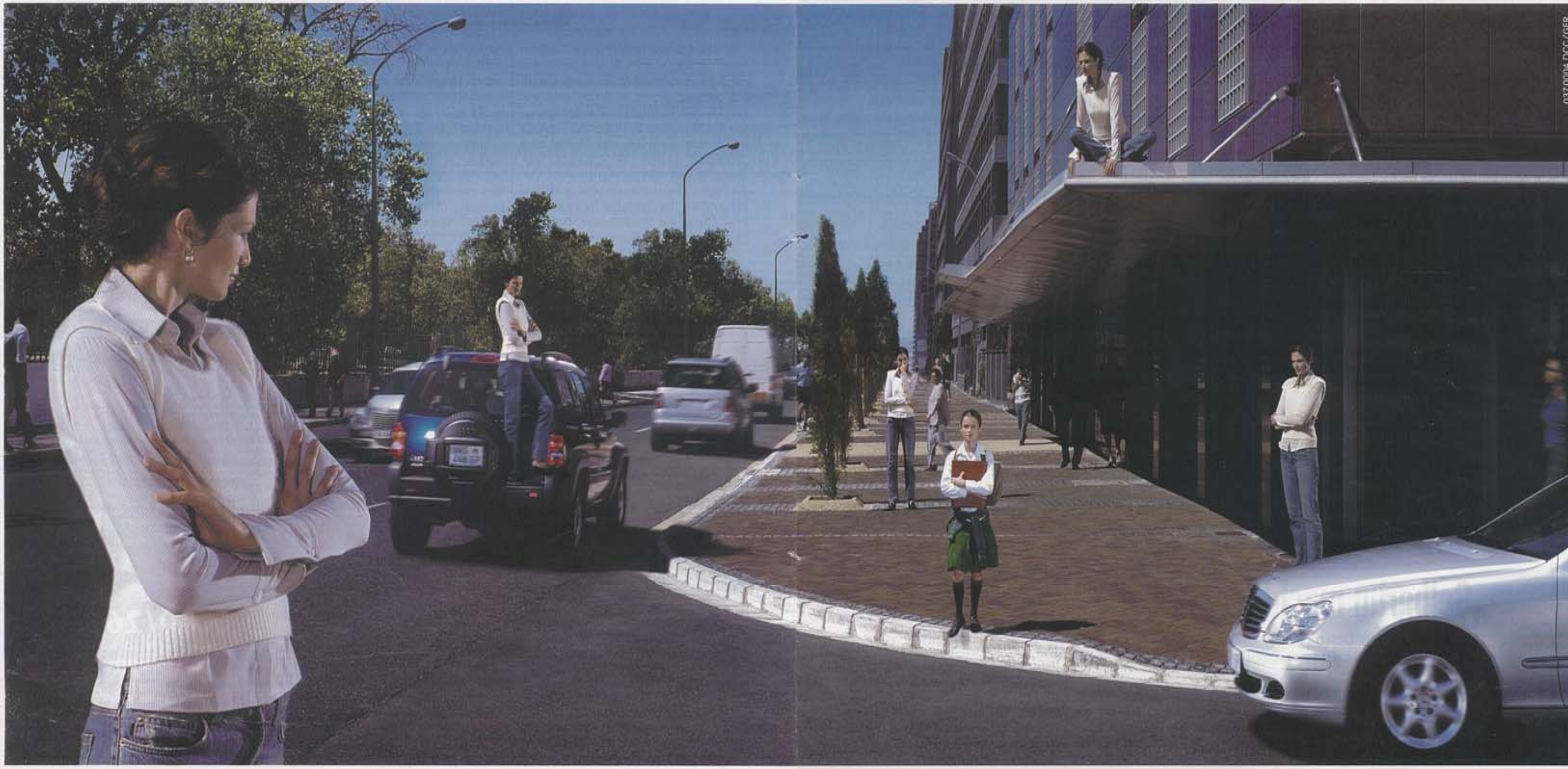




Papageorgiou and Poggio, 1998

We did well with shallow learning architectures (SVMs):
~10 year old CBCL computer vision work:
SVM-based pedestrian detection system in
Mercedes test car...
now becoming a product (MobilEye, Israeli company)





Wir bringen unseren Autos das Sehen bei, weil eine Mutter nicht überall sein kann.

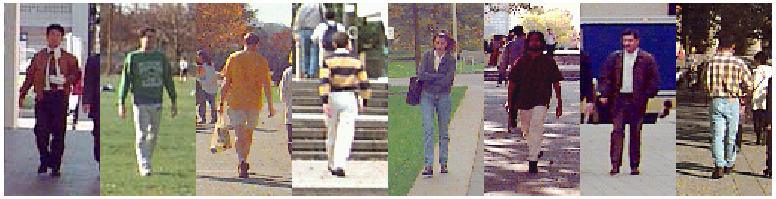
Eine Mutter kann ihre Kinder nicht immer beschützen. Besonders dann nicht, wenn sie alleine im Straßenverkehr unterwegs sind. Deshalb arbeiten wir an Fußgängererkennungs-Systemen für unsere Autos, die dem Fahrer helfen, Menschen auf der Straße schneller zu erkennen. Innerhalb von Bruchteilen einer Sekunde warnt das System den Fahrer, damit er besser reagieren kann. Diese intelligenten Technologien zur Vermeidung von Unfällen entwickelt die DaimlerChrysler Forschung schon heute. Für die Automobile von morgen.

Tiefere Einblicke in die Vision vom ‚Unfallfreien Fahren' erhalten Sie unter: www.daimlerchrysler.com

DaimlerChrysler

Answers for questions to come.

Remark: training set defines task
People classification/detection

 . . .

1848 patterns

 . . .

7189 patterns

Representation: overcomplete dictionary of Haar wavelets; high dimensional feature space (>1300 features)

pedestrian detection

Remark: training set defines task
Face detection

 . . .  . . .

Representation: grey levels (normalized) or overcomplete dictionary of Haar wavelets

face detection

Remark: training set defines task
Face identification

. . .

. . .

Representation: grey levels (normalized) or overcomplete dictionary of Haar wavelets

face identification

Learning from Examples: engineering applications

Computer Vision
- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

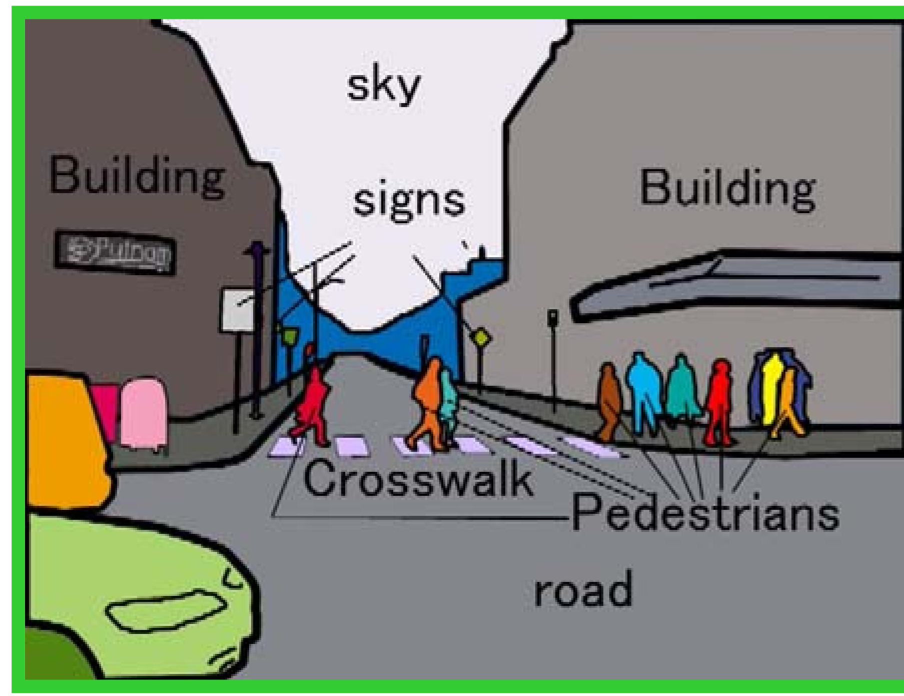
Graphics

Text Classification

Artificial Markets

.....

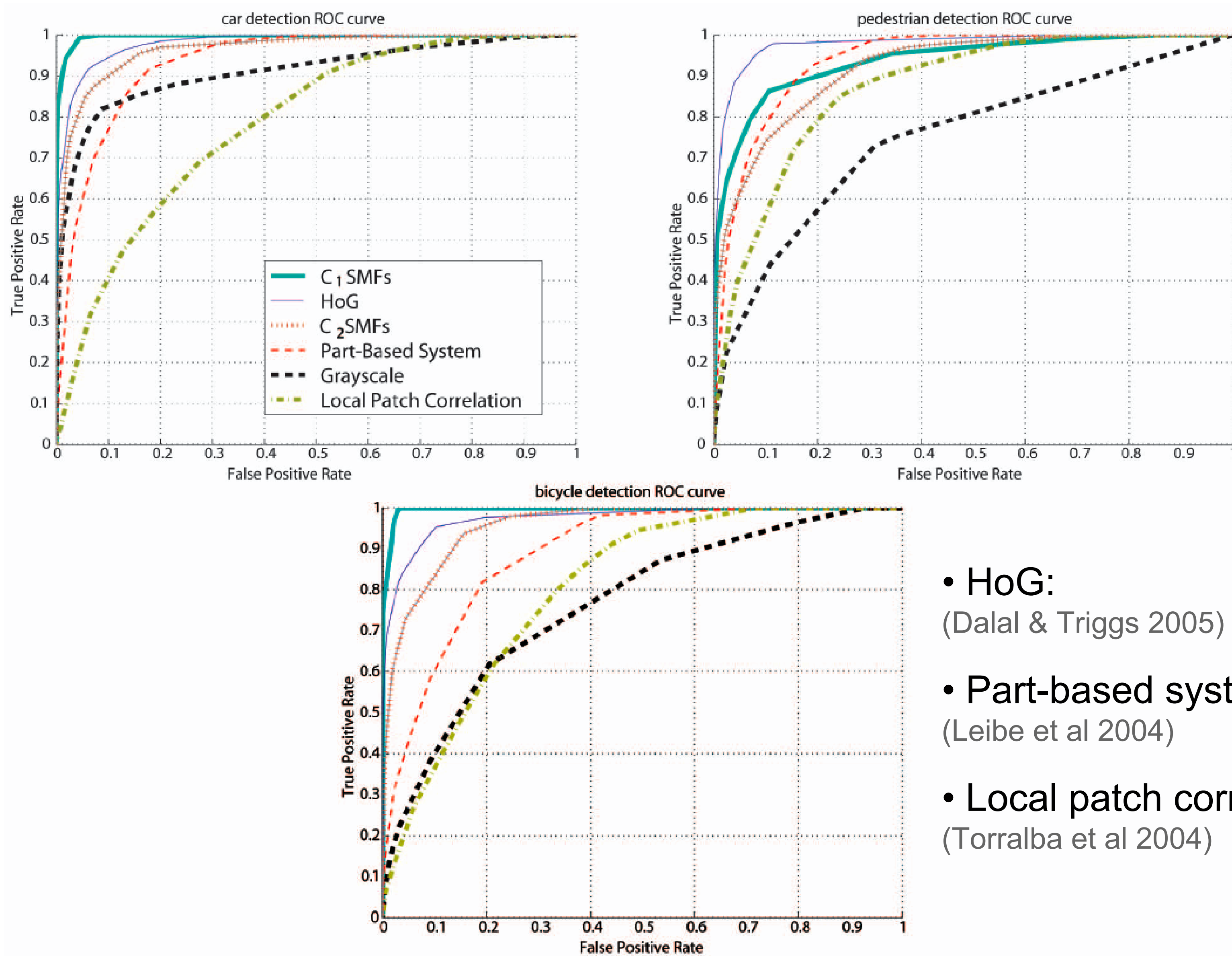
The street scene project



Source: Bileschi, Wolf & Poggio

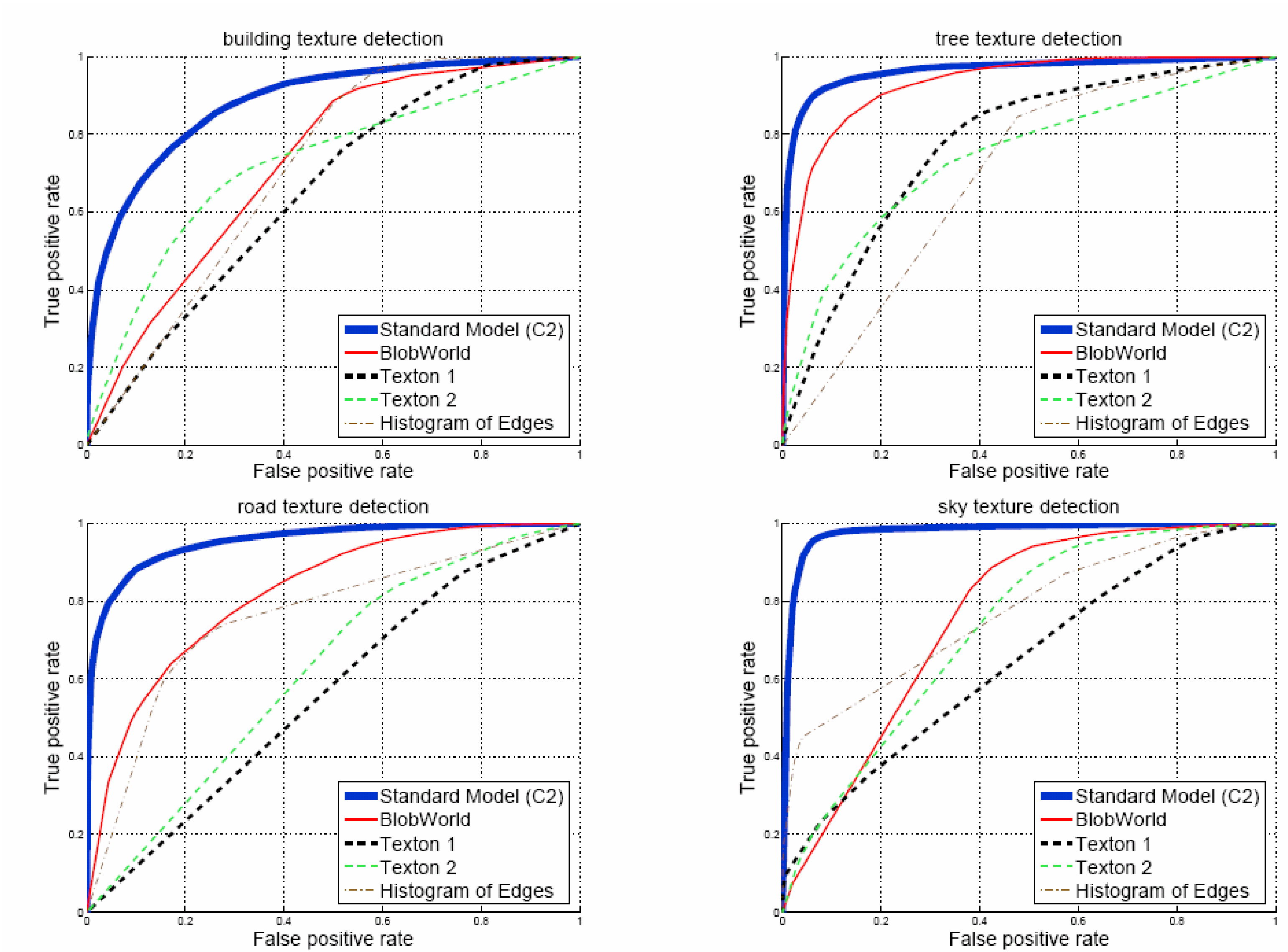
StreetScenes Database. Subjective Results

Results



- HoG:
  (Dalal & Triggs 2005)
- Part-based system:
  (Leibe et al 2004)
- Local patch correlation:
  (Torralba et al 2004)

Serre Wolf Bileschi Riesenhuber & Poggio PAMI 2007



Serre Wolf Bileschi Riesenhuber & Poggio PAMI 2007

Learning from Examples: engineering applications

Computer Vision
- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

Graphics

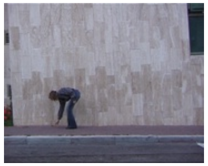
Text Classification

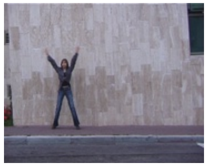
Artificial Markets

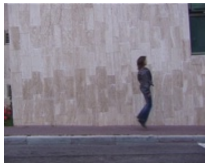
.....

The problem: action recognition

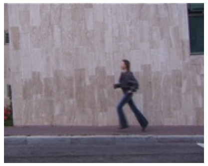
Training Videos

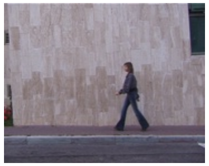

bend

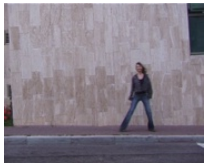

jack

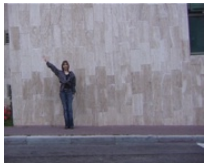

jump

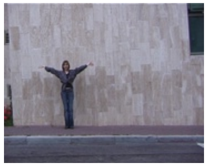

jump


run


walk


side


wave1

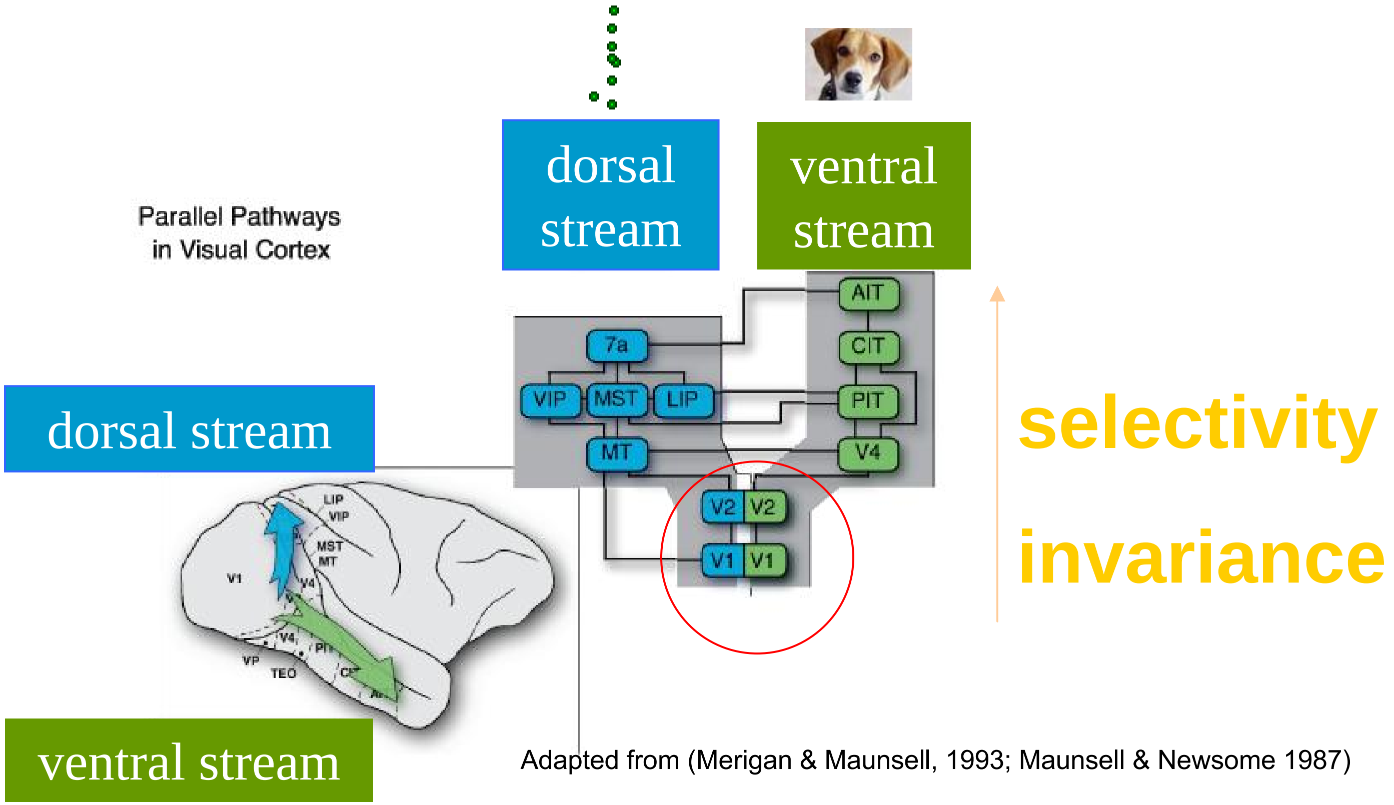

wave2

Testing videos

*each video~4s, 50~100 frames

Dataset from (Blank et al, 2005)

A new model of the dorsal stream (motion) following the ventral stream model



Adapted from (Merigan & Maunsell, 1993; Maunsell & Newsome 1987)

<u>Unsupervised</u> learning in MT (S2) from natural video sequences

Using a large dictionary of MT-like units for action recognition works well!

	(Dolllar et al. 2005)	model	chance
KTH Human	81.3%	**91.6%**	16.7%
UCSD Mice	75.6%	**79.0%**	20.0%
Weiz. Human	86.7%	**96.3%**	11.1%

- Cross-validation: 2/3 training, 1/3 testing, 10 repeats
- Source code for benchmark graciously provided by Piotr Dollar

(Jhuang Serre Wolf & Poggio ICCV 2007)

Learning from Examples: engineering applications

Computer Vision

- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

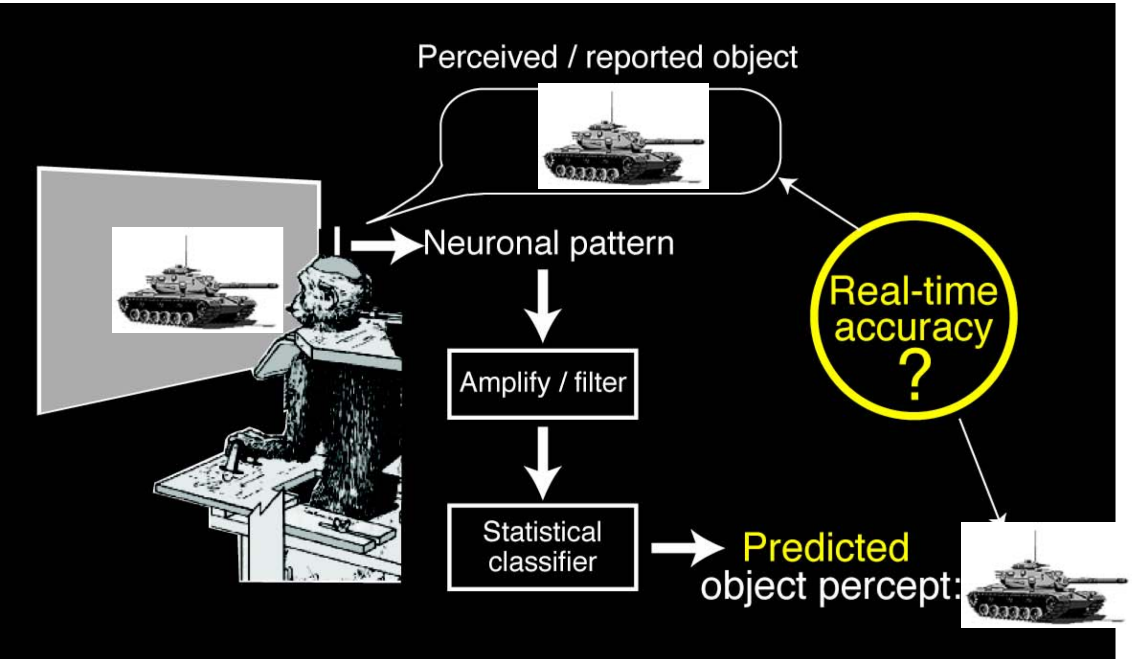
Graphics

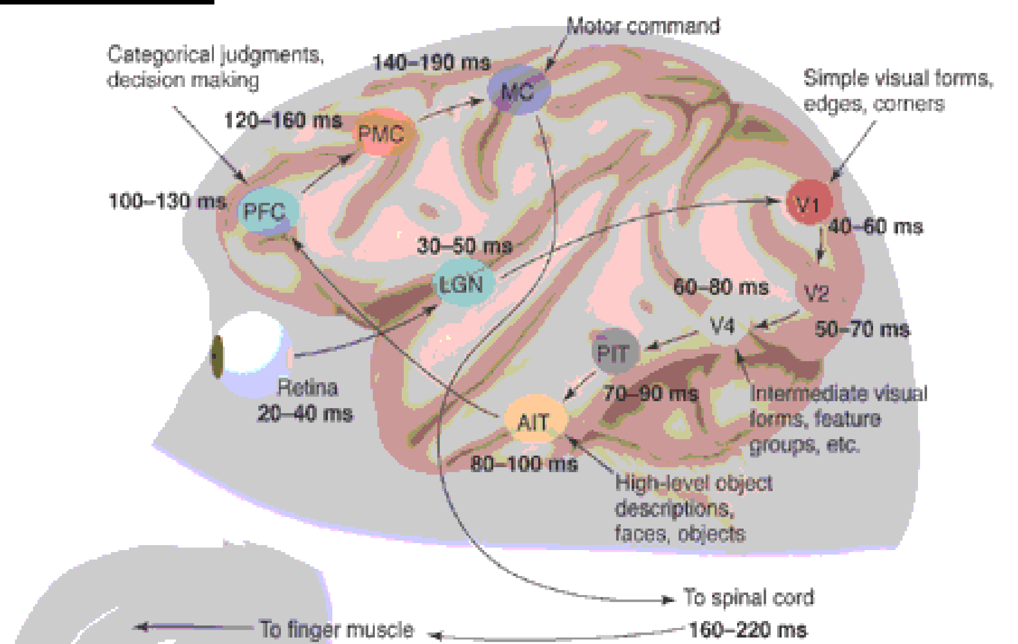
Text Classification

Artificial Markets

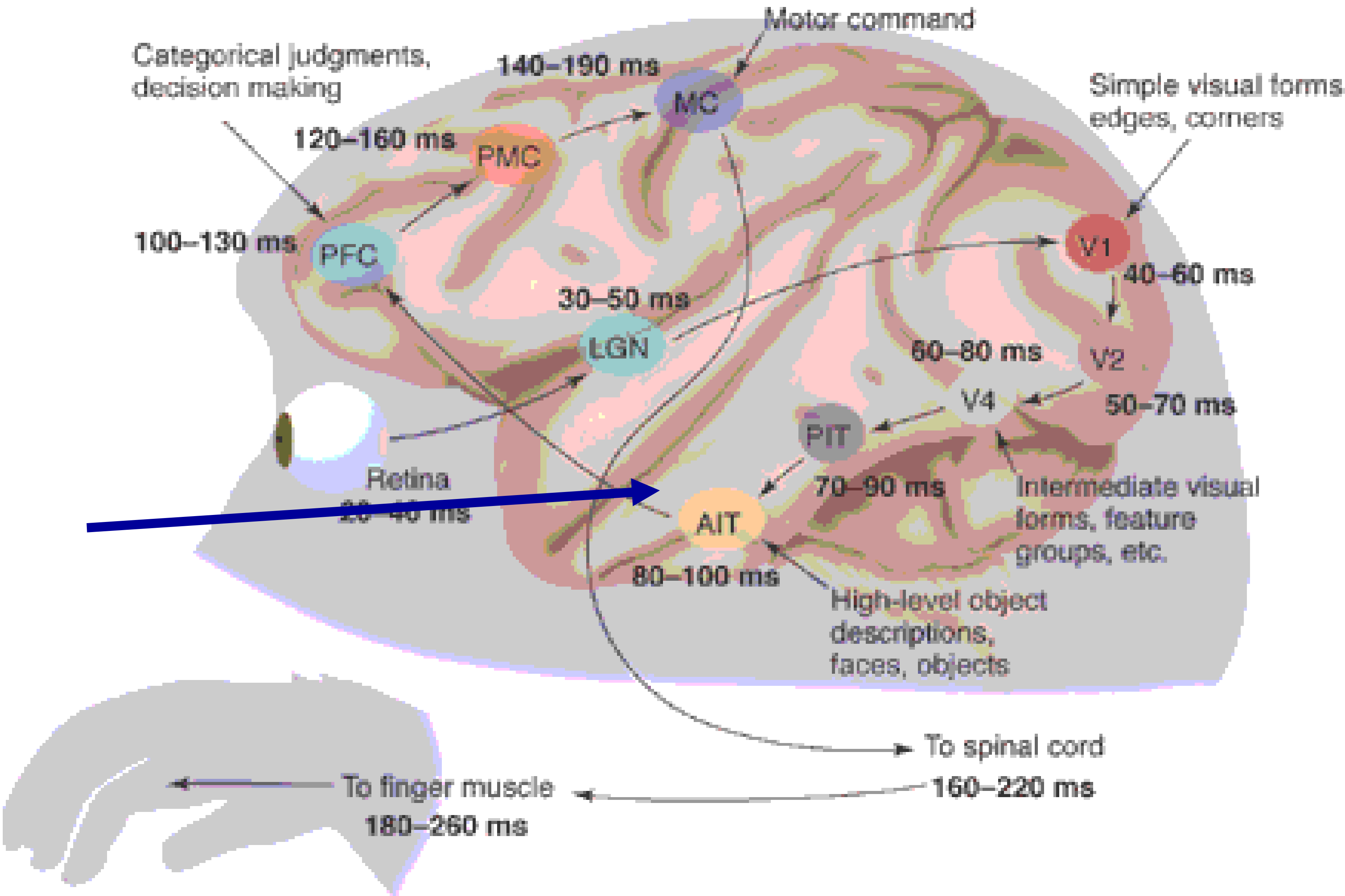
.....

Can we “read-out” from visual cortex what the monkey sees?



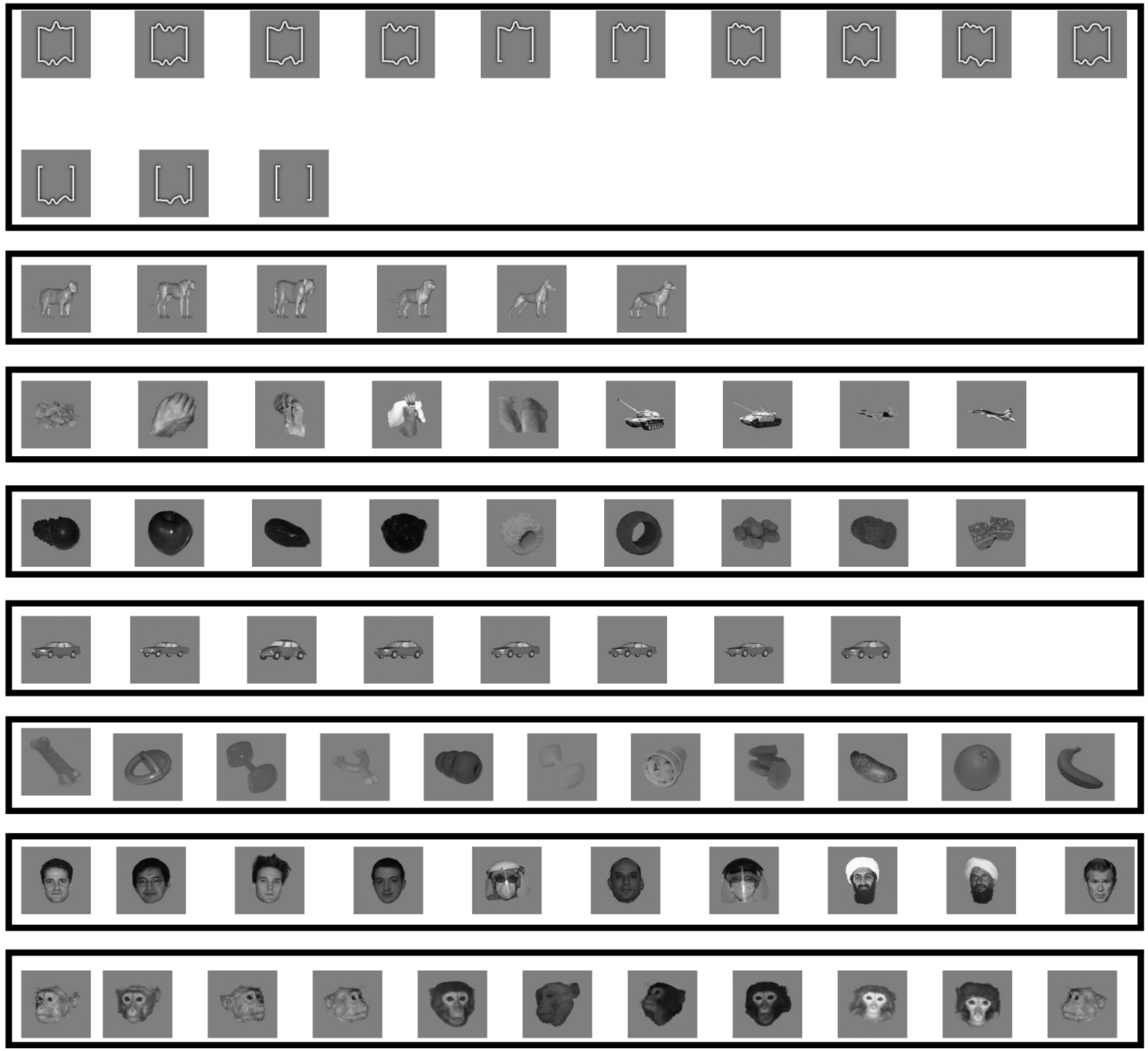


The end station of the ventral stream in visual cortex is IT



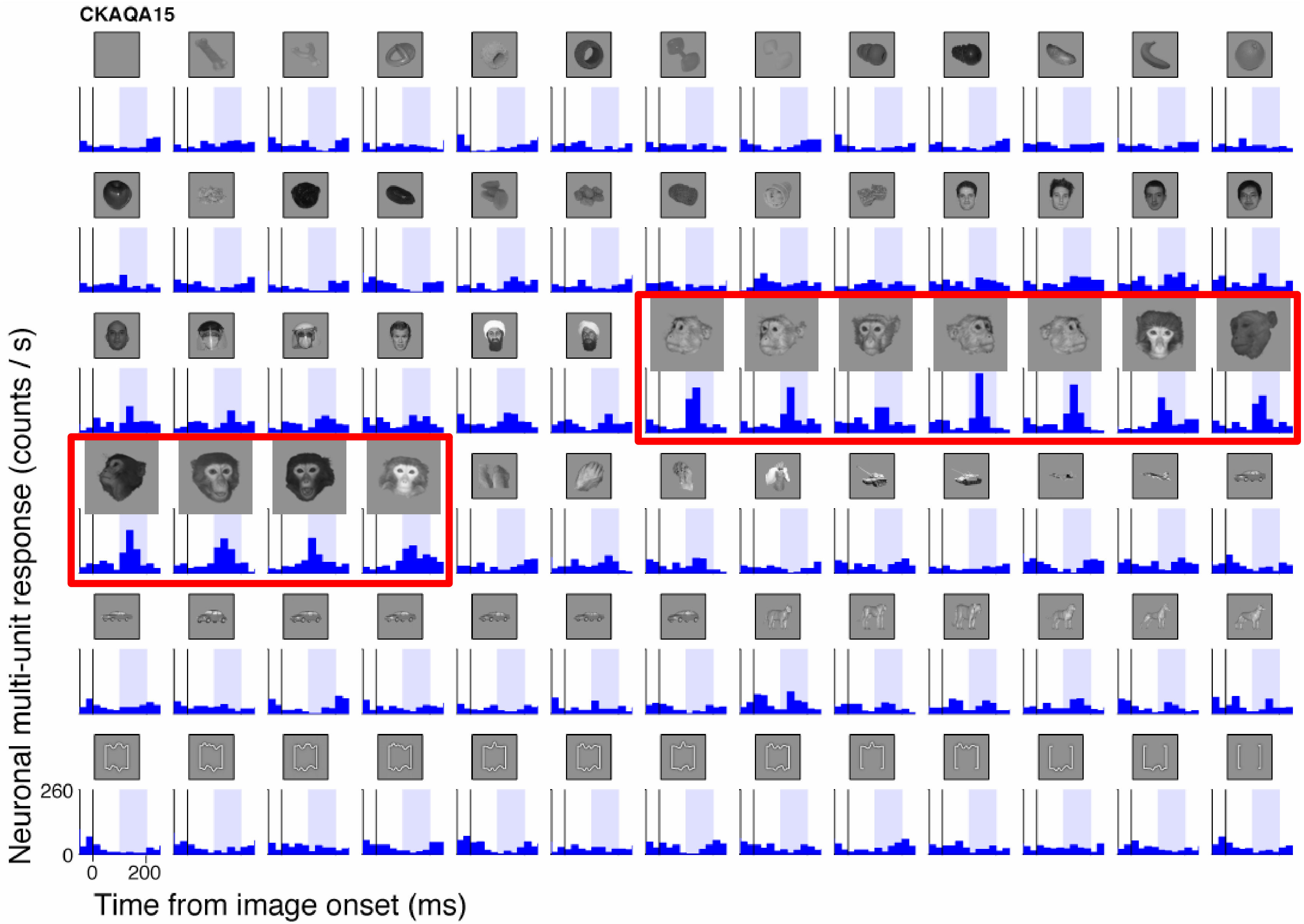
Reading-out the neural code in AIT

77 objects,
8 classes

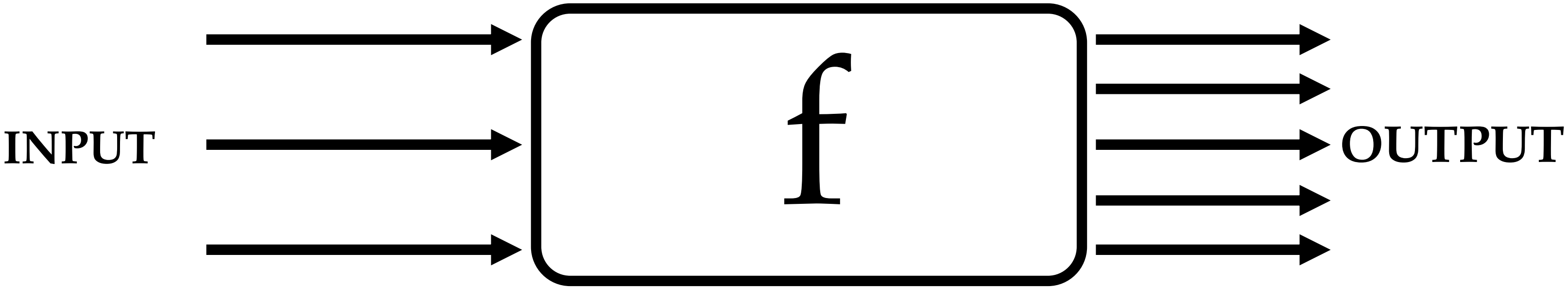


Chou Hung, Gabriel Kreiman, James DiCarlo, Tomaso Poggio, *Science, Nov 4, 2005*

Example of one AIT cell



Training a classifier on neuronal activity.



From a set of data (vectors of activity of n neurons (x) and object label (y)

$$\{(x_1, y_1), (x_2, y_2), \ldots, (x_\ell, y_\ell)\}$$

Find (by training) a classifier eg a function *f* such that $f(x) = \hat{y}$

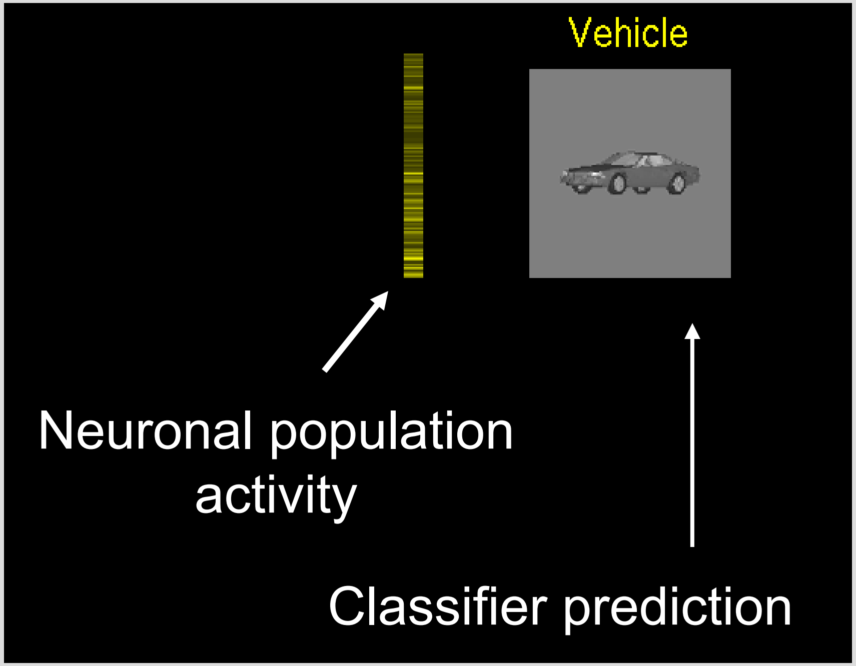
is a ***good predictor*** of object label *y* for a ***future*** neuronal activity *x*

We can decode the brain's code and read-out from neuronal populations:

reliable object categorization (>90% correct) using ~200 <u>arbitrary</u> AIT "neurons"

Video speed: 1 frame/sec

Actual presentation rate: 5 objects/sec



<u>Categorization</u>

- Toy
- Body
- Human Face
- Monkey Face
- Vehicle
- Food
- Box
- Cat/Dog

Hung, Kreiman, Poggio, DiCarlo. *Science 2005*

Learning from Examples: engineering applications

Computer Vision

- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

Graphics

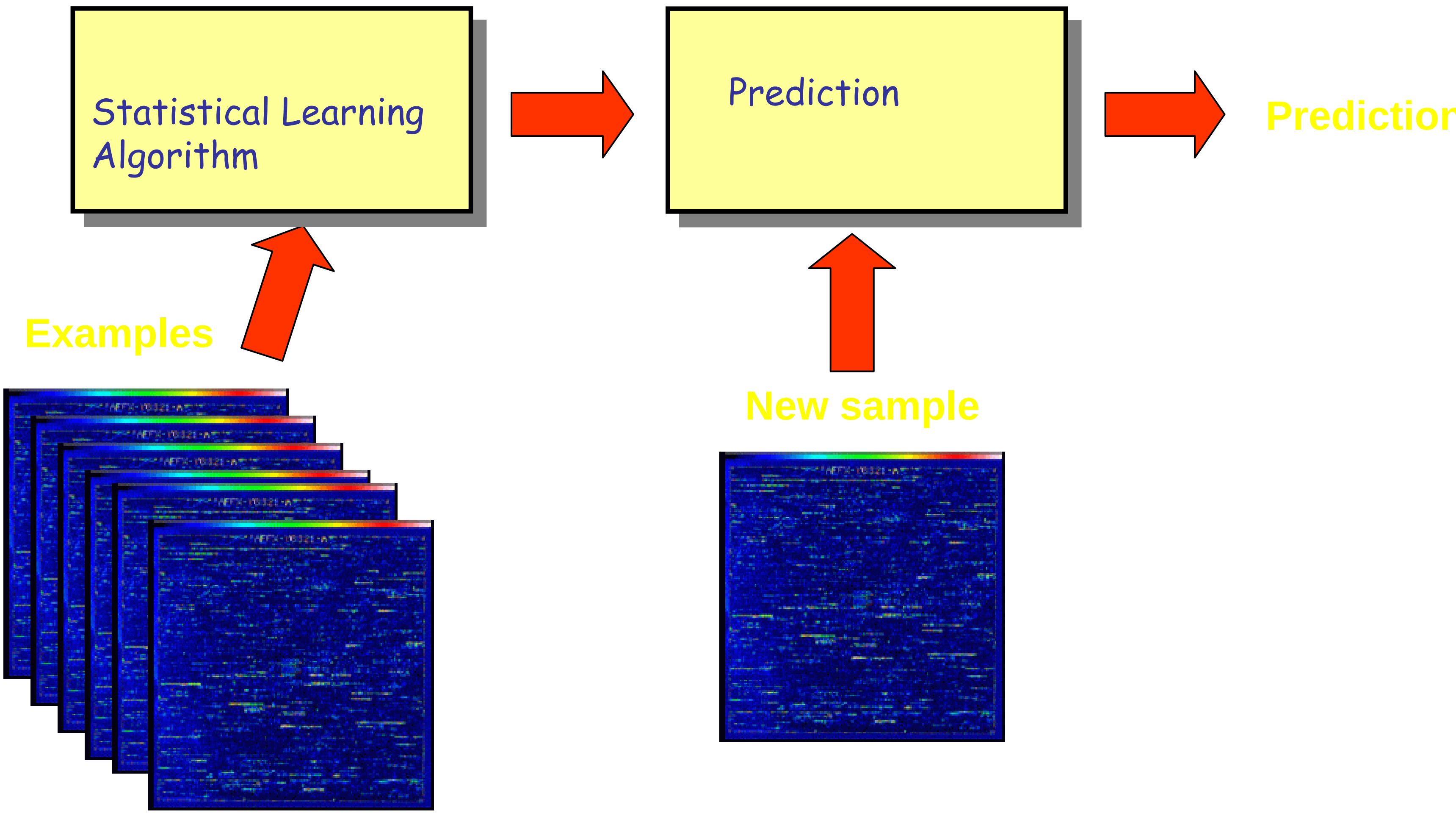
Text Classification

Artificial Markets

.....

Bioinformatics application: predicting type of cancer from DNA chips signals

Learning from examples paradigm



Bioinformatics application: predicting type of cancer from DNA chips

New feature selection SVM:

Only 38 training examples, 7100 features

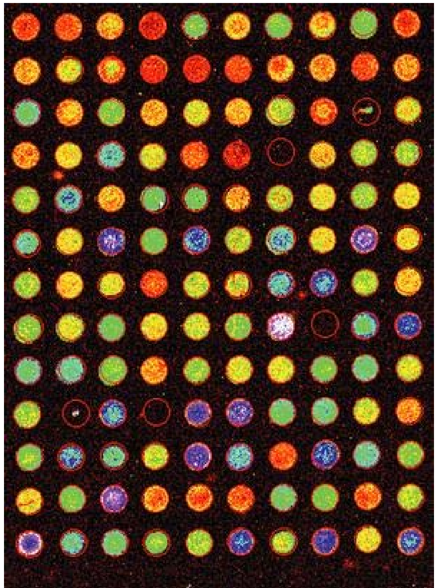
AML vs ALL: 40 genes 34/34 correct, 0 rejects.
5 genes 31/31 correct, 3 rejects of which 1 is an error.

A.I. Memo No.1677
C.B.C.L Paper No.182

Support Vector Machine Classification of Microarray Data

S. Mukherjee, P. Tamayo, D. Slonim, A. Verri, T. Golub, J.P. Mesirov, and T. Poggio

Pomeroy, S.L., P. Tamayo, M. Gaasenbeek, L.M. Sturia, M. Angelo, M.E. McLaughlin, J.Y.H. Kim, L.C. Goumnerova, P.M. Black, C. Lau, J.C. Allen, D. Zagzag, M.M. Olson, T. Curran, C. Wetmore, J.A. Biegel, T. Poggio, S. Mukherjee, R. Rifkin, A. Califano, G. Stolovitzky, D.N. Louis, J.P. Mesirov, E.S. Lander and T.R. Golub. Prediction of Central Nervous System Embryonal Tumour Outcome Based on Gene Expression, *Nature*, 2002.



Learning from Examples: engineering applications

Computer Vision

- Face detection
- Pedestrian detection
- Scene understanding
- Video categorization

Decoding the Neural Code

Bioinformatics

Graphics

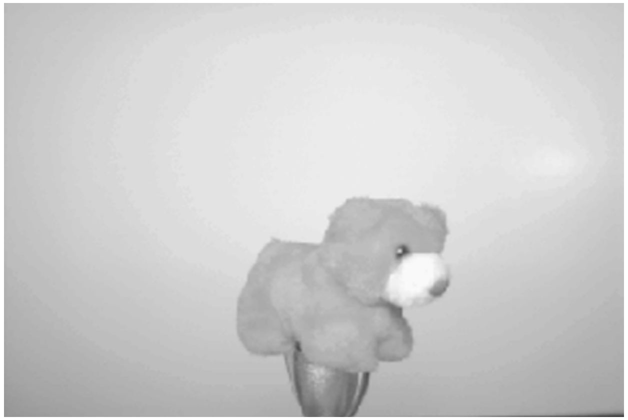
Text Classification

Artificial Markets

.....

Image Analysis

⇒ **Bear (0° view)**



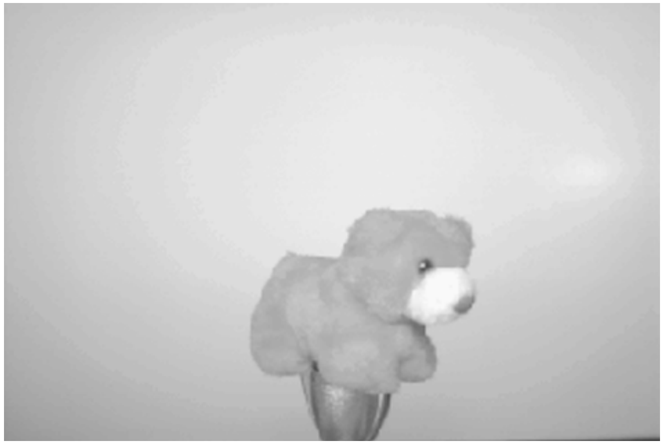
⇒ **Bear (45° view)**

Image Synthesis

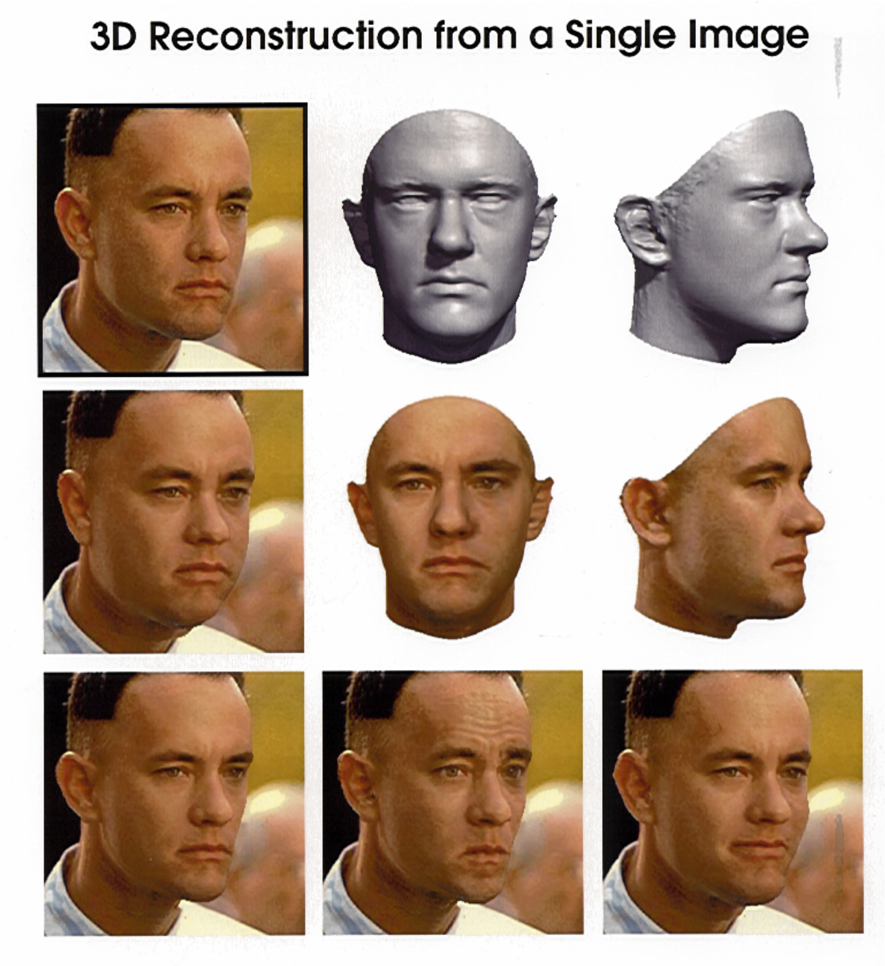
UNCONVENTIONAL GRAPHICS

Θ = 0° view $\Rightarrow$

Θ = 45° view $\Rightarrow$



Reconstructed 3D Face Models from 1 image



Blanz and Vetter,
MPI
SigGraph '99

Reconstructed 3D Face Models from 1 image

Blanz and Vetter,
MPI
SigGraph '99

Vermeer,
Tischbein,
raffaello,
Hopper

V. Blanz, C. Basso,
T. Poggio
and
T. Vetter, 2003

Extending the same basic learning techniques (in 2D): Trainable Videorealistic Face Animation

Ezzat, Geiger, Poggio, SigGraph 2002

Trainable Videorealistic Face Animation

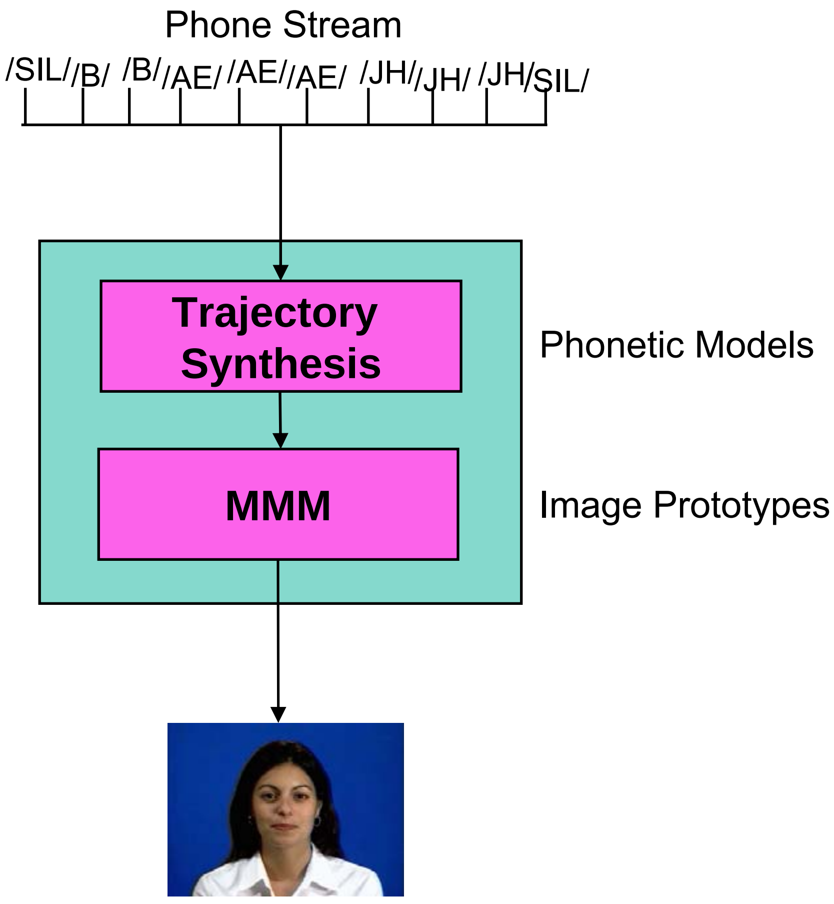
1. Learning

System learns from 4 mins of video the face appearance (Morphable Model) and the speech dynamics of the person

Tony Ezzat, Geiger, Poggio, **SigGraph 2002**

2. Run Time

For any speech input the system provides as output a synthetic video stream



Movies

Marylin,
Rehema

A Turing test: what is real and what is synthetic?

We assessed the realism of the talking face with psychophysical experiments.

Data suggest that the system passes a visual version of the Turing test.

Experiment	# subjects	% correct	t	p<
Single pres.	22	54.3%	1.243	0.3
Fast single pres.	21	52.1%	0.619	0.5
Double pres.	22	46.6%	-0.75	0.5

Table 1: Levels of correct identification of real and synthetic sequences. t represents the value from a standard t-test with significance level of p<.

Overview of overview

- The problem of supervised learning: “real” math behind it

- Examples of engineering applications (from our group)

- Learning and the brain

Beyond classical (shallow) architectures

An additional learning "principle": hierarchical architectures?

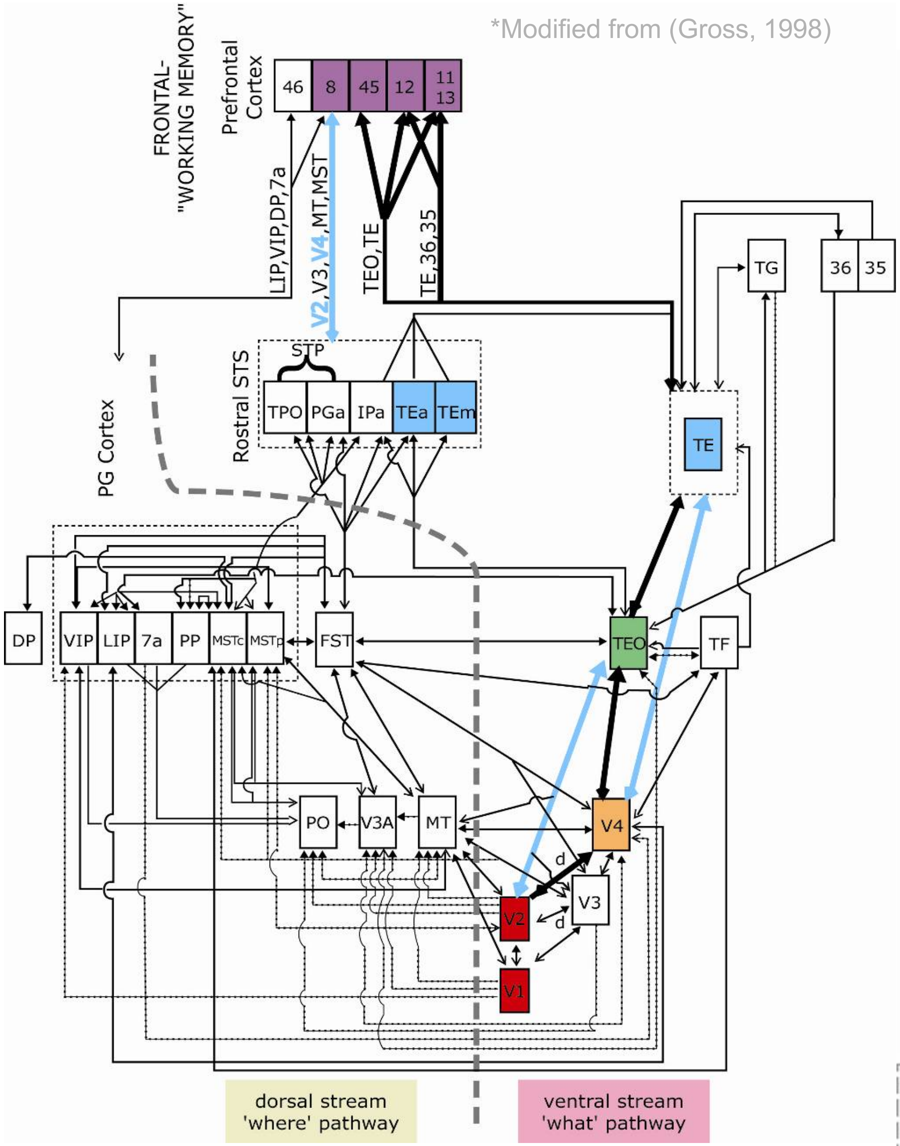
How then do the learning machines described in the theory compare with brains?

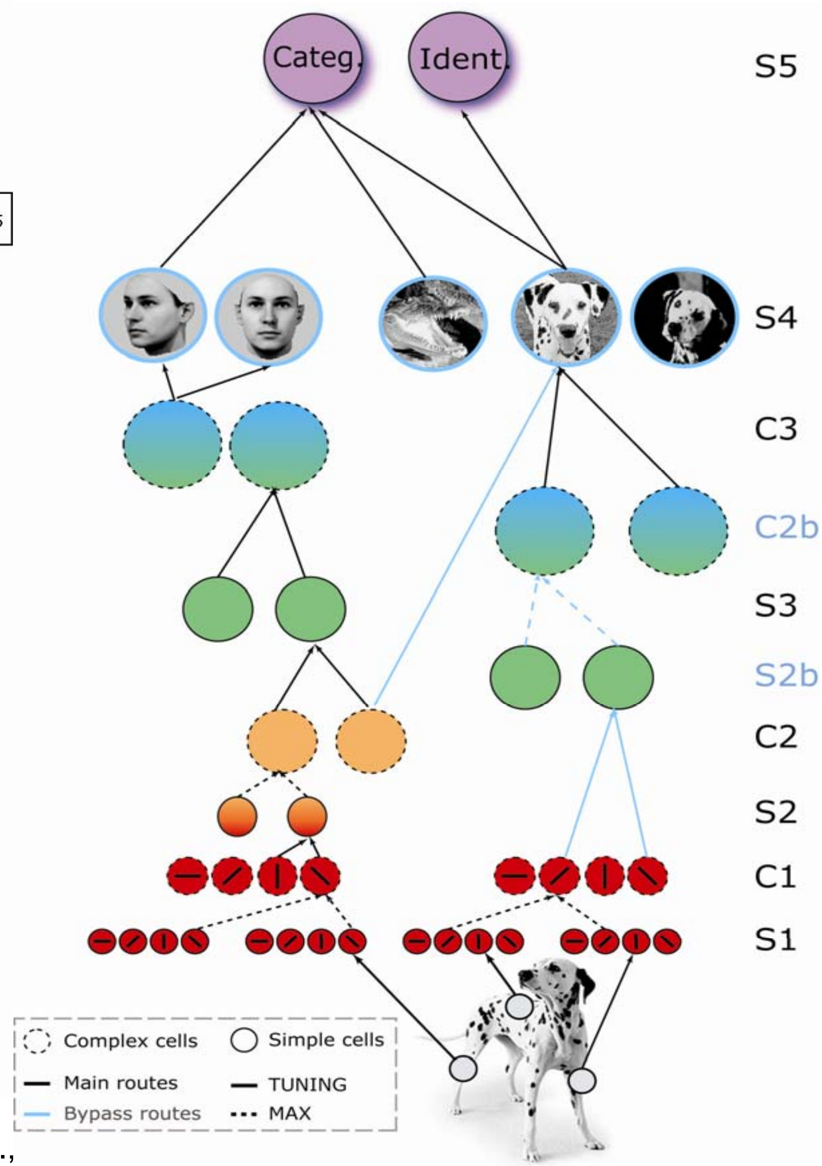
- One of the most obvious differences is the ability of people and animals to learn from very few examples ("poverty of stimulus" problem).

- A comparison with real brains offers another, related, challenge to learning theory. Classical "learning algorithms" correspond to one-layer architectures. The cortex suggests a hierarchical architecture. Thus…are hierarchical architectures with more layers justifiable in terms of statistical learning theory?

Notices of the American Mathematical Society (AMS), Vol. 50, No. 5,
537-544, 2003.
The Mathematics of Learning: Dealing with Data
Tomaso Poggio and Steve Smale

A hierarchical model of the ventral stream, which is also a (unsupervised + supervised) learning algorithm...





Hierarchical model: Riesenhuber & Poggio 1999, 2000; Serre et al., 2005; Serre Oliva Poggio 2007…following previous ideas/work by Hubel and Wiesel, Fukushima, et al.

[software available online]

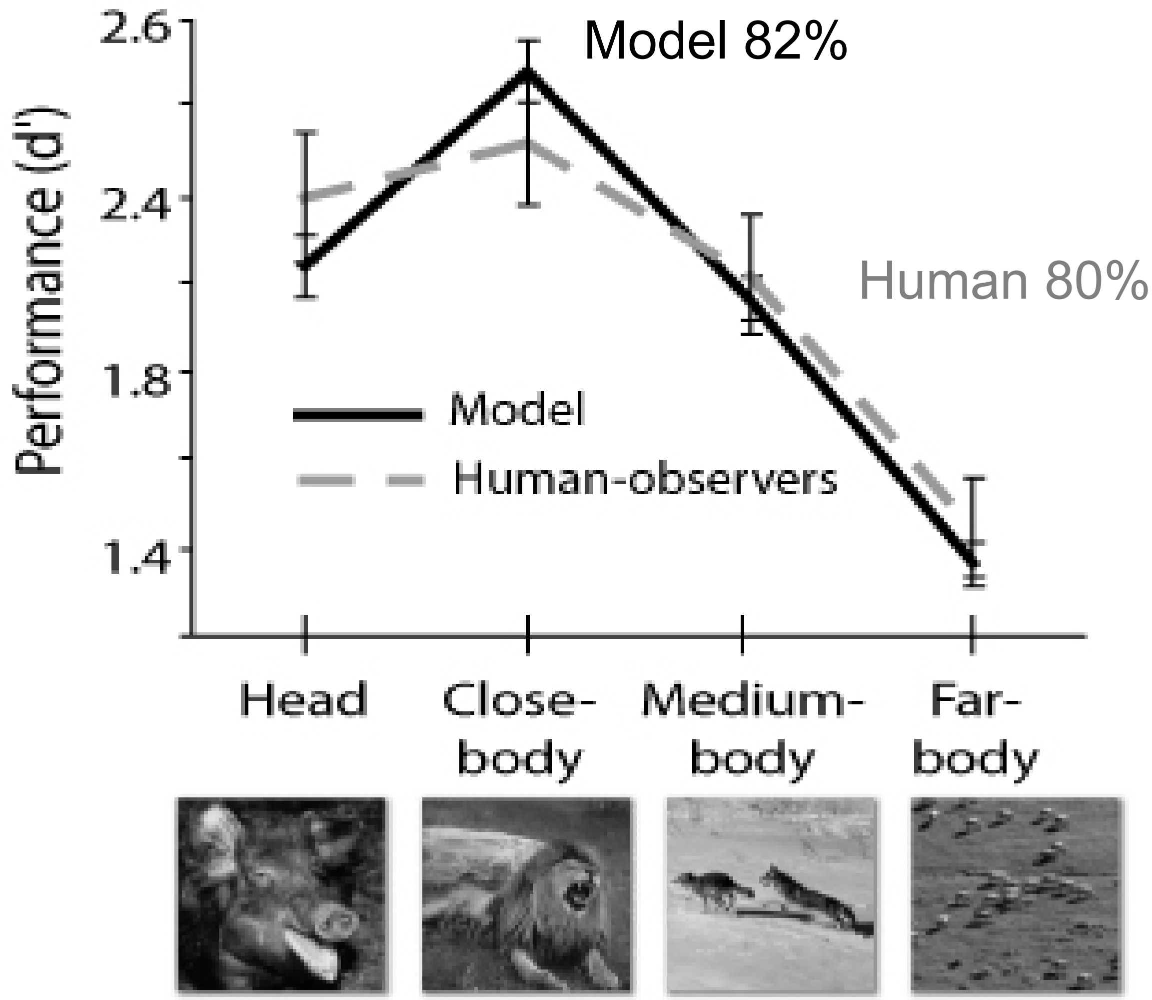
...predicts and is consistent with neural data...

- V1:
  - Simple and complex cells tuning (Schiller et al 1976; Hubel & Wiesel 1965; Devalois et al 1982)
  - MAX operation in subset of complex cells (Lampl et al 2004)
- V4:
  - Tuning for two-bar stimuli (Reynolds Chelazzi & Desimone 1999)
  - MAX operation (Gawne et al 2002)
  - Two-spot interaction (Freiwald et al 2005)
  - Tuning for boundary conformation (Pasupathy & Connor 2001, Cadieu et al., 2007)
  - Tuning for Cartesian and non-Cartesian gratings (Gallant et al 1996)
- IT:
  - Tuning and invariance properties (Logothetis et al 1995)
  - Differential role of IT and PFC in categorization (Freedman et al 2001, 2002, 2003)
  - Read out data (Hung Kreiman Poggio & DiCarlo 2005)
  - Pseudo-average effect in IT (Zoccolan Cox & DiCarlo 2005; Zoccolan Kouh Poggio & DiCarlo 2007)
- Human:
  - Rapid categorization (Serre Oliva Poggio 2007)
  - Face processing (fMRI + psychophysics) (Riesenhuber et al 2004; Jiang et al 2006)

(Serre Kouh Cadieu Knoblich Kreiman & Poggio 2005)

...mimics human recognition performance in rapid categorization (and does well as on computer vision benchmarks)

- d'~ standardized error rate
- the higher the d', the better the perf.



Serre Oliva & Poggio 2007

Thus from neuroscience a challenge for “classical” learning theory:

an unusual, hierarchical architecture
with unsupervised and supervised learning…

…but we need a theory -- not just a model!

Derived Kernels

joint work with J. Bouvrie (MIT), A. Caponnetto (CityU), L. Rosasco (MIT and Genoa), S. Smale (TTI-C)

January 24, 2009

It is just possible that the brain ....

…will tell us more on learning theory!

Overview of overview

- The problem of supervised learning: “real” math behind it

- Examples of engineering applications (from our group)

- Learning and the brain